NOVA SCHOOL OF SCIENCE & TECHNOLOGY

DEPARTMENT OF MATHEMATICS

INÊS LANÇA DE OLIVEIRA

Master in Physics Engineering

SPATIAL ANALYSIS OF STREET CRIME IN URBAN AREAS UNDER THE JURISDICTION OF THE GUARDA NACIONAL REPUBLICANA

THE CASE STUDY OF THE CITY OF ALMADA (2022-2023)

MASTER IN MATHEMATICS AND APPLICATIONS
SPECIALIZATION IN DATA SCIENCE AND DECISION THEORY

NOVA University Lisbon
November, 2025

NOVA SCHOOL OF SCIENCE & TECHNOLOGY

DEPARTMENT OF MATHEMATICS

SPATIAL ANALYSIS OF STREET CRIME IN URBAN AREAS UNDER THE JURISDICTION OF THE GUARDA NACIONAL REPUBLICANA

THE CASE STUDY OF THE CITY OF ALMADA (2022-2023)

INÊS LANÇA DE OLIVEIRA

Master in Physics Engineering

Co-advisers: Dr.ª Isabel Natário
Professora Associada, Faculdade de Ciências e Tecnologia da Universidade NOVA de Lisboa
Dr.ª Paula Simões
Professora Coordenadora, Instituto Superior de Engenharia de Lisboa

Examination Committee

Chair: Dr. Filipe Marques
Professor Associado com Agregação, Faculdade de Ciências e Tecnologia da Universidade NOVA de Lisboa
Rapporteur: Dr. Miguel Fonseca
Professor Auxiliar, Faculdade de Ciências e Tecnologia da Universidade NOVA de Lisboa
Co-adviser: Dr.ª Isabel Natário
Professora Associada, Faculdade de Ciências e Tecnologia da Universidade NOVA de Lisboa

MASTER IN MATHEMATICS AND APPLICATIONS
SPECIALIZATION IN DATA SCIENCE AND DECISION THEORY

NOVA University Lisbon
November, 2025

Spatial Analysis of Street Crime in Urban Areas Under the Jurisdiction of the Guarda Nacional Republicana

The case study of the city of Almada (2022-2023)

This document was created with the (pdf/Xe/Lua)LaTeX processor and the NOVAthesis template (v7.2.5) [1].

Acknowledgements

Em primeiro lugar, gostaria de agradecer às minhas orientadoras de tese, a Professora Doutora Isabel Natário e a Professora Doutora Paula Simões, pela mentoria, disponibilidade e paciência, que são indispensáveis para que um trabalho de mestrado chegue a bom porto.

Em segundo lugar, agradeço à Guarda Nacional Republicana (GNR), e em particular à Direção de Informações da GNR, por ter fornecido os dados para este projeto e prestado todos os esclarecimentos solicitados. Devo também um agradecimento ao Centro de Investigação, Desenvolvimento e Inovação da Academia Militar (CINAMIL), sem o qual este tema não existiria, visto que possibilitou a apresentação do projeto à NOVA-SST. Ainda dentro da esfera militar, gostaria de agradecer de forma particular ao Aspirante Aluno GNR Duarte Branco, do curso de Ciências Militares com especialização em Segurança, por ter servido de ponte entre o mundo académico e militar (tal como a Professora Paula!), e pela sua dissertação, cuja defesa elucidou bastante sobre a perceção da GNR relativamente à realidade do crime em Almada.

Quero ainda agradecer às minhas colegas de curso, em especial à minha homónima Inês e à Dulce, pela amizade e companhia durante as sessões de estudo assim como pelos momentos de convívio, tornando o ano de dissertação de mestrado menos isolante, mais produtivo e, sobretudo, mais feliz.

Por fim, não podia deixar de agradecer à minha família, em especial à minha mãe, padrasto, irmã, gata e amigos mais próximos, pelo apoio emocional constante que me proporcionam.

Abstract

Crime can exhibit strong spatial concentration, with a small number of *hotspots* accounting for a significant share of events, as described by Weisburd's law of crime concentration. This study investigates the spatial distribution of street crime in Almada, Portugal, using 2022–2023 data from the Guarda Nacional Republicana (GNR). The aim is to identify patterns that can inform Intelligence-Led Policing strategies and contribute to a better understanding of microgeographic crime patterns in Almada. We employ spatial point pattern analysis and model crime as a Log-Gaussian Cox Process (LGCP), suitable for clustered events. Inference is conducted using a Bayesian framework via the Integrated Nested Laplace Approximation (INLA) and the Stochastic Partial Differential Equation (SPDE) approach, allowing inclusion of covariates and structured random spatial effects.

Exploratory spatio-temporal non-parametric analysis reveals concentrated hotspots in the north-east, along the *Costa da Caparica* coastline, and the north-west corner of *Caparica e Trafaria*, with significant clustering up to 1.25 km. LGCP models incorporating socio-economic and urban covariates derived from census data indicate that urban density, male-dominated populations, and unemployment increase crime risk, while older populations and immigrants are protective. Covariate inclusion partially explains spatial clustering, though residual analysis and model selection criteria suggest model specification could be improved, namely with richer covariates such as points of interest.

Keywords: INLA, Log-Gaussian Cox Processes, Spatial Statistics, SPDE, Street Crime

Resumo

O crime pode apresentar uma forte concentração espacial, com um pequeno número de *hotspots* responsável por uma parte significativa dos eventos, conforme descrito pela lei da concentração do crime de Weisburd. Este estudo investiga a distribuição espacial do crime de rua em Almada, Portugal, utilizando dados de 2022–2023 fornecidos pela Guarda Nacional Republicana (GNR). O objetivo é identificar padrões que possam informar estratégias de Policiamento Orientadas pela Inteligência e contribuir para uma melhor compreensão dos padrões criminais ao nível microgeográfico em Almada. É utilizada a análise de padrões pontuais espaciais e modelado o crime como um Processo de Cox Log-Gaussiano (LGCP), adequado para padrões pontuais agregados. A inferência é realizada num enquadramento Bayesiano, usando a metodologia INLA (*Integrated Nested Laplace Approximation*) combinada com a abordagem SPDE (*Stochastic Partial Differential Equation*) baseada em equações diferenciais parciais estocásticas, permitindo a inclusão de covariáveis e efeitos aleatórios espaciais estruturados.

A análise exploratória não paramétrica espaço-temporal revela *hotspots* na região Nordeste, ao longo da Costa da Caparica, e a Noroeste da região da Caparica e Trafaria, com correlação entre eventos significativa até 1, 25 km. Os modelos LGCP que incorporam covariáveis socioeconómicas e urbanísticas, derivadas de dados dos Census, indicam que a densidade urbana, populações dominadas por indivíduos do sexo masculino e o indivíduos desempregados aumentam o risco de criminalidade, enquanto populações mais idosas e com proporções mais elevadas de imigrantes têm um efeito repulsor do crime. A inclusão de covariáveis explica parcialmente a agregação espacial do crime, embora a análise de resíduos e os critérios de seleção de modelos sugiram que a especificação do modelo poderia ser melhorada, nomeadamente com a inclusão de Pontos de Interesse relevantes.

Palavras-chave: Crime de Rua, Estatística Espacial, INLA, Processos de Cox Log-Gaussianos, SPDE

Contents

List of Figures

List of Tables

Acronyms

AIC	Akaike Information Criterion
CCD	Central Composite Design
CSR	Complete Spatial Randomness
DIC	Deviance Information Criterion
FEM	Finite Element Method
GAM	Generalised Additive Model
GIS	Geographic Information Systems
GLM	Generalised Linear Model
GMRF	Gaussian Markov Random Field
GNR	*Guarda Nacional Republicana*
GRF	Gaussian Random Field
INLA	Integrated Nested Laplace Approximation
KDE	Kernel Density Estimation
KLD	Kullback-Leibler Divergence
LGCP	Log-Gaussian Cox Process
LGM	Latent Gaussian Model
MCMC	Markov Chain Monte Carlo
PC	Penalised Complexity
PJ	Polícia Judiciária
POIs	Points of Interest
PSP	*Polícia de Segurança Pública*

SAM-GLM Spatially Aware Mixture of Poisson Generalised Linear Models
SEHP Self-Exciting Hawkes Process
SIG-SIRESP *Sistema de Informação Geográfica dos meios SIRESP*
SPDE Stochastic Partial Differential Equation
STSec Statistical Section
STSub Statistical Subsection

VIF Variance Inflation Factor

WAIC Watanabe–Akaike Information Criterion
WGS84 World Geodesic System 1984

Chapter 1

Introduction

1.1 Motivation

Crime is *not* random

Approximately half of all criminal events are concentrated in microgeographic areas, known as *hotspots*, a striking pattern consistently validated by empirical evidence across numerous cities [2]. This finding has profoundly influenced crime control efforts, particularly following a major paradigm shift in criminology during the late 1980s. Rather than focusing solely on offender profiles, criminologists began to examine the spatial patterns of crime. As Weisburd [3] refers, the term *criminology of place* was introduced by Sherman, Gartin, and Buerger in 1989 [4]. In their seminal paper, they analysed 323,979 calls for service in Minneapolis, United States of America, revealing that 50% of calls originated from just 3% of locations. As the phenomenon started being observed in an increasing number of cities, Weisburd [3] claimed it to be a universal experience and proposed a *law of crime concentration*, in the fashion of a physics law:

> This law [of crime concentration] states that for a defined measure of crime at a specific microgeographic unit, the concentration of crime will fall within a narrow bandwidth of percentages for a defined cumulative proportion of crime.[1]

Weisburd was first able to test the hypothesis in North American cities [3] and Tel Aviv-Jaffa, Israel [5]. In a more recent paper [6], the same author and collaborators present a comprehensive review of studies on crime concentration conducted over the last 35 years in different cities. Overall, the findings corroborate the law. Weisburd has also shown that concentrations are relatively stable across the years [3]. Gill et al. [7] sought to extend this

[1] In 2009, for Cincinnati, United States of America, 50% of crime at street segments happened in just 6% of all street segments and 25% was concentrated in 1.6% of all street segments [3]. In this example, the 25% and 50% represent bandwidths of specific cumulative proportions of crime and the microgeographic unit is the street segment.

paradigm to suburban areas, finding strong support in the case-study of Brooklyn Park, a suburban city outside Minneapolis, over a 15-year period from 2000 to 2014. However, it should be stated the authors mention a higher variability of crime hotspots over time in these areas.

The strong empirical support for the law of crime concentration leads to our first motivating practical issue: police resources should primarily be allocated to hotspots. Following this reasoning, over the last decades, the hotspot policing model has emerged as an effective policing strategy. The slogan "Put cops on dots", as coined by former New York Police Department Deputy Commissioner Jack Maple and mentioned by Braga et al. [2], reflects a remarkable simplicity in the approach, which leads one to question its revolutionary nature. However, for many years, street patrolling relied solely on a reactive model leaning on officer experience. Braga et al. [2], conducted a formal meta-analysis of 62 eligible studies from various cities to evaluate the success of hotspot policing programs compared to traditional approaches. Their analysis demonstrated that these programs are more effective at reducing crime within targeted areas. Moreover, the findings suggest that focusing on hotspots may lead to crime diffusion, producing positive effects on crime reduction in surrounding areas rather than causing crime displacement. Additionally, Dau et al. [8] emphasize that the number and duration of such interventions play a significant role in their overall effectiveness, underscoring the necessity of incorporating the temporal dimension in crime analysis at finer scales than the yearly mark.

Nowadays, the collection and storage of georeferenced crime occurrence data by police departments has become a standard practice worldwide, with Geographic Information Systems (GIS) being widely utilised [9]. These systems provide tools for capturing, storing, analysing, managing, and visualising spatial or geographic data [10]. As a result, Police Forces already possess some statistical tools to conduct an *Intelligence-led Policing* strategy, where the goal is to exploit analysed information on past crimes, to support informed decision-making and resource allocation. However, in this work, we aim to explore the potential of a more in-depth spatial statistical analysis that goes beyond descriptive studies and visual data analytics, specially as a means to define a long-term strategy.

Crime records can be interpreted as dynamic (i.e., spatio-temporal) point patterns on a map. We believe spatial point pattern analysis, both non-parametric and parametric through spatial process models, offers suitable tools to investigate crime distribution. Moreover, given the explanatory power of statistical modelling, another motivation is to examine the socio-economic and environmental factors that influence crime, in the hope of scientifically evaluating what drives criminal activity to certain places in detriment of others. Criminology theories exist to try to explain this sociological phenomenon, however, they should be supported by statistical evidence, to guide the design of effective policies in the context of crime prevention.

Lastly, the motivation for this work also stems from recognising the scarcity of studies in Portugal that analyse crime at the microgeographical level. This gap may be attributed to limited access to fine-scale georeferenced data, privacy regulations that restrict data

sharing, and a long-standing tradition of aggregate-level analyses within criminological research. Furthermore, the adoption of GIS technologies in Portugal has occurred relatively recently compared to North America and several other European countries. More broadly, as noted by Weisburd et al. [6], European cities in general lack systematic reviews of the law of crime concentration, representing early steps in the development of a criminology of place that countries such as the United States of America have undertaken long ago. This urgency may be, in part, motivated by necessity, due to higher crime rates. Nevertheless, lower crime rates, as is the case in Portugal, one of the safest countries in the world[2], should not diminish the importance of crime prevention and a well-thought-out use of police resources.

The case-study of the city of Almada

This study was conducted in collaboration with the *Guarda Nacional Republicana* (GNR), which provided access to the data, and the idea for the dissertation was to apply our analysis to a specific spatio-temporal domain and see what conclusions could be taken regarding certain delineated objectives.

In Portugal, the GNR is the oldest and most extensive security force, responsible for 94% of national territory, in which 54% of the Portuguese population resides [12]. Using the *Sistema de Informação Geográfica dos meios SIRESP* (SIG-SIRESP), the GNR monitors the locations of patrol units via radio signals, storing this data in a centralised database [13].

According to paragraph 1 of Article 161 of the General Regulations of Service of the GNR (*Regulamento Geral do Serviço da Guarda Nacional Republicana*), police service is conducted primarily through patrols led by officers from stations or other ranks. For decades, these patrols followed a reactive policing model, where officers either responded to calls or patrolled routes defined to some degree at random [13]. Recognizing the limitations of this non-preventive approach, the GNR has been trying to adopt a proximity policing model, emphasizing prevention through visibility and vigilance in strategically important locations. This shift aligns closely with the principles of Intelligence-led Policing.

The data raw provided by the information division of the GNR consists of crime events recorded by officers of the *Destacamento Territorial*[3] of Almada, for which legal criminal proceedings were initiated between 2022 and 2023, although the GNR has been georeferencing crime since 2019. Since our analysis focuses on the date of occurrence rather than the date of initiation of criminal proceedings, we included only events that occurred between 2022 and 2023.

The Municipality[4] of Almada is a region of Portugal located near Lisbon in the opposing margin of the Tejo river (Figure 1.1). It was considered a relevant case-study as it exhibits

[2] In 2025, Portugal has ranked 7 out of 163 countries in the annual Global Peace Index (GPI) compiled by the Institute for Economics & Peace (IEP), making the top 10 consistently since 2016. [11]

[3] A *Destacamento Territorial* is a regional-level unit of the GNR that coordinates and supervises several local posts (*Postos*) within its jurisdiction.

[4] Portugal is administratively divided into districts (*distritos*), municipalities (*munícipios*) and parishes (*freguesias*).

a higher-than-average incidence of crime. For example, it registered a crime rate of 43.0‰ in 2023, compared to 35.0‰ in mainland Portugal in the same year[5]. Our study, however, concerned only the regions of the Municipality of Almada under the GNR jurisdiction, as the *Destacamento Territorial* of Almada includes two regions outside of this municipality. Another, perhaps coincidental but surely relevant, factor in this selection was the proximity with the reality, as the School where this work was developed is located within the municipality. Finally, the dataset was further narrowed to focus specifically on street crime, those most directly addressed by routine police patrols.

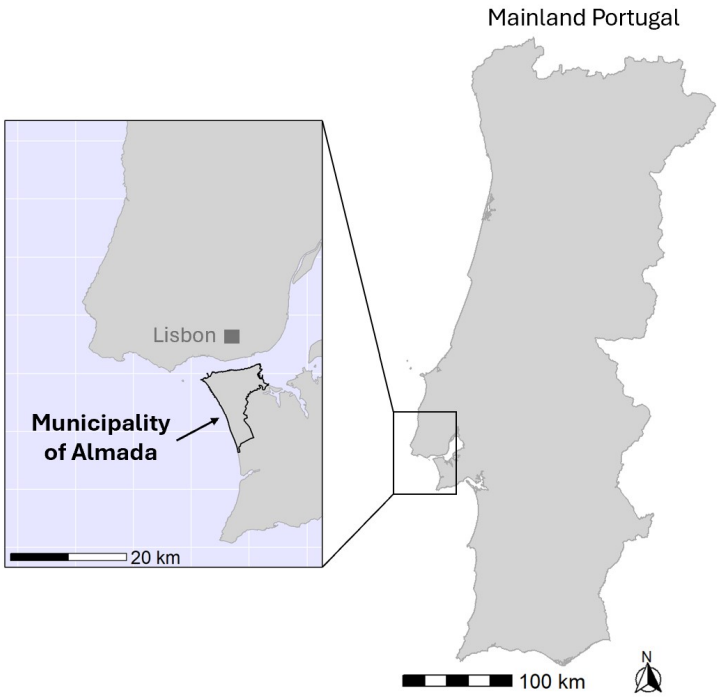


Figure 1.1: Location of the Municipality of Almada within mainland Portugal.

1.2 Objectives

Research Aim

To assess the potential of spatial point pattern techniques in analysing crime data from the Municipality of Almada within the areas under jurisdiction of the GNR, with the ultimate goal of having a better understanding of the crime phenomenon in this region and possibly improving patrol allocation.

[5]Crime rate calculated as the number of crimes per resident population × 1000. Statistics obtained from Instituto Nacional de Estatística (INE), Taxa de criminalidade (‰) por Localização geográfica (NUTS - 2024) e Categoria de crime; Anual. Available at: https://www.ine.pt/xportal/xmain?xpid=INE&xpgid=ine_indicadores&indOcorrCod=0012260&selTab=tab0. Accessed: 08/01/2024.

In order to achieve this overarching goal, we have compiled a list with a series of concrete objectives pre-defined for this project:

1. To employ spatio-temporal exploratory analysis methods, particularly to estimate spatial dependencies between points (i.e. crime events).
2. To implement a spatial Log-Gaussian Cox Process (LGCP), a well-established statistical model in the literature, to analyse crime data, as it is suited for clustered point patterns and accounts for stochastic dependency [14]. The model should consider at least dependence among points and relevant spatially varying covariates.
3. To adopt a Bayesian approach for model inference using the state-of-the-art Integrated Nested Laplace Approximation (INLA) with the Stochastic Partial Differential Equation (SPDE) approach.
4. To identify potential hotspots in the regions of Almada under the jurisdiction of the GNR and interpret the results in light of criminology theory.

1.3 Document structure

The remainder of this dissertation is structured as follows: Chapter 2 presents the literature review, providing context and examining relevant works by other authors; Chapter 3 introduces the basic theory of spatial point pattern analysis that underlies the adopted methodology; Chapter 4 introduces the dataset and addresses the non-parametric spatio-temporal analysis of the street crime point pattern, while Chapter 5 focuses on the spatial parametric analysis, employing spatial LGCP models fitted with INLA using the SPDE approach. Chapters 4 and 5 present the theoretical foundations of the methodologies applied, as well as the corresponding implementation and results. Finally, Chapter 6 presents the conclusions of this study and offers directions for future research.

This project was implemented using the `R` programming language, version 4.4.2. The most relevant code snippets can be found in Appendix A.

Chapter 2

Literature Review

Crime analysis is inherently multidisciplinary, drawing from both criminology theory and statistical methodologies. The combination of these areas allows for a deeper understanding of crime data and the development of effective interventions. This chapter opens with a brief overview of the main criminology theories, which inform the analysis of crime data and the formulation of crime prevention strategies. It then focuses on spatio-temporal statistical crime models employed by others, with a focus on works involving Log-Gaussian Cox Process (LGCP) models, as it is the specific type of model we used. The chapter continues with a brief discussion of the two most widely used computational methods for Bayesian inference: Markov Chain Monte Carlo (MCMC) and the Integrated Nested Laplace Approximation (INLA). We conclude with an overview of Portugal's criminal landscape during 2022–2023, alongside a discussion of recent research on crime modelling in the country.

2.1 Criminology Background

For over three decades it has been evident that crime does not distribute evenly, as showed by Weisburd's law of crime concentration stated in Section 1.1. To effectively analyse crime data, it is essential to understand the theoretical frameworks that explain the spatial and social dynamics of criminal activity. This paradigm central to *environmental criminology*, can be succinctly captured by the question: "Does crime fundamentally occur in specific environments and under certain conditions?"

There are three core theories rooted in the broader foundation of environmental criminology [10]:

- *Routine activity theory* [15], perhaps the most widely recognised theory, emphasises the convergence of three elements in time and space: motivated offenders, suitable targets, and lack of capable guardianship. Crime hotspots emerge when these factors are present in high concentration, leading to an accumulation of criminal opportunities in specific locations and at specific time instants.

- *Rational choice theory* [16] states that offenders undergo a decision-making process before committing the crime. They weigh the potential benefits and risks, even if very briefly and in a simplified manner, with a specific goal in mind. The criminal act is therefore a rational decision, rather than merely a spontaneous reaction to an opportune moment.

- *Crime pattern theory* [17] converges the former two theories by providing a reasoning for where the offence is likely to happen. It suggests that criminals often operate within familiar environments. An important concept that emergences in this theory is *near repeat victimization*, which suggests that crime often happens in clusters or in nearby locations. If an offender commits a crime in a certain area, there is a higher plausibility that they will commit similar crimes in that area or nearby locations. This underscores in part the importance of accounting for spatial autocorrelation in crime spatial point patterns.

These theories share a common focus on how opportunities for crime arise and how offenders exploit them, being collectively known as *opportunity theories*. The field of criminology of place emphasises the spatial dynamics of crime while being closely aligned with opportunity theories [18].

Another highly relevant theory in criminology is the *social disorganisation theory* [19], which posits that crime is more prevalent in communities with weakened social structures, where high levels of poverty and instability disrupt social control, leading to higher crime rates. Proponents of this theory have traditionally focused on larger geographic units, such as communities or broader societal regions. However, while originally sociological and community-oriented, social disorganisation theory has been adapted to fit the criminology of place framework by explaining how neighbourhood-level factors contribute to the creation of environments conducive to crime. Notably, Weisburd et al. [18] identified several socio-economic indicators that reflect social disorganisation at the street segment level in the city of Seattle, such as socio-economic status, residential property values, racial heterogeneity, and distance to the city center, among others.

In practice, opportunity theories underpin concrete crime prevention strategies. For instance, situational crime prevention focuses on reducing opportunities for crime by increasing the risk of being caught, raising the effort required to commit the crime, or decreasing the reward associated with it [10]. Similarly, hotspot policing is heavily influenced by opportunity theories, as these suggest that preventing the spatial and temporal convergence of victims and offenders can reduce criminal activity. Deploying police resources strategically in crime-prone areas can effectively manipulate the environmental setting and deter offenders [2].

Criminology theories also aid modelling, since they can (and should, as argued in a passage by Groff and LaVigne highlighted by Chainey and Ratcliffe [10]) serve as a foundation for the *a priori* selection of relevant covariates in crime models. Social disorganisation theory, in particular, emphasises the importance of incorporating socio-demographic

indices (such as income, population density, education levels, and unemployment rates, among others) at appropriate geographic scales, as these factors are indicators of the level of social cohesion. Povala et al. [20] provide a compelling illustration of criminology-based covariate choice for modelling burglary target selection. They categorise the explanatory variables into the three dimensions already mentioned in the context of situational crime prevention: risk, effort, and reward.

Lastly, criminology theories provide the framework for interpreting results, namely explaining correlations and reasoning about potential causal relationships between contextual factors and crime distributions in each specific case-study.

2.2 Statistical crime modelling

Recalling the opening sentences of this document, we remind crime does not occur uniformly across space or time. It is heavily influenced by physical environment, routine human activity, and social dynamics, all of which exhibit local spatial and temporal patterns. Thus, naturally, the interest of analysis lies in the *spatial* or *spatio-temporal crime point pattern* and not the isolated crime counts themselves.

Scouring the literature, it is noticeable that most statistical modelling, specially in earlier studies, focuses on *areal crime data*, where crime counts aggregated at the regional level serve as the response variable [21–23]. Such data is more readily available to the public and socio-economic covariates are typically easier to obtain at this spatial resolution. However, areal models are limited in their ability to capture local clustering [24], which we already established crime frequently exhibits at the street level. This limitation motivates the statistical modelling of crime point patterns directly, where individual crime events are treated as realisations of an underlying point process.

For simplicity, one may think of the relationship between a point process and a point pattern as analogous to that between a random variable and an observed sample: the point pattern is a realisation of the underlying stochastic mechanism, the point process, that governs the occurrence of events in space and time [14]. The focus is to model the intensity function of the point process, which characterises the expected spatial (or spatio-temporal) distribution of crime events, introducing covariates to account for factors influencing the distribution and including random effects to capture unexplained variation.

In Table 2.1 we have summarized a series of details from studies found in the literature that implement spatio-temporal point pattern models in the context of crime analysis. Burglary is the most extensively studied crime category at the micro-level and studies examining other crime types do not necessarily focus on street crime. Crime point pattern data originates from two main sources: emergency calls or police records. Regardless of the source, the literature consistently acknowledges that crime data is affected by under-reporting. As a result, it is important to keep in mind that recorded crime events represent a fraction of the actual number of incidents. Thus we should be aware of a possible underestimation of true crime intensity.

Table 2.1: Summary of relevant research cases of application of point pattern models to study crime.

Reference	Case-study	Model	Fixed Effects	Relevant results
Briz-Redón and Mateu, 2023 [25]	• **Crime types**: property crime (burglary and shoplifting) and robbery (violent and theft). • **Study region**: Valencia, Spain. • **Study period**: 2017.	Mechanistic Bivariate Spatio-temporal Model (Self-exciting Hawkes Process (**SEHP**) based)	**Socio-economic factors:** total population, % population aged 65, % population aged between 15 and 29, % foreign-born population, average household income.	• Spatio-temporal interaction relevant up to a lag of 30 weeks and 120 weeks for property crime and robbery, respectively. • Spatial influence of past events is greater for property crimes. • Crime levels peaked in summer for both crime types; • **Significant covariates**: robberies and property (+)[1] % foreign-born population; robberies (+) average household income.
González et al., 2024 [26]	• **Crime types**: homicide, car theft, shoplifting, burglary, motorcycle theft, and robbery. • **Study region**: Kennedy locality of Bogotá, Colombia. • **Study period**: 2012 to 2017.	Spatio-temporal Log-Gaussian Cox Process (**LGCP**) (for each crime type)	**Spatial - Points of interest (POIs)**: construction, health, injuries, libraries, markets, parks, pharmacies, schools, transport, and water); **Temporal**: precipitation, temporal trends (linear, semiannual, quarterly, weekly), days of the week, days of the month.	• Crime levels increase on weekends. • Different crime types have different risks according to days of the week and months of the year. • Homicides behave differently, showing no seasonality besides a slight decrease over time. • **Significant covariates for street related crime**: car theft (+) transport service, (-) criminal injury, (+) construction zone, (-) pharmacy, (-) park; motorcycle theft (-) criminal injury, (-) pharmacy; shoplifting (-) criminal injury, (-) pharmacy, (-) medical centre; robbery (-) transport service, (-) criminal injury, (-) construction zone, (-) water canal, (+) pharmacy, (-) medical centre, (+) park, (-) market.

(Continued on next page)

[1]The (+) designates a positive contribution for crime intensity while (-) designates a negative contribution.

(Continued from previous page)

Reference	Case-study	Model	Fixed Effects	Relevant results
Escudero et al., 2024 [27]	• **Crime types**: include thefts of individuals, houses, and commercial premises, robberies, intimidation, scams,... • **Study region**: Riobamba, Ecuador • **Study period**: 2010 to 2014	Spatio-temporal **LGCP**; Spatio-temporal **SEHP**	Covariates of the LGCP: **Spatial** shortest distance to Community Police Units (CPU), cameras, shopping centres and parks; **Temporal**: weekly average of environmental temperature, thermal sensation, global solar radiation, soil temperature, and relative humidity, temporal trends (linear, annual, semiannual), days of the week, months of the year.	**LGCP** • Crime peaks on Fridays, but lowers on weekends and Tuesdays as well as from March to July, September and November. • Correlation range is close to 120 m (spatial) and close to 1 day (temporal). • **Most significant covariates**: (+) global solar radiation, (+) thermal sensation, (-) ambient temperature, (+) shortest distance to CPU, (-) shopping centres and parks. **SEHP** • Overall increase in reported crimes from 2012 to 2014, however it is attributed to improvements in the crime reporting system. • Sundays and Mondays report lower crime rates, with highest values on Fridays and Saturdays. The authors relate it to more pedestrians circulating. • Many events follow along the street network. • An event triggers new ones in a radius of about 175 m and in the following 3.5 days. • About 75% of criminal activity is explained by the triggering effect (repeat offences) and only 3% by a fix background rate.

(Continued on next page)

(Continued from previous page)

Reference	Case-study	Model	Fixed Effects	Relevant results
Povala et al., 2020 [20]	• **Crime types**: Burglary • **Study region**: City of London, United Kingdom • **Study period**: 2015, 2013 to 2015	Spatially Aware Mixture Poisson Generalised Linear Model (**SAM-GLM**) with 3 mixture components; Spatial **LGCP** (for performance comparison)	Tested 4 model specifications built using: **Socio-economic factors**: housing and household related, unemployment rate, ethnic heterogeneity measure, accessibility, urbanization index,...; **POIs**: retail, eating/drinking, education/health, accommodation, sport/entertainment.	• The SAM-GLM identified 3 very distinct components driving burglary intensity: (1) residential areas (+) household density, (+) POIs, (+) house prices; (2) city neighbourhoods (+) POIs, (+) accessibility, (+) ethnic diversity, (+) occupation variation, (-) residential turnover; (3) low-intensity zones (e.g., parks, industrial areas), with very minimal burglary rates. • Across all components, spatial dependence was important to model burglary rates. • In several situations, the SAM-GLM predicted hotspots better than the LGCP.
Zhao et al., 2019 [28]	**Crime types**: burglary; **Study region**: A city in China (unnamed); **Study period**: 2016.	Spatio-temporal **LGCP**	**Spatial - POIs**[2]: village centres, residential locations, wholesale and retail points, restaurants, companies, education related places, commercial/service facilities, public and government facilities, and sports and recreational places; **Temporal**: 24h day divided into 8 time periods.	• Spatial correlation of more than 0.5 up to about 10 km. • Consistent clusters of high-risk areas across time periods. • Southern areas show higher burglary intensity trends, surpassed by north-eastern areas in later periods of the day. • **Significant covariates**: (+) wholesale and retail points, (+) education related places, (+) sports and recreational places.

(Continued on next page)

[2]The Points of Interest from the work of Zhao et al. [28] were translated from Mandarin to English using Google translate.

(Continued from previous page)

Reference	Case-study	Model	Fixed Effects	Relevant results
Rodrigues and Diggle, 2012 [29]	• **Crime types**: Homicides • **Study region**: Belo Horizonte, Brazil • **Study period**: 2000-2006	Spatio-temporal **LGCP**, with a convolution-based non-separable model for the Gaussian Process covariance	No covariates mentioned.	• Identified 3 possible anomalies (crime clusters) near the southwest, northwest, and east boundaries of Belo Horizonte, with the southwest being more persistent.
D'Angelo et al., 2024 [30]	• **Crime types**: Armed Robberies • **Study region**: Downtown Bucaramanga city, Colombia • **Study period**: 2018	Spatio-temporal **SEHP** adapted to events on a linear (street) network; Planar spatio-temporal **SEHP** (to compare with the network case)	36 covariates in total. **Socio-economic factors**: unemployment rate, education level, literacy rate, public services coverage, socio-economic strata, and housing quality; **Demographic aspects** total population, gender ratio, and elderly population,...; **Environmental conditions**: street slope, road type,...; **Geographical (POIs)**.	• An event can trigger new ones within the next 10 days, mostly up to a week, and in a radius of approximately 100 m. • Higher intensities in the second semester of the year, with a maximum value in December, and also on Wednesdays, Fridays, and Saturdays. • **Most significant covariates**: present/absence covariates (+) cycleway, (+) pedestrian, (+) primary, (+) residential, (+) secondary, (+) service, (+) tertiary highways, (+) bridges, (+) steep slope, (-) one-way street, (-) flat slope; continuous covariates literacy rate, housing quality score, water/sewerage coverage, unemployment rate, hospitals/street pop., police stations/street pop. • The street network model performed better than the planar model.

A popular choice of model seems to be the *Self-Exciting Hawkes Process (SEHP)*, which looks at the criminal phenomenon as a contagion-like process, where past events increase the plausibility of future ones within specific spatial and temporal ranges. It combines a background rate of independent events with a triggering component that captures near-repeat victimization. Covariates are most often included in the background component [25, 30], although authors state that both parts may incorporate them.

These models can estimate the spatial radius and temporal window within which an event may trigger subsequent ones, as well as quantify how much this triggering component explains crime compared to the background part. Looking at Table 2.1, for crime data, these ranges are typically on the order of a few hundred meters and a few days, as observed in the results of Briz-Redón and Mateu [25], Escudero et al. [27], and D'Angelo et al. [30].

Particularly relevant to the study of street crime is the observation by Escudero et al. [27] that their models' estimated crime intensity closely followed the street network pattern. There are two ways to go about this situation: one can simply include streets as a covariate and treat them as crime attractors or, we could address the issue by formulating the point process directly on a linear network. D'Angelo et al. [30] adopted the latter approach to model armed robberies in Colombia and reported an improved fit compared to the planar model specification.

Povala et al. [20] stand out from the rest with their proposal of a *Spatially Aware Mixture of Poisson Generalised Linear Models (SAM-GLM)*, which divides the study area into n grid cells grouped into census tract blocks. Instead of assuming a single process for all cells, the model uses a fixed number, K, of mixture components, each with its own regression coefficients for spatial covariates. Cells are probabilistically assigned to components based on latent weights, that vary by block and component, modelled via a Gaussian Process to capture spatial dependence. This framework flexibly accounts for spatial heterogeneity while maintaining spatial coherence.

2.2.1 Log-Gaussian Cox Process

In Table 2.1 we included all the Log-Gaussian Cox Process (LGCP) crime models found in the literature. As González et al. [26] refer, the LGCP is not as widely used as, for instance, mechanistic models (such as the aforementioned SEHP) in the context of crime. However, the many successful applications of this process for modelling, for instance, in the fields of ecology [31], epidemiology [32, 33] and seismology [34], has been consolidating it as a worthy relevant alternative.

Møller, Syversveen and Waagepetersen [35] laid down the theoretical foundations of the spatial LGCP, a specific type of Cox Process in which the logarithm of the intensity function itself is stochastic and modelled as a Gaussian Process. As the authors denote, the LGCP is a flexible model for situations where points tend to form clusters, as is the case of crime. Building on this pivotal article, Diggle et al. [24] describe the class of spatial

and spatio-temporal LGCPs, extending the previous work to the temporal dimension and focusing on the development of models using these processes, to include covariates to explain real-world phenomena. They present a diverse array of concrete applications, showcasing the potential of the LGCP from simpler univariate models, to multivariate response cases.

Research cases show the LGCP can not only excel at spatial prediction of intensity surfaces and risk maps, but also that it possesses good potential in forecasting. Escudero et al. [27] were able to forecast crime trends for 6 weeks based on data from the previous 250+ weeks with very acceptable differences between predicted and real values, both for total number of crimes and crime intensity surfaces. González et al. [26] propose a surveillance system that reports predictive probabilities, using data up to a certain time instant, in order to assist police in resource allocation. More specifically, the system calculates the probability of exceedance of the Gaussian Random Field over a certain threshold value above which corrective police actions should be imposed, as these peaks in the random field denote abnormally elevated and unexplained criminal activity.

From the works we gathered, González et al. [26] applied the most complete spatio-temporal LGCP model, and while Zhao et al. [28] have a simpler model, these authors use a spatial Matérn covariance function for the Gaussian Process, as in our work, and follow the same fitting procedure, combining INLA and the SPDE approach. González et al. [26] preferred a spatio-temporal exponential covariance function and used a MCMC procedure.

Similarly to the SEHP, the LGCP supports the adaptation of the point process to a linear network scenario [36]. This has been explored particularly when using the LGCP to model traffic accidents, which are inherently tied to the street network. Abebe et al. [37] examined traffic accident patterns in Addis Ababa, Ethiopia, from 2016 to 2019. As in our work, the authors utilised INLA combined with the SPDE approach to fit their model. However, crash locations were projected onto the nearest roads segments and the Finite Element Method (FEM) mesh built for INLA was constrained to the street network.

Finally, we once again mention the works of Povala et al. [20] and Escudero et al. [27] as these authors applied other models alongside a LGCP model.

Escudero et al. [27] provide a great example of the SEHP and LGCP applied in a complementary manner: a comprehensive spatio-temporal LGCP serves as the baseline analysis of criminal activity, to which an SEHP is added to assess the contagion effect and to re-analyse the temporal dynamics of crime in light of this effect.

On the other hand, Povala et al. [20] view the SAM-GLM and LGCP as competitors, comparing their prediction capabilities. The authors concluded the SAM-GLM performed better than the LGCP overall. However, it was also observed that the SAM-GLM needed $K = 2$ components for out-of-sample performance to match the LGCP using 1-year data from 2015. Furthermore, $K = 4$ components were needed when using 3-year data, from 2013-2015, to match both held-out log-likelihood and the root mean square error. A possible explanation is that the more years are added, the smoother the point pattern and

thus it is easier to interpolate the Gaussian random field.

