
New Approaches in Statistical Modeling

Dissertation

an der Fakultät für Mathematik, Informatik und Statistik der
Ludwig-Maximilians-Universität München

vorgelegt von

Marc Schneble

Eingereicht am 05.08.2021

Dissertation an der Fakultät für Mathematik, Informatik und Statistik der Ludwig-Maximilians-Universität München

1. Berichterstatter: Prof. Dr. Göran Kauermann (LMU München)
2. Berichterstatterin: Prof. Dr. Annika Hoyer (LMU München)
3. Berichterstatter: Prof. Dr. Brian D. Marx (Louisiana State University, Baton Rouge)

Tag der Einreichung: 05.08.2021

Tag der Disputation: 29.10.2021

New Approaches in Statistical Modeling

Dissertation

an der Fakultät für Mathematik, Informatik und Statistik der
Ludwig-Maximilians-Universität München

vorgelegt von

Marc Schneble

Eingereicht am 05.08.2021

Dissertation an der Fakultät für Mathematik, Informatik und Statistik der Ludwig-Maximilians-Universität München

1. Berichterstatter: Prof. Dr. Göran Kauermann (LMU München)
2. Berichterstatterin: Prof. Dr. Annika Hoyer (LMU München)
3. Berichterstatter: Prof. Dr. Brian D. Marx (Louisiana State University, Baton Rouge)

Tag der Einreichung: 05.08.2021

Tag der Disputation: 29.10.2021

Acknowledgment

I dedicate my most enormous thank you to my supervisor Göran Kauermann, who has always been at my side with advice and assistance over the past three years. Thanks as well for many cappuccino dates where we shared our opinion about “the Franconian”. Furthermore, Annika Hoyer deserves big thanks for joining my examination committee. Thanks as well for teaching several courses together, starting with my first up to my final semester at the statistics department. I also want to thank Brian D. Marx for being the external reviewer of my dissertation. Without wishing to exaggerate, some parts of my work would not have been created if he had not published the “P-splines” 25 years ago, together with Paul H.C. Eilers.

Many more people from the department of statistics have contributed to this thesis, directly and indirectly. First, I want to thank the COVID-19 data analysis group (CODAG) for engaging discussions. It was a pleasure to work with all of you, but I think none of us would have thought in March 2020 that we would still meet regularly today and prepare reports. In this matter, special thanks go to Ursula Berger from the IBE and Giacomo De Nicola (if possible, I would insert a particular emoji at this place) for being co-authors in two papers that we successfully published together.

Another special thank you goes to Cornelius Fritz, who always had valuable advice when I struggled with a scientific problem. In this matter, I also want to thank Michael Lebacher, whom I had a great time with at two conferences. Thank you, Benjamin Sischka, for many funny moments and for sharing your office with me for a long time until corona separated us. Likewise, I want to thank Christopher Küster and Matthias Aßenmacher for granting me “asylum” in their office during the past months. Thank you, Michael Windmann and Sevag Kevork, for interesting discussions in the corridor and for handling much of the administrative work at our department. Thank you, Iris Burger, for supporting the scientific staff. Enjoy your retirement! Thank you, Christoph Berninger (behind the “steel door”), for the barbecues on your balcony and for giving me tennis lessons. Thank you, Alexander Bauer, Maximilian Weigert and Martje Rave, for some fabulous poker nights and hoping for more to come.

Moreover, I would like to thank my friends from Ulm University, especially those who have accompanied me since my first days in the “Trainingscamp”. During this fantastic time, I also got to know my girlfriend, Sabrina. Thank you so much for being patient with me and your understanding when I had a stressful time. Last but not least, I want to thank my parents and my sister Sophia, looking forward to seeing you soon at Lake Constance.

Kurzfassung

Diese kumulative Dissertation befasst sich mit der statistischen Modellierung von räumlichen Netzwerkdaten, sowie von Daten zur Pandemie des SARS-CoV-2-Virus. Statistische Modellierung kann im übertragenden Sinne als ein großer “Werkzeugkasten” verstanden werden, mit dem man Phänomene der realen Welt durch eine geeignete mathematische Formalisierung approximiert. Die in dieser Arbeit verwendeten Modelle beruhen in erster Linie auf Regression, wobei die Schwerpunkte auf der Glättung mit penalisierten Splines unter Einbeziehung von zufälligen Effekten liegen. Im Allgemeinen bestehen die Vorteile von Regressions- und statistischen Modellen darin, dass sie interpretierbare Modellergebnisse liefern und Vorhersagen über unbeobachtete Zustände erlauben. Gleichzeitig ist eine Beurteilung der zugrunde liegenden Unsicherheit der Schätzungen möglich. Diese drei Schlüsselaspekte des statistischen Modellierens spielen eine entscheidende Rolle in den fünf Beiträgen dieser kumulativen Dissertation.

Die ersten drei Artikel befassen sich mit statistischen Modellen und ihrer Anwendung auf Daten, die auf Netzwerken beobachtet werden. Netzwerke sind Strukturen, die aus durch Kanten verbundene Knoten bestehen. Während Netzwerke in natürlicher Weise abstrakte Beziehungen wie soziale Netzwerke oder ein Netzwerk von Geschäftspartnern darstellen können, liegt der Schwerpunkt in dieser Arbeit auf Netzwerken mit einer räumlichen Interpretation. Im ersten Artikel wird ein neues Modell entwickelt, welches erlaubt, statistische Rückschlüsse auf unbeobachtete Fahrten in Bike-Sharing-Netzwerken zu ziehen. Dabei stellen die Fahrradstationen die Eckpunkte des Netzwerks dar, und die Wege zwischen den Fahrradstationen entsprechen den Kanten. Der darauf folgende Artikel behandelt räumliche Netzwerke und die Schätzung der Intensität von stochastischen Prozessen, deren Realisierungen in räumlichen Netzwerken beobachtet werden. Die Methodik erlaubt auch die Einbeziehung von Kovariablen bei der Schätzung der Intensität. Diese Art der Modellierung ist neu und mit den aktuellen, auf Kerndichteschätzung basierenden Methoden, nicht möglich. Um die Methode frei zugänglich zu machen, wurde ein **R**-Paket implementiert. Der letzte Beitrag im Bereich der Netzwerke befasst sich mit der Vorhersage der Belegung von Parkplätzen, die entlang eines Straßennetzes verteilt sind. In diesem Zusammenhang wird die Netzwerkstruktur genutzt, um räumliche Abhängigkeiten zu modellieren. Darüber hinaus basieren die Vorhersagen auf einem Semi-Markov-Modell, um die nicht-exponentielle Dauer der einzelnen Zustände zu berücksichtigen. Die Übergangintensitäten werden mit Hilfe von Überlebenszeitmodellen geschätzt.

Der zweite Teil dieser Dissertation befasst sich mit der Pandemie des SARS-CoV-2-Virus, das die Krankheit COVID-19 verursacht. Das deutsche Robert-Koch-Institut (RKI) stellt täglich Daten zu COVID-19-Infektionen und Todesfällen im Zusammenhang mit COVID-19 zur Verfügung, mit zusätzlichen Angaben zu Region, Geschlecht und Alter der Infizierten. Aus verschiedenen Gründen geben die Rohdaten keinen ausreichenden Aufschluss über den Schweregrad der Pandemie, weswegen statistische Modelle auf die Daten angewandt werden. Ein Beitrag befasst sich mit der Vorhersage tödlicher Infektionen auf regionaler Ebene unter Berücksichtigung der lokalen Bevölkerungsstruktur. Damit ist das Modell in der Lage, auch eine regionalspezifische Beurteilung der Schwere der Pandemie vorzunehmen. In einem zweiten Beitrag werden die tödlich endenden Infektionen mit der Anzahl der registrierten Infektionen zueinander in Beziehung gesetzt, um die Veränderung der Fallentdeckungsrate im Laufe der Zeit zu quantifizieren. Darüber hinaus ermöglicht die Methode, den Verlauf der tatsächlichen Zahl der Infektionen zu schätzen, während die gemeldeten Infektionszahlen durch verschiedene Teststrategien beeinflusst sind.

Summary

This cumulative dissertation is concerned with statistical modeling of data observed on geometric networks and data related to the pandemic of the SARS-CoV-2 virus. Statistical modeling in its broadest sense encompasses a large “toolbox” to approximate real-world phenomena in a mathematically formalized manner. Models used in this work are primarily regression-based, with an emphasis on penalized spline smoothing and the inclusion of random effects to control for latent heterogeneities. In general, the benefits of regression and statistical models include creating interpretable model results and making predictions about unobserved states while adequately communicating the underlying uncertainty. These three key aspects of statistical modeling play a crucial role in the five contributions of this cumulative dissertation.

The first three articles cover statistical models and their application to data observed on networks, i.e. structures consisting of vertices connected by a set of edges. While networks serve as a natural device to represent abstract relationships such as social networks or a network of commercial partners, the focus here is on spatial networks. The first article develops a new model to draw statistical inference about unobserved trips in bike-sharing networks. Here, bike stations represent the network’s vertices, and the paths between the bike stations correspond to the edges. The consecutive article treats spatial networks, focusing on estimating stochastic processes’ intensity functions with realizations observed on spatial networks. The methodology also allows fitting the intensity with covariates, which is novel and not feasible with the current state-of-the-art methods based on kernel smoothing. To make the methodology freely available, an **R** package has been implemented. The last contribution in the field of networks covers the prediction of on-street parking occupancy, where parking lots are distributed along a street network. In this context, the network structure is utilized to model spatial dependencies. Moreover, predictions are based on a semi-Markov model to account for non-exponential duration times in each state and the transition intensities are estimated employing time to event models.

The second part of this dissertation deals with the pandemic of the SARS-CoV-2 virus, which causes the disease COVID-19. The German Robert Koch Institute (RKI) daily provides data concerning COVID-19 infections and deaths related to COVID-19 with information on the infected’s region, gender, and age. For several reasons, the raw data do not indicate the seriousness of the pandemic sufficiently well, which is why statistical models are used to get a clearer picture of the pandemic. One contribution is concerned with nowcasting fatal infections on a regional level while accounting for the local population structure. Thus, the model is capable of evaluating the region-specific seriousness of the pandemic. A second paper relates infections ending fatally to registered infections aiming at quantifying the change of the case detection ratio over time. Furthermore, the method allows assessing the relative course of the actual number of infections while testing strategies influence the reported numbers.

Contents

	Page
1 Introduction	1
1.1 Overview	1
1.2 Generalized additive mixed models	3
1.2.1 Model formulation	3
1.2.2 Penalized splines	4
1.2.3 Inference	8
1.2.4 Extension to non-exponential families	13
1.3 Geometric networks	14
1.3.1 Definition	14
1.3.2 Estimation of latent network flows	15
1.3.3 Intensity estimation of point processes	16
1.3.4 On-street parking	19
1.4 Time-to-event models	19
1.4.1 Model for continuous data	20
1.4.2 Model for discrete data	21
1.5 Stochastic processes	23
1.5.1 Markov processes	23
1.5.2 Semi-Markov processes	24
1.6 Software and computational aspects	26
1.7 Discussion	27
I Statistical modeling of spatial network data	35
2 Estimation of latent network flows in bike-sharing systems	37
3 Intensity estimation on geometric networks	39
3.1 Intensity estimation on geometric networks with penalized splines	39
3.A Intensity estimation on geometric networks with the R package geonet	64
4 Statistical modeling of on-street parking lot occupancy in smart cities	95
II Statistical modeling of COVID-19 data	125
5 Nowcasting fatal COVID-19 infections on a regional level in Germany	127
6 A statistical model for the dynamics of COVID-19 infections and their case detection ratio in 2020	149
Affidavit	161

Chapter 1

Introduction

“Les questions les plus importantes de la vie ne sont en effet, pour la plupart, que des problèmes de probabilité.”

— Pierre-Simon Laplace (* 1749, † 1827),
French mathematician

1.1. Overview

Probability and random variables More than 200 years ago, Pierre-Simon Laplace stated in a philosophical essay on probabilities that “the most important questions of life are indeed, for the most part, only problems of probability” (Laplace, 1814). This statement has proven to be correct. Even in the field of physics, which has for a long time been thought of as a purely deterministic discipline, probability plays an important role as e.g. in quantum mechanics (Gottfried and Yan, 2013). There have been several approaches to formalize the term “probability”. A mathematical definition was proposed by the Soviet mathematician Andrey Nikolaevich Kolmogorov who introduced an axiomatic system which is now widely known as the “Foundations of the Theory of Probability” (Kolmogorov, 1950). In other writings, probability has been viewed as a physical propensity (Popper, 1959) or has been grounded on a set of plausibility assumptions (Cox, 1946) where the latter serves as justification for the Bayesian view on probability. The measure-theoretic perception of probability by Kolmogorov can be formalized in terms of a probability space $(\Omega, \mathcal{F}, \mathbb{P})$ where Ω is a sample space, \mathcal{F} is an event space (a σ -algebra) and $\mathbb{P} : \mathcal{F} \rightarrow [0, 1]$ is a probability measure which assigns every event $\mathcal{A} \in \mathcal{F}$ the probability $\mathbb{P}(\mathcal{A})$. A random vector of dimension p on $(\Omega, \mathcal{F}, \mathbb{P})$ is a measurable function $\mathbf{X} : \Omega \rightarrow \mathbb{R}^p$ and for $\omega \in \Omega$ the vector $\mathbf{x} = \mathbf{X}(\omega)$ is a realization of \mathbf{X} . If $p = 1$, X is denoted as a real-valued random variable and $x = X(\omega)$ is a realization of X . For a thorough introduction to measure theoretic probability, see Pollard (2002). Since this dissertation focuses on applying probabilistic theory to statistical models, technical details play a subordinate role. Hence, it is implicitly assumed that probabilities and random variables are well-defined.

Statistical models Even though the general formulation of a statistical model is of minor importance in applied statistics, for completeness, a concise definition shall be given here, see also Cox and Hinkley (1979), Bernardo and Smith (2009) and McCullagh et al. (2002). In a nutshell, a statistical model can be comprehended as a set of probability distributions \mathcal{P} that enables to draw inference about some characteristics of a population of interest. The inference is based on the outcome of a statistical experiment or an observational study (the data) which

itself consists of a set of statistical units \mathcal{U} , a response scale \mathcal{Y} and a covariate space \mathcal{X} . In this context, the set $\mathcal{Y}^{\mathcal{U}}$ of mappings $y : \mathcal{U} \rightarrow \mathcal{Y}$ is the sample space with $y_i \in \mathcal{Y}$ denoting the observed response of unit $i \in \mathcal{U}$. Likewise the set $\mathcal{X}^{\mathcal{U}}$ of mappings $\mathbf{x} : \mathcal{U} \rightarrow \mathcal{X}$ is the design space of the model with $\mathbf{x}_i \in \mathcal{X}$ being a point in the covariate space. The notation in a normal font and a bold font, respectively, already suggests that usually y_i is one-dimensional and \mathbf{x}_i is multivariate. If \mathcal{P} is characterized by a set of parameters Θ , one can write $\mathcal{P} = \{\mathbb{P}_\theta \mid \theta \in \Theta\}$ and call it a parametric model with parameter space Θ . A parametric model is said to be identifiable if $\mathbb{P}_{\theta_1} = \mathbb{P}_{\theta_2}$ implies that $\theta_1 = \theta_2$ for $\theta_1, \theta_2 \in \Theta$, i.e. if \mathbb{P}_θ is injective. The identification of a parametric model is usually required since most of the model parameters are associated with the effect of covariates \mathbf{x} on the response y . In general, the identifiability constraints need to be chosen with care since they affect the quantification of uncertainty of the parameter estimates.

Konishi and Kitagawa (2008) identify three primary purposes of a statistical model. First, the underlying stochastic structure of the data shall be explained, which involves, in practice, the estimation of uncertainty and bias. Second, statistical models can be employed to predict future or unobserved data based on observed data. Lastly, the interpretation of the modeling results or the extraction of information is of importance. A comprehensive textbook for statistical inference is Casella and Berger (2001).

Contributing articles Each of the Chapters 2 - 6 encompasses a contributing article which has been either published in a statistics journal, has been accepted for publication or is currently under review for being published. In the latter case, the latest version of the article, which has been submitted to the respective journal, is included. Furthermore, the contributions of all authors involved in creating each of the manuscripts are specified.

Outline of this chapter While the catalog of statistical models and their various applications is rather extensive, this introductory chapter is devoted to those models which are vital for Chapters 2 - 6.¹ Section 1.2 introduces generalized additive mixed models, a rich and flexible class of regression models where the response variable belongs to the exponential family of distributions. In doing so, emphasis is placed on the flexible and smooth modeling of covariate effects and the modeling with random effects. Statistical inference is discussed with a focus on the duality of smoothness and random effects. Moreover, the extension of the model to a broader class of distributions that do not necessarily belong to the exponential family is shortly sketched. Section 1.3 deals with statistical models where the response variable is not observed in a Euclidean space but on a spatial network. A simple idea allows using the theory introduced in Section 1.2 to estimate the intensity of point processes whose realizations are observed on a geometric network. Thus, covariate effects can be estimated as well, which is novel. Section 1.4 treats time-to-event models, which are closely related to generalized additive mixed models. While the latter class of models aims to infer the mean value of some population characteristics, when employing time-to-event models, the interest lies in modeling the distribution function or survivor function, respectively, of a random duration time. Moreover, different censoring mechanisms play an important role in this model class. Section 1.5 gives a brief introduction to the theory of continuous-time stochastic processes with emphasis on (semi-)Markov processes, which can be used for predicting future states of these processes. In Section 1.6, a short overview of the software is presented, which was used to implement the statistical models proposed in later chapters. Section 1.7 concludes the introduction with a short general discussion.

¹At this point it should be noted that the notation in Chapters 2 - 6 does not necessarily coincide with the notation in this chapter.

1.2. Generalized additive mixed models

“Statisticians, like artists, have the bad habit of falling in love with their models.”

— George Edward Pelham Box (* 1919, † 2013),
British statistician

1.2.1. Model formulation

Exponential family To establish the comprehensive class of generalized additive mixed models let us start with the definition of the exponential family of probability distributions being the basic building block of generalized linear models (GLMs, Nelder and Wedderburn, 1972, McCullagh and Nelder, 1989). A random variable Y with mean $\mathbb{E}(Y) = \mu$ belongs to the exponential family if its density (with respect to the Lebesgue measure or the counting measure, respectively) can be expressed as

$$f_Y(y; \theta, \phi) = \beta(y, \phi) \exp\left(\frac{y\theta - b(\theta)}{\phi}\right), \quad (1.1)$$

where $\theta = \theta(\mu)$ denotes the natural parameter and $\phi > 0$ is the dispersion parameter. Examples comprise the binomial distribution, the negative binomial distribution and the Poisson distribution for a discrete response scale as well as the gamma distribution and the normal distribution for a continuous response scale.

Generalized additive model Let us assume that the set of statistical units can be described as the index set $\mathcal{U} = \{1, \dots, n\}$. Further, let Y_i belong to the exponential family with observations $y_i \in \mathcal{Y}$ for $i \in \mathcal{U}$ and $\mathbf{x}_i = (x_{i1}, \dots, x_{iK})^\top \in \mathcal{X}$ be a point in the K -dimensional covariate space. Assuming that $Y_i | \mathbf{x}_i$ are independent for $i = 1, \dots, n$ with mean $\mu_i = \mathbb{E}(Y_i | \mathbf{x}_i)$ and following Wood (2017), a generalized additive model (GAM, Hastie and Tibshirani, 1990) has the general structure

$$g(\mu_i) = \eta_i = \mathbf{A}_i \boldsymbol{\beta} + \sum_{S_1 \subset \{1, \dots, K\}} f_k(x_{ik}; \boldsymbol{\gamma}_k) + \sum_{S_2 \subset \{1, \dots, K\}^2} f_{kl}(x_{ik}, x_{il}; \boldsymbol{\gamma}_{kl}). \quad (1.2)$$

The so-called link function $g(\cdot)$ is strictly monotonic and connects the linear predictor η_i with the mean μ_i . If $g(\mu_i) = \theta(\mu_i)$, the link is said to be canonical. The matrix \mathbf{A}_i is the i -th row of the model matrix representing the parametric part of the model with corresponding parameter vector $\boldsymbol{\beta}$. The functions $f_k(\cdot, \boldsymbol{\gamma}_k)$ and $f_{kl}(\cdot, \cdot; \boldsymbol{\gamma}_{kl})$ represent univariate or bivariate smooth covariate effects which are shaped by parameters $\boldsymbol{\gamma}_k$ and $\boldsymbol{\gamma}_{kl}$, respectively. However, the linear predictor can contain smooth terms of order three or higher which is, for simplicity of notation, not considered here. Furthermore, note that also the smooth covariate effects are represented in a way such that they are linear in their parameters. Details are provided in Section 1.2.2. The sets S_1 and S_2 are index sets and refer to the respective covariates modeled smoothly. Finally, let $\boldsymbol{\theta} = (\boldsymbol{\beta}^\top, \boldsymbol{\gamma}^\top)^\top$ denote the vector of all GAM model parameters.

From GAMs to GAMMs Suppose now that the set of statistical units \mathcal{U} is additionally characterized by an attribute t , i.e. $\mathcal{U} = \{i, t \mid i = 1, \dots, n, t \in \mathcal{T}_i\}$ with \mathcal{T}_i being a countable set that might depend on subject index i . Examples are longitudinal studies where t is associated with time and the pair $\{i, t\} \subset \mathcal{U}$ stands for the t -th observation of the i -th individual (e.g. Datar and Sturm, 2004). For simplicity of notation, \mathcal{T} is assumed to be identical for all subjects $i = 1, \dots, n$, but results generalize easily to unbalanced designs. Moreover, in many studies \mathcal{U} has a nested, hierarchical structure, see e.g. Antweiler (2001) for an example from econometrics.

In most settings, Y_{it_1} and Y_{it_2} are not independent for $t_1, t_2 \in \mathcal{T}$. In those cases, covariates can mostly not completely account for subject specific heterogeneity. Therefore, random effects $\mathbf{u} = (\mathbf{u}_1^\top, \dots, \mathbf{u}_n^\top)^\top$ are added to the linear predictor as specified in (1.2) which leads to the following definition of a generalized additive mixed model (GAMM). It is assumed that $Y_{it} \mid \mathbf{x}_{it}, \mathbf{u}_i$ are independently distributed according to an exponential family distribution with conditional mean $\mu_{it} = \mathbb{E}(Y_{it} \mid \mathbf{x}_{it}, \mathbf{u}_i)$, where $g(\mu_{it}) = \eta_{it} + \mathbf{Z}_{it}\mathbf{u}$. As above, g is a monotonic link function, η_{it} is the predictor for the t -th observation of the i -th unit and \mathbf{Z}_{it} is the respective row of the random effects model matrix \mathbf{Z} . The random effects are assumed to be independent following a prior distribution which is usually Gaussian such that $\mathbf{u}_i \stackrel{\text{indep.}}{\sim} \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_\vartheta)$ or, equivalently, $\mathbf{u} \sim \mathcal{N}(\mathbf{0}, \tilde{\boldsymbol{\Sigma}}_\vartheta)$, where $\tilde{\boldsymbol{\Sigma}}_\vartheta = \mathbf{I}_n \otimes \boldsymbol{\Sigma}_\vartheta$. The parameters ϑ shape the covariance matrix of the random effects and \otimes denotes the Kronecker product (Graham, 2018). A special case is the random intercept model where $\mathbf{u} = (u_1, \dots, u_n) \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_n \sigma^2)$ with $\vartheta = \sigma^2$ and \mathbf{Z} has the form of an ANOVA (Girden, 1992) model matrix. Note that $\mathbb{E}(\mathbf{u}) = \mathbf{0}$ ensures the identifiability of random effects.

1.2.2. Penalized splines

Univariate B-splines In this dissertation, the smooth effect $f_k(\cdot; \boldsymbol{\gamma}_k)$ of the k -th covariate is modeled utilizing a B-spline basis representation

$$f_k(x_{ik}; \boldsymbol{\gamma}_k) = \sum_{j=1}^{d_k} \gamma_{kj} B_{kj}^q(x_{ik}) \quad (1.3)$$

where $B_{kj}^q(\cdot)$ is a d_k -dimensional B-spline basis of order $q \in \mathbb{N}_0$ and $\boldsymbol{\gamma}_k = (\gamma_{k1}, \dots, \gamma_{kd_k})^\top$ is the corresponding vector of basis coefficients (Ruppert et al., 2003, Fahrmeir et al., 2007). Alternative basis representations are thin plate splines (“TP basis”, Wood, 2003). To generally introduce B-splines, the covariate index k is dropped in the following.

Let $[a, b] \subset \mathbb{R}$ be the domain of a continuous covariate x and $a = \tau_1 < \dots < \tau_{d-q+1} = b$ a sequence of equally spaced interior knots. In order to construct B-splines of order $q \in \mathbb{N}_0$, further $2q$ outer knots with $\tau_{1-q} < \dots < \tau_0 < a$ and $b < \tau_{d-q+2} < \dots < \tau_{d+q-1}$ with the same distance between two adjacent knots are required. The resulting d B-splines of order q are for $j = 1, \dots, d$ recursively defined by

$$B_j^q(x) = \frac{x - \tau_{j-q}}{\tau_j - \tau_{j-q}} B_{j-1}^{q-1}(x) + \frac{\tau_{j+1} - x}{\tau_{j+1} - \tau_{j+1-q}} B_j^{q-1}(x), \quad (1.4)$$

where a B-spline of order $q = 0$ is an indicator function $B_j^0(x) = \mathbb{1}_{[\tau_j, \tau_{j+1})}(x)$ (De Boor, 1972). As a result, each basis function $B_j(\cdot)$ is nonnegative, fulfills $\sum_{j=1}^d B_j(x) = 1$ for $x \in [a, b]$ and is supported at most on an interval defined by $q + 2$ consecutive knots. Sometimes, it is required that a function $f(\cdot)$ represented through a linear combination of B-splines fulfills $f(a) = f(b)$,

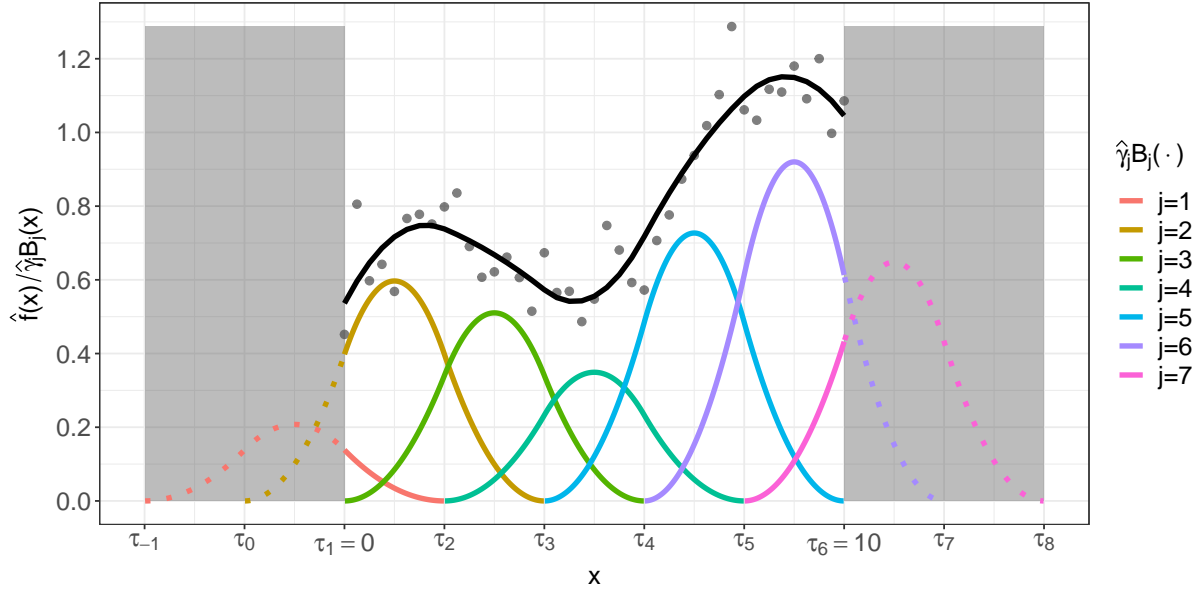


Figure 1.1.: Representation of an estimated smooth effect (black line) as the linear combination of quadratic B-splines (colored lines). Within the domain of the outer knots (grey background), the scaled B-splines are shown as dotted lines.

e.g. if circular effects such as the course of the year should be modeled. This only requires slight modification of the B-spline basis (Eilers and Marx, 1996).

In a similar manner as in the introducing example of Eilers and Marx (2021), Figure 1.1 exemplifies the use of B-splines in order to estimate the smooth effect of x on a response Y . The grey points represent covariate/response pairs (x_i, y_i) where y_i are independent realizations of a $\mathcal{N}(\mu_i = f(x_i), \sigma^2 = 0.05^2)$ random variable with $f(x) = 0.2 \sin(3x/\pi) + 0.005x^2 + 0.6$ and x_i being equally spaced in the interval $[0, 10] = [\tau_1, \tau_6]$. In this context, the goal is to estimate a function $\hat{f}(\cdot)$ which fits these n points in an optimal way but is likewise as smooth as possible. Therefore, a GAM with normally distributed response y and an identity link is fitted to the data. Here, the linear predictor has the form $\eta_i = f(x_i) = \mathbf{X}_i \boldsymbol{\gamma}$ with $\boldsymbol{\gamma} = (\gamma_1, \dots, \gamma_d)^\top$ being the parameters of the model. Furthermore, $\mathbf{X}_i = (B_1(x_i), \dots, B_d(x_i))$ is the i -th row of the model matrix $\mathbf{X} \in \mathbb{R}^{n \times d}$ that is related to the smooth effect of x . Details with regard to the estimation of the model parameters follow later. The sampled data points and the estimated smooth effect of x are shown as grey points and a black line, respectively, in Figure 1.1. Here, $\hat{f}(\cdot)$ is represented through $d = 7$ quadratic B-splines, i.e. $q = 2$. Thus, $\hat{f}(x) = \sum_{j=1}^7 \hat{\gamma}_j B_j(x)$ and $\hat{\gamma}_j$ are the estimated B-spline coefficients. Moreover, Figure 1.1 also suggests how B-splines are constructed by the use of the outer knots.

Dealing with the identification issue Smooth covariate effects suffer from identifiability problems, which is not the case when modeling random effects. To illustrate this, suppose that a model without intercept has the form $g(\mu_i) = f(x_{i1}; \boldsymbol{\gamma}_1) + f(x_{i2}; \boldsymbol{\gamma}_2)$, i.e. the parameter vector is given by $\boldsymbol{\theta} = (\boldsymbol{\gamma}_1^\top, \boldsymbol{\gamma}_2^\top)^\top$. However, the model with parameter vector $\tilde{\boldsymbol{\theta}} = (\tilde{\boldsymbol{\gamma}}_1^\top, \tilde{\boldsymbol{\gamma}}_2^\top)^\top$ which is for $c \neq 0$ defined by

$$g(\mu_i) = [f(x_{i1}; \boldsymbol{\gamma}_1) + c] + [f(x_{i2}; \boldsymbol{\gamma}_2) - c] = f(x_{i1}; \tilde{\boldsymbol{\gamma}}_1) + f(x_{i2}; \tilde{\boldsymbol{\gamma}}_2)$$

has a different parameterization, i.e. $\boldsymbol{\theta} \neq \tilde{\boldsymbol{\theta}}$, but apparently $\mathbb{P}_{\boldsymbol{\theta}} = \mathbb{P}_{\tilde{\boldsymbol{\theta}}}$. Thus, the statistical model is not identifiable.

In order to ensure the identifiability of general smooth effects, Wood (2003) proposed to center smooths around zero which leads to the constraint

$$\sum_{i=1}^n f(x_i) = \mathbf{1}^\top \mathbf{X} \boldsymbol{\gamma} = 0, \quad (1.5)$$

where $\mathbf{X} \in \mathbb{R}^{n \times d}$ is the design matrix for this smooth term and $\mathbf{1} \in \{1\}^n$ is a vector of ones. Since (1.5) must hold for any $\boldsymbol{\gamma} \in \mathbb{R}^d$, this is equivalent to $\mathbf{1}^\top \mathbf{X} = \mathbf{0}^\top$ which means that the column entries of \mathbf{X} sum up to zero. This can be achieved by creating a column centered matrix $\mathbf{X}_{\text{cent}} = \mathbf{X} - \mathbf{1} \mathbf{1}^\top \mathbf{X} / n$. Since the dimension of the null space of \mathbf{X}_{cent} is one, the identification can be achieved by deleting an arbitrary column, say the first, leading to the matrix $\mathbf{X}_{\text{ident}} \in \mathbb{R}^{n \times (d-1)}$. This matrix satisfies constraint (1.5) and the parameter vector $\boldsymbol{\gamma}$ now has dimension $d - 1$. Alternatively, constraint (1.5) can be achieved by making use of householder transformations (Wood, 2017). Similarly, this method reduces the column rank of \mathbf{X} by one. In general, identifiability constraints should be chosen with care since different constraints lead to different confidence intervals for the smooths (Wood et al., 2013).

Penalization It remains an open question on choosing the number of knots to properly model the smooth effect of a covariate x on a response Y . While choosing only a few knots carries the risk of underfitting the data, choosing too many knots results in overfitting the data. A simple but elegant approach to handle this tradeoff between bias and variance was proposed by Eilers and Marx (1996). They use a reasonably large number of equidistant knots and penalize the squared difference of adjacent basis coefficients. To illustrate their idea, suppose the number of knots is chosen to be too high, usually resulting in very wiggly effects $\hat{f}(x)$, i.e. the absolute difference of neighboring coefficients is relatively large. If those differences were penalized, the estimated effect of x would turn out to be smoother. Kauermann and Opsomer (2011) considered the number of knots as an additional parameter of the model while simultaneously penalizing the parameters related to smooth terms. However, if the number of knots is high enough and an appropriate penalization is attributed to the model parameters, further increasing the dimension of the B-spline basis is not beneficial. On the other side, if the dimension of the B-spline basis is chosen too low, it is not possible to reduce the bias since the degrees of freedom of $\hat{f}(x)$ are bounded from above (Ruppert, 2002).

When penalizing the squared r -th order differences of B-spline basis coefficients, a suitable penalty according to the above considerations for a single smooth term is

$$P(\rho; \boldsymbol{\gamma}) = \rho \sum_{j=r+1}^d (\Delta^r \gamma_j)^2 = \rho (\mathbf{D}_r \boldsymbol{\gamma})^\top (\mathbf{D}_r \boldsymbol{\gamma}) = \rho \boldsymbol{\gamma}^\top \mathbf{K} \boldsymbol{\gamma}, \quad (1.6)$$

where the r -th order differences of coefficients are recursively defined via $\Delta^r \gamma_j = \Delta^{r-1} \gamma_j - \Delta^{r-1} \gamma_{j-1}$ and $\Delta^1 \gamma = \gamma_j - \gamma_{j-1}$ (Fahrmeir et al., 2007). The parameter ρ is called the smoothing parameter and controls the amount of smoothing of a nonlinearly modeled effect. The optimal selection of smoothing parameters can be integrated in the maximum likelihood estimation of the model parameters as treated in Section 1.2.3. Usually, it is of advantage to express the penalty (1.6) as a quadratic form $P(\rho; \boldsymbol{\gamma}) = \rho \boldsymbol{\gamma}^\top \mathbf{K} \boldsymbol{\gamma}$ where $\mathbf{K} = \mathbf{D}_r^\top \mathbf{D}_r \in \mathbb{Z}^{d \times d}$ is a sparse penalty matrix and \mathbf{D}_r is called the difference matrix of order r . The penalty matrix \mathbf{K}_r is

rank deficient which gets more important later when a prior distribution is attributed to the parameter vector γ in order to imbed penalized splines in random effects modeling.

Asymptotic behavior of penalized splines with a focus on bias and variance was examined by Li and Ruppert (2008), who compared the results with other nonparametric regression methods such as the Nadaraya-Watson kernel estimator (Bierens, 1987). They also show that penalizing B-splines with an r -th order penalty is asymptotically equivalent to a derivative-based penalty of order r , where the latter combination of B-splines and penalization is known as a smoothing spline (Rice et al., 1983). Furthermore, Kauermann et al. (2009) asymptotically justify the usage of a higher dimensional basis of splines if the sample size n increases. A comprehensive monograph that treats both theory and various applications of penalized splines is Eilers and Marx (2021) who coined the term ‘‘P-splines’’ for this type of smoother.

Penalized splines in two dimensions The idea of penalized splines based on B-splines can easily be extended to two dimensions in order to model smooth interactions of covariates, e.g. x_k and x_l by $f_{kl}(x_k, x_l; \gamma_{kl})$ as in (1.2). B-splines in two dimensions can be constructed via one-dimensional B-splines from above according to $B_{kl,jm}^q(x_{ik}, x_{il}) = B_{kj}^q(x_{ik}) \cdot B_{lm}^q(x_{il})$ such that a B-spline basis representation of the two-dimensional smoother $f_{kl}(\cdot, \cdot; \gamma_{kl})$ is given by

$$f_{kl}(x_{ik}, x_{il}; \gamma_{kl}) = \sum_{j=1}^{d_k} \sum_{m=1}^{d_l} \gamma_{kl,jm} B_{kl,jm}^q(x_{ik}, x_{il}), \quad (1.7)$$

where $\gamma_{kl} = (\gamma_{kl,11}, \dots, \gamma_{kl,d_k 1}, \dots, \gamma_{kl,1 d_l}, \dots, \gamma_{kl,d_k d_l})^\top$ is the corresponding parameter vector of dimension $d_k \cdot d_l$. Figure 1.2 illustrates a B-spline basis in two dimensions where, however, for reasons of presentation, not all basis functions are shown. The black crosses mark the fictive knots in two dimensions. It can be seen that as in the one-dimensional setting, the basis functions are identical except for their position. In general, modeling the smooth interactions of three or more covariates is possible as well. An example is modeling the smooth interaction of space (in two or three dimensions) and time. However, it can be deduced from (1.7) that already the modeling of smooth 2-way interactions involves a high-dimensional B-spline basis which, due to the curse of dimensionality, increases in three or four dimensions.

The construction of an appropriate penalty is accordingly more involved when compared to the construction of the basis functions itself. Therefore, suppose that covariate x_k shall be penalized with a penalty of order r_k and corresponding penalty matrices $\mathbf{K}_k \in \mathbb{Z}^{d_k \times d_k}$, same for covariate x_l . Following Currie et al. (2006), a penalty matrix for the two-dimensional B-splines defined by (1.7) is given by

$$P(\rho_k, \rho_l; \gamma_{kl}) = \gamma_{kl}^\top (\rho_k [\mathbf{I}_{d_l} \otimes \mathbf{K}_k] + \rho_l [\mathbf{K}_l \otimes \mathbf{I}_{d_k}]) \gamma_{kl}. \quad (1.8)$$

Thus, each dimension of a tensor-product spline is penalized by a separate tuning parameter. However, requiring $\rho_{kl} = \rho_k = \rho_l$ enables to represent the penalty in the same fashion as before with $P(\rho_{kl}; \gamma_{kl}) = \rho_{kl} \gamma_{kl}^\top \mathbf{K}_{kl} \gamma_{kl}$ and $\mathbf{K}_{kl} = [\mathbf{I}_{d_l} \otimes \mathbf{K}_k] + \rho_l [\mathbf{K}_l \otimes \mathbf{I}_{d_k}]$.

Varying coefficient terms and functional random effects A special case of 2-way interactions are smooth terms of the form $f_{kl}(x_{ik}, x_{il}) = x_{il} f_k(x_{ik})$, where $f_k(\cdot)$ is a one-dimensional smoother as in (1.3). Hastie and Tibshirani (1993) denote this kind of smoother as a varying coefficient term since the coefficient for the l -th covariate varies smoothly with the observed value of the k -th covariate. Finally, B-splines and random effects can be combined to functional random

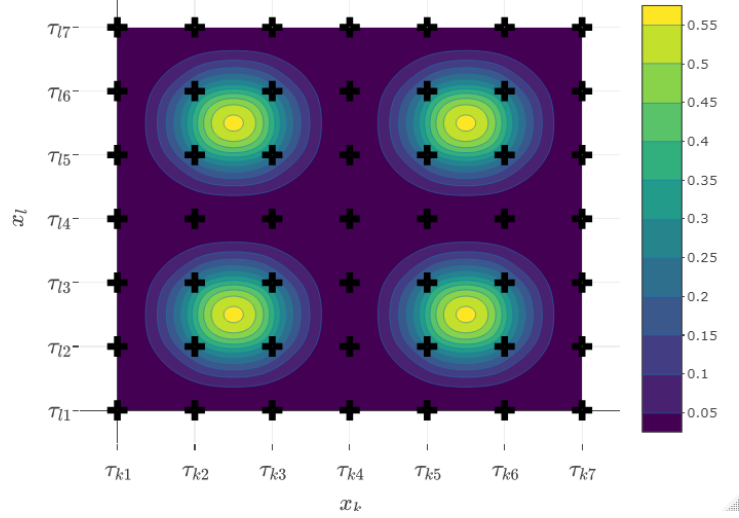


Figure 1.2.: A contour plot of quadratic B-splines in two dimensions. The color bar shows the function values of the two-dimensional B-splines.

effects, i.e. instead of a linear random slope, a random function can be fitted for every subject (Durban and Aguilera-Morillo, 2017). An instance of a model which has both extensions is illustrated in Chapter 6.

1.2.3. Inference

The foundation of inference in GAMMs is maximum likelihood estimation (MLE) which is based on the joint density $f_{\boldsymbol{\theta}}(\mathbf{y}, \mathbf{u})$ of the observed data $\mathbf{y} = (y_1, \dots, y_n)^\top$ and the unobserved random effects \mathbf{u} . Various methods for finding maximum likelihood estimates are available, and some of those shall be discussed here, focusing on the duality between smooths and random effects. Therefore, MLE in GAMs and GLMMs is first treated separately, followed by MLE in GAMMs.

Maximum likelihood estimation in GAMs

Estimation for fixed smoothing parameters For the moment, suppose that the smoothing parameters of the model, denoted as a vector $\boldsymbol{\rho}$, are fixed. According to Wood (2017), the model parameters $\boldsymbol{\theta} = (\boldsymbol{\beta}^\top, \boldsymbol{\gamma}^\top)^\top$ in a GAM can be estimated by maximizing the penalized log-likelihood

$$\ell_{\text{pen}}(\boldsymbol{\theta} \mid \boldsymbol{\rho}) = \sum_{i=1}^n \log f_{\boldsymbol{\theta}}(y_i) - \frac{1}{2} \sum_{s \in S} \rho_s \boldsymbol{\theta}^\top \mathbf{K}_s \boldsymbol{\theta}, \quad (1.9)$$

where $f_{\boldsymbol{\theta}}(\cdot)$ is the density of an exponential family distributed random variable parameterized by $\boldsymbol{\theta}$. As before, S is an index set which refers to all nonlinearly modeled covariates (or interaction thereof). The matrix \mathbf{K}_s is the penalty matrix of the s -th smoother, which is constructed as shown in Section 1.2.2, but here augmented with zeros such that it fits the dimension of $\boldsymbol{\theta}$. Maximization of (1.9) with respect to $\boldsymbol{\theta}$ conditional on $\boldsymbol{\rho}$ can be achieved via penalized iteratively re-weighted least squares (PIRLS, Wood, 2017). If $\phi \neq 1$, the dispersion parameter can consistently be estimated by employing a scaled Pearson statistic.

Estimation of the smoothing parameters In contrast to optimizing the penalized log-likelihood (1.9), estimation of the smoothing parameters is not straightforward. A possible approach is a two-stage estimation procedure in which the smoothing parameters (outer iteration) and the model (inner iteration) are alternately estimated. In the outer iteration, a model criterion needs to be optimized for the smoothing parameters. One popular option is the minimization of the generalized cross-validation (GCV) score (Craven and Wahba, 1978), which, however, suffers from the curse of dimensionality if many smoothing parameters need to be estimated. This issue is resolved by Wood (2004) who regards the GCV score as a twice-differentiable function of the (log-)smoothing parameters and makes use of Newton’s method to minimize the criterion.

A similar idea was initially developed by Schall (1991). The embedding into the GAM context was proposed by Wood and Fasiolo (2017) which is known as the generalized Fellner-Schall method. This is a comparably simple but very efficient method to estimate smoothing parameters in generalized regression models which can also be applied beyond exponential family models. The idea is to maximize the log-restricted marginal likelihood (Wood, 2011) of the model with respect to the smoothing parameters for a given estimate $\hat{\boldsymbol{\theta}}$ which is again accomplished by computing the respective derivatives with respect to the smoothing parameters. The update proceeds as follows. If $\rho_s, s \in S$ denote the current estimates of the smoothing parameters and $\hat{\boldsymbol{\theta}}_\rho$ is the argument which maximizes (1.9) for given $\boldsymbol{\rho}$, then the single update proceeds as follows

$$\rho_s^{(\text{new})} = \rho_s \frac{\text{tr}(\mathbf{K}_\rho^- \mathbf{K}_s) - \text{tr}(\hat{\mathbf{V}}_\rho \mathbf{K}_s)}{\hat{\boldsymbol{\theta}}_\rho^\top \mathbf{K}_s \hat{\boldsymbol{\theta}}_\rho}. \quad (1.10)$$

Here, $\text{tr}(\cdot)$ denotes the trace operator and \mathbf{K}_ρ^- is a generalized inverse of $\mathbf{K}_\rho = \sum_{s \in S} \rho_s \mathbf{K}_s$. The matrix $\hat{\mathbf{V}}_\rho = \mathbf{V}(\hat{\boldsymbol{\theta}}_\rho)$ is the expected Hessian of the negative log-likelihood evaluated at the maximum likelihood estimate $\hat{\boldsymbol{\theta}}_\rho$. In exponential family generalized additive models, the expected Hessian is given by $\mathbf{V}(\boldsymbol{\theta}) = (\mathbf{X}^\top \mathbf{W}(\boldsymbol{\theta}) \mathbf{X} + \mathbf{K}_\rho)^{-1}$. Here, \mathbf{X} denotes the design matrix of the model and $\mathbf{W}(\boldsymbol{\theta})$ is a diagonal weight matrix which appears in the PIRLS algorithm. For example, in a log-linear Poisson model the diagonal elements are given by $\mathbf{W}(\boldsymbol{\theta})_{ii} = \exp(\eta_i)$. In a non-exponential family setting, the expected Hessian can be replaced by the observed Hessian. However, this does not ensure the positivity of the smoothing parameter updates anymore which requires the matrix $\hat{\mathbf{V}}_\rho$ to be positive-definite.

Maximum likelihood estimation in GLMMs

Now, estimation of a generalized linear mixed model (GLMM) shall be discussed which is a GAMM without smooth covariate effects. The marginal likelihood of the model results from integrating out the random effects from the joint density of the data and random effects $f_{\boldsymbol{\theta}, \boldsymbol{\vartheta}}(\mathbf{y}, \mathbf{u}) = f_{\boldsymbol{\theta}}(\mathbf{y} | \mathbf{u}) f_{\boldsymbol{\vartheta}}(\mathbf{u})$, where $f_{\boldsymbol{\vartheta}}(\cdot)$ is the assumed density of the random effects with parameters $\boldsymbol{\vartheta}$ (Fahrmeir and Tutz, 2013). In particular,

$$\ell_{\text{marg}}(\boldsymbol{\theta}, \boldsymbol{\vartheta}) = \sum_{i=1}^n \log \int \prod_{t=1}^T f_{\boldsymbol{\theta}}(y_{it} | \mathbf{u}_i) f_{\boldsymbol{\vartheta}}(\mathbf{u}_i) d\mathbf{u}_i. \quad (1.11)$$

However, the integrals that appear in (1.11) can seldom be solved analytically as in the case of a linear mixed model with Gaussian random effects or for gamma shared frailty time-to-event models (see Section 1.4.1). A standard tool to derive explicit formulas of the marginal

likelihood is applying a Laplace approximation (Davison, 2003 and Shun and McCullagh, 1995 for asymptotic properties) to the integral in (1.11). Taking the logarithm of this approximation and ignoring constant terms, an approximation of the marginal log-likelihood thus is

$$\ell_{\text{marg}}(\boldsymbol{\theta}, \boldsymbol{\vartheta}) \stackrel{\text{Laplace}}{\approx} \log f_{\boldsymbol{\theta}}(\mathbf{y} \mid \hat{\mathbf{u}}) - \frac{1}{2} \hat{\mathbf{u}}^{\top} \tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\vartheta}}^{-1} \hat{\mathbf{u}} - \frac{1}{2} \log |\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\vartheta}}| - \frac{1}{2} \log \left| \frac{1}{\phi} \mathbf{Z}^{\top} \mathbf{W} \mathbf{Z} + \tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\vartheta}}^{-1} \right|,$$

where $\hat{\mathbf{u}}$ is the maximizer of the joint likelihood for fixed $\boldsymbol{\theta}$ and $\boldsymbol{\vartheta}$ (Wood, 2017).

Alternative approaches to computing the marginal log-likelihood involve numerical techniques such as adaptive Gauss–Hermite quadrature (Hartzel et al., 2001), Monte Carlo integration (Friel and Pettitt, 2008) or the employment of an EM algorithm (Dempster et al., 1977). The former method is only feasible if the dimension of the random effects is low, where the Monte Carlo integration techniques can remedy this problem since the numerical effort rises linearly with the dimension of the random effects rather than exponentially. The latter method views \mathbf{y} and \mathbf{u} as observed and missing data, respectively, and iteratively maximizes the log-likelihood with the current expected values of the missing data.

Having obtained estimates for $\boldsymbol{\theta}$ and $\boldsymbol{\vartheta}$, it remains to estimate the random effects \mathbf{u}_i in order to obtain fitted values of the conditional means $\mathbb{E}(Y_{it} \mid \mathbf{x}_{it}, \mathbf{u}_i)$. An estimate $\hat{\mathbf{u}}_i$ is the posterior mean $\mathbb{E}(\mathbf{u}_i \mid \mathbf{y}_i; \hat{\boldsymbol{\theta}}, \hat{\boldsymbol{\vartheta}})$. This involves again the computation of an intractable integral which can be carried out by applying one of the methods discussed above.

Maximum likelihood estimation in GAMMs

Treating random effects as smooths If both, smooth components and random effects are present in the model, the same estimation methods as above can be employed if the r random effects per subject are considered to be independent. These assumptions lead to the following decomposition of the joint density of the data and random effects

$$f_{\boldsymbol{\theta}}(\mathbf{y}, \mathbf{u}) = f_{\boldsymbol{\theta}}(\mathbf{y} \mid \mathbf{u}) \cdot f_{\boldsymbol{\vartheta}}(\mathbf{u}) = \left(\prod_{i=1}^n \prod_{t=1}^T f_{\boldsymbol{\theta}}(y_{it} \mid \mathbf{u}_i) \right) \cdot \left(\prod_{i=1}^n \prod_{j=1}^r f_{\boldsymbol{\vartheta}}(u_{ir}) \right). \quad (1.12)$$

Similar arguments as in Breslow and Clayton (1993) for penalized quasi-likelihood maximization lead for Gaussian random effects to the penalized log-likelihood

$$\ell_{\text{pen}}(\boldsymbol{\theta}, \mathbf{u} \mid \boldsymbol{\rho}, \boldsymbol{\vartheta}) = \sum_{i=1}^n \sum_{t=1}^T \log f_{\boldsymbol{\theta}}(y_{it} \mid \mathbf{u}_i) - \frac{1}{2} \sum_{s \in S} \rho_s \boldsymbol{\theta}^{\top} \mathbf{K}_s \boldsymbol{\theta} - \frac{1}{2} \sum_{j=1}^r \frac{1}{\sigma_j^2} \mathbf{u}_j^{\top} \mathbf{u}_j, \quad (1.13)$$

where $\mathbf{u}_j = (u_{1j}, \dots, u_{nj})^{\top}$. Denoting $\rho_j = \frac{1}{\sigma_j^2}$ and setting \mathbf{K}_j to the identity matrix, it can be seen that the second penalty in (1.13) can be put into the same shape as the first penalty. Therefore, treating independent random effects as smooths is equivalent to imposing a Ridge penalty on each random effect.

Treating smooths as random effects An alternative to the approach discussed in the previous paragraph is to treat smooth terms in a GAMM as random effects in the framework of GLMMs. For reasons of simplicity, only one-dimensional smooths (or multi-dimensional smooths with a single smoothing parameter) are discussed here. Writing tensor-product smooths with the penalty having multiple smoothing parameters as in (1.8) in a mixed model representation is

much more involved and not discussed here. For details, see Wood et al. (2013).

The basic idea is to specify a prior distribution on the model parameters $\boldsymbol{\gamma}$ of a smooth term, i.e. the problem is seen from the Bayesian perspective. Due to the already established connection to Gaussian random effects, the parameters $\boldsymbol{\gamma}$ of a smoother are equipped with an exponential prior of the form

$$f(\boldsymbol{\gamma}) \propto \exp(-\rho \boldsymbol{\gamma}^\top \mathbf{K}_r \boldsymbol{\gamma}).$$

This results in an improper Gaussian prior $\boldsymbol{\gamma} \sim \mathcal{N}(\mathbf{0}, \mathbf{K}_r^-/\rho)$, since \mathbf{K}_r is rank deficient with $\text{rk}(\mathbf{K}_r) = d - r$. After reparameterization as proposed by Wood et al. (2013), it follows that $f(x_i) = \mathbf{X}_i^* \boldsymbol{\beta}^* + \mathbf{Z}_i^* \mathbf{u}^*$ with fixed effects $\boldsymbol{\beta}^* \in \mathbb{R}^d$ and $\mathbf{u}^* \sim \mathcal{N}_{d-r}(\mathbf{0}, \mathbf{I}/\rho)$, i.e. $f(\cdot)$ has a mixed model representation. Conducting this procedure for all smooths $f_s(\cdot)$, $s \in S$ in the model leads to a large GLMM which can be estimated as shown above.

Approximate EM Algorithm As shown in the last two paragraphs, both directions of the equivalence of smooths and random effects can be exploited, which standard software uses to fit exponential family GAMMs. It is also possible to exploit this connection implicitly and employ an EM-type algorithm that is easy to implement and works if the response Y does not belong to the exponential family of distributions. Moreover, in contrast to 1.12, the random effects do not need to be assumed to be independent. This approach is chosen for the generalized regression model in Chapter 2 and will thus be sketched in the following.

First, assume that the smoothing parameters $\boldsymbol{\rho}$ and the covariance parameters $\boldsymbol{\vartheta}$ are fixed. If $\boldsymbol{\theta}$ is not considered to be a fixed parameter vector any more but, however, now being equipped with a flat prior density, it follows from Bayes's theorem that $f_{\boldsymbol{\theta}}(\mathbf{y} \mid \mathbf{u}) \cdot f_{\boldsymbol{\vartheta}}(\mathbf{u})$ from above is proportional to the joint posterior density $f(\boldsymbol{\theta}, \mathbf{u} \mid \mathbf{y}, \boldsymbol{\vartheta})$. Thus, following Fahrmeir and Tutz (2013) and assuming Gaussian random effects, maximization of the latter with respect to $\boldsymbol{\zeta} = (\boldsymbol{\theta}^\top, \mathbf{u}^\top)^\top$ is equivalent to maximizing the penalized log-likelihood (while also adding penalties for smooth terms)

$$\ell_{\text{pen}}(\boldsymbol{\zeta}) = \sum_{i=1}^n \sum_{t=1}^T \log f_{\boldsymbol{\theta}}(y_{it} \mid \mathbf{u}_i) - \frac{1}{2} \sum_{s \in S} \rho_j \boldsymbol{\theta}^\top \mathbf{K}_s \boldsymbol{\theta} - \frac{1}{2} \mathbf{u}_i^\top \boldsymbol{\Sigma}_{\boldsymbol{\vartheta}}^{-1} \mathbf{u}_i. \quad (1.14)$$

Note that (1.14) now involves two penalties, one for the smooth terms and one for the random effects. If the response does not fit in the exponential family framework, other numerical optimization methods than PIRLS algorithm might be required for the maximization of (1.14). Possible alternatives are quasi-Newton methods such as the BFGS algorithm, which does not require the computation of second derivatives (Wright and Nocedal, 1999).

Now, let us assume that $\hat{\boldsymbol{\zeta}} = \arg\max_{\boldsymbol{\zeta}} \ell_{\text{pen}}(\boldsymbol{\zeta})$ has been obtained when having assumed fixed values for $\boldsymbol{\rho}$ and $\boldsymbol{\vartheta}$. A simple update of the smoothing parameters can be performed via the Fellner-Schall method (1.10). A method to derive estimates for the covariance parameters $\boldsymbol{\vartheta}$ was proposed by Laird and Ware (1982) for normally distributed responses and was extended to the exponential family setting by Fahrmeir et al. (2007). The basic idea is to maximize the marginal log-likelihood $\ell_{\text{marg}}(\boldsymbol{\vartheta})$ which is obtained by integrating the joint likelihood of the data and the random effects with respect to the random effects \mathbf{u}_i , $i = 1, \dots, n$ and $\boldsymbol{\theta}$. This, in turn, is generally a sophisticated task. Hence, maximization is performed indirectly via an EM

algorithm. The M-step consists of maximizing the conditional expectation (E-step)

$$M(\boldsymbol{\vartheta} \mid \boldsymbol{\vartheta}^{(k)}) = \mathbb{E}(\log f_{\boldsymbol{\vartheta}}(\boldsymbol{\zeta}) \mid \mathbf{y}; \boldsymbol{\vartheta}^{(k)})$$

with respect to $\boldsymbol{\vartheta}$, where $\boldsymbol{\vartheta}^{(k)}$ denotes the estimate for $\boldsymbol{\vartheta}$ from the previous cycle of the EM algorithm. It can be shown that in case of Gaussian prior for the random effects, the resulting update of the covariance matrix is given by

$$\boldsymbol{\Sigma}(\boldsymbol{\vartheta}^{(k+1)}) = \frac{1}{n} \sum_{i=1}^n \left[\text{Cov}(\mathbf{u}_i \mid \mathbf{y}_i; \boldsymbol{\vartheta}^{(k)}) + \mathbb{E}(\mathbf{u}_i \mid \mathbf{y}_i; \boldsymbol{\vartheta}^{(k)}) \mathbb{E}(\mathbf{u}_i \mid \mathbf{y}_i; \boldsymbol{\vartheta}^{(k)})^\top \right].$$

Maximization of (1.14) delivers posterior modes but not the posterior means $\mathbb{E}(\mathbf{u}_i \mid \mathbf{y}_i; \boldsymbol{\Sigma}(\boldsymbol{\vartheta}^{(k)}))$. In general, the posterior mode is different from the posterior mean since the posterior distribution of \mathbf{u}_i is not normal for non-gaussian responses. For the same reason, the posterior covariance matrix is not exactly equal to $\mathbf{V}(\hat{\boldsymbol{\zeta}})$, the hessian of the negative log-likelihood evaluated at $\hat{\boldsymbol{\zeta}}$. However, these quantities still serve as an approximation which leads to the update

$$\boldsymbol{\Sigma}(\boldsymbol{\vartheta}^{(k+1)}) = \frac{1}{n} \sum_{i=1}^n \left[\mathbf{V}(\hat{\boldsymbol{\zeta}})_{ii}^{(k)} + \hat{\mathbf{u}}_i^{(k)} (\hat{\mathbf{u}}_i^{(k)})^\top \right]. \quad (1.15)$$

Since (1.15) is not based on the posterior mean and the posterior covariances, but approximations thereof, Fahrmeir and Tutz (2013) denote this modification of the EM-algorithm as ‘‘EM-type’’ algorithm or ‘‘approximate EM algorithm’’.

Uncertainty

One of the main goals of statistical modeling, which was phrased in Section 1.1 was to capture the stochastic structure of the model. Usually, two types of uncertainties are distinguished, aleatoric and epistemic uncertainty (Der Kiureghian and Ditlevsen, 2009). The former source of uncertainty mirrors the intrinsic randomness of the data. Thus, this type of uncertainty can not be avoided but appropriately handled by assuming a probability distribution of the response. Epistemic uncertainty is introduced by the lack of knowledge, which can either be caused by the lack of data or an inappropriate model formulation (‘‘all models are wrong, but some models are useful’’, Box, 1979).

In particular, it is desirable to quantify the uncertainty of the maximum likelihood estimates $\hat{\boldsymbol{\theta}}$ which allows to deduce the uncertainty of the estimated means $\hat{\mu}_{it}$ which are random variables. Asymptotic properties of the MLE in GLMs have been derived by Fahrmeir and Kaufmann (1985). The results can be generalized to the GAM case such that $\hat{\boldsymbol{\theta}} \mid \boldsymbol{\rho} \sim \mathcal{N}(\boldsymbol{\theta}, \mathbf{V})$ holds asymptotically, where \mathbf{V} is the inverse of the Hessian of the negative log-likelihood evaluated at $\boldsymbol{\theta}$. Viewing the estimation problem from a Bayesian perspective, the large sample approximation yields $\boldsymbol{\theta} \mid \mathbf{y}, \boldsymbol{\rho} \sim \mathcal{N}(\hat{\boldsymbol{\theta}}, \hat{\mathbf{V}})$ (Wood, 2006). Therefore, Bayesian confidence intervals for nonlinear functions of the parameters, e.g. for smooth covariate effects, can be obtained via simulation from the posterior distribution of $\boldsymbol{\theta}$. However, note that this quantification of the uncertainty neglects additional variability that is introduced by the estimation of the smoothing parameters. This additional uncertainty is discussed by Marra and Wood (2012) and Wood et al. (2016).

1.2.4. Extension to non-exponential families

So far, it has been assumed that the response variable Y belongs to a univariate exponential family with a density that can be factorized according to (1.1), which is advantageous as the whole GAMM theory can be put in one unified framework. However, either the exponential family of distributions might be too restrictive for a specific application, or it is desired to not only model the mean of the response with covariates. Therefore, two extensions of generalized additive mixed models shall be presented here.

GAMM for non-exponential families In Chapter 2, a regression model is constructed where the response variable is assumed to follow a Skellam distribution. This specific distribution arises from the difference of two independent Poisson distributed random variables (Skellam, 1946). In particular, $D = X - Y \sim \text{Skellam}(\mu_1, \mu_2)$ if $X \sim \text{Poi}(\mu_1)$ and $Y \sim \text{Poi}(\mu_2)$ are independent. The density of D with respect to the counting measure is given by

$$f(k; \mu_1, \mu_2) = e^{-(\mu_1 + \mu_2)} \left(\frac{\mu_1}{\mu_2} \right)^k I_k(2\sqrt{\mu_1 \mu_2})$$

for $k \in \mathbb{Z}$ where $I_k(\cdot)$ are Bessel functions of the first kind (Abramowitz and Stegun, 1964). From the construction of D , it immediately follows that $\mathbb{E}(D) = \mu_1 - \mu_2$. As one can see, the density of D is already parameterized in terms of its expected value, now with two parameters to be modeled. A model formulation without random effects is $D_i \sim \text{Skellam}(\mu_{i1}, \mu_{i2})$ independent given covariates \mathbf{x}_i and $g(\mu_{ij}) = \eta_{ij}$ for $j = 1, 2$. Such a model can be used to identify the means μ_{i1} and μ_{i2} related to the observed differences D_i .

GAMLSS The Skellam modeling approach formulated above allows modeling both parameters of the response D with covariates. Regression models which generally allow modeling the whole distribution of a univariate response Y are generalized additive models for location, scale and shape (GAMLSS, Rigby and Stasinopoulos, 2005). As the name of the model class suggests, the density of the response Y is usually parameterized in terms of a location parameter (often the mean), a scale parameter (related to the variance) and shape parameters (related to skewness and kurtosis). All the flavors of statistical modeling formulated until now can also be used in the context of a GAMLSS. A software implementation that allows fitting dozens of different distributions is proposed by Stasinopoulos et al. (2007). GAMLSS are not part of Chapters 2 - 6. However, they should be presented at the end of this section as an alternative if the statistical modeling approach when employing GAMMs showed some issues. For example, the overdispersion parameter ϕ is assumed to be constant in the negative binomial GAMM framework. When employing a GAMLSS, this parameter that affects the response's variance could also be parameterized by covariates.

1.3. Geometric networks

“A knowledge of statistics is like a knowledge of foreign languages or of algebra; it may prove of use at any time under any circumstances.”

— Sir Arthur Lyon Bowley (* 1869, † 1957),
British statistician and economist

One explanation for the term “network” is “a chain or system of interconnected immaterial things”.² Kolaczyk (2009) reformulates this as a network being “simply [...] a collection of elements and their inter-relations”. This collection can be formalized in terms of a (network) graph $G = (V, E)$, a mathematical structure known from graph theory where $V = \{v_1, \dots, v_W\}$ is a set of W vertices (“elements”) and $E = \{e_1, \dots, e_M\} \subset V \times V$ a set of M edges (“inter-relations”). The statistical modeling of network-related data is a wide field, and a typical example is outlined by Hoff et al. (2002) where the aim is to draw inference about an unobserved social space. This view on networks is from a rather abstract point of view. On the other hand, the Oxford English Dictionary defines “networks” as “any netlike or complex system or collection of interrelated things, as topographical features, lines of transportation...”. This definition assigns more spatial properties to a network. Three of the contributing articles in this dissertation revolve around such networks. Therefore, they will be introduced formally in the following.

1.3.1. Definition

A geometric (or spatial) network \mathbf{G} embedded in a Euclidean space of dimension $q \in \mathbb{N}$ with $q \geq 2$ is defined as

$$\mathbf{G} = \bigcup_{m=1}^M \mathbf{e}_m \subset \mathbb{R}^q,$$

where $\mathcal{E} = \{e_1, \dots, e_M\}$ is a set of network segments. Each \mathbf{e}_m can be understood as the image set of a parametric curve $\nu_m : [a_m, b_m] \rightarrow \mathbb{R}^q$, i.e. $\mathbf{e}_m = \nu_m([a_m, b_m])$. The length $d_m = |\mathbf{e}_m|$ of the m -th network segment is defined to be the integral along the curve ν_m . In practice, \mathbf{e}_m can be approximated by an alignment of straight line segments such that

$$d_m = \lim_{N \rightarrow \infty} \sum_{i=1}^N \|\nu_m(t_i) - \nu_m(t_{i-1})\|_q,$$

with $a = t_0 < t_1 < \dots < t_N = b$ and with $\|\cdot\|_q$ denoting the Euclidean distance. These lengths imply the shortest path distance measure $d_{\mathbf{G}} : \mathbf{G} \times \mathbf{G} \rightarrow [0, \infty)$ on \mathbf{G} . Note that $d_{\mathbf{G}}$ is seldom a metric since the symmetry is usually not fulfilled if the curves are additionally equipped with a direction. Moreover, $\mathcal{V} = \{\mathbf{v}_1, \dots, \mathbf{v}_W\}$ is a set of vertices (or nodes) which are defined as the unique set of endpoints of segments \mathcal{E} , i.e. $\mathcal{V} = \{\nu_m(a_m), \nu_m(b_m) \mid m = 1 \dots, M\}$. A more detailed introduction into the notion of geometric networks is given in Chapter 3. Figure 1.3 shows a network of highways in the southern part of Montgomery County, Maryland. Here,

²<https://www.oed.com/>

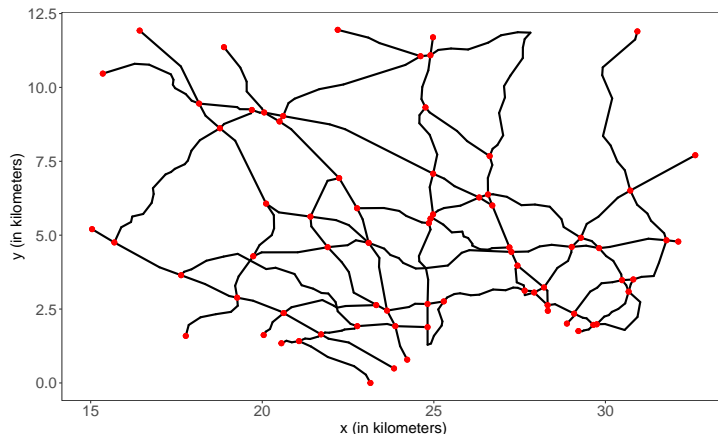


Figure 1.3.: A network of highways in Montgomery County, Maryland. The red dots represent the vertices of the network.

the vertices are visualized through red dots, and each of them corresponds to an intersection of three or more roads or the terminus of a street, respectively.

1.3.2. Estimation of latent network flows

A frequent problem in the analysis of networks is the estimation of latent network flows. An example is the estimation of unobserved traffic in infrastructure networks such as public transportation networks. Usually, the problem's formulation is that one observes an m -dimensional process \mathbf{y} which is used to estimate an n -dimensional process of interest \mathbf{z} with $m \ll n$, where the latter process corresponds to the network flows of interest (Airoldi and Blocker, 2013). Hansen (1998) generally qualifies these kinds of problems as ill-posed problems since the solution is usually not unique. They are also classified as inverse problems since one needs to determine the input from the output of a system. When tackling inverse problems, often regularization methods are used (Zhang et al., 2003).

Chapter 2 is concerned with an inverse and ill-posed latent network flow estimation problem. Here, an m -dimensional process $\mathbf{y} = (\mathbf{y}_t)_{t=1, \dots, T}$ is observed which is used to estimate the $n = m^2$ -dimensional network flows $\mathbf{z} = (\mathbf{z}_t)_{t=1, \dots, T}$. Thus, the process \mathbf{y} is defined on the nodes of the network and \mathbf{z} is a dyadic process. In the specific application of Chapter 2, $\mathbf{c}_t \in \mathbb{N}_0^m$ are the station feeds in a bike-sharing system at time point $t = 0, \dots, T$ which imply the differences of station feeds $\mathbf{y}_t = \mathbf{c}_t - \mathbf{c}_{t-1} \in \mathbb{Z}^n$ for $t = 1, \dots, T$. This observed process is used to estimate the latent network flows $\mathbf{z}_t \in \mathbb{N}_0^n$, which are defined as the number of bikes leaving station $i \in \{1, \dots, m\}$ in the interval $[t-1, t)$ and arriving, possibly after time point t , at station $j \in \{1, \dots, m\}$. Note that $n = m^2$ implies that trips back and forth from the same station, so-called loops, are allowed as well. The path between bike stations is implied by a geometric network \mathbf{G} which in this application is a street network. The distance between bike stations i and j is the shortest path distance $d_{\mathbf{G}}(\mathbf{s}_i, \mathbf{s}_j)$ between their respective locations \mathbf{s}_i and \mathbf{s}_j which is one of the covariates in the model.

Figure 1.4 shows the station feeds \mathbf{c}_{t-1} (left panel) and \mathbf{c}_t (right panel) in a simplified bike-sharing network with five bike stations at two consecutive time points. When assuming the incoming and the outgoing flows to/from station i being independently distributed Poisson random variables, their differences $y_{i,t-1} = c_{i,t-1} - c_{i,t}$ follow a Skellam distribution. However,

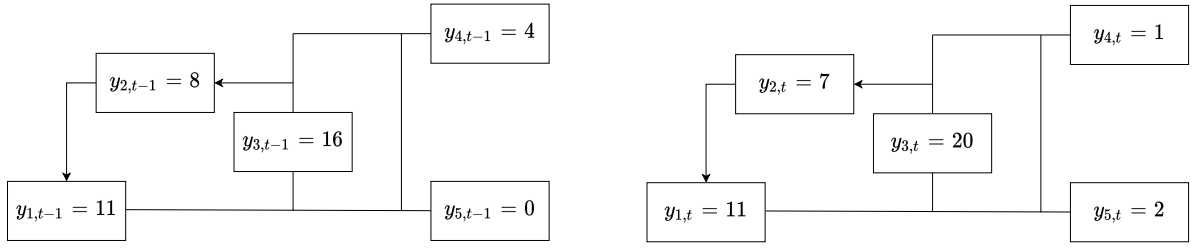


Figure 1.4.: Illustration of the ill-posed bike-sharing problem. The left panel shows stations feeds of the bike sharing network at time point $t - 1$, the right panel at time point t . The observed 5-dimensional vectors $\mathbf{y}_t = \mathbf{c}_t - \mathbf{c}_{t-1}$ are used to estimate the 25-dimensional vectors \mathbf{z}_t . The arrows represent one-way routes, i.e. the network is directed.

this result only holds approximately since the bike station’s capacities are finite in reality. To respect this, a truncated Skellam distribution can be used as motivated by Ntzoufras et al. (2019). In Figure 1.4, the observed differences are $\mathbf{y}_t = (0, -1, 4, 2, -3)^\top$. When observing these differences over a longer period together with covariates $\mathbf{x} = (\mathbf{x}_t)_{t=1, \dots, T}$, a Skellam regression model (see Section 1.2.4) can be employed in order to estimate the expected flows $\mathbb{E}(\mathbf{z}_t)$ in the bike-sharing network. Such a model is elaborated in detail in Chapter 2 and is essentially a GAMM where the response variable does not belong to the exponential family of distributions.

1.3.3. Intensity estimation of point processes

For general spaces $\mathcal{S} \subset \mathbb{R}^q$, a spatial point process \mathcal{X} is a random countable subset of \mathcal{S} (Moller and Waagepetersen, 2003). In Chapter 3, the special case $\mathcal{S} = \mathcal{G}$ is considered, i.e. point processes on a geometric network as defined above. A realization $\mathbf{x} = \mathcal{X}(\omega) \in \mathcal{G}$ of the point process \mathcal{X} is denoted as a point pattern. Spatial point processes can be characterized in terms of a nonnegative intensity function $\varphi_{\mathcal{X}} : \mathcal{G} \rightarrow [0, \infty)$ with $\varphi_{\mathcal{X}}(\mathbf{u})$ denoting the expected number of points per unit length of the network, in the vicinity of a point $\mathbf{u} \in \mathcal{G}$ (Baddeley et al., 2015). Therefore, the expected number of points which fall in a subset $\mathcal{B} \subset \mathcal{G}$ is given by

$$\mathbb{E}_{\mathcal{X}}(\mathcal{B}) = \int_{\mathcal{B}} \varphi_{\mathcal{X}}(\mathbf{u}) \, d\mathbf{u} = \sum_{m=1}^M \int_{e_m \cap \mathcal{B}} \varphi_{\mathcal{X}}(\mathbf{u}) \, d\mathbf{u}$$

Chapter 3 is devoted to the estimation of the intensity function $\varphi_{\mathcal{X}}(\cdot)$ of a point process \mathcal{X} on a geometric network \mathcal{G} where in Chapter 3 the focus is on the deployment of the methodology and the appendix of Chapter 3 treats the implementation of the model in \mathbf{R} .

Intensity estimation with kernel based methods Within the \mathbf{R} package `spatstat` (Baddeley et al., 2015) various kernel smoothing approaches are implemented that estimate the intensity of a point process on a linear network, i.e. a geometric network where the connections between two vertices are straight lines. In a seminal paper, Okabe et al. (2009) introduced the “equal-split continuous” kernel based method. In general, they define a kernel estimator (Silverman, 1986)

of the density $f(\mathbf{u}) \propto \lambda(\mathbf{u})$ at a point $u \in \mathbf{G}$ by

$$\hat{f}(\mathbf{u}) = \frac{1}{n} \sum_{i=1}^n c_{\mathbf{G}}(\mathbf{u}, \mathbf{x}_i) k_h(d_{\mathbf{G}}(\mathbf{u}, \mathbf{x}_i)),$$

where $k_h(\cdot)$ is a continuous, nonnegative and unimodal kernel function with modal point \mathbf{x}_i and kernel bandwidth h . An example is the Epanechnikov kernel (Epanechnikov, 1969). The factor $c_{\mathbf{G}}(\mathbf{u}, \mathbf{x}_i)$ is a correction factor to account for the geometry of the network which results in unbiased estimates if the true intensity is uniform. Without going into the technical details, the idea behind this correction factor is to split the kernel mass at vertices equally across the line segments. For more details see also Okabe and Sugihara (2012).

Based on the considerations for correcting edge effects and fastening the convergence of kernel based estimators in one dimension (Botev et al., 2010), kernel based estimation of the intensity on linear networks was further developed by McSwiggan et al. (2017). They replaced the kernel function multiplied by the correction factor from above through a heat kernel with bandwidth $h = \sigma^2$ (Kostyrykin et al., 2007) that has the form

$$k_{\sigma^2}(\mathbf{u}, \mathbf{x}_i) = \sum_{\pi \in \Pi} a(\pi) \varphi_{\sigma^2}(\ell(\pi)). \quad (1.16)$$

Here, π is a path from \mathbf{x}_i to \mathbf{u} on \mathbf{G} , $\ell(\pi)$ is the length of this path, φ_{σ^2} is the density of a $\mathcal{N}(0, \sigma^2)$ random variable and $a(\pi)$ is product involving all vertices that the path π passes. Note that the set of paths Π is infinite since there is no restriction on the complexity of a path. For example, π might pass a point $\mathbf{u} \in \mathbf{G}$ several times. Nonetheless, the sum (1.16) converges and more generally, the equal-split continuous estimator is asymptotically equivalent to the heat kernel estimator.