The authors also tested the hotspot prediction success. They considered several scenarios, where hotspots correspond to the n cells with highest expected value of burglaries. Depending on the number of cells flagged as hotspots, when using measures of performance of crime hotspot models, the LGCP outperformed or matched the SAM-GLM with $K < 3$ and up to $K < 5$ if there was a small number of cells selected as hotspots (0-150 cells). However, if $K \geq 3$ and the number of hotspots flagged was higher, the SAM-GLM proved to be superior. Overall, the great advantage of the SAM-GLM, as the authors put it, is the ability to "allocate each cell to a cluster that better describes the burglary count in that location". This is particularly noticeable when the pattern is not as smooth, namely in the 1-year data case. Naturally, there is the computational complexity cost of adding more components and block dependency, which was relaxed in the final model by assuming *a priori* that the blocks are independent given a specific parameter. Also, the MCMC trace plots of each model parameter need to be carefully examined to assure there are no identifiability issues. The likelihood of a mixture model is invariant under the relabelling of mixture components, meaning one needs to be sure of which components are associated with which covariates in order to interpret results properly.

2.2.2 Computational methods for Bayesian inference

Bayesian inference uses the likelihood function and prior distributions of model parameters in order to apply Bayes' theorem to estimate the parameters' posterior distributions, which often lack closed-form solutions. In opposition, classical statistics sees parameters as unknown yet fixed entities. The Bayesian approach has gained popularity, specially in spatio-temporal statistics, due to tools such as: `spatstat` [14], `mcmc` and `lgcp` [38, 39] which use *Markov Chain Monte Carlo (MCMC)* methods [40]; or `R-INLA` [41–43] and the more recent `inlabru` [44], both based on the *Integrated Nested Laplace Approximation (INLA)* [41].

The LGCP Gaussian field can be approximated (traditionally through a regular lattice) and included in the model, which can be estimated by MCMC or INLA. MCMC is simulation-based, constructing a Markov chain to sample from the posterior distribution, while INLA employs numerical analysis techniques, such as Laplace approximations. INLA's efficiency is boosted by approximating *Gaussian Random Field (GRF)* to *Gaussian Markov Random Field (GMRF)*, leveraging matrix sparsity for computation efficiency [41].

A great advancement in INLA for LGCP models was Simpson et al.'s work [45]. The authors separated the task of approximating the Gaussian Random Field from the approximation of the exact positions of observations, using the Stochastic Partial Differential Equation (SPDE) approach proposed by Lindgren and Rue [43]. The traditional lattice served both purposes simultaneously but imposed the same resolution everywhere within the study area, wasting computational effort in regions with little or no data. Lindgren and Rue showed that a spatial Gaussian Process with Matèrn covariance is a solution to a particular SPDE, dependent on certain known parameters. Using the FEM, one generates

a triangular mesh over the study region that can be used to build a GMRF representation of the solution, which by definition has conditional independence properties that contribute to fast computation.

INLA requires a *Latent Gaussian Model (LGM)* structure, which supports a broad range of statistical models, including regression models, dynamic models, and spatial and spatio-temporal models [41]. This versatility allows INLA to handle a wide number of real-world applications efficiently [46]. Nevertheless, it is acknowledged that MCMC remains slightly more flexible as it can handle complex models beyond these structures and has direct access to the joint posterior distributions of the parameters, while INLA only provides a marginal posterior distribution for each parameter. However, Diggle et al. [24], supported by the work of Taylor et al. [38], rightly emphasise that MCMC has struggles with the estimation of the Latent Gaussian Field parameters.

Comparisons show that INLA is generally faster but may slightly reduce predictive accuracy in some scenarios [38, 41]. MCMC, on the other hand, while computationally intensive, excels in predictive accuracy for complex models. A hybrid approach combining INLA and MCMC has been developed by Gómez-Rubio and Rue [47] to take advantage of the strengths of both methods, extending INLA's applicability to models outside the LGM structure.

We also highlight Teng et al.'s work [48], which provided a comparative analysis of INLA (with and without SPDEs), a more complex Monte Carlo-based method, the Hamiltonian Monte Carlo and the Variational Bayes Method to fit LGCP models. It was observed that the SPDE-based INLA approaches showed a tendency to over-smooth the latent field, though this effect was mitigated when using a finer mesh. From here we extract a warning that the mesh parameters should be carefully tuned, a crucial step as seen in INLA tutorials [49, 50].

Finally, we believe the overall predominance of Monte Carlo methods should also be attributed to the fact that (1) they are older and already very well established in Bayesian statistics [24], and (2) INLA seems to have a steeper learning curve. Ultimately, the choice between INLA and MCMC depends on the trade-off between computational efficiency and accuracy, with INLA excelling in rapid inference and MCMC offering more flexibility for intricate models.

2.2.3 Covariates and random effects

As can be seen by Table 2.1, crime models often follow a *Generalised Additive Model (GAM)* structure and can incorporate a wide range of covariates.

In the spatial domain, these include socio-economic indices, *Points of Interest (POIs)*, and environmental characteristics. Social disorganisation theory provides a strong rationale for including socio-economic variables. In contrast, opportunity theories support the use of POIs, locations of physical structures that may function as crime attractors or deterrents. They also reinforce the need to study the influence of environmental conditions, for

instance, street illumination, surveillance systems [27], street slope and street type [30]. For estimation and prediction purposes, point process models require these covariates to be defined at event locations and also at a finite number of extra locations. Indeed, spatial covariate selection usually brings about the challenge of how to incorporate them into the model.

Socio-economic variables are usually piece-wise, i.e., spatially aggregated (for example, in census tract regions). A very direct approach passes by simply assign a covariate value to a location according to the region it falls in. The finer the aggregation, the more precise this allocation is. A possible alternative consists in building a continuous map out of the piece-wise covariates, namely through *Kernel Density Estimation (KDE)*. However, as Fadluronhman et al. [32] rightly alert, kernel smoothing of piece-wise covariates can obscure more localised changes, leading to loss of information. Given this context, the authors follow a classical maximum-likelihood approach and develop a two-step procedure adequate for fitting piece-wise covariates using a spatial LGCP model with a zero-mean Gaussian Process with exponential covariance structure. As the authors highlight, a closed form solution is found without the need for approximations to continuity.

Kernel smoothing is frequently adopted for modelling POIs. González et al [26] computed a map of the distance to the nearest locations for several POIs. Then, the authors applied a kernel smoothing method to reduce noise, known as the Nadaraya–Watson smoother. The amount of smoothing is controlled by the kernel bandwidth, which was chosen automatically using the standard method of least-squares cross-validation. Although not explicitly mentioned, we believe Zhao et al. [28] might have followed a similar route.

Climate-related covariates, such as precipitation, thermal sensation, solar radiation, among others, are associated to events through time. In conjunction with these external factors, the behaviour of data in time can be modelled with other fixed effects, for instance: a simple linear time trend added to the model formula, and sines and cosines included as weekly and monthly seasonal effects. In addition, one can opt for introducing the months of the year, the days of the week, or even parts of the day, as factors, leaving a reference category out for each one [25, 27].

Regarding random effects, the models may include unstructured components, such as independent and identically distributed (i.i.d.) random variables for each observed data response, in both the spatial and temporal domains. The temporal windows (weeks, months, parts of day) mentioned may be present as i.i.d. random effects as well. Aside from these, all spatio-temporal models include a structured effect, typically modelled as a zero-mean Gaussian Random Field. For a purely spatial structured effect the Matérn covariance function is the go-to choice, while spatio-temporal structured effects rely on a simpler separable exponential covariance function for both space and time. A zero-mean Gaussian Process with exponentially decaying covariance function in time is essentially the continuous-time analogue of an Autoregressive Stationary Process of order 1 [28].

Rodrigues et al. [29] tackled the particular issue of developing a non-separable spatio-temporal covariance function for the Gaussian Process. Separable covariance functions are certainly more computationally feasible, however they may oversimplify complex spatio-temporal dependencies. Therefore, building on the concept of positive and negative non-separability, they proposed a convolution-based non-separable model for the covariance function. A Bayesian approach with MCMC is used for parameter estimation and spatial prediction. This model was designed for real-time spatio-temporal surveillance of homicides in Belo-Horizonte, Brazil, enabling the calculation of predictive probabilities for high-risk areas or times in order to identify possible anomalies. We note no covariates were mentioned in the study, as the authors fully focused on the potential of the new covariance structure.

Lastly, in what concerns structured random effects, time can also be incorporated as a Random Walk of order 1 or 2, which corresponds to a non-stationary process where variance grows with time, appropriate for modelling gradual temporal trends [50].

2.3 Crime analysis in Portugal

2.3.1 General crime landscape in 2022-2023

To get a sense of the general crime landscape in Portugal, we will be referring back to the 2023 Annual Internal Security Report [51]. This public annual report emitted by the Portuguese government integrates data of recorded crime incidents from eight criminal police services, including the GNR. Through maps and descriptive statistics, it paints a general picture of crime in Portugal and its different districts. It is also a tool for assessing the outcomes of the work carried out by the various entities, listing police operations in the territory.

It was reported that overall crime incidences have increased in 2022 and 2023 (Figure 2.1). The increase in 2022 was in part expected since this year there were considerably less mobility restriction measures in Portugal compared with the COVID-19 pandemic years of 2020 and 2021. However, we observe with caution that in 2023 the number of recorded crime incidents surpassed pre-pandemic crime levels.

The GNR classifies crime incidents into seven primary categories which can be further divided into subcategories, resulting in a total of 179 subcategories. Table 2.2 presents the absolute values of reported crime incidents in 2022 and 2023 as well as the absolute and relative year-to-year changes per crime category.

Crimes against property (which includes theft, robbery and, damage to property, among others) represent more than half of the total crimes in both years. Saraiva et al. [52] had already noted in their description of crime from 2009-2019 that this is by far the most prevalent category of crime in Portugal, usually followed by crimes against persons (which includes violence, homicide and rape). Both of these categories saw a rise in the number of incidents in 2023 compared to 2022, although they did not have the highest

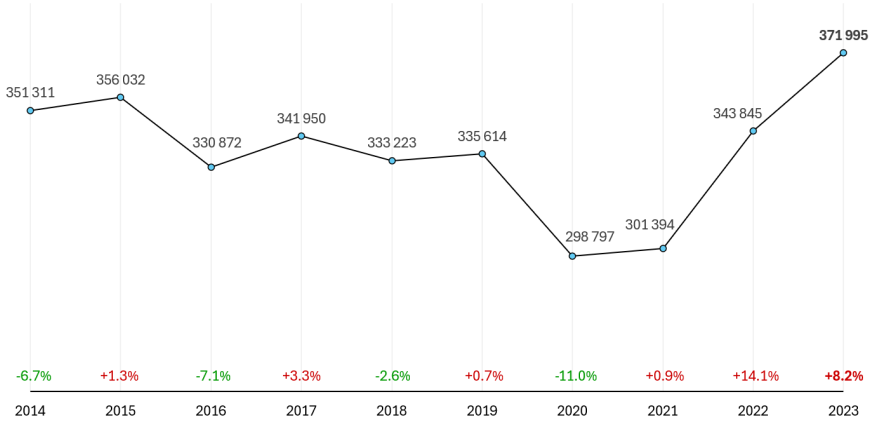


Figure 2.1: Recorded crime incidents from 2014 to 2023 in Portugal. Adapted from [51].

Table 2.2: Variation in categories of crime in Portugal during 2022-2023. Adapted from [51].

Crime categories	Year 2022	Year 2023	Var %	Diff
Crimes against persons	85,841	90,840	5.8%	4,999
Crimes against property	174,268	189,657	8.8%	15,389
Crimes against cultural identity and personal integrity	289	367	27.0%	78
Crimes against life in society	44,015	44,439	0.96%	424
Crimes against the state	5,593	7,713	17.0%	1,120
Crimes against companion animals	2,022	1,729	-14.5%	-293
Crimes under other legislation	30,817	37,250	20.9%	6,433

values in relative change. Only the crimes against companion animals marks a decrease in incidence. Crimes against cultural identity and personal integrity had a significant relative change increase in 2023, however this category has the lowest registered absolute counts.

In general geographic terms, Saraiva et al. [52] note in their 2009-2019 crime analysis a stable pattern of criminality. It is primarily concentrated in the coastline of mainland Portugal, which also contains most of the population. The Lisbon and Porto metropolitan areas report the highest concentrations, followed by the coastline of Algarve and Alentejo. Almada is very close to Lisbon, in fact, it is located in the other side of the river Tejo, which divides the two regions. We also emphasise that consulting the annexes of the 2022 [53] and 2023 [51] Annual Internal Security Reports, we verified Almada had the highest number of crime incidents in the district of Setúbal.

2.3.2 State of the art

The literature on the geography of crime in Portugal is scarce and no works were encountered performing statistical modelling on point pattern data, only at most on areal data at municipality level. One of the reasons for this gap in research, as mentioned by Tavares and Costa [54], could be the lack of open data with spatial resolutions beyond regional divisions. Indeed, our study was only possible through a collaboration with the GNR and precautions were taken due to data privacy concerns, among them, not

including in this work any image that would identify precise crime locations.

We briefly mention three studies and one master's thesis. Saraiva and its collaborators focused in the city of Porto, following approaches based on machine learning to monitor and predict crime [55] or visual data analytics using GIS to explore the relationship between street crime and greenspaces [56]. Tavares in her master's thesis [57] had used classical Poisson-based regression models on property crimes in mainland Portugal and later a paper on the topic was also published [54]. No studies were found related to point pattern analysis, Bayesian statistics, or LGCP models for crime in Portugal, reinforcing the innovative character of this study.

Saraiva and Teixeira [56] found that almost half of street crime (46%) occurred within a 5-minute walking distance of greenspaces. Smaller inner-city urban gardens experienced crime hotspots while larger municipal parks had lower crime densities, which suggests these public spaces could constitute interesting POIs in a statistical model.

Saraiva et al. [55] report that in the city of Porto crime is positively associated with factors such as high street population (especially female), concentrated disadvantage, and high-risk juveniles. Conversely, higher education levels, a larger male population, and surveillance systems presence tend to reduce crime. The influence of building density and dwelling concentration is mixed, depending on the context and method used.

Tavares and Costa [54] observed spatial non-stationarity in property crime determinants across municipalities, with coefficients varying locally. Young age groups have negative associations with crime in northern municipalities, but positive ones in Lisbon and the South, supporting routine activity theory. Drop out rates are positively linked to property crimes, particularly in Northwest municipalities, aligning with social disorganisation theory. Conventional dwellings show a positive association with crime in coastal and rural areas, possibly due to vacant houses. Poverty, measured by income assistance, is positively related to property crime in Lisbon and Northwest municipalities, while income inequality has mixed effects. Unemployment rates are generally negatively associated with crime, though some areas show positive effects, suggesting long-term offender motivational impacts. The authors recognise that spatial aggregation of crimes may mask heterogeneity at lower geographic units.

We have inspired ourselves in the factors used in these works to decide candidates to covariates in our own models, as well as others that could be relevant already mentioned throughout the literature review.

Chapter 3

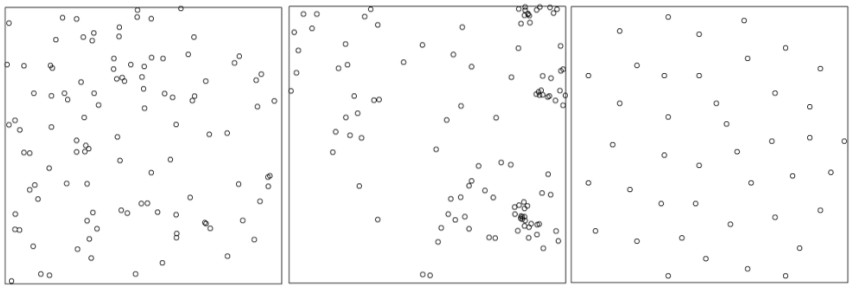
Point pattern analysis basics

The goal in this chapter is to introduce the essential concepts of point pattern analysis, laying the foundations for the next chapters where we focus on our case-study and concepts that concern the methodology, namely estimating point process properties non-parametrically and parametrically (modelling).

3.1 Spatial point pattern

The main dataset for this project consists of a *spatial point pattern*, $\boldsymbol{x} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$, which is a collection of countable spatial locations, $\boldsymbol{x}_i$, called the events (e.g., crimes occurrences) in a set $W \subset \mathbb{R}^2$ designated the study window [58].

It is possible to distinguish three types of spatial point patterns based on the disposition of events in space [59, 60]. Let us follow Figure 3.1, which shows simple examples where the three cases can be easily distinguish visually. In scenario (a) we simulated a *random* point pattern, where visually the events appear distributed completely at random. Formally, this means the point pattern adheres to the *complete spatial randomness* (CSR) hypothesis, whereby events occur independently of one another and there is an equal probability of an event surging in any location. Now, point patterns can deviate from randomness, originating (b) *aggregated* or (c) *regular* or point patterns.



(a) Random (b) Aggregated (c) Regular

Figure 3.1: Examples of different point patterns: (a) random, (b) aggregated and (c) regular, generated using the package `spatstat`.

3.2 Spatial point process

3.2.1 Definition

The locations and number of events are not known *a priori*, but rather observed quantities, and the interest resides in understanding how these events are generated. Thus, a spatial point pattern $\boldsymbol{x} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$ is treated as a realization of a *spatial point process* $\boldsymbol{X} = \{X_1, \ldots, X_{N(W)}\}$, which is a stochastic process where $X_1, \ldots, X_{N(W)}$ and $N(W)$ are random variables. For a bounded region $B \subset W$, the number of events from $\boldsymbol{X}$ that might fall in B is also a random variable denoted by $N(B \cap \boldsymbol{X})$ [58]. The distribution of the spatial point process is completely specified by the joint distributions of $N(B_1 \cap \boldsymbol{X}), \ldots, N(B_k \cap \boldsymbol{X})$ for all possible partitions of the study window into a sequence of k bounded sets $(B_1, \ldots, B_k)$ [60].

We note that in most real applications, as is the case of crime, we are in fact considering a *finite* spatial point pattern, for which $\boldsymbol{x}$ has a finite number of points in any bounded region $B \in W$ [14]. In practice, this translates to maintaining a special awareness for possible edge effects [59, 60]. Point patterns may also contain additional information in the form of continuous or categorical attributes, referred to as *marks*, associated with each point. In this case we say we are dealing with a *marked* point patterns, to which correspond a *marked* point process [14, 60].

3.2.2 Properties

This point process framework provides a means to statistically analyse the point pattern, and that analysis revolves around two key properties of the point pattern: intensity and inter-point interaction. The first describes the expected number of points per unit area while the latter focuses on the dependencies between points at different locations [14].

Consider a spatial point process $\boldsymbol{X}$ and the expected number of points, $\mathbb{E}[N(B_{\boldsymbol{u}} \cap \boldsymbol{X})]$, in a small region $B_{\boldsymbol{u}}$, of area $|B_{\boldsymbol{u}}|$, around an arbitrary location $\boldsymbol{u} \in B_{\boldsymbol{u}} \subset W$. The first-order *intensity function* (or simply, *intensity*) λ, of the spatial point process is defined by

$$\lambda(\boldsymbol{u}) = \lim_{|B_{\boldsymbol{u}}| \to 0} \frac{\mathbb{E}[N(B_{\boldsymbol{u}})]}{|B_{\boldsymbol{u}}|}, \quad \forall \boldsymbol{u} \in W. \tag{3.1}$$

The limit makes the intensity function a local measure, quantifying the potential of an event to occur at any location. The absolute value of the intensity gives an unscaled measure of point density while the way how it changes across the study area gives us an idea of spatial trend. A point process is said to be *homogeneous* if its intensity is constant and *inhomogeneous* otherwise.

By integrating the intensity function over a bounded region $A \subset W$ we obtain the expected number of events in that region:

$$\mathbb{E}[N(A)] = \int_A \lambda(\boldsymbol{u}) \mathrm{d}\boldsymbol{u}. \tag{3.2}$$

Inter-point interaction is included in the second-order and higher-order intensity functions. Let us consider now two small regions B_u and B_v, of areas $|B_u|$ and $|B_v|$, around two arbitrary locations $\boldsymbol{u}, \boldsymbol{v} \in B_u, B_v \subset W$. The *second-order intensity function*, λ_2, is defined by

$$\lambda_2(\boldsymbol{u}, \boldsymbol{v}) = \lim_{|B_u| \to 0, |B_v| \to 0} \frac{\mathbb{E}[N(B_u)N(B_v)]}{|B_u||B_v|}, \quad \forall \boldsymbol{u}, \boldsymbol{v} \in W. \tag{3.3}$$

Higher order intensity-functions are of increasingly difficult interpretation and as such are not explored very exhaustively, leading us to focus on the first two. It is also important to notice that equation 3.3 includes the effect of the first-order intensity function. In order to isolate inter-point dependence, we assess this property through the *covariance density function*, $\text{Cov}(\boldsymbol{u}, \boldsymbol{v})$, expressed as

$$\text{Cov}(\boldsymbol{u}, \boldsymbol{v}) = \lambda_2(\boldsymbol{u}, \boldsymbol{v}) - \lambda(\boldsymbol{u})\lambda(\boldsymbol{v}), \tag{3.4}$$

or through the *pair correlation function*, $g(\boldsymbol{u}, \boldsymbol{v})$, defined as

$$g(\boldsymbol{u}, \boldsymbol{v}) = \frac{\lambda_2(\boldsymbol{u}, \boldsymbol{v})}{\lambda(\boldsymbol{u})\lambda(\boldsymbol{v})}. \tag{3.5}$$

Both of these quantities can be easily related by

$$\text{Cov}(\boldsymbol{u}, \boldsymbol{v}) = \lambda(\boldsymbol{u})\lambda(\boldsymbol{v}) \left(g(\boldsymbol{u}, \boldsymbol{v}) - 1 \right). \tag{3.6}$$

Inter-point dependence can induce attraction or repulsion between points. In Table 3.1, we summarize how these factors are reflected in the covariance density function and the pair correlation function.

Table 3.1: Comparison of covariance and pair correlation functions in terms of spatial dependence at arbitrary locations $\boldsymbol{u}, \boldsymbol{v} \in W$.

Covariance Function $\text{Cov}(\boldsymbol{u}, \boldsymbol{v})$	**Pair Correlation Function** $g(\boldsymbol{u}, \boldsymbol{v})$	**Interpretation of Spatial Dependence**
$\text{Cov}(\boldsymbol{u}, \boldsymbol{v}) > 0$	$g(\boldsymbol{u}, \boldsymbol{v}) > 1$	*Clustering*: Points are more likely to co-occur at $\boldsymbol{u}$ and $\boldsymbol{v}$ than expected under independence.
$\text{Cov}(\boldsymbol{u}, \boldsymbol{v}) = 0$	$g(\boldsymbol{u}, \boldsymbol{v}) = 1$	*Complete Spatial Randomness (CSR)*: No dependence; points occur independently at $\boldsymbol{u}$ and $\boldsymbol{v}$.
$\text{Cov}(\boldsymbol{u}, \boldsymbol{v}) < 0$	$g(\boldsymbol{u}, \boldsymbol{v}) < 1$	*Repulsion*: Points are less likely to co-occur at $\boldsymbol{u}$ and $\boldsymbol{v}$ than expected under independence.

We should alert that deviations from CSR can also arise due to a non-constant intensity function, in which case the point pattern is generated by an inhomogeneous point process. It is important to be aware that in real-world scenarios, it is often challenging to isolate the first order and second order effects, and it is therefore prudent to assume that both may

coexist. For example, in the context of crime events, offences may occur more frequently in economically disadvantaged neighbourhoods or poorly lit areas due to environmental factors. At the same time, criminology theory suggests that repeat offences are likely to occur near previous crime locations as a result of offender behaviour, which reflects interaction between events.

3.2.3 Assumptions

Several assumptions about the point process may be established *a priori* to aid analysis, as long as they are reasonable for the application at hand. For instance, a point process is *orderly* if it produces point patterns such that the probability of observing more than one point in any infinitesimally small region is zero [60]:

$$\lim_{|B|\to 0} \frac{\mathrm{P}(N(B) > 1)}{|B|} = 0. \tag{3.7}$$

In practical terms, this implies that the observed point pattern does not contain coincident points.

Two other important assumptions are related to how a point process reacts to geometric transformations. A spatial point process is said to be *strongly stationary* if its properties are invariant under all translations in $\mathbb{R}^2$. Specifically, this means that the distribution of $\boldsymbol{X}$ is the same as the distribution of the shifted process $\boldsymbol{X} + \boldsymbol{m}$, for any vector $\boldsymbol{m} \in \mathbb{R}^2$. Equivalently, we can say that the joint distribution $(N(B_1), \ldots, N(B_k))$ has to be statistically identical to the joint distribution obtained after any translation of the sets $(B_1, \ldots, B_k) \subset \mathbb{R}^2$ [14, 60].

When a point pattern is said to be stationary, however, we usually refer to a weaker form of stationarity. A spatial point pattern $\boldsymbol{X}$ is *weakly stationary* if the first-order and second-order properties of the process are invariant under translations. Therefore $\lambda(\boldsymbol{u}) = \lambda$, $\mathrm{Cov}(\boldsymbol{u}, \boldsymbol{v}) = \mathrm{Cov}(\boldsymbol{h})$ and $\lambda_2(\boldsymbol{u}, \boldsymbol{v}) = \lambda_2(\boldsymbol{h})$, where $\boldsymbol{h} = \boldsymbol{u} - \boldsymbol{v}$ and $\boldsymbol{u}, \boldsymbol{v} \in W$ [59, 60].

If a process is *isotropic*, its properties remain invariant under rotations in $\mathbb{R}^2$. In the case a process is simultaneously (strongly or weakly) stationary and isotropic, $\lambda(\boldsymbol{u}) = \lambda$, meaning the process is homogeneous, $\lambda_2(\boldsymbol{u}, \boldsymbol{v}) = \lambda_2(r)$ and $\mathrm{Cov}(\boldsymbol{u}, \boldsymbol{v}) = \mathrm{Cov}(r) = \lambda_2(r) - \lambda^2$, where $r = \|\boldsymbol{h}\|$ [59].

If both first-order and second-order properties remain invariant under translations and are both expressed in terms of differences between locations, the spatial point process is *intrinsically stationary* [60]. Finally, we can also have an *intensity-reweighed stationary* point process, in which the pair correlation function only depends on the distance between points while the intensity function may vary spatially [59, 60]. In this case, we only impose that $g(\boldsymbol{u}, \boldsymbol{v}) = g(r)$.

Lastly, a point process is *independent* if the number of event occurrences in disjoint regions is independent. The counterpart to this is stochastic dependence, where the occurrence of an event conditions the probability of other events occurring nearby [14, 60].

Independent point processes have a covariance function equal to 0 and a pair correlation function of 1, as showed previously in Table 3.1.

3.2.4 The K-function and the L-function

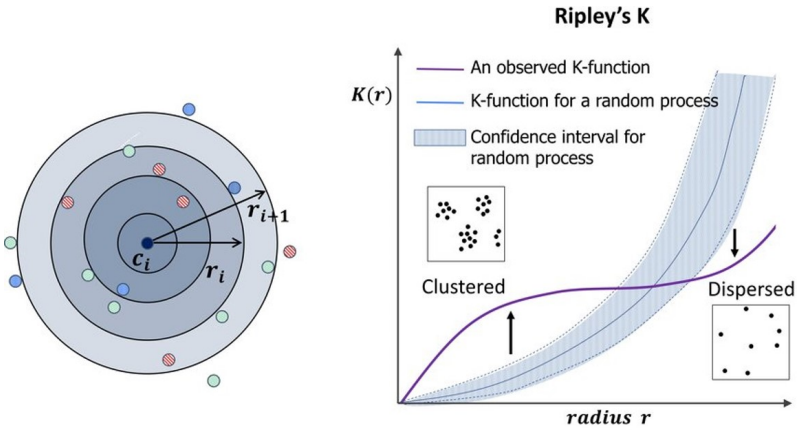
Ripley's K-function can be defined under the assumption of stationarity and isotropy as [58, 60]

$$K(r) = 2\pi\lambda^{-2} \int_0^r \lambda_2(\boldsymbol{u})\boldsymbol{u}\mathrm{d}\boldsymbol{u} \tag{3.8}$$

and if we consider $N_0(r)$ as the number of events at distance r or less from an arbitrary event at $r = 0$, then it can be equivalently defined as

$$K(r) = \lambda^{-1}\mathbb{E}[N_0(r)]. \tag{3.9}$$

Equation 3.9 provides a clear interpretation: if events tend to aggregate in a pattern, then the expected number of events close to an arbitrary event will be relatively high for small values of r. On the other hand, if the expected number of events for small values of r is small, that is an indication that the events are most likely surrounded by empty space. These behaviours are illustrated graphically in Figure 3.2. The K-function for a process obeying CSR is $K(r) = \pi r^2$, the area of a circle of radius r. In this case, the expected number of events within radius r of an arbitrary point is $\lambda\pi r^2$, scaling only with the area of the circle, simply based on chance and without any interactions.



.

Figure 3.2: Comparing the observed Ripley's K-function plot with the one of a random process can serve as a tool to evaluate point pattern aggregation or dispersion (Picture from Abousamra et al. [61])

Assuming also orderliness, the K-function is related to the pair correlation function through, [59],

$$g(r) = \frac{K'(r)}{2\pi r}, \tag{3.10}$$

making it easier to obtain one function from the other.

Ripley's K function can be modified to incorporate non-stationarity and anisotropy [14], however interpretation will be more delicate. For an intensity-reweighted stationary process, Diggle [59] suggests the extension of the K-function to such cases proposed by Baddeley, Møller and Waagepetersen,

$$K(t) = 2\pi \int_0^t g(\boldsymbol{u})\boldsymbol{u}\mathrm{d}\boldsymbol{u}. \tag{3.11}$$

As mentioned before, we can also use the *L-function*, created by applying a transformation that centres the K-function,

$$L(r) = K(r) - \pi r^2, \tag{3.12}$$

or stabilizes its variance, in which case we define,

$$L(r) = \sqrt{\frac{K(r)}{\pi}} \tag{3.13}$$

The idea here is to discount the effect of the CSR hypothesis K-function for an easier comparison with that baseline.

3.3 Poisson Point Process

The CSR hypothesis introduced earlier is, in fact, the simultaneous assumption of homogeneity and independence. Together, these imply that the probability of finding a point at any location within the window is constant. Baddeley et al. [14] derive from these simple principles that a point pattern obeying CSR is necessarily a realisation of a Homogeneous Poisson Point Process, which can be defined as follows:

Definition 3.1. Let $N(B \cap \mathbf{X})$ be the number of points in region $B \subset W \subseteq \mathbb{R}^2$. A (spatial) *Homogeneous Poisson Point Process* with intensity $\lambda > 0$ is a point process $\mathbf{X}$ with the following properties:

(1) Homogeneity: $N(\mathbf{X} \cap B)$ has a mean value proportional to the area of region B,

$$\mathbb{E}(\mathbf{X} \cap B) = \lambda |B|$$

(2) Independence: if we consider k non-overlapping regions $(B_1, ..., B_k)$ then the random variable $(N(B_1 \cap \mathbf{X}), \ldots, N(B_k \cap \mathbf{X}))$ are independent.

(3) Poisson-distributed counts: $N(B \cap \mathbf{X}) \sim \text{Poisson}(\mathbb{E}(\mathbf{X} \cap B))$

This process, although rarely observed in reality, is central to point pattern analysis. It serves as a null hypothesis model and is the basis for constructing more complex point processes. For instance, if the intensity of the Poisson Point Process can vary, we get an Inhomogeneous Poisson Point Process.

Definition 3.2. Let $N(B \cap \mathbf{X})$ be the number of points in region $B \subset W \subseteq \mathbb{R}^2$. An (spatial) *Inhomogeneous Poisson Point Process* with intensity $\lambda(\boldsymbol{u}) > 0$ is a point process $\mathbf{X}$ with the properties of independence and Poisson-distributed counts of the Homogeneous Poisson Point Process, but a spatially varying intensity function:

$$\mathbb{E}(B \cap \mathbf{X}) = \int_B \lambda(\boldsymbol{u}) \mathrm{d}\boldsymbol{u}$$

One thing, however, is always common to Poisson Processes, whether homogeneous or inhomogeneous: they do not account for inter-point interactions, as they are defined from the assumption of independence between points.

3.4 Log-Gaussian Cox Process (LGCP)

The Log-Gaussian Cox Process (LGCP) is a cluster inducing point process, built from a Cox Process and a Gaussian Random Field.

A Cox Process is a doubly stochastic process in the sense that, in addition to the point process itself being stochastic, the intensity function of the process is also modelled as a stochastic process [35].

Definition 3.3. A (spatial) *Cox Process* is a spatial point process $\mathbf{X}$ defined by the following postulates:

(1) The intensity follows a nonnegative-valued stochastic process $\Lambda = \{\Lambda(\boldsymbol{u}) : \boldsymbol{u} \in \mathbb{R}^2\}$.

(2) Conditional on the realisation $\Lambda(\boldsymbol{u}) = \lambda(\boldsymbol{u})$, $\boldsymbol{u} \in \mathbb{R}^2$, $\mathbf{X}$ is an Inhomogeneous Poisson Point Process with intensity $\lambda(\boldsymbol{u})$.

The second element is the Gaussian Random Field.

Definition 3.4. A 2-dimensional *Gaussian Random Field* is a random function from $S : W \subseteq \mathbb{R}^2 \rightarrow \mathbb{R}$ with the following properties:

(1) $S = \{S(\boldsymbol{u}) : \boldsymbol{u} \in W\}$ is a real-value stochastic process.

(2) For any integer $n \geq 1$ and any finite set of locations $(\boldsymbol{u}_1, ..., \boldsymbol{u}_n)$ the corresponding random vector $(S(\boldsymbol{u}_1), ..., S(\boldsymbol{u}_n))$ follows a multivariate normal (Gaussian) distribution of dimension n.

The LGCP is defined as a Cox Process in which the logarithm of the stochastic intensity is modelled by a Gaussian Random Field [24]:

Definition 3.5. A *Log-Gaussian Cox Process* is a Cox Process with intensity $\Lambda(\boldsymbol{u}) = e^{S(\boldsymbol{u})}$, where S is a Gaussian Random Field.

The LGCP is completely specified by its mean and covariance functions [14], which it inherits form the corresponding Cox Process. Note that the use of the logarithm in the LGCP can be interpreted as a log-link function, which transforms values from a real

scale to a positive real-valued scale, ensuring that the intensity remains positive. S is also frequently designated the *Latent Gaussian Field*. We should recognize that in this framework, given the underlying latent field, the point pattern generated derives from a Poisson Process.

The intensity function of the LGCP can be modelled using generalized linear models or generalized additive models. Moreover, inter-point interaction can be indirectly introduced through the covariance function of the Gaussian Random Field. Diggle et al. [24] describe a simple reparametrisation of the LGCP intensity function. An intercept term β_0 as well as a set of explanatory covariates, $\boldsymbol{z}(\boldsymbol{u}) = \{z_1(\boldsymbol{u}), \ldots, z_j(\boldsymbol{u})\}$, and corresponding coefficients vector, $\boldsymbol{\beta} = \{\beta_1, \ldots, \beta_j\}$, is introduced to model the intensity of the LGCP, in such a way that the Gaussian Random Field represents deviations around the mean value:

$$\Lambda(\boldsymbol{u}) = \exp\{\beta_0 + \boldsymbol{z}(\boldsymbol{u})\boldsymbol{\beta} + S(\boldsymbol{u})\} \tag{3.14}$$

where $\mathbb{E}[S(\boldsymbol{u})] = -0.5\sigma^2$, so that $\mathbb{E}[e^{S(\boldsymbol{u})}] = 1$ and therefore $\lambda(\boldsymbol{u}) = e^{\beta + z(\boldsymbol{u})\boldsymbol{\beta}}$. The end result is an *intensity-reweighted stationary Cox point process* corresponding to the "analogue of a real-valued process with a spatially varying mean and a stationary residual" [24].

Consider the simplest case, without explanatory covariates. The intensity reduces to: $\Lambda(\boldsymbol{u}) = e^{\beta_0 + S(\boldsymbol{u})}$, where β_0 controls the overall mean intensity and the Gaussian Random Field, $S(\boldsymbol{u})$ introduces local deviations around this mean. When covariates are included, the term $\beta_0 + \boldsymbol{z}(\boldsymbol{u})\boldsymbol{\beta}$ defines a spatially varying mean intensity. As before, $S(\boldsymbol{u})$ acts as a stationary residual capturing unobserved spatial structure. The reparametrisation ensures that the Gaussian Random Field does not affect the marginal intensity, $\lambda(\boldsymbol{u})$, allowing covariate effects and spatial dependence to be clearly separated.

Chapter 4

Pre-modelling stage

Let us start diving into our case-study: street crime in Almada in the areas under the jurisdiction of the GNR. The main objective here is to introduce the data. We define the working dataset from raw data and clarify the study domain through data pre-processing. Next, we conduct an exploratory data analysis to gain an initial sense of the dataset's characteristics and patterns. In point patter analysis, this usually encompasses descriptive statistics summaries and also non-parametric estimations of point process properties, that can help us guide model choices.

4.1 Crimes dataset

The data used in this project consisted of a series of crime records, kindly provided by the GNR's information division. These pertain to crimes investigated by the Almada Territorial Unit of the GNR, with legal proceedings initiated between 2022 and 2023.