Penalized splines on a geometric network Kernel-based methods for the intensity estimation of a point process on a network geometry adopt the drawbacks from kernel smoothing in Euclidean spaces, such as the non-consistency of the estimators near the boundaries of the support (Karunamuni and Alberts, 2005). Beyond this, it is not straightforward to estimate the intensity when covariates are available. A method that can do so and makes use of GAM theory is presented in Chapter 3. The cornerstone of this new method is to define penalized splines on a geometric network which is briefly motivated in the following.

On every curve e_m with endpoints, say \mathbf{v}_i and \mathbf{v}_j , an equidistant sequence of I_m knots $\mathbf{v}_i = \boldsymbol{\tau}_{m,1}, \dots, \boldsymbol{\tau}_{m,I_m} = \mathbf{v}_j$ is defined with δ_m denoting the knot distance on e_m . This is exemplified for a small network in the left panel of Figure 1.5. The knots can be used to construct linear B-splines on the geometric network which is described in the following. A mathematical definition of those B-splines can be found in Chapter 3.

In the Euclidean case, linear B-splines are supported between three adjacent knots. In the same manner, $J_m = I_m - 2$ linear B-splines $B_{m,k}(\cdot)$ are defined on the m -th curve which satisfy $B_{m,k}(\boldsymbol{\tau}_{m,k}) = 0$, $B_{m,k}(\boldsymbol{\tau}_{m,k+1}) = 1$ and $B_{m,k}(\boldsymbol{\tau}_{m,k+2}) = 0$ for $k = 1, \dots, J_m$, i.e. the mode of the B-spline $B_{m,k}$ equals $\boldsymbol{\tau}_{m,k+1}$. To supplement these B-splines to a basis on \mathbf{G} , another W linear B-splines $B_{(i)}$ are constructed around each vertex. Therefore, $B_{(i)}(\mathbf{v}_i) = 1$, i.e. the mode of $B_{(i)}$ equals \mathbf{v}_i , and linear decrease towards the adjacent knots. In the setting of Figure 1.5, $B_{(4)}(\mathbf{v}_4) = 1$ and $B_{(4)}(\boldsymbol{\tau}_{24}) = B_{(4)}(\boldsymbol{\tau}_{33}) = B_{(4)}(\boldsymbol{\tau}_{42}) = B_{(4)}(\boldsymbol{\tau}_{62}) = 0$. The right panel of Figure 1.5 shows the location of the B-splines which are defined by the knots in the left panel, where the vertex specific B-splines are marked as blue squares.

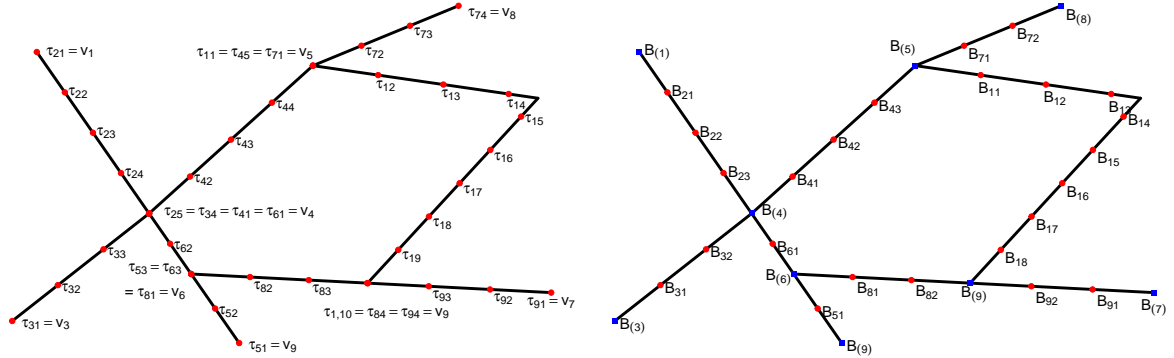


Figure 1.5.: Left panel: Allocation of knots on a small geometric network. Right panel: Position of the modes of the linear B-splines defined by the knots.

As in the Euclidean setting, every B-spline has a basis coefficient while, however, the number in the geometric network setting is much higher, which becomes evident when considering the small network in Figure 1.5. Therefore, penalization becomes even more relevant. The construction of the penalty is similar as proposed by Eilers and Marx (1996) for the real numbers, where neighboring B-spline coefficients are penalized. A penalty of order one is straightforward, i.e. all pairs of neighbored coefficients are penalized. Switching notation and indexing B-splines and their coefficients with $j = 1, \dots, J$, a penalty of order two has the form

$$P(\rho; \gamma) = \rho \sum_{\mathcal{D}_2} (\gamma_i - 2\gamma_k + \gamma_j)^2 = \rho (\mathbf{D}_2 \gamma)^\top (\mathbf{D}_2 \gamma) = \rho \gamma^\top \mathbf{K}_2 \gamma,$$

where \mathcal{D}_2 is the set of all triples of indices that are neighbors of order two. Thus, a penalty can also be represented as a quadratic form such as in the Euclidean case. As an example, the coefficients of the triple $(B_{3,2}, B_{(4)}, B_{6,1}) \in \mathcal{D}_2$.

Intensity estimation with penalized splines The above defined penalized B-splines can now be used to estimate the intensity of a point process on a geometric network. The observed point pattern \mathbf{x} is binned on \mathbf{G} which is visualized in Figure 1.6. Let $y_{m,k} \in \mathbb{N}_0$ be the observed count of points in the k -th bin of the m -th curve which is represented by the mid point $\mathbf{z}_{m,k}$ and let h_m denote the curve-specific bin width. The general assumption of the model is that $Y_{m,k} \mid \mathbf{z}_{m,k} \stackrel{\text{indep.}}{\sim} \text{Poi}(\lambda_{m,k})$, where $\lambda_{m,k}$ is approximated through the log-linear model

$$\lambda_{m,k} = \exp(\nu_{\mathcal{X}}(\mathbf{z}_{m,k}) + \eta_{m,k} + \log h_m) \quad (1.17)$$

Here, h_m serves as offset to ensure the appropriate scaling and $\nu_{\mathcal{X}}(\mathbf{z}_{m,k}) = \sum_{j=1}^J \gamma_j B_j(\mathbf{z}_{m,k})$ is a B-spline basis representation of the log-baseline intensity. The linear predictor $\eta_{m,k}$ has the same structure as in (1.2) and allows to estimate effects of covariates on the intensity.

It can be seen that model 1.17 is, in fact, a generalized additive (Poisson) model. Therefore, inference can be drawn by making use of the concepts which were already outlined in Section 1.2. In particular, the log-likelihood is of the form (1.9), smoothing parameter estimation can be carried out by employing the Fellner-Schall update (1.10), and the uncertainty of the estimates can be quantified via the Bayesian large sample approximation as in the GAM setting.

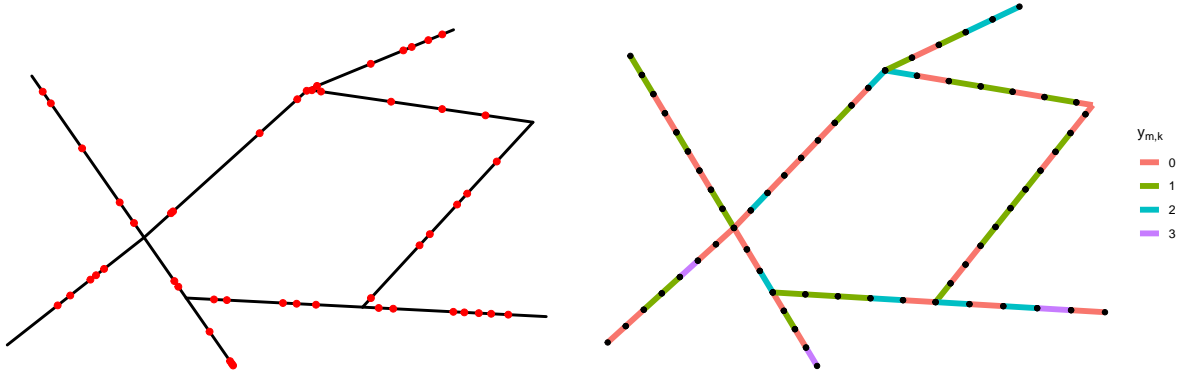


Figure 1.6.: Left panel: Point pattern on a small geometric network. Right panel: Binning of the point pattern.

1.3.4. On-street parking

Sections 1.3.2 and 1.3.3 have already highlighted two scopes of application for geometric networks, latent network flow estimation and intensity estimation of point processes. Chapter 4 covers the third application of this dissertation and is devoted to the prediction of on-street parking availability. Here, parking lots are located on a geometric network of streets and can either be in state 0 (clear) or 1 (occupied). The network approach allows to capture the correlation of nearby parking lots by introducing the following spatial covariate

$$\text{nearby}_{j,t}^{(i)} = \sum_{k \neq i} \mathbb{1}\{X_t^{(k)} = j, d_{\mathbf{G}}(\mathbf{s}_i, \mathbf{s}_k) \leq h\} / \sum_{k \neq i} \mathbb{1}\{d_{\mathbf{G}}(\mathbf{s}_i, \mathbf{s}_k) \leq h\}, \quad (1.18)$$

where $X_t^{(k)} = j \in \{0, 1\}$ is the state of the k -th parking lot at time point t and $\mathbf{s}_i \in \mathbf{G}$ is the location of the i -th parking lot. Therefore, $\text{nearby}_{j,t}^{(i)}$ quantifies the fraction of parking lots located within a (shortest-path) distance of less than h around parking lot i , that are in state j at time point t . A covariate as defined in (1.18) is employed in a time-to-event model, a class of models akin to GAMMs.

1.4. Time-to-event models

“While nothing is more uncertain than a single life, nothing is more certain than the average duration of a thousand lives.”

— Elizur Wright (* 1804, † 1885),
American mathematician and abolitionist

When modeling time-to-event data, the response variable of interest usually is a nonnegative number, the time of an event. What makes time-to-event models different from regression models, which were covered in Section 1.2, are possible censoring and truncation mechanisms affecting the response. The scope of application for time-to-event models is diverse, but the models are mainly employed in the life sciences such as medicine or biology. For the former example, the response variable often is the time from the beginning of a study until cure or

death and observations are right-censored if an individual withdraws before the outcome has been observed (Kardaun, 1983). For this reason, the term “survival analysis” is often used as a synonym for time-to-event models. In engineering, the field is called “reliability analysis” or “failure time analysis”. Here, the failure rate of machinery, e.g. of a wind turbine, is of interest (Tavner et al., 2007). Thus, time-to-event models can also be employed to model transition rates (or intensities) from one state (e.g. alive, operational) to another state (e.g. dead, non-operational) of a stochastic process with countably many states, which is pursued in Chapter 4.

1.4.1. Model for continuous data

Let D be a nonnegative and continuous random variable with cumulative distribution function $F_D(d) = \mathbb{P}(D \leq d)$. Thus, the function $S_D(d) = 1 - F_D(d)$ is decreasing for $d > 0$ and is denoted as the “survival function” or the “reliability function”, depending on the context. The hazard rate or intensity function of D is defined by

$$\lambda(d) = \lim_{\Delta d \rightarrow 0} \frac{\mathbb{P}(d \leq D < d + \Delta d \mid D \geq d)}{\Delta d} \quad (1.19)$$

which quantifies the conditional failure rate (Klein and Moeschberger, 2006). The intensity $\lambda(\cdot)$ is related to the distribution of D via

$$F_D(d) = 1 - \exp\left(-\int_0^d \lambda(x) dx\right) = 1 - \exp(-\Lambda(d)), \quad (1.20)$$

where the integral $\Lambda(d)$ in (1.20) denotes the cumulative hazard or cumulative intensity. Therefore, to draw inference about $F_D(\cdot)$ or $S_D(\cdot)$, respectively, a model for $\lambda(\cdot)$ is required which characterizes D uniquely.

Statistical model for intensities Let us assume that the same setting as in a GLMM in Section 1.2.1. However, now the response is a tuple (d_{it}, δ_{it}) with d_{it} denoting the observed duration of the t -th observation of the i -th unit and δ_{it} is a censoring indicator where $\delta_{it} = 1$ means that the corresponding observation is right-censored after duration d_{it} . In general, the censoring times are allowed to be statistically dependent on the actual duration times, known as informative censoring. For simplicity, only noninformative right censoring is considered in the following.

A simple model for the hazard rate is

$$\lambda_{it}(d) = \lambda_0(d) \exp(\eta_{it} + u_i) = \lambda_0(d) v_i \exp(\eta_{it}), \quad (1.21)$$

where $\lambda_0(\cdot)$ is a common baseline intensity for all subjects, η_{it} is a linear predictor as in (1.2) and u_i is a random effect of the i -th subject. In the context of time-to-event analysis, one rather specifies the subject specific randomness in terms of the so called frailties $v_i = \log u_i$ which act multiplicatively on the baseline intensity. The density of the frailties is denoted with $f(\cdot; \boldsymbol{\psi})$ where $\boldsymbol{\psi}$ are the parameters shaping the distribution of the frailties. Note that the exponent in (1.21) is assumed to be independent of d which is why a model of the form (1.21) is called a proportional hazards model (Vaida and Xu, 2000). Structural assumptions for the baseline intensity can be parametric in which case the baseline intensity $\lambda_0(\cdot; \boldsymbol{\alpha})$ is shaped by parameters $\boldsymbol{\alpha}$. A commonly used model is a Weibull time-to-event model in which $\lambda_0(\cdot; \alpha) = \alpha t^{\alpha-1}$ with parameter $\alpha > 0$. Alternatively, $\lambda_0(\cdot)$ can be modeled non-parametrically as in the Cox model

(Therneau and Grambsch, 2000). Model parameters in the Cox model can be estimated by employing a partial likelihood approach (Cox, 1975). Moreover, by partitioning the data into a fixed number of intervals and assuming the hazard to be constant in those intervals, a time-to-event model can be represented as a Poisson generalized linear model. Such a model is known as piece-wise exponential model (PEM, Friedman, 1982) and was extended by Bender et al. (2018) to a piece-wise exponential additive mixed model (PAMM), which allows for flexible modeling of the baseline intensity. Moreover, this model facilitates an extension of (1.21) towards the inclusion of (duration) time varying covariate effects. A third representation of time-to-event models builds on counting process theory which is outlined by Kalbfleisch and Prentice (2011).

Estimation According to Klein and Moeschberger (2006), the conditional likelihood of the data restricted to the i -th individual is given by

$$L_i(\boldsymbol{\alpha}, \boldsymbol{\theta} \mid u_i) = \prod_{k=1}^{n_i} [\lambda(d_{ik})]^{\delta_{ik}} \exp[-\Lambda(d_{ik})],$$

where n_i is the number of observations for subject i . Similar as in a GLMM, the maximum likelihood estimate $\hat{\boldsymbol{\zeta}} = (\hat{\boldsymbol{\alpha}}^\top, \hat{\boldsymbol{\theta}}^\top, \hat{\boldsymbol{\psi}}^\top)^\top$ can be obtained by maximizing the marginal log-likelihood

$$\ell_{\text{marg}}(\boldsymbol{\zeta}) = \sum_{i=1}^n \log \int L_i(\boldsymbol{\alpha}, \boldsymbol{\theta} \mid v_i) f(v_i; \boldsymbol{\psi}) dv_i. \quad (1.22)$$

In most situations, the integral in (1.22) is not analytically tractable, and therefore, methods that were discussed above in the context of GLMMs, such as a Laplace approximation, need to be employed. In special cases, analytic expressions are available. An example is the gamma shared frailty model, where $v_i = \log u_i \sim \text{Gamma}(1/\psi, 1/\psi)$ (Duchateau and Janssen, 2007). This model is applied in Chapter 4 to predict the duration times of parking lots being clear or occupied.

1.4.2. Model for discrete data

Let us now assume that the duration time D is a discrete random variable with support $\{1, \dots, d_{\max}\}$. In Chapter 5, D is the time in days from the registration as infected with the SARS-CoV-2 virus which causes the disease COVID-19 (Velavan and Meyer, 2020) until the fatal outcome of the disease. Therefore, D can also be considered as an ordered categorical random variable. Such data can be modeled by making use of a sequential multinomial model, which, in its simplest form, is according to Albert and Chib (2001) given by

$$\pi(d; \boldsymbol{\gamma}, \boldsymbol{\beta}) = \mathbb{P}(D = d \mid D \geq d; \boldsymbol{\gamma}, \boldsymbol{\beta}) = F(\boldsymbol{\gamma}_d - \boldsymbol{x}^\top \boldsymbol{\beta}) \quad (1.23)$$

for $d = 1, \dots, d_{\max} - 1$. Moreover, $F(\cdot)$ is a fixed distribution function, e.g. the logistic distribution function. The regression coefficients $\boldsymbol{\gamma} = (\gamma_1, \dots, \gamma_{d_{\max}-1})^\top$ define the transition from category d to category $d+1$ and $\boldsymbol{\beta}$ is a vector of linear covariate effects. However, the structure of the data, that is modeled in Chapter 5 and sketched in the following, does not allow to directly use the above formulation of the sequential model (1.23).

Structure of the data Let $t = 0, \dots, T-1$ be a sequence of registration dates, e.g. for being infected with the SARS-CoV-2 virus, and let $t = T$ be the current date. The random variable

$t \backslash d$	1	2	...	d_{\max}	Reported Deaths at time point $t = T$	
0	$N_{0,1}$	$N_{0,2}$...	$N_{0,d_{\max}}$	$C_{0,d_{\max}}$	Final number of deaths is known
1	$N_{1,1}$	$N_{1,2}$...	$N_{1,d_{\max}}$	$C_{1,d_{\max}}$	
\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	
$T - d_{\max}$	$N_{T-d_{\max},1}$	$N_{T-d_{\max},2}$...	$N_{T-d_{\max},d_{\max}}$	$C_{T-d_{\max},d_{\max}}$	
$T - d_{\max} + 1$	$N_{T-d_{\max}+1,1}$	$N_{T-d_{\max}+1,2}$...	NA	$C_{T-d_{\max}+1,d_{\max}-1}$	Final number of deaths is unknown
\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	
$T - 2$	$N_{T-2,1}$	$N_{T-d_{\max}+1,1}$	NA	NA	$C_{T-2,2}$	
$T - 1$	$N_{T-2,1}$	NA	NA	NA	$C_{T-1,1}$	
T	NA	NA	NA	NA	$C_{T,0} = 0$	

Figure 1.7.: Illustration of the data modeled in Chapter 5.

$N_{t,d}$ denotes the number of deaths with registration date t and duration d with fixed maximum duration d_{\max} , i.e. at the latest d_{\max} days after the registration date every infected person is considered to be either cured or dead. However, the terminal number of deaths for registration date t becomes only available at time point $t + d_{\max}$ which is why $N_{t,d}$ is unknown for $t + d > T$. Further, $C_{t,d} = \sum_{l=1}^d N_{t,l}$ is the partial cumulated sum of reported deaths, i.e. at time $t = T$ one observes $C_{T-d,d}$. This scheme is illustrated in Figure 1.7. Note that such a data structure is also employed in actuarial loss reserving for claims which have incurred but not yet been reported (Mack, 1993).

Statistical model based on a sequential model Apparently, the data structure does not allow a modeling approach of the form (1.23) but instead, one could consider the related model

$$\pi(d; t, \boldsymbol{\theta}) = \mathbb{P}(D = d \mid D \leq d; \boldsymbol{\theta}) = F(s(d; \boldsymbol{\gamma}) + \eta_{t,d}) \quad (1.24)$$

for $d = 2, \dots, d_{\max}$, where the reference category has changed from d_{\max} to 1. Furthermore, the sequence of binary transitions is now modeled through a smooth function $s(\cdot; \boldsymbol{\gamma})$ which can be represented through a B-spline basis representation as proposed in Section 1.2.2. Therefore, the linear predictor in (1.24) is of the general form (1.2). Due to the above considerations and (1.24) the number of newly reported deaths can be modeled according to

$$N_{t,d} \stackrel{\text{indep.}}{\sim} \text{Bin}(C_{t,d}, \pi(d; t, \boldsymbol{\theta})). \quad (1.25)$$

In Chapter 5, a quasi-Binomial model instead of a binomial model is used in order to account for possible overdispersion, which is a phenomenon arising if the variances in the data are larger than implied by the binomial model (Collett, 2002).

Model (1.25) can be used to predict the probabilities (1.24), that, in turn, can be employed to estimate the distribution function $F_t(\cdot)$ of the duration D_t from registration to death for

registration date t since

$$\begin{aligned}
F_t(d) &= \mathbb{P}(D_t \leq d) = \mathbb{P}(D_t \leq d \mid D_t \leq d+1) \cdot \mathbb{P}(D_t \leq d+1) \\
&= \mathbb{P}(D_t \leq d \mid D_t \leq d+1) \cdot \dots \cdot \mathbb{P}(D_t \leq d_{\max} - 1 \mid D_t \leq d_{\max}) \cdot \underbrace{\mathbb{P}(D_t \leq d_{\max})}_{=1} \\
&= \prod_{k=d+1}^{d_{\max}} [1 - \mathbb{P}(D_t \geq k \mid D_t \leq k)] = \prod_{k=d+1}^{d_{\max}} [1 - \mathbb{P}(D_t = k \mid D_t \leq k)] = \prod_{k=d+1}^{d_{\max}} [1 - \pi(k; t, \boldsymbol{\theta})].
\end{aligned}$$

1.5. Stochastic processes

“Prediction is very difficult, especially of the future.”

— Niels Henrik David Bohr (* 1885, † 1962),
Danish physicist

Consider again the case where data $y_{it} \in \mathcal{Y}$ and $\mathbf{x}_{it} \in \mathcal{X}$ for subjects $i = 1, \dots, n$ and observation times $t \in \mathcal{T}$ are observed. However, the index set \mathcal{T} is now supposed to be a connected subset of \mathbb{R} and thus, switching notation slightly, $Y_i = (Y_{it})_{t \in \mathcal{T}}$ can be considered as a stochastic process in continuous time with associated covariate process $\mathbf{x}_i = (\mathbf{x}_{it})_{t \in \mathcal{T}}$. Each realization $y_i = Y_i(\omega)$ is a sample path on the set \mathcal{T} such that Y_i could also be considered as a random function with domain \mathcal{T} . An example of a stochastic process for continuous \mathcal{Y} is a Brownian motion where Y_{it} is normally distributed (Mörters and Peres, 2010). To name one of many scopes of application, Brownian motions play a crucial role in the construction of the heat kernel (1.16), for more details see McSwiggan et al. (2017). Nonetheless, the focus should be here on stochastic processes where Y_{it} does have countable support, more precisely, on processes of the Markov type which are the basis of the prediction model that is framed in Chapter 4. The task is to predict Y_{it} at time $t > t_0$ when having observed Y_i until t_0 . Therefore, instead of postulating a conditional independence assumption as in the GAMM setting, a reference is drawn between time-to-event models from Section 1.4.1 and the theoretical properties of the stochastic processes. Vice versa, continuous time-to-event models can also be formulated in terms of stochastic processes as outlined in Kalbfleisch and Prentice (2011). In the following, a brief introduction into the theory of (semi-)Markov processes is given with a focus on deriving interval transition probabilities. These are used in Chapter 4 to predict the future states of a two-state stochastic process. For a general overview of stochastic processes in general, see Ross et al. (1996).

1.5.1. Markov processes

Definition Let $\mathcal{T} = [0, \infty)$ and, for simplicity of notation, subject index i being disregarded. A stochastic process $Y = (Y_t)_{t \geq 0}$ with countable state space (or support) \mathcal{S} is said to be a continuous time Markov process if

$$\mathbb{P}(Y_{t+s} = j \mid Y_s = i, \{Y_u, 0 \leq u < s\}) = \mathbb{P}(Y_{t+s} = j \mid Y_s = i) \quad (1.26)$$

for $s, t \geq 0$ and $i, j \in \mathcal{S}$. In other words, given the state of Y at time s , the state of Y at time $t + s$ is independent of the history $\{Y_u, 0 \leq u < s\}$ of the process. Moreover, denote with

D_i the random duration that Y stays in state i . It then follows from the Markov property (1.26) that $\mathbb{P}(D_i > s + t \mid D_i > s) = \mathbb{P}(D_i > t)$, i.e. the distribution of D_i is memorylessness. Since the exponential distribution is the only continuous distribution which has this property, it follows that $D_i \sim \text{Exp}(\lambda_i)$. Thus, a Markov process can be characterized in terms of rates λ_i for $i \in \mathcal{S}$ and a transition probability matrix $\mathbf{P} = [p_{ij}, i, j \in \mathcal{S}]$, i.e. each row sums up to one. Equivalently, a Markov process can be defined via the transition rate matrix $\mathbf{Q} = [q_{ij}, i, j \in \mathcal{S}]$ with $q_{ij} = \lambda_i p_{ij}$ for $i \neq j$. Usually, it is assumed that $\lambda_i > 0$ and that Y is regular, i.e. there are no absorbing states and the probability of infinitely many transitions in an infinitesimal small interval is zero.

Interval transition probabilities The probabilities $P_{ij}(s, s + t) = \mathbb{P}(Y_{s+t} = j \mid Y_s = i)$ are for $s, t \geq 0$ denoted as the interval transition probabilities from state i to state j in an interval of length t . The stationarity of Y implies that $P_{ij}(s, s + t) = P_{ij}(0, t)$ and therefore, setting $P_{ij}(t) = P_{ij}(0, t)$, these probabilities can be obtained as the solution of one of the following two differential equations

$$\frac{\partial}{\partial t} P_{ij}(t) = \sum_{k \neq i} q_{ik} P_{kj}(t) - \lambda_i P_{ij}(t), \quad \frac{\partial}{\partial t} P_{ij}(t) = \sum_{k \neq j} q_{kj} P_{ik}(t) - \lambda_j P_{ij}(t), \quad (1.27)$$

which are known as the Kolmogorov backward/forward equations. In the simplest case where $\mathcal{S} = \{0, 1\}$ with transition intensities defined as $\lambda_0 = q_{01}$ and $\lambda_1 = q_{10}$, the solution of either one of the Kolmogorov equations (1.27) is given by

$$\mathbf{P}(t) = \begin{pmatrix} P_{00}(t) & P_{01}(t) \\ P_{10}(t) & P_{11}(t) \end{pmatrix} = \frac{1}{\lambda_0 + \lambda_1} \begin{pmatrix} \lambda_1 + \lambda_0 e^{-t(\lambda_0 + \lambda_1)} & \lambda_0 [1 - e^{-t(\lambda_0 + \lambda_1)}] \\ \lambda_1 [1 - e^{-t(\lambda_0 + \lambda_1)}] & \lambda_0 + \lambda_1 e^{-t(\lambda_0 + \lambda_1)} \end{pmatrix}.$$

Such a model can predict the outcome of a binary random variable associated with a stochastic process, e.g. the short-time occupation of a parking lot. The parameters of the model are λ_0 and λ_1 , which can be estimated with a parametric time-to-event model, restricting the baseline intensity to be constant.

1.5.2. Semi-Markov processes

Definition If the transition intensities $\lambda_{ij}(d)$ from a state $i \in \mathcal{S}$ into another state $j \in \mathcal{S}$ of a stochastic process $Y = (Y_t)_{t \geq 0}$ are a function of the duration time d , the resulting random duration times D_i in state i are not exponentially distributed anymore. It follows that the memorylessness property now only holds at the instance of the transition to another state which is evident if the distribution of the duration times D_i is of the form (1.20). Therefore, such a process is denoted as a semi-Markov process which can be characterized by the means of a renewal kernel $\mathbf{Q}(d) = [Q_{ij}(d), i, j \in \mathcal{S}]$ with

$$Q_{ij}(d) = \mathbb{P}(Y_{t_{(n+1)}} = j, D_{(n)} \leq d \mid Y_{t_{(n)}} = i, \{D_{(k)}, Y_{t_{(k)}}, k = 0, \dots, n-1\}) \quad (1.28)$$

$$= \mathbb{P}(Y_{t_{(n+1)}} = j, D_{(n)} \leq d \mid Y_{t_{(n)}} = i), \quad (1.29)$$

where $0 = t_{(0)} < t_{(1)} < \dots$ are the time points of state changes and $D_{(n)}$ is the duration in state $X_{t_{(n)}}$ (Grabski, 2014). Thus, by the law of total probability the cumulative conditional

distribution function of the duration D_i in state i can be computed by

$$\begin{aligned} F_i(d) &= \mathbb{P}(D_{(n)} \leq d \mid Y_{t_{(n)}} = i) = \sum_{k \in \mathcal{S}} \mathbb{P}(D_{(n)} \leq d \mid Y_{t_{(n)}} = i, Y_{t_{(n+1)}} = j) \mathbb{P}(Y_{t_{(n+1)}} = j) \\ &= \sum_{k \in \mathcal{S}} \mathbb{P}(Y_{t_{(n+1)}} = j, D_{(n)} \leq d \mid Y_{t_{(n)}} = i) = \sum_{k \in \mathcal{S}} \mathcal{Q}_{ij}(d). \end{aligned}$$

Defining with $p_{ij} = \lim_{d \rightarrow \infty} \mathcal{Q}_{ij}(d)$ a stochastic matrix $\mathbf{P} = [p_{ij}, i, j \in \mathcal{S}]$, equivalently to (1.29), a semi-Markov process can be defined via conditional distribution functions $\mathbf{F}(d) = [F_i(d), i \in \mathcal{S}]$ and the transition probability matrix \mathbf{P} . This shows that a semi-Markov process actually generalizes Markov processes allowing for arbitrarily distributed duration times. Asanjarani et al. (2021) define semi-Markov processes equivalently via transition intensities, emphasizing the close connection between discrete state space stochastic processes and time-to-event models.

Interval transition probabilities There are several ways how interval transition probabilities $P_{ij}(t) = \mathbb{P}(Y_t = j \mid Y_0 = i)$ for a semi-Markov process can be determined. One option is to repeatedly simulate from the distribution of the duration times D_i and to use \mathbf{P} to generate sample paths of Y . However, transition probabilities can also be derived as the solutions of the following integral equations

$$P_{ij}(t) = (1 - F_i(t))\delta_{ij} + \sum_{k \in \mathcal{S}} \int_0^t P_{kj}(t-x)q_{im}(x) dx \quad (1.30)$$

with initial condition $P_{ij}(0) = \delta_{ij}$ (Grabski, 2014). Here, $\delta_{ij} = \mathbb{1}\{i = j\}$ denotes the Kronecker delta and $q_{ij}(\cdot)$ is the derivative of $\mathcal{Q}_{ij}(\cdot)$. In order to solve (1.30) for $P_{ij}(t)$, Laplace transforms can be employed (Widder, 2015). For a real valued function $f : [0, \infty) \rightarrow \mathbb{R}$ the Laplace transform $\mathcal{L}\{f\} : C \rightarrow \mathbb{C}$ is given by

$$\mathcal{L}\{f\}(u) = \int_0^\infty f(t)e^{-ut} dt,$$

where $C = \{u \in \mathbb{C} \mid \Re(u) > \gamma\}$ is the region of convergence and γ is called the abscissa of convergence (Hall et al., 1992). Applying the Laplace transform to (1.30) leads to a linear equation system for the Laplace transformed transition probabilities $\mathcal{L}\{P_{ij}\}(u)$ which can easily be solved, details are given in Chapter 4. The transition probabilities in the real domain can finally be obtained by applying the inverse Laplace transform \mathcal{L}^{-1} to $\mathcal{L}\{P_{ij}\}(u)$. Therefore, $P_{ij}(t)$ is given by the following Bromwich integral (Weideman and Trefethen, 2007)

$$P_{ij}(t) = \mathcal{L}^{-1}\{\mathcal{L}\{P_{ij}\}(u)\}(t) = \frac{1}{2\pi i} \lim_{T \rightarrow \infty} \int_{\gamma_0 - iT}^{\gamma_0 + iT} e^{ut} \mathcal{L}\{P_{ij}\}(u) du, \quad (1.31)$$

where $\gamma_0 > \gamma$, i.e. (1.31) is a line integral through the region of convergence of $\mathcal{L}\{P_{ij}\}$. In most cases, numerical techniques need to be employed to compute an inverse Laplace transform. The approach proposed by Valsa and Brančik (1998) builds on an approximation of the complex exponential in (1.31) and is summarized in the appendix of Chapter 4. An alternative has been proposed by De Hoog et al. (1982), who use the convergence of Fourier series and approximate the integral (1.31) by making use of the trapezoidal rule. The review article Kuhlman (2013) compares various inversion methods in terms of numerical efficiency.

1.6. Software and computational aspects

“In some ways, programming is like painting. You start with a blank canvas and certain basic raw materials. You use a combination of science, art, and craft to determine what to do with them.”

— Andy Hunt (* 1964),
American author of books on software development

Undoubtedly, the most sophisticated statistical model is worth nothing if it can not be applied to an actual data set, demanding a robust numerical implementation with statistical software. Every contributed manuscript in this dissertation also deals with the implementation of the respective model in **R**, an open source software for statistical computing and graphics (R Core Team, 2021). The functionality of **R** is organized within a few core packages, which are already included when installing **R**, informally known as “Base **R**”. However, the power of this statistical language could only arise through thousands of additional packages available through different repositories. Packages which have undergone a standardized quality check are available from the comprehensive R archive network (CRAN).³ In the following, essential packages which have been used for this dissertation are briefly summarized. Afterward, the own software contribution, which partially builds on already existing software, is outlined.

Existing software The first collection of **R** packages which needs to be mentioned here, is the tidyverse (Wickham et al., 2019). These packages share the same grammar and provide functions for importing data from different sources (packages `readr` and `readxl`) as well as functionalities for data wrangling and storing data in a tidy form (packages `tidyr` and `tibble`). Moreover, data transformation plays a fundamental role where the package `dplyr` provides, among others, tools for mutating, rearranging or joining data. Packages for the manipulation of specific types of data such as strings (`stringr`), factors (`forcats`) or dates (`lubridate`) are also heavily used. Once the data are in a neat format, first exploratory analyses can be visualized using the package `ggplot2` (Wickham, 2010) which is also part of the tidyverse and creates powerful graphics based on Wilkinson (2012). This package builds on the “layered grimmer of graphics” and, as the name suggests, a `ggplot2` graphics is organized via different layers.

For the statistical modeling with generalized additive mixed models, the comprehensive package `mgcv` is used (Wood, 2017). The routine `gam` exploits the equivalence of smooths and random effects as outlined in Section 1.2.3 and treats simple random effects as smooths. Kernel based intensity estimation of point processes on linear networks can be handled with the functionality of the family of `spatstat` packages (Baddeley et al., 2015). For survival data the `survival` package is used (Therneau, 2021). Finally, for computing inverse Laplace transforms the algorithm which is already implemented within the package `pracma` (Borchers, 2021) is used.

New software The Skellam regression model in Chapter 2 can not be fitted with routines of available GAMM software such as the `mgcv` package since it can neither handle the Skellam distribution nor the form of the linear predictor. Therefore, maximum likelihood estimation for this model is performed via a quasi-Newton algorithm which is implemented in the routine `optim` of the `stats` package. Here, the design matrix must be stored in a sparse format using the `Matrix`

³<https://cran.r-project.org/>

package, which tremendously reduces computation time. Nonetheless, numerical problems in this modeling approach arise especially in conjunction with underflow when computing modified Bessel functions.

Second, the statistical model developed in Chapter 3 in order to estimate the intensity of point processes on geometric networks needs its own implementation since network-based B-splines and penalties can not be handled by routines such as `gam`. The implementation is bundled into the R package `geonet` which is presented in the Appendix of Chapter 3. Thereby, special attention is given to the compatibility with the `spatstat` package, i.e. networks in linear representation can be represented as geometric networks and vice versa. The package is available from CRAN, i.e. it can easily be downloaded with the command `install.packages("geonet")`. A development version which is updated more often can be downloaded from a GitHub repository by using the command `devtools::install_github("MarcSchneble/geonet")`.

1.7. Discussion

“Far better an approximate answer to the right question, which is often vague, than an exact answer to the wrong question, which can always be made precise.”

— John Wilder Tukey (* 1915, † 2000),
American mathematician and statistician

This dissertation addresses various statistical modeling approaches across different areas of application. The first part is devoted to modeling data regarding spatial networks, while the second part covers applications in infectious disease modeling focusing on deadly outcomes. Each of the contributed manuscripts has a different focus. Chapter 2 deals with explaining the unobserved bike trips in a bike-sharing network from the observed station feeds only. The model could, in principle, also be used for forecasting the bike trip pattern, which is, however, not the primary goal here. The results show that using the proposed semi-Markov model distinctly outperforms existing Markov models in prediction accuracy. The model presented in Chapter 3 aims at providing an alternative to an already existing model. Here, the first contribution focuses on the methodology and the justification for the need of the model, and the second contribution deals with the implementation in **R**. The principal aim of the contribution in Chapter 4 is the prediction of parking availability, and even though the results of the time-to-event model allow various interpretations, this is not of primary interest here. The goals of Chapter 5 are twofold. First, the spatio-temporal patterns of COVID-19 mortality in Germany are explained and interpreted. Second, a nowcasting model predicts the future number of COVID-19 deaths which have already been infected. This fits into the nowcasting terminology by Bańbura et al. (2010), who define nowcasting to be “the prediction of the present, the very near future and the very recent past”. Lastly, the aim of the contribution in Chapter 6 is to provide a comparatively simple model, which on the other hand, allows interpreting the course COVID-19 infections and its case detection ratio in the first year of the pandemic in Germany.