Before processing the data, it is essential to define the spatial and temporal scope of the study. Portugal is administratively divided into districts, which are further subdivided into municipalities (*municípios*) and parishes (*freguesias*), forming the basis of local governance. In 2013, a nationwide administrative reform led to the merging of many *freguesias*, resulting in lengthy names that reflect the union of two or more parishes now counted as a single administrative unit. The municipality of Almada comprises five *freguesias*: the Union of *Almada, Cova da Piedade, Pragal e Cacilhas*; the Union of *Laranjeiro e Feijó*; the Union of *Caparica e Trafaria*; the Union of *Charneca de Caparica e Sobreda*; and the *freguesia* of *Costa da Caparica*. Only the last three fall under the jurisdiction of the GNR, while the remaining two are policed by the *Polícia de Segurança Pública* (PSP). Consequently, the study area was limited to the regions covered by the *freguesias* of *Caparica e Trafaria*, *Charneca de Caparica e Sobreda*, and *Costa da Caparica*.

Since we were working with georeferenced data, it was crucial to decide on an appropriate *Complete Spatial Randomness (CSR)*. A CSR consists of a *datum*, a coordinate system, and a projection. The *datum* is a mathematical model that specifies the shape, size, and origin of the Earth, enabling the use of coordinates to describe any position in three-dimensional

space. The coordinate system determines how coordinates are expressed, latitude and longitude in degrees being the most common. Lastly, the projection allows us to visually represent these locations on a two-dimensional surface, such as a map. In `R`, a CSR can be specified using an EPSG code, a unique identifier available for the most common CSRs.

The original dataset provided the longitude and latitude coordinates for each crime occurrence using the standard World Geodesic System 1984 (WGS84) *datum*. However, a different CSR was chosen, one that is recommended for mainland Portugal: PT-TM06, which is based on the ETRS89 *datum* and constructed using the Transverse Mercator projection [62]. In PT-TM06/ETRS89, coordinates are expressed in meters, representing distances from a false origin located near the geographic center of Portugal. Because the municipality of Almada is situated to the south-west of this origin, all coordinates within this area have negative values.

Spatial data was handled using the `sf` package [63, 64], which represents spatial objects, such as points, lines, and polygons, as simple features (`sf`) along with their attributes, stored in a data frame-like structure. A shapefile containing the boundaries of all *freguesias* in Portugal was used to define the study window. As it was already provided in the PT-TM06/ETRS89 CSR, no coordinate transformation was necessary. Then, the data was read into an `sf data.frame` object using function `st_read()` and the *freguesias* of interest were selected and merged with function `st_union()` resulting in an `sfc_POLYGON` object. The study window was saved as an `sfc_POLYGON` object to a file for future use. It is plotted in Figure 4.1 for reference.

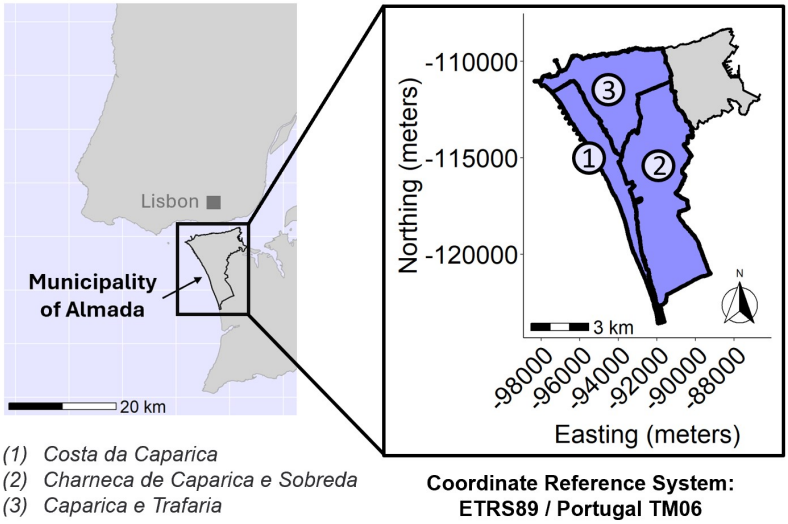


Figure 4.1: Study window (region in darker blue shade). The subdivisions correspond to *freguesias* of Almada.

Regarding the study time period, we will focus on the crime occurrences between 2022 and 2023. Notice this is different from the date of initiation of crime proceedings. Crime proceedings initiated between 2022-2023 can refer to crimes that already happened months, if not years, before such legal actions take place. In addition, we alert that some

crime records are inevitably missing from the dataset. For instance, crimes that occurred in 2022–2023 but for which legal proceedings were only initiated in 2024–2025 are not included. Besides, it is possible that other police forces, particularly those operating near Almada, may have handled incidents that occurred within the municipality. Nevertheless, we consider the dataset to be broadly representative of criminal activity within the defined spatio-temporal domain. Recall also that, in any analytical context, the actual number of crime incidents is likely to exceed the number of recorded cases, due to the crimes that go unreported.

Finally, as indicated in the title of this work, the focus will be on street crimes. The GNR classifies all crime events into one of 7 main categories (Table I.1) and further into 179 subcategories. Following the classification of street crimes by Neves [65], and also used by Saraiva et al. [56], we consider as street crimes the thirteen subcategories: (1) bank robbery or robbery of another financial establishment, (2) damage to cultural property; (3) fire or arson in buildings, constructions, or means of transport, (4) other damage; (5) pickpocketing, (6) robbery in public transports, (7) robbery in the street, (8) service station robbery, (9) theft in a supermarket, (10) theft in motor vehicle, (11) theft of motor vehicle, (12) treasury or post office robbery, (13) wallet theft. The original designations can be found in annex in Table I.2.

With the exception of fire or arson in buildings, constructions, or means of transport, which is included in crimes against life in society category, all the others are classified as crimes against property. The crime *of* theft of motor vehicle concerns the theft of the vehicle itself while the theft *in* motor vehicle is related to the theft of objects from or in the vehicle.

4.2 Data pre-processing

The original dataset consisted of 52 columns and 9849 rows. We began by looking at each column and annotating in an excel file a description, range of values, percentage of missing values, some basic descriptive statistics and/or plots and, an assessment of its usefulness. This metadata, which we will not be discussing in detail here, allowed us to conclude that only five columns would be useful for this study: date of occurrence, latitude, longitude, crime category and crime subcategory.

The data pre-processing step required some auxiliary information such as the list of categories and subcategories of crimes and the study window file. We loaded all these necessary files plus the excel crimes data file into `R`. We then excluded three rows corresponding to crimes handled by the Polícia Judiciária (PJ) instead of the GNR, excluded a few rows which had no coordinates and selected only the five relevant columns, renaming them in the process.

A `crimes_sf` variable was created, of class `sf data.frame`. The CSR was specified using code ESPG:4326, which corresponds to *datum* WGS84 for latitude and longitude coordinates, as this is the original CSR of the coordinates in the crimes excel file.

We then converted transformed the `crimes_sf` into the desired CSR, PT-TM06/ETRS89 using code ESPG:3763 and fixed the time stamp column data type, converting its values into `date_format YYYY-MM-DD hh:mm` through function `as.POSIXct()`, which returns of objects of class `POSIXlt` and `POSIXct`, representing calendar dates and times. In addition, the `crimetypes` data frame was translated to English and then used to translate the `category` and `subcategory` columns. We then began selecting the rows of interest, which were the ones whose

- dates of occurrence were between 2022 and 2023;
- locations were within the borders of the study window;
- crime subcategories correspond to street crimes, whose names are storing in a list called `dict_street_crimes`.

Analysing the `crimes_sf` coordinates we verify there are many duplicates, i.e., points with the same exact coordinates. This is a problem for building models based on Poisson point processes, such as the LGCP, which exist under the assumption of orderliness, which assumes only one event per exact location. We can intuitively grasp the computational errors that could arise since one of the consequence of multiple points in the same location is a non-invertible covariance matrix in the underlying Gaussian process. As Chen et al. [66] recall the three main ways in which we can overcome it *ad hoc*: by eliminating them entirely, jittering the points slightly or introducing a small value in the diagonal of the covariance matrix to make it invertible. We opted for the second option to deal with the 794 out of 1426 points which were duplicates. We choose to jitter the duplicated points in relation to original within a circle of radius 0.5 m, following a uniform distribution. This distance seemed reasonable enough so that points are distinguishable without moving away from the street/block they originally belong to. We also checked no points went out of the study window after being jittered and did all this process setting a seed to have reproducible results.

We note that to change the coordinates of the original points into the jittered ones we have to do it in a normal data frame object and then reconvert it to a `sf data.frame` object again.

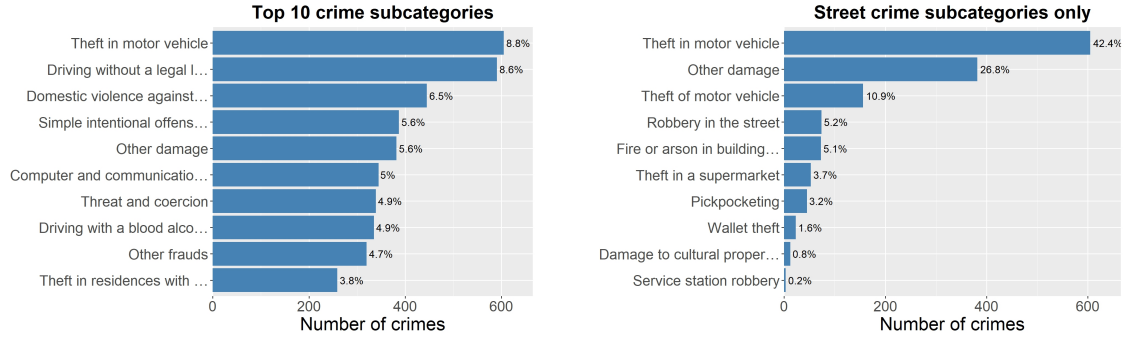
Lastly, we verified that no missing values were in place and that all columns had correct information. The pre-processing stage is of utmost importance. If the data used for analysis is not appropriate, that could compromise the whole study.

The final crimes dataset, `crimes_sf` is a `sf data.frame` with 1426 features, corresponding to `POINT` simple features representing each crime event location in meters in the PT-TM06/ETRS89 CSR, with no duplicated locations, and 3 fields, namely, time stamp, crime category and crime subcategory. We saved this object into a file as we had previously done with the study window.

4.3 Preliminary exploratory data analysis

We began by addressing some initial questions about the clean dataset, such as: Which street crimes are most frequent? What does the spatial point pattern look like? Are there any evident clusters? How is crime distributed over time, specifically across seasons, months, weeks, and hours?

Figure 4.2b shows that slightly more than 40% of street crimes within the study window during 2022-2023 were thefts in motor vehicles, with a total of 600 occurrences. Notice that looking at Figure 4.2a we verify this is also the most common subcategory within the study region from all the possible 176. The third most common type is also related to motor vehicles, specifically the theft of the vehicle itself, and corresponds to 10.9% of all street crimes. The second most common type is a very broad category called "other damage", encompassing close to 30% of street crimes. We also note that only 10 of the 13 possible street crime subcategories are present in our study, since there were no reports of treasury or post office robbery, bank robbery or robbery of another financial establishment, or robbery in public transports.



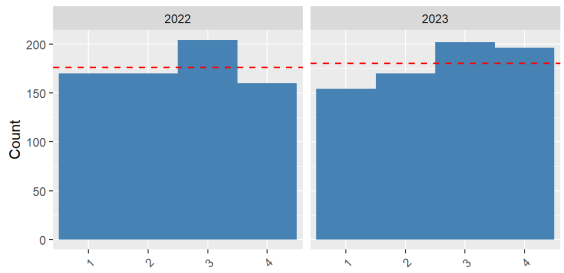
(a) Overall criminal activity in the study area.

(b) Street crimes in the study area.

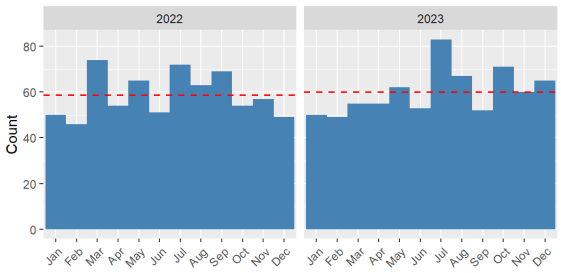
Figure 4.2: Absolute and relative frequency of crimes in 2022-2023: (a) top 10 from all the registered crime subcategories and (b) considering only street crime subcategories.

The number of street crime events in the study window during 2022 was 704, while in 2023 there was an increase to 722. This follows the trend observed for the whole country, which saw an increase in overall crime rates, as described in Section 2.3.1. Looking at Figure 4.3 it is observed that the number of crimes increases in the 3rd quarter of the year both in 2022 and 2023, and maintained close to that increase in the last quarter of 2023. In 2022 the month with most crimes was March, followed by July and September, while in 2023 the highest value was in July followed by October and August. The month of February had the lowest number of occurrences in both years. In 2022, Mondays and Thursdays stand out with the highest and lowest accumulation of crime records. However, in 2023, the number of crimes is higher on Thursdays and more uniform throughout the week compared to the previous year. Lastly, it is evident that criminal activity is much less intense during the night, from 1 to 8 a.m. and in 2023 we distinguish at 6 p.m. a prominent peak.

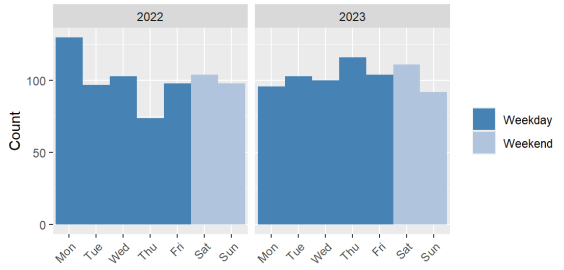
For confidentially reasons we do not show the point pattern in this dissertation. However, the reader will have opportunity in the next section to get a feel for the spatial distribution of crimes through the intensity maps.



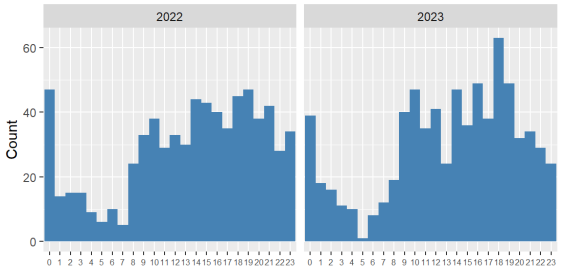
(a) Number of street crimes by quarter per year.



(b) Number of street crimes by month per year.



(c) Number of street crimes by day of the week per year.



(d) Number of street crimes by hour per year.

Figure 4.3: Number of street crimes using different time aggregations. When the red dashed line is present it indicates the yearly average calculated with the aggregated data.

4.4 Non-parametric estimation

We now proceed to a non-parametric estimation of the properties of the spatial point process underlying our point pattern, following Baddeley et al.'s [14] book, namely section II. These authors are the creators of `spatstat` [14], the most popular `R` library for conducting point pattern non-parametric analysis. Regarding the programming aspects, we followed the works of Spychala [67] and for tools on non-homogeneous spatio-temporal point patterns, we followed the methods in González and Moraga's recent article [26]. All these sources were very useful since they provide explanations alongside code snippets.

4.4.1 Tools for non-parametric estimation

4.4.1.1 Intensity estimation

If we assume the point pattern is generated by a Homogeneous Poisson Point Process, an unbiased estimation, $\hat{\lambda}$, of the true intensity, λ, can be obtained simply by dividing the total number of events $n(\boldsymbol{x}) = n(\boldsymbol{X} \cap W)$ by the area of the study window, $|W|$. However,

this is a very naive approach based on a very strong assumption that hardly holds in the real world.

We can instead perform *quadrat counts*, dividing the study area W into m subdivisions, called quadrats, $B_1, \ldots, B_m$. The question to ask ourselves is: do regions with equal areas contain roughly equal numbers of points? If the answer leans toward yes, then we can suspect homogeneity. Otherwise, we should discard that option. Assuming homogeneity, the estimated intensity will correspond to the average of these counts:

$$\hat{\lambda} = \frac{1}{m} \sum_{i=1}^{m} \frac{n(\boldsymbol{x} \cap B_i)}{|B_i|}, \tag{4.1}$$

and further assuming the counts per quadrat are independent and identically distributed variables, the standard error for the intensity estimate is given by

$$\mathrm{SE}(\hat{\lambda}) = \frac{\sqrt{\mathrm{Var}(\hat{\lambda})}}{m-1}. \tag{4.2}$$

A few words of caution should be said. The quadrat count procedure always implies a hidden trade-off between bias and variability. As Baddeley et al. [14] explain, if we increase quadrat size, the relative error (standard error divided by the mean) reduces, but at the cost of a loss of spatial variation in intensity within each quadrat. Quadrat shape is also relevant. Study windows might not be perfect squares, in which case not all quadrats have the same area, compromising the intensity estimates.

For a smooth estimation of the intensity function, the most common non-parametric option is kernel estimation. In fact, we estimate the probability density function via *Kernel Density Estimation (KDE)*, as this function can be related to the intensity function by the expression

$$\hat{\lambda}(\boldsymbol{u}) = n\, \hat{f}(\boldsymbol{u}), \tag{4.3}$$

where $\hat{f}(\boldsymbol{u})$ is the KDE of the observed points and n is the total number of events in the study window.

The idea is to place a kernel function κ (a smooth, symmetric function such as a Gaussian) at each observed data point and then sum these contributions. The smoothness of the resulting estimate is controlled by a bandwidth parameter $h > 0$, which determines the spread of each kernel $\kappa_h(\cdot)$,

$$\hat{\lambda}(\boldsymbol{u}) = \sum_{i=1}^{n} \kappa_h(\boldsymbol{u} - \boldsymbol{x}_i), \tag{4.4}$$

where, for the 2-dimensional spatial point pattern case,

$$\kappa_h(\boldsymbol{u}) = \frac{1}{h^2} K\left(\frac{\boldsymbol{u}}{h}\right). \tag{4.5}$$

We opted for the Gaussian kernel with an improved edge correction term, $e(\boldsymbol{x}_i)$, developed by Diggle [68] which corresponds to the convolution of the Gaussian kernel with the study window calculated at each data point $\boldsymbol{x}_i$:

$$\hat{\lambda}^{(D)}(\boldsymbol{u}) = \sum_{i=1}^{n} e(\boldsymbol{x}_i)\kappa_h(\boldsymbol{u} - \boldsymbol{x}_i), \tag{4.6}$$

with,

$$e(\boldsymbol{x}_i) = \frac{1}{\int_W \kappa_h(\boldsymbol{x}_i - \boldsymbol{v})\,\mathrm{d}\boldsymbol{v}}. \tag{4.7}$$

and

$$\kappa_h(\boldsymbol{z}) = \frac{1}{(2\pi)^{d/2}h^2}\exp\left(-\frac{\|\boldsymbol{z}\|^2}{2h^2}\right). \tag{4.8}$$

One should always bear in mind kernel estimators of the intensity function are slightly biased in general, because they smooth out details in the intensity function. The choice of bandwidth is critical, more than the shape of the kernel itself, and as such several methods have been developed by different authors to optimize this parameter. We experimented with three of the most popular ones:

- *Scott's bandwidth*: calculated using Scott's rule of thumb which states that $h \propto n^{-\frac{1}{d+4}}$, where n is the number of observed events and d the number of dimensions (in our case, $d = 2$).
- *ppl bandwidth*: chosen so as to maximise the point process likelihood cross-validation given by $\mathrm{LCV}(h) = \sum_i \log \hat{\lambda}_{-i} - \int_W \hat{\lambda}(\boldsymbol{u})\mathrm{d}(\boldsymbol{u})$.
- *Diggle's bandwidth*: minimises a mean squared error criterion defined by Diggle [68] to find the optimal bandwidth.

We also used an adaptive kernel estimator, as suggest by González and Moraga [26], based on a variable-bandwidth kernel estimator. We kept the same edge correction mentioned previously.

4.4.1.2 Second-order properties estimation

The tools here concern the estimation of the pair-correlation function and the K-function. We considered two cases: an estimation assuming stationarity and another allowing for non-constant intensity.

In the stationary case, the K-function is estimated using function `Kest()`, which according to the documentation provides estimates of the form

$$\hat{K}(r) \;=\; \frac{|W|}{n(n-1)} \sum_{i=1}^{n} \sum_{\substack{j=1 \\ j\neq i}}^{n} \mathbf{1}(d_{ij} \le r)\, e_{ij}(r) \tag{4.9}$$

where $|W|$ is the area of the study window, n is the number of observed events, and the sum is taken over all ordered pairs of points (i, j). Here d_{ij} is the distance between the two points, and $\mathbf{1}(d_{ij} \leq r)$ is the indicator function that equals 1 if the distance is less than or equal to r. The term e_{ij} is the edge correction term, as before. This time we utilized a Ripley's isotropic edge correction which adjusts for boundary effects by comparing the full circumference of a circle of radius r around a point to the portion of that circle that actually lies inside the observation window.

The pair-correlation function is using function `pcf()`, typically based in the expression, [14],

$$\hat{g}(r) = \frac{|W|}{2\pi r\, n(n-1)} \sum_{i=1}^{n} \sum_{\substack{j=1 \\ j \neq i}}^{n} \kappa_h(r - d_{ij})\, e_{ij}(r). \tag{4.10}$$

Alternatively, one could also estimate the pair-correlation function from an estimation of the K-function, following Equation 3.10.

In the inhomogeneous case, both function estimates require a calculation of the intensity for all pair of points. The K-function is now estimated by function `Kinhom()`, via the expression

$$\hat{K}_{\text{inhom}}(r) = \frac{1}{|W|} \sum_{i=1}^{n} \sum_{\substack{j=1 \\ j \neq i}}^{n} \frac{\mathbf{1}\{d_{ij} \leq r\}\, e_{ij}(r)}{\lambda(\boldsymbol{x}_i)\, \lambda(\boldsymbol{x}_j)}, \tag{4.11}$$

and the pair-correlation function utilising `pcfinhom()`. Although not explicit in the documentation, in analogy to the K-function, an estimation would follow the expression

$$\hat{g}_{\text{inhom}}(r) = \frac{1}{2\pi r\, |W|} \sum_{i=1}^{n} \sum_{\substack{j=1 \\ j \neq i}}^{n} \frac{\kappa_h(r - d_{ij})\, e_{ij}(r)}{\lambda(\boldsymbol{x}_i)\, \lambda(\boldsymbol{x}_j)}. \tag{4.12}$$

We also followed González and Moraga [26] for an estimation of the non-homogeneous spatio-temporal K-function and pair-correlation function, following closely their implementation. The expressions are similar to Equations 4.11 and 4.12, but now an event is characterised by two spatial coordinates and a time instant. Naturally, besides considering the spatial distance between events, one also considers the time interval between them.

4.4.1.3 Testing for inter-point dependence

Through quadrat counting it is possible to design a simple χ^2 test to assess the goodness of fit of the underlying point process to a Homogeneous Poisson Point Process of parameter $\lambda|W|$. The null hypothesis corresponds to the CSR hypothesis. Meanwhile, the alternative hypothesis assumes an inhomogeneous point process of some sort in a two sided test, or one that is regular or clustered in the one sided test version. This was implemented using function `quadrat.test()`.

To complement the more simple CSR quadrat test and overcome previously exposed issues of quadrat counting, we also applied the rank envelope composite test based on the descriptor $L(r) - r$, where $L(r)$ is given by Equation 3.13, as described by González and Moraga [26]. The null hypothesis supposes an Inhomogeneous Poisson Point Process with the same estimated intensity as the observed data, meaning no clustering is considered beyond what population distribution would explain. Notice that for a Poisson Process, $L(r) = r$, and so we center the function around zero under the null hypothesis when we plot $L(r) - r$.

Chosen the test statistic T one wishes to compare an empirical estimation T_{obs} for the observed pattern with the distribution of T under the null model, T_0. Monte Carlo tests are commonly employed as T_0 is often difficult to obtain or unknown. They allows to overcome this issue by simulating s test statistics T_i, $i = 1, ..., s$ under the null hypothesis and computing a Monte Carlo p-value.

In spatial point pattern analysis, test statistics are often functions of the distance r, such as $L(r) - r$. Consequently, statistical inference is typically based on simulation envelopes. The test statistic $T_0(r)$, with $r \in I$, a predefined interval, is treated as an entire curve under the null hypothesis. By generating s point patterns under the null model and computing the corresponding statistics $T_i(r)$, $i = 1, \ldots, s$, at a set of selected distances within the interval I, we obtain Monte Carlo simulations of the functional test statistic. [14]

To perform inference, lower and upper boundaries of an envelope must be defined. In point-wise envelopes, the boundaries at each distance r are determined independently by taking appropriate quantiles (or equivalently, the minimum and maximum) of the simulated values $\{T_i(r)\}_{i=1}^{s}$. Such envelopes assess extremeness separately at each distance and do not control the type I error simultaneously over the interval I. In contrast, global envelopes are constructed by treating each simulated curve $T_i(r)$ as a single object and determining which curves are globally extreme according to a joint extremeness criterion. The remaining, non-extreme curves then define the envelope over all distances r.

A principled approach for constructing global envelopes is provided by the rank envelope test, implemented through the `GET` library. In this method, extremeness is quantified using an extreme rank measure, which states that for each simulated curve $T_i(r)$, point-wise ranks are computed across the simulations at each distance r, and the minimum of these ranks over the interval I is taken as the curve's extreme rank. Curves that exhibit extreme deviations at any distance therefore receive small extreme ranks and are classified as globally extreme. Ordering the simulated curves by their extreme ranks allows the construction of a global envelope at a prescribed significance level, against which the observed function $T_0(r)$ can be compared. The rank envelope test thus yields both a formal Monte Carlo test with controlled global type I error and a graphical envelope that highlights distances at which departures from the null model occur.

In our case, we wanted to employ a formal test in addition to a graphical interpretation. A standard one-stage global envelope Monte Carlo test is not sufficient because our null hypothesis is composite; that is, it depends *a priori* on parameters that must be

estimated. Simple envelope tests are therefore invalid for summary statistics such as the pair correlation function, K-function, and L-function, which require estimation of the intensity function. To address this, we adopt the two-stage global envelope test proposed by González and Moraga [69].

In the two-stage procedure, the parameters of the null model are first estimated from the observed point pattern. Using these estimated values, a set of point patterns is simulated under the null hypothesis. For each simulated dataset, the model is re-fitted to obtain new parameter estimates, which are then used to generate realisations of the chosen summary function. Global envelopes are constructed from this collection of simulated curves, accounting not only for the natural variation under the model but also for the uncertainty introduced by parameter estimation. The observed summary function is then compared with these envelopes, and the null model is rejected if it falls outside.

As Diggle [59] mentions, there is a large amount of available tests, often involving functions associated with higher order point process properties relating to the spacing between points, such as the G-, F-, and J-functions. We decided not to dive into this domain, as non-parametric tests on non-stationary scenarios in general carry limited statistical weight.

4.4.2 Results

The quadrat count plots and kernel density estimated intensity maps (Figures 4.4-4.6) reveals several hotspots. We highlight the crime concentration in the north-east, at the coast line of *Costa da Caparica* and in the north-west corner of the study area, in *Caparica e Trafaria*. Two other high-concentration isolated locations (in dark-blue) stand out, especially in the adaptive kernel intensity estimate of Figure 4.7. It is important to mention that the dark spot on the right side corresponds in fact to a police post. To make sense of the magnitude itself of the intensity estimates, we recall this quantity measures the expected number of crimes per square meter, meaning an intensity of 0.001 corresponds to expecting one crime in an area of 1000 m^2 by the two years.

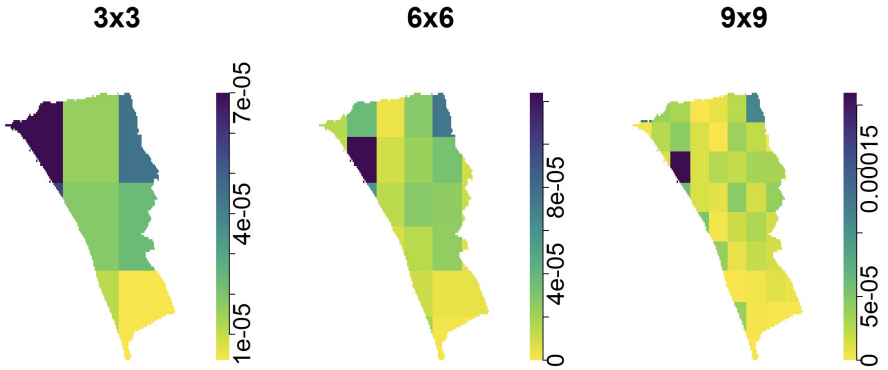


Figure 4.4: Quadrat counts using 3x3, 6x6 and 9x9 tiles.

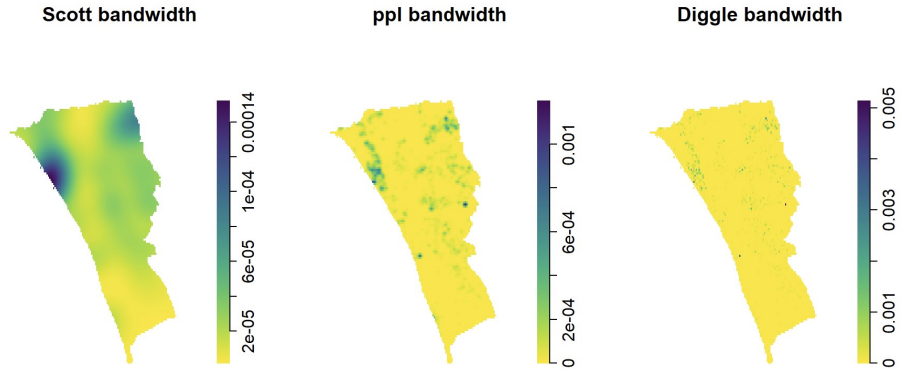


Figure 4.5: Kernel estimation of the intensity surface considering a fixed bandwidth calculated following the Scott, ppl and Diggle optimization procedures.

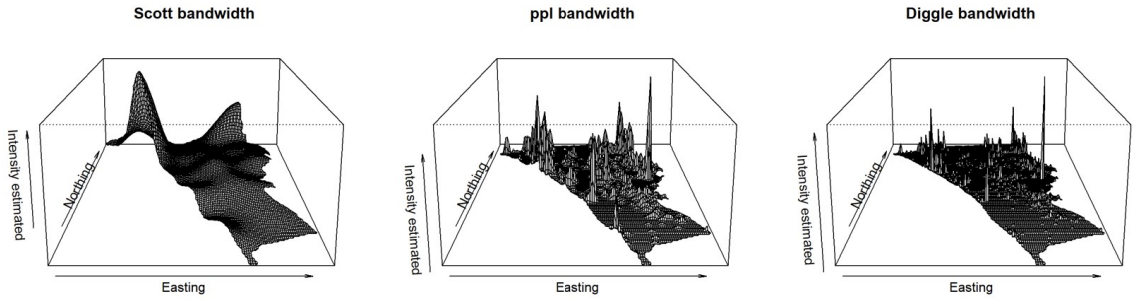


Figure 4.6: Visualization in 3D of the intensity estimates, following Spychala's work [67].

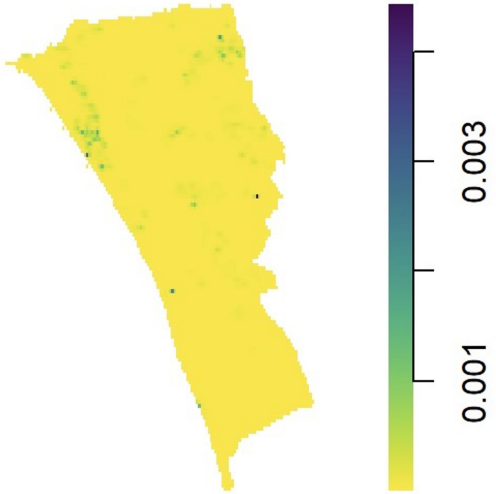
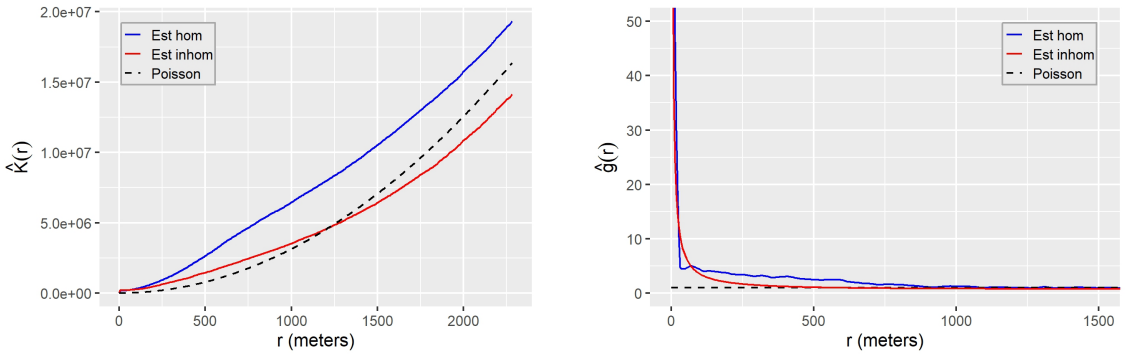


Figure 4.7: Adaptive kernel intensity estimate, using a variable bandwidth.

We have also included in annex II maps of estimated intensity using subsets of street crimes per quarter, day of the week and part of the day, to complete the descriptive analysis of Figure 4.3. We see that the two major hotspots are constant regardless of the temporal aggregation, although there seems to be more crime dispersion in the first quarters of

the year in other areas. In terms of the days of the week, some hotspots fade lightly on Tuesdays and Wednesdays, relative to other weekdays. Regarding the times of the day, we see a decrease of hotspot intensity from 8 a.m. to 16 p.m., while the highest levels of crime concentration appear during the night. It is interesting that crime absolute frequencies show an opposite tendency during these periods, as seen in Figure 4.3d, meaning that the period with more crime is not necessarily that with the tightest hotspots.



(a) Estimated K-function.

(b) Estimated pair-correlation function.

Figure 4.8: Estimated second-order properties. The blue line represents the estimate assuming a stationary point process, while the red line correspond to the estimate performed in the scenario of an inhomogeneous point process. The black dashed line corresponds to the Poisson point process.

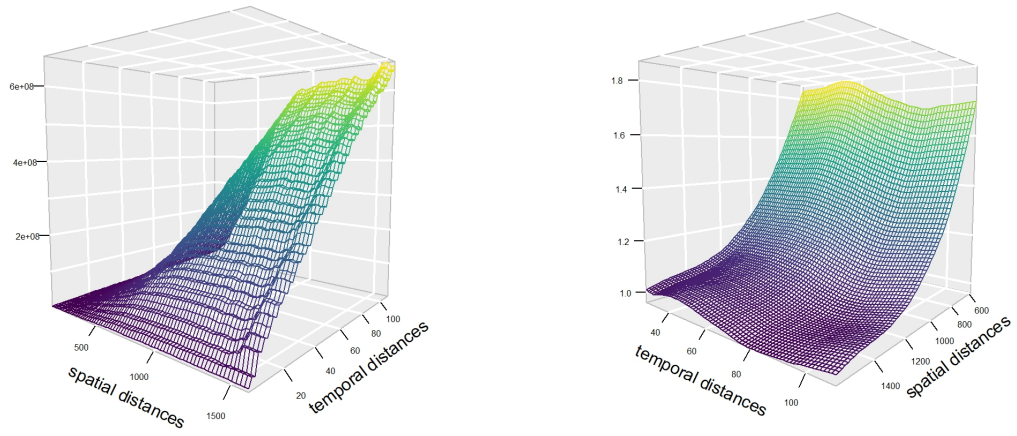
In Figures 4.8a and 4.8b, we present estimates of the K-function and the pair correlation function under two modeling assumptions: a stationary homogeneous point process (blue line) and an inhomogeneous point process (red line). These estimates are compared against the theoretical expectations for a Poisson point process, namely $K(r) = \pi r^2$ and $g(r) = 1$. The plots indicate a clustering tendency, which, under the inhomogeneous scenario, persists up to approximately $r = 1250$ m according to $K(r)$ and up to about $r = 500$ m according to $g(r)$.

Figure 4.9 shows the spatio-temporal K-function and pair correlation function estimates, in the scale of days. The K-function increases with both time and space, suggesting a complex interaction between these dimensions. In contrast, the pair correlation function maintains a relatively constant shape over time, while decreasing with spatial distance, as already observed in the purely spatial non-parametric estimation.

Lastly, we present the results of the CSR tests conducted. The quadrat count two-sided hypothesis test using $6x6$ tiles in a square fitted to the study window resulted practice in 26 quadrats. The p-value of 2.2×10^{-16}, leading us to reject the CSR hypothesis.

For the rank envelop test we specified the alternative hypothesis as "greater", thus testing for clustering (i.e., more aggregation than expected). In Figure 4.10, the black solid line represents the observed $L_{\text{inhom}}(r) - r$, while the dashed black line shows the central function from the first set of simulations under the null. The grey shaded band is the 95% global rank envelope, indicating the expected variation under the null model, and

red dots mark distances where the observed curve lies outside this envelope, signifying statistically significant departures.



(a) Estimated spatio-temporal K-function.

(b) Estimated spatio-temporal pair-correlation function.

Figure 4.9: Absolute and relative frequency of crimes in 2022-2023: (a) top 10 from all the registered crime subcategories and (b) considering only street crime subcategories.