Most of the models which are employed in Chapters 2 - 6 can be put in the context of regression, and this dissertation shows that the class of generalized additive mixed models can be seen as one of the major workhorses in statistical modeling. An overview of the regression-based models which are used in the contributed manuscripts of this dissertation is shown in Table 1.1. The third column emphasizes the importance of the smoothing techniques which

Chapter	Model	Structure	Implementation
2	Skellam GAMM	(Cyclic) P-splines and bivariate random effects	own
3	Poisson GAM	P-splines (on a geometric network)	own
4	Weibull time-to-event model	B-splines and random effects	<code>survival</code> package
5	Binomial GAM and negative binomial GAMM	Bivariate random effects and bivariate P-splines	<code>mgcv</code> package
6	Negative binomial GAMM	Varying coefficients, functional random effects	<code>mgcv</code> package

Table 1.1.: Overview of the regression models used throughout this dissertation.

were introduced in Section 1.2, and it should be emphasized again that random effects can be interpreted as smoothers as well. The last column of Table 1.1 states how the respective model has been implemented. Whenever possible, standard software has been used in order to fit the models.

Both own implementations partially make use of already existing software, which illustrates that, also from a practical point of view, statistical modeling can be seen as a large toolbox. When implementing new models in a formal language such as \mathbf{R} , some of the already existing components can be used which also holds for the theoretical development of the models. Figuratively speaking, model components such as smoothers of any kind are some of the basements of statistical modeling, and this dissertation demonstrates their universal scopes of application.

Bibliography

- Abramowitz, M. and I. A. Stegun (1964). *Handbook of mathematical functions with formulas, graphs, and mathematical tables*, Volume 55. US Government Printing Office.
- Airoldi, E. M. and A. W. Blocker (2013). Estimating latent processes on a network from indirect measurements. *Journal of the American Statistical Association* 108(501), 149–164.
- Albert, J. H. and S. Chib (2001). Sequential ordinal modeling with applications to survival data. *Biometrics* 57(3), 829–836.
- Antweiler, W. (2001). Nested random effects estimation in unbalanced panel data. *Journal of Econometrics* 101(2), 295–313.
- Asanjarani, A., B. Liquef, and Y. Nazarathy (2021). Estimation of semi-Markov multi-state models: A comparison of the sojourn times and transition intensities approaches. *The International Journal of Biostatistics*.
- Baddeley, A., E. Rubak, and R. Turner (2015). *Spatial point patterns: Methodology and applications with R*. CRC press.
- Bañbura, M., D. Giannone, and L. Reichlin (2010). Nowcasting. Technical report, ECB Working Paper.
- Bender, A., A. Groll, and F. Scheipl (2018). A generalized additive model approach to time-to-event analysis. *Statistical Modelling* 18(3-4), 299–321.
- Bernardo, J. M. and A. F. Smith (2009). *Bayesian theory*, Volume 405. John Wiley & Sons.
- Bierens, H. J. (1987). Kernel estimators of regression functions. In *Advances in econometrics: Fifth world congress*, Volume 1, pp. 99–144. Cambridge University Press New York.
- Borchers, H. W. (2021). *pracma: Practical Numerical Math Functions*. R package version 2.3.3.
- Botev, Z. I., J. F. Grotowski, and D. P. Kroese (2010). Kernel density estimation via diffusion. *The Annals of Statistics* 38(5), 2916–2957.
- Box, G. E. (1979). Robustness in the strategy of scientific model building. In *Robustness in statistics*, pp. 201–236. Elsevier.
- Breslow, N. E. and D. G. Clayton (1993). Approximate inference in generalized linear mixed models. *Journal of the American Statistical Association* 88(421), 9–25.
- Casella, G. and R. Berger (2001). *Statistical Inference*. Duxbury Resource Center.
- Collett, D. (2002). *Modelling binary data*. CRC press.
- Cox, D. R. (1975). Partial likelihood. *Biometrika* 62(2), 269–276.

- Cox, D. R. and D. V. Hinkley (1979). *Theoretical statistics*. CRC Press.
- Cox, R. T. (1946). Probability, frequency and reasonable expectation. *American Journal of Physics* 14(1), 1–13.
- Craven, P. and G. Wahba (1978). Smoothing noisy data with spline functions. *Numerische Mathematik* 31(4), 377–403.
- Currie, I. D., M. Durban, and P. H. Eilers (2006). Generalized linear array models with applications to multidimensional smoothing. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 68(2), 259–280.
- Datar, A. and R. Sturm (2004). Physical education in elementary school and body mass index: Evidence from the early childhood longitudinal study. *American Journal of Public Health* 94(9), 1501–1506.
- Davison, A. C. (2003). *Statistical models*, Volume 11. Cambridge University Press.
- De Boor, C. (1972). On calculating with B-splines. *Journal of Approximation Theory* 6(1), 50–62.
- De Hoog, F. R., J. Knight, and A. Stokes (1982). An improved method for numerical inversion of Laplace transforms. *SIAM Journal on Scientific and Statistical Computing* 3(3), 357–366.
- Dempster, A. P., N. M. Laird, and D. B. Rubin (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)* 39(1), 1–22.
- Der Kiureghian, A. and O. Ditlevsen (2009). Aleatory or epistemic? Does it matter? *Structural Safety* 31(2), 105–112.
- Duchateau, L. and P. Janssen (2007). *The frailty model*. Springer Science & Business Media.
- Durban, M. and M. C. Aguilera-Morillo (2017). On the estimation of functional random effects. *Statistical Modelling* 17(1-2), 50–58.
- Eilers, P. H. and B. D. Marx (1996). Flexible smoothing with B-splines and penalties. *Statistical science* 11(2), 89–121.
- Eilers, P. H. and B. D. Marx (2021). *Practical Smoothing: The Joys of P-splines*. Cambridge University Press.
- Epanechnikov, V. A. (1969). Non-parametric estimation of a multivariate probability density. *Theory of Probability & Its Applications* 14(1), 153–158.
- Fahrmeir, L. and H. Kaufmann (1985). Consistency and asymptotic normality of the maximum likelihood estimator in generalized linear models. *The Annals of Statistics*, 342–368.
- Fahrmeir, L., T. Kneib, S. Lang, and B. Marx (2007). *Regression*. Springer.
- Fahrmeir, L. and G. Tutz (2013). *Multivariate statistical modelling based on generalized linear models*. Springer Science & Business Media.

- Friedman, M. (1982). Piecewise exponential models for survival data with covariates. *The Annals of Statistics* 10(1), 101–113.
- Friel, N. and A. N. Pettitt (2008). Marginal likelihood estimation via power posteriors. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 70(3), 589–607.
- Girden, E. R. (1992). *ANOVA: Repeated measures*. Number 84. Sage.
- Gottfried, K. and T.-M. Yan (2013). *Quantum mechanics: Fundamentals*. Springer Science & Business Media.
- Grabski, F. (2014). *Semi-Markov processes: Applications in system reliability and maintenance*. Elsevier.
- Graham, A. (2018). *Kronecker products and matrix calculus with applications*. Courier Dover Publications.
- Hall, P., J. L. Teugels, and A. Vanmarcke (1992). The abscissa of convergence of the Laplace transform. *Journal of Applied Probability*, 353–362.
- Hansen, P. C. (1998). *Rank-deficient and discrete ill-posed problems: Numerical aspects of linear inversion*. SIAM.
- Hartzel, J., I.-M. Liu, and A. Agresti (2001). Describing heterogeneous effects in stratified ordinal contingency tables, with application to multi-center clinical trials. *Computational Statistics & Data Analysis* 35(4), 429–449.
- Hastie, T. and R. Tibshirani (1993). Varying-coefficient models. *Journal of the Royal Statistical Society: Series B (Methodological)* 55(4), 757–779.
- Hastie, T. J. and R. J. Tibshirani (1990). *Generalized additive models*, Volume 43. CRC press.
- Hoff, P. D., A. E. Raftery, and M. S. Handcock (2002). Latent space approaches to social network analysis. *Journal of the American Statistical Association* 97(460), 1090–1098.
- Kalbfleisch, J. D. and R. L. Prentice (2011). *The statistical analysis of failure time data*, Volume 360. John Wiley & Sons.
- Kardaun, O. (1983). Statistical survival analysis of male larynx-cancer patients—a case study. *Statistica Neerlandica* 37(3), 103–125.
- Karunamuni, R. J. and T. Alberts (2005). On boundary correction in kernel density estimation. *Statistical Methodology* 2(3), 191–212.
- Kauermann, G., T. Krivobokova, and L. Fahrmeir (2009). Some asymptotic results on generalized penalized spline smoothing. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 71(2), 487–503.
- Kauermann, G. and J. D. Opsomer (2011). Data-driven selection of the spline dimension in penalized spline regression. *Biometrika* 98(1), 225–230.
- Klein, J. P. and M. L. Moeschberger (2006). *Survival analysis: techniques for censored and truncated data*. Springer Science & Business Media.

- Kolaczyk, E. D. (2009). *Statistical analysis of network data. Methods and Models*. Springer Science+Business Media.
- Kolmogorov, A. N. (1950). *Foundations of the theory of probability*. New York: Chelsea Pub. Co.
- Konishi, S. and G. Kitagawa (2008). *Information criteria and statistical modeling*. Springer Science & Business Media.
- Kostykin, V., J. Potthoff, and R. Schrader (2007). Heat kernels on metric graphs and a trace formula. *Contemporary Mathematics* 447, 175.
- Kuhlman, K. L. (2013). Review of inverse Laplace transform algorithms for Laplace-space numerical approaches. *Numerical Algorithms* 63(2), 339–355.
- Laird, N. M. and J. H. Ware (1982). Random-effects models for longitudinal data. *Biometrics*, 963–974.
- Laplace, P. S. (1814). Essai philosophique sur les probabilités.
- Li, Y. and D. Ruppert (2008). On the asymptotics of penalized splines. *Biometrika* 95(2), 415–436.
- Mack, T. (1993). Distribution-free calculation of the standard error of chain ladder reserve estimates. *ASTIN Bulletin: The Journal of the IAA* 23(2), 213–225.
- Marra, G. and S. N. Wood (2012). Coverage properties of confidence intervals for generalized additive model components. *Scandinavian Journal of Statistics* 39(1), 53–74.
- McCullagh, P. et al. (2002). What is a statistical model? *Annals of Statistics* 30(5), 1225–1310.
- McCullagh, P. and J. Nelder (1989). *Generalized Linear Models (2nd ed.)*. Chapman and Hall, London.
- McSwiggan, G., A. Baddeley, and G. Nair (2017). Kernel density estimation on a linear network. *Scandinavian Journal of Statistics* 44(2), 324–345.
- Møller, J. and R. P. Waagepetersen (2003). *Statistical inference and simulation for spatial point processes*. CRC Press.
- Mörters, P. and Y. Peres (2010). *Brownian motion*, Volume 30. Cambridge University Press.
- Nelder, J. A. and R. W. Wedderburn (1972). Generalized linear models. *Journal of the Royal Statistical Society: Series A (General)* 135(3), 370–384.
- Ntzoufras, I., V. Palaskas, and S. Drikos (2019). Bayesian models for prediction of the set-difference in volleyball. *arXiv preprint arXiv:1911.04541*.
- Okabe, A., T. Satoh, and K. Sugihara (2009). A kernel density estimation method for networks, its computational method and a GIS-based tool. *International Journal of Geographical Information Science* 23(1), 7–32.
- Okabe, A. and K. Sugihara (2012). *Spatial analysis along networks: Statistical and computational methods*. John Wiley & Sons.

- Pollard, D. (2002). *A user's guide to measure theoretic probability*. Number 8. Cambridge University Press.
- Popper, K. R. (1959). The propensity interpretation of probability. *The British Journal for the Philosophy of Science* 10(37), 25–42.
- R Core Team (2021). *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing.
- Rice, J., M. Rosenblatt, et al. (1983). Smoothing splines: Regression, derivatives and deconvolution. *The Annals of Statistics* 11(1), 141–156.
- Rigby, R. A. and D. M. Stasinopoulos (2005). Generalized additive models for location, scale and shape. *Journal of the Royal Statistical Society: Series C (Applied Statistics)* 54(3), 507–554.
- Ross, S. M., J. J. Kelly, R. J. Sullivan, W. J. Perry, D. Mercer, R. M. Davis, T. D. Washburn, E. V. Sager, J. B. Boyce, and V. L. Bristow (1996). *Stochastic processes*, Volume 2. Wiley New York.
- Ruppert, D. (2002). Selecting the number of knots for penalized splines. *Journal of computational and graphical statistics* 11(4), 735–757.
- Ruppert, D., M. P. Wand, and R. J. Carroll (2003). *Semiparametric regression*. Number 12. Cambridge University Press.
- Schall, R. (1991). Estimation in generalized linear models with random effects. *Biometrika* 78(4), 719–727.
- Shun, Z. and P. McCullagh (1995). Laplace approximation of high dimensional integrals. *Journal of the Royal Statistical Society: Series B (Methodological)* 57(4), 749–760.
- Silverman, B. W. (1986). *Density estimation for statistics and data analysis*, Volume 26. CRC press.
- Skellam, J. G. (1946). The frequency distribution of the difference between two Poisson variates belonging to different populations. *Journal of the Royal Statistical Society* 109(3), 296–296.
- Stasinopoulos, D. M., R. A. Rigby, et al. (2007). Generalized additive models for location scale and shape (gamlss) in r. *Journal of Statistical Software* 23(7), 1–46.
- Tavner, P., J. Xiang, and F. Spinato (2007). Reliability analysis for wind turbines. *Wind Energy: An International Journal for Progress and Applications in Wind Power Conversion Technology* 10(1), 1–18.
- Therneau, T. M. (2021). *A Package for Survival Analysis in R*. R package version 3.2-11.
- Therneau, T. M. and P. M. Grambsch (2000). The Cox model. In *Modeling survival data: extending the Cox model*, pp. 39–77. Springer.
- Vaida, F. and R. Xu (2000). Proportional hazards model with random effects. *Statistics in medicine* 19(24), 3309–3324.

- Valsa, J. and L. Brančik (1998). Approximate formulae for numerical inversion of Laplace transforms. *International Journal of Numerical Modelling: Electronic Networks, Devices and Fields* 11(3), 153–166.
- Velavan, T. P. and C. G. Meyer (2020). The COVID-19 epidemic. *Tropical Medicine & International Health* 25(3), 278.
- Weideman, J. and L. Trefethen (2007). Parabolic and hyperbolic contours for computing the Bromwich integral. *Mathematics of Computation* 76(259), 1341–1356.
- Wickham, H. (2010). A layered grammar of graphics. *Journal of Computational and Graphical Statistics* 19(1), 3–28.
- Wickham, H., M. Averick, J. Bryan, W. Chang, L. D. McGowan, R. François, G. Grolemond, A. Hayes, L. Henry, J. Hester, et al. (2019). Welcome to the tidyverse. *Journal of Open Source Software* 4(43), 1686.
- Widder, D. V. (2015). *Laplace transform (PMS-6)*. Princeton University Press.
- Wilkinson, L. (2012). The grammar of graphics. In *Handbook of Computational Statistics*, pp. 375–414. Springer.
- Wood, S. N. (2003). Thin plate regression splines. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 65(1), 95–114.
- Wood, S. N. (2004). Stable and efficient multiple smoothing parameter estimation for generalized additive models. *Journal of the American Statistical Association* 99(467), 673–686.
- Wood, S. N. (2006). On confidence intervals for generalized additive models based on penalized regression splines. *Australian & New Zealand Journal of Statistics* 48(4), 445–464.
- Wood, S. N. (2011). Fast stable restricted maximum likelihood and marginal likelihood estimation of semiparametric generalized linear models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 73(1), 3–36.
- Wood, S. N. (2017). *Generalized additive models: An introduction with R*. CRC press.
- Wood, S. N. and M. Fasiolo (2017). A generalized Fellner-Schall method for smoothing parameter optimization with application to tweedie location, scale and shape models. *Biometrics* 73(4), 1071–1081.
- Wood, S. N., N. Pya, and B. Säfken (2016). Smoothing parameter and model selection for general smooth models. *Journal of the American Statistical Association* 111(516), 1548–1563.
- Wood, S. N., F. Scheipl, and J. J. Faraway (2013). Straightforward intermediate rank tensor product smoothing in mixed models. *Statistics and Computing* 23(3), 341–360.
- Wright, S. and J. Nocedal (1999). Numerical optimization. *Springer Science* 35(67-68), 7.
- Zhang, Y., M. Roughan, C. Lund, and D. Donoho (2003). An information-theoretic approach to traffic matrix estimation. In *Proceedings of the 2003 conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*, pp. 301–312.

Part I.

**Statistical modeling of spatial network
data**

Chapter 2

Estimation of latent network flows in bike-sharing systems

Contributing Article Schneble, M., Kauermann, G. (2020). Estimation of latent network flows in bike-sharing systems. *Statistical Modelling*. <https://doi.org/10.1177/1471082X20971911>

Code and data <http://www.statmod.org/smij/archive.html>

Copyright Statistical Modelling Society, SAGE Publications Ltd, 2020.

Further versions Schneble, M., Kauermann, G. (July 2019). Estimation of latent network flows in bike-sharing systems. Proceedings of the 34th International Workshop on Statistical Modelling, 1:141-146.

Author Contributions The general idea of modeling latent trips in a bike-sharing network when only the station feeds are known stems from Göran Kauermann. He also had the idea to employ the Skellam distribution therefore and developed the basic statistical model. The contribution of Marc Schneble is given by the extension of the primary considerations by the station-based model, the truncated Skellam model and the treatment of (un)known provider interventions. Moreover, he designed the simulation study in the main part of the paper and was responsible for the covariate design of the model when applied to the Vienna bike-sharing network data. Furthermore, Marc Schneble implemented the model in the **R** language, including data preparation and visualization. The manuscript was mainly written by Marc Schneble. Both authors were involved in extensive proofreading.

Chapter 3

Intensity estimation on geometric networks

3.1. Intensity estimation on geometric networks with penalized splines

Contributing Article Schneble, M., Kauermann, G. (2021). Intensity estimation on geometric networks with penalized splines. *The Annals of Applied Statistics* (to appear).

Code and data <https://github.com/MarcSchneble/NetworkSplines>

Further versions Schneble, M., Kauermann, G. (July 2020). Intensity estimation on geometric networks with penalized splines. Proceedings of the 35th International Workshop on Statistical Modelling, 1:204-209.

Author Contributions The idea of estimating the intensity of network point processes with penalized splines can be attributed to Marc Schneble. He also wrote the major part of the manuscript and was responsible for implementing the model in the **R** language, including data preparation and visualization. Moreover, Marc Schneble conducted the analyses with real data and designed the simulation study. Göran Kauermann was involved in elaborating the model formulation and further proposed the composition of the mean integrated squared error as the sum of the integrated variance and the integrated squared bias. Both authors were involved in extensive proofreading of the manuscript.

Disclaimer At the day of the disputation, the article has not yet been published on the website of The Annals of Applied Statistics. Therefore, the accepted version of the manuscript that has been sent to The Annals of Applied Statistics on July 08, 2021 is attached.

INTENSITY ESTIMATION ON GEOMETRIC NETWORKS WITH PENALIZED SPLINES *

BY MARC SCHNEBLE¹ AND GÖRAN KAUEMANN²

¹Department of Statistics, Ludwig-Maximilians Universität Munich, marc.schneble@stat.uni-muenchen.de

²Department of Statistics, Ludwig-Maximilians Universität Munich, goeran.kauermann@stat.uni-muenchen.de

In the past decades, the growing amount of network data lead to many novel statistical models. In this paper, we consider so-called geometric networks. Typical examples are road networks or other infrastructure networks. Nevertheless, the neurons or the blood vessels in a human body can also be interpreted as a geometric network embedded in a three-dimensional space. A network-specific metric rather than the Euclidean metric is usually used in all these applications, making the analyses of network data challenging. We consider network-based point processes, and our task is to estimate the intensity (or density) of the process, which allows us to detect high- and low-intensity regions of the underlying stochastic processes. Available routines that tackle this problem are commonly based on kernel smoothing methods. This paper uses penalized spline smoothing and extends this towards smooth intensity estimation on geometric networks. Furthermore, our approach easily allows incorporating covariates, enabling us to respect the network geometry in a regression model framework. Several data examples and a simulation study show that penalized spline-based intensity estimation on geometric networks is a numerically stable and efficient tool. Furthermore, it also allows estimating linear and smooth covariate effects, distinguishing our approach from already existing methodologies.

1. Introduction. In statistical network analysis, a (static) network is usually considered as a graph that is characterized by a set of vertices which are connected by a set of, possibly weighted, edges (Kolaczyk and Csárdi, 2014). In this matter, the interest usually lies in the mutual relationship and the dependencies of the vertices, sometimes called “actors” (Snijders, 1996), that are induced by the edges. For a general overview of this research area, see e.g. Goldenberg et al. (2010). In the context of this paper, a network is rather considered as a geometric object embedded in a Euclidean space. We use the term “geometric network” and a typical example is a network of streets. The setting is that we observe a spatial point process on the network edges and focus is on estimating the intensity (or density) of this process.

Regarding the data structure, the question arises why one should analyze data points on a geometric network and not in the Euclidean space itself. To illustrate this, consider a point pattern that seems to be clustered in the plane. However, the points might be uniformly distributed on a network where many network segments are clustered within a small area. A typical example is the distribution of traffic accidents in an urban area (McSwiggan, Baddeley and Nair, 2017). Theoretically, such events can only occur on a network of streets which is often considered being embedded in the plane. Therefore, the statistical analyses of point patterns distributed across a Euclidean space and point patterns distributed only on a geometric network are tremendously different.

*We would like to thank the elite graduate program Data Science at LMU Munich and the Munich Center for Machine Learning (MCML) for funding.

Keywords and phrases: Intensity estimation of stochastic point processes, generalized additive models, geometric networks, penalized splines, poisson regression with offset, spatstat package

Due to the increasing availability of network-based data, the last 25 years have seen a broad range of literature concerned with network-based point processes. Amongst the first statistical analyses of spatial point patterns on a network were proposed by [Okabe, Yomono and Kitamura \(1995\)](#), [Okabe and Yamada \(2001\)](#) and [Spooner et al. \(2004\)](#). They all noted that in the context of geometric network data, the Euclidean distance needs to be replaced by the shortest path distance to respect the network geometry. This resulted e.g. in the geometrically corrected network K-function ([Ang, Baddeley and Nair, 2012](#)), a modified version of Ripley's K-function in two dimensions. The network K-function can be used to analyze the correlation structure of point patterns on a network. [Baddeley, Rubak and Turner \(2015\)](#) discuss the topic in general and especially from an application point of view. Among other contributions, a huge library of functions is provided to create, manipulate and analyze both a point pattern on a linear network, embedded in the plane, and the network itself. Furthermore, [Baddeley, Rubak and Turner \(2015\)](#) also treat marked point processes, intensities of point processes depending on covariates and point processes on trees which are networks without loops. Marked point processes on directed linear networks are further discussed by [Rasmussen and Christensen \(2020\)](#) for various kinds of stochastic point processes. Finally, we would like to highlight [Baddeley et al. \(2020\)](#) who provide a broad overview on how to analyze point patterns on linear networks.

The focus of this paper is on intensity estimation in geometric networks. [Borruso \(2008\)](#) and [Xie and Yan \(2008\)](#) developed kernel density estimation on a network geometry which is performed by respecting the shortest path distance. However, both articles did not consider that around vertices with more than two adjoining segments, there is more network mass within a certain shortest path distance. Hence, this approach leads to biased estimates, especially if the point pattern is distributed according to a uniform distribution on the network. [Okabe, Satoh and Sugihara \(2009\)](#) solved this problem by introducing equal-split (dis-)continuous kernel density estimation. The idea is to split the mass of the kernel functions equally across all other segments that depart from a vertex when approaching this vertex from one side. The approach was refined in [McSwiggan, Baddeley and Nair \(2017\)](#). Instead of a finite sum of paths over the network, they consider an infinite sum leading to a diffusion estimate that can be computed via a heat equation on the network. Furthermore, [Moradi, Rodríguez-Cortés and Mateu \(2018\)](#) showed in their application that an extension of Diggle's ([Diggle, 1985](#)) non-parametric edge-corrected kernel-based intensity estimator is superior to the equal-split discontinuous estimator that was proposed by [Okabe, Satoh and Sugihara \(2009\)](#). Most recently, [Rakshit et al. \(2019\)](#) proposed to perform kernel smoothing on networks making use of a two-dimensional kernel that is still robust against errors in network geometry. This approach is especially well suited in scenarios of vast networks.

All the models discussed so far are kernel-based and produce continuous intensity estimates. However, there are various non kernel-based approaches where the fitted intensity is not continuous. To begin with the special case of a river network, which results as a directed and acyclic geometric network, [O'Donnell et al. \(2014\)](#) used penalized piecewise constant functions for estimating the river flow, which can be interpreted as penalized splines (P-splines, [Eilers and Marx, 1996](#)) of order 0. The paper was reviewed by [Rushworth et al. \(2015\)](#) who implemented the theory in the **R** package `smnet`. Fused density estimation was proposed by [Bassett and Sharpnack \(2019\)](#) to estimate the density on a geometric network. This estimator is the solution to a total variation regularized maximum-likelihood problem. Another very recent method that does also not result in continuous estimates is the smoothed Voronoi estimate ([Moradi et al., 2019](#)). Therefore, the Voronoi estimate ([Barr and Schoenberg, 2010](#)) is computed several times while only retaining a fraction f of the data points in each iteration. The smoothed Voronoi estimate is then equal to the average over the re-scaled intensities.

When referring again to kernel-based methods, [Eilers and Marx \(1996\)](#) argued that kernel density estimators of points on the real line suffer from boundary effects, e.g. if the domain of the data is not specified correctly. Instead, the authors proposed to estimate the density by making use of penalized splines in order to smooth the histogram that is created by binning the data with small bin widths. With time, this concept has been extended, among others, to allow for density estimation of multiple dimensional data ([Currie, Durban and Eilers, 2006](#)) or to represent the density as a mixture of weighted penalized spline densities ([Schellhase and Kauermann, 2012](#)). A comprehensive survey of penalized spline theory and its application is given in [Eilers, Marx and Durbán \(2015\)](#). Generally, penalized spline estimation has become a major workhorse in statistical modeling as demonstrated in [Ruppert, Wand and Carroll \(2003, 2009\)](#).

In this paper, we extend the penalized spline-based intensity estimation approach of [Eilers and Marx \(1996\)](#) to work on geometric networks. The focus is to estimate the intensity (or density) of a point process on the network, given realizations of this process as data. In contrast to the intensity estimation methods summarized above, our approach allows us to estimate the smooth baseline intensity and the effects of network internal and external covariates. The embedding in the context of generalized additive models further enables us to assess the uncertainty of both the network intensity and covariate effects. Besides this manuscript our major contribution is the implementation¹ of our methodology in **R** ([R Core Team, 2013](#), version 4.1.0) for linear networks, where a lot of the functionality is based on the family of `spatstat` packages ([Baddeley, Rubak and Turner, 2015](#), version 2.2-0).

The remainder of this paper is structured as follows. Section 2 introduces some basic notation related to network graphs, geometric networks and stochastic point processes on networks. Section 3 treats our new methodology to estimate the intensity of a point process on a geometric network with penalized splines. This is followed by Section 4, the presentation of the networks and the data which we employ in this paper. Sections 5 - 7 cover applications to real data while Section 8 explores the performance of our model when fitted to simulated data. Section 9 concludes the paper and discusses possible supplements of our model.

2. Notation and Problem. Consider a set $V = \{v_1, \dots, v_W\}$ of $W \in \mathbb{N}$ elements which we call vertices, where $\mathbb{N} = \{1, 2, 3, \dots\}$ denotes the set of positive integers. Further, let $E = \{e_1, \dots, e_M\} \subset V \times V$ be a set of $M \in \mathbb{N}$ pairs $e_m = (v_i, v_j)$ which we call edges. Putting these together leads to the network graph $L = (V, E)$ and we denote L as the graph representation of the network defined by a set of vertices V and a set of edges E . In this paper, we only consider undirected networks, i.e. there is an edge from v_i to v_j if and only if there is an edge from v_j to v_i . An edge e is called incident to a vertex v if there is another vertex $v_i \in V (v_i \neq v)$ such that $e = (v, v_i) \in E$. The degree of a vertex v , denoted by $\deg(v)$ is defined as the count of edges which are incident to v and for our purpose we always remove a vertex v from V if $\deg(v) = 0$.

A geometric network also typically exhibits a geometric representation as a subset of a Euclidean space \mathbb{R}^q for $q \geq 2$. In this case, the set of vertices V in the network graph representation can be viewed as a set $\mathbf{V} = \{\mathbf{v}_1, \dots, \mathbf{v}_W\}$ of vectors with $\mathbf{v}_i \in \mathbb{R}^q$ for $i = 1, \dots, W$. Consequently, the edges can be viewed as a set $\mathbf{E} = \{e_1, \dots, e_M\}$ of network segments with each $e_m \subset \mathbb{R}^q$ being the connection between two vertices \mathbf{v}_i and \mathbf{v}_j . More generally, such an edge can be described by the image set $e_m = \nu_m([a_m, b_m])$ of a parametric curve ([Heuser, 2006](#)) $\nu_m : [a_m, b_m] \rightarrow \mathbb{R}^q$ with $a_m < b_m, \nu_m(a_m) = \mathbf{v}_i$ and $\nu_m(b_m) = \mathbf{v}_j$, where the length

¹The **R** code that can be used to reproduce the results in this paper can be downloaded from <https://github.com/MarcSchneble/NetworkSplines>.

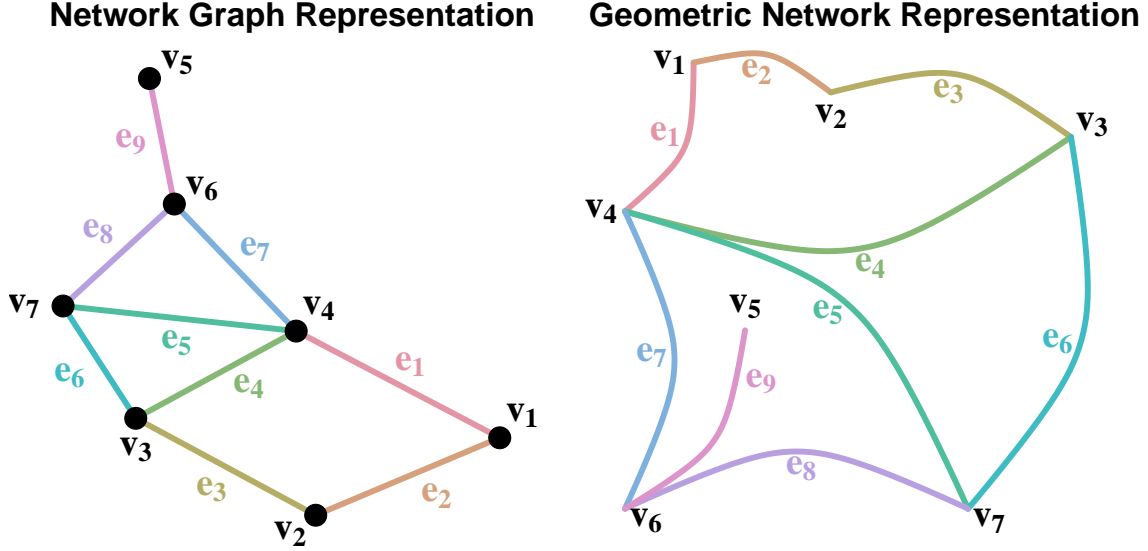


FIG 1. Two different representations of a network: Left panel: Network graph representation L . Right Panel: Geometric network representation \mathbf{L} .

of the curve segment e_m is given by

$$(1) \quad d_m = |e_m| = \lim_{N \rightarrow \infty} \sum_{i=1}^N \|\nu_m(t_i) - \nu_m(t_{i-1})\|_q,$$

with $t_i = a_m + i(b_m - a_m)/N$ for $i = 1, \dots, N$ and $\|\cdot\|_q$ denotes the Euclidean distance in \mathbb{R}^q . Thus, (1) already suggest to approximate a parametric curve by a number of N straight line segments with endpoints $\nu_m(t_0), \dots, \nu_m(t_N)$. This representation is used in the **R** package `spatstat` (Baddeley, Rubak and Turner, 2015), which allows to analyze linear networks in the plane.

We can now define the geometric representation of a network graph L as $\mathbf{L} = \bigcup_{m=1}^M e_m \subset \mathbb{R}^q$. Hence, there is a one-to-one correspondence between L and \mathbf{L} which we exploit consistently in this paper. According to the network graph representation, we also define a vertex degree for the geometric network representation, meaning that $\deg(v)$ denotes the count of segments which have an endpoint equal to v . Furthermore, the lengths d_m of the curves e_m from (1) imply a metric $d_{\mathbf{L}} : \mathbf{L} \times \mathbf{L} \rightarrow [0, \infty)$ on \mathbf{L} . More precisely, $d_{\mathbf{L}}(z_1, z_2)$ denotes the shortest path distance between two points z_1, z_2 on \mathbf{L} and with $[z_1; z_2] \subset \mathbf{L}$ or with $[z_1; z_2] \subset \mathbf{L}$ we denote the corresponding path, where a round bracket indicates that an endpoint is not contained in the set. The total length of the geometric network is $|\mathbf{L}| = \sum_{m=1}^M d_m$. If the network is not connected, i.e. the corresponding network graph L consists of more than one connected component, we can use the extended metric (Beer, 2013) $d_{\mathbf{L}} : \mathbf{L} \times \mathbf{L} \rightarrow [0, \infty) \cup \infty$. In this case, the same methodology can be applied unmodified.

Also note that the geometric representation \mathbf{L} is not necessarily unique which can be seen from the following consideration: Let v be a vertex (in network graph representation) with exactly two incident edges $e_m = (v_i, v)$ and $e_n = (v, v_j)$, i.e. $\deg(v) = 2$ and $(v_i, v_j) \notin E$. If we remove v from V as well as e_m, e_n from E but add the edge $e = (v_i, v_j)$ to E , the network graph representation of $L = (V, E)$ has changed. In the geometric representation, we can remove v from \mathbf{V} as well as e_m, e_n from \mathbf{E} and add the segment $e = e_m \cup e_n$ to \mathbf{E} which does not change \mathbf{L} . To exemplify the notation introduced in this section, Figure 1 shows a small network as a network graph (left panel) and as a geometric network embedded in the plane (right panel). The figure also visualizes that the role of a vertex v in the geometric

network representation is merely being the endpoint of $\deg(\mathbf{v})$ curves. Hence, in this network the vertex \mathbf{v}_2 could be removed from \mathbf{L} without changing its geometric representation.

We now consider the following setting, see also [McSwiggan, Baddeley and Nair \(2017\)](#). Let \mathcal{X} be a stochastic point process on the geometric network \mathbf{L} with continuous intensity $\varphi_{\mathcal{X}} : \mathbf{L} \rightarrow [0, \infty)$. The expected number of points in a set $\mathbf{K} \subset \mathbf{L}$ is then defined through

$$\int_{\mathbf{K}} \varphi_{\mathcal{X}}(z) dz = \sum_{m=1}^M \int_{\mathbf{K} \cap e_m} \varphi_{\mathcal{X}}(z) d_{|m} z,$$

where $d_{|m} z$ denotes integration with respect to the curve e_m . Our aim is to estimate the intensity of the point process \mathcal{X} on \mathbf{L} given that we observe realizations $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n$ of this process. The point process \mathcal{X} can equivalently be defined through a density function $f_{\mathcal{X}} : \mathbf{L} \rightarrow [0, \infty)$. The probability that a random point $\mathbf{X}_i \sim f_{\mathcal{X}}$ falls into a subset $\mathbf{K} \subset \mathbf{L}$ is then given by $\mathbb{P}(\mathbf{X}_i \in \mathbf{K}) = \int_{\mathbf{K}} f_{\mathcal{X}}(z) dz$.

3. Methodology.

3.1. B-Splines on a Network. First, we briefly review B-splines (compare [Ruppert, Wand and Carroll, 2003](#) or [Fahrmeir et al., 2013](#)). To start, assume a simple point process \mathcal{X} with intensity $\varphi_{\mathcal{X}}(z)$, where z is univariate and takes values in the bounded interval $[a, b]$. The goal is to estimate $\varphi_{\mathcal{X}}(z)$ in a smooth and flexible way. To do so, we approximate the logarithmized intensity $\nu_{\mathcal{X}} = \log \varphi_{\mathcal{X}}$ through a B-spline basis representation $\nu_{\mathcal{X}}(z) = \sum_{j=1}^J \gamma_j B_j^l(z)$, where $B_j^l(\cdot)$ are B-splines of order $l \in \mathbb{N}_0$ and $\gamma = (\gamma_1, \dots, \gamma_J)^\top$ is a vector of regression coefficients that needs to be estimated from the data. For the construction of B-splines, we use I interior knots $a = \tau_1 < \dots < \tau_I = b$. The $J = I + l - 1$ basis functions $B_j^l(\cdot)$ are each locally supported on $l + 2$ adjacent knots and can be calculated recursively from lower order basis functions ([De Boor, 1972](#)). An important property of a B-spline basis is that $\sum_{j=1}^J B_j^l(z) = 1$ holds for $z \in [a, b]$ and any order of B-splines l . This property also needs to be respected in the geometric network case.

Subsequently, we restrict ourselves to linear B-spline bases for simplicity of presentation. For simplicity of notation we drop the superscript l in the B-spline notation, i.e. we construct B-splines of order $l = 1$ on a geometric network \mathbf{L} . Such a basis can be constructed straightforwardly using the one-dimensional definitions from above. On every curve e_m , which has endpoints \mathbf{v}_i and \mathbf{v}_j , we specify an equidistant sequence of I_m knots $\mathbf{v}_i = \tau_{m,1}, \dots, \tau_{m,I_m} = \mathbf{v}_j$ with $\tau_{m,k} \in e_m$ for $k = 1, \dots, I_m$, where $d_{\mathbf{L}}(\tau_{m,k}, \tau_{m,k-1}) = \delta_m$. Note that a knot which is equal to a vertex \mathbf{v} is contained in the knot sequence of $\deg(\mathbf{v})$ segments but it still represents the same knot. Other than in the one-dimensional setup, it is in general not possible to choose the set of knots to be equidistant on the entire geometric network \mathbf{L} with respect to all curve lengths d_m . However, we may choose a global knot distance δ such that it is close to an equidistant allocation of knots on the entire geometric network. Let therefore $\lceil \cdot \rceil$ denote the upwards rounded integer and $\lfloor \cdot \rfloor$ the corresponding downwards rounded integer. We then define

$$(2) \quad \delta_m = \begin{cases} d_m / \lfloor \frac{d_m}{\delta} \rfloor, & \frac{d_m}{\delta} - \lfloor \frac{d_m}{\delta} \rfloor < 0.5 \\ d_m / \lceil \frac{d_m}{\delta} \rceil, & \frac{d_m}{\delta} - \lfloor \frac{d_m}{\delta} \rfloor \geq 0.5 \end{cases},$$

which leads to curve-specific knot distances δ_m which are as similar as possible for a given overall knot distance δ . Generally, we will choose δ rather small such that the differences between the δ_m are small and can be considered as negligible. This will become more clear later, when we also introduce a penalization component in the estimation.

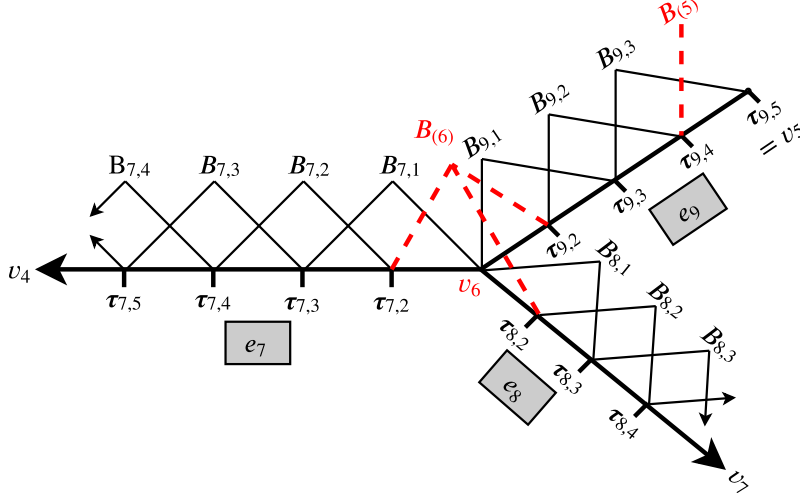


FIG 2. Schematic representation of a linear B-spline basis around v_6 of Figure 1. Here, $v_6 = \tau_{7,1} = \tau_{8,1} = \tau_{9,1}$ with adjacent segments e_7, e_8, e_9 . The red dotted lines show the linear B-splines which are contained in B_v . The peaks of all B-splines are equal to 1 as in the Euclidean setting.

Having the set of knots defined as above, we can construct a linear B-spline basis B on the geometric network \mathbf{L} . First, we use for every segment e_m with endpoints v_i and v_j the equidistant sequence of knots $v_i = \tau_{m,1}, \dots, \tau_{m,I_m} = v_j$ from above to construct $J_m = I_m - 2$ linear B-splines $B_{m,1}, \dots, B_{m,J_m}$. These B-splines are defined accordingly to the univariate case by

$$(3) \quad B_{m,k}(z) = \frac{d_{\mathbf{L}}(z, \tau_{m,k})}{\delta_m} \mathbb{1}_{[\tau_{m,k}, \tau_{m,k+1})}(z) + \frac{d_{\mathbf{L}}(\tau_{m,k+2}, z)}{\delta_m} \mathbb{1}_{[\tau_{m,k+1}, \tau_{m,k+2})}(z)$$

for $z \in \mathbf{L}, m = 1, \dots, M$ and $k = 1, \dots, J_m$. Therefore, the B-splines $B_{m,k}$ are only supported on e_m and we denote with $B_e = \{B_{m,1}, \dots, B_{m,J_m} \mid m = 1, \dots, M\}$ the set of all these B-splines. We further require that $J_m \geq 1$ for all $m = 1, \dots, M$ which is fulfilled if $\delta_m \leq \frac{d_m}{2}$ for all m . If δ_m from (2) does not fulfill this constraint, we set $\delta_m = \frac{d_m}{2}$.

In addition to the B-splines defined by (3) we construct a single B-spline around each vertex $v_i \in \mathbf{V}$. Therefore, we consider the $\deg(v_i)$ segments which have an endpoint equal to v_i and we numerate them (without loss of generality) with $e_1, \dots, e_{\deg(v_i)}$. Again, without loss of generality, let $v_i = \tau_{1,1} \dots, \tau_{\deg(v_i),1}$, i.e. we order the knots such that the first knot of every segment starting in v_i equals v_i itself, see Figure 2 as example. Then, we define the vertex specific B-spline $B_{(i)}$ for vertex v_i by

$$(4) \quad B_{(i)}(z) = \mathbb{1}_{\{v_i\}}(z) + \sum_{k=1}^{\deg(v_i)} \left[1 - \frac{d_{\mathbf{L}}(v_i, z)}{\delta_k} \right] \mathbb{1}_{(v_i; \tau_{k,2})}(z).$$

for $z \in \mathbf{L}$ and $i = 1, \dots, W$. These B-splines have support $\text{supp}(B_{(i)}) = \bigcup_{k=1}^{\deg(v_i)} [v_i; \tau_{k,2}) \subset \mathbf{L}$, i.e. they are supported on $\deg(v_i)$ segments. Note that all summands in (4) are nonnegative and at most one of the summands is positive. This set of B-splines is denoted with $B_v = \{B_{(1)}, \dots, B_{(W)}\}$. Altogether, we specify the linear B-spline basis on \mathbf{L} by $B = B_e \cup B_v$ with dimension $J = |B| = \sum_{m=1}^M J_m + W$. For simplicity of presentation, we index from now on the B-spline Basis by $1, \dots, J$ and by construction, it holds that $\sum_{j=1}^J B_j(z) = 1$ for $z \in \mathbf{L}$. In Figure 2, we depict linear B-splines around the vertex v_6 with $\deg(v_6) = 3$ of the network that is shown Figure 1.

3.2. *Intensity Estimation on a Network.* We can now easily adopt the density estimation approach proposed by [Eilers and Marx \(1996\)](#) for univariate data. On our geometric network \mathbf{L} , we specify a bin width h_m on every segment e_m and then divide e_m into $N_m = \frac{d_m}{h_m}$ bins of the same length such that \mathbf{L} is partitioned into $N = \sum_{m=1}^M \frac{d_m}{h_m}$ bins in total. As for the knot distances δ_m it is clear, that h_m can not be the same for all curve segments of \mathbf{L} . However, also the bin widths are chosen very small when performing intensity estimation with penalized splines. We therefore specify a small global bin width h and define accordingly to (2)

$$h_m = \begin{cases} d_m / \lfloor \frac{d_m}{h} \rfloor, & \frac{d_m}{h} - \lfloor \frac{d_m}{h} \rfloor < 0.5 \\ d_m / \lceil \frac{d_m}{h} \rceil, & \frac{d_m}{h} - \lfloor \frac{d_m}{h} \rfloor \geq 0.5 \end{cases}.$$

If the left endpoint of e_m is \mathbf{v} , the bins are given by the N_m subsets $[\mathbf{b}_{m,k-1}; \mathbf{b}_{m,k}) \subset e_m$ for $k = 1, \dots, N_m$, where $\mathbf{b}_{m,k} \in e_m$ satisfies $d_{\mathbf{L}}(\mathbf{v}, \mathbf{b}_{m,k}) = kh_m$ and $\mathbf{b}_{m,0} = \mathbf{v}$. Each bin is characterized by its midpoint $\mathbf{z}_{m,k}$ which satisfies $d_{\mathbf{L}}(\mathbf{b}_{m,k-1}, \mathbf{z}_{m,k}) = d_{\mathbf{L}}(\mathbf{z}_{m,k}, \mathbf{b}_{m,k})$.

Assume now that data on n independently observed points \mathbf{x}_i of the point process \mathcal{X} on the geometric network have been observed, with $i = 1, \dots, n$. We assume that the observed points are not equal to the network's vertices for identifiability reasons, i.e. each point lies on a single edge. We define with $y_{m,k} \in \mathbb{N} \cup \{0\}$ the number of observations which are contained in the k -th bin of the m -th segment, i.e.

$$y_{m,k} = \#\{\mathbf{x}_i \in \mathbf{L} \mid \mathbf{x}_i \in [\mathbf{b}_{m,k-1}; \mathbf{b}_{m,k}), i = 1, \dots, n\}$$

for $m = 1, \dots, M$ and $k = 1, \dots, N_m$. Based on our considerations for the point process \mathcal{X} we assume a Poisson distribution for the counts $y_{m,k}$ such that we have

$$(5) \quad y_{m,k} \mid \mathbf{z}_{m,k} \stackrel{\text{indep.}}{\sim} \text{Poi}(\lambda_{m,k}),$$

where $\lambda_{m,k}$ is approximated through

$$(6) \quad \lambda_{m,k} = \varphi_{\mathcal{X}}(\mathbf{z}_{m,k}) \cdot h_m = \exp(\nu_{\mathcal{X}}(\mathbf{z}_{m,k}) + \log h_m).$$

We can consider $\log h_m$ as offset and aim to estimate $\nu_{\mathcal{X}}(\mathbf{z})$ as continuous log-intensity for $\mathbf{z} \in \mathbf{L}$ treating the pairs $(y_{m,k}, \mathbf{z}_{m,k})$ as independent observations from (5). Therefore, we replace $\nu_{\mathcal{X}}(\mathbf{z})$ through the B-spline basis representation

$$(7) \quad \nu_{\mathcal{X}}(\mathbf{z}) = \sum_{j=1}^J B_j(\mathbf{z})\gamma_j = \mathbf{B}(\mathbf{z})\boldsymbol{\gamma},$$

where $\mathbf{B}(\mathbf{z}) = (B_1(\mathbf{z}), \dots, B_J(\mathbf{z}))$ is a row vector consisting of the B-spline basis from above evaluated at $\mathbf{z} \in \mathbf{L}$ and $\boldsymbol{\gamma} = (\gamma_1, \dots, \gamma_J)^\top$ is the vector of B-spline coefficients that needs to be estimated from the data $\mathbf{x}_1, \dots, \mathbf{x}_n$. Imposing a penalty on the resulting Poisson likelihood leads to the penalized log-likelihood (constant terms are ignored)

$$(8) \quad \ell_{\mathcal{P}}(\boldsymbol{\gamma}; \rho) = \sum_{m=1}^M \sum_{k=1}^{N_m} [y_{m,k} \log \lambda_{m,k} - \lambda_{m,k}] - \rho \mathcal{P}_r(\boldsymbol{\gamma}),$$

where $\mathcal{P}_r(\boldsymbol{\gamma})$ is a penalty which is defined in the next section and ρ is the smoothing parameter. The estimation of the smoothing parameter is treated later in this section.

If we replace $\boldsymbol{\gamma}$ in (7) with the maximum-likelihood estimate $\hat{\boldsymbol{\gamma}} = \arg\max_{\boldsymbol{\gamma}} \ell_{\mathcal{P}}(\boldsymbol{\gamma}; \rho)$, then $\hat{\nu}_{\mathcal{X}}(\mathbf{z}) = \mathbf{B}(\mathbf{z})\hat{\boldsymbol{\gamma}}$ is an estimate of the log-intensity and thus $\hat{\varphi}_{\mathcal{X}}(\mathbf{z}) = \exp(\hat{\nu}_{\mathcal{X}}(\mathbf{z}))$ is an estimate of the intensity of the point process \mathcal{X} for $\mathbf{z} \in \mathbf{L}$. Also note that for a given n an estimate of the density of \mathcal{X} is given by $\hat{f}_{\mathcal{X}}(\mathbf{z}) = \hat{\varphi}_{\mathcal{X}}(\mathbf{z})/n$.

3.3. Penalties on a Network. In order to control the smoothness of the intensity estimate and to overcome singularity issues, the penalty $\mathcal{P}_r(\gamma)$ multiplied with the smoothing parameter ρ is subtracted from the maximum likelihood criterion, which leads to the penalized log-likelihood (8). In the one-dimensional setting Eilers and Marx (1996) proposed to impose a penalty on the vector of coefficients γ that is proportional to the r -th order differences of adjacent spline coefficients. The penalty is given by $\sum_{j=r+1}^J (\Delta^r \gamma_j)^2$ and for $r = 1$ we have $\Delta^1 \gamma_j = \gamma_j - \gamma_{j-1}$. Higher order penalty terms can be calculated recursively using $\Delta^r(\gamma_j) = \Delta^1 \Delta^{r-1} \gamma_j$ starting with the first order differences $\Delta^1 \gamma_j$. It is straightforward to extend this idea to penalties on a network. Let $i, j = 1, \dots, J$ where J is the dimension of the B-spline basis on the geometric network. According to the one-dimensional case, we are interested in the set of pairwise adjacent B-splines or their coefficients, respectively. Hence, we can view the B-splines on the geometric network L itself as a network graph L_B which is defined through a $J \times J$ adjacency matrix \mathbf{A} . From the definition of the linear B-splines, it follows that $\mathbf{A}(i, j) = 1$, if $\text{supp}(B_{(i)}) \cap \text{supp}(B_{(j)}) \neq \emptyset$ and else $\mathbf{A}(i, j) = 0$. In order to define penalties of arbitrary order, we need the $J \times J$ shortest path matrix $\mathbf{S}_\mathbf{A}$ where $\mathbf{S}_\mathbf{A}(i, j) = s$, if the B-Splines $B_{(i)}$ and $B_{(j)}$ have minimum distance s in L_B . This all-pairs shortest path problem can be solved with complexity $\mathcal{O}(J|\mathbf{A}|)$ where $|\mathbf{A}|$ is the number of non-zero entries in \mathbf{A} (Chan, 2012). For illustration, consider again Figure 2. Here, $B_{7,1}$ is adjacent to $B_{7,2}$ as well as $B_{(6)}$ and the shortest path from $B_{7,2}$ to $B_{8,1}$ via $B_{7,1}$ and $B_{(6)}$ has length 3 in L_B .

Now, let $\mathcal{D}_1 = \{(i, j) \mid \mathbf{S}_\mathbf{A}(i, j) = 1, 1 \leq i < j \leq J\}$. According to Eilers and Marx (1996) we penalize neighboring coefficients. A first order penalty is then defined by

$$(9) \quad \mathcal{P}_1(\gamma) = \sum_{\mathcal{D}_1} (\gamma_i - \gamma_j)^2 = (\mathbf{D}_1 \gamma)^\top (\mathbf{D}_1 \gamma) = \gamma^\top \mathbf{K}_1 \gamma,$$

where $\mathbf{D}_1 \in \mathbb{Z}^{|\mathcal{D}_1| \times J}$ and $\mathbf{K}_1 = \mathbf{D}_1^\top \mathbf{D}_1 \in \mathbb{Z}^{J \times J}$ define the difference matrix and the resulting quadratic form according to the pairwise differences in (9). Further, let

$$\mathcal{D}_2 = \{(i, k, j) \mid \mathbf{S}_\mathbf{A}(i, j) = 2, \mathbf{S}_\mathbf{A}(i, k) = \mathbf{S}_\mathbf{A}(k, j) = 1, 1 \leq i < j \leq J\}.$$

Therewith, a second order penalty can be defined by

$$(10) \quad \mathcal{P}_2(\gamma) = \sum_{\mathcal{D}_2} ((\gamma_i - \gamma_k) - (\gamma_k - \gamma_j))^2 = \sum_{\mathcal{D}_2} (\gamma_i - 2\gamma_k + \gamma_j)^2 = (\mathbf{D}_2 \gamma)^\top (\mathbf{D}_2 \gamma) = \gamma^\top \mathbf{K}_2 \gamma,$$

where $\mathbf{D}_2 \in \mathbb{Z}^{|\mathcal{D}_2| \times J}$ and again, $\mathbf{K}_2 = \mathbf{D}_2^\top \mathbf{D}_2 \in \mathbb{Z}^{J \times J}$ results as matrix version from the sum in (10). For illustration, we revisit the B-splines which we depicted in Figure 2, but, restricted to the B-splines $B_1 = B_{7,2}$, $B_2 = B_{7,1}$, $B_3 = B_{(6)}$, $B_4 = B_{8,1}$ and $B_5 = B_{9,1}$. Thus, for $\gamma = (\gamma_1, \dots, \gamma_5)$ the first- and second order penalties $\mathcal{P}_1(\gamma)$ and $\mathcal{P}_2(\gamma)$ are defined by the difference matrices

$$\mathbf{D}_1 = \begin{pmatrix} 1 & -1 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & 1 & 0 & -1 \end{pmatrix} \quad \text{and} \quad \mathbf{D}_2 = \begin{pmatrix} 1 & -2 & 1 & 0 & 0 \\ 0 & 1 & -2 & 1 & 0 \\ 0 & 1 & -2 & 0 & 1 \\ 0 & 0 & -2 & 1 & 1 \end{pmatrix},$$

respectively. By taking advantage of the shortest path matrix $\mathbf{S}_\mathbf{A}$ we can define penalties of any order r , but usually first and second order differences are used when applying penalized splines.

3.4. Estimation of the Smoothing Parameter. For the estimation of the smoothing parameter ρ , we apply the generalized Fellner-Schall method (Wood and Fasiolo, 2017) which is an iterative procedure to estimate the smoothing parameter in generalized additive models (Wood, 2017). The idea behind the Fellner-Schall method is to apply a mixed model approach

and to optimize the log Laplace approximate marginal likelihood of the model with respect to the smoothing parameter. In each iteration step the estimated model parameters $\hat{\gamma}_\rho$ are obtained by maximizing the penalized log-likelihood (8) while treating ρ from the previous cycle as fixed. Then, the update ρ_{new} is calculated through

$$(11) \quad \rho_{\text{new}} = \rho \frac{\text{tr}((\rho \mathbf{K}_r)^- \mathbf{K}_r) - \text{tr}((\mathbf{B}^\top \mathbf{W}(\hat{\gamma}_\rho) \mathbf{B} + \rho \mathbf{K}_r)^- \mathbf{K}_r)}{\hat{\gamma}_\rho^\top \mathbf{K}_r \hat{\gamma}_\rho}$$

where $\text{tr}(\cdot)$ denotes the trace operator and $(\rho \mathbf{K}_r)^-$ denotes a generalized inverse of $\rho \mathbf{K}_r$. The calculation of \mathbf{K}_r^- is numerical demanding if the dimension of the parameter vector γ is large. However, using standard linear algebra tools we can show that $\text{tr}((\rho \mathbf{K}_r)^- \mathbf{K}_r) = \text{rk}(\mathbf{K}_r)/\rho$, where $\text{rk}(\cdot)$ denotes the rank operator. Thus, the calculation of \mathbf{K}_r^- is not necessary. The design matrix $\mathbf{B} \in \mathbb{R}^{N \times J}$ of the Poisson model is build by storing the row vectors $\mathbf{B}(z_{m,k})$, which are defined accordingly to (7) for $m = 1, \dots, M$ and $k = 1, \dots, N_m$, as a matrix. Furthermore, $\mathbf{W}(\hat{\gamma}_\rho) = \text{diag}(\hat{\lambda}_{1,1}, \dots, \hat{\lambda}_{1,N_1}, \dots, \hat{\lambda}_{M,1}, \dots, \hat{\lambda}_{M,N_M})$ is a weight matrix, where $\hat{\lambda}_{m,k} = \exp(\hat{\gamma}_\chi(z_{m,k}) + \log h_m)$ is defined through (6). The matrix $\mathbf{B}^\top \mathbf{W}(\hat{\gamma}_\rho) \mathbf{B} + \rho \mathbf{K}_r$ is the Fisher information of our model and is therefore positive definite, which guarantees that $\rho_{\text{new}} > 0$ (Wood and Fasiolo, 2017). The iterative procedure stops, if ρ_{new} in (11) differs only slightly from the previous ρ .

3.5. Quantification of Uncertainty. Since (6) directly relates to the setting of a generalized additive model we can use already existing GAM theory in order to assess the uncertainty of our predictions. Therefore, we follow the functionality of the `mgcv` package (Wood, 2017, Version 1.8-33) in **R** which produces standard errors based on the Bayesian posterior covariance matrix $\mathbf{V} = \mathbf{V}(\hat{\gamma})$ of the model parameters. Treating the smoothing parameter ρ as fixed this covariance matrix is given as the inverse of the Fisher information from above, i.e.

$$\mathbf{V} = (\mathbf{B}^\top \mathbf{W}(\hat{\gamma}_\rho) \mathbf{B} + \rho \mathbf{K}_r)^{-1}.$$

3.6. Intensity Depending on Covariates. The methodology from above can easily be extended to allow the intensity to depend on one or several covariates, distinguishing between two kinds of covariates. Firstly, we denote purely network-related covariates as internal covariates. Examples are longitude/latitude, distance to the nearest vertex or the location on the network segment. In applications with road networks, an internal covariate could also be the direction or the kind of a road. Technically, internal covariates can always be associated with a network segment index m and a respective bin index k . Secondly, we denote covariates that are not directly related to the network geometry as external covariates. Examples are time, weather conditions, but also the kind of a crime when the point pattern represents the spatial distribution of crimes along a network of streets.

Let x_1, \dots, x_C be the set of $C \in \mathbb{N}$ covariates to be considered in the model which are already suitably transformed, if required. According to the above considerations the index set $\{1, \dots, C\}$ disjointly splits into sets \mathcal{I} and \mathcal{E} referring to internal/external covariates, i.e. $\mathcal{I} \cup \mathcal{E} = \{1, \dots, C\}$. Further, if $\mathcal{E} \neq \emptyset$ let U be the number of unique combinations of the outcomes of the external covariates, otherwise we set $U = 1$. By introducing an additional index $u = 1, \dots, U$ we change (6) to

$$(12) \quad \lambda_{m,k,u} = \exp \left(\nu_\chi(z_{m,k}) + \sum_{c=1}^C s_c(x_{c_{m,k,u}}) + \log h_m \right),$$

where $s_c(\cdot)$ is the influence function of the covariate x_c which is defined below and the function ν_χ now serves as smooth baseline log-intensity. In (12), $x_{c_{m,k,u}}$ is the value of

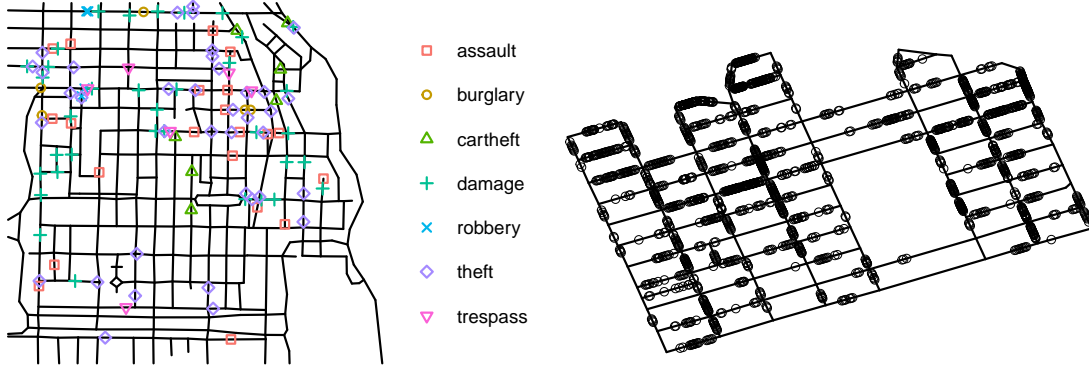


FIG 3. Left panel: The Chicago crimes network. Right panel: Major roads in the CBD of Melbourne, Australia, allocated to a street network. The circles show the location of parking bays with installed in-ground sensors.

the covariate x_c measured at location $z_{m,k}$ on the network if the value of x_c in this bin corresponds to the u -th unique combination of all external covariates. This notation suggests that an external covariate can also be network related, and indeed, we can e.g. model varying weather conditions over time at different locations of the network. The function $s_c(\cdot)$ determines whether the c -th covariate is modeled (log-)linearly or as a smooth term. In the former case $s_c(x_{c_{m,k,u}}) = \beta_c x_{c_{m,k,u}}$, where β_c is the corresponding parameter that needs to be estimated. In the latter case the covariate x_c has a B-spline basis representation $s_c(x_{c_{m,k,u}}) = \sum_{j=1}^{J_c} \gamma_j^{(c)} B_j^l(x_{c_{m,k,u}})$, where the objective is to estimate the parameter vector $\gamma_c = (\gamma_1^{(c)}, \dots, \gamma_{J_c}^{(c)})$, see Section 3.1 for details. Moreover, the spline functions s_c are centered around zero for ensuring identifiability of smooths effects. Altogether, the former parameter vector γ from Section 3.2 now extends to a parameter vector θ which further includes the parameters β_c and B-spline coefficients γ_c .

Suppose now that indices $\mathcal{S} \subset \{1, \dots, C\}$ represent covariates, internal or external, which are modeled smoothly. Therefore, we need further $|\mathcal{S}|$ penalties $\mathcal{P}_{r_s}^{(s)}$ and smoothing parameters ρ_s associated with each of these smooth terms (Section 3.3), where ρ denotes the vector containing these smoothing parameters. Therefore, the log-likelihood that we now need to maximize is given by

$$\ell_{\mathcal{P}}(\theta; \rho, \rho) = \sum_{m=1}^M \sum_{k=1}^{N_m} \sum_{u=1}^U [y_{m,k,u} \log(\lambda_{m,k,u}) - \lambda_{m,k,u}] - \rho \mathcal{P}_r(\gamma) - \sum_{s \in \mathcal{S}} \rho_s \mathcal{P}_{r_s}^{(s)}(\gamma_s),$$

where $y_{m,k,c}$ denotes the count of observations in the k -th bin of the m -th curve segment with covariate combination u and $\rho \mathcal{P}_r(\gamma)$ is the same penalty as in (8). The smoothing parameters ρ_s associated with the smooth functions can also be updated using the Fellner-Schall method from Section 3.4. In particular, we can update many smoothing parameters at practically no additional costs. Moreover, quantification of uncertainty of linear and smooth covariate effects easily extends by taking the resulting inverse penalized Fisher information, as discussed in the previous subsection.

4. Networks and Data. In this section, we introduce three geometric networks that we use throughout the rest of this paper. We further visualize point processes living on these networks and describe properties of the networks, such as the count of edges. In our **R** implementation all networks are represented as an instance of the class `linnet` in the **R** package `spatstat` (Baddeley, Rubak and Turner, 2015), i.e. all these networks have a representation as a linear network.

Network	Hyde Park, Chicago	CBD, Melbourne	South Montgomery County
Unit	feet	meters	kilometers
Resolution	all streets	only major streets	only highways
Source	spatstat package	own representation	own representation
$ V $	338	96	339
$ E $	503	158	369
$ L $	31,150.21	25,473.90	175.42

TABLE 1

Summary of the geometric networks covered in this paper.

First, we consider the Chicago crimes network, which has already been treated in many papers before, see e.g. [Rakshit et al. \(2019\)](#). This network is available as an object named `chicago` in the **R** package `spatstat` and shown in the left panel of Figure 3. Various symbols visualize 116 crimes recorded over two weeks in the year 2002 in Hyde Park, Chicago, subdivided into seven kinds of crimes. Therefore, these data represent a marked point process on this geometric network. Most of the crimes seem to occur in the northeastern and northwestern parts of the map extract. A summary of the geometric network itself is given in Table 1.

Secondly, we employ data from the City of Melbourne, Australia. Between August 2011 and May 2012, the city installed in-ground sensors underneath around 4,600 out of more than 20,000 on-street parking lots in the city center of Melbourne. These sensors are capable of recording the arrival time and the departing time of a car to the second.² We take data from the period June-August 2019 and consider a subset of 1,618 on-street parking lots with installed in-ground sensors, which are all located in the Central Business District (CBD) of Melbourne and which are released at least once a day on average. Altogether the database compresses 1,907,941 events where one event is defined as the release of a parking lot. Our goal is to detect regions in this area where the occupancy of parking lots fluctuates most, also concerning the time of the day. Therefore, we first need to specify a geometric network where the point process is living on. This network is constructed by only including major streets in this area as well as side streets in which on-street parking lots with sensors are located. In Figure 3 we show the location of the considered on-street parking lots on the corresponding geometric network. A summary of the network is given in Table 1.

The third and endmost data example are road accidents recorded on a network of highways (state highways, interstate highways and US highways) in the southern part of Montgomery County, Maryland, which borders on the District of Columbia in the north. We take data³ related to 14,571 traffic collisions from the years 2015-2019, which occurred between 6 am and 10 pm. The locations of these incidents on the underlying road network of highways are shown in Figure 4. Here, we already see that the network of highways is denser in the southern part of the map extract with many south-north routes originating from Washington, D.C. Note that we excluded Maryland Route 200 from the network since traffic collisions occurring on this highway are not included in the database. Besides the location of each collision, the dataset includes covariates such as the type of the highway (internal) or the date and time (external) of the incident. Moreover, we added the direction of a street section as a further internal covariate. Again, a summary of the network itself is provided in Table 1.

²The massive amounts of data that these sensors produce and many more data related to parking in the City Melbourne are available for gratuitous download at <https://data.melbourne.vic.gov.au/>.

³Data on car crashes can be downloaded from <https://data.montgomerycountymd.gov/Public-Safety/Crash-Reporting-Incidents-Data/bhju-22kf>.

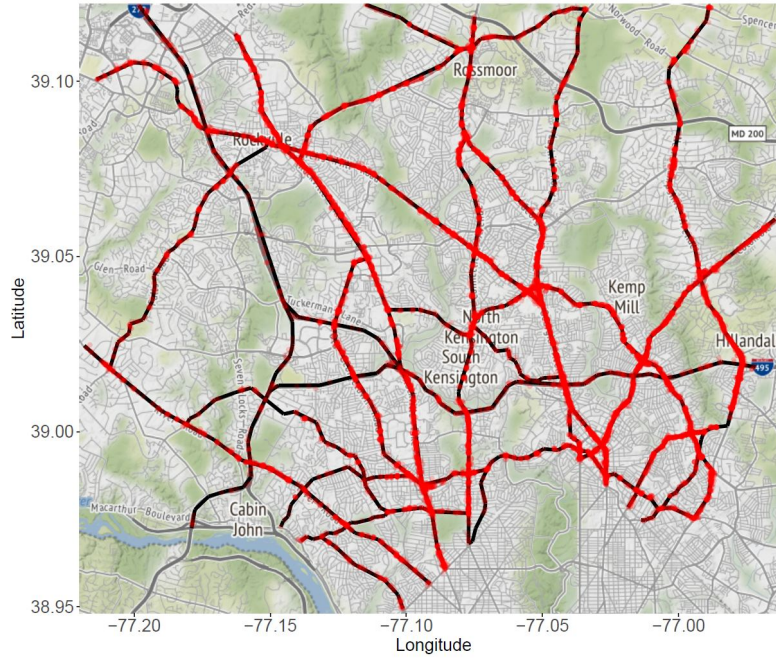


FIG 4. Network of highways in the southern part of Montgomery County, Maryland. Data points represent accidents occurring between the years 2015 and 2019 from 6 am in the morning until 10 pm in the evening.

5. The Chicago Crimes Network. To begin with the analysis of the Chicago crimes data, we neglect the kind of crime. When estimating the intensity of crimes with our approach, we set $\delta = 10$, $h = 2$ feet and make use of a second-order penalty. We compare the intensity fit with two baseline methods which are both kernel-based and implemented as function `density.lpp` in the `spatstat` package. Firstly, we fit a kernel-based model according to [McSwiggan, Baddeley and Nair \(2017\)](#) which computes the estimates by solving a heat equation on the network. We refer to this as method 1. This method exclusively relies on the shortest path distance as the metric. The bandwidth σ of the kernel smoother is selected via likelihood cross-validation using the function `bw.lpp1` while setting the argument `distance = "path"`, yielding an optimal bandwidth of $\sigma = 158.49$ feet. Secondly, we fit another kernel-based model, now using an adaptive two-dimensional smoothing kernel as proposed by [\(Rakshit et al., 2019\)](#). Here, Scott's rule of thumb ([Scott, 2015](#)), which is for linear networks implemented within the function `bw.scott.iso`, yields an optimal bandwidth of $\sigma = 119.55$ feet. We refer to this as method 2. Note that the ratio of the two optimal bandwidths which we obtained for methods 1 and 2, respectively, are in line with the general arguments of [Rakshit et al. \(2019\)](#), equation 18.

The top plot of Figure 5 shows the intensity estimate when employing the penalized spline-based approach. The lower plots show the fitted intensity when method 1 (bottom left) or method 2 (bottom right), respectively, is used. We find that the high-intensity regions on the network are similarly located for all three methods. However, we see the following two major deviations. First, the penalized spline-based method yields higher intensity estimates for the region on top in the middle of the plot. Otherwise, the estimate is akin to the estimate, resulting when fitting method 1 to the data. Second, we consider the fitted intensity with method 2 in the area, marked by the rectangle in the top right corner of the map. It can be seen that in this area the kernel smoother, which is based on the Euclidean distance, estimates a distinctly higher estimate when compared to the other two methods based on the shortest path distance. This might be caused by the data located in the top right corner of the map. These data are close to the rectangle with respect to the Euclidean distance, but the shortest-path distance is larger by a factor of around three.

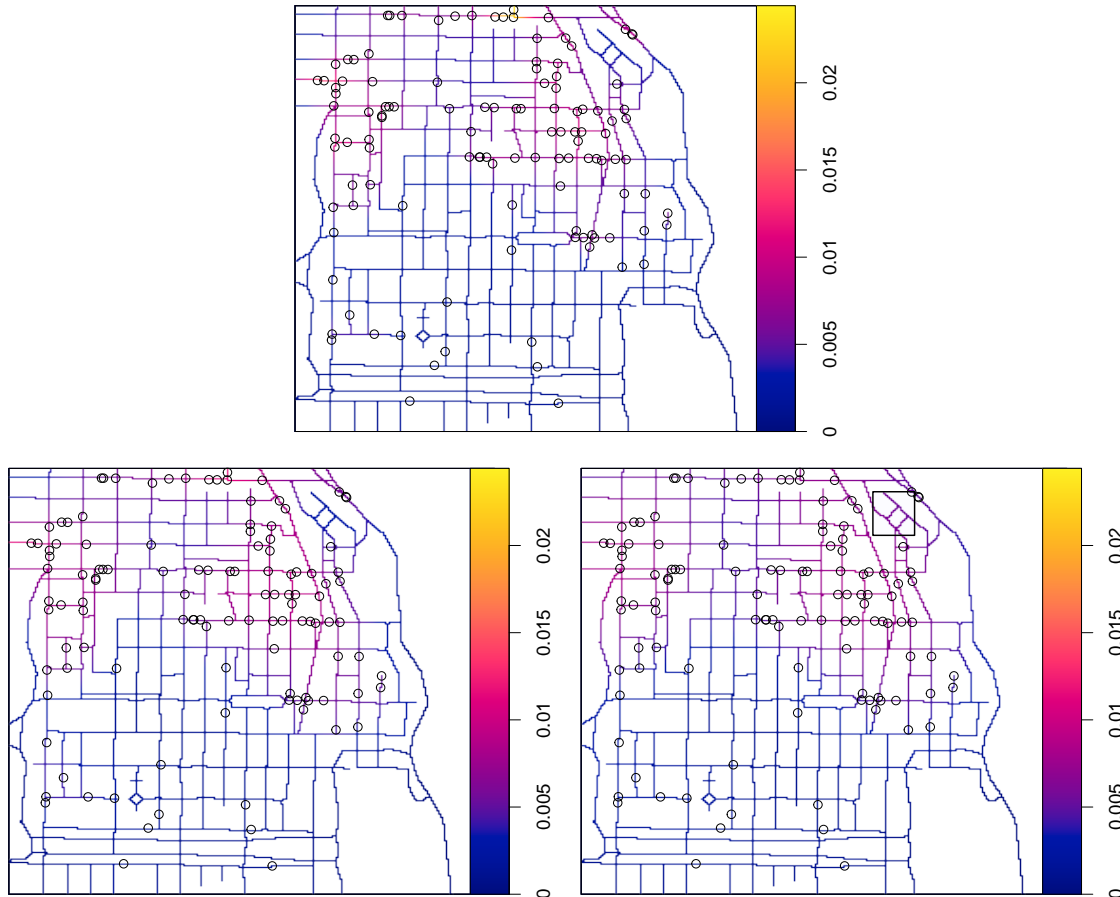


FIG 5. Top plot: Penalized spline based intensity estimate. Bottom left panel: Kernel intensity estimate based on the shortest path distance (method 1). Bottom right panel: Kernel intensity estimate based on the Euclidean distance (method 2). On top of each estimate the original data are plotted.

We repeat the penalized spline intensity estimation from above but now include the kind of crime as a covariate in our model. For comparison, we also fit a Poisson process on the network with the kind of crime as a single covariate and taking the estimate resulting from method 1 from above as offset, for details see [McSwiggan \(2019\)](#). Therefore, we employ the function `lppm` from the `spatstat` package. The resulting parameter estimates on the log-scale of both models, including 95% confidence intervals, are shown in Figure 6. We see that the effects and their respective standard errors are very similar in both models. However, our model has the advantage that the nonparametric baseline intensity and covariate effects can be estimated within one model. This can be extended to work with multiple covariates if available. When the baseline intensity shall be estimated with a kernel smoother, covariate effects can generally be estimated employing an alternating two-step approach, see e.g. [Kauermann \(2002\)](#) for asymptotic results on the real line. We stress that this is not provided within any of the functions in the `spatstat` package.