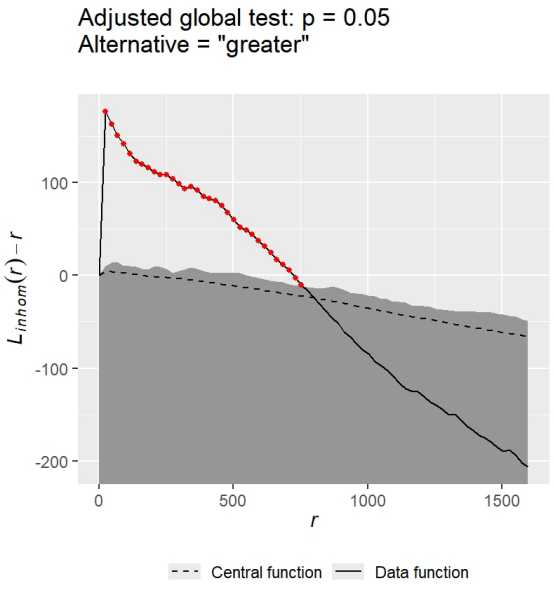


Figure 4.10: Inhomogeneous L-function rank envelope test.

The results show that up to about 750 m, the observed curve lies above the envelope, indicating significant clustering at short distances if we reject the null hypothesis at the 5% significance level. This suggests cases are more tightly clustered than expected under a Poisson process, and the effect is unlikely to be due to chance. Beyond 750 m, however,

the observed curve falls within the envelope, meaning no significant clustering is detected at larger distances.

Chapter 5

Modelling stage

The modelling stage followed three main steps: covariate identification, covariate selection, and spatial Log-Gaussian Cox Process (LGCP) modelling. Covariate identification is carried out through a triangulation of the options explored in the literature review and the publicly available databases in Portugal, to determine factors that may influence the spatial distribution of crime. Once potential covariates are identified, a selection process is undertaken using both manual, theory-driven choices and automatic, data-driven procedures to refine the set of variables. With the covariates defined, the modelling is conducted within the framework of a spatial LGCP, where the Latent Gaussian Model (LGM) is specified and fitted using the Integrated Nested Laplace Approximation (INLA) in conjunction with the Stochastic Partial Differential Equation (SPDE) approach. This allows for flexible spatial modelling while accounting for spatial dependencies in the data. The final stage involves evaluating the model outputs, including fixed and random effects, as well as the spatial distribution of predicted crime intensities.

The chapter is further divided into methods, implementation details tied to our specific case-study and results.

5.1 Methods

In this section, we present the methods applied. We begin with the automatic covariate selection procedure, which is grounded in classical statistical techniques. We then shift to the Bayesian framework, introducing the formulation of Latent Gaussian Models (LGMs) and the use of INLA combined with the SPDE approach. Finally, we discuss the criteria employed for model selection.

5.1.1 Automatic covariate selection procedure

When dealing with a large amount of candidate covariates, it is essential to perform a pre-filtering step before introducing them in the spatial LGCP model. Including such a large number of predictors is not only computationally inefficient but also statistically

problematic. Many covariates are likely to be strongly correlated with each other, which can lead to multicollinearity, unstable parameter estimates, and difficulties in interpreting the effects of individual variables. Moreover, overfitting becomes a major concern: with too many predictors relative to the amount of information in the data, the model risks capturing noise rather than true signal, reducing its generalisability and predictive performance.

The adopted procedure consisted of five steps: (1) normalising all covariates; (2) eliminating near-zero variance covariates; (3) applying stepwise Variance Inflation Factor (VIF); (4) fitting a simple generalized linear model with the remaining variables, and (5) applying stepwise model selection based on the the Akaike Information Criterion (AIC) with bootstrapping.

5.1.1.1 Step 1: Normalise the covariates

We standardised each quantitative variable X_j by subtracting the mean, $\bar{X}_j$, and dividing by the sample standard deviation, s_j, so that they are on comparable scales. This also facilitates interpretation of coefficients and improves numerical stability in the LGCP:

$$X_j^{\text{scaled}} = \frac{X_j - \bar{X}_j}{s_j}. \tag{5.1}$$

5.1.1.2 Step 2: Eliminate near-zero variance covariates

Standardised variables with near-zero variance were eliminated by function `nearZeroVar()`. This function helps to exclude variables with near-zero variance, i.e., variables that have almost the same value across all observations and thus are unlikely to be useful in modelling. As specified in the documentation, the function flags predictors when both of the following criteria are met:

- Frequency ratio (`freqCut`, default 95/5): The ratio of the frequency of the most common value to the second most common value. If this ratio is very large, the variable is flagged.
- Percent of unique values (`uniqueCut`, default 10%): The percentage of unique values relative to the total number of samples. If this percentage is very low, the variable is flagged.

5.1.1.3 Step 3: Apply Stepwise VIF (Variance Inflation Factor)

The correlation matrix is useful to assess collinearity between pairs of predictors. However, multicollinearity can exist among three or more variables even if no pairwise correlation is high. The *Variance Inflation Factor (VIF)* captures such situations [70].

The VIF for coefficient $\hat{\beta}_j$ is defined as the ratio between: the variance of $\hat{\beta}_j$ when estimated in the full linear regression model (including all predictors) and the variance of $\hat{\beta}_j$ when estimated in a model where X_j is the only predictor. Formally,

$$\text{VIF}(\hat{\beta}_j) = \frac{1}{1 - R^2_{X_j|X_{-j}}} \tag{5.2}$$

where $R^2_{X_j|X_{-j}}$ is the coefficient of determination from linear regressing X_j onto all other predictors. Let x_{ij} be the i-th observed value of predictor X_j, then

$$R^2_{X_j|X_{-j}} = \frac{\text{SS}_{\text{tot},j}}{\text{SS}_{\text{res},j}} = \frac{\sum_{i=1}^{n}(x_{ji} - \bar{x}_j)^2}{\sum_{i=1}^{n}(x_{ji} - \hat{x}_{ji})^2} \tag{5.3}$$

where $\text{SS}_{\text{tot},j}$ is the total sum of squares and $\text{SS}_{\text{res},j}$ is the residual sum of squares after regressing X_j on X_{-j}, and

$$\bar{x}_j = \frac{1}{n}\sum_{i=1}^{n} x_{ji}. \tag{5.4}$$

The closer $R^2_{X_j|X_{-j}}$ is to 1, the larger the VIF, approaching $+\infty$. VIF values range from 1 (no collinearity) to $+\infty$. A common rule of thumb is that a VIF exceeding 5 or 10 indicates problematic collinearity [70].

We utilised the `R` function `vifstep()`, which removes multicollinear variables based on a stepwise procedure:

1. Compute VIFs for all variables.
2. Identify the variable with the highest VIF.
3. Check if that VIF exceeds a specified threshold.
4. Remove the variable with the highest VIF (one at a time).
5. Recalculate VIFs on the remaining variables.
6. Repeat until all remaining variables have VIF $\leq$ threshold.

5.1.1.4 Step 4: Generalized Linear Model with the filtered variables

Considering the response variable Y represents counts of crime events per Statistical Section, we used a Generalised Linear Model (GLM) to model it. The Poisson GLM with a log link is a natural choice in this scenario. However, a quick inspection of the estimated mean and variance of y revealed over-dispersion, contrary to the Poisson assumption where $\text{Var}(Y) = \mathbb{E}[Y]$.

To account for over-dispersion, we instead used a negative binomial model. The negative binomial distribution supports positive integers and allows for variance larger than the mean. It can be parametrized by the mean μ and over-dispersion parameter θ:

$$\mathbb{E}[Y] = \frac{\theta p}{1-p} = \mu, \qquad \text{Var}[Y] = \frac{\theta p}{(1-p)^2} = \mu + \frac{\mu^2}{\theta}, \quad \text{with } p = \frac{\theta}{\theta + \mu}. \tag{5.5}$$

The `glm.nb()` function from the `MASS` library allows us to quickly fit a classical statistics formulation of negative binomial GLMs:

$$y_i \sim \text{NegBin}(\mu_i, \theta), \qquad \log(\mu_i) = x_i^T \beta = \beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip}, \tag{5.6}$$

where y_i is the crime count for statistical section i and x_{ij} is the value of covariate j for section i.

5.1.1.5 Stepwise Model Selection with AIC and Bootstrap

To ensure robust covariate selection, we combined stepwise selection based on the Akaike Information Criterion (AIC) [71] with a bootstrap resampling procedure [70].

The AIC, a model selection criterion based on maximum-likelihood estimation, is defined as

$$\text{AIC} = -2\log(\mathcal{L}_{\max}) + 2d,$$

where d is the number of estimated parameters (including the intercept) and $\mathcal{L}_{\max}$ is the maximum likelihood of the model evaluated at the parameter estimates. Lower AIC indicates a better balance between goodness-of-fit and model complexity.

For the stepwise procedure, we defined:

- Full model: includes all the variables retained from previous steps.
- Null model: includes only the intercept.

We used the R function `stepAIC()` to perform bidirectional stepwise selection:

1. Evaluate the current model's AIC.
2. Try removing each predictor one at a time and check AIC.
3. Try adding predictors and check AIC.
4. Choose the step (add or drop) that reduces AIC the most.
5. Repeat until no improvement is possible.

Because variable selection can be sensitive to the sample, we applied the selection process on multiple bootstrap samples. In each iteration, we resampled the dataset with replacement to create a new dataset of the same size. Some observations may appear multiple times, others not at all, simulating sampling variability.

This approach allows assessment of the stability and consistency of covariate inclusion across resamples, reducing overfitting risk and improving confidence in selected variables. Covariates selected in at least 50% of bootstrap samples were considered relevant.

5.1.2 Bayesian paradigm

Classical (frequentist) statistics treats parameters in a model as fixed and unknown quantities and relies on point estimates and confidence intervals to infer about their value. In contrast, Bayesian statistics interprets parameters as random variables, characterized by probability distributions that incorporate both prior knowledge and uncertainty. This distinction allows for a fully probabilistic interpretation of inference results and is within this paradigm that we developed our model.

5.1.2.1 General Bayesian model

Let $\boldsymbol{\theta}$ represent a vector of parameters of interest, and let $\boldsymbol{y}$ denote the observed data. A Bayesian model consists of two key components [50]:

1. The *likelihood*, $\mathrm{p}(\boldsymbol{y}|\boldsymbol{\theta})$, which represents the distribution of the observed data given the parameters. While $\mathrm{p}(\boldsymbol{y}|\boldsymbol{\theta})$ is derived from the data distribution, in the context of inference, it is treated as a function of the parameters $\boldsymbol{\theta}$, with the data $\boldsymbol{y}$ held fixed. It can also be expressed as $\mathrm{L}(\boldsymbol{\theta}|\boldsymbol{y})$ to emphasize this dependence on the parameters.

2. The *prior distribution*, $\mathrm{p}(\boldsymbol{\theta})$, which reflects prior knowledge or assumptions about the parameters before observing the data. It is a probability distribution chosen *a priori*.

Mathematically, a Bayesian model is expressed as:

$$\begin{aligned} \boldsymbol{y}|\boldsymbol{\theta} &\sim \mathrm{p}(\boldsymbol{y}|\boldsymbol{\theta}) \\ \boldsymbol{\theta} &\sim \mathrm{p}(\boldsymbol{\theta}) \end{aligned} \tag{5.7}$$

and at the heart of Bayesian inference lies the Bayes' theorem, which updates prior beliefs about a parameter in light of new data:

$$\mathrm{p}(\boldsymbol{\theta}|\boldsymbol{y}) = \frac{\mathrm{p}(\boldsymbol{y}|\boldsymbol{\theta})\mathrm{p}(\boldsymbol{\theta})}{\mathrm{p}(\boldsymbol{y})} \propto \mathrm{p}(\boldsymbol{y}|\boldsymbol{\theta})\mathrm{p}(\boldsymbol{\theta}). \tag{5.8}$$

Here, $\mathrm{p}(\boldsymbol{\theta}|\boldsymbol{y})$ is the posterior distribution, that quantifies updated belief about $\boldsymbol{\theta}$. The denominator, $\mathrm{p}(\boldsymbol{y})$, is the marginal likelihood and serves as a normalizing constant which ensures the posterior distribution integrates to one over the whole parameter space, Θ:

$$\mathrm{p}(\boldsymbol{y}) = \int_{\Theta} \mathrm{p}(\boldsymbol{y}|\boldsymbol{\theta})\mathrm{p}(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta}. \tag{5.9}$$

5.1.2.2 Choosing a prior distribution

The choice of prior distribution, $\mathrm{p}(\boldsymbol{\theta})$, influences the posterior distribution, particularly when data is sparse. Priors can reflect subjective beliefs, expert knowledge, or general assumptions about parameter behavior [50]. Selecting an appropriate prior is a critical step, as it directly affects the final inference.

Consider a simple example of a univariate linear regression Bayesian model:

$$y_i = \beta x_i + \varepsilon, \quad \varepsilon \sim \mathcal{N}(0, \sigma_0^2)$$

with parameters β and known σ_0^2. Suppose we want to chose a prior for the coefficient β. Three possible choices are provided in Table 5.1.

Table 5.1: Different types of prior distributions and an example of prior choice for the covariate coefficient of a univariate linear regression Bayesian model.

Prior type	**Description**	**Example**
Non-informative	Designed to have minimal influence on the posterior, e.g., uniform or Jeffrey's priors.	Improper prior, we let the data speak entirely: $\beta \sim \mathcal{U}(-\infty, +\infty)$
Weakly-informative	Balance between non-informativeness and guarding against unrealistic estimates.	We believe β should be somewhere in the range $[-20, 20]$: $\beta \sim \mathcal{N}(0, 10^2)$
Informative	Encode specific prior knowledge, such as tight Gaussian priors for parameters expected to be near certain values.	There is strong belief, despite the data, that β is very close to 3: $\beta \sim \mathcal{N}(3, 0.1^2)$

Since the univariate linear regression is such a simple case, we can construct a conjugate prior, which is a prior distribution that results in a posterior from the same family, thereby simplifying analytical calculations. In this case, a normal likelihood with known variance and a normal prior on the regression coefficient yields a normal posterior. However, conjugate priors are only available for a limited set of likelihood-prior combinations. In more complex models where conjugacy is not available or tractable, one typically resorts to non-informative or weakly informative priors, chosen independently of conjugacy considerations.

Another increasingly popular form of weakly-informative priors is the class of *Penalised Complexity (PC) priors*, which offers a principled way to construct priors that penalise model complexity by assigning higher prior mass to a simpler (base) model and decreasing mass exponentially as complexity increases, based on prior beliefs [72].

When using PC priors we indirectly specify a prior for a flexibility parameter ξ that controls model complexity. Typically $\xi = 0$ for the base model and increases in value with model complexity, reflecting the principle of Occam's razor. The parameter ξ is then associated to some actual model parameters we are seeking a prior for. Simpson et al. [72] explain in detail how we can define a prior for ξ, $\mathrm{p}(\xi)$, from first principles. We give here a general idea.

To measure model complexity, we use the *Kullback-Leibler Divergence (KLD)*, which is a measure of the difference between two probability distribution functions defined over the same variables. This concept was introduced from information and probability theory and when applied to PC priors is stated as follows:

$$\mathrm{KLD}\big(\mathrm{p}(x|\xi)||\mathrm{p}(x|\xi=0)\big) = \int_{x\in\mathcal{X}} \mathrm{p}(x|\xi)\log\left(\frac{\mathrm{p}(x|\xi)}{\mathrm{p}(x|\xi=0)}\right)\mathrm{d}x \tag{5.10}$$

Distribution $\mathrm{p}(x|\xi)$ represents the more complex model, while $\mathrm{p}(x|\xi=0)$ corresponds to a simpler base model. The symbol $||$ is standard notation for divergence between two distributions in information theory. The KLD is non-negative and quantifies the average

information loss incurred when approximating $\mathrm{p}(x|\xi)$ by $\mathrm{p}(x|\xi = 0)$. The result is expressed in nats (or bits, if a base-2 logarithm is used).

In order to make the KLD interpretable, i.e., behaving locally as a metric and with distance units, we utilise as a measure of model complexity of the form:

$$d(\mathrm{p}(x|\xi)||\mathrm{p}(x|\xi = 0)) = \sqrt{2\,\mathrm{KLD}(\mathrm{p}(x|\xi)||\mathrm{p}(x|\xi = 0))} \tag{5.11}$$

We then construct a distribution for d, $\mathrm{p}(d)$ that (1) penalises deviations from the base model parametrised with distanced d and (2) retains the *memoryless property*:

$$\frac{\mathrm{p}(d+\xi)}{\mathrm{p}(d)} = r^{\sigma}, \;\; d, \sigma \geq 0 \tag{5.12}$$

The intuition is that the belief in a model decreases exponentially, at a constant rate, for each additional unit of complexity accepted. The distribution that allows us to retains both conditions, ensuring as well the mode is at $d = 0$, is the exponential decay prior:

$$\mathrm{p}(d) = \lambda e^{-\lambda d}, \;\; r = e^{-\lambda} \tag{5.13}$$

With a change-of-variables, one obtains the prior distribution of ξ:

$$\mathrm{p}(\xi) = \lambda e^{-\lambda d(\xi)} \left| \frac{\partial d(\xi)}{\partial \xi} \right| \tag{5.14}$$

The user of a PC-prior selects λ by controlling the prior mass in the tail of the distribution of ξ. This is done indirectly since ξ is expressed as a function of a more meaningful quantity, $Q(\xi)$, such as a standard deviation or a correlation parameter, for instance. Concretely, one controls probabilities $\mathrm{P}(Q(d) > U) = \alpha$ or $\mathrm{P}(Q(d) < L) = \alpha$, where U and L are upper and lower limits, respectively, and α is an upper or lower tail probability of the prior distribution.

In this work, PC-priors are employed for the spatial random effect, following a construction already established in the literature, and non-informative priors are defined for covariate coefficients. The details will be clarified later in the document.

5.1.2.3 Inference and prediction

Bayesian inference is conducted using the posterior distribution, $\mathrm{p}(\boldsymbol{\theta}|\boldsymbol{y})$. Summary statistics such as the mean, median, or credible intervals can be derived for individual parameters θ_i, by marginalizing the joint posterior distribution over the vector of the other parameters, $\boldsymbol{\theta}_{-i}$ [50]:

$$\mathrm{p}(\theta_i|\boldsymbol{y}) = \int_{\Theta} \mathrm{p}(\boldsymbol{\theta}|\boldsymbol{y}) \mathrm{d}\boldsymbol{\theta}_{-i}, \; i = 1, \ldots, \dim(\boldsymbol{\theta}). \tag{5.15}$$

Bayesian analysis also facilitates prediction, which is possible by obtaining the predictive distribution for new observations. For instance, consider a new observation y^*. The

predictive distribution for new data is obtained integrating over the uncertainty in the posterior distribution of $\boldsymbol{\theta}$:

$$\mathrm{p}(y^*|\boldsymbol{y}) = \int_{\Theta} \mathrm{p}(y^*|\boldsymbol{\theta})\mathrm{p}(\boldsymbol{\theta}|\boldsymbol{y})\,\mathrm{d}\boldsymbol{\theta}. \tag{5.16}$$

This integration is performed over the whole parameter space $\boldsymbol{\Theta}$ and weighted according to the posterior distribution for $\boldsymbol{\theta}$, therefore it accounts for both the uncertainty in the model parameters and the inherent variability in the data, providing a probabilistic prediction that fully reflects uncertainty.

Bayesian statistics has gained increasing popularity as computational methods have advanced. Computing the posterior and marginal posterior distributions often poses significant challenges, since closed-form solutions are typically available only for a limited class of models with conjugate priors. Consequently, numerical and approximate techniques are frequently employed. Among these, INLA will be used in this work.

5.1.3 Integrated Nested Laplace Approximation (INLA)

As mentioned in the Chapter 2, the *Integrated Nested Laplace Approximation (INLA)* was developed by Rue et al. [41] as a faster computational alternative to Markov chain Monte Carlo (MCMC) methods for Bayesian inference. Its primary goal is to estimate the posterior marginal distributions of model parameters, from Equation 5.15, efficiently.

The three building blocks of INLA are Latent Gaussian Models, Gaussian Markov Random Fields, and the Laplace Approximation. INLA is designed for models that can be expressed as Latent Gaussian Models. What makes INLA particularly fast is the use of Gaussian Markov Random Fields to approximate the Latent Gaussian Field and the application of the Laplace approximation in the integration of the marginal distributions [41, 49]. We now present the Latent Gaussian Model structure and briefly go through the mathematical basis of INLA.

5.1.3.1 Latent Gaussian Model (LGM) formulation

Latent Gaussian Models (LGMs) are part of a broader class of models: the *Structured Additive Regression Models*. In this family of models, the observed response variables, y_i, follows a distribution from the exponential family, where the mean, μ_i, is linked to a structured additive predictor η_i via a link function g, i.e., $\eta_i = g(\mu_i)$ [24, 73] and

$$\eta_i = \beta_0 + \sum_{j=1}^{n_f} f^{(j)}(u_{ji}) + \sum_{k=1}^{n_\beta} \beta_k z_{ki} + \varepsilon_i, \tag{5.17}$$

where β_0 is the intercept representing the overall mean of the response, $\{f^{(j)(u_{ij})}\}$ are unknown functions of covariates u_j, the set $\{\beta_k\}$ represents linear effects on covariates $\mathbf{z}$ and ε_i is an additional unstructured random effect term. The class of structured additive regression models is incredible flexible since the unknown functions can take many forms.

A *Latent Gaussian Model* has a predictor of the form shown in Equation 5.17, plus one more requirement: Gaussian priors must be assigned to each of the unknown parameters in the predictor, β_0, $\{\beta_k\}$ and $\{f^{(j)}\}$. These entities can be grouped in a vector $\boldsymbol{x} = [\beta_0, \{\beta_k\}, \{f^{(j)}\}]$ designated the *vector of latent Gaussian variables* which form a Latent Gaussian Field in continuous space. Note that here $\{f^{(j)}\}$ represents the set of unknown parameters of functions of the covariates, not the functions themselves. Using the general Bayesian framework of Equation 5.7, the LGM can be presented as a hierarchical model with three layers [46]:

$$\begin{aligned} \boldsymbol{\theta} &\sim \mathrm{p}(\boldsymbol{\theta}) && \text{(hyperparameters)} \\ \mathbf{x} \mid \boldsymbol{\theta} &\sim \mathcal{N}(0, Q(\boldsymbol{\theta})^{-1}) && \text{(Latent Gaussian Field)} \\ \mathbf{y} \mid \mathbf{x}, \boldsymbol{\theta} &\sim \prod_i^N \mathrm{p}(y_i \mid \eta_i(\mathbf{x}), \boldsymbol{\theta}) && \text{(observations)} \end{aligned} \tag{5.18}$$

Starting from the bottom, the last layer represents the distribution of the data, consisting of N observations. Note this is also where the predictor enters. Meanwhile, the first two layers describe the model parameters for which prior distributions are assigned. Here, $\boldsymbol{\theta}$ is the vector of hyperparameters, shared across both the latent Gaussian variables and the observations. As noted by Opitz [46], these hyperparameters can be, for instance, dispersion parameters that appear in the likelihood function (e.g., the variance of the Gaussian distribution, an overdispersion parameter of a negative binomial, or the shape parameter of a gamma distribution); but they can also be parameters that characterize the structure of the Latent Gaussian Model (e.g., variances, spatial correlation parameters, or autoregression coefficients). Indeed, the observation distribution depends on the first kind of hyperparameters, θ_1, while the Latent Gaussian Field depends on the second kind, θ_2. However, to simplify notation, it is common to say all terms depend on the full hyperparameter vector, $\boldsymbol{\theta}$, which is in fact $\boldsymbol{\theta} = [\theta_1, \theta_2]^T$.

The hyperparameters can have a non-Gaussian prior, $\mathrm{p}(\boldsymbol{\theta})$. However, the vector of latent Gaussian variables, $\boldsymbol{x}$, has a Gaussian prior, as required by the LGM structure, with mean 0 and a covariance matrix $Q(\boldsymbol{\theta})^{-1}$, written in terms of the precision matrix $Q(\boldsymbol{\theta})$:

$$\mathrm{p}(\boldsymbol{x}|\boldsymbol{\theta}) = |Q(\boldsymbol{\theta})|^{1/2} \exp\left\{\frac{1}{2}\boldsymbol{x}^T Q(\boldsymbol{\theta})\boldsymbol{x}\right\}. \tag{5.19}$$

The precision matrix, Q, is of greater interest than the covariance, Σ, in INLA, as shall become clear later. Following Bayes' theorem (Equation 5.8), the joint posterior distribution of $\boldsymbol{\theta}$ for the LGM can be derived:

$$\begin{aligned} \mathrm{p}(\boldsymbol{x}, \boldsymbol{\theta}|\boldsymbol{y}) &\propto \mathrm{p}(\boldsymbol{x}, \boldsymbol{\theta})\mathrm{p}(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{\theta}) \\ &= \mathrm{p}(\boldsymbol{\theta})\mathrm{p}(\boldsymbol{x}|\boldsymbol{\theta}) \prod_{i=1}^N \mathrm{p}(y_i|x_i, \boldsymbol{\theta}) \end{aligned}$$

$$= \mathrm{p}(\boldsymbol{\theta})|Q(\boldsymbol{\theta})|^{1/2}\exp\left\{\frac{1}{2}\boldsymbol{x}^T Q(\boldsymbol{\theta})\boldsymbol{x}\right\}\prod_{i=1}^{N}\mathrm{p}(y_i|x_i,\boldsymbol{\theta})$$

$$= \mathrm{p}(\boldsymbol{\theta})|Q(\boldsymbol{\theta})|^{1/2}\exp\left\{-\frac{1}{2}\boldsymbol{x}^T Q(\boldsymbol{\theta})\boldsymbol{x} \; + \; \sum_{i=1}^{N}\log\left\{\mathrm{p}(y_i|\eta_i,\boldsymbol{\theta})\right\}\right\}. \tag{5.20}$$

This multivariate posterior distribution is not typically from a known family of distributions. Besides, it is usually more relevant to evaluate the posterior distribution of each parameter. INLA approximates the (univariate) marginal posterior distributions for each latent Gaussian variable, x_i, including the resulting predictor values η_i, and each hyperparameter, θ_j:

$$\mathrm{p}(\theta_j|\boldsymbol{y}) = \int\int \mathrm{p}(\boldsymbol{x},\boldsymbol{\theta}|\boldsymbol{y})d\boldsymbol{x}d\boldsymbol{\theta}_{-j} = \int \mathrm{p}(\boldsymbol{\theta}|\boldsymbol{y})d\boldsymbol{\theta}_{-j} \tag{5.21}$$

$$\mathrm{p}(x_i|\boldsymbol{y}) = \int\int \mathrm{p}(\boldsymbol{x},\boldsymbol{\theta}|\boldsymbol{y})d\boldsymbol{x}_{-i}d\boldsymbol{\theta} = \int \mathrm{p}(x_i|\boldsymbol{\theta},\boldsymbol{y})\mathrm{p}(\boldsymbol{\theta},\boldsymbol{y})d\boldsymbol{\theta}) \tag{5.22}$$

The distributions $\mathrm{p}(\boldsymbol{\theta}|\boldsymbol{y})$, $\mathrm{p}(x_i|\boldsymbol{\theta},\boldsymbol{y})$ and $\mathrm{p}(\boldsymbol{\theta},\boldsymbol{y})$ are approximated in INLA and strategies to explore the hyperparameter space combined with numerical integration methods are employed. For an efficient application of INLA, the number of hyperparameters must be small, with Rue et al. [41] suggesting $\dim(\boldsymbol{\theta}) \leq 6$. On the other hand, the Latent Gaussian Field $\boldsymbol{x}$ can be high-dimensional, with efficiency guaranteed as long as it approximates to a Gaussian Markov Random Field.

5.1.3.2 Gaussian Markov Random Fields (GMRFs)

Gaussian Markov Random Fields (GMRFs) can be thought of as a Gaussian Random Field with conditional independence properties, hence the term "Markov". Their utility in INLA arises from the sparsity of their precision matrices, $Q(\boldsymbol{\theta})$. Sparse matrices enable significantly faster matrix operations compared to dense matrices, making GMRFs computationally efficient for Bayesian inference [41]. Moreover, in a spatial model, they capture inter-point dependence.

The key concept in understanding GMRFs is conditional independence [74]. Let us first consider the more familiar concept of variable independence. Two random variables x and y are said to be independent if and only if their joint distribution can be expressed as a product of their individual distributions, that is, $\mathrm{p}(x,y) = \mathrm{p}(x)\mathrm{p}(y)$. Building on this, conditional independence describes a scenario where x and y are independent only when conditioned on a third variable z. Formally stating:

Definition 5.1. Two random variables, x and x are said to be *conditionally independent* given a third variable z if and only if

$$\mathrm{p}(x,y|z) = \mathrm{p}(x|z)\mathrm{p}(y|z)$$

and we write $x \perp y|z$.

The same way variable independence easily extends to the multivariate case, so does conditional independence. To associate this concept to GMRFs, it is necessary to understand the meaning of conditional independence in the context of an undirected graph:

Definition 5.2. An undirected graph is a tuple $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of nodes in graph $\mathcal{G}$ and $\mathcal{E}$ is the set of edges $\{i, j\}$, where $i, j \in \mathcal{V}$ and $i \neq j$. If $\{i, j\} \in \mathcal{E}$ there is an undirected edge from node i to node j, otherwise, there is no edge between nodes i and j.

In an undirected graph, nodes are connected by edges that do not have a specified direction. When all nodes are connected either directly or indirectly, such that no node remains isolated, the graph is referred to as connected. If all nodes are connected to each other directly, the graph is fully connected or complete. Additionally, a graph where each node has a distinct identifier is known as a labelled graph.

Another relevant concept is that of neighbourhood. The neighbours of a given node in a graph refer to the set of all nodes that are directly connected to that node by edges.

Definition 5.3. The *neighbours* of node i are all nodes j in $\mathcal{G}$ that have an edge to node i:

$$\text{ne}(i) = \{j \in \mathcal{V} : \{i, j\} \in \mathcal{E}\}.$$

Let us consider now the vector of normally distributed variables, $\boldsymbol{x} = (x_1, \ldots, x_n)^T$, with mean $\boldsymbol{\mu}$ and covariance Σ. It is said that $\boldsymbol{x}$ is a GMRF with respect to the labelled graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with $\mathcal{V} = 1, \ldots, n$ and $\mathcal{E}$ such that there is no edge between nodes i and j in $\mathcal{G}$ if and only if $x_i \perp x_j \mid \boldsymbol{x}_{-\{i,j\}}$[1].

From this definition provided by Rue and Held [74], it follows that two elements of the GMRF are neighbours if and only if they are not conditionally independent, given all other variables. Therefore, the neighbourhood of a node consists of all other nodes that are dependent on it, meaning there is a direct stochastic dependence between the corresponding variables. In this way, the GMRF captures inter-point dependence through its structure.

Finally, it is possible to establish a connection between the graph $\mathcal{G}$ and the parameters that define the normal distribution of the GMRFs. As Rue and Held [74] explain, this is done through the precision matrix, $Q = \Sigma^{-1}$.

Theorem 5.1. Let $\boldsymbol{x} = (x_1, \ldots, x_n)^T$ be normally distributed with mean μ and precision matrix $Q > 0$. Then, for $i \neq j$:

$$x_i \perp x_j \mid \boldsymbol{x}_{-\{i,j\}} \iff Q_{ij} = 0.$$

From this result, one concludes the non-zero pattern in Q determines the structure of $\mathcal{G}$ (which nodes are connected) and vice-versa. By combining the theorem with the previous definition it follows that the neighbours of an element $\boldsymbol{x}_i$ of the random field are

[1] $\boldsymbol{x}_{-\{i,j\}}$ denotes the set of all variables excluding x_i and x_j

precisely the elements $\boldsymbol{x}_j$ for which $Q_{ij} \neq 0$. In practice, conditional dependence is not expected among all elements since not all points will be connected, resulting in many zero entries in Q and making it sparse.

Additionally to this fundamental conclusion, as the authors [74] highlight, the elements of Q have useful interpretations, aside from the non-zero pattern. The diagonal elements of Q, Q_{ii}, are the conditional precisions of x_i given $\mathbf{x}_{-i}$. Meanwhile, the off-diagonal elements Q_{ij}, with $i \neq j$, scaled by the factor $-1/\sqrt{Q_{ii}Q_{jj}}$, give the correlation between x_i and x_j, given $x_{-\{i,j\}}$.

The more formal and complete definition for a GMRF is thus:

Definition 5.4. A random vector $\mathbf{x} = (x_1, \ldots, x_n)^\top \in \mathbb{R}^n$ is called a *Gaussian Markov Random Field (GMRF)* with respect to a labelled graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, with mean $\boldsymbol{\mu}$ and precision matrix $Q > 0$, if and only if its density has the form

$$\pi(\mathbf{x}) = (2\pi)^{-n/2}|Q|^{1/2} \exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^\top Q(\mathbf{x}-\boldsymbol{\mu})\right)$$

and

$$Q_{ij} = 0 \iff \{i,j\} \notin \mathcal{E}, \quad \text{for all } i \neq j.$$

5.1.3.3 Laplace Approximation

The final building block of INLA is now presented: the Laplace approximation. This is the method employed to approximate the value of high-dimensional integrals that arise in Bayesian inference, namely those performed over the parameter space.

Suppose we wish to approximate the multi-dimensional integral [50]

$$\int_{\mathbb{R}^n} f(\boldsymbol{x})\mathrm{d}\boldsymbol{x}. \tag{5.23}$$

For the Laplace approximation to be effective, the integrand f must be "well-behaved". Specifically, f should be unimodal and close to a Gaussian distribution. A useful property in this context is strict log-concavity [46].

We begin by expressing $f(\boldsymbol{x})$ as an exponential function, where $g(\boldsymbol{x}) = \log(f(\boldsymbol{x}))$,

$$\int_{\mathbb{R}^n} f(\boldsymbol{x})d\boldsymbol{x} = \int_{\mathbb{R}^n} e^{g(\boldsymbol{x})}\mathrm{d}\boldsymbol{x}. \tag{5.24}$$

Our reasoning will be that the integral's value is largely determined by the behaviour of g around the mode, $\boldsymbol{x}^*$, where $g(\boldsymbol{x}^*)$ is the global maximum. To approximate g around the mode, a second-order Taylor expansion is used,

$$g(\boldsymbol{x}) \approx g(\boldsymbol{x}^*) + \nabla g(\boldsymbol{x}^*)(\boldsymbol{x}-\boldsymbol{x}^*) + \frac{1}{2}(\boldsymbol{x}-\boldsymbol{x}^*)^T H_{\mathrm{G}}(\boldsymbol{x}^*)(\boldsymbol{x}-\boldsymbol{x}^*). \tag{5.25}$$

Since $\boldsymbol{x}^*$ is a global maximum, it follows that $\nabla g(\boldsymbol{x}^*) = 0$ and the Hessian of g evaluated at the mode, $H_{\mathrm{G}}(\boldsymbol{x}^*)$, is negative definite, simplifying the expansion to

$$g(\boldsymbol{x}) \approx g(\boldsymbol{x}^*) + \frac{1}{2}(\boldsymbol{x}-\boldsymbol{x}^*)^T H_{\mathrm{G}}(\boldsymbol{x}^*)(\boldsymbol{x}-\boldsymbol{x}^*). \tag{5.26}$$

Substituting this approximation into the integral, the following approximation to its real value is obtained:

$$\begin{aligned}
\int_{\mathbb{R}^n} f(\boldsymbol{x})d\boldsymbol{x} &\approx \int_{\mathbb{R}^n} \exp\left\{ g(\boldsymbol{x}^*) + \frac{1}{2}(\boldsymbol{x}-\boldsymbol{x}^*)^T H_{\mathrm{G}}(\boldsymbol{x}^*)(\boldsymbol{x}-\boldsymbol{x}^*) \right\} d\boldsymbol{x} \\
&= e^{g(\boldsymbol{x}^*)} \int_{\mathbb{R}^n} \exp\left\{ \frac{1}{2}(\boldsymbol{x}-\boldsymbol{x}^*)^T H_{\mathrm{G}}(\boldsymbol{x}^*)(\boldsymbol{x}-\boldsymbol{x}^*) \right\} \mathrm{d}\boldsymbol{x} \\
&= e^{g(\boldsymbol{x}^*)} \sqrt{\frac{(2\pi)^n}{|H_{\mathrm{G}}(\boldsymbol{x}^*)|}} \underbrace{\int_{\mathbb{R}^n} \sqrt{\frac{|H_{\mathrm{G}}(\boldsymbol{x}^*)|}{(2\pi)^n}} \exp\left\{ \frac{1}{2}(\boldsymbol{x}-\boldsymbol{x}^*)^T H_{\mathrm{G}}(\boldsymbol{x}^*)(\boldsymbol{x}-\boldsymbol{x}^*) \right\} \mathrm{d}\boldsymbol{x}}_{=1 \text{ since the integrand is the pdf of } X \sim \mathcal{N}\left(\mu=\boldsymbol{x}^*,\ \Sigma=H_{\mathrm{G}}^{-1}(\boldsymbol{x}^*)\right)} \\
&= e^{g(\boldsymbol{x}^*)} \sqrt{\frac{(2\pi)^n}{|H_{\mathrm{G}}(\boldsymbol{x}^*)|}}.
\end{aligned} \tag{5.27}$$

If the integral is performed over a specific interval $[\alpha, \beta]$, the Laplace approximation becomes

$$\int_{\alpha}^{\beta} f(\boldsymbol{x})d\mathbf{x} \approx e^{g(\boldsymbol{x}^*)} \sqrt{\frac{(2\pi)^n}{|H_{\mathrm{G}}(\boldsymbol{x}^*)|}} (\Phi(\beta) - \Phi(\alpha)), \tag{5.28}$$

where Φ is the cumulative probability distribution of $X \sim \mathcal{N}\left(\mu = \boldsymbol{x}^*,\ \Sigma = H_{\mathrm{G}}^{-1}(\boldsymbol{x}^*)\right)$. Note the power of this method lies also in the fact that it approximates the integrand f to a Gaussian distribution whose parameters depend on the mode of g. In practice, the mode x^* can be found by solving the unconstrained optimization problem

$$\max_{\boldsymbol{x} \in \mathbb{R}^n} g(\boldsymbol{x}), \tag{5.29}$$

typically using iterative methods such as the Newton-Raphson algorithm. The approximation improves when the shape of f is very close to Gaussian and when the interval of integration is tight around the mode [50].