6. On-Street Parking in Melbourne, Australia. Our goal in this example is to detect locations in the street network of the CBD of Melbourne where the occupation of on-street parking lots fluctuates most. Therefore, we define an event to be the clearing of an on-street parking lot, and the point process that we observe now has a spatial as well as a temporal structure. However, in areas with more allocated parking lots, there is per se a higher chance of finding a cleared lot. Therefore, we first need to estimate the intensity φ_Z of parking lots

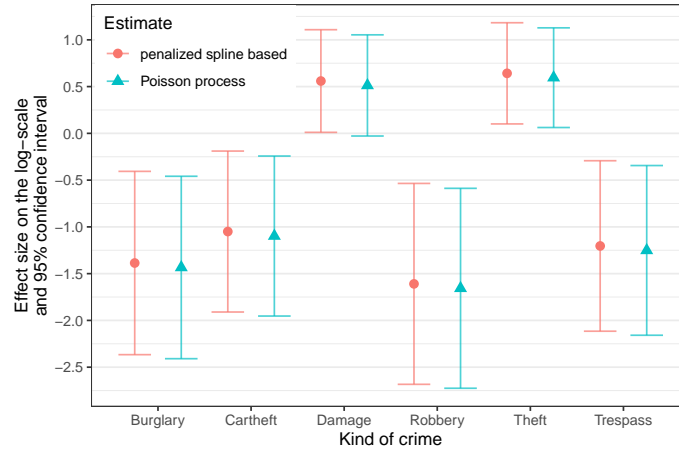


FIG 6. *Parameter estimates of linear effects and 95% confidence intervals for the kind of crime in the Chicago crimes network, when fitting a penalized spline model with covariates or a Poisson process with constant baseline intensity.*

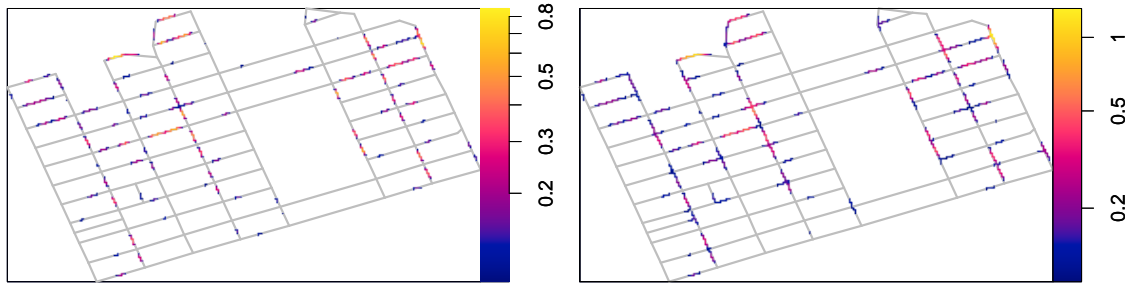


FIG 7. *(Baseline) intensity of on-street parking lots (in parking lots per meter) in the CBD of Melbourne. Left panel: Intensity fitted without covariates. Right panel: Baseline intensity of a fit including a covariate which considers closeness to a vertex. Both plots only show (baseline) intensities ≥ 0.1 on a logarithmic color-scale with the underlying network being visualized in grey.*

on the network of streets, where \mathcal{Z} is the point process already visualized in the right panel of Figure 3.

The fitted intensity $\hat{\varphi}_{\mathcal{Z}}$ of allocated parking lots using the penalized spline-based approach is depicted in the left panel of Figure 7. Here, the intensity is visualized on a logarithmic scale, and for reasons of presentation we only show areas on the network where the intensity of on-street parking lots is expected to be at least 0.1 parking lots per meter. However, we also find that the intensity around street crossings is throughout lower than 0.1, which is reasonable when considering the allocation of parking lots shown in Figure 3. Therefore, we fit the model again with a dichotomous network internal covariate, which has a value of 1 if a location on the network is closer than 20 meters to a vertex and 0 else. The resulting baseline intensity is shown in the right panel of Figure 7, again only showing baseline intensities being 0.1 or higher on a logarithmic color scale. We now find that around many intersections, the baseline intensity exceeds the value 0.1. The estimated effect of the internal covariate is -1.87 on the log-scale with a standard error of 0.12. Consequently, we must multiply the baseline intensity by a factor $\exp(-2.35) \approx 0.15$ in order to get the intensity estimates at areas closer than 20 meters to a vertex. Note that the overall intensity is not continuous by including this covariate anymore even though the baseline intensity is still continuous.

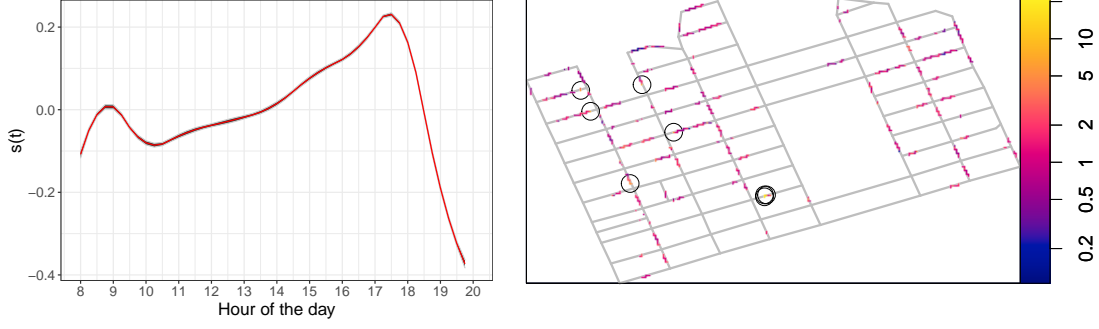


FIG 8. *Left panel: Smooth effect of the time of the day on the log-scale. Right panel: Estimate of the baseline fluctuation rate (per hour per parking lot).*

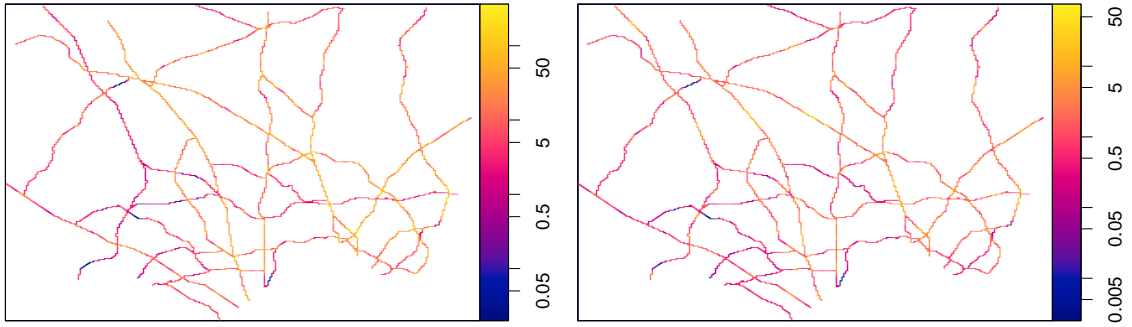


FIG 9. *Intensity estimate of traffic collisions. Left panel: Fit without covariates. The intensity is expressed in terms of collisions per kilometer and year. Right panel: Estimate of the baseline intensity when fitting with covariates.*

We now look at cleared parking lots, see Section 4 for details. The data are considered as results of the clearing point process \mathcal{Y} , whose log-baseline intensity $\varphi_{\mathcal{Y}}$ is again estimated using penalized spline smoothing. We further include a smooth effect $s(t)$ for the time of the day, where the estimate on the log-scale is shown in the left panel of Figure 8. The effect yields an increasing intensity towards the evening, followed by a rapid decrease of the intensity after 6 pm. To quantify, the intensity of the clearing process \mathcal{Y} drops by factor $\exp(-0.4 - 0.2) \approx 0.55$, i.e. by more than a half, within two hours. Note that due to the large amounts of data, the confidence bands of the smooth effect $s(t)$ are very narrow.

Overall we want to determine the ratio $\varphi_{\mathcal{X}} = \varphi_{\mathcal{Y}}/\varphi_{\mathcal{Z}}$, which expresses the expected fluctuation rate of parking lots along the network. However, we are only interested in the fluctuation rate where we expect a reasonable number of parking lots, here the locations where $\widehat{\varphi}_{\mathcal{Z}}(z) \geq 0.1$, see Figure 7. Here, we make use of $\widehat{\varphi}_{\mathcal{Z}}$ which results from the fit when accounting for the distance to the nearest vertex. The baseline intensities $\widehat{\varphi}_{\mathcal{X}}(z)$ are normalized such that they can be interpreted as the expected hourly fluctuation rate of a parking lot, which is located at $z \in \mathbf{L}$. In order to get the expected fluctuation rate at a specific time of the day t , $\widehat{\varphi}_{\mathcal{X}}(z)$ needs to be multiplied by factor $\exp(s(t))$. The estimates of the fluctuation process \mathcal{X} are shown in right panel of Figure 8, where spots with $\varphi_{\mathcal{X}}(z) \geq 5$ are surrounded by a black circle. We find that high fluctuation rates occur, especially in the southwestern part of the CBD.

7. Car Crashes in Montgomery County, Maryland. As a start, we determine the penalized spline-based intensity fit of traffic collisions on the network of highways shown in

Effect	Estimate (s.e.)	Relative risk	95% CI of relative risk
Interstate highway	-1.560 (0.196)	0.21	[0.14, 0.31]
US highway	0.597 (0.222)	1.82	[1.18, 2.81]
East-west	-0.085 (0.110)	0.92	[0.74, 1.14]
Southeast-northwest	0.059 (0.112)	1.06	[0.85, 1.32]
Southwest-northeast	-0.435 (0.158)	0.65	[0.47, 0.88]
Distance to intersection in km	-1.970 (0.466)	0.14	[0.06, 0.35]

TABLE 2

Summary of estimated fixed effects on the log-scale including standard errors, relative risk = $\exp(\text{estimate})$ and the 95% confidence interval (CI) of the relative risk.

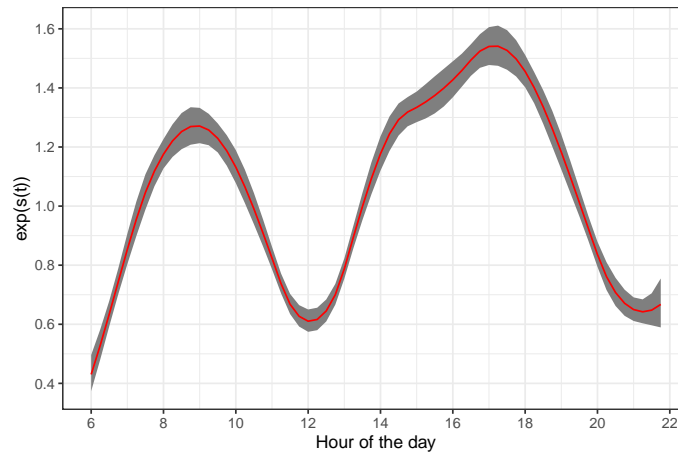


FIG 10. Smooth effect of the time of the day on the exp-scale.

Figure 4, where we do not include covariates in the model. In the left panel of Figure 9 we illustrate the fitted intensity, where the unit of the intensity estimate is traffic collisions per kilometer and year. We see that some routes exhibit throughout very high intensities, most of them originating from Washington, D.C. To name a few of them, these are US highway 29 or Maryland state highways 97 and 355. On the two interstate highways in this area with numbers 270 and 495, there seems to be a relatively low risk of traffic collisions, i.e. there are only a few crashes per kilometer of highway.

As a next step, we include covariates in the model as proposed in Section 3.6. Firstly, these are two categorical covariates, namely the type of highway (categories: state, interstate, or US highway) as well as the direction of the highway (categories: south-north, east-west, southeast-northwest, southwest-northeast), with the first category always representing the respective reference category. Secondly, we include a linear effect for the distance in kilometers to the nearest intersection with another highway. Finally, we include a smooth effect $s(t)$ for the time of the day t as we already did in Section 6.

The fitted baseline intensity, when including covariates, is shown in the right panel of Figure 9. We do not have such a clear picture as before when fitting the intensity without covariates, suggesting that covariate effects now explain a large part of the variance. The resulting estimates of the fixed linear effects are listed in Table 2. Indeed, the relative risk of a traffic incident is five times lower on an interstate highway and nearly twice as large on a US highway when compared to Maryland state highways. Both effects are significant on the 95% confidence level. The effect of interstate highways might be associated with the type of construction as these have several lanes with different driving directions being structurally separated. Moreover, interstate highways are usually connected with other highways

through several ramps to avoid contact with oncoming traffic. Concerning the direction of the highways, there seems to be a significant difference of routes proceeding from southwest to northeast when controlling for the other effects included in the model. To quantify the effect, the relative risk of observing a traffic collision is 35% lower when compared to south-north highways. Lastly, we can infer that high-risk areas in the fit without covariates are mainly located close to intersections since the relative risk decreases by 86% ($\approx \exp(-1.97)$) per kilometer distance to the nearest intersection. Finally, we depict in Figure 10 the estimated smooth effect of the time of the day. Here, we see that the significant risk of observing a traffic accident varies enormously with the hour of the day, where we see a peak in the morning between 8 am and 10 am as well as in the afternoon between 2 pm and 6 pm. In the latter period, the relative risk of observing a road accident is more than twice as large as at noon.

8. Simulation Study.

8.1. *Integrated Squared Error.* We start by employing data simulated on the Chicago crimes network in order to explore the performance of penalized spline based intensity estimation on geometric networks, also with respect to two methods which we briefly discussed in Section 5 above. Therefore, we specify intensity functions $\varphi_{\mathcal{X}_n}$ on the Chicago network which satisfy $\int_{\mathbf{L}} \varphi_{\mathcal{X}_n}(\mathbf{z}) d\mathbf{z} = n$ and for each of the sample sizes $n = 100, 200, 500, 1000$ we simulate $R = 100$ point processes $\mathcal{X}_n = \{\mathbf{x}_1, \dots, \mathbf{x}_n\}$. We quantify the prediction error of the r -th simulation with sample size n through

$$\text{ISE}(\widehat{\varphi}_{\mathcal{X}_n}; r) = \frac{1}{n^2} \int_{\mathbf{L}} (\widehat{\varphi}_{\mathcal{X}_n}(\mathbf{z}; r) - \varphi_{\mathcal{X}_n}(\mathbf{z}))^2 d\mathbf{z}$$

where $\widehat{\varphi}_{\mathcal{X}_n}(\cdot; r)$ denotes the estimate of $\varphi_{\mathcal{X}_n}$ based on the r -th sample. That is, we quantify the prediction error through the integrated squared error (ISE) between the the estimated density $\widehat{f}_{\mathcal{X}}(\mathbf{z}) = \widehat{\varphi}_{\mathcal{X}_n}(\mathbf{z})/n$ and the true density $f_{\mathcal{X}}(\mathbf{z}) = \varphi_{\mathcal{X}_n}(\mathbf{z})/n$, which enables us to compare the ISE for different sample sizes. Denoting with $\mathbb{E}[\cdot]$ the sample mean in the following, the mean integrated squared error (MISE) estimated from a sample of size n is given by $\text{MISE}(\widehat{\varphi}_{\mathcal{X}_n}) = \mathbb{E}[\text{ISE}(\widehat{\varphi}_{\mathcal{X}_n}; r)] = \frac{1}{R} \sum_{r=1}^R \text{ISE}(\widehat{\varphi}_{\mathcal{X}_n}; r)$. Moreover, denoting with $\mathbb{E}[\widehat{\varphi}_{\mathcal{X}_n}(\mathbf{z}; r)] = \frac{1}{R} \sum_{r=1}^R \widehat{\varphi}_{\mathcal{X}_n}(\mathbf{z}; r)$ the point-wise sample mean of the intensity estimate, we can express the MISE as the sum of the integrated variance (IVar) and the integrated squared bias (ISBias), i.e.

$$\begin{aligned} \text{MISE}(\widehat{\varphi}_{\mathcal{X}_n}) &= \text{IVar}(\widehat{\varphi}_{\mathcal{X}_n}) + \text{ISBias}(\widehat{\varphi}_{\mathcal{X}_n}) \\ (13) \quad &= \frac{1}{n^2} \int_{\mathbf{L}} \mathbb{E} \left[(\widehat{\varphi}_{\mathcal{X}_n}(\mathbf{z}; r) - \mathbb{E}[\widehat{\varphi}_{\mathcal{X}_n}(\mathbf{z}; r)])^2 \right] d\mathbf{z} \\ &+ \frac{1}{n^2} \int_{\mathbf{L}} (\mathbb{E}[\widehat{\varphi}_{\mathcal{X}_n}(\mathbf{z}; r)] - \varphi_{\mathcal{X}_n}(\mathbf{z}))^2 d\mathbf{z}. \end{aligned}$$

We now carry out the process described above for two intensity functions. First, the intensity is chosen to be proportional to the intensity estimate fitted with the kernel method based on the shortest path distance from below, compare the bottom left panel of Figure 5. Second, we simulate point processes \mathcal{X} according to

$$(14) \quad \varphi_{\mathcal{X}}(\mathbf{z}) \propto \begin{cases} 1, & \mathbf{z} \in e_m \text{ and } m \equiv 0 \pmod{10} \\ 0, & \text{else} \end{cases},$$

which means that we simulate data according to a uniform distribution on line segments whose edge index is completely divisible by 10 and on all other edges the probability of

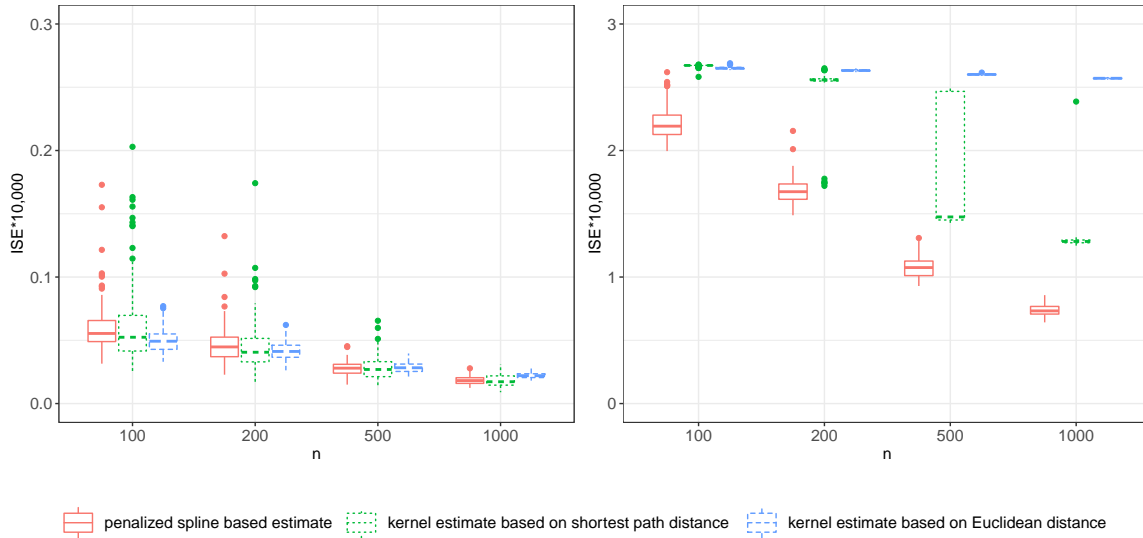


FIG 11. *Integrated squared error (scaled by factor 10,000) of the intensity estimate depending on model choice and sample size. Left panel: True density is proportional to the intensity as shown in the bottom left panel of Figure 5. Right panel: True density is defined according to (14).*

observing a datum is zero. Therefore, (14) specifies an intensity function where the data are clustered on some edges of the network and the intensity is not smooth but discontinuous. When fitting the simulated data with one of the three models, we use the same hyperparameters or strategies to determine the hyperparameters as described in Section 5 when fitting the original data.

In Figure, 11 we show a boxplot of the resulting ISEs⁴ when fitting the intensity of simulated point patterns with different sample sizes making use of the three models discussed above. The left panel of Figure 11 illustrates that the distribution of the ISEs is very similar for a given sample size if the data are simulated according to the smooth intensity function shown in the bottom left panel of 5, and there is an apparent reduction of the ISE if the sample size increases. When simulating from the discontinuous intensity function (14), the ISE is generally more than ten times larger when compared to the first example, see the right panel of Figure 11. In this situation, penalized spline-based estimation is favored against the two kernel-based methods in terms of ISE. Moreover, the two kernel-based methods perform similarly for small sample sizes, but the estimate based on the Euclidean distance shows only a slight reduction of the ISE if the sample size increases. Therefore, we can conclude that if the actual intensity is sufficiently smooth, all the three considered methods exhibit similar estimation errors. However, the penalized spline-based method shows more robustness towards misspecified smoothness when compared to the two considered kernel-based methods.

Figure 12 shows the same simulation results as in Figure 11, but now in terms of the MISE as the decomposition of IVar and ISBias. We see that in both settings, i.e. with data simulated from a smooth intensity function and a discontinuous intensity function, respectively, our method seems to solve the bias-variance trade-off reasonably well. Note that this is, as already known from penalized spline smoothing on the real line (Fahrmeir et al., 2013), achieved by the optimal choice of the smoothing parameter. In this matter, overestimation or underestimation of the smoothing parameter leads to higher bias and less variability or less

⁴It is essential to note that for computing the ISEs, we have used the development version 2.2-1.003 of the `spatstat.linnet` package.

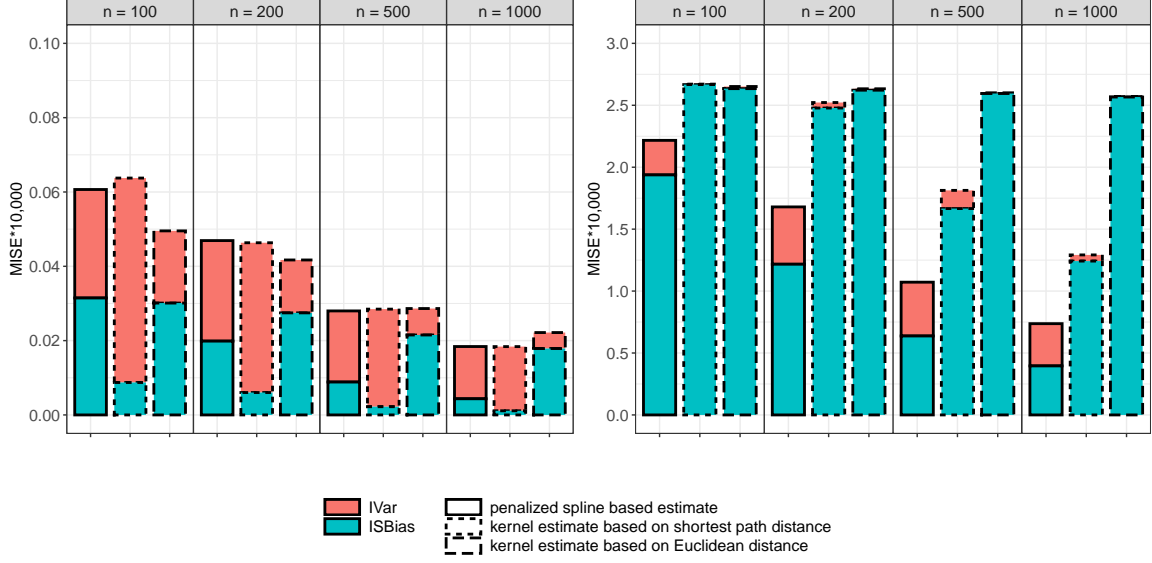


FIG 12. Decomposition of the mean integrated squared error (MISE) into integrated variance (IVar, top bars) and integrated squared bias (ISBias, lower bars), computed from the simulation results shown in Figure 11. All values are scaled by factor 10,000. Left panel: True density is proportional to the intensity as shown in the bottom left panel of Figure 5. Right panel: True intensity is defined according to (14).

bias and more variability, respectively. In some settings of our simulation study, it can be seen that the two kernel-based methods do not solve the bias-variance trade-off well. When considering estimates of the method based on the shortest path distance in the first setting, most of the portion of the MISE can be attributed to the IVar if the sample size is large. On the other hand, in the second setting the major portion of the MISE can be attributed to the ISBias, where the same holds for the kernel methods based on the Euclidean distance.

Finally, we also explore the effect of the bin width h and the dimension J of the B-spline basis, which is determined by the knot distance δ . We vary δ with 5, 10 and 20 feet, the global bin width h is chosen to be the half, a fifth or a tenth of δ , respectively. Here, the analysis is restricted on the sample size $n = 200$ and $\varphi_{\mathcal{X}}$ according to the bottom left panel of Figure 5. The results in Figure 13 show the ISEs with $R = 100$ simulations for each configuration. We find that the ISE hardly varies with different choices of δ and h . Thus, as long as δ and h are small enough, we can not considerably increase the prediction performance by reducing these two hyperparameters. This result is in line with the motivating arguments in Eilers and Marx (1996) and corresponds to the general results for penalized spline smoothing as derived in Kauermann and Opsomer (2011).

In order to find a proper value for δ in general, it is often helpful to start with $\delta \approx \frac{1}{2} \min_m d_m$ which is for the Chicago network given by 4.7 feet. This ensures that there is at least one segment-specific B-spline on each e_m and that the segment-specific knot distances δ_m are of similar size, see Section 3.1. In the above simulation study on the Chicago network, δ can be increased by at least factor four without loss of prediction performance which is the merit of the penalization. Moreover, since we are operating with linear B-splines, there is no benefit by setting the global bandwidth h to an unnecessarily small value, which is also supported by the simulation results shown in Figure 13.

8.2. *Estimation of Covariate Effects.* We now want to study the performance of the model extension which we have elaborated in Section 3.6. Therefore, we first define an intensity function $\varphi_{\mathcal{X}}$ of a point process \mathcal{X} on the Chicago street network from above according

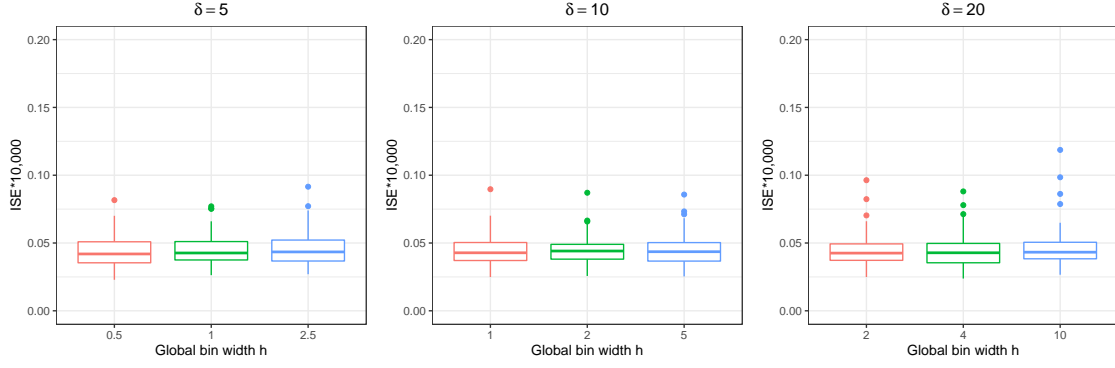


FIG 13. *Integrated squared error (scaled by factor 10,000) of the penalized spline based intensity estimate depending on δ (left panel: $\delta = 5$, middle panel: $\delta = 10$, right panel: $\delta = 20$) and the global bin width h . Data are simulated according to the bottom left panel of Figure 5 with $n = 200$ sample points.*

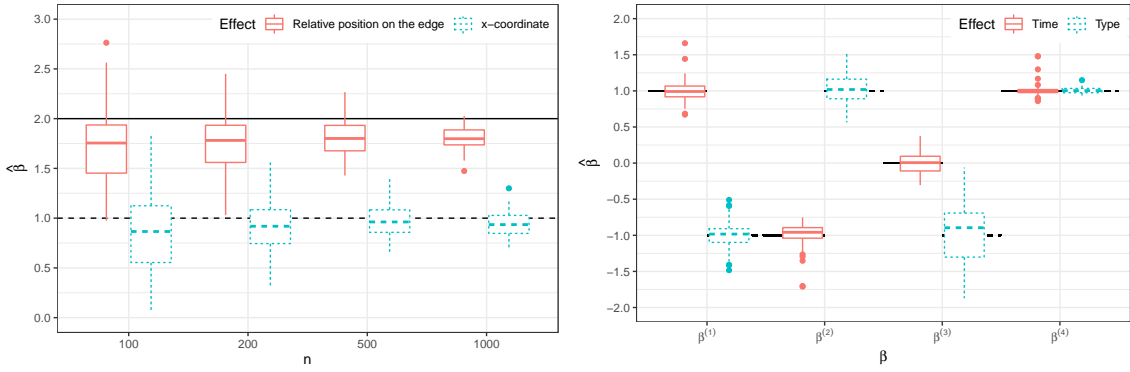


FIG 14. *Left panel: Parameter estimates when simulating with internal covariates t_p and x setting $\beta_{t_p} = 2$ and $\beta_x = 1$. Right panel: Parameter estimates when simulating with external covariates time and type, i.e. $\beta = (\beta_0, \beta_{time}, \beta_{type})^\top$, and setting $\beta^{(1)} = (2, 1, 1)^\top$, $\beta^{(2)} = (1, -1, 1)^\top$, $\beta^{(3)} = (1, 0, -1)^\top$ and $\beta^{(4)} = (4, 1, 1)^\top$.*

to

$$(15) \quad \varphi_{\mathcal{X}}(z) \propto \exp(2 \cdot t_p + x),$$

where $t_p \in [0, 1]$ measures the relative location of the point z on its line segment, with $t_p = 0$ or $t_p = 1$ meaning that z is located on one of its endpoints. Furthermore, x is the x -coordinate (in 1000 feet) of z in the plane. Note that this intensity function is not continuous and data simulated according to (15) are clustered towards the right end of each line segment. We simulate $R = 100$ point patterns of sample sizes $n = 100, 200, 500, 1000$ according to (15) and estimate the intensity including t_p and x as linear covariates. These covariates are both internal covariates with effect sizes $\beta_{t_p} = 2$ and $\beta_x = 1$ on the log-scale. The resulting parameter estimates are shown as boxplots in Figure 14. We see that the estimates of both effects are slightly biased towards zero, while the bias is generally larger for the effect of t_p and decreases when the sample size increases. Likewise, for both effects the variances decrease with increasing sample size.

Finally, we simulate again according to the intensity function $\varphi_{\mathcal{X}}$ which yields from the shortest path dependent kernel based intensity estimate of the Chicago crimes data. However, the data shall now also depend on two external covariates. Therefore, we first draw a sample of size 10 from a time dependent covariate $x_t \sim \mathcal{N}(0, 1)$. Secondly, we also consider a dichotomous covariate x_d with values ‘‘A’’ and ‘‘B’’. Thus, there are

$U = 20$ unique combinations of these two covariates. We further introduce a parameter vector $\boldsymbol{\beta} = (\beta_0, \beta_1, \beta_2)^\top$ of effects on the log-scale, where β_0 scales the baseline intensity, β_1 is the linear effect of x_t and β_2 is the effect of $x_d = \text{B}$ with respect to $x_d = \text{A}$. Then, for each covariate combination $u = 1, \dots, U$ we draw the sample size n_u from a Poisson distribution with parameter $\mu_u = \exp(\beta_0 + \beta_1 \cdot x_{t,u} + \beta_2 \cdot \mathbb{1}\{x_{d,u} = \text{B}\})$. In the end, we have simulated $n = \sum_{u=1}^U n_u$ data points on the network. The right panel of Figure 14 shows boxplots of the parameter estimates when conducting this simulation study with $\boldsymbol{\beta}^{(1)} = (2, 1, 1)^\top$, $\boldsymbol{\beta}^{(2)} = (1, -1, 1)^\top$, $\boldsymbol{\beta}^{(3)} = (1, 0, -1)^\top$ and $\boldsymbol{\beta}^{(4)} = (4, 1, 1)^\top$, respectively. We find that these estimates are generally unbiased. Note that the baseline intensity is chosen to be the highest in scenario 4, which results in the lowest variances of the estimates when compared to scenarios with lower baseline intensity.

9. Discussion and further Work. In this article, we developed a new method for estimating the intensity (or density) of a stochastic process living on a geometric network. We exploited and extended penalized spline estimation to work on a subset of connected curves, denoted as geometric networks. A benefit of this model is its inherent simplicity and the relatedness to well-elaborated statistical concepts such as penalized spline smoothing and generalized additive models. Note that by our definition, an interval $[a, b]$ is a special case of a geometric network \mathbf{L} with $|\mathbf{E}| = 1$ and $|\mathbf{V}| = 2$.

This paper shows that our methodology works for point processes on two-dimensional linear networks embedded in the plane, a particular case of a geometric network. In the future, we plan to implement intensity estimation on geometric networks in general, which is straightforward when considering our general derivations. More precisely, we want to enable to represent geometric networks as the union of parametric curves (see Section 2), also in higher dimensional spaces, and provide visualizations of fitted intensities in two and three dimensions.

As seen in the simulation study and the application examples, the penalization also compensates for non-equidistant knots and bin widths on different segments of \mathbf{L} . However, these differences can be made as small as desired by reducing δ and h . In the end, this leads to a trade-off between accuracy and computational effort. In a Euclidean space, the penalties used for estimation with B-splines are often based on derivatives of the smoother. However, in a geometric network the question arises how one could define differentiability of a function f at vertices \mathbf{v} with $\deg(\mathbf{v}) > 2$. Adapting the penalization technique of Eilers and Marx (1996) circumvents this question and proves to be the right choice for our setting.

We envisage many more generalizations and extensions of our method. First, the linear penalized spline approach could be extended to work with higher-order penalized splines, particularly with quadratic or cubic penalized splines. Therewith, the estimated intensities could become even smoother along the network. However, B-splines of order two or higher in Euclidean spaces are differentiable. Therefore, as stated above, it would be much more complicated to construct network-based B-splines of order two or higher around vertices \mathbf{v} with $\deg(\mathbf{v}) > 2$.

Furthermore, if we drop the assumption that the network graph L should not be directed, we need the geometric representation \mathbf{L} to be possibly directed as well. This means that a curve e_m additionally is equipped with a direction if $e_m = (v_i, v_j)$ is a directed edge from v_i to v_j but there is no edge from v_j to v_i . In this case, the distance measure $d_{\mathbf{L}}$ from above does not define a metric any more since then, $d_{\mathbf{L}}(z_1, z_2) = d_{\mathbf{L}}(z_2, z_1)$ for $z_1, z_2 \in \mathbf{L}$ does not hold in general. This extension of the model could especially be applied to the Maryland road accident data to investigate whether the intensity varies with the direction of the lane. However, we consider this to be beyond the scope of this paper and aim to tackle this in the future.

REFERENCES

- ANG, Q. W., BADDELEY, A. and NAIR, G. (2012). Geometrically corrected second order analysis of events on a linear network, with applications to ecology and criminology. *Scandinavian Journal of Statistics* **39** 591–617.
- BADDELEY, A., RUBAK, E. and TURNER, R. (2015). *Spatial point patterns: methodology and applications with R*. Chapman and Hall/CRC.
- BADDELEY, A., NAIR, G., RAKSHIT, S., MCSWIGGAN, G. and DAVIES, T. M. (2020). Analysing point patterns on networks—A review. *Spatial Statistics* 100435.
- BARR, C. D. and SCHOENBERG, F. P. (2010). On the Voronoi estimator for the intensity of an inhomogeneous planar Poisson process. *Biometrika* **97** 977–984.
- BASSETT, R. and SHARPNACK, J. (2019). Fused density estimation: theory and methods. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **81** 839–860.
- BEER, G. (2013). The structure of extended real-valued metric spaces. *Set-Valued and Variational Analysis* **21** 591–602.
- BORRUSO, G. (2008). Network density estimation: a GIS approach for analysing point patterns in a network space. *Transactions in GIS* **12** 377–402.
- CHAN, T. M. (2012). All-pairs shortest paths for unweighted undirected graphs in $o(mn)$ time. *ACM Transactions on Algorithms (TALG)* **8** 1–17.
- CURRIE, I. D., DURBAN, M. and EILERS, P. H. (2006). Generalized linear array models with applications to multidimensional smoothing. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **68** 259–280.
- DE BOOR, C. (1972). On calculating with B-splines. *Journal of Approximation theory* **6** 50–62.
- DIGGLE, P. (1985). A kernel method for smoothing point process data. *Journal of the Royal Statistical Society: Series C (Applied Statistics)* **34** 138–147.
- EILERS, P. H. and MARX, B. D. (1996). Flexible smoothing with B-splines and penalties. *Statistical Science* **11** 89–102.
- EILERS, P. H., MARX, B. D. and DURBÁN, M. (2015). Twenty years of P-splines. *SORT: statistics and operations research transactions* **39** 0149–186.
- FAHRMEIR, L., KNEIB, T., LANG, S. and MARX, B. (2013). *Regression*. Springer.
- GOLDENBERG, A., ZHENG, A. X., FIENBERG, S. E., AIROLDI, E. M. et al. (2010). A survey of statistical network models. *Foundations and Trends® in Machine Learning* **2** 129–233.
- HEUSER, H. (2006). *Lehrbuch der Analysis 1, 2*. Teubner, Stuttgart.
- KAUERMANN, G. (2002). On a small sample adjustment for the profile score function in semiparametric smoothing models. *Journal of multivariate analysis* **82** 471–485.
- KAUERMANN, G. and OPSOMER, J. D. (2011). Data-driven selection of the spline dimension in penalized spline regression. *Biometrika* **98** 225–230.
- KOLACZYK, E. D. and CSÁRDI, G. (2014). *Statistical analysis of network data with R* **65**. Springer.
- MCSWIGGAN, G. (2019). Spatial point process methods for linear networks with applications to road accident analysis, Doctoral Thesis, University of Western Australia.
- MCSWIGGAN, G., BADDELEY, A. and NAIR, G. (2017). Kernel density estimation on a linear network. *Scandinavian Journal of Statistics* **44** 324–345.
- MORADI, M. M., RODRÍGUEZ-CORTÉS, F. J. and MATEU, J. (2018). On kernel-based intensity estimation of spatial point patterns on linear networks. *Journal of Computational and Graphical Statistics* **27** 302–311.
- MORADI, M. M., CRONIE, O., RUBAK, E., LACHIEZE-REY, R., MATEU, J. and BADDELEY, A. (2019). Resample-smoothing of Voronoi intensity estimators. *Statistics and computing* **29** 995–1010.
- O'DONNELL, D., RUSHWORTH, A., BOWMAN, A. W., MARIAN SCOTT, E. and HALLARD, M. (2014). Flexible regression models over river networks. *Journal of the Royal Statistical Society: Series C (Applied Statistics)* **63** 47–63.
- OKABE, A., SATOH, T. and SUGIHARA, K. (2009). A kernel density estimation method for networks, its computational method and a GIS-based tool. *International Journal of Geographical Information Science* **23** 7–32.
- OKABE, A. and YAMADA, I. (2001). The K-function method on a network and its computational implementation. *Geographical Analysis* **33** 271–290.
- OKABE, A., YOMONO, H. and KITAMURA, M. (1995). Statistical analysis of the distribution of points on a network. *Geographical Analysis* **27** 152–175.
- RAKSHIT, S., DAVIES, T., MORADI, M. M., MCSWIGGAN, G., NAIR, G., MATEU, J. and BADDELEY, A. (2019). Fast Kernel Smoothing of Point Patterns on a Large Network using Two-dimensional Convolution. *International Statistical Review* **87** 531–556.
- RASMUSSEN, J. G. and CHRISTENSEN, H. S. (2020). Point processes on directed linear networks. *Methodology and Computing in Applied Probability* 1–21.