5.1.3.4 Fitting procedure

Recall the goal of INLA is to approximate the marginal posterior distributions of Equations 5.21 and 5.22. The inner integrals over the parameter space, $\boldsymbol{x}$ and $\boldsymbol{x}_{-j}$, are usually high-dimensional, reaching orders of magnitude of 10^2 to 10^8, whereas the outer integrals are performed over the hyperparameters space, typically of small dimension. Therefore, the strategy followed by INLA commences by approximating functions $\mathrm{p}(\boldsymbol{\theta}, \boldsymbol{y})$ and $\mathrm{p}(x_i, \boldsymbol{\theta}, \boldsymbol{y})$. The first function is approximated using the Laplace approximation. For the second one, however, INLA offers 3 options: (1) direct Gaussian approximation, (2)

Laplace approximation or a (3) simplified Laplace approximation (the default option), depending on the desired trade-off between accuracy and fast computation. Afterwards, numerical integration schemes can be applied to the integrals in the hyperparameter space using the already obtained approximated functions $\tilde{p}(\boldsymbol{\theta}, \boldsymbol{y})$ and $\tilde{p}(x_i, \boldsymbol{\theta}, \boldsymbol{y})$ [41, 46].

We begin by applying the Laplace approximation, described previously, to the integral

$$\int p(\boldsymbol{x}, \boldsymbol{\theta}|\boldsymbol{y}) d\boldsymbol{x} = p(\boldsymbol{\theta}|\boldsymbol{y}), \tag{5.30}$$

from which one obtains the approximation of the posterior distribution of the hyperparameters,

$$\tilde{p}(\boldsymbol{\theta}|\boldsymbol{y}) \propto \left. \frac{p(\boldsymbol{x}, \boldsymbol{\theta}, \boldsymbol{y})}{\tilde{p}_G(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})} \right|_{\boldsymbol{x}=\boldsymbol{x}^*(\boldsymbol{\theta})}, \tag{5.31}$$

where $\boldsymbol{x}^*(\boldsymbol{\theta})$ is the mode of $p(\boldsymbol{x}, \boldsymbol{\theta}, \boldsymbol{y})$ for a fixed vector $(\boldsymbol{\theta}, \boldsymbol{y})$ and $\tilde{p}_G(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})$ is the approximation of distribution $p(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})$ to a Gaussian distribution with precision matrix $Q(\boldsymbol{\theta})$ and mean $\boldsymbol{x}^*(\boldsymbol{\theta})$:

$$\tilde{p}_G(\mathbf{x}|\boldsymbol{\theta}, \mathbf{y}) = (2\pi)^{-n/2} |Q(\boldsymbol{\theta})|^{1/2} \exp\left(-\frac{1}{2} (\mathbf{x} - \mathbf{x}^*(\boldsymbol{\theta}))' Q(\boldsymbol{\theta}) (\mathbf{x} - \mathbf{x}^*(\boldsymbol{\theta})) \right) \tag{5.32}$$

where $n = \dim(\boldsymbol{x})$ is the number of latent Gaussian variables in the model.

Next, the focus is on approximating the integral

$$\int p(\boldsymbol{x}, \boldsymbol{\theta}|\boldsymbol{y}) d\boldsymbol{x}_{-i} = p(x_i|\boldsymbol{\theta}, \boldsymbol{y}). \tag{5.33}$$

Note that this integral differs from the one of Equation 5.30, since it is performed over a different parameter space: $\boldsymbol{x}_{-i}$. It needs to be recalculated for each index i, increasing computational costs.

A first approach consists of marginalizing over the Gaussian approximation of the multivariate distribution, $\tilde{p}_G(\boldsymbol{x}|\boldsymbol{\theta}, \boldsymbol{y})$, to obtain the univariate Gaussian approximation, $\tilde{p}_G(x_i|\boldsymbol{\theta}, \boldsymbol{y})$, for each parameter x_i:

$$\tilde{p}_G(x_i|\boldsymbol{\theta}, \mathbf{y}) \sim \mathcal{N}(\mu = x_i^*(\boldsymbol{\theta}), \Sigma = (Q_{ii}^{-1}(\boldsymbol{\theta}))). \tag{5.34}$$

Experience with INLA shows that this is usually not accurate enough, since it fails at accommodating any skewness of the true unknown posterior distribution and mislocates the mode. However, it is the fastest option and it should be used when the data likelihood is Gaussian, in which case $\tilde{p}_G(\mathbf{x}|\boldsymbol{\theta}, \mathbf{y})$ is an exact approximation of $p(\mathbf{x}|\boldsymbol{\theta}, \mathbf{y})$ [41, 46].

A second possible approach is to apply the Laplace approximation to integral 5.33, obtaining an approximation analogous to Equation 5.31:

$$\tilde{p}_{LA}(x_i|\boldsymbol{\theta}, \boldsymbol{y}) \propto \left. \frac{p(\boldsymbol{x}, \boldsymbol{\theta}, \boldsymbol{y})}{\tilde{p}_G(\boldsymbol{x}_{-i}|x_i, \boldsymbol{\theta}, \boldsymbol{y})} \right|_{\boldsymbol{x}_{-i}=\boldsymbol{x}_{-i}^*(\boldsymbol{\theta})} \tag{5.35}$$

This choice yields the best results, but is also very costly computationally. A Laplace approximation must be performed for each x_i as the mode depends on both x_i and $\boldsymbol{\theta}$. We note that Rue et al. [41] propose two modifications to the Laplace approximation that can reduce costs.

A third option, which lies midway between the two previous approaches, is presented by the same authors [41]: the simplified Laplace approximation. This approach is based on a third-order Taylor series expansion of $\tilde{\mathrm{p}}_{\mathrm{LA}}(x_i|\boldsymbol{\theta}, \boldsymbol{y})$ and it corrects the aforementioned issues of the Gaussian approximation $\tilde{\mathrm{p}}_{\mathrm{G}}(x_i|\boldsymbol{\theta}, \boldsymbol{y})$. This is the default option in `R-INLA`.

Now, all that is left is to approximate the marginal posterior distributions $\mathrm{p}(\boldsymbol{\theta}_j|\boldsymbol{y})$ and $\mathrm{p}(\boldsymbol{x}_i|\boldsymbol{y})$ in the following way [50]:

$$\tilde{\mathrm{p}}(x_i|\boldsymbol{y}) = \sum_k \tilde{\mathrm{p}}(x_i|\boldsymbol{\theta}^{(k)}, \boldsymbol{y})\tilde{\mathrm{p}}(\boldsymbol{\theta}^{(k)}|\boldsymbol{y})\Delta\boldsymbol{\theta}^{(k)} \tag{5.36}$$

$$\tilde{\mathrm{p}}(\boldsymbol{\theta}_j|\boldsymbol{y}) = \int \tilde{\mathrm{p}}(\boldsymbol{\theta}|\boldsymbol{y})\mathrm{d}\boldsymbol{\theta}_{-j}, \tag{5.37}$$

where $\{\tilde{\mathrm{p}}(x_i|\boldsymbol{\theta}^{(k)}, \boldsymbol{y})\}$ and $\{\tilde{\mathrm{p}}(\boldsymbol{\theta}^{(k)}|\boldsymbol{y})\}$ are the density values calculated in each point θ_k found by non-parametrically exploring the hyperparameter space of $\tilde{\mathrm{p}}(\boldsymbol{\theta}|\boldsymbol{y})$. The terms $\Delta\boldsymbol{\theta}^{(k)}$ are the area weights of the finite sum integration method. To compute the integral of Equation 5.37 the strategy proposed by Rue et al. [41] and detailed by Martins et al. [42] is the use of an interpolation function $I(\boldsymbol{\theta}|\boldsymbol{y})$ with the values already obtained from the grid exploration of the hyperparameter joint posterior distribution, $\{\mathrm{p}(\boldsymbol{\theta}^{(k)}|\boldsymbol{y})\}$:

$$\tilde{\mathrm{p}}(\boldsymbol{\theta}_j|\boldsymbol{y}) = \int I(\boldsymbol{\theta}|\mathrm{y})\mathrm{d}\boldsymbol{\theta}_{-j}. \tag{5.38}$$

In order to define the k integration points, `R-INLA` offers two possibilities: the grid method which uses nodes as points on an even grid around the mode of $\tilde{\mathrm{p}}(\boldsymbol{\theta}|\boldsymbol{y})$ and the Central Composite Design (CCD) method which selects a smaller set of nodes around the mode, and thus is less costly. We can also decide to simply use the mode [41, 46].

5.1.4 Spatial modelling: Stochastic Partial Differential Equation (SPDE) approach

The INLA framework relies on the latent field being formulated as a GMRF. This property is fundamental for efficient inference, independently of whether the model includes spatial dependence. In non-spatial settings, GMRFs arise, for example, in the representation of temporal or structured random effects. The incorporation of spatial dependence simply extends this principle by defining a spatial GMRF.

We already have a clue on how to incorporate spatial dependence. In Section 3.4 we included in the LGCP intensity model a spatial random effect, S(x), as a Gaussian Random Field. This Latent Gaussian Random Field can reflect inter-point dependency through its covariance function.

The covariance function is typically assumed to be stationary, meaning it depends only on the distance between any pair of locations. We also expect covariance to decrease as the distance increases, embodying Tobler's First Law of Geography: *"everything is related to everything else, but near things are more related than distant things"*. Furthermore, we must guarantee the covariance function is positive definite [24], a requirement that is not always straightforward to satisfy. Nevertheless, several well-studied functions in the literature meet these conditions, notably the exponential decay covariance function and the Matérn covariance function. The *Matérn class* is a family of covariance functions of the form

$$C(h) = \sigma^2 \, r\big(h; \nu, \kappa\big) \tag{5.39}$$

where

$$r(h; \nu, \kappa) = \frac{1}{\Gamma(\nu) 2^{\nu-1}} \big(\kappa h\big)^{\nu} K_{\nu}\big(\kappa h\big), \quad h = ||\boldsymbol{u} - \boldsymbol{v}|| \geq 0. \tag{5.40}$$

Each function in this family is defined by a set of positive parameters: the marginal variance σ, the range (or scale) κ and the smoothness (or shape) ν. The scale, κ, has units of distance and controls correlation decay with distance, while ν corresponds to the order of the modified Bessel function $K_{\nu}(u)(\cdot)$. $\Gamma(\cdot)$ designates the complete Gamma function and $||\cdot||$ is the Euclidean distance. Frequently, we utilize the practical range, ρ, defined as the distance where correlation drops to ≈ 0.14 [49], which corresponds to the expression

$$\rho \approx \frac{\sqrt{8\nu}}{\kappa}. \tag{5.41}$$

The reason for us to focus on the Matérn covariance function is the *Stochastic Partial Differential Equation (SPDE)* approach. Lindgren et al. discovered that the solution, $Z(\boldsymbol{u})$, to a particular SPDE of the form

$$(\kappa - \nabla)^{\alpha/2}(\tau Z(\boldsymbol{u})) = \mathcal{W}(\boldsymbol{u}) \tag{5.42}$$

is a continuous Gaussian Random Field with Matérn covariance. The process $\mathcal{W}(\boldsymbol{u})$ denotes Gaussian white noise process, $\nabla = \sum_{i=1}^{d} \frac{\partial^2}{\partial u_i^2}$ is the Laplacian operator, where d is the dimension of the space, and $\alpha = \nu - \frac{d}{2}$ is an integer. We can also express the marginal variance in the Matérn covariance function in terms of the parameters α, ν and κ,

$$\sigma^2 = \frac{\Gamma(\nu)}{\Gamma(\alpha)(4\pi)^{\alpha/2}\kappa^{2\nu}} \tag{5.43}$$

and τ is proportional to σ. By choosing appropriate values for κ and α, it is possible to control the properties of the resulting solution. However, for computational purposes, it is necessary to use a discrete representation of this continuous solution. Lindgren et al. demonstrated that this can be achieved through the Finite element method (FEM), yielding a discrete Gaussian Markov Random Field (GMRF) representation of the solution.

Utilizing FEM, we begin by searching for a solution to the weak (integral) version the SPDE 5.42, $Z(\boldsymbol{u})$, of the form:

$$Z(\boldsymbol{u}) = \sum_{i=1}^{n} z_i \phi_i(\boldsymbol{u}), \tag{5.44}$$

where n is the number of vertices (or nodes) in a triangular mesh of non-intersecting adjacent triangles, $\boldsymbol{z} = (z_1, \ldots, z_n)^T$ is an unknown multivariate Gaussian random vector and $\{\phi_i(\boldsymbol{u})\}_{i=1}^{n}$ is the set of known deterministic basis functions.

The SPDE approach, therefore, allows INLA to determine the prior distribution of the spatial random effect. Notice however, that the user still has to specify the hyperparameters, i.e., the parameters of the Matérn covariance, encapsulated in vector $\boldsymbol{\theta}$. Typically, one either uses the default settings of INLA or utilizes PC-priors for σ and the practical range g or specifies PC-priors for these parameters.

We can choose a set of locations u_i at which to evaluate the solution 5.44, and project it onto these points using a projection matrix, A. Specifically, the number of rows in A corresponds to the number of locations and the number of columns to the number of mesh vertices. Each entry a_{ij} of A is thus filled with the value of the basis function j at location i and each row has at most three non-zero values that sum to 1 and are defined by barycentric coordinates. Following Figure 5.1, we can distinguish three different situations:

- Red point: the location lies inside a triangle but does not coincide with any vertex or edge. The row of A corresponding to this point contains three non-zero values, each corresponding to one of the triangle's vertices. These values are obtained through linear interpolation of the basis functions within the triangle and sum to 1. In practice this means the value for each of the participating basis functions is obtained by dividing the area of the smaller triangle opposite to the vertex by the area of the bigger mesh triangle.

- Blue point: the location falls on an edge of a triangle (but not at a vertex). The row in A for this point contains two non-zero entries corresponding to the two vertices connected by that edge. Again, the values are linearly interpolated and sum to 1.

- Green point: the location coincides with a mesh vertex. In this case, the corresponding row in A contains a single 1 in the column associated with that vertex, and zeros elsewhere.

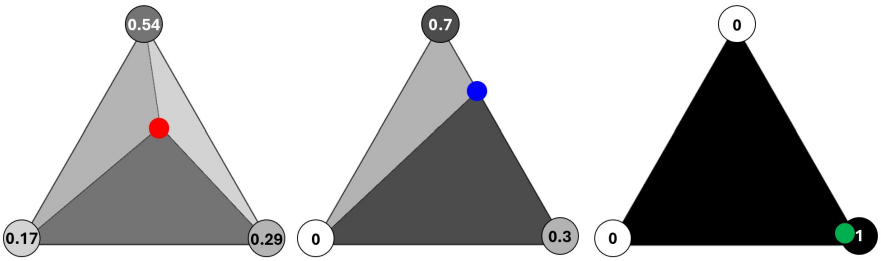


Figure 5.1: Three different scenarios of barycentric coordinates attribution in a triangle. In the vertices of the triangles we have the coordinate values for the coloured point. Figure inspired by [49].

Following this coordinate generation system we obtain basis functions in the FEM mesh similar to those represented in Figure 5.3 for two different mesh vertices.

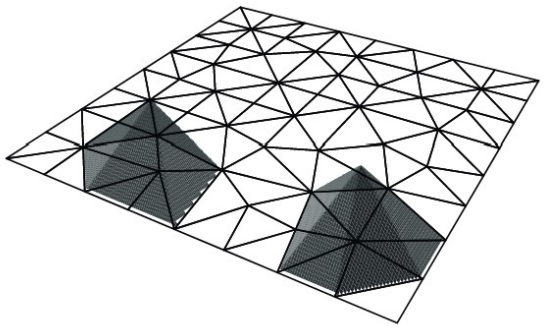


Figure 5.2: Mesh and basis function non-zero domain for two different mesh vertices. Image from [49].

The fact that each basis function is non-zero only in the neighbourhood of its associated vertex is what conveys the Markovian properties to the final solution. When we solve the weak form of the SPDE by plugging in the solution from Equation 5.44 using these basis functions, we obtain a zero-mean Gaussian distribution for the weights vector with a sparse precision matrix Q,

$$Z \sim \mathcal{N}\big(\mathbf{0}, Q^{-1}(\boldsymbol{\theta})\big). \tag{5.45}$$

Another aspect requiring special care is the approximation the likelihood of the log-Gaussian Cox process over a continuous spatial domain, in order to accurately integrate the intensity over space. For this purpose, a dual mesh is constructed from the (primal) mesh, by connecting the centroids of the surrounding triangles, forming polygons around each node as seen in Figure 5.2. The area of each dual polygon provides a natural integration weight for the Poisson likelihood, ensuring that the discretized model correctly approximates the continuous point process, as discussed by Simpson et al. [45].

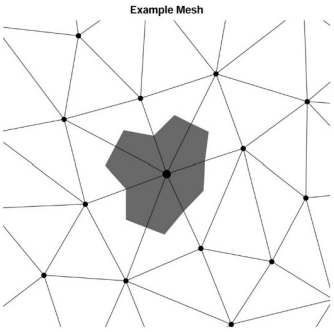


Figure 5.3: Dual mesh polygon represented for one of the (primal) mesh vertices. Image by Flagg and Hoegh [75].

5.1.4.1 Implementing a spatial LGCP model using `R-INLA`

We have implemented a spatial LGCP model with covariates fitted through the INLA + SPDE approach. Within the LGM framework presented in general in Equation 5.18, we can define our model as follows:

$$\begin{aligned} \mathbf{x} \mid \boldsymbol{\theta} &\sim \mathcal{N}\big(0, Q(\boldsymbol{\theta})^{-1}\big) \\ \eta_i(\mathbf{x}) &= \beta_0 + \mathbf{X}_i^{\top}\boldsymbol{\beta} + (A\mathbf{z})_i \\ y_i \mid \mathbf{x}, \boldsymbol{\theta} &\sim \text{Poisson}\big(w_i \exp\{\eta_i(\mathbf{x})\}\big) \end{aligned} \tag{5.46}$$

The hyperparameters $\boldsymbol{\theta} = [\sigma_z^2, \rho]$ control the spatial Gaussian field $\mathbf{z}$, where σ_z^2 is the marginal variance and ρ is the spatial range, both of which were assigned penalized complexity (PC) priors. The latent vector $\mathbf{x} = (\boldsymbol{\beta}, \mathbf{z})$ contains the regression coefficients and the spatial field, and is Gaussian with mean zero and precision $Q(\boldsymbol{\theta})$. The linear predictor contains the intercept term β_0, a set of spatially varying covariates $\boldsymbol{X}_i$ and A is the projection matrix interpolating the spatial field from mesh nodes to observation locations. The observed counts y_i are Poisson distributed with mean $\mathbb{E}[\boldsymbol{y}] = w_i \exp\{\eta_i(\mathbf{x})\}$, where w_i represents the integration weight associated with observation i, obtained from the dual mesh used to discretise the domain.

The regression coefficients β_j have vague Gaussian priors, since we use INLA's default specifications. For the SPDE model, as demonstrated by Fuglstad et al. [76], the joint PC-prior can be defined as

$$\mathrm{p}(\sigma, \rho) = d/2\, \tilde{\lambda}_1 \tilde{\lambda}_2\, \rho^{-d/2-1} \exp\{-\tilde{\lambda}_1 \rho^{-d/2} - \tilde{\lambda}_2 \sigma\}, \quad \sigma > 0,\ \rho > 0, \tag{5.47}$$

where parameters $\tilde{\lambda}_1$ and $\tilde{\lambda}_2$ correspond to

$$\tilde{\lambda}_1 = -\log(\alpha_1)\, \rho_0^{d/2} \quad \text{and} \quad \tilde{\lambda}_2 = -\frac{\log(\alpha_2)}{\sigma_0}. \tag{5.48}$$

and are associated to ρ and σ by

$$\mathrm{P}(\rho < \rho_0) = \alpha_1 \quad \text{and} \quad \mathrm{P}(\sigma > \sigma_0) = \alpha_2. \tag{5.49}$$

We specified in INLA that a PC-prior more on the vague realm by deciding a probability of $\alpha_1 = 0.5$ and a range of $\rho_0 = 1000$ m. We considered that σ should be encouraged to be small, an thus settled for a common specification of $\sigma_0 = 1$ and $\alpha_2 = 0.01$.

The implementation of this model in `R-INLA` requires a sequence of well-defined steps. The pictogram of Figure 5.5 explains the code workflow in `R-INLA`, assembled by following Moraga et al [58] and Krainski et al. [49], and which can be consulted in annex A alongside the code for the automatic covariate selection procedure. We do estimation and prediction jointly, so that we can directly retrieve from the model output the predicted intensity for a set locations covering the study window in a prediction grid. Notice also that we need to assign values to each data point, mesh vertex and prediction location, so that we can build an estimation and prediction stacks informing INLA of the linear predictor structure and data to be included in the model. We follow here the Simpson et al. [45] "Off the grid" approach, since we create an augmented dataset, with mesh nodes and observed locations. The weights we include in the stacks of step 7 correspond to dual mesh polygon areas for mesh nodes and the value zero for observed and predicted locations.

INLA requires the creation of an inner mesh and outer mesh. Since our study window's border are very irregular and we have several events near the borders, we applied a buffer of 500 m to this border and defined that as the inner mesh border (Figure 5.4). We tuned the parameters `max.edge` to 300 m for the inner mesh and 2000 m for the outer mesh and `cutoff` at 150 m. In this setting, the parameter `max.edge` controls the maximum allowed edge length of the triangles in the mesh, with smaller values producing a finer mesh and larger values producing a coarser mesh, and `cutoff` parameter specifies a minimum distance between points: events that are closer than this threshold are merged so that the mesh does not become unnecessarily dense in areas with many nearby points. The chosen values were decided based on the distribution of distances between pairs of events and also on how thin the mesh needed to be for the algorithm to run properly.

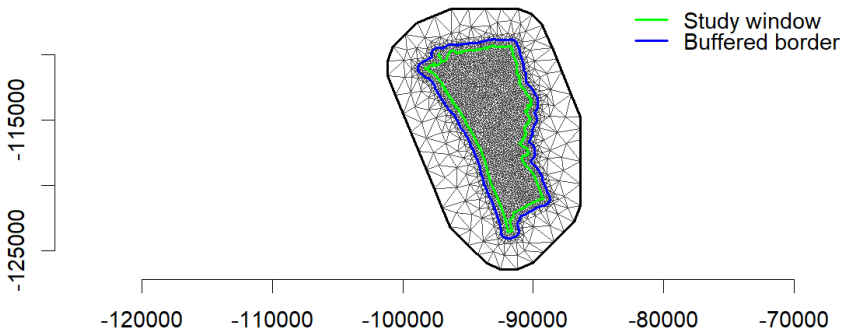


Figure 5.4: Mesh built using function `inla.mesh.2d`.

5.1.5 Model evaluation criteria

A simple residual analysis to assess the quality of the model fit, by calculating the Pearson residuals for each grid cell i [75]:

$$\text{res}_i = \frac{O_i - E_i}{\sqrt{E_i}}, \tag{5.50}$$

where O_i corresponds to the observed number of crimes in cell i and E_i to the expected number of crimes based on the predicted intensity.

To compare different fitted models we look at the Deviance Information Criterion (DIC) and Watanabe–Akaike Information Criterion (WAIC) [77], automatically calculated by `R-INLA`. The DIC is defined as

$$\text{DIC} = \bar{D} + p_D, \tag{5.51}$$

where $\bar{D} = \mathbb{E}_{\theta|y}[D(\theta)]$ is the posterior mean deviance and $p_D = \bar{D} - D(\hat{\theta})$ is the effective number of parameters, with $\hat{\theta}$ denoting the posterior mean of the parameters. Lower DIC values indicate a better trade-off between model fit and complexity. WAIC, on the other hand, is a fully Bayesian predictive measure given by

$$\text{WAIC} = -2\sum_{i=1}^{N} \log\left(\mathbb{E}_{\theta|y}[\text{p}(y_i|\theta)]\right) + 2\sum_{i=1}^{N} \text{Var}_{\theta|y}\left(\log \text{p}(y_i|\theta)\right), \tag{5.52}$$

where the first term is the expected log point-wise predictive density and the second term is a correction for overfitting. WAIC provides an estimate of the model's out-of-sample predictive performance, making it particularly useful for comparing competing models.

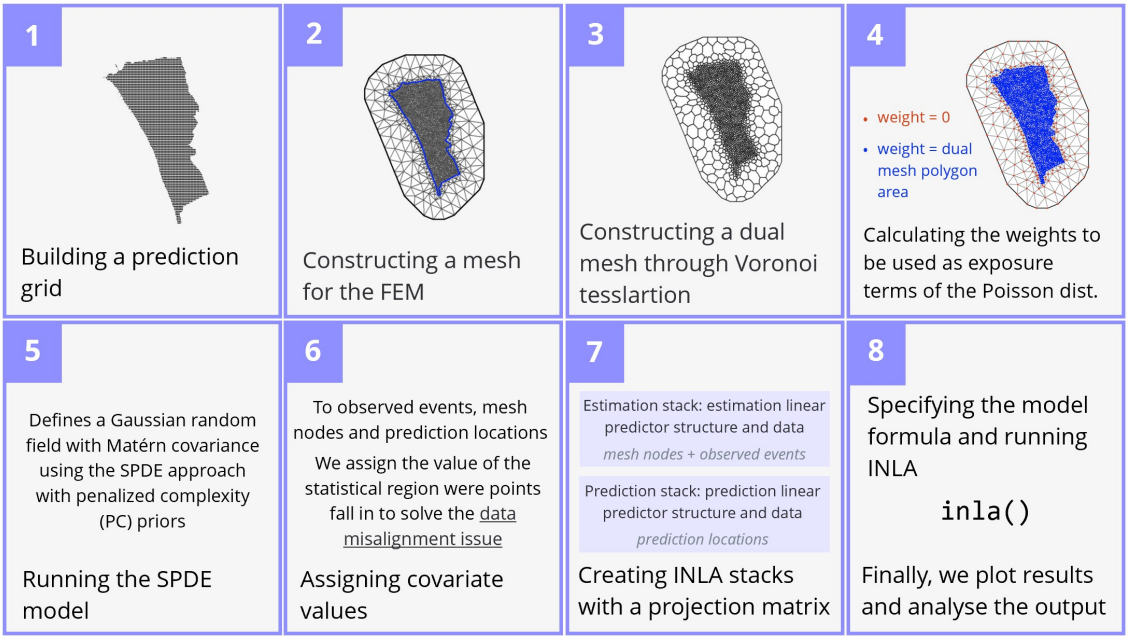


Figure 5.5: Workflow in `R-INLA` for the implementation of the spatial LGCP model.

5.2 Results

5.2.1 Identification of candidate covariates

Inspired by the studies reviewed in Chapter 2, we focused on socio-economic variables collected in the 2021 Census [78, 79]. These variables include raw counts related to demographics, education, and employment (both totals and disaggregated by gender), as well as housing and building characteristics.

The 2021 Census provides data at different areal levels, with the smallest publicly available units being the Statistical Section (STSec) and the Statistical Subsection (STSub). The latter consists of smaller subdivisions within each Statistical Section. Figure 5.6 shows the STSec and STSub divisions in our study area. The smaller unit, STSub is naturally preferable, however only a limited set of 32 variables are available, in comparison with the 177 from the STSec dataset. Thus, we opted for the STSec dataset.

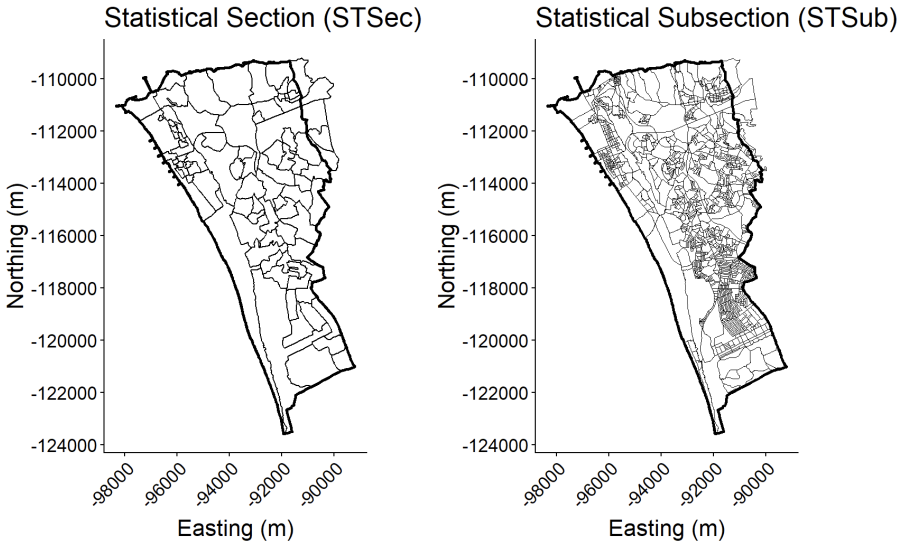


Figure 5.6: Statistical Section STSec and Statistical Subsection STSub regions within the study window.

In our study, instead of using directly raw counts, we used always densities, dividing each covariate value by the area of the corresponding region.

5.2.2 Covariate selection

5.2.2.1 Automatic selection procedure

Many of the 2021 Census columns represent sub-levels of broader categories. In the STSec dataset we identified a hierarchy structure with up to 5 levels of detail. We then applied the automatic covariate selection procedure described earlier in Section 5.1.1 separately for levels 1, 2, 3, 4, and 5 (leaf level). The VIF threshold was defined at 5 and we used 200 bootstrap samples.

The bootstrap procedure encountered convergence issues at some hierarchy levels, as noted in Table 5.2. For hierarchy level 3, the procedure failed because of an error in the `nb.glm()` function, indicating a struggle to find valid regression coefficients. For hierarchy level 2, only one additional bootstrap iteration was required to compensate for a single run, in which `nb.glm()` did not converge within the pre-set iteration limit. This illustrates the challenges of applying bootstrap procedures to models with complex parameter estimation, such as the the negative binomial GLM which can struggle with estimating the dispersion parameter. Indeed, this approach is usually less robust than, for instance, applying the traditional `glm()` function within the Poisson family.

Table 5.2: Automatic covariate selection process details.

Dataset	**Number of variables**			**Bootstrap successful?**
	Original	**near-zero Var**	**VIF < 5**	
Hierarchy Level 1	5	None	4	Yes
Hierarchy Level 2	45	None	16	Yes (One iteration was ignored and redone)
Hierarchy Level 3	57	1	18	No (Negative binomial convergence issues)
Hierarchy Level 4	94	1	19	Yes
Hierarchy Level 5	111	1	17	Yes

The results of the bootstrap selection are presented in Table 5.3, showing all variables from different hierarchy levels of the STSec dataset along with the frequency with which they were selected by the bootstrap procedure. Only variables selected more or exactly 10% of the time are reported. Covariates "Buildings Constructed 1991–2000" and "Buildings Constructed 2011–2015" were chosen for modelling, as they were selected more than 50% of the time at all hierarchy levels where they were present. For hierarchy level 1 we see "Total Buildings" was also selected in more than 50% of the bootstrap iterations. Other variables generally did not approach this threshold. Nevertheless, some models including additional covariates were also implemented, particularly those that appeared more frequently in subsequent selections, to see if in the LGCP context they could bring any improvement to the fit.

Table 5.3: Bootstrap selection frequencies of covariates from STSec dataset (translated to English). Only variables selected more than 10% of the time are shown.

Covariate	**Selection Frequency**
Hierarchy Level 1	
Total Buildings	0.52
Total Individuals	0.16
Total Dwellings	0.16
Hierarchy Level 2	
Buildings Constructed 1991–2000	0.78
Buildings Constructed 2011–2015	0.72
Households With More Than 5 People	0.28
Buildings Constructed 2016–2021	0.18
Individuals With Higher Education	0.17
Buildings Constructed 1981–1990	0.16
Buildings Constructed 2001–2005	0.12
Buildings Constructed 2006–2010	0.12
Buildings Constructed 1919–1945	0.10
Hierarchy Level 4	
Buildings Constructed 1991–2000	0.71
Buildings Constructed 2011–2015	0.70
Buildings With Medium Repair Needs	0.22
Domestic Individuals	0.20
Individuals Aged 65–69 (Male)	0.18
Buildings Constructed 2016–2021	0.17
Buildings Constructed 1981–1990	0.16
Classical Family Dwellings	0.16
Buildings Constructed 2001–2005	0.14
Buildings Constructed 2006–2010	0.12
Buildings With Light Repair Needs	0.11
Hierarchy Level 5	
Buildings Constructed 1991–2000	0.80
Buildings Constructed 2011–2015	0.66
Individuals Employed in Primary Sector (Male)	0.26
Buildings With Medium Repair Needs	0.20
Domestic Individuals	0.18
Buildings Constructed 2016–2021	0.18

Continued on next page

Table 5.3 – continued from previous page

Individuals With Higher Education (Male)	0.16
Buildings Without Repair Needs	0.13
Dwellings With Area > 200m^2	0.10
Buildings Constructed 2006–2010	0.10
Unemployed Individuals Searching 1st Job (Male)	0.10

5.2.2.2 Manual variable creation

Taking inspiration from the socio-economic indices studied in the various works analysed in Chapter 2, we created a set of 13 covariates from the census 2021 STSec original variables that could have a more meaningful interpretation. Table 5.4 contains the names and descriptions of the manually created covariates.

Table 5.4: Manually constructed covariates from Census 2021 STSec variables.

Covariate	**Definition**
`urban_level`	Measures the proportion of tall buildings in the area, giving a weight of 0.8 to buildings with 5 or more floors and 0.2 to buildings with 3–4 floors, relative to the total number of classical buildings.
`modern_constr_prop`	Represents the share of buildings constructed after the year 2000 relative to the total number of classical buildings.
`degradation_level`	Weighted measure of building repair needs, assigning 0 to buildings needing no repairs, 1 to minor repairs, 2 to medium repairs, and 3 to major repairs, averaged across all classical buildings.
`vacant_prop`	Proportion of dwellings that are currently vacant, relative to the total number of dwellings.
`parking_ratio`	Ratio of habitual residences with parking to those without parking.
`homesize_measure`	Weighted measure of dwelling size, assigning 0 to homes under 50 m^2, 1 to 50–99 m^2, 2 to 100–149 m^2, 3 to 150–199 m^2, and 4 to homes of 200 m^2 or more, averaged across all habitual residences.
`pop_density`	Total number of individuals living in the area.
`young_prop`	Proportion of individuals aged 15 to 29 in the total population.
`old_prop`	Proportion of individuals aged 65 or older in the total population.

Continued on next page

Table 5.4 – continued from previous page

`male_to_female_ratio`	Ratio of male to female individuals in the population.
`highereduc_prop`	Proportion of individuals aged 25 or older who have completed higher education.
`unemployed_prop`	Proportion of economically active individuals who are unemployed.
`immigrants_prop`	Proportion of individuals who were born outside the country, relative to the total population.

We alert the covariates `homesize_measure` and `highereduc_prop` were not included for modelling as they had higher than 0.7 correlation values with other covariates.

5.2.3 Model results

We ran six different models following the specifications indicated in Table 5.5. All models included the structured spatial random effect and an intercept term. Fits were compared in terms of the DIC and WAIC criteria in order to select a few models worthy of a more in-depth result analysis. We also took into consideration the marginal Log-likelihood values as well as the posterior results of the range and standard deviation parameters associated with the Matérn covariance function.

To complement the tabled results, in annex III we included the plotted posterior distributions for the fixed effects coefficients and hyperparameters, and the 95% credible intervals for the posterior intensity and posterior Gaussian Random Field (GRF) for Models 1, 5 and 6.

Table 5.5: Covariates specification for the spatial LGCP models.

Model Specification	**Covariates included**
Model 1	Intercept only
Model 2	Total Buildings
Model 3	Buildings Constructed 1991-2000, Buildings Constructed 2011-2015
Model 4	Buildings Constructed 1991-2000, Buildings Constructed 2011-2015, Buildings With Medium Repair Needs
Model 5	Buildings Constructed 1991-2000, Buildings Constructed 2011-2015, Buildings With Medium Repair Needs, Individuals Employed in Primary Sector (Male)
Model 6	Manual covariates specification

Across the six model specifications, differences in DIC and WAIC are generally modest (Table 5.6), indicating that the baseline spatial LGCP (Model 1; DIC = 30239.23, WAIC = 31626.74, marginal log-likelihood = –15260.22) captures a substantial portion of the spatial signal. Model 6 achieves the lowest DIC (30093.76), suggesting that its richer covariate structure and spatial formulation improve the model fit compared to simpler specifications. However, Model 5 attains a lower WAIC (31671.29) than Model 6 (31928.89), indicating

better predictive performance and balance between fit and complexity. Marginal log-likelihood values are broadly similar across models, with smaller differences. In this context, for detailed analysis, we focus on Models 5 and 6.

Table 5.6: Comparison of spatial LGCP model specifications: fit statistics.

Model specification	**DIC**	**WAIC**	**Marginal Log-likelihood**
Model 1	30239.23	31626.74	−15260.22
Model 2	30240.15	31637.28	−15266.28
Model 3	30242.88	31636.57	−15271.84
Model 4	30240.77	31654.36	−15276.72
Model 5	30219.61	31671.29	−15273.21
Model 6	30093.76	31928.89	−15268.11

Examining the intercept and hyperparameters, the baseline Model 1 has $\beta_0 = -12.062$, a spatial range of ≈ 1.33 km, and a standard deviation of 1.68 (Table 5.7). Incorporating covariates in Models 2–5 slightly lowers the intercept and modestly reduces the spatial range, reflecting that these covariates capture part of the spatial structure. Across Models 4-6 the spatial range decreases from ≈ 1.33 km in Model 1, indicating that the added covariates explain some of the broader-scale spatial variation. Model 6, which includes the full set of covariates, shows an intercept of $\beta_0 = -12.047$, the shortest spatial range of ≈ 1.14 km, and a slightly higher residual standard deviation of 1.77, indicating that while covariates explain additional variation, substantial fine-scale spatial structure remains. Overall, these results complement the fit statistics, confirming that Models 5 and 6 provide the most informative balance between covariate effects and residual spatial structure.

Table 5.7: Comparison of spatial LGCP model specifications: intercept coefficient and hyperparameter posterior mean and respective 95% credible intervals.

Model specification	**Posterior mean β_0 [95%CI]**	**Posterior mean range [95%CI]**	**Posterior mean standard deviation [95%CI]**
Model 1	−12.062 [−12.715; −11.455]	1325.34 [1060.32; 1657.16]	1.68 [1.42; 1.99]
Model 2	−12.057 [−12.717; −11.447]	1320.31 [1053.28; 1652.02]	1.67 [1.41; 1.99]
Model 3	−12.045 [−12.701; −11.437]	1320.33 [1054.80; 1650.01]	1.67 [1.41; 1.98]
Model 4	−12.029 [−12.669; −11.438]	1290.21 [1030.10; 1612.78]	1.65 [1.39; 1.96]
Model 5	−12.020 [−12.645; −11.442]	1251.15 [999.68; 1562.08]	1.65 [1.40; 1.95]
Model 6	−12.047 [−12.711; −11.450]	1135.35 [912.19; 1410.19]	1.77 [1.50; 2.09]

We note that these hyperparameter estimations are within expectations given the non-parametric analysis, which admitted clustering effects up to 1.25 km. In addition, it is consistent with the values of Zhao et al. [28] which also use a Matérn covariance structure with the INLA + SPDE approach. Nevertheless, the estimate is above what it is registered in other studies, as can be assessed by consulting Table 2.1 from Section 2.2 of Chapter 2 where the range is typically lower, on the hundreds scale. We should, however, be

cautious with comparisons, as some authors preferred an exponential covariance function and SEHP models that rely on a different mechanism (the triggering effect) to explain near-repeat victimization.

The posterior mean intensity surfaces (Figures 5.7) represent the combined effects of the intercept, covariates, and the spatial random field, while the posterior mean Gaussian Random Field (GRF) surfaces (Figure 5.9) isolate only the contribution of the latent spatial field.

The posterior mean intensity surfaces are similar for the three models, with a few more high intensity peaks in the case of Model 6. In the three surfaces we can see the same hotspots already identified non-parametrically. Observing the posterior mean GRF surfaces, we visually assess that the contribution from the GRF doesn't change considerably between the three cases. We would ideally like to see a general decay in the GRF contribution for Models 5 and 6, as more explanatory power is attributed to the covariates.

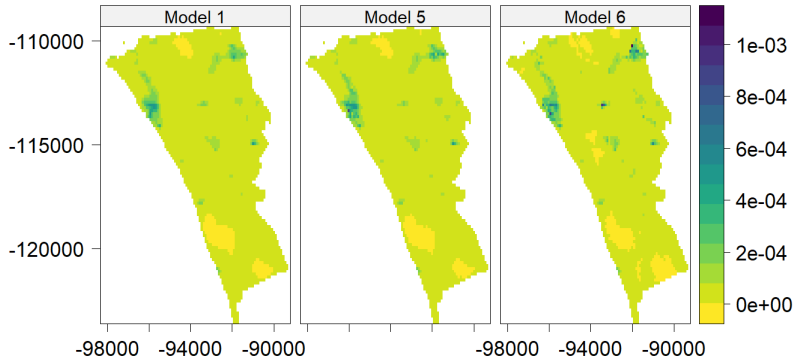


Figure 5.7: Posterior mean intensity predicted surfaces for Models 1, 5 and 6.

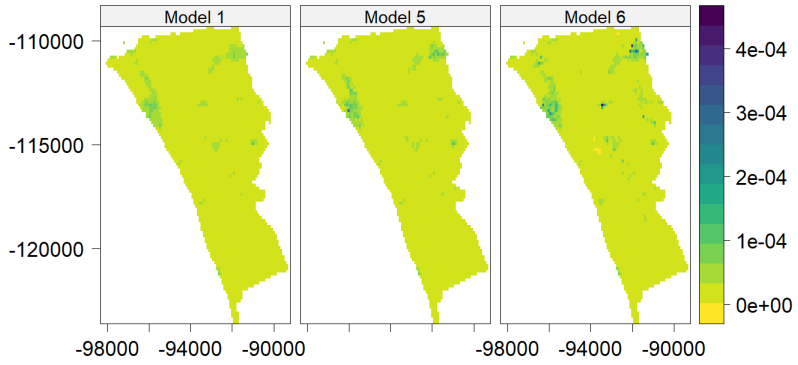


Figure 5.8: Posterior standard deviation intensity predicted surfaces for Models 1, 5 and 6.

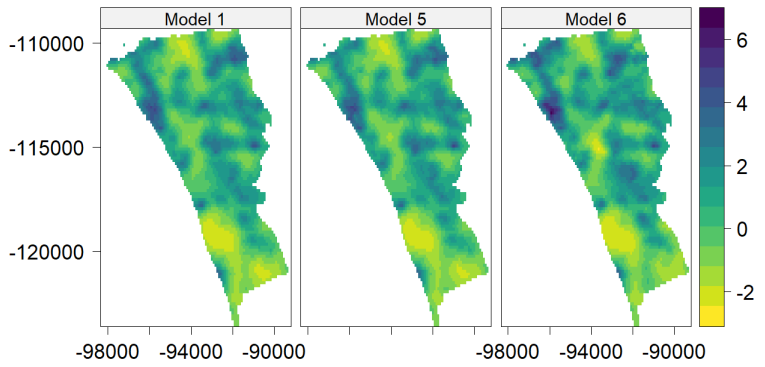


Figure 5.9: Posterior mean Gaussian Random Field (GRF) intensity predicted surfaces for Models 1, 5 and 6.

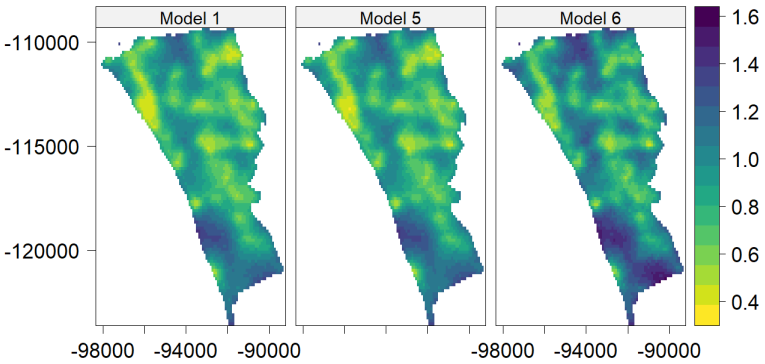


Figure 5.10: Posterior standard deviation Gaussian Random Field (GRF) surfaces for Model 1, 5 and 6.

Figure 5.11 displays the posterior covariance functions for Models 1, 5, and 6, showing the posterior mean and 95% credible intervals. The covariance functions describe how spatial dependence decays with distance, as implied by the Matérn covariance structure underlying the SPDE models. For Model 1, the posterior covariance is highest at short distances, reflecting the fact that the GRF is responsible for capturing all spatial dependence in the absence of covariates. In contrast, Models 5 and 6 exhibit lower posterior covariances at short distances, indicating that part of the spatial correlation has been explained by the inclusion of covariates. The narrower credible intervals for these models also suggest that the estimated spatial structure is more stable when explanatory variables are included.

As distance increases, the covariance decays towards zero for all three models, consistent with the finite correlation range of the Matérn specification. The rate of decay is similar across models, suggesting that the spatial range parameter is relatively robust to the inclusion of covariates.

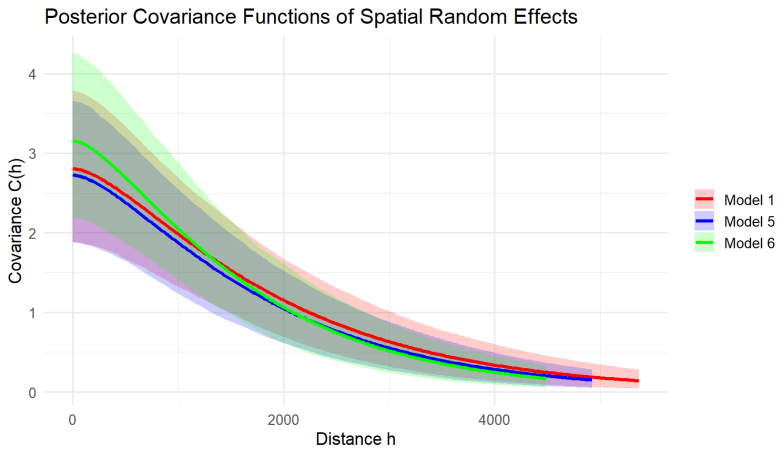


Figure 5.11: Posterior covariance for Models 1, 5 and 6. The image shows the posterior mean (line) and 95% credible interval (shaded).

The Pearson-type residual maps from Figure 5.12 provide an additional check on model adequacy by comparing observed and predicted point patterns. Across Models 1, 5, and 6, the residual surfaces are largely dominated by white and light blue shading, indicating that the fitted intensities closely match the observed data over most of the study region or that, at most, there is a slight overestimation. We also see localized red spots, with values up to +10 in some cases, which highlight areas where the models consistently under-predicted the observed intensity, especially in hotspot regions.

Importantly, the overall residual structure is very similar across models, reflecting that while covariates help explain some of the spatial heterogeneity, the main spatial clustering features is being captured by the LGCP framework, and the remaining residuals likely reflect unobserved local effects or stochastic variability.

Figure 5.12: LGCP Pearson-type residuals.

Table 5.8 lists the coefficient posterior means and 95% credible intervals for Models 5 and 6. We signal with the symbol (*) the significant variables, namely those with posterior mean estimates within the aforementioned interval.

Results from Model 5 indicate that areas with higher numbers of males employed in the primary sector experience higher street crime rates. Model 6 provides more information for the analysis by including urban, demographic, and socio-economic characteristics, providing a more detailed understanding of street crime patterns. Higher urban level, identifying areas with taller buildings, is strongly associated with increased crime, consistent with environmental criminology, as denser urban areas offer more targets for offenders. We recall that the most prevalent crime is the theft in motor vehicle and more urban areas may accommodate more vehicles. Demographic factors show significant influences: higher proportions of older residents and immigrants are linked to lower street crime, reflecting social control and cohesive community effects, where stronger informal supervision and neighbourhood networks may reduce crime opportunities. Conversely, higher unemployment and a male-dominated population are associated with increased street crime, consistent with social disorganization theory, indicating that structural strain and demographic imbalances weaken social control and facilitate opportunistic crimes.

Model 6 demonstrates that urbanity, demographic composition, and socio-economic strain jointly shape street crime patterns. This suggests that interventions targeting high-risk urban areas, communities with higher unemployment or demographic imbalances could help reduce street crime, while also addressing broader street crime risks in the city.

Table 5.8: Fixed effects coefficient posterior mean estimates and 95% credible intervals for Models 5 and 6.

Model 5	
Covariate	Coefficient posterior mean (95% CI)
Intercept (β_0)	$-12.020\,[-12.645;\,-11.442]\,(*)$
Buildings Constructed 1991-2000	$-0.130\,[-0.309;\,0.045]$
Buildings Constructed 2011-2015	$0.098\,[-0.064;\,0.260]$
Individuals Employed in Primary Sector (Male)	$0.339\,[0.185;\,0.497]\,(*)$
Buildings With Medium Repair Needs	$0.106\,[-0.013;\,0.225]$
Model 6	
Covariate	Coefficient posterior mean (95% CI)
Intercept (β_0)	$-12.047\,[-12.711;\,-11.450]\,(*)$
Urban level	$1.065\,[0.730;\,1.418]\,(*)$
Modern construction proportion	$0.152\,[-0.144;\,0.450]$
Degradation level	$-0.150\,[-0.324;\,0.024]$
Vacant proportion	$0.135\,[-0.050;\,0.321]$
Parking ratio	$0.024\,[-0.261;\,0.311]$
Population density	$-0.173\,[-0.416;\,0.061]$
Young proportion	$-0.246\,[-0.502;\,0.005]$
Old proportion	$-0.435\,[-0.735;\,-0.139]\,(*)$
Male-to-female ratio	$0.265\,[0.095;\,0.441]\,(*)$
Unemployed proportion	$0.440\,[0.200;\,0.686]\,(*)$
Immigrants proportion	$-0.553\,[-0.779;\,-0.336]\,(*)$

Relating our results to studies conducted in Portugal, we find common ground with the conclusions of Saraiva et al. [55] in their analysis of crime in the city of Porto. These authors reported that concentrated disadvantage is associated with higher levels of crime concentration, a tendency that may also underlie our finding of a positive association between crime intensity and the proportion of unemployed individuals as well as men employed in the primary sector, traditionally linked to lower wages and more precarious jobs. Saraiva et al. further observed that higher education levels are linked to reduced crime levels. Although education was not included as a covariate in our analysis, due to its very high correlation with unemployment, this mechanism may also be at play, given the behaviour of the unemployment proportion in our results. Adding to these reflections, we recall that Tavares and Costa [54] identified a positive relationship between property crime intensity in Lisbon and poverty, measured by income assistance. Despite differences in scope, both authors analysed broader sets of crime categories (in which are included street crime subcategories) and both studies found a positive association between crime and social disadvantage. This convergence suggests that the phenomenon is probably not limited to specific crime types, but reflects a broader dynamic consistent with social disorganisation theory.

Some differences were identified regarding the findings of Saraiva et al. [55]. For instance, in our case, regions with a higher proportion of men relative to women tend to show higher concentrations of crime, whereas the mentioned authors reported an opposite connection. One possible explanation is that gender imbalances may interact with opportunity structures and social dynamics differently depending on the local context. According to routine activity theory, areas with more men tend to exhibit increased levels of guardianship absence or a higher concentration of potential offenders, particularly if unemployment and precarious employment are also prevalent. Conversely, Saraiva et al.'s findings may reflect a situation where gender balance interacts differently with community social organisation.

Saraiva et al. [55] also note that building density has mixed effects in the city of Porto, whereas in our analysis higher densities of taller buildings reveal the strongest positive association with crime intensity. Considering that the most relevant subcategory in our study is theft in motor vehicles, it is reasonable to attribute this pattern to opportunity-driven dynamics: denser urban areas with taller buildings are more likely to concentrate both vehicles and potential offenders, thus amplifying the risk of victimisation. Povala et al. [20] also had identified this positive association in the residential areas component of their SAM-GLM model regarding burglary.

Finally, we would also like to note that Briz-Redón and Mateu [25] reported a positive association between the proportion of foreign-born population and robberies and property crimes, whereas in our case we observed a negative association for street crime. Moreover, Povala et al. [20] found that ethnic diversity was positively associated with crime intensity in the neighbourhood component of their model. A possible explanation for this divergence lies in the spatial and social context: while their findings reflect dynamics of central urban

neighbourhoods, our model is likely more strongly associated with the residential-area component characteristic of suburban regions. This suggests that the relationship between immigration, diversity, and crime may be context-dependent or more specific to particular crime types.

Chapter 6

Conclusions and future work

Street crime is a phenomenon that affects societies worldwide, not only in terms of safety but also regarding social cohesion, economic well-being, and the quality of life of communities. Understanding its patterns and drivers is essential for designing effective prevention strategies and allocating limited policing resources efficiently. In Portugal, although crime rates are relatively low compared to many other countries, street crime still represents a significant concern, particularly in urban areas where population density and social interactions create more opportunities for criminal activity.

Given this context, we set out to investigate the potential of spatial point pattern analysis, both exploratory and model-based, in studying the distribution of street crime in the Municipality of Almada. At the modelling stage, we focused on constructing a framework assuming that the crime point pattern originates from a Log-Gaussian Cox Process (LGCP), a point process particularly suitable for describing clustered patterns. Such is often the case with crime, governed by Weisburd's law of crime concentration at the microlevel. Model inference was carried out in a Bayesian framework using the Integrated Nested Laplace Approximation (INLA) combined with the Stochastic Partial Differential Equation approach, an increasingly popular alternative to the long-standing Markov Chain Monte Carlo methods. By applying state-of-the-art statistical methods to georeferenced street crime data, this study aimed to advance the understanding of how crime concentrates and evolves in microgeographic areas, while also contributing to the broader discussion of how such insights can inform intelligence-led policing strategies. The project encompassed the full analytical pipeline, from raw data preprocessing and covariate selection based on publicly available Portuguese datasets to the implementation and evaluation of the final model.

The literature review was essential to establish the foundations for determining the course of this project. It provided both the necessary theoretical background in criminology and an understanding of which approaches have proven successful in applying point pattern analysis to the study of crime. This review also highlighted the scarcity of studies conducted in Portugal or those focused on street crime. Existing investigations in this field primarily use Generalised Additive Models based on Self-Exciting Hawkes

Point Processes or Log-Gaussian Cox Processes. These approaches differ notably: the former is a mechanistic model with an intrinsic triggering component, suitable for modelling near-repeat victimization, while the latter is a more empirical approach, where crime explanation is enhanced through covariate incorporation. When well-specified, LGCP models demonstrate strong spatial prediction and temporal forecasting capabilities. Although international research is often directed towards burglary, it is still valuable to examine covariates from the various studies to understand potential influences on crime. This knowledge was then triangulated with Portuguese studies to have a more curated vision of the possible factors affecting crime in our case.

A working dataset was constructed consisting of 1426 georeferenced street events occurring between 2022 and 2023, restricted to the GNR-policed *freguesias* of Almada and includes precise coordinates in PT-TM06/ETRS89 CRS. The work was divided into two main stages: a pre-modelling stage and a modelling stage.

In the pre-modelling stage, non-parametric analyses using quadrat counts and kernel density estimations revealed several concentrated hotspots, notably in the north-east, along the *Costa da Caparica* coastline, and in the north-west corner of the study area. These areas should be prioritised for patrolling, as crime reduction here could yield a substantial decrease in overall crime within Almada's region under the GNR's jurisdiction, while minimising the spatial dispersion of resources. Naturally, with limited resources, the potential for crime displacement to other areas should also be considered. The adaptive kernel estimation further highlighted localized clusters, that could a secondary source of patrol focus. Second-order analyses, via the K-function and pair-correlation function, indicated a clear tendency toward clustering, with significant aggregation at short distances up to approximately 500 – 1250 meters, depending on the measure and homogeneity assumption. Spatio-temporal analysis suggested that clustering persists across both space and time, although the strength of spatial interaction diminishes with distance. CSR tests, including quadrat count-based and the inhomogeneous L-function rank envelope test, confirmed also that street crime exhibits statistically significant clustering at short ranges.

In the modelling stage, we implemented spatial LGCP models using the INLA + SPDE methodology, incorporating both selected census-derived covariates and a structured spatial random effect. We used automatic selection, combining the analysis of the Variance Inflation Factor (VIF) and automated bootstrap procedures based on a non-negative binomial classical model fitted to the crime counts per statistical section. We also used manually constructed socio-economic and urban indicators from the original covariates, which turned out to produce better results, emphasizing the importance of criminology backed covariate selection. We selected the best model using covariates selected automatically and the model incorporating manually created covariates as well as a baseline model including an intercept term and a structured spatial random effect for a further analysis. The inclusion of covariates reduced the spatial range of the Latent Gaussian Field, indicating that covariates explain a portion of the observed spatial clustering. Comparing our results on the socio-economic variable effects with the literature, specially within

Portugal, we find some strong evidence for social disorganisation theory. This suggests that the GNR is right in trying to follow a strategy of policing aligned with community proximity.

It is important to mention we should be cautious in analysis, since model selection shows a mild improvement in DIC from models including covariates compared to the intercept only model, although the latter was still preferred via WAIC. Posterior intensity surfaces and GRF contributions highlighted persistent hotspots consistent with exploratory non-parametric analyses. Key covariate effects revealed that higher urbanity, male-dominated populations, and unemployment were positively associated with street crime, whereas higher proportions of older residents and immigrants had a protective effect. Residual analysis shows also a struggle in the model to do accurate predictions, suggesting that the model specification could be improved.

Upon reflection, socio-economic Census 2021 covariates may not be optimal for modelling crime patterns. Literature suggests that points of interest (POIs) could provide stronger explanatory power, though such data are not readily available. Attempts to use aggregated census covariates at the Statistical Subsection level were unsuccessful in running the model in INLA. These challenges also reflect technical difficulties in implementing covariates within the `R-INLA` framework following Simpson et al.'s [72] approach to inference with INLA + SPDE. We also attempted the use of smoothed covariate maps to deal with the misalignment issue between areal covariates and point data, but did not achieve better results or easier computational performance.

Future work can follow several directions. We argue that, as a first step, a richer set of covariates should be explored, alongside integrating the temporal dimension, which is critical for capturing human activity patterns that drive crime dynamics. At this stage, the street network could be incorporated as a covariate map. In a more advanced phase, street crime could benefit greatly from a street network model.

We believe this work serves as a starting point, demonstrating that statistical analysis can provide valuable insights into social phenomena and offer guidance on which regions should be prioritized for resource allocation by the GNR in the municipality of Almada. While this study focused specifically on areas under GNR jurisdiction in Almada, the methodology could be generalized to other regions of Portugal with high criminal activity, offering a scientific reference framework to inform GNR patrol organization nationwide. To translate these findings into practice, we believe it is also essential to compare such analyses with officers' perceptions and the strategies currently in place.

Finally, during the course of the master thesis program it was possible to disseminate this work in several conferences under the title "Modelling Street Crime in Almada, Portugal, Using Point Processes". Intermediate results were presented in poster format at the international Spatial Statistics 2025 conference in Noordwijk, The Netherlands. At present, a possible publication of these results in the special issue of the conference is under review. Final results were presented orally both at the national conference of Sociedade Portuguesa de Estatística (SPE) 2025 in Algarve, Portugal and the Encontro

de Ciências Militares of 2025 in Lisbon, Portugal. The author is very grateful for the opportunity to share this work in these different contexts and for the encouragement of the thesis advisers to do so. This has been an enriching experience personally and, fundamentally, it demonstrates how ideas can reach a broader community and be part of scientific discussion.

Bibliography

[1] J. M. Lourenço. *The NOVAthesis LaTeX Template User's Manual*. NOVA University Lisbon. 2021. URL: https://github.com/joaomlourenco/novathesis/raw/main/template.pdf.

[2] A. A. Braga et al. "Hot Spots Policing of Small Geographic Areas Effects on Crime". In: *Campbell Systematic Reviews* 15.3 (2019), e1046. DOI: 10.1002/cl2.1046.

[3] D. Weisburd. "The Law of Crime Concentration and the Criminology of Place". In: *Criminology* 53.2 (2015), pp. 133–157. DOI: 10.1111/1745-9125.12070.

[4] L. W. Sherman, P. R. Gartin, and M. E. Buerger. "Hot Spots of Predatory Crime: Routine Activities and the Criminology of Place". In: *Criminology* 27.1 (1989), pp. 27–56. DOI: 10.1111/j.1745-9125.1989.tb00862.x.

[5] D. Weisburd and S. Amram. "The Law of Concentrations of Crime at Place: The Case of Tel Aviv-Jaffa". In: *Police Practice and Research* 15.2 (2014), pp. 101–114. DOI: 10.1080/15614263.2013.874169.

[6] D. Weisburd et al. "Crime Concentrations at Micro Places: A Review of the Evidence". In: *Aggression and Violent Behavior* 78 (2024), p. 101979. DOI: 10.1016/j.avb.2024.101979.

[7] C. Gill, A. Wooditch, and D. Weisburd. "Testing the "Law of Crime Concentration at Place" in a Suburban Setting: Implications for Research and Practice". In: *Journal of Quantitative Criminology* 33.3 (2017), pp. 519–545. DOI: 10.1007/s10940-016-9304-y.

[8] P. M. Dau et al. "How Concentrated Are Police on Crime? A Spatiotemporal Analysis of the Concentration of Police Presence and Crime". In: *Cambridge Journal of Evidence-Based Policing* 6 (2022), pp. 109–133. DOI: 10.1007/s41887-022-00079-6.

[9] S. Roy and I. R. Chowdhury. "Three Decades of GIS Application in Spatial Crime Analysis: Present Global Status and Emerging Trends". In: *The Professional Geographer* 75.6 (2023), pp. 882–904. DOI: 10.1080/00330124.2023.2223250.

[10] S. Chainey and J. Ratcliffe. *GIS and Crime Mapping*. Mastering GIS: Technology, Applications and Management Series. John Wiley & Sons, Ltd, 2005.

[11] Institute for Economics & Peace. *Global Peace Index*. 2025. URL: https://www.economicsandpeace.org/global-peace-index/ (visited on 2025-08-26).

[12] Guarda Nacional Republicana (GNR). *Estratégia Da Guarda 2025 - Uma Estratégia Centrada Nas Pessoas*. Portal da Guarda Nacional Republicana. 2019. URL: https://www.gnr.pt/estrategia.aspx (visited on 2024-11-15).

[13] Pereira, Ana Rosa Pires. "Do Modelo de Policiamento Tradicional ao Modelo Intelligence-Led Policing: Estudo Comparativo". MA thesis. Lisboa: Academia Militar, 2020.

[14] A. Baddeley, E. Rubak, and R. Turner. *Spatial Point Patterns: Methodology and Applications with R*. 1st ed. Interdisciplinary Statistics Series. Chapman and Hall/CRC, 2015. ISBN: 978-1-4822-1021-7.

[15] Cohen, L. E. and Felson, M. "Social change and crime rate trends: A routine activity approach". In: *American Sociological Review* 44 (1979), pp. 588–608.

[16] D. Cornish and R. V. Clarke. *The Reasoning Criminal: Rational Choice Perspectives on Offending*. Hague: Springer-Verlag, 1986.

[17] P. J. Brantingham and P. L. Brantingham. *Environmental Criminology*. Beverly Hills, CA: Sage Publications, 1981.

[18] D. Weisburd, E. Groff, and S.-M. Yang. *The Criminology of Place: Street Segments and Our Understanding of the Crime Problem*. Oxford: Oxford University Press, 2012. ISBN: 978-0-19-992863-7 978-0-19-536908-3.

[19] C. R. Shaw and H. D. McKay. *Juvenile Delinquency and Urban Areas: A Study of Rates of Delinquents in Relation to Differential Characteristics of Local Communities in American Cities*. Chicago: The University of Chicago Press, 1942.

[20] J. Povala, S. Virtanen, and M. Girolami. "Burglary in London: Insights from Statistical Heterogeneous Spatial Point Processes". In: *Journal of the Royal Statistical Society Series C: Applied Statistics* 69.5 (2020), pp. 1067–1090. ISSN: 0035-9254. DOI: 10.1111/rssc.12431.

[21] H. Luan, M. Quick, and J. Law. "Analyzing Local Spatio-Temporal Patterns of Police Calls-for-Service Using Bayesian Integrated Nested Laplace Approximation". In: *ISPRS International Journal of Geo-Information* 5.9 (2016), p. 162. DOI: 10.3390/ijgi5090162.

[22] D. Liu et al. "Understanding the Spatiotemporal Pattern of Crimes in Changchun, China: A Bayesian Modeling Approach". In: *Sustainability* 13.19 (2021), p. 10500. DOI: 10.3390/su131910500.

[23] M. Mahfoud et al. "Forecasting Spatio-Temporal Variation in Residential Burglary with the Integrated Laplace Approximation Framework: Effects of Crime Generators, Street Networks, and Prior Crimes". In: *Journal of Quantitative Criminology* 37.4 (2021), pp. 835–862. ISSN: 1573-7799. DOI: 10.1007/s10940-020-09469-3. (Visited on 2024-11-04).

[24] P. J. Diggle et al. "Spatial and Spatio-Temporal Log-Gaussian Cox Processes: Extending the Geostatistical Paradigm". In: *Statistical Science* 28.4 (2013). ISSN: 0883-4237. DOI: 10.1214/13-STS441.

[25] Á. Briz-Redón and J. Mateu. "A Mechanistic Bivariate Point Process Model for Crime Pattern Analysis". In: *Stat* 12.1 (2023), e537. ISSN: 2049-1573, 2049-1573. DOI: 10.1002/sta4.537.

[26] J. A. González et al. "A Doubly Stochastic Point Process Approach for Spatio-Temporal Dynamics of Crime Data". In: *Statistical Modelling* (2024). DOI: 10.1177/1471082X241264690.

[27] I. Escudero et al. "Crime Risk Assessment through Cox and Self-Exciting Spatio-Temporal Point Processes". In: *Stochastic Environmental Research and Risk Assessment* 39 (2024), pp. 181–203. DOI: 10.1007/s00477-024-02857-2.

[28] C. Zhao et al. "Research on Analysis Technology of Burglary Point Pattern Based on LGCP Model". In: *2019 Chinese Automation Congress (CAC)*. 2019 Chinese Automation Congress (CAC). Hangzhou, China: IEEE, 2019, pp. 5507–5511. DOI: 10.1109/CAC48633.2019.8997198.

[29] A. Rodrigues and P. J. Diggle. "Bayesian Estimation and Prediction for Inhomogeneous Spatiotemporal Log-Gaussian Cox Processes Using Low-Rank Models, With Application to Criminal Surveillance". In: *Journal of the American Statistical Association* 107.497 (2012), pp. 93–101. DOI: 10.1080/01621459.2011.644496.

[30] N. D'Angelo et al. "Self-Exciting Point Process Modelling of Crimes on Linear Networks". In: *Statistical Modelling* 24.2 (2024), pp. 139–168. DOI: 10.1177/1471082X221094146.

[31] J. Illian et al. "Using INLA To Fit A Complex Point Process Model With Temporally Varying Effects – A Case Study". In: *Journal of Environmental Statistics* 3 (2014), pp. 1–29.

[32] A. Fadlurohman, A. Choiruddin, and J. Mateu. "Inhomogeneous Log-Gaussian Cox Processes with Piecewise Constant Covariates: A Case Study in Modeling of COVID-19 Transmission Risk in East Java". In: *Stochastic Environmental Research and Risk Assessment* 38.7 (2024), pp. 2891–2901. ISSN: 1436-3259. DOI: 10.1007/s00477-024-02720-4.

[33] F. L. Bayisa et al. "Large-Scale Modelling and Forecasting of Ambulance Calls in Northern Sweden Using Spatio-Temporal Log-Gaussian Cox Processes". In: *Spatial Statistics* 39 (2020), p. 100471. DOI: 10.1016/j.spasta.2020.100471.

[34] N. D'Angelo et al. "Local Spatial Log-Gaussian Cox Processes for Seismic Data". In: *ASta Advances in Statistical Analysis* 106.4 (2022), pp. 633–671. DOI: 10.1007/s10182-022-00444-w.

[35] J. Møller, A. R. Syversveen, and R. P. Waagepetersen. "Log Gaussian Cox Processes". In: *Scandinavian Journal of Statistics* 25.3 (1998), pp. 451–482. DOI: 10.1111/1467-9469.00115.

[36] J. Møller and J. G. Rasmussen. "Cox Processes Driven by Transformed Gaussian Processes on Linear Networks— A Review and New Contributions". In: *Scandinavian Journal of Statistics* 51.3 (2024-09), pp. 1288–1322. ISSN: 0303-6898, 1467-9469. DOI: 10.1111/sjos.12720.

[37] Y. T. Abebe, A. M. Seid, and L. Roininen. *Log-Gaussian Cox Processes for Spatiotemporal Traffic Fatality Estimation in Addis Ababa*. 2024. arXiv: 2408.02612.

[38] B. M. Taylor and P. J. Diggle. *INLA or MCMC? A Tutorial and Comparative Evaluation for Spatial Prediction in Log-Gaussian Cox Processes*. 2012. URL: http://arxiv.org/abs/1202.1738 (visited on 2024-11-13).

[39] B. M. Taylor et al. "**Lgcp** : An *R* Package for Inference with Spatial and Spatio-Temporal Log-Gaussian Cox Processes". In: *Journal of Statistical Software* 52.4 (2013). DOI: 10.18637/jss.v052.i04.

[40] W. R. Gilks, S. Richardson, and D. J. Spiegelhalter. *Markov Chain Monte Carlo in Practice*. London: Chapman & Hall, 1996. ISBN: 978-0-429-17023-2.

[41] H. Rue, S. Martino, and N. Chopin. "Approximate Bayesian Inference for Latent Gaussian Models by Using Integrated Nested Laplace Approximations". In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 71.2 (2009), pp. 319–392. DOI: 10.1111/j.1467-9868.2008.00700.x.

[42] T. G. Martins et al. "Bayesian Computing with INLA: New Features". In: *Computational Statistics & Data Analysis* 67 (2013), pp. 68–83. DOI: 10.1016/j.csda.2013.04.014.

[43] F. Lindgren, H. Rue, and J. Lindström. "An Explicit Link between Gaussian Fields and Gaussian Markov Random Fields: The Stochastic Partial Differential Equation Approach". In: *Journal of the Royal Statistical Society Series B: Statistical Methodology* 73.4 (2011), pp. 423–498. DOI: 10.1111/j.1467-9868.2011.00777.x.

[44] F. E. Bachl et al. "inlabru: an R package for Bayesian spatial modelling from ecological survey data". In: *Methods in Ecology and Evolution* 10.6 (2019), pp. 760–766. DOI: 10.1111/2041-210X.13168.

[45] D. Simpson et al. "Going off Grid: Computationally Efficient Inference for Log-Gaussian Cox Processes". In: *Biometrika* 103.1 (2016), pp. 49–70. DOI: 10.1093/biomet/asv064.

[46] T. Opitz. *Latent Gaussian Modeling and INLA: A Review with Focus on Space-Time Applications*. 2017. arXiv: 1708.02723.

[47] V. Gómez-Rubio and H. Rue. "Markov Chain Monte Carlo with the Integrated Nested Laplace Approximation". In: *Statistics and Computing* 28.5 (2018), pp. 1033–1051. DOI: 10.1007/s11222-017-9778-y.

[48] M. Teng, F. Nathoo, and T. D. Johnson. "Bayesian Computation for Log-Gaussian Cox Processes: A Comparative Analysis of Methods". In: *Journal of Statistical Computation and Simulation* 87.11 (2017), pp. 2227–2252. DOI: 10.1080/00949655.2017.1326117.

[49] E. Krainski et al. *Advanced Spatial Modeling with Stochastic Partial Differential Equations Using R and INLA*. Chapman and Hall/CRC, 2018. ISBN: 978-0-429-62985-3. DOI: 10.1201/9780429031892.

[50] M. Blangiardo and M. Cameletti. *Spatial and Spatio-Temporal Bayesian Models with R-INLA*. John Wiley & Sons, Ltd, 2015. ISBN: 978-1-118-95020-3.

[51] Sistema de Segurança Interna, Gabinete do Secretário-Geral. *Relatório Anual de Segurança Interna 2023*. Portugal: Ministério da Administração Interna, 2024. URL: https://www.portugal.gov.pt/pt/gc24/comunicacao/documento?i=relatorio-anual-de-seguranca-interna-2023 (visited on 2024-01-18).

[52] M. Saraiva et al. "Perfis territoriais de criminalidade em Portugal (2009-2019)". In: *Finisterra* 56.116 (2021). DOI: 10.18055/FINIS20682.

[53] Sistema de Segurança Interna, Gabinete do Secretário-Geral. *Relatório Anual de Segurança Interna 2022*. Portugal: Ministério da Administração Interna, 2023. URL: https://www.portugal.gov.pt/pt/gc23/comunicacao/documento?i=relatorio-anual-de-seguranca-interna-2022 (visited on 2024-01-18).

[54] J. P. Tavares and A. C. Costa. "Spatial Modeling and Analysis of the Determinants of Property Crime in Portugal". In: *ISPRS International Journal of Geo-Information* 10.11 (2021), p. 731. DOI: 10.3390/ijgi10110731.

[55] M. Saraiva et al. "Crime Prediction and Monitoring in Porto, Portugal, Using Machine Learning, Spatial and Text Analytics". In: *ISPRS International Journal of Geo-Information* 11.7 (2022), p. 400. DOI: 10.3390/ijgi11070400.

[56] M. Saraiva and B. Teixeira. "Exploring the Spatial Relationship between Street Crime Events and the Distribution of Urban Greenspace: The Case of Porto, Portugal". In: *ISPRS International Journal of Geo-Information* 12.12 (2023), p. 492. DOI: 10.3390/ijgi12120492.

[57] Tavares, Joana Paulo. "Modelação e Análise Espacial das Condicionantes do Crime Contra o Património em Portugal Continental". Master's thesis. Campolide: NOVA Information Management School, 2021.

[58] P. Moraga. *Spatial Statistics for Data Science: Theory and Practice with R*. Data Science Series. Chapman and Hall/CRC, 2023. ISBN: 978-1-032-63351-0.

[59] P. J. Diggle. *Statistical Analysis of Spatial and Spatio-temporal Point Patterns*. 3rd ed. Monographs on Statistics and Applied Probability. Chapman and Hall/CRC, 2013. ISBN: 978-1-4665-6023-9.

[60] M. L. Carvalho and I. C. Natário. *Análise de Dados Espaciais*. Sociedade Portuguesa de Estatística, 2008.

[61] S. Abousamra et al. "Topology-Guided Multi-Class Cell Context Generation for Digital Pathology". In: (2023). DOI: 10.48550/arXiv.2304.02255. arXiv: 2304.02255.