- RUPPERT, D., WAND, M. P. and CARROLL, R. J. (2003). *Semiparametric regression* **12**. Cambridge university press.
- RUPPERT, D., WAND, M. P. and CARROLL, R. J. (2009). Semiparametric regression during 2003–2007. *Electronic journal of statistics* **3** 1193.
- RUSHWORTH, A., PETERSON, E., VER HOEF, J. and BOWMAN, A. (2015). Validation and comparison of geostatistical and spline models for spatial stream networks. *Environmetrics* **26** 327–338.
- SCHELLHASE, C. and KAUERMANN, G. (2012). Density estimation and comparison with a penalized mixture approach. *Computational Statistics* **27** 757–777.
- SCOTT, D. W. (2015). *Multivariate density estimation: theory, practice, and visualization*. John Wiley & Sons.
- SNIJDERS, T. A. (1996). Stochastic actor-oriented models for network change. *Journal of mathematical sociology* **21** 149–172.
- SPOONER, P. G., LUNT, I. D., OKABE, A. and SHIODE, S. (2004). Spatial analysis of roadside Acacia populations on a road network using the network K-function. *Landscape ecology* **19** 491–499.
- R CORE TEAM (2013). R: A language and environment for statistical computing.
- WOOD, S. N. (2017). *Generalized additive models: an introduction with R*. Chapman and Hall/CRC.
- WOOD, S. N. and FASIOLO, M. (2017). A generalized Fellner-Schall method for smoothing parameter optimization with application to Tweedie location, scale and shape models. *Biometrics* **73** 1071–1081.
- XIE, Z. and YAN, J. (2008). Kernel density estimation of traffic accidents in a network space. *Computers, environment and urban systems* **32** 396–406.

Appendix

3.A. Intensity estimation on geometric networks with the R package `geonet`

Manuscript in preparation Schneble, M., Kauermann, G. (2021). Intensity estimation on geometric networks with the R package `geonet`.

Code and data The package can be downloaded from the comprehensive R archive network (<https://cran.r-project.org/web/packages/geonet/index.html>). A development version is available at GitHub (<https://github.com/MarcSchneble/geonet>).

Author Contributions The R package `geonet` was designed and written by Marc Schneble, who also is the maintainer of the package. Marc Schneble has written the major part of the manuscript with advice from Göran Kauermann. Both authors were involved in extensive proof-reading of the manuscript.

Intensity Estimation on Geometric Networks with the R Package **geonet**

Marc Schneble

Ludwig-Maximilians-Universität
München

Göran Kauermann

Ludwig-Maximilians-Universität
München

Abstract

This article presents and discusses the R package **geonet** which is designed for intensity estimation of point processes on a geometric network. The underlying methodology is based on penalized spline smoothing and generalized additive models. The intensity itself is approximated through a linear B-spline basis where the related regression coefficients are appropriately penalized. The package allows incorporating internal and external covariates when estimating the intensity, which is novel and not possible with the current state-of-the-art kernel smoothing techniques implemented in the R package **spatstat**. Moreover, the package **geonet** inherits new classes for an advanced representation as a geometric network. Great care has been taken to provide full compatibility between the classes from our new package **geonet** and the classes from the already established package **spatstat**. The model's possible applications are manifold and we illustrate the usage of the package based on the Chicago crimes network and the Montgomery highway network. The first is a widely employed example and the latter is novel and encompasses a network of highways in Montgomery County, Maryland, where the point process relates to traffic accidents on the network of highways.

Keywords: B-splines, generalized additive models, geometric networks, intensity estimation, R package **geonet**.

1. Introduction

The statistical analysis of spatial data is an active research area which has generated many associated software contributions in R (R Core Team 2021). Some of the best-known R packages for handling spatial data are **sp** (Bivand, Pebesma, and Gomez-Rubio 2013), **sf** (Pebesma 2018), **stplanr** (Lovelace and Ellison 2018), **SSN** (Ver Hoef, Peterson, Clifford, and Shah 2014) and **spatstat** (Baddeley, Rubak, and Turner 2015). The first two packages provide mostly methods and classes for spatial data in Euclidean spaces. The packages **stplanr** and **SSN** cover spatial data on networks while the former focuses on spatial transport data and the latter is designed for modeling and predicting data on stream networks. The package **spatstat** is the currently most comprehensive package and is actually a bundle of packages which allows to model and visualize both, data in Euclidean spaces (package **spatstat.geom**) and network spaces (package **spatstat.linnet**).

Data that are observed on a spatial network arise in many applications. Examples are crimes or traffic accidents that occur on a network of streets. The problem formulation often involves finding the network locations where most of the events are expected to occur and quantifying the intensity of expected events per unit length of the network. One of the main functionality of the **spatstat.linnet** package is kernel-based estimation of the intensity of a point process on a linear network from an observed point pattern. Since the distance of two points on a network is usually measured through the shortest path distance, estimation techniques do need to account for the geometry of the network and thus, the methods are much more elaborated compared to analyses in Euclidean spaces.

In this paper, we present the R package **geonet** (version 0.6.0) which provides an alternative to the package **spatstat** (version 2.2.0) by making use of penalized spline smoothing and generalized additive models as proposed by the **mgcv** package (Wood 2017) and extended by Schneble and Kauermann (2020) towards network data. Moreover, the package introduces new classes for representing networks and point patterns on geometric networks. Thereby, methods for these classes also focus on the compatibility with the respective classes of the package **spatstat**. That is, our implementation allows us to convert objects which are represented through a class of the package **spatstat** to the respective class of the **geonet** package and vice versa. Standard methods for generic functions such as **print**, **plot** and **summary** are also included in the new package. The package **geonet** can be downloaded from the comprehensive R archive network (CRAN¹). A more frequently updated development version of the package is available via GitHub².

The rest of the paper is structured as follows. Section 2 provides an introduction to geometric networks with focus on their representation used in the package **geonet**. Section 3 introduces point processes and the related intensity function on a network. Section 4 treats the new methodology for intensity estimation on geometric networks as proposed by Schneble and Kauermann (2020) and its implementation within the package **geonet**. We further briefly review the current state-of-the-art kernel smoothing techniques as implemented in the package **spatstat**. In the subsequent Section 5, we illustrate the functionality of the new package based on two observed point patterns. The relation of the package **geonet** to the package **spatstat** is shown in Section 6. Section 7 provides formulae and algorithms to simulate from an intensity that has the proposed B-spline basis representation and in Section 8 we treat some of the

¹<https://cran.r-project.org/>

²<https://github.com/MarcSchneble/geonet>

computational aspects of the penalized spline-based intensity estimation method in more detail. Finally, Section 9 summarizes the paper and gives an outlook on possible extensions of the package.

2. Geometric networks

2.1. Definition and properties

In this section, we introduce networks in the sense of spatial objects embedded in a Euclidean space. Note that the term network often refers to abstract network graphs, see e.g. [Kolaczyk and Csárdi \(2014\)](#). Nonetheless, both types of networks have in common that they are constructed by vertices and their pairwise connections. Let $\mathcal{V} = \{\mathbf{v}_1, \dots, \mathbf{v}_W\}$ be a set of vertices (or nodes) with $\mathbf{v}_i \in \mathbb{R}^q$ for $i = 1, \dots, W$ and $q \in \mathbb{N}$. Further, let $\mathbf{A} \in \{0, 1\}^{W \times W}$ be an adjacency matrix with $\mathbf{A}_{ij} = 1$, if vertices indexed by i and j are connected and $\mathbf{A}_{ij} = 0$ otherwise. The network is assumed to be undirected and free of self-loops, i.e. \mathbf{A} is symmetric with its diagonal elements being equal to zero. Moreover, we denote with $\deg(\mathbf{v}_i) = \sum_{j=1}^W \mathbf{A}_{ij}$ the degree of the i -th vertex. If $\mathbf{A}_{ij} = 1$ we define $e_{ij} \subset \mathbb{R}^q$ to be the connection between vertices \mathbf{v}_i and \mathbf{v}_j . We index these connections with $m = 1, \dots, M$ and say that $\mathcal{E} = \{e_1, \dots, e_M\}$ represents the set of edges of the network. This leads to the definition of a geometric (or spatial) network \mathbf{G} as being the union of curves (or network segments) $\mathbf{G} = \bigcup_{m=1}^M e_m$.

A spatial network is typically non-linear, e.g. streets from one place to another do not necessarily follow a straight line. We can, of course, approximate curves through line segments which is pursued in the package `geonet`. A curved edge e_m between two nodes \mathbf{v}_i and \mathbf{v}_j is then approximated through the alignment of L_m linear segments resulting in many additional (artificial) vertices of degree two. More formally, we describe each path as a parametric curve, see details in [Schneble and Kauermann \(2020\)](#). That is, $e_m = \{\nu(t), t \in [0, 1] \mid \nu(0) = \mathbf{v}_i, \nu(1) = \mathbf{v}_j\} \subset \mathbb{R}^q$ where $\nu : [0, 1] \rightarrow \mathbb{R}^q$ defines a continuous, but in general non-differentiable, parametric curve. If the curve is discontinuous at $\mathbf{a}_{m,1}, \dots, \mathbf{a}_{m,L_m-1} \in \mathbb{R}^q$ we denote these points as the artificial vertices of the network which are introduced to represent e_m as a polygonal chain. Defining with $\mathbf{a}_{m,0} = \mathbf{v}_i$ and $\mathbf{a}_{m,L_m} = \mathbf{v}_j$ the endpoints of the m -th edge, such an edge is the union of straight line segments

$$e_m = \bigcup_{k=1}^{L_m} \{\mathbf{a}_{m,k-1} + t(\mathbf{a}_{m,k} - \mathbf{a}_{m,k-1}) \mid t \in [0, 1]\}.$$

Thus, the length $d_m = |e_m|$ of the m -th curve thus results to

$$d_m = \sum_{k=1}^{L_m} \|\mathbf{a}_{m,k} - \mathbf{a}_{m,k-1}\|_q,$$

where $\|\cdot\|_q$ denotes the Euclidean distance in q dimensions.

The artificial vertices do not have an actual function concerning the geometry of the network. We illustrate this in [Figure 1](#) where we plot a network of streets in Chicago, Illinois. This network is available as the object `chicago` from the package `spatstat.data`. Here, a non-straight street is approximated through linear segments and additional (artificial) vertices.

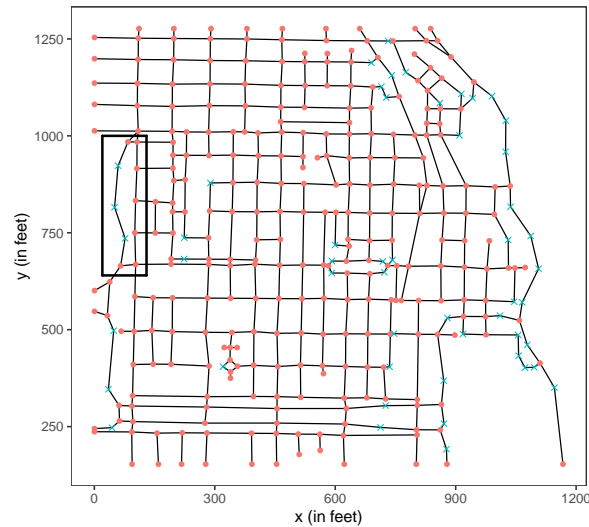


Figure 1: A network of streets around Hyde Park in Chicago, Illinois. The red dots represent the actual vertices of the network. The blue crosses show the artificial vertices of degree two.

The red dots show the network's actual vertices, which have a degree of one or a degree of more than two, i.e. they represent the terminus of a street or the intersection of more than two streets, respectively. The blue crosses show vertices that have exactly degree two. These (artificial) vertices are located on the approximated path between two adjacent vertices v_i and v_j . In case that the artificial vertices are treated as actual vertices, a spatial network is uniquely determined by the set of vertices \mathcal{V} and the adjacency matrix \mathbf{A} . Since all the edges of such a network are straight line segments, these kinds of spatial networks are denoted as linear networks (Baddeley *et al.* 2015).

2.2. The class `gn`

A geometric network in the `geonet` package is represented as an object of class `gn` and the generic function `as_gn` transmutes an existing object, e.g. a linear network of class `linnet` from the package `spatstat.linnet`, into a geometric network of class `gn`. The printed `summary` of a geometric network shows basic information with regards to the number of vertices and network segments, the length of the network and the distribution of the vertex degrees. Below, we show the summary of the Chicago network and we see that this geometric network does not have vertices of degree two.

```
R> library(spatstat)
R> library(geonet)
R> L <- as.linnet(chicago)
R> G <- as_gn(L)
R> summary(G)
```

```
Geometric network in 2 dimensions with 287 vertices
and 452 curve segments.
```

The linear representation of the network has 338 vertices and 503 straight line segments.

Total length of the network: 31150.21 feet

Minimum segment length: 10.789 feet

Maximum segment length: 373.871 feet

Distribution of vertex degrees:

1	3	4	5
44	114	127	2

Plots in the package **geonet** are created by making use of the package **ggplot2** (Wickham 2016), which builds on the grammar of graphics and each **ggplot** object consists of different layers. The **plot** method for an object of class **gn** silently returns an object of class **ggplot** and prints it to the console. Therefore, the plots can be customized afterward, but the **plot** methods also provide an interface to the most common arguments of the **ggplot** layers such as **title**. The network plot can be framed by setting **frame = TRUE** and in this case, a coordinate system will be added to the plot as well.

Using the example of the Chicago network, we illustrate an object of class **gn** which consists of nine attributes. The attribute **\$vertices** is a **tibble** with four columns and each row represents an artificial vertex or an actual vertex, respectively. Tibbles are an advanced representation of **data.frame** objects in R and the corresponding **tibble** functions are re-exported by the package **dplyr** (Wickham, François, Henry, and Müller 2021) which we use for data manipulation. The first column, named **a**, represents a common identifier for both the artificial vertices and the actual vertices, The indices of the latter kind of vertices are shown in the second column, named **v**, and otherwise filled with **NA** for the artificial vertices of degree two. The Chicago network has 51 of those vertices. Moreover, the columns named **x** and **y** refer to the *x*- and *y*-coordinates of a vertice, respectively. The following tibble shows the **\$vertices** of the Chicago network restricted to those vertices which are located within the rectangle shown in Figure 1. The vertices with identifiers 119-121 correspond to the blue squares in this rectangle.

```
R> length(which(is.na(G$vertices$v)))
```

```
[1] 51
```

```
R> library(dplyr)
```

```
R> G$vertices %>% filter(a %in% c(96, 97, 115, 119:123, 125, 134))
```

```
# A tibble: 10 x 4
```

	a	v	x	y
	<int>	<int>	<dbl>	<dbl>
1	96	83	106.	984.
2	97	84	83.7	984.
3	115	102	106.	916.
4	119	NA	59.2	923.
5	120	NA	50.4	816.
6	121	NA	76.6	735.

```

7  122  106  65.4  665.
8  123  107  103.   833.
9  125  109  101.   750.
10 134  116  103.   668.

```

The attribute `$lins` is a tibble that represents the single straight line segments of the network, i.e. the number of rows of this tibble is equal to $\sum_{m=1}^M L_m$. The columns `l` and `e` represent the identifier of a straight line segment and the index of the associated curve segment, respectively. The columns `a1` and `a2` are the identifiers of the endpoints of the straight lines and the subsequent four columns match their coordinates. The length of each line segment is saved in the column `length`. The column `frac1` is the cumulative length of a curve excluding this line segment and `frac2` is the fraction which a line segment has with respect to the whole length of the curve segment. Possibly more columns are associated with internal network covariates. The following tibble shows those four line segments which build the curve with index $m = 11$ that is located within the rectangle shown in Figure 1. Note that the line segments are always ordered, such that adjoining segments are listed sequentially. We can also deduce that the length of the m -th curve is equal to $66.1 + 107.0 + 84.6 + 71.7 = 329.4$ feet.

```
R> all.equal(nrow(G$lins), L$lines$n)
```

```
[1] TRUE
```

```
R> G$lins %>% filter(l %in% 167:170)
```

```
# A tibble: 4 x 11
```

	l	e	a1	a2	a1_x	a1_y	a2_x	a2_y	length	frac1	frac2
	<int>	<dbl>	<int>	<int>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
1	167	11	97	119	83.7	984.	59.2	923.	66.1	0	0.200
2	168	11	119	120	59.2	923.	50.4	816.	107.	0.200	0.325
3	169	11	120	121	50.4	816.	76.6	735.	84.6	0.526	0.257
4	170	11	121	122	76.6	735.	65.4	665.	71.7	0.783	0.217

The attributes `$adjacency` and `$incidence` represent the adjacency matrix of dimension $W \times W$ and the incidence matrix of dimension $W \times M$, respectively. The attribute `$d` is a vector of length M containing the path length of all curve segments and with `$unit` one obtains the unit in which lengths in the network are measured. The attributes `$q`, `$W` and `$M` are self-explanatory. Currently, only $q = 2$ is supported by the package *geonet*, i.e. geometric networks which are embedded in the plane. However, the methodology to estimate the intensity on geometric networks presented later in this paper is general and simply allows for application to networks embedded in $q > 2$ dimensions.

3. Point processes on a geometric network

3.1. Intensity function

A point process \mathcal{X} on a geometric network \mathbf{G} is a random countable subset and a realization $\mathbf{x} = \mathcal{X}(\omega) \subset \mathbf{G}$ is called a point pattern. The function $\varphi_{\mathcal{X}} : \mathbf{G} \rightarrow [0, \infty)$ denotes the intensity

function of \mathcal{X} , where $\varphi_{\mathcal{X}}(\mathbf{u})$ can be interpreted as the expected number of points per unit length of the network in the vicinity of a point $\mathbf{u} \in \mathbf{G}$. More generally, it holds that the expected number of points falling in a set $\mathbf{B} \subset \mathbf{G}$ is given by

$$\mathbb{E}_{\mathcal{X}}(\mathbf{B}) = \int_{\mathbf{B}} \varphi_{\mathcal{X}}(\mathbf{u}) \, d\mathbf{u}.$$

Sometimes, one may rather characterize a point process \mathcal{X} on a network through a density function $f_{\mathcal{X}}(\cdot)$ where

$$\mathbb{P}_{\mathcal{X}}(\mathbf{u} \in \mathbf{B}) = \int_{\mathbf{B}} f_{\mathcal{X}}(\mathbf{u}) \, d\mathbf{u}$$

is the probability that a point \mathbf{u} which is generated by \mathcal{X} will fall in the subset \mathbf{B} . Note that the density is the normalized intensity. In particular, if $\int_{\mathbf{G}} \varphi_{\mathcal{X}}(\mathbf{u}) \, d\mathbf{u} = n$, then $f_{\mathcal{X}}(\cdot) = \frac{1}{n} \varphi_{\mathcal{X}}(\cdot)$ and thus, $\mathbb{P}(\mathbf{u} \in \mathbf{B}) = \frac{1}{n} \mathbb{E}_{\mathcal{X}}(\mathbf{B})$.

3.2. The class `gnpp`

A point pattern on a geometric network in the package `geonet` is an object of class `gnpp` (geometric network point pattern) which has two attributes, the underlying geometric network (`$network`) and a tibble which describes the observed point pattern (`$data`). The generic function `as_gnpp` transmutes an existing object into an object of class `gnpp`. Methods exist for example for point patterns on a linear networks, i.e. objects of class `lpp` from the package `spatstat.linnet`. The attribute `$data` of a `gnpp` object is a tibble that has one row for every observation and at least the following six columns. The columns `l` and `e` refer to the line segment identifier and the curve segment index, respectively, where `tp_l` and `tp_e` specify the relative location on the respective segment as a number in the unit interval. Columns `x` and `y` refer to the coordinates of each point. Possibly more columns are associated with external covariates.

Again, we illustrate an object of class `gnpp` through the Chicago network. The original object `chicago` is a point pattern of size $n = 116$ on the network already shown above. The points represent crimes committed between April 25 and May 8, 2020, and among others, the dataset was already analyzed by [Ang, Baddeley, and Nair \(2012\)](#) and [Baddeley *et al.* \(2015\)](#). In the case of the Chicago crimes data, the seventh row is a categorical covariate which is the type of crime committed at the respective location. The first six rows of the data are shown in the following.

```
R> X <- as_gnpp(chicago)
R> head(X$data)
```

```
# A tibble: 6 x 7
   l   tp_l     e   tp_e     x     y marks
  <int> <dbl> <dbl> <dbl> <dbl> <dbl> <fct>
1    42 0.242     3 0.120  715. 1191. cartheft
2    42 0.698     3 0.347  730. 1170. damage
3   104 0.145     9 0.652  869. 1003. damage
4   292 0.949    19 0.691  743.  683. theft
5     5 0.926    44 0.926  282. 1251. damage
6     5 0.515    44 0.515  245. 1251. robbery
```

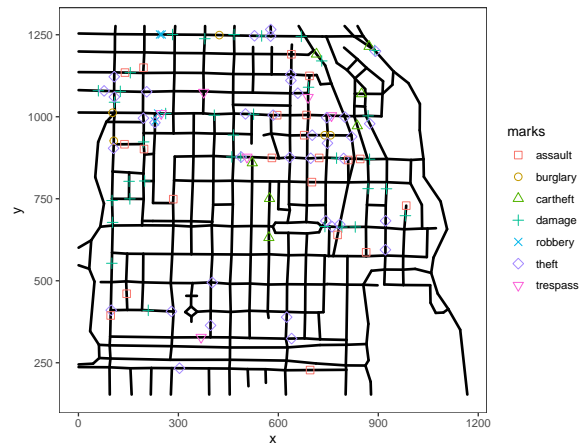


Figure 2: The Chicago crimes data.

The printed summary of a `gnpp` object first shows information similar to that of a geometric network. Besides, it also provides information about the size of the point pattern and the average intensity. Furthermore, a summary of all internal and external covariates is shown. Below, we show the summary of the `chicago` point pattern when represented as an object of class `gnpp`.

```
R> summary(X)
```

```
Point pattern on a geometric network in 2 dimensions with 287 vertices
and 452 curve segments.
```

```
The linear representation of the network has 338 vertices
and 503 straight line segments.
```

```
Total length of the network: 31150.21 feet
```

```
Network has no internal covariates
```

```
Number of points: 116
```

```
Average intensity: 0.00372 points per foot
```

```
Number of external covariates: 1
```

```
1) factor variable "marks":
```

assault	burglary	cartheft	damage	robbery	theft	trespass
21	5	7	35	4	38	6

A plot of the point pattern is shown in Figure 2 which is the output of the following command.

```
R> plot(X, frame = TRUE, covariate = "marks")
```

In general, the `plot` method for an object of class `gnpp` silently returns an object of class `ggplot` and prints the result to the console. The argument `covariate` allows stratifying the plotted point pattern according to a categorical covariate which is, in the case of the Chicago crimes data, the covariate “marks” representing the kind of a crime.

4. Intensity estimation of point processes on spatial networks

4.1. Intensity estimation based on penalized spline smoothing

Methodology

In the package **geonet**, intensity estimation on a geometric network is based on penalized spline smoothing (Eilers and Marx 1996) and generalized additive models (GAMs, Hastie and Tibshirani 1986). Therefore, network internal and external covariates can be considered when estimating the intensity with this method. The methodology of the penalized spline smoothing technique for point patterns on geometric networks has been developed by Schneble and Kauermann (2020).

The cornerstone of the method is to represent the log-baseline intensity $\nu_{\mathcal{X}}(\cdot) = \log \varphi_{\mathcal{X}}(\cdot)$ of a point process \mathcal{X} through a linear combination of B-splines (Ruppert, Wand, and Carroll 2003 and Fahrmeir, Kneib, Lang, and Marx 2007), where each single B-spline is only locally supported on the geometric network. Therefore, we define with $[\mathbf{u}_1, \mathbf{u}_2) \subset \mathbf{G}$ the shortest path with length $d_{\mathbf{G}}(\mathbf{u}_1, \mathbf{u}_2)$ between two points \mathbf{u}_1 and \mathbf{u}_2 on the network \mathbf{G} , where a square bracket indicates that an endpoint is included and otherwise it is not. Moreover, we construct on each curve e_m with endpoints, say \mathbf{v}_i and \mathbf{v}_j , an equidistant sequence of I_m knots $\mathbf{v}_i = \boldsymbol{\tau}_{m,1}, \dots, \boldsymbol{\tau}_{m,I_m} = \mathbf{v}_j$. Thereby, the curve-specific knot distances $\delta_m = d_{\mathbf{G}}(\boldsymbol{\tau}_{m,k}, \boldsymbol{\tau}_{m,k+1})$ shall be chosen to be as similar as possible for a given global knot distance δ , for details see Schneble and Kauermann (2020).

We now define $J_m = I_m - 2$ B-splines on the m -th curve according to

$$B_{m,k}(\mathbf{u}) = \frac{d_{\mathbf{G}}(\mathbf{u}, \boldsymbol{\tau}_{m,k})}{\delta_m} \mathbb{1}_{[\boldsymbol{\tau}_{m,k}, \boldsymbol{\tau}_{m,k+1})}(\mathbf{u}) + \frac{d_{\mathbf{G}}(\boldsymbol{\tau}_{m,k+2}, \mathbf{u})}{\delta_m} \mathbb{1}_{[\boldsymbol{\tau}_{m,k+1}, \boldsymbol{\tau}_{m,k+2})}(\mathbf{u}) \quad (1)$$

for $\mathbf{u} \in \mathbf{G}$ and $k = 1, \dots, I_m - 2$. In fact, (1) defines linear B-splines based on a recursive formula which is used when constructing B-splines on the real line, see De Boor (1978). Thus, each of the B-splines $B_{m,k}(\cdot)$ is supported between three adjacent knots on the m -th segment and the mode of $B_{m,k}(\cdot)$ is equal to $\boldsymbol{\tau}_{m,k+1}$. In order to supplement the B-splines defined by (1) to a basis on \mathbf{G} we need further W B-splines which are supported around the vertices of the geometric network. Therefore, let with a simplified notation and without loss of generality $e_1, \dots, e_{\deg(v_i)}$ be the adjacent curves of the i -th vertex and $\mathbf{v}_i = \boldsymbol{\tau}_{1,1}, \dots, \boldsymbol{\tau}_{\deg(v_i),1}$. Then, we define a B-spline $B_{(i)}$ around the i -th vertex according to

$$B_{(i)}(\mathbf{u}) = \mathbb{1}_{\mathbf{v}_i}(\mathbf{u}) + \sum_{k=1}^{\deg(v_i)} \left(1 - \frac{d_{\mathbf{G}}(\mathbf{v}_i, \mathbf{u})}{\delta_k} \right) \mathbb{1}_{(\mathbf{v}_i, \boldsymbol{\tau}_{k,2})}(\mathbf{u}) \quad (2)$$

for $\mathbf{u} \in \mathbf{G}$ and $i = 1, \dots, W$. Altogether, the B-spline basis defined by (1) and (2) has dimension $J = \sum_{m=1}^M J_m + W$ and for simplicity of notation, we index these B-splines with $j = 1, \dots, J$. Thus, a B-spline basis representation of the log-baseline intensity at a point $\mathbf{u} \in \mathbf{G}$ is given by

$$\nu_{\mathcal{X}}(\mathbf{u}) = \sum_{j=1}^J \gamma_{0,j} B_j(\mathbf{u}), \quad (3)$$

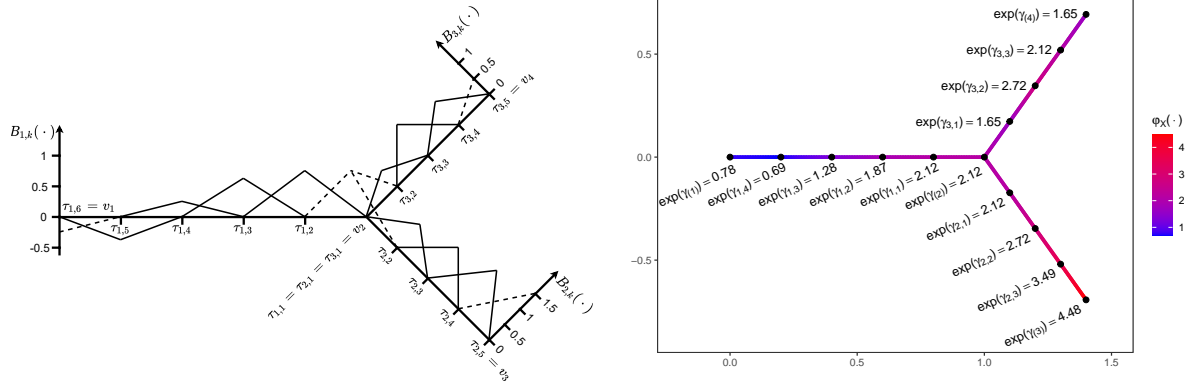


Figure 3: Approximating the log-intensity of a point process on a network through linear B-splines. Left panel: B-splines $B_{m,k}(\cdot)$ and $B_{(i)}$ weighted by their respective coefficients. Right panel: Resulting intensity on the network.

where $\boldsymbol{\gamma}_0 = (\gamma_{0,1}, \dots, \gamma_{0,J})^\top$ is a vector of B-spline coefficients that needs to be estimated from the data. We exemplify in Figure 3 the above considerations on a small artificial network which consists of three lines joining at one vertex. The left panel sketches the single linear B-splines weighted by their coefficients as in (3) and the right panel shows the resulting intensity $\varphi_{\mathcal{X}}(\cdot) = \exp \nu_{\mathcal{X}}(\cdot)$ on the network. Note that at the mode of the B-spline B_j , the intensity equals $\exp(\gamma_j)$.

As a next step, we include (3) in a regression model. Therefore, we bin the observed point pattern \boldsymbol{x} appropriately on the network. We choose a bin width h_m for the m -th curve such that \boldsymbol{e}_m is subdivided into $N_m = \frac{d_m}{h_m} \in \mathbb{N}$ bins and $y_{m,k}$ denotes the number of points which fall into the k -th bin of the m -th curve. Moreover, we assume to have p covariates which we collect in a vector $\boldsymbol{z}_{m,k} = (z_{m,k}^{(1)}, \dots, z_{m,k}^{(p)})^\top$ where $p = 0$ is suitable as well and $\boldsymbol{z}_{m,k} = \emptyset$ in this case. Covariates can also be time-dependent which we, however, neglect in our notation for simplicity of presentation. We assume a Poisson distribution for $y_{m,k}$ and set up the model

$$y_{m,k} \mid \boldsymbol{z}_{m,k} \stackrel{\text{indep.}}{\sim} \text{Poi}(\lambda_{m,k}), \quad (4)$$

where $\lambda_{m,k}$ is modeled in a log-linear fashion according to

$$\lambda_{m,k} = \exp \left(\nu_{\mathcal{X}}(\boldsymbol{u}_{m,k}) + \sum_{l=1}^p f_l(z_{m,k}^{(l)}) + \log h_m \right). \quad (5)$$

Here, $\boldsymbol{u}_{m,k}$ is the mid-point of the k -th bin of the m -th edge and we replace $\nu_{\mathcal{X}}(\boldsymbol{u}_{m,k})$ with the B-spline basis representation (3). Moreover, $\log h_m$ serves as an offset which ensures appropriate scaling of the baseline intensity. The function $f_l(\cdot)$ denotes the effect of the l -th covariate on the intensity at $\boldsymbol{u}_{m,k}$. In particular, if covariate effects are assumed to be linear, then $f_l(z_{m,k}^{(l)}) = \beta_l z_{m,k}^{(l)}$ with parameter β_l to be estimated. Alternatively, $f_l(\cdot)$ can also be approximated through a B-spline basis representation with parameter vector $\boldsymbol{\gamma}_l$ to be estimated and we denote with \mathcal{S} the index set that refers to all covariates which are modeled as a smooth function. Finally, we collect all model parameters in the parameter vector $\boldsymbol{\theta}$.

The parameters of the model defined by (4) and (5) can be estimated by maximizing the likelihood of the model. Following Wood (2017) the maximum likelihood estimate $\hat{\boldsymbol{\theta}}$ is the argument which maximizes the penalized log-likelihood

$$\ell_{\text{penal}}(\boldsymbol{\theta}) = \sum_{m=1}^M \sum_{k=1}^{N_m} \left[y_{m,k} \log \lambda_{m,k} - \lambda_{m,k} \right] - \sum_{l \in \{0\} \cup \mathcal{S}} \rho_l \mathcal{P}_r(\boldsymbol{\gamma}_l). \quad (6)$$

Here, \mathcal{P}_r are penalties of order r which penalize the squared r -th order differences of neighboring B-spline coefficients. This approach has been proposed by Eilers and Marx (1996) to penalize equidistant B-spline coefficients in the Euclidean setting. Moreover, ρ_l are smoothing parameters which control the amount of smoothing of the log-baseline intensity and the smooth covariate effects, respectively. In order to penalize the coefficients $\boldsymbol{\gamma}_0$, we construct a distance matrix \mathbf{D}_γ of the B-spline coefficients. For example, a second order penalty is based on the set of triples

$$\mathcal{D}_2 = \{(i, k, j) \mid \mathbf{D}_\gamma(i, j) = 2, \mathbf{D}_\gamma(i, k) = \mathbf{D}_\gamma(k, j) = 1, 1 \leq i < j \leq J\}.$$

When considering the toy-example in Figure 3 and following the initial notation, e.g. the triple $(\gamma_{1,1}, \gamma_{(2)}, \gamma_{3,1}) \in \mathcal{D}_2$. In general, a second order penalty for the coefficients $\boldsymbol{\gamma}_0$ is given by

$$\mathcal{P}_2(\boldsymbol{\gamma}_0) = \sum_{\mathcal{D}_2} ((\gamma_{0,i} - \gamma_{0,k}) - (\gamma_{0,k} - \gamma_{0,j}))^2 = \boldsymbol{\gamma}_0^\top \mathbf{K}_2 \boldsymbol{\gamma}_0$$

with quadratic and positive semi-definite penalty matrix $\mathbf{K}_2 \in \mathbb{Z}^{J \times J}$. The construction of the penalty matrices for smooth covariate effects occurs in the same fashion, for details see e.g. Fahrmeir *et al.* (2007).

The smoothing parameters need to be estimated from the data as well. In practice, we alternate between estimation of the smoothing parameters and the optimization of the log-likelihood (6) while assuming the current estimates of the smoothing parameters to be fixed. In the package **geonet**, estimation of the smoothing parameters is based on the generalized Fellner-Schall method which yields a simple update formula of the smoothing parameters in generalized additive models (Wood and Fasiolo 2017). Since the above model formulation is in fact a GAM, we can simply adapt this formula to our approach. If $\hat{\boldsymbol{\theta}}_\rho$ is the current estimate of the model parameters which have been obtained with smoothing parameters $\boldsymbol{\rho}$, an update of the smoothing parameter ρ_l is

$$\rho_l^{\text{new}} = \rho \frac{\text{tr}(\mathbf{K}_\rho^- \mathbf{K}_l) - \text{tr}((\mathbf{Z}^\top \mathbf{W}(\hat{\boldsymbol{\theta}}_\rho) \mathbf{Z} + \mathbf{K}_\rho)^{-1} \mathbf{K}_l)}{\hat{\boldsymbol{\theta}}_\rho^\top \mathbf{K}_l \hat{\boldsymbol{\theta}}_\rho}. \quad (7)$$

Here, $\text{tr}(\cdot)$ denotes the trace operator and superscript $-$ denotes the generalized inverse of a matrix. The matrix \mathbf{K}_l is the penalty matrix for index l filled with zeros such that it fits the dimensions of $\boldsymbol{\theta}$, and $\mathbf{K}_\rho = \sum_{l \in \{0\} \cup \mathcal{S}} \rho_l \mathbf{K}_l$. Moreover, \mathbf{Z} is the design matrix of the model and $\mathbf{W}(\hat{\boldsymbol{\theta}}_\rho)$ is a diagonal matrix with entries $\lambda_{m,k}$ and estimated model parameters $\hat{\boldsymbol{\theta}}_\rho$. It is important to note that all these matrices are sparse, i.e. most of the entries are equal to zero. This gets more important later when we discuss the implementation of the algorithm. A quantity that is often of interest are the effective degrees of freedom (edf) of a

model parameter. According to Wood (2017), an estimate of the effective degrees of freedom of the i -th coefficient θ_i is the i -th diagonal element of the matrix

$$\mathbf{F}(\hat{\boldsymbol{\theta}}) = (\mathbf{Z}^\top \mathbf{W}(\boldsymbol{\theta}) \mathbf{Z} + \mathbf{K}_\rho)^{-1} \mathbf{Z}^\top \mathbf{W}(\boldsymbol{\theta}) \mathbf{Z}.$$

Thus, if the first J entries of $\boldsymbol{\theta}$ correspond to the vector of coefficients $\boldsymbol{\gamma}_0$, the effective degrees of freedom of the estimated baseline intensity are $\text{edf}_0 = \sum_{j=1}^J \mathbf{F}_{jj}(\hat{\boldsymbol{\theta}})$. Note that the edf of each unpenalized coefficient is equal to one.

Finally, we can estimate the uncertainty of the model parameters, yielding confidence intervals for covariate effects and the baseline intensity. This is based on the asymptotic posterior distribution $\hat{\boldsymbol{\theta}} \stackrel{a}{\sim} \mathcal{N}(\boldsymbol{\theta}, \mathbf{V}(\boldsymbol{\theta}))$ of the model parameters, where

$$\mathbf{V}(\boldsymbol{\theta}) = (\mathbf{Z}^\top \mathbf{W}(\boldsymbol{\theta}) \mathbf{Z} + \mathbf{K}_\rho)^{-1}. \quad (8)$$

is the inverse of the penalized Fisher information. In practice, we make use of the Bayesian large sample approximation which yields confidence intervals for the parameter estimates $\hat{\boldsymbol{\theta}}$ in terms of the Fisher information $\mathbf{V}(\hat{\boldsymbol{\theta}})$ evaluated at the maximum likelihood estimate $\hat{\boldsymbol{\theta}}$.