[62] Direção-Geral do Território. *PT-TM06/ETRS89*. https://www.dgterritorio.gov.pt/geodesia/sistemas-referencia/portugal-continental/PT-TM06-ETRS89. Accessed: 2025-06-28. 2025.

[63] E. Pebesma. "Simple Features for R: Standardized Support for Spatial Vector Data". In: *The R Journal* 10.1 (2018), pp. 439–446. DOI: 10.32614/RJ-2018-009. URL: https://doi.org/10.32614/RJ-2018-009.

[64] E. Pebesma and R. Bivand. *Spatial Data Science: With applications in R*. Chapman and Hall/CRC, 2023. DOI: 10.1201/9780429459016. URL: https://r-spatial.org/book/.

[65] Neves, Ana Verónica Cabral. "O crime e o contexto: dois estudos de caso em Lisboa". PhD thesis. Faculdade de Ciências Sociais e Humanas da Universidade NOVA de Lisboa, 2019.

[66] L. Chen, M. Jun, and S. J. Cook. *Addressing Duplicated Data in Point Process Models*. 2024. arXiv: 2405.15192.

[67] C. Spychala. "Statistical analysis of road accidents in the region Franche-Comté : risk factors for accident injuries and spatial modelling for accident occurrences". English. NNT: 2022UBFCD064, tel-04323459. PhD Thesis. Université Bourgogne Franche-Comté, 2022.

[68] P. Diggle. "A Kernel Method for Smoothing Point Process Data". In: *Applied Statistics* 34.2 (1985), p. 138. ISSN: 00359254. DOI: 10.2307/2347366.

[69] J. A. González and P. Moraga. "Non-Parametric Analysis of Spatial and Spatio-Temporal Point Patterns". In: *The R Journal* 15.1 (2023), pp. 65–82. DOI: 10.32614/RJ-2023-025.

[70] G. James et al. *An Introduction to Statistical Learning: With Applications in R*. Springer Texts in Statistics. New York, NY: Springer US, 2021. ISBN: 978-1-0716-1417-4 978-1-0716-1418-1. DOI: 10.1007/978-1-0716-1418-1. URL: https://link.springer.com/10.1007/978-1-0716-1418-1 (visited on 2025-04-25).

[71] H. Akaike. "A New Look at the Statistical Model Identification". In: *IEEE Transactions on Automatic Control* 19.6 (1974), pp. 716–723. ISSN: 0018-9286. DOI: 10.1109/TAC.1974.1100705. (Visited on 2025-09-21).

[72] D. Simpson et al. "Penalising Model Component Complexity: A Principled, Practical Approach to Constructing Priors". In: *Statistical Science* 32.1 (2017). DOI: 10.1214/16-STS576.

[73] V. Gómez-Rubio. *Bayesian Inference with INLA*. Boca Raton, FL: Chapman & Hall/CRC Press, 2020. URL: https://becarioprecario.bitbucket.io/inla-gitbook/ (visited on 2024-11-21).

[74] H. Rue and L. Held. *Gaussian Markov Random Fields: Theory and Applications*. Monographs on Statistics and Applied Probability 104. Boca Raton: Chapman & Hall/CRC, 2005. ISBN: 978-1-58488-432-3.

[75] K. Flagg and A. Hoegh. "The Integrated Nested Laplace Approximation Applied to Spatial Log-Gaussian Cox Process Models". In: *Journal of Applied Statistics* 50.5 (2023-04-04), pp. 1128–1151. ISSN: 0266-4763, 1360-0532. DOI: 10.1080/02664763.2021.2023116.

[76] G.-A. Fuglstad et al. *Constructing Priors That Penalize the Complexity of Gaussian Random Fields*. 2017. arXiv: 1503.00256.

[77] A. Gelman, J. Hwang, and A. Vehtari. "Understanding Predictive Information Criteria for Bayesian Models". In: *Statistics and Computing* 24.6 (2014-11), pp. 997–1016. ISSN: 0960-3174, 1573-1375. DOI: 10.1007/s11222-013-9416-2.

[78] Instituto Nacional de Estatística. *Censos 2021. XVI Recenseamento Geral da População. VI Recenseamento Geral da Habitação: Resultados definitivos*. Lisboa: INE, 2022. ISBN: 978-989-25-0619-7. URL: https://www.ine.pt/xurl/pub/65586079.

[79] Instituto Nacional de Estatística. *Mapas INE – Download de Informação Geográfica, Censos 2021: Seções Estatísticas*. Accessed on: 2025-09-22. 2021. URL: %7Bhttps://mapas.ine.pt/download/index2021Seccoes.phtml%7D.

Appendix A

Code blocks

A.1 Data pre-processing

```
#>------------------------------------------------------------------------------
#> 0. Libraries
#>------------------------------------------------------------------------------

library(readxl)
library(dplyr)
library(sf)
library(lubridate)
library(ggplot2)

#>------------------------------------------------------------------------------
#> 1. Creating the study window
#>------------------------------------------------------------------------------

freg_sf <- st_read("Data/Raw/freguesias-shapefile/freguesias.shp")
target_freg <- c("União das freguesias de Caparica e Trafaria", "Costa da
    Caparica", "União das freguesias de Charneca de Caparica e Sobreda")
almada_freg_sf <- freg_sf %>%
   filter(Concelho == "ALMADA" & Freguesia %in% target_freg)
window_sf <- st_union(almada_freg_sf)
#> Saving the study window into an R object file
saveRDS(window_sf, "Data/Processed/window_sf.rds")

#>------------------------------------------------------------------------------
#> 2. Data cleansing
#>------------------------------------------------------------------------------

#> Import files
crimes2022 <- read_excel("E:/Data/241205_INF_E158582_Criminalidade DTer Almada
    e VN Gaia.xlsx", sheet = "Almada 2022")
```

```
crimes2023 <- read_excel("E:/Data/241205_INF_E158582_Criminalidade DTer Almada
    e VN Gaia.xlsx", sheet = "Almada 2023")
crimes <- bind_rows(crimes2022, crimes2023)
crimetypes <- read_excel("Data/Raw/Tipos de Crime.xlsx")

#> Fix the value of row 4506, column "crimes activação (Processo Crime)"
crimes$`Data activação (Processo Crime)`[4506] <- "2022-12-31 12:02"

#> Selecting only crimes handled by GNR and with lat/lon filled in
crimes <- crimes %>%
  filter(`ENTIDADE NOTADORA` == "GNR") %>%
  filter(!is.na(`LONGITUDE`), !is.na(`LATITUDE`)) %>%
  # Selecting only 5 columns from the original 52
  dplyr::select(timestamp = `Data Ocorrência`,
         category = `Título Crime`,
         subcategory = `Tipificação do Crime`,
         longitude = `LONGITUDE`,
         latitude = `LATITUDE`)

#> Creating a crime_sf object
crimes$latitude <- as.numeric(crimes$latitude)
crimes$longitude <- as.numeric(crimes$longitude)
crimes_sf <- st_as_sf(crimes, coords = c("longitude", "latitude"), crs = 4326)
crimes_sf <- st_transform(crimes_sf, 3763)

#> Fix the timestamp type
crimes_sf$timestamp <- as.POSIXct(crimes_sf$timestamp, format = "%Y-%m-%d
    \%H:%M")

#> Checking which subcategories are missing and if any is not correct
all_types <- crimetypes$`Tipificação do Crime`
unique_types <- unique(crimes_sf$subcategory)
existing_types <- all_types[all_types %in% unique_types]
misidentified_types <- setdiff(unique_types, all_types)
missing_types <- all_types[!all_types %in% unique_types]
#> Fix misidentified crime types
crimes_sf <- crimes_sf %>%
  mutate(subcategory = recode(subcategory,
                              "Condução de veículo com taxa de álcool
                                  igual/superior a 1,2g/l" = "Condução de
                                  veículo com taxa de álcool igual/superior a
                                  1,2g/l e/ou sob influência de substâncias
                                  psicotrópicas, estupefacientes ou produtos
                                  análogos",
                              " OUTROS CRIMES" = "Outros crimes"))
```

```
#> Translating categories and subcategories to English.
#> The translation_dict_category and translation_dict_subcategory are
#> ommited here.
#> crimes_sf <- crimes_sf %>%
#>   mutate(category = recode(category, !!!translation_dict_category),
#>          subcategory = recode(subcategory, !!!
    translation_dict_subcategory))

#>------------------------------------------------------------------------
#> 3. Selecting only street crimes within the study domain
#>------------------------------------------------------------------------

#> Select only crimes that occurred between 2022 and 2023
crimes_sf <- crimes_sf %>%
  filter(format(timestamp, "%Y") %in% c("2022", "2023"))

#> Select only crimes within the study window
crimes_sf <- st_filter(crimes_sf, window_sf, .predicate = st_within)

#> Select only street crimes
dict_street_crimes <- c("Pickpocketing",
                        "Theft of motor vehicle",
                        "Theft in motor vehicle",
                        "Wallet theft",
                        "Theft in a supermarket",
                        "Robbery in the street",
                        "Bank robbery of another financial establishment",
                        "Treasury or post office robbery",
                        "Service station robbery",
                        "Robbery in public transports",
                        "Damage to cultural property",
                        "Fire or arson in buildings, constructions, or means
                            of transport",
                        "Other damage")

crimes_sf <- crimes_sf %>%
  filter(subcategory %in% dict_street_crimes)

#>------------------------------------------------------------------------
#> 4. Handling duplicated points
#>------------------------------------------------------------------------

#> Reproducible results
set.seed(1)

#> Extract coordinates from the sf object
```

```
coo <- st_coordinates(crimes_sf)

#> Identify duplicated coordinates
duplicates <- duplicated(coo) | duplicated(coo, fromLast = TRUE)

#> Count the number of duplicated points
n_duplicates <- sum(duplicates)

eps <- 0.5  # meters
coo_jittered <- coo
coo_jittered[duplicates, 1] <- coo[duplicates, 1] + runif(sum(duplicates),
    -eps, eps)
coo_jittered[duplicates, 2] <- coo[duplicates, 2] + runif(sum(duplicates),
    -eps, eps)

crimes_sf <- crimes_sf %>%
  st_drop_geometry() %>%  # Drop existing geometry
  mutate(longitude = coo_jittered[, 1],
         latitude = coo_jittered[, 2]) %>%
  st_as_sf(coords = c("longitude", "latitude"), crs = 3763)  # Rebuild
      geometry

#> Saving the final crimes_sf dataframe into an R object file
saveRDS(crimes_sf, "E:/Data/crimes_sf.rds")
```

A.2 Non-parametric analysis

```
#>-----------------------------------------------------------------------------
#> 0. Libraries
#>-----------------------------------------------------------------------------

library(spatstat)
library(ggplot2)
library(plot3D)
library(dplyr)
library(sf)
library(viridis)
library(kernstadapt)
library(foreach)
library(doParallel)
library(GET)
library(KernSmooth)
library(stpp)
```

```
#>------------------------------------------------------------------------
#> 1. Load files
#>------------------------------------------------------------------------

crimes_sf <- readRDS("E:/Data/crimes_sf.rds")
window_sf <- readRDS("Data/Processed/window_sf.rds")

#>------------------------------------------------------------------------
#> 2. Create spatstat point patter objects
#>------------------------------------------------------------------------

#> create a spatstat window object
owin_crime <- as.owin(window_sf)
#> create a crime point pattern spatstat object  (for using spatstat tools)
crimes_ppp <- as.ppp(st_coordinates(crimes_sf), owin_crime)

#>------------------------------------------------------------------------
#> 3. Quadrat test
#>------------------------------------------------------------------------

#> color palette
col_scale <- viridis(256, option = "viridis", direction = -1)

#> Quadrat counts
qc3 <- quadratcount(crimes_ppp, nx = 3, ny = 3)
qc6 <- quadratcount(crimes_ppp, nx = 6, ny = 6)
qc9 <- quadratcount(crimes_ppp, nx = 9, ny = 9)

#> Raw quadrat counts
par(mfrow = c(1,3)) # make 3 plots in one row
plot(qc3, main = "3x3", cex = 0.8)
plot(qc6, main = "6x6", cex = 0.8)
plot(qc9, main = "9x9", cex = 0.8)

#> Plot quadrat counts using a color scale
plot(intensity(qc3, image = TRUE), main = "3x3", box = FALSE, col = col_scale)
plot(intensity(qc6, image = TRUE), main = "6x6", box = FALSE, col = col_scale)
plot(intensity(qc9, image = TRUE), main = "9x9", box = FALSE, col = col_scale)
par(mfrow = c(1,1)) # back to 1 plot only

#> Quadrat test
quadrat.test(crimes_ppp, nx = 6, ny = 6)

#>------------------------------------------------------------------------
#> 4. Kernel estimation of the intensity function
#>------------------------------------------------------------------------
```

```
#> KDE with default options
plot(density.ppp(crimes_ppp), col=col_scale, main = "", box = FALSE)
#> Plot point pattern on top
plot(crimes_ppp, main="", pch=20, cex=0.8, add=TRUE)

#> KDE with Scott's bandwidth + improved Diggle edge correction
dens_scott <- density.ppp(crimes_ppp, sigma=bw.scott, edge=T, diggle=T)
#> KDE with Diggle's bandwidth + improved Diggle edge correction
dens_diggle <- density.ppp(crimes_ppp, sigma=bw.diggle, edge=T, diggle=T)
#> KDE with ppl bandwidth + improved Diggle edge correction
dens_ppl <- density.ppp(crimes_ppp, sigma=bw.ppl, edge=T, diggle=T)

par(ps = 18)
par(mfrow = c(1,3))
#> KDE plots
plot(dens_scott, main = "Scott bandwidth", col = col_scale, box = FALSE)
plot(dens_ppl, main = "ppl bandwidth", col = col_scale, box = FALSE)
plot(dens_diggle, main = "Diggle bandwidth", col = col_scale, box = FALSE)

#> Perspective plots for each KDE
persp(dens_scott, main="Scott bandwidth",
      zlab = "Intensity estimated", xlab = "Easting", ylab = "Northing")
persp(dens_ppl, main="ppl bandwidth",
      zlab = "Intensity estimated", xlab = "Easting", ylab = "Northing")
persp(dens_diggle, main="Diggle bandwidth",
      zlab = "Intensity estimated", xlab = "Easting", ylab = "Northing")

#> Adaptive bandwidth (varying bandwidth)
par(mfrow = c(1,1))
IntensityEst <- adaptive.density(crimes_ppp, method ="kernel", edge=T,
    diggle=T)
summary(IntensityEst)
plot(IntensityEst, auto.axes = F, col = col_scale,
     main = "Adaptive kernel intensity estimate", box = F)

#> Resacle values to pixel area
pixel_area <- 71.3 * 111.616   # 7958.1248 m^2
#> Rescale intensity to events per pixel
Intensity_per_pixel <- eval.im(IntensityEst * pixel_area)
#> Summary
summary(Intensity_per_pixel)
#> Plot with rescaled intensity
plot(Intensity_per_pixel,
     main = "Adaptive kernel intensity estimate",
     sub = "Events per pixel",
```

```
     box = FALSE, col = col_scale, ribargs = list(las = 1))
mtext("Estimated number of events per pixel",
      side = 3, line = 0.5, cex = 0.8)

#>-----------------------------------------------------------------------------
#> 5. Non-parametric K-function estimation
#>-----------------------------------------------------------------------------

#> 5.1 Estimate assuming a stationary (spatially homogeneous) point process
K_est <- Kest(crimes_ppp, correction = "iso")
#> 5.2 Estimate assuming an inhomogeneous point process
K_est_inhom <- Kinhom(crimes_ppp, correction="iso")

#> Extract relevant values
r_vals <- K_est$r
K_poisson <- K_est$theo  # theoretical Poisson K = pi * r^2
K_estim <- K_est$iso
K_estim_inhom <- K_est_inhom$iso

#> Plot K estimate and Poisson K-function line
ggplot() +
  # Plot lines
  geom_line(aes(x = r_vals, y = K_estim,
                color = "Est hom", linetype = "Est hom"), linewidth = 0.8) +
  geom_line(aes(x = r_vals, y = K_estim_inhom,
                color = "Est inhom", linetype = "Est inhom"), linewidth = 0.8)
                    +
  geom_line(aes(x = r_vals, y = K_poisson,
                color = "Poisson", linetype = "Poisson"), linewidth = 0.8) +
  # Axis and legend names
  labs(x = "r (meters)",
       y = expression(hat(K)(r)),
       color = NULL,
       linetype = NULL
  ) +
  # Graph theme and text size
  theme_grey(base_size = 16) +
  # Legend
  scale_color_manual(values = c("Est hom" = "blue", "Est inhom" = "red",
      "Poisson" = "black")) +
  scale_linetype_manual(values = c("Est hom" = "solid", "Est inhom" = "solid",
      "Poisson" = "dashed")) +
  # Adjust legend details
  theme(
    # top-left corner
    legend.position = c(0.05, 0.95),
```

```
    # top-left corner
    legend.justification = c("left", "top"),
    # legend rectangle colors
    legend.background = element_rect(fill = "grey92", colour = "darkgrey"),
    # remove vertical spacing between items
    legend.spacing.y = unit(0, "pt"),
    # remove extra box spacing due to title missing
    legend.box.spacing = unit(0, "pt"),
    # tight padding inside box
    legend.margin = margin(2, 2, 2, 2)
  )

#>------------------------------------------------------------------------------
#> 6. Non-parametric pair correlation function estimation
#>------------------------------------------------------------------------------

#> Nearest-neighbor summary (descriptive statistics)
nn <- nndist(crimes_ppp)
summary(nn)
hist(nn, breaks=50, main="Nearest-neighbour distances", xlab="distance (m)")

#> Estimate the pair correlation function (constant intensity case)
pcf_est <- pcf(crimes_ppp, correction = "iso")

#> Extract relevant values
r_vals <- pcf_est$r
pcf_estim <- pcf_est$iso
pcf_poisson <- pcf_est$theo

#> Estimate the pair-correlation function (inhomogeneous case)
#> We starting following González & Moraga (2024) here
X  <- crimes_ppp
sigmaD <- bw.scott(X)

#> Intensity map (for plotting)
MD <- density.ppp(X,
                  sigma = sigmaD,
                  #> Diggle's edge correction
                  diggle = T,
                  #> Force estimated values to be >= 0
                  positive = T)

#> Intensity estimated at observed points (bw.pcf and pcfinhom need this)
MP <- density.ppp(X,
                  sigma = sigmaD,
                  # Diggle's edge correction
```

```
                  diggle = T,
                  # Returns a numeric vector of intensity estimates ONLY at
                      the
                  # locations of observed points (default is at = "pixels"
                      and
                  # returns a pixel image that can be plotted)
                  at = "points",
                  # Force estimated values to be >= 0
                  positive = T)

#> Bandwith estimation
bwG <- bw.pcf(X, cv.method = "compLik", divisor = "d", lambda = MP)

#> Distance (r values)
r0 <- 0.7 * rmax.rule("K", X$window, intensity(X))
rr <- seq(0, r0, length.out = 71)

#> Esimate and plot the inhomogeneous case pair-correlation function
pcf_inhom <- pcfinhom(X, lambda = MP, bw = bwG, rmax = r0,
                      correction = "isotropic")
pcf_inhom_estim <- pcf_inhom$iso

ggplot() +
  geom_line(aes(x = r_vals, y = pcf_estim,
                color = "Est hom", linetype = "Est hom"), linewidth = 0.8) +
  geom_line(aes(x = r_vals, y = pcf_poisson,
                color = "Poisson", linetype = "Poisson"), linewidth = 0.8) +
  geom_line(aes(x = r_vals, y = pcf_inhom_estim,
                color = "Est inhom", linetype = "Est inhom"), linewidth = 0.8)
                    +
  coord_cartesian(xlim = c(0, 1500), ylim = c(0, 50)) +
  # Axis and legend names
  labs(x = "r (meters)",
       y = expression(hat(g)(r)),
       #title = "Pair-correlation function",
       color = NULL,
       linetype = NULL
  ) +
  # Graph theme and text size
  theme_grey(base_size = 16) +
  # Legend
  scale_color_manual(values = c("Est hom" = "blue", "Est inhom" = "red",
      "Poisson" = "black")) +
  scale_linetype_manual(values = c("Est hom" = "solid", "Est inhom" = "solid",
      "Poisson" = "dashed")) +
  # Adjust legend details
```

```
  theme(
    # top-left corner
    legend.position = c(0.95, 0.95),
    # top-left corner
    legend.justification = c("right", "top"),
    # legend rectangle colors
    legend.background = element_rect(fill = "grey92", colour = "darkgrey"),
    # remove vertical spacing between items
    legend.spacing.y = unit(0, "pt"),
    # remove extra box spacing due to title missing
    legend.box.spacing = unit(0, "pt"),
    # tight padding inside box
    legend.margin = margin(2, 2, 2, 2)
  )

#>------------------------------------------------------------------------------
#> 7. Visualizing the estimate of intensity per quarter
#> (an analugous code was used for weekdays and parts of the day)
#>------------------------------------------------------------------------------

crimes_sf <- crimes_sf %>%
  mutate(
    date = as.Date(timestamp),
    mark = as.integer(difftime(date, min(date), units = "days")) + 1
  )

#> Extract coordinates matrix
coords <- sf::st_coordinates(crimes_sf)
#> Combine into a data.frame with marks
crimes <- data.frame(
  x = coords[, "X"],
  y = coords[, "Y"],
  marks = crimes_sf$mark
)

#> Use the true dates
timelabels <- crimes_sf$date

#> Bandwidth for KDE
X <- as.ppp(st_coordinates(crimes_sf), owin_crime)
sigmaD <- bw.ppl(X)  # or set a fixed bandwidth value

par(mfrow = c(2, 4), mar = c(3, 3, 5, 1))  # bigger top margin

#> Bandwidth for KDE (can fix or compute)
sigmaD <- bw.ppl(X)  # or set manually
```

```
#> Function to subset points by date and plot KDE
plot_quarter_kde <- function(start_date, end_date, col_fill, main_title) {
  sel <- timelabels >= start_date & timelabels <= end_date
  X_sub <- X[sel]

  if (X_sub$n == 0) {
    plot.new()
    title(main = paste(main_title, "\nNo points"), cex.main = 0.9)
    return()
  }

  #> KDE with fixed bandwidth and Diggle correction
  kde <- density.ppp(X_sub, sigma = sigmaD, diggle = TRUE, positive = TRUE)

  #> Plot without title, then add manually
  plot(kde, main = "", col = col_scale, riblength = 0, legend = FALSE)
  title(main = main_title, cex.main = 1.5)  # force title to appear
}

#> Quarters 2022
quarters_2022 <- list(c(as.Date("2022-01-01"), as.Date("2022-03-31")),
c(as.Date("2022-04-01"), as.Date("2022-06-30")),
c(as.Date("2022-07-01"), as.Date("2022-09-30")),
c(as.Date("2022-10-01"), as.Date("2022-12-31")))
#> Quarters 2023
quarters_2023 <- list(c(as.Date("2023-01-01"), as.Date("2023-03-31")),
c(as.Date("2023-04-01"), as.Date("2023-06-30")),
c(as.Date("2023-07-01"), as.Date("2023-09-30")),
c(as.Date("2023-10-01"), as.Date("2023-12-31")))

#> Plot KDE for each 2022 quarter
for (i in seq_along(quarters_2022)) {
  start_date <- quarters_2022[[i]][1]
  end_date <- quarters_2022[[i]][2]
  main_title <- paste0("2022 Q", i, "\n", format(start_date, "%b %d"), "-",
                       format(end_date, "%b %d"))
  plot_quarter_kde(start_date, end_date, col_fill = "blue", main_title)
}

#> Plot KDE for each 2023 quarter
for (i in seq_along(quarters_2023)) {
  start_date <- quarters_2023[[i]][1]
  end_date <- quarters_2023[[i]][2]
  main_title <- paste0("2023 Q", i, "\n", format(start_date, "%b %d"), "-",
                       format(end_date, "%b %d"))
```

```
  plot_quarter_kde(start_date, end_date, col_fill = "red", main_title)

#>-----------------------------------------------------------------------------
#> 8. Non-parametric L-function estimation (González and Moraga, 2023)
#>-----------------------------------------------------------------------------

#> L-function estimate with envelopes computed through Monte Carlo simulation
L1 <- envelope(X, nsim = 39, savefuns = TRUE, fun = "Linhom", diggle = T,
               transform = expression(.-r), sigma = sigmaD, r = rr,
               simulate = expression(rpoispp(lambda = MD)), verbose = F)

#> Because we simulated a priori the intensity function, we have a composite
#> hypothesis, and so we need an extra set of simulations because we are
#> performing a two-stage test
Simpatterns <- rpoispp(lambda = MD, nsim = 39)

simL <- function(rep) {
  sim_fit <- density.ppp(Simpatterns[[rep]], diggle = T,
                         sigma = sigmaD, positive = T)
  envelope(Simpatterns[[rep]], nsim = 39, savefuns = T, fun = "Linhom",
           transform = expression(.-r), r = rr, diggle = T, sigma = sigmaD,
           simulate = expression(rpoispp(lambda = sim_fit)))
}

#> Paralell computation because this is a very heavy computation otherwise
c0 <- parallel::makeCluster(detectCores() - 1)
doParallel::registerDoParallel(c0)
L.ls <- foreach(i = 1:39, .packages = c("spatstat")) %dopar% {simL(i)}
parallel::stopCluster(c0)

resL <- GET.composite(X = L1, X.ls = L.ls, type = "erl",
                      alternative = "greater", savefuns = T)
plot(resL)

#>-----------------------------------------------------------------------------
#> 9. Spatio-temporal estimation of K and g (González and Moraga, 2023)
#>-----------------------------------------------------------------------------

#> Rename "marks" to "t"
crimes$t <- crimes$marks

#> Convert to ppp object
xrange <- range(crimes$x)
yrange <- range(crimes$y)
window <- owin(xrange, yrange)
crimes_ppp <- ppp(x = crimes$x,
```

```
                  y = crimes$y,
                  window = window,
                  marks = crimes$t)

#> Spatio-temporal separability test
SepTest <- separability.test(crimes_ppp, nx = 5, ny = 4, nt = 16, nperm =
    50000)
SepTest

Times <- crimes$marks
bwt <- bw.nrd0(Times)
edgewt.t <- pnorm((max(Times) - Times) / bwt) - pnorm((min(Times)- Times) /
    bwt)

nonseparable <- function(time){
  contrib <- (Times >= time - 3 * bwt) & (Times <= time + 3 * bwt)
  Wh <- dnorm(x = Times[contrib], mean = time, sd = bwt) / edgewt.t[contrib]
  density.ppp(X[contrib], weights = Wh, diggle = TRUE)
}
nonsep <- lapply(unique(Times), nonseparable)

n.slices <- which(unique(timelabels) %in% as.Date(c("2022-01-01",
    "2022-07-01",
                                                    "2023-01-01",
                                                        "2023-07-01")))
Snap <- list(nonsep[[n.slices[1]]], nonsep[[n.slices[2]]],
             nonsep[[n.slices[3]]], nonsep[[n.slices[4]]])

plot.imlist(Snap, equal.ribbon = T, ncols = 4, box = F, main = "", log = F,
            main.panel = unique(timelabels)[n.slices], col = viridis(1200),
            ribscale = 100, mar.panel=c(0, 0, 1, 1), panel.end = X$window)

#> Create marked point pattern
crimes_ppp <- spatstat.geom::ppp(
  x = coords[,1],
  y = coords[,2],
  window = owin_crime,
  marks = crimes_sf$mark
)

lambda <- NULL
for (i in 1:length(nonsep)){
  lambda <- c(lambda, safelookup(nonsep[[i]],
                                 crimes_ppp[marks(crimes_ppp) ==
                                     unique(Times)[i]]))
}
```

```
PP <- X %mark% data.frame(time = crimes_ppp$marks, Lambda = lambda)

dt <- dist(unique(Times))
ht <- dpik(dt, kernel = "epanech", gridsize = 50, scalest = "iqr")
ht <- 2 * ht
t0 <- 0.15 * max(dt)

FMD <- as.3dpoints(PP$x, PP$y, PP$marks$time)
s.region <- as.matrix(data.frame(x = PP$window$bdry[[1]]$x,
                                 y = PP$window$bdry[[1]]$y))

hs <- bwG
u1 <- seq(hs, r0, length.out = 71)[-1]
v1 <- seq(ht, t0, length.out = 71)[-1]

#> ! Heavy computation !
g <- PCFhat(xyt = FMD, s.region = s.region, t.region = range(Times),
            lambda = lambda, dist = u1, times = v1, ks = "epanech",
            kt = "epanech", hs = hs, ht = ht)

par(mar = c(3,3,3,3))
persp3D(x = u1, y = v1, z = g$pcf, facets = NA, curtain = F, col =
    viridis(200),
        colkey = F, bty = "g", pch = 20, cex = 1.5, theta = 130, phi = 5,
        border = NA, ticktype = "detailed", cex.axis = 0.5, zlab = "",
        xlab = "spatial distances", ylab = "temporal distances")

u <- seq(0, r0, length.out = 71)
v <- seq(0, t0, length.out = 71)

#> ! Heavy computation !
stik <- STIKhat(xyt = FMD, s.region = s.region, t.region = range(Times),
                lambda = lambda, dist = u, times = v, infectious = F)
stik <- readRDS("outputs/k_function_temporal.rds")

KS <- stik$Khat - stik$Ktheo

par(mar = c(3,3,3,3))
persp3D(x = u[-1], y = v[-1], z = KS, facets = NA, zlab = "",
        curtain = F, col = viridis(100), colkey = F, bty = "g",
        pch = 20, cex = 1.5, theta = 40, phi = 5, border = NA,
        ticktype = "detailed", cex.axis = 0.5, xlab = "spatial distances",
        ylab = "temporal distances")
}
```

A.3 Automatic covariate selection

This code concerns an example of the automatic covariate selection procedure for the subset of Census covariates at leaf-level of hierarchy.

```
#>-----------------------------------------------------------------------------
#> 0. Libraries
#>-----------------------------------------------------------------------------

library(ggcorrplot)
library(sf)
library(dplyr)
library(terra)
library(spatstat)
library(ggplot2)
library(patchwork)
library(caret)
library(glmnet)
library(MASS)

#>-----------------------------------------------------------------------------
#> 1. Created functions for the automatic selection procedure
#>-----------------------------------------------------------------------------

#> Remove near-zero variance variables
nzv_filter <- function(data, verbose = TRUE) {
  if (verbose) cat("Removing near-zero variance covariates...\n")
  nzv <- nearZeroVar(st_drop_geometry(data), names = TRUE)

  if(length(nzv) > 0) {
    data <- dplyr::select(data, -all_of(nzv))
    if (verbose) {
      cat("Removed", length(nzv), "variables (near-zero variance):\n")
      print(nzv)
    }
  } else if (verbose) cat("No near-zero variance found.\n")

  return(data)
}

#> Remove high-VIF variables
vif_filter <- function(data, vif_threshold, verbose = TRUE) {
  if (verbose) cat("Removing variables with VIF >", vif_threshold,
      "stepwise...\n")

  start_time <- Sys.time()
```

```
  vif_filtered <- usdm::vifstep(st_drop_geometry(data), th = vif_threshold)
  end_time <- Sys.time()

  kept_vars <- vif_filtered@results$Variables
  removed_vars <- setdiff(colnames(st_drop_geometry(data)), kept_vars)

  if (verbose) {
    cat("Removed", length(removed_vars), "variables with high VIF.\n")
    print(vif_filtered@results)
    cat("Runtime of vifstep:", round(end_time - start_time, 2), "\n")
  }

  return(dplyr::select(data, all_of(kept_vars)))
}

#> Bootstrap selection the negative binomial
bootstrap_variable_selection <- function(data, full_model_formula,
    response_var,
                                         n_boot = 200, verbose = TRUE) {

  selected_vars <- list()   # store selected variables
  success_count <- 0        # count successful fits
  selection_freq <- NULL
  i <- 1                    # attempt counter

  set.seed(123)             # reproducibility

  if (verbose) cat("Bootstrap NB variable selection started, target n =",
      n_boot, "\n")

  while (success_count < n_boot) {
    if (verbose) cat("Bootstrap attempt (try):", i, "\n")

    # Sample with replacement
    boot_data <- data[sample(nrow(data), replace = TRUE), ]

    # Flag to indicate if iteration had warning
    iteration_failed <- FALSE

    # Catch warnings as errors
    step_model <- withCallingHandlers({

      ctrl <- glm.control(maxit = 100, epsilon = 1e-8)
      full_model <- glm.nb(full_model_formula, data = boot_data, link = log,
          control = ctrl)
      null_model <- glm.nb(as.formula(paste(response_var, "~ 1")), data =
          boot_data, link = log, control = ctrl)
```

```
      # Stepwise selection
      stepAIC(null_model, scope = list(lower = null_model, upper =
          full_model),
              direction = "both", trace = FALSE)

    }, warning = function(w) {
      iteration_failed <<- TRUE
      if (verbose) cat("Warning in bootstrap iteration", i, ":",
          conditionMessage(w), "\n")
      invokeRestart("muffleWarning")  # suppress printing the warning
          normally
    })

    # Check if iteration had warning
    if (!iteration_failed && !inherits(step_model, "try-error")) {
      selected_vars[[length(selected_vars) + 1]] <-
          names(coef(step_model))[-1]  # exclude intercept
      success_count <- success_count + 1
    } else {
      if (verbose && iteration_failed) cat("Skipping iteration", i, "due to
          warning.\n")
    }

    i <- i + 1
  }

  if (verbose) cat("Bootstrap completed. Successful iterations:",
      success_count, "\n")

  # Compute selection frequencies
  selection_freq <- sort(table(unlist(selected_vars)), decreasing = TRUE) /
      n_boot
  return(round(selection_freq, 2))
}

#>------------------------------------------------------------------------
#> 2. Load data
#>------------------------------------------------------------------------

#> Study window
window_sf <- readRDS("Data/Processed/almada_freg_sf.rds")
#> Crimes dataset
crimes_sf <- readRDS("Data/Processed/crimes_sf.rds")
#census 2021 public data by statistical section (ss)
census_ss_sf <- st_read("Data/Raw/SECCOES2021 data/C2021_SECCOES_1503.gpkg")
```

```
#census 2021 public data by BGRI, a sub level of statistical section (bgri)
census_bgri_sf <- st_read("Data/Raw/BGRI2021 data/BGRI2021_1503.gpkg")

#> Selecting relevant columns from raw data
census_ss_sf <- census_ss_sf[,10:186]

#> Selecting relevant variables
#>census_ss_sf_5 <- census_ss_sf %>%
#>  dplyr::select(N_EDIFICIOS_CLASSICOS_1OU2_ALOJ,
#>  ...
#>  N_INDIVIDUOS_RESID_FORA_PAIS_M)

#>------------------------------------------------------------------------------
#> 3. Automatic selection procedure
#>------------------------------------------------------------------------------

#> Data in use
census_sf <- census_ss_sf_5

#> Filter to study window
census_sf <- census_sf %>%
  filter(st_intersects(geom, window_sf, sparse = FALSE) %>% apply(1, any))
#Divide by section area to get densities
census_dens_sf <- census_sf %>%
  mutate(area_m2 = as.numeric(st_area(geom))) %>%
  transmute(across(where(is.numeric) & !matches("area_m2"),
                   ~ .x / area_m2, .names = "{.col}"), geom)
#> Standardize
census_norm_sf <- census_dens_sf %>%
  mutate(across(!all_of(attr(., "sf_column")),
                ~ (.x - mean(.x, na.rm = TRUE)) / sd(.x, na.rm = TRUE)))

#> Filter near-zero variance and highly correlation covariates
census_filt_sf <- census_norm_sf %>%
  nzv_filter() %>%
  vif_filter(vif_threshold = 5)

#> Create dataframes for model in bootstrap procedure
census_df <- st_drop_geometry(census_filt_sf)
counts_per_area <- lengths(st_intersects(census_filt_sf, crimes_sf))
#> Response: crime counts per section
census_df$counts <- counts_per_area
#> Covariates: census covariate density per section
X <- as.matrix(census_df)

#> Bootstrap
```

```
result <- bootstrap_variable_selection(data = census_df,
                                       full_model_formula = counts ~ .,
                                       response_var = "counts")
```

A.4 Spatial LGCP model with covariates

```
#>-----------------------------------------------------------------------------
#> 0. Libraries
#>-----------------------------------------------------------------------------

library(INLA)
library(terra)
library(raster)
library(ggplot2)
library(sf)
library(viridis)
library(lattice)
library(rasterVis)
library(FNN)
source("R/book_dual_mesh.R")
set.seed(1)

#>-----------------------------------------------------------------------------
#> 1. Load study window and crime events
#>-----------------------------------------------------------------------------

crimes_sf <- readRDS("Data/Processed/crimes_sf.rds")
window_sf <- readRDS("Data/Processed/almada_freg_sf.rds")
covs_sf <- readRDS("Data/Processed/census_socioeco_manual_norm_sf.rds")

#>-----------------------------------------------------------------------------
#> 2. Create prediction grid
#>-----------------------------------------------------------------------------

#> Grid raster object with 10 000 cells in the correct crs
grid <- terra::rast(terra::vect(window_sf),
                    nrows = 100, ncols = 100,
                    crs = "EPSG:3763")

#> Collect cell centres from each grid cell
xy <- xyFromCell(grid, 1:ncell(grid)) # vector

#> Transform cell centres in an sf object in the correct crs
dp <- st_as_sf(as.data.frame(xy), coords = c("x", "y"), crs =
    st_crs(window_sf))
```

```
#> Save indices of cell centres within the study window
#> This will be useful for plotting intensity and spatial random effect maps
indicespointswithin <- which(st_intersects(dp, window_sf, sparse = FALSE))

#> Keep only the cell centres within the study window
dp <- st_filter(dp, window_sf)
#> Prediction coordinates
coop <- st_coordinates(dp)

#> VISUAL CHECK: Plot prediction locations
ggplot() + geom_sf(data = window_sf) +
  geom_sf(data = dp, cex=0.5) + coord_sf(datum = "EPSG:3763")

#>------------------------------------------------------------------------
#> 3. Create mesh
#>------------------------------------------------------------------------

#> Point pattern coordinates
coo <- st_coordinates(crimes_sf)

#> Add a buffer to the borders of the study window (500 m)
window_buffered <- st_buffer(window_sf, dist=500)
#> Create a mesh
mesh <- inla.mesh.2d(
  boundary = window_buffered,
  max.edge = c(300, 2000),
  cutoff = 150)

#> VISUAL CHECK: plot the mesh and the point pattern
plot(mesh)
points(coo, col = "red", cex=0.3)
plot(window_sf, add=T)
axis(1)
axis(2)

#>------------------------------------------------------------------------
#> 4. Run SPDE model using PC-priors
#>------------------------------------------------------------------------

spde <- inla.spde2.pcmatern(
  mesh = mesh,
  prior.range = c(1000, 0.5),
  prior.sigma = c(1, 0.01)
)
```

```
#>------------------------------------------------------------------------
#> 5. Create dual mesh
#>------------------------------------------------------------------------

#> Using Krainski et al. (2019) function provided in Moraga (2024) section
#> 23.3.10 of the book
dmesh <- book_mesh_dual(mesh)

#> VISUAL CHECK: Plot the dual mesh
plot(dmesh)
axis(1)
axis(2)

#>------------------------------------------------------------------------
#> 6. Calculate weights
#>------------------------------------------------------------------------

#> Converting the dual mesh into an sf object
dmesh_sf <- st_as_sf(dmesh)
st_crs(dmesh_sf) <- st_crs(window_sf)

#> Calculate weights of the Poisson discretised distribution. These
    correspond
#> to the intersection between each dual mesh polygon and the study area.
#> Polygon outside the study area receive the value 0. This function is an
#> adaptation of Krainski et al. (2019) using the sf library instead of the
#> deprecated rgeos library
w <- sapply(1:nrow(dmesh_sf), function(i) {
  intersection <- st_intersection(dmesh_sf[i, ], window_sf)
  if (nrow(intersection) > 0) {
    as.numeric(st_area(intersection))  # convert units object to numeric
  } else {
    0
  }
})

#> CHECK: the sum of weights should equal the area of the study window
sum(w)
as.numeric(st_area(window_sf))

#> VISUAL CHECK: Plot of mesh with the nodes with positive weight, i.e.,
    inside
#> the study window (black) and with 0 weight, i.e., outside (red)
plot(mesh)
points(mesh$loc[which(w > 0), 1:2], col = "black", pch = 20)
points(mesh$loc[which(w == 0), 1:2], col = "red", pch = 20)
```

```
plot(st_geometry(window_sf), add = TRUE, col = NA, border = "green", lwd = 1)
axis(1)
axis(2)

#>-----------------------------------------------------------------------------
#> 7. Build a projection matrix
#>-----------------------------------------------------------------------------

#> Nr of mesh nodes
nv <- mesh$n
#> Nr of observed points
n <- nrow(coo)
#> Vector of size n+nv: [0 ... 0 (n times) | 1 ... 1 (nv times)]'
y.pp <- rep(0:1, c(nv, n))
#> Vector of size n+nv: [w ... w (n times) | 0 ... 0 (nv times)]'
e.pp <- c(w, rep(0, n))

#> Projection matrix for the integration points (mesh vertices).
A.int <- Diagonal(nv, rep(1, nv))
#> Projection matrix for observed points (event locations).
A.y <- inla.spde.make.A(mesh = mesh, loc = coo)
#> Projection matrix for mesh vertices and event locations
A.pp <- rbind(A.int, A.y)
#> Projection matrix for the prediction locations.
Ap.pp <- inla.spde.make.A(mesh = mesh, loc = coop)

#>-----------------------------------------------------------------------------
#> 8. Assign covariates to mesh and observed locations
#>-----------------------------------------------------------------------------

#> Get observed location coordinates
crimes_sf <- crimes_sf["geometry"]

#> Get mesh locations (vertices)
v <- mesh$loc
#> Convert them into an sf dataframe with the correct crs
v_sf <- st_as_sf(data.frame(x = v[,1], y = v[,2]), coords = c("x", "y"),
                 crs = 3763)
#> Find indices of vertices inside the study window and get their locations
v_in_index <- st_within(v_sf, window_sf, sparse = FALSE)
v_in_sf <- v_sf[v_in_index, ]

#> Assign covariates to each mesh location inside the study window
v_covs_in_sf <- st_join(v_in_sf, covs_sf, left = TRUE)
colSums(is.na(v_covs_in_sf)) # CHECK: if there are NA's
```

```
#> Create an sf data.frame with mesh nodes coordinates + covariates
    initialized
#> to 0
covs_name <- names(st_drop_geometry(covs_sf))
v_covs_sf <- v_sf
v_covs_sf[covs_name] <- 0

#> Assign the covariate values only to mesh nodes inside the study window
for (cov_name in covs_name) {
  v_covs_sf[[cov_name]][v_in_index] <- v_covs_in_sf[[cov_name]]
}

#> Convert the sf vertice locations with covariate values into a matrix
v_covs_mat <- as.matrix(st_drop_geometry(v_covs_sf))

#> Assign covariates to observed points
crimes_covs_sf <- st_join(crimes_sf, covs_sf, left = TRUE)
colSums(is.na(crimes_covs_sf)) # CHECK: if there are NA's
crimes_covs_mat <- as.matrix(st_drop_geometry(crimes_covs_sf))

#> Matrix with mesh + observed points covariates
covs_mat <- rbind(v_covs_mat, crimes_covs_mat)

#> Assign covariates to prediction points
pred_covs_sf <-  st_join(dp, covs_sf, left = TRUE)
colSums(is.na(pred_covs_sf)) # CHECK: if there are NA's
pred_covs_mat <- as.matrix(st_drop_geometry(pred_covs_sf))

#>-----------------------------------------------------------------------------
#> 9. Constructing the INLA stacks
#>-----------------------------------------------------------------------------

#> Stack for estimation
stk.e.pp <- inla.stack(
  #> A label for the estimation stack
  tag = "est.pp",
  #> Observed events and exposure term for the Poisson distribution
  data = list(y = y.pp, e = e.pp),
  #> List of projection matrices, one per effect group
  A = list(
    #> Maps the fixed effects group to all rows of the stack,
    1,
    #> SPDE projection matrix from mesh nodes to integration points and
    #> observation locations
    A.pp),
  #> List of fixed and random effects
```

```
  effects = list(
    #> Fixed effect: intercept (called b0) and covariates
    c(list(b0 = 1), as.list(as.data.frame(covs_mat))),
    #> Random spatial effect (called s)
    list(s = 1:nv)))

#> Stack for prediction
stk.p.pp <- inla.stack(
  #> A label for the prediction stack
  tag = "pred.pp",
  #> Prediction locations and exposure term for the Poisson distribution
  #> We assign NA to the prediction locations because no events are there. We
  #> also assign exposure value of 0 to these locations
  data = list(y = rep(NA, nrow(coop)), e = rep(0, nrow(coop))),
  #> List of projection matrices, one per effect group. Similar to the
  #> estimation stack but using the prediction SPDE projection matrix Ap.pp
  A = list(1, Ap.pp),
  #> List of fixed and random effects. Similar to the estimation stack but
      now
  #> our intercept should have as many rows as prediction locations.
  effects = list(
    c(list(b0 = 1), as.list(as.data.frame(pred_covs_mat))),
    list(s = 1:nv)))

#> Full stack: combining both estimation and prediction stacks
stk.full.pp <- inla.stack(stk.e.pp, stk.p.pp)

#>------------------------------------------------------------------------
#> 10. Model formulation
#>------------------------------------------------------------------------

formula_str <- paste("y ~ 0 + b0 +",
                     paste(covs_name, collapse = " + "),
                     "+ f(s, model = spde)")
formula <- as.formula(formula_str)

#>------------------------------------------------------------------------
#> 11. Run INLA: fit the model
#>------------------------------------------------------------------------

res <- inla(
  formula,
  family = "poisson",
  data = inla.stack.data(stk.full.pp),
  control.predictor = list(
    compute = TRUE,
```

```
    link = 1,
    A = inla.stack.A(stk.full.pp)
  ),
  E = inla.stack.data(stk.full.pp)$e,
  control.compute = list(dic = TRUE, waic = TRUE),
  verbose = TRUE
)

#> Check main estimation results
summary(res)

#>------------------------------------------------------------------------
#> 12. Plot results
#>------------------------------------------------------------------------

#> Plotting the intensity map (following Moraga (2023), Section 23.3.9) ------

index <- inla.stack.index(stk.full.pp, tag = "pred.pp")$data

pred_mean <- res$summary.fitted.values[index, "mean"]
pred_sd <- res$summary.fitted.values[index, "sd"]
pred_ll <- res$summary.fitted.values[index, "0.025quant"]
pred_ul <- res$summary.fitted.values[index, "0.975quant"]

grid$mean <- NA
grid$sd <- NA
grid$ll <- NA
grid$ul <- NA

grid$mean[indicespointswithin] <- pred_mean
grid$sd[indicespointswithin] <- pred_sd
grid$ll[indicespointswithin] <- pred_ll
grid$ul[indicespointswithin] <- pred_ul

#Color palette
col_scale <- viridis(256, option = "viridis", direction = -1)

levelplot(raster::brick(grid[["mean"]]), layout = c(1, 1),
          names.attr = c("mean"),
          col.regions = col_scale,
          main = "",
          par.settings = list(par.main.text = list(cex = 2),    # title size
                              axis.text = list(cex = 1.5),      # tick labels
                              par.xlab.text = list(cex = 1.5),  # x-axis label
                              par.ylab.text = list(cex = 1.5))) # y-axis label
```

```
levelplot(raster::brick(grid[[c("mean", "sd")]]), layout = c(2, 1),
          names.attr = c("mean", "sd"),
          col.regions = col_scale)

levelplot(raster::brick(grid[[c("mean", "ll", "ul")]]), layout = c(3, 1),
          names.attr = c("mean", "0.025quant", "0.975percentile"),
          col.regions = col_scale)


#> Plot the Gaussian random field ------------------------------------------

#> Projecting from mesh to grid points
proj_grid <- inla.mesh.projector(mesh, loc = dp)

#> Interpolate Gaussian random field onto grid
grf_mean <- as.vector(inla.mesh.project(proj_grid, res$summary.random$s$mean))
grf_sd   <- as.vector(inla.mesh.project(proj_grid, res$summary.random$s$sd))
grf_ll   <- as.vector(inla.mesh.project(proj_grid,
    res$summary.random$s$`0.025quant`))
grf_ul   <- as.vector(inla.mesh.project(proj_grid,
    res$summary.random$s$`0.975quant`))

#> Initialize layers
grid$grf_mean <- NA
grid$grf_sd   <- NA
grid$grf_ll   <- NA
grid$grf_ul   <- NA

#> Assign values only at predicted points
grid$grf_mean[indicespointswithin] <- grf_mean
grid$grf_sd[indicespointswithin] <- grf_sd
grid$grf_ll[indicespointswithin] <- grf_ll
grid$grf_ul[indicespointswithin] <- grf_ul

#> Color palette
col_scale <- viridis(256, option = "viridis", direction = -1)

levelplot(raster::brick(grid[["grf_sd"]]),
          main = "",
          col.regions = col_scale,
          par.settings = list(
            par.main.text = list(cex = 2),  #> title size
            axis.text = list(cex = 1.5),    #> tick labels
            par.xlab.text = list(cex = 1.5),#> x-axis label
            par.ylab.text = list(cex = 1.5) #> y-axis label
          ))
```

```
levelplot(raster::brick(grid[[c("grf_mean", "grf_sd")]]),
          layout = c(2, 1),
          names.attr = c("Mean", "SD"),
          col.regions = col_scale)

levelplot(raster::brick(grid[[c("grf_mean", "grf_ll", "grf_ul")]]),
          layout = c(3, 1),
          names.attr = c("Mean", "0.025 Quantile", "0.975 Quantile"),
          col.regions = col_scale)

#> Plot Matérn covariance with the Bayesian credible interval bands ----------

nsamples <- 1000

#> Extract hyperparameter posteriors summaries
range_mean <- res$summary.hyperpar["Range for s", "mean"]
range_sd   <- (res$summary.hyperpar["Range for s", "sd"])
sigma_mean <- res$summary.hyperpar["Stdev for s", "mean"]
sigma_sd   <- (res$summary.hyperpar["Stdev for s", "sd"])

#> Draw samples from truncated normal to avoid negative values
rtruncnorm <- function(n, mean, sd, a=0) {
  q <- pnorm(a, mean, sd)
  u <- runif(n, q, 1)
  qnorm(u, mean, sd)
}

range_samples <- rtruncnorm(nsamples, range_mean, range_sd, a = 0)
sigma_samples <- rtruncnorm(nsamples, sigma_mean, sigma_sd, a = 0)

#> Distance vector (up to max sampled range * 3 for safety)
max_range <- max(range_samples)
h <- seq(0, 3 * max_range, length.out = 200)

#> For each posterior sample, compute covariance vector
cov_samples <- sapply(1:nsamples, function(i) {
  fields::Matern(h, range = range_samples[i],
                 smoothness = nu) * sigma_samples[i]^2
})

#> Compute mean covariance curve (posterior mean)
cov_mean <- rowMeans(cov_samples)

#> Compute pointwise 2.5% and 97.5% quantiles (credible band)
cov_lower <- apply(cov_samples, 1, quantile, probs = 0.025)
```

```
cov_upper <- apply(cov_samples, 1, quantile, probs = 0.975)

#> Data frame for plotting
df <- data.frame(distance = h,
                 cov_mean = cov_mean,
                 cov_lower = cov_lower,
                 cov_upper = cov_upper)
#> Plot
ggplot(df, aes(x = distance)) +
  geom_ribbon(aes(ymin = cov_lower, ymax = cov_upper), fill = "lightblue",
              alpha = 0.4) +
  geom_line(aes(y = cov_mean), color = "blue", size = 1.2) +
  labs(title = "", x = "Distance h", y = "Covariance C(h)")

#> Plot the posterior distribution of the hyperparameters --------------------

#> Extract marginals for "Range for s" and "Stdev for s"
range_marginal <- res$marginals.hyperpar[[which(names(res$marginals.hyperpar)
    == "Range for s")]]
sigma_marginal <- res$marginals.hyperpar[[which(names(res$marginals.hyperpar)
    == "Stdev for s")]]

#> Convert to data frames for ggplot
df_range <- data.frame(value = range_marginal[,1], density =
    range_marginal[,2],
                       param = "Range")
df_sigma <- data.frame(value = sigma_marginal[,1], density =
    sigma_marginal[,2],
                       param = "Sigma")

df_post <- rbind(df_range, df_sigma)

#> Plot densities
ggplot(df_post, aes(x = value, y = density, color = param, fill = param)) +
  geom_line(size = 1) +
  geom_area(alpha = 0.3) +
  facet_wrap(~ param, scales = "free") +
  labs(title = "", x = "Value", y = "Density")

#> Plot the fixed effects posterior distributions ----------------------------

#> Extract posterior marginals for fixed effects
fixed_margs <- res$marginals.fixed

#> Extract posterior means for fixed effects
fixed_means <- res$summary.fixed$mean
```

```
names(fixed_means) <- rownames(res$summary.fixed)

#> Prepare dataframe for marginals
df_list <- lapply(names(fixed_margs), function(param) {
  df <- data.frame(
    value = fixed_margs[[param]][, 1],
    density = fixed_margs[[param]][, 2],
    parameter = param,
    mean = fixed_means[param])
    return(df)
})

df_fixed <- do.call(rbind, df_list)

ggplot(df_fixed, aes(x = value, y = density)) +
  geom_line(color = "blue") +
  geom_vline(aes(xintercept = mean), linetype = "dashed", color = "red", size
      = 0.5) +
  facet_wrap(~ parameter, scales = "free") +
  labs(title = "", x = "Coefficient Value", y = "Density") +
  theme(text = element_text(size = 10),          # base text size
        axis.title = element_text(size = 18),    # x and y axis titles
        axis.text = element_text(size = 10),     # tick labels
        strip.text = element_text(size = 14),    # facet labels
        plot.title = element_text(size = 20, hjust = 0.5))  # plot title

#> Pearson-like residuals ----------------------------------------------------

grid_res <- 200  #> grid resolution in meters

#> Create grid of polygons
grid_sf <- st_make_grid(window_sf, cellsize = grid_res, what = "polygons") %>%
  st_as_sf()

#> Keep only polygons whose centroids are inside the window
grid_sf <- grid_sf[st_within(st_centroid(grid_sf), window_sf, sparse = FALSE),
    ]

#> Interpolate posterior mean from mesh to grid
eta_mean <- res$summary.linear.predictor$mean
mesh_points <- as.matrix(mesh$loc)

#> Use centroid coordinates for interpolation
grid_coords <- st_coordinates(st_centroid(grid_sf))[,1:2]

#> Nearest-neighbor interpolation
```

```
nn <- get.knnx(mesh_points[,1:2], grid_coords, k = 1)
grid_sf$eta <- eta_mean[nn$nn.index[,1]]

#> Compute fitted intensity per pixel
pixel_area <- grid_res^2
grid_sf$lambda_hat <- exp(grid_sf$eta)

#> Count events per pixel
grid_sf$Y_counts <- lengths(st_intersects(grid_sf, crimes_sf))

#> Compute Pearson residuals
grid_sf$residual <- (grid_sf$Y_counts - grid_sf$lambda_hat * pixel_area) /
  sqrt(grid_sf$lambda_hat * pixel_area)

#> Summarize residuals
summary(grid_sf$residual)
table(grid_sf$Y_counts)

#> Signed log transform
#grid_sf$residual_log <- sign(grid_sf$residual) * log1p(abs(grid_sf$residual))

ggplot(grid_sf) +
  geom_sf(aes(fill = residual), color = NA) +
  scale_fill_gradient2(low = "blue", mid = "white", high = "red", midpoint =
      0, limits = c(-20, 20) ) +
  theme_minimal() +
  labs(title = "LGCP Pearson-type residuals", fill = "Signed log Residual") +
  theme(axis.title = element_blank(),
        axis.text = element_blank(),
        axis.ticks = element_blank(),
        panel.grid = element_blank())
```

annex I

Crime categories and subcategories classification according to the GNR

Table I.1: Translation of crime categories.

Original (Portuguese)	**Translation (English)**
Animais de companhia	Companion animals
Estado	The state
Identidade cultural e integridade pessoal	Cultural identity and personal integrity
Legislação avulsa	Other legislation
Património	Property
Pessoas	Persons
Vida em sociedade	Life in society

Table I.2: Translation of selected street crime subcategories

Original (Portuguese)	**Translation (English)**
Roubo a banco ou outro estabelecimento de crédito	Bank robbery or robbery of another financial establishment
Dano contra o património cultural	Damage to cultural property
Incêndio/fogo posto em edifício, construção ou meio de transporte	Fire or arson in buildings, constructions, or means of transport
Outro dano	Other damage
Roubo por esticão	Pickpocketing
Roubo em transportes públicos	Robbery in public transports
Roubo na via pública (excepto por esticão)	Robbery in the street
Roubo a posto de abastecimento de combustível	Service station robbery
Furto em supermercado	Theft in a supermarket
Furto em veículo motorizado	Theft in motor vehicle
Furto de viatura	Theft of motor vehicle
Roubo a tesouraria ou estação de correio	Treasury or post office robbery

Original (Portuguese)	**Translation (English)**
Furto por carteirista	Wallet theft

annex II

Non-parametric estimation of the intensity surface across different time scales

Here we find the non-parametric estimations of the intensity surface by different subsets of the street crimes dataset, aggregated by parts of the day, days of the week and quarters of the study period (2022 and 2023). The estimates were obtained using Kernel Density Estimation with ppl bandwidth selection described in Section 4.4.1 from Chapter 4.

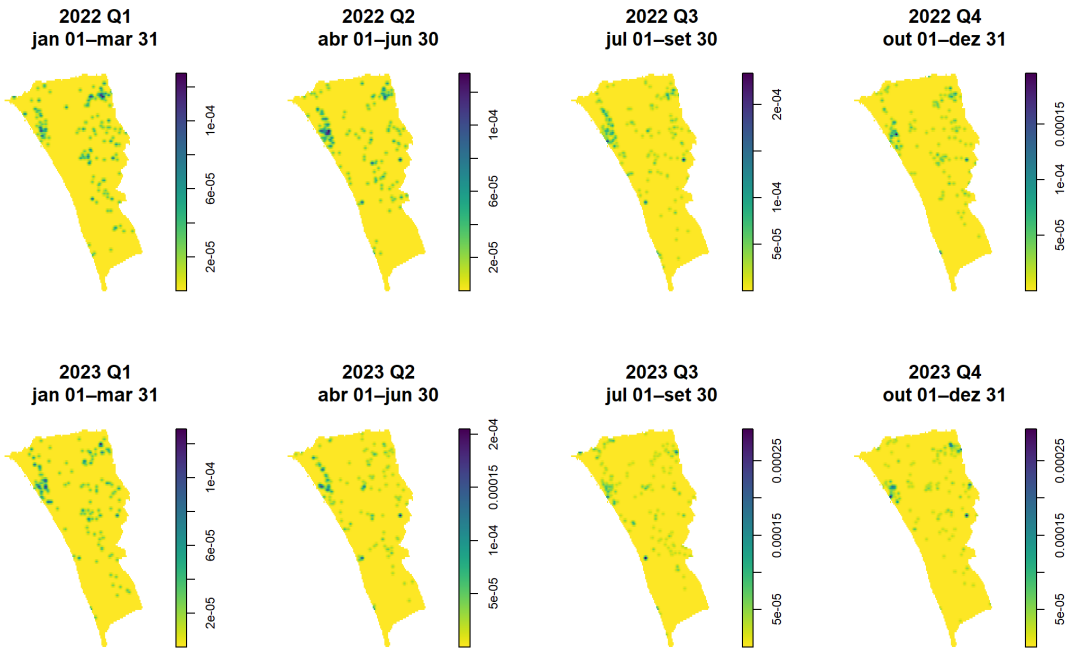


Figure II.1: Intensity surface non-parametric estimation per quarters.

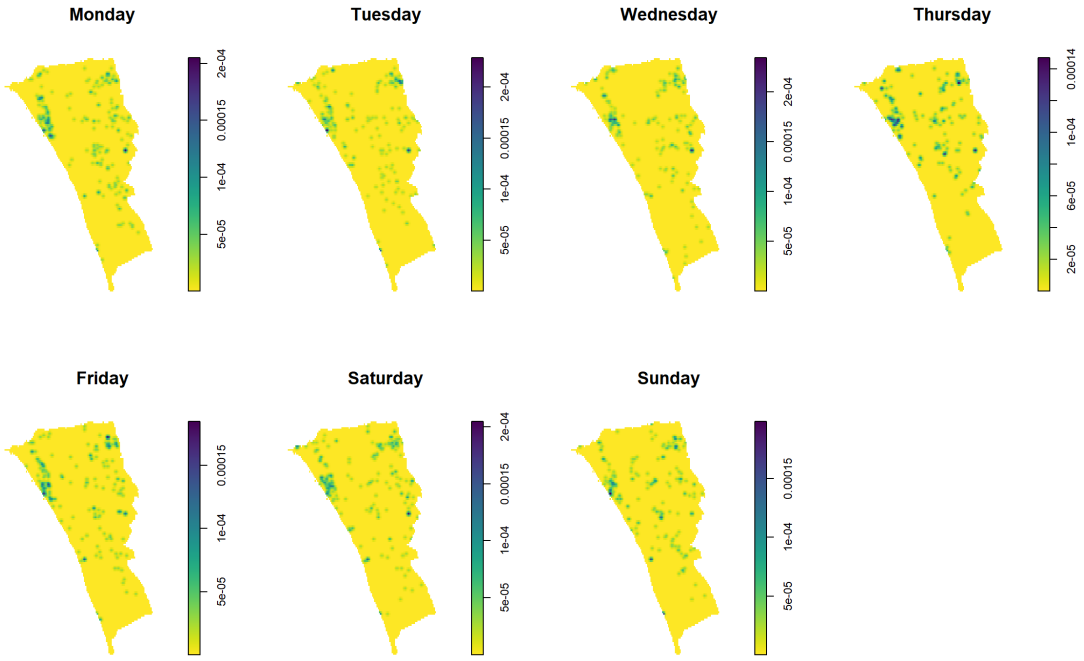


Figure II.2: Intensity surface non-parametric estimation per weekdays.

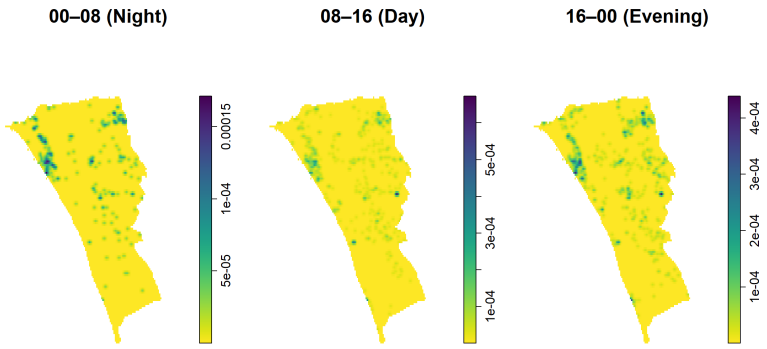


Figure II.3: Intensity surface non-parametric estimation per parts of the day.

annex III

Extra results from the LGCP model inference

This annex contains the posterior distributions for the hyperparameters and fixed effects coefficients, and the 95% credible intervals for the posterior intensity and posterior Gaussian Random Field (GRF) for Models 1, 5 and 6.

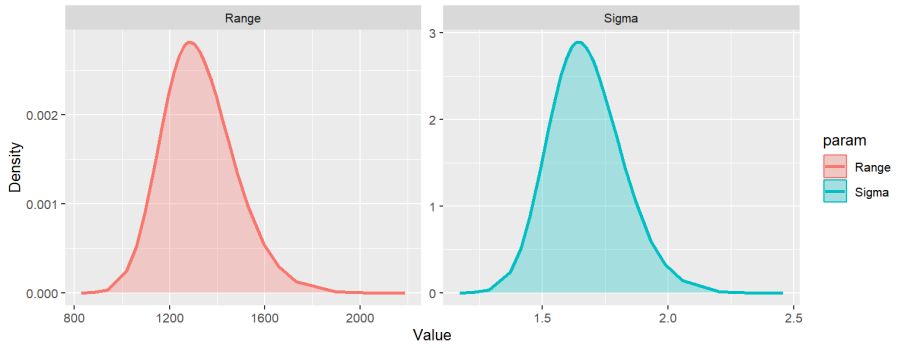


Figure III.1: Model 1: posterior distribution of the hyperparameters.

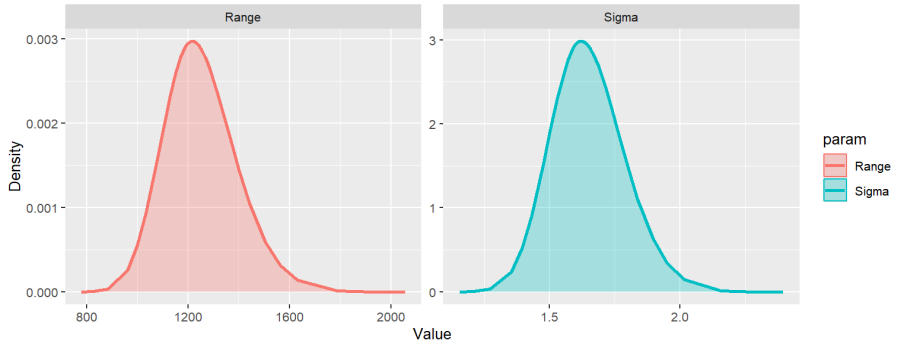


Figure III.2: Model 5: posterior distribution of the hyperparameters.

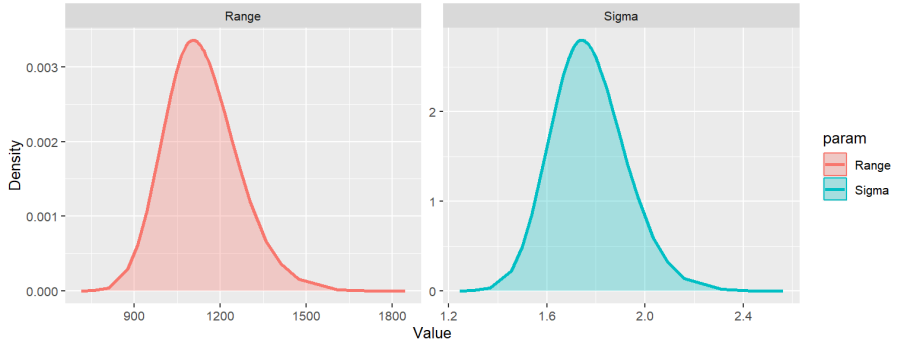


Figure III.3: Model 6: posterior distribution of the hyperparameters.

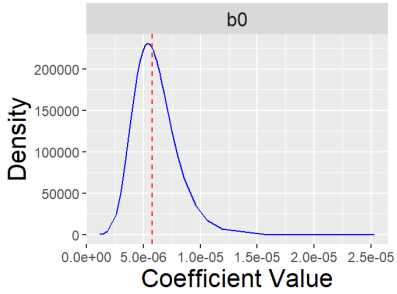


Figure III.4: Model 1: posterior distributions of the fixed effects (solid blue line) and posterior mode (dashed red line).

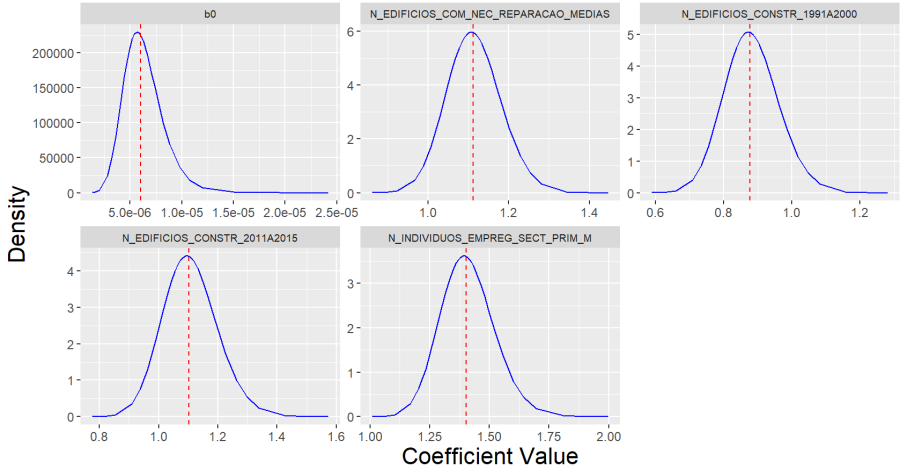


Figure III.5: Model 5: posterior distributions of the fixed effects (solid blue line) and posterior mode (dashed red line).

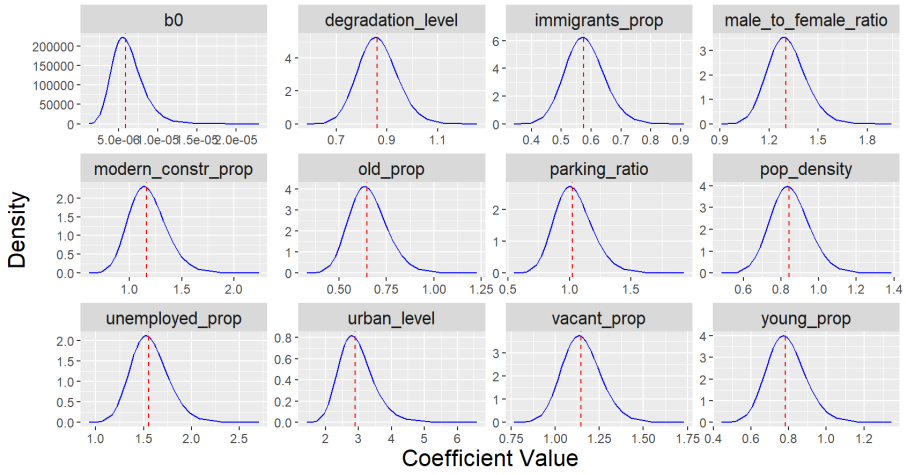


Figure III.6: Model 6: posterior distributions of the fixed effects (solid blue line) and posterior mode (dashed red line).

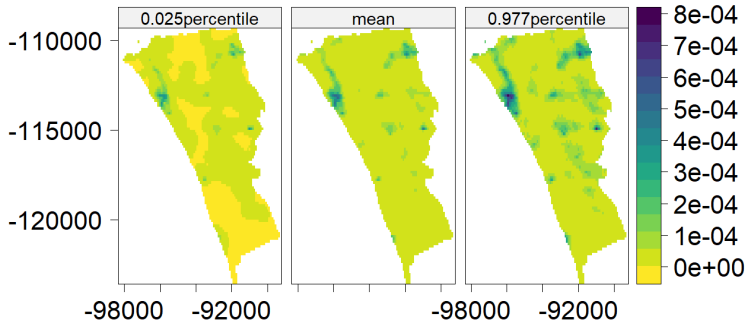


Figure III.7: Model 1: predicted mean intensity surface and 95% credible intervals.

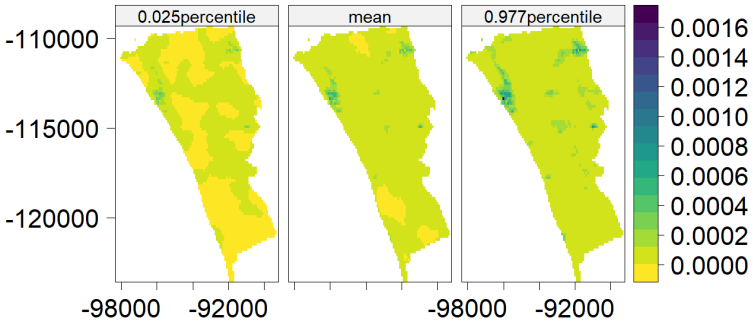


Figure III.8: Model 5: predicted mean intensity surface and 95% credible intervals.

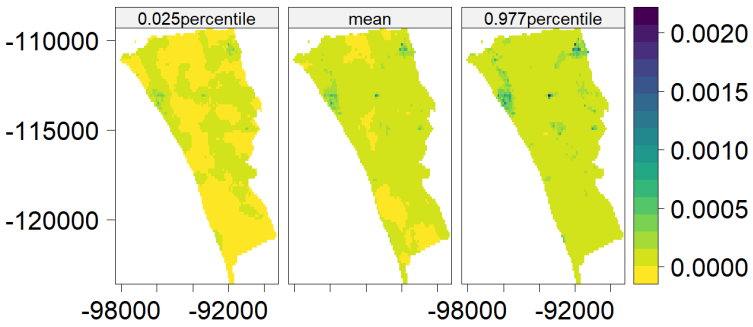


Figure III.9: Model 6: predicted mean intensity surface and 95% credible intervals.

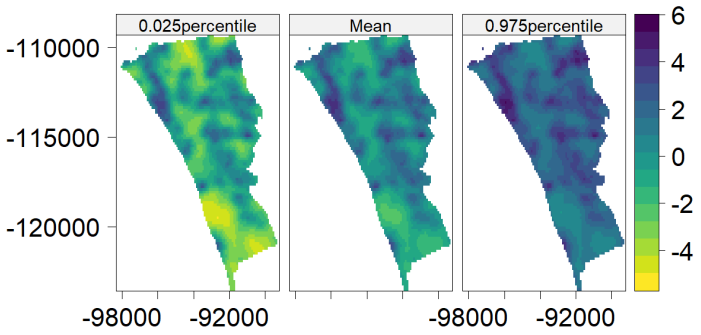


Figure III.10: Model 1: predicted mean Gaussian Random Field surface and 95% credible intervals.

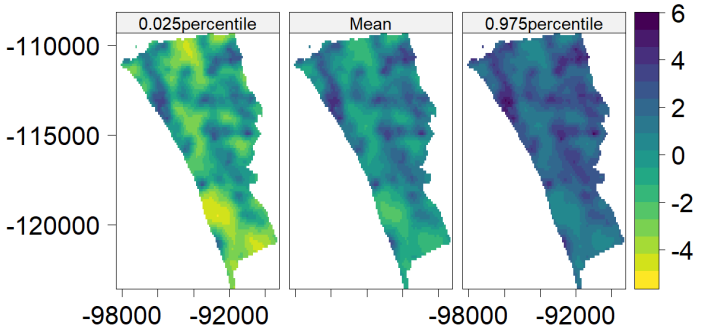


Figure III.11: Model 5: predicted mean Gaussian Random Field surface and 95% credible intervals.

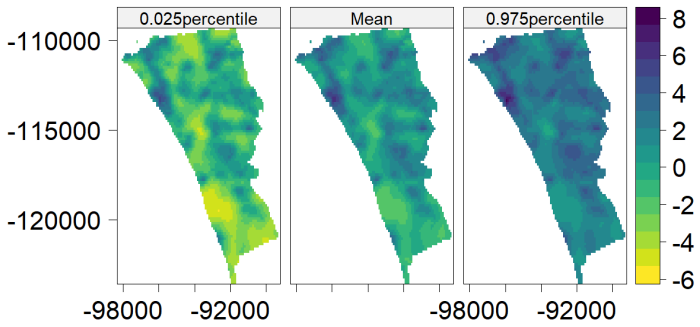


Figure III.12: Model 6: predicted mean Gaussian Random Field surface and 95% credible intervals.