Implementation

The function `intensity_pspline` allows estimating the intensity of a point process on a linear network based on the methodology which we have presented above. The function can be used by calling

```
intensity_pspline(
  X,
  ...,
  formula = ~1,
  delta = "0",
  h = "0.5",
  r = 2,
  scale = NULL,
  density = FALSE,
  verbose = FALSE,
  control = list()
)
```

Since `intensity_pspline` is the main routine of the package *geonet*, we explain it in detail here. The only argument which the user is required to supply is the point pattern on a geometric network `X`, an object of class `gnpp`. By default, the function fits the intensity without covariates. A model with covariates can be specified through the `formula` argument, in the same manner as in the function `gam` from the package *mgcv* (Wood 2017). However, the only smoothers which the package *geonet* supports are penalized splines, i.e. the `s` function uses by default the argument `bs = "ps"`. The left-hand side of the formula can be left blank. Otherwise, it is ignored. With the arguments `delta` and `h` we can specify the global knot distance δ and the global bin width h which are used to compute the curve specific knot distances δ_m and bin widths h_m , respectively, see Schneble and Kauermann (2020). In general, the knot distance can be supplied as a numeric value or as a quantile of the half of

all curve lengths. In the latter case, the quantile needs to be specified as a character. For example, `delta = "0.05"` sets δ to the 0.05 quantile of the set $\{d_m/2, m = 1, \dots, M\}$. By default, $\delta = \frac{1}{2} \min\{d_m\}$. The global bin width h can be supplied as a numerical value or a fraction of δ . In the latter case, the argument needs to be set in quotes. For example, the default `h = "0.5"` sets h to $h = 0.5\delta$. The argument `r` determines the order of the penalty, which is used for penalizing the coefficients γ_0 . By default, a second-order penalty is employed. It is also possible to penalize the first-order differences of the coefficients by setting `r = 1`. The `scale` argument allows scaling internal covariates, details are discussed later in Section 5. Finally, by setting `density = TRUE`, the fitted intensity function is scaled such that it integrates to one, i.e. it can be interpreted as a density function of the point process. Setting `verbose=TRUE` prints information with regards to the progress of the intensity fit to the console, such as the expected remaining computation time. This might be helpful if data on vast networks are fitted.

A fitted intensity on a geometric network is an object of class `gnppfit` which has, among others, the following attributes. The attributes `$coefficients` and `$V` are the maximum likelihood estimates $\hat{\theta}$ and the covariance matrix $V(\hat{\theta})$ of the parameter estimates. Furthermore, the fitting routine `intensity_spline` returns the specification of the knots (`$knots`) and the bins (`$bins`). There also is a `summary` method for an object of class `gnpp` which is illustrated in the Section 5.

For a first illustration, we fit the Chicago data with the function `intensity_pspline` when using its default arguments. Assuming that this point pattern of size $n = 116$ is the realization of a point process \mathcal{X} , we want to estimate its intensity function $\varphi_{\mathcal{X}}$. The resulting intensity estimate and the data on top are shown in the left panel of Figure 4, which is produced by the `plot.gnppfit` method for a fitted point process on a geometric network. Setting the `trans` argument of the `plot` method to `"sqrt"` shows the intensities on a square root scale.

```
R> X <- as_gnpp(chicago)
R> fit <- intensity_pspline(X)
R> g <- plot(fit, frame = TRUE, data = TRUE, trans = "sqrt")
```

4.2. Intensity estimation based on kernel smoothing

Methodology

The current state-of-the-art tools for intensity estimation of a point process, defined on a linear network, are primarily based on kernel smoothing techniques. However, if the network geometry is not considered appropriately, intensity estimates which result from kernel smoothing are biased if the actual intensity is uniform. This result was shown by [Okabe, Satoh, and Sugihara \(2009\)](#) who have proposed two methods that do not suffer from this drawback, see also [Okabe and Sugihara \(2012\)](#) for more details. Consequently, kernel smoothing on networks differs from kernel smoothing in Euclidean spaces. The general idea is to equally split the kernel mass appropriately when approaching a vertex with a degree of three or more from one side, which leads to discontinuities of the intensity estimate at the network's vertices. Therefore, this method is known as “equal-split discontinuous” method. In a similar manner, the “equal-split continuous” method adapts the kernels around the vertices such that they

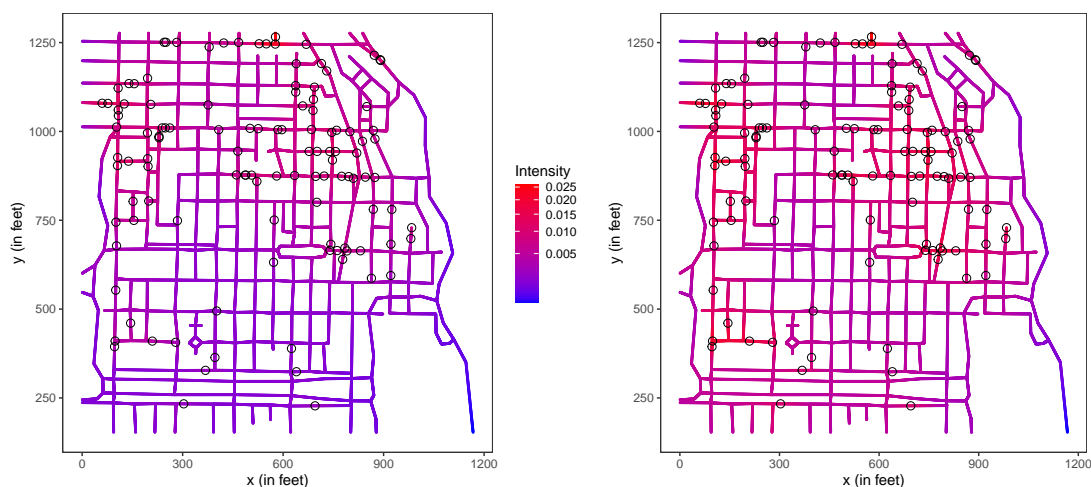


Figure 4: Penalized spline-based intensity estimate of the Chicago crimes data. Left panel: Fit without covariates. Right panel: Fit with covariates.

are continuous but still do produce unbiased estimates. One drawback of the “equal-split” methods is the slow computation time if the bandwidth of the kernel increases, especially when using a Gaussian kernel. [McSwiggan, Baddeley, and Nair \(2017\)](#) developed a method that solves a heat equation on the network and which is asymptotically equivalent to the “equal-split continuous” estimator as proposed by [Okabe *et al.* \(2009\)](#). In order to find the optimal bandwidth of the kernel, likelihood cross-validation can be employed, see [McSwiggan, Baddeley, and Nair \(2020\)](#). An efficient method for intensity estimation on large-scale linear networks was proposed by [Rakshit, Davies, Moradi, McSwiggan, Nair, Mateu, and Baddeley \(2019\)](#). The approach builds on two-dimensional kernels and the fast Fourier transform and the optimal bandwidth is chosen according to Scott’s rule of thumb ([Scott 2015](#)).

Implementation

Kernel smoothing methods on linear networks are implemented within the `spatstat.linnet` package. The method `density.lpp` for the generic function `density` allows the user to choose between various kernel smoothing algorithms discussed above. The function can be called via

```
density.lpp(x, sigma=NULL, ...,
            weights=NULL,
            distance=c("path", "euclidean"),
            continuous=TRUE,
            kernel="gaussian").
```

Here, `x` is a point pattern on a linear network to be smoothed (an object of class `lpp`) and `sigma` is the standard deviation (bandwidth) of the kernel, which needs to be estimated from the data. Likelihood cross-validation to find the optimal bandwidth of the kernel is implemented within the function `bw.lpp1` for methods based on the shortest path distance. The function `bw.scott.iso` returns an optimal bandwidth according to Scott’s rule of thumb for the method based on the Euclidean distance. Similar to kernel smoothing in one dimension,

different kernel functions such as a Gaussian kernel (default) or an Epanechnikov kernel (`kernel = "epanechnikov"`) can be used. By default, the heat kernel method developed by McSwiggan *et al.* (2017) is used. The two-dimensional kernel method is employed when setting `distance = "euclidean"`. A comparison of those two kernel smoothing implementations with an implementation of the penalized spline-based estimator on a linear network has been evaluated by Schneble and Kauermann (2020) for two simulation scenarios. The results suggest that if the true intensity is not smooth but has many discontinuities, the penalized spline-based estimator is superior in terms of the integrated squared error with respect to the true intensity. Furthermore, we note that intensity estimation with covariates is not supported by the method `density.lpp` for point patterns on linear networks.

5. Illustration

5.1. Crimes in a district of Chicago

As a first example, we consider the Chicago crimes network, which we have repeatedly treated above. In Figure 4 we have shown an intensity fit without covariates where we have used the default arguments of the fitting routine `intensity_p spline`. We further illustrate a model fit with covariates. The package `geonet` can fit the following internal covariates for every geometric network: `"x"` (x -coordinate), `"y"` (y -coordinate) and `"dist2V"` (distance to the closest vertex). We fit the intensity of the Chicago crimes data with the covariates mentioned above, i.e. we set the formula argument to `formula = marks + x + y + dist2V`. To avoid very small covariate effects, we further set `scale = list(x = 1/1000, y = 1/1000, dist2v = 1/1000)` which expresses the linear additive effects of these internal covariates in terms of per-1000-feet.

A summary of the model fit is shown below. The general form of the `summary` method for an object of class `gnpp` is similar to the summary of a fitted `gam` object from the `mgcv` package. In particular, a table is printed which shows the effects of the parametric coefficients, their standard errors and p-values. We can see that there is no effect of the x -coordinate, but there might be an effect of the y -coordinate, i.e. more crimes occur in the northern part of the network. The right panel of Figure 4 shows the baseline intensity fit. The effective degrees of freedom of the baseline intensity are equal to 35.5, while the edf of the intensity fitted without covariates amount to 38.8. Therefore, some of the variability of the true intensity $\varphi_{\mathcal{X}}(\cdot)$ can be explained by the covariates included in the above model.

```
R> X <- as_gnpp(chicago)
R> formula <- ~ marks + x + y + dist2V
R> scale <- list(x = 1/1000, y = 1/1000, dist2V = 1/1000)
R> model_covariates <- intensity_p spline(X, formula = formula, scale = scale)
R> summary(fit)
```

Intensity estimation on a geometric network in 2 dimensions
with 287 vertices and 452 curve segments.
Log-linear Poisson model fitted with maximum likelihood.

Global knot distance: 5.394

Global bin width: 2.697

Formula: ~marks + x + y + dist2V

Pparametric coefficients:

	Estimate	Std. Error	z value	Pr(> z)	
marksburglary	-1.38629	0.50000	-2.7726	0.005561	**
markscartheft	-1.04982	0.43915	-2.3906	0.016823	*
marksdamage	0.55962	0.28030	1.9965	0.045884	*
marksrobbery	-1.60944	0.54772	-2.9384	0.003299	**
markstheft	0.64185	0.27625	2.3234	0.020156	*
markstrespass	-1.20397	0.46547	-2.5866	0.009694	**
x	0.15837	1.04944	0.1509	0.880047	
y	1.70743	1.10215	1.5492	0.121337	
dist2V	-9.08850	6.79145	-1.3382	0.180822	

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Effective degrees of freedom of the baseline intensity: 35.515

Number of Fellner-Schall-iterations: 16

In a second step, we increase the global knots distance from the default $\delta = \frac{1}{2} \min d_m = 5.39$ feet to $\delta = 20$ feet. The resulting summary is printed below. We see that the effects of the type of a crime are the same as before and also, the effective degrees of freedom of the baseline intensity are almost the same. It appears that especially the effect of the x -coordinate is significantly different when increasing δ . However, the difference only amounts to $|0.158 - (-0.004)|/1.049 = 0.15$ standard deviations since the effect of x is highly non-significant. For the effects of y and the distance to the nearest vertex, these differences are only 0.032 and 0.018 standard deviations, respectively. Therefore, the differences are only minor but the second model needs fewer than 90% computation time due to the lower number of parameters contained in the model. We treat this trade-off between accuracy and computation time later in more detail in a simulation study.

```
R> fit <- intensity_pspline(X, formula = formula, scale = scale, delta = 20)
R> summary(fit)
```

Intensity estimation on a geometric network in 2 dimensions
with 287 vertices and 452 curve segments.
Log-linear Poisson model fitted with maximum likelihood.

Global knot distance: 20

Global bin width: 10

Formula: ~marks + x + y + dist2V

Pparametric coefficients:

	Estimate	Std. Error	z value	Pr(> z)	
marksburglary	-1.3862944	0.4999972	-2.7726	0.005561	**
markscartheft	-1.0498221	0.4391525	-2.3906	0.016823	*
marksdamage	0.5596158	0.2803044	1.9965	0.045884	*
marksrobbery	-1.6094379	0.5477195	-2.9384	0.003299	**
markstheft	0.6418539	0.2762516	2.3234	0.020156	*
markstrespass	-1.2039728	0.4654720	-2.5866	0.009694	**
x	-0.0038448	1.0853283	-0.0035	0.997174	
y	1.7427380	1.0971509	1.5884	0.112191	
dist2V	-8.9663884	6.8073547	-1.3172	0.187784	

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Effective degrees of freedom of the baseline intensity: 35.485

Number of Fellner-Schall-iterations: 9

5.2. Car crashes in Montgomery County, Maryland

The amount of publicly available datasets which governments provide has tremendously increased during the past years. An example is Montgomery County, located north of Washington, D.C. in the US-state Maryland (M.D.). The county runs the open data platform “dataMontgomery” which supplies data related to different areas such as business, education, transportation and public safety.³ The latter category comprises a dataset that contains information about traffic accidents on major streets in Montgomery County since January 2015. As in Schneble and Kauermann (2020) we restrict the data to accidents that have occurred on a highway (state highway, interstate highway or U.S. highway). The resulting network of highways builds the geometric network, which we consider in this illustration. The point pattern which we observe on this network represents traffic collisions. This point pattern on a geometric network is available as the object `montgomery` of class `gnpp` in the package `geonet`. The `summary` output of the `montgomery` object shows that the underlying network has a length of 175.419 kilometers and consists of $M = 103$ curve segments, which are built through 369 straight line segments in total. Two internal covariates represent the “type” of the highway (state highway, interstate highway or U.S.-highway) and the “direction” (south/north, east/west, southeast/northwest, southwest/northeast) of the street. The point pattern has a size of 14,571 and there is one external covariate that marks the “hour” of the day of each car crash.

```
R> summary(montgomery)
```

```
Point pattern on a geometric network in 2 dimensions with 73 vertices
and 103 curve segments.
```

```
The linear representation of the network has 339 vertices
and 369 straight line segments.
```

³<https://data.montgomerycountymd.gov/>

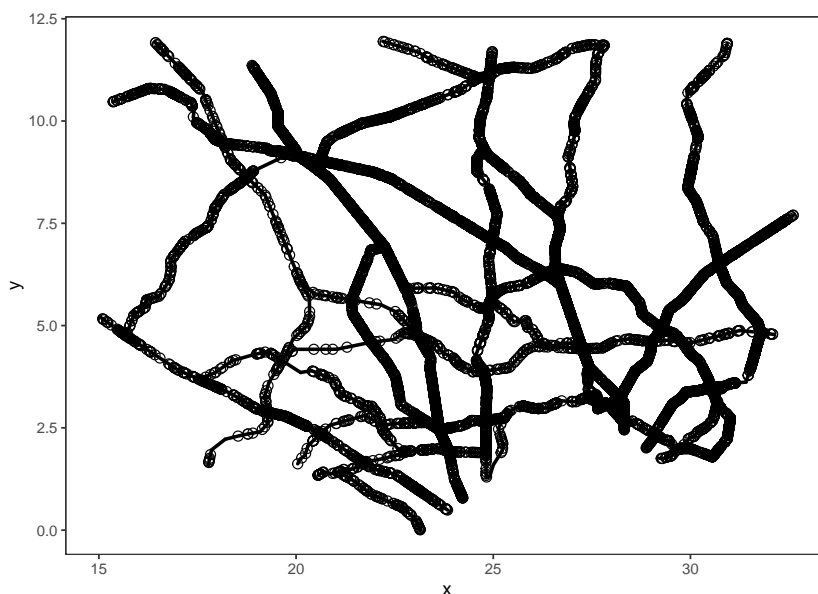


Figure 5: Car crashes on the Montgomery highways network.

Total length of the network: 175.419 kilometers

Number of network internal covariates: 2

1) factor variable "type":

state	interstate	US
304	51	14

2) factor variable "direction":

SN	EW	SE	SW	NE	SW	NE
100	114	106	49			

Number of points: 14571

Average intensity: 83.06418 points per kilometer

Number of external covariates: 1

3) numeric variable "hour":

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
6.00	10.00	14.75	14.08	17.75	21.75

Figure 5 shows the results of the `plot` method for the object `montgomery`. The network is denser in the south at the border to Washington, D.C., i.e. there is more network length per square kilometer than in the northern part of the network. Therefore, an analysis in the Euclidean space would not respect that there is more network mass and with the network approach, we can estimate the intensity in terms of “car crashes per kilometer street”. In contrast to the Chicago crimes data, the high-intensity regions are not apparent from the plot of the point pattern since there are just too many traffic accidents plotted in Figure 5. Therefore, we fit the intensity to the data, at first without covariates.

```
R> X <- montgomery
R> model <- intensity_pspine(X, delta = 0.05, r = 2)
R> plot(model, frame = TRUE, trans = "log")
```

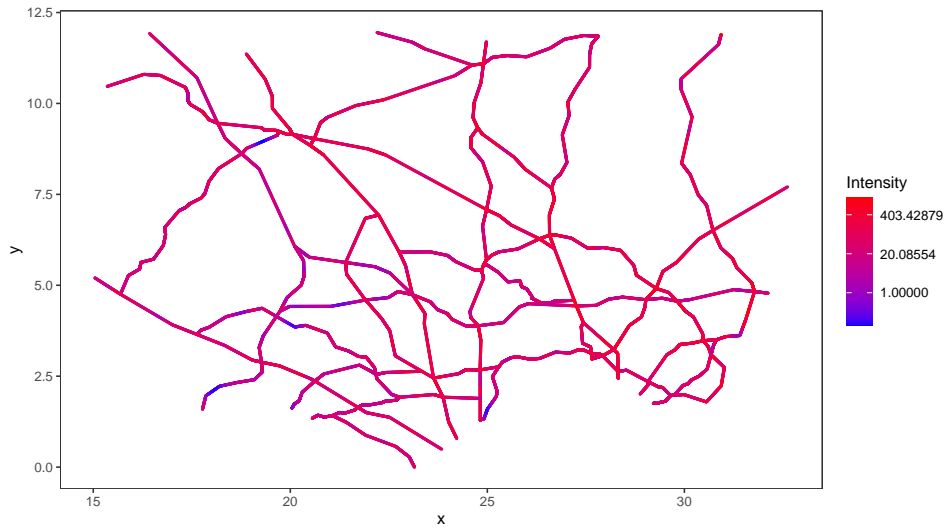


Figure 6: Intensity fit of the Montgomery traffic accidents data when fitted without covariates.

The resulting estimate of the intensity is shown in Figure 6 on a log-scale. We see that the high-intensity regions are mainly located on highways that run away from Washington, D.C. Moreover, the fitted intensity is much less smooth than the intensity fit of the Chicago crimes data. The smoothness can be quantified via the effective degrees of freedom of the penalized spline smoother, amounting to 1311. Note that the B-spline basis representation of the network involves 3,477 parameters which would be the degrees of freedom without penalization.

```
R> summary(model)
```

```
Intensity estimation on a geometric network in 2 dimensions
with 73 vertices and 103 curve segments.
Log-linear Poisson model fitted with maximum likelihood.
```

```
Global knot distance: 0.05
Global bin width: 0.025
```

```
Formula: ~1
```

```
Model has no parametric coefficients.
```

```
Effective degrees of freedom of the baseline intensity: 1311.009
```

```
Number of Fellner-Schall-iterations: 18
```

We fit a model with the covariates supplied along with the `montgomery` object as a next step. These are the internal factor variables `type` and `direction` as well as the external, numeric variable `hour`. The latter is included as a smooth effect using `mgcv`'s `s` function. The following summary output of the model fit shows the estimates of the linear effects.

```
R> formula <- ~ s(hour) + type + direction
R> model_covariates <- intensity_kspline(X, formula = formula,
+                                     delta = 0.05, r = 2)
R> summary(model_covariates)
```

Intensity estimation on a geometric network in 2 dimensions with 73 vertices and 103 curve segments.
Log-linear Poisson model fitted with maximum likelihood.

Global knot distance: 0.05

Global bin width: 0.025

Formula: ~s(hour) + type + direction

Pparametric coefficients:

	Estimate	Std. Error	z value	Pr(> z)
typeinterstate	-1.671148	0.197286	-8.4707	< 2e-16 ***
typeUS	0.251985	0.226239	1.1138	0.26537
directionEW	-0.137999	0.112644	-1.2251	0.22054
directionSENE	-0.078722	0.120030	-0.6559	0.51192
directionSWNE	-0.329414	0.168017	-1.9606	0.04993 *

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Effective degrees of freedom of the baseline intensity: 1270.611

Number of Fellner-Schall-iterations: 25

It can be seen that the risk of a car crash is significantly lower on an interstate highway when compared to a state highway. To quantify, the relative risk of a car accident decreases by factor $\exp(-1.671) = 0.188$. This finding can be explained by the fact that most interstate highways have separate lanes for each driving direction and the intersections with other streets are organized such that no cars are running into each other. The risk of an accident seems to be slightly higher on a U.S. highway, but the effect is not significant. Highways that run from southwest to northeast seem to exhibit the lowest risk of accidents. Those streets mainly run far away from Washington, D.C., so that we can expect much less traffic there. The baseline intensity is shown in Figure 7 on a log-scale. Finally, Figure 8 shows the smooth effect of the hour of the day. It can be seen that the risk of an accident rises from 6 am until 9 am by factor $\exp(0.25 - (-0.6)) = 2.34$, drops for the time around noon, and rises again to the highest level at 4 pm.

6. Relation to the package *spatstat*

In the previous sections, we have shown that some objects that belong to a *spatstat* specific class can be transmuted to the respective objects of the *geonet* classes. In particular, those are linear networks (objects of class *linnet*) and point pattern on linear networks (objects

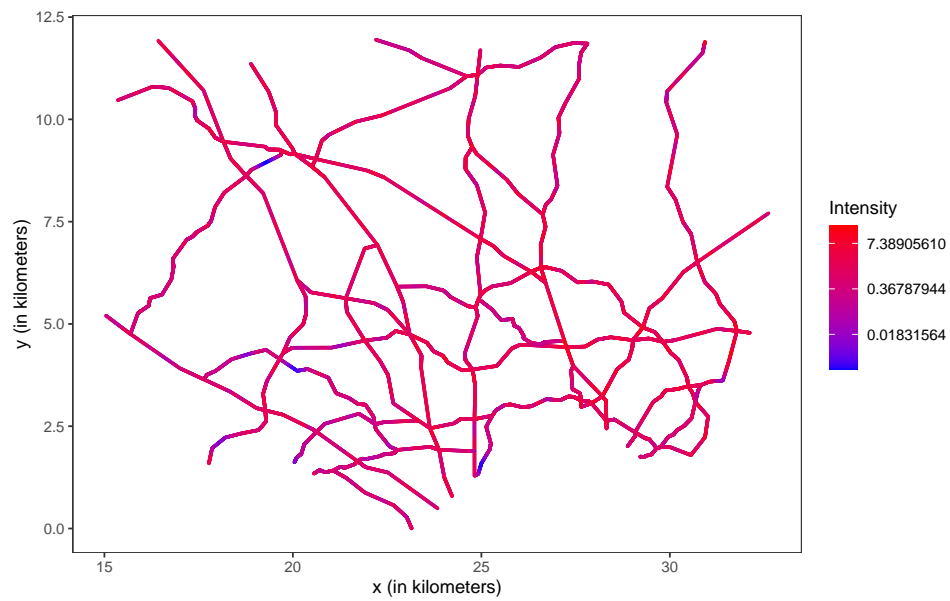


Figure 7: Baseline intensity fit of the Montgomery traffic accidents data when fitted with covariates.

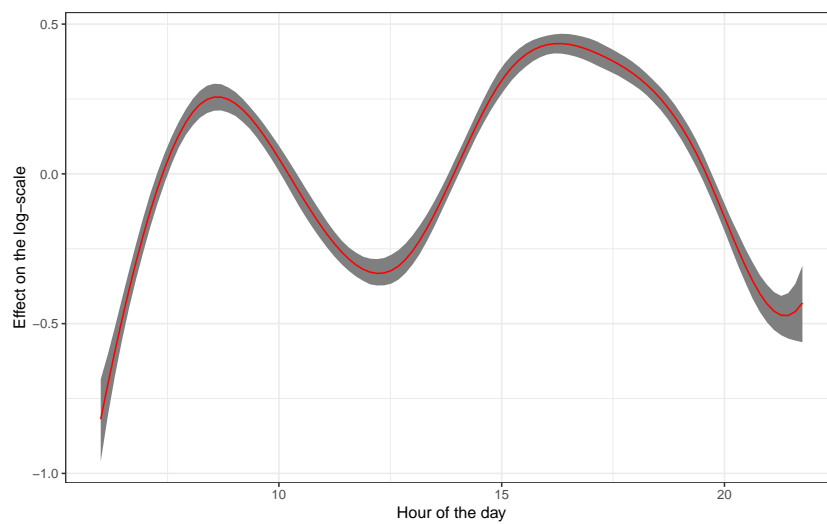


Figure 8: Smooth effect of the hour of the day. Grey area shows pointwise 95% confidence bands.

of class `lpp`). We now conduct a more detailed comparison of the two packages, which also involves the difference between representing a spatial network as a linear network or a geometric network. First, we note that also the package `spatstat` is currently restricted to $q = 2$, referring e.g. to street networks with v_i representing location coordinates.

The package `geonet` contains the method `as.linnet.gn` for transmuted objects of class `gn` to an object of class `linnet`. However, we see that if we transmute an object of class `linnet` to an object of class `gn` and back to a linear network, the linear network representation has changed in 18 components.

```
R> L <- as.linnet(chicago)
R> L2 <- as.linnet(as_gn(chicago))
R> length(all.equal(L, L2))
```

```
[1] 18
```

This finding is usually caused by the renumbering of the line segment identifiers when transmuting from one class representation to another. In order to make the transformation one-to-one, the method `as_gn.linnet` has an additional argument `spatstat`. Setting this argument to `TRUE` will make the transformation one-to-one.

```
R> L3 <- as.linnet(as_gn(L, spatstat = TRUE))
R> all.equal(L, L3)
```

```
[1] TRUE
```

Moreover, the `plot` method for an object of class `linnet` makes use of base R's plotting principles. In comparison to graphics created with the package `ggplot` as in the package `geonet`, this only allows for a limited scope of operation when plotting point patterns and intensity estimates on networks.

The package `geonet` further contains the wrapper function `intensity_kernel(X, kernel = "heat")` which allows to apply the kernel smoothing algorithms discussed above to a point pattern on a geometric network X . By default, the kernel smoothing technique developed by [McSwiggan *et al.* \(2017\)](#) which is based on the heat kernel (`kernel = "heat"`) is employed. Alternatively, by specifying the argument `kernel = "Euclidean"` the adaptive two-dimensional kernel smoother as proposed by [Rakshit *et al.* \(2019\)](#) is used. In both cases, estimation of the optimal bandwidth is carried out automatically with the respective optimization routines.

The package `spatstat.linnet` allows generating random point patterns \mathbf{x} according to an intensity which is either specified as a function on a linear network or which results from an intensity fit using the method `density.lpp`. The functions `runiflpp` and `rlpp` return a point pattern of a specified sample size n , where the former generates random points according to a uniform distribution on a linear network and the latter generates random points according to a specified probability density which is normalized if an intensity is supplied. Moreover, the function `rpoislpp` generates a realization of a Poisson process from a specified intensity function. The package `geonet` has similar functions which allow the user to simulate from an

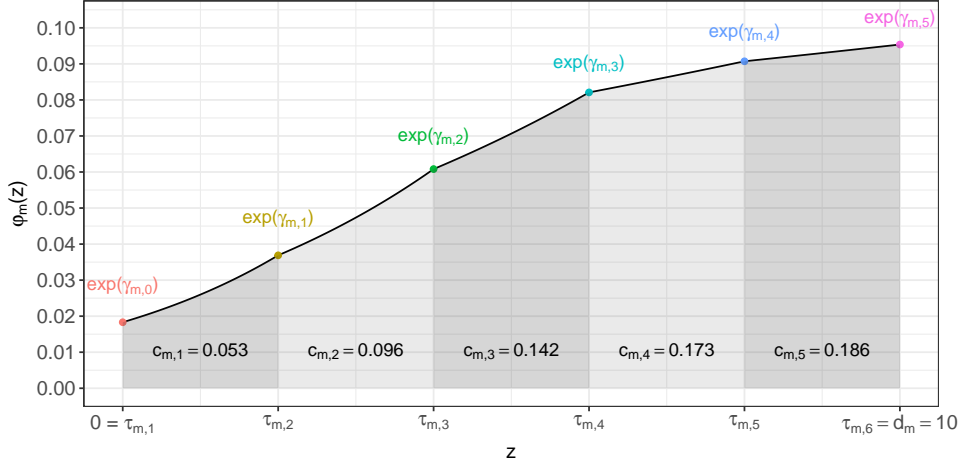


Figure 9: Representation of the intensity fitted on the m -th network segment with length $d_m = 10$ on the real numbers.

intensity estimate, where the logarithm of the intensity has a B-spline basis representation. Details are supplied in the following section.

7. Simulation of point processes on a geometric network

First, the function `runifgn(n, G)` returns a point pattern of size n which is drawn from a uniform distribution on the geometric network G . Note that for a uniform intensity it holds that $\gamma_j \equiv \gamma$ for $j = 1, \dots, J$. Second, we exploit the representation of the log-baseline intensity as a linear combination of linear B-splines (3) in order to simulate a point pattern from a fitted intensity. Therefore, we define for $m = 1, \dots, M$ a function φ_m on the real numbers which has the same image set as the function $\varphi_{\mathcal{X}}$ restricted to the curve e_m . That is, we define the function $\varphi_m : [0, d_m] \rightarrow [0, \infty)$ on the real numbers which satisfies $\varphi_{\mathcal{X}}(e_m) = \varphi_m([0, d_m])$. Furthermore, we denote with $\tau_{m,k}$ the respective projection of $\boldsymbol{\tau}_{m,k}$ to the interval $[0, d_m]$. In particular, it holds that $d_G(\boldsymbol{\tau}_{m,k}, \boldsymbol{\tau}_{m,k+1}) = \tau_{m,k+1} - \tau_{m,k}$, i.e. the shortest path distance on the network segment matches the Euclidean distance in the one dimensional space.

Figure 9 exemplifies the situation on the m -th segment. The intensity on e_m which has endpoints v_i and v_j is shaped by the parameters $\gamma_{m,0}, \gamma_{m,1}, \dots, \gamma_{m,J_m}, \gamma_{m,J_m+1}$, where we define with $\gamma_{m,0} = \gamma_{(i)}$ and $\gamma_{m,J_m+1} = \gamma_{(j)}$ the coefficients of the B-splines $B_{(i)}$ and $B_{(j)}$. At the location of the k -th knot $\boldsymbol{\tau}_{m,k}$, the intensity has value $\varphi_{\mathcal{X}}(\boldsymbol{\tau}_{m,k}) = \exp(\gamma_{m,k-1})$. Moreover, the density between two adjacent knots $\boldsymbol{\tau}_{m,k}$ and $\boldsymbol{\tau}_{m,k+1}$ is the intensity normalized by the integral

$$c_{m,k} = \int_{[\tau_{m,k}, \tau_{m,k+1}]} \varphi_{\mathcal{X}}(\mathbf{u}) \, d\mathbf{u} = \int_{\tau_{m,k}}^{\tau_{m,k+1}} \varphi_m(u) \, du$$

We now define with Z_m a random variable which is supported on $[0, d_m]$ and has a density which is proportional to $\varphi_m(\cdot)$. Putting the above arguments together, the conditional density of $Z_m \mid Z_m \in [\tau_{m,k}, \tau_{m,k+1}]$ is given by

Algorithm 1: Simulating n points from a specified density.

Data: The sample size n and an intensity function $\varphi_{\mathcal{X}}$, where $\nu_{\mathcal{X}} = \log \varphi_{\mathcal{X}}$ has a B-spline basis representation.

Result: Point pattern of size n on a geometric network.

begin

 Compute the integrals $c_{m,k}$ for $m = 1, \dots, M$ and $k = 1, \dots, I_m - 1$;

 Sample n points from a discrete distribution with probabilities proportional to $c_{m,k}$;

for $m \in 1, \dots, M$ **do**

for $k \in 1, \dots, I_m - 1$ **do**

 If $n_{m,k} \in \mathbb{N}_0$ points are sampled on the interval $[\tau_{m,k}, \tau_{m,k+1}]$, simulate vector of $n_{m,k}$ independent $\mathcal{U}(0, 1)$ random variables $\mathbf{u}_{m,k}$;

$\mathbf{x}_{m,k} = F_{m,k}^{-1}(\mathbf{u}_{m,k})$;

 Project $\mathbf{x}_{m,k}$ back to the geometric network;

end

end

end

$$\begin{aligned} f_{m,k}(z) &= f(z \mid Z_m \in [\tau_{m,k}, \tau_{m,k+1}]) = \frac{1}{c_{m,k}} \exp \left[\gamma_{m,k-1} + \left(\frac{\gamma_{m,k} - \gamma_{m,k-1}}{\delta_m} \right) (z - (k-1)\delta_m) \right] \\ &= \frac{1}{c_{m,k}} \exp \left[k\gamma_{m,k-1} + \gamma_{m,k}(1-k) + \left(\frac{\gamma_{m,k} - \gamma_{m,k-1}}{\delta_m} \right) z \right] \end{aligned}$$

for $z \in [\tau_{m,k}, \tau_{m,k+1}]$. Thus, the conditional distribution function of Z_m results to

$$\begin{aligned} F_{m,k}(z) &= \int_{\tau_{m,k}}^z f_{m,k}(y \mid Z_m \in [\tau_{m,k}, \tau_{m,k+1}]) dy \\ &= \frac{\delta_m \exp(k\gamma_{m,k-1} + \gamma_{m,k}(1-k))}{c_{m,k}(\gamma_{m,k} - \gamma_{m,k-1})} \cdot \left[\exp \left(\frac{\gamma_{m,k} - \gamma_{m,k-1}}{\delta_m} z \right) - \exp \left(\frac{\gamma_{m,k} - \gamma_{m,k-1}}{\delta_m} \tau_{m,k} \right) \right] \end{aligned}$$

for $z \in [\tau_{m,k}, \tau_{m,k+1}]$. Finally, the inverse of the conditional distribution function, i.e. the conditional quantile function, is for $u \in [0, 1]$ given by

$$F_{m,k}^{-1}(u) = \log \left[\frac{uc_{m,k}(\gamma_{m,k} - \gamma_{m,k-1})}{\delta_m \exp(k\gamma_{m,k-1} + \gamma_{m,k}(1-k))} + \exp \left(\frac{\gamma_{m,k} - \gamma_{m,k-1}}{\delta_m} \tau_{m,k} \right) \right] \cdot \frac{\delta_m}{\gamma_{m,k} - \gamma_{m,k-1}}.$$

The function `rgnpp(n, fit)` returns a point pattern of size `n` that is simulated from the fitted intensity `fit`, which needs to be supplied as an object of class `gnppfit`. The routine is based on the inversion method (Devroye 1986) and follows pseudo algorithm 1. Moreover, the function `rpoisgnpp(fit)` returns a realization of a Poisson process which is generated according to the object `fit`. The routine follows algorithm 2.

Algorithm 2: Simulating a Poisson process from an intensity fit.

Data: An intensity function $\varphi_{\mathcal{X}}$, where $\nu_{\mathcal{X}} = \log \varphi_{\mathcal{X}}$ has a B-spline basis representation.

Result: A realization of a Poisson process on a geometric network.

begin

 Compute the integrals $c_{m,k}$ for $m = 1, \dots, M$ and $k = 1, \dots, I_m - 1$;

for $m \in 1, \dots, M$ **do**

for $k \in 1, \dots, I_{m-1}$ **do**

 Sample $n_{m,k} \in \mathbb{N}_0$ points from a Poisson distribution with parameter $\lambda = c_{m,k}$;

 Simulate vector of $n_{m,k}$ independent $\mathcal{U}(0, 1)$ random variables $\mathbf{u}_{m,k}$;

$\mathbf{x}_{m,k} = F_{m,k}^{-1}(\mathbf{u}_{m,k})$;

 Project $\mathbf{x}_{m,k}$ back to the geometric network;

end

end

end

8. Computational aspects

In this section, we highlight some computational aspects of our implementation. Therefore, we first vary the arguments `delta` and `h` when fitting the model to simulated data on the Chicago network. The model specifications and the respective results are summarized in Table 1. The first two columns define the varying arguments `delta` and `h`. The following four columns show the resulting global knot distance δ and the global bin width h along with the number of parameters J and the total number of bins N . The next two rows indicate the memory (in megabyte) which is used to store the design matrix \mathbf{Z} and the penalty matrix \mathbf{K} if a non-sparse (memNS) or a sparse (memS) matrix representation is used, respectively. Note that both matrices do not depend on the data if a model without external covariates is fitted. We see that the sparse matrix representation that is employed by the `geonet` package reduces the memory that is needed to store the huge matrices \mathbf{Z} and \mathbf{K} by more than 99% when compared to storing them as an object of class `matrix`, which lessens the computation time and allows to fit point processes on much larger networks.

Moreover, we generate point patterns of size $n = 200$ using the function `rgnpp`, where the true intensity $\varphi_{\mathcal{X}}$ is assumed to be proportional to the fitted intensity shown in the left panel of Figure 4. For each setting, we repeat the data generating process and the intensity fit to the simulated data using the function `intensity_pspline` $R = 100$ times. We compare in the last two columns of Table 1 the settings in terms of the mean computation time in seconds (MCT) and the mean integrated squared error (MISE). As in Schneble and Kauermann (2020), we define the integrated squared error (ISE) of an estimate $\hat{\varphi}_{\mathcal{X}}(\cdot)$ for the true intensity $\varphi_{\mathcal{X}}(\cdot)$ as

$$\text{ISE}(\hat{\varphi}_{\mathcal{X}}) = \frac{1}{n^2} \int_{\mathcal{G}} (\varphi_{\mathcal{X}}(\mathbf{u}) - \hat{\varphi}_{\mathcal{X}}(\mathbf{u}))^2 d\mathbf{u}. \quad (9)$$

The constant n^{-2} works as a normalization constant such that (9) results as the ISE in terms of the density $f_{\mathcal{X}}(\cdot) = \varphi_{\mathcal{X}}(\cdot)/n$, thus generally allowing to compare the intensity fit for different sample sizes. The MISE is defined to be the mean of the ISE over R replications. We see that the computation time increases with decreasing δ , which is apparent since this choice determines the number of parameters. However, there is no appreciable effect on the

<code>delta</code>	<code>h</code>	δ	h	J	N	memNS	memS	MCT	MISE·10 ⁶
"0.2"	"0.5"	20.77	10.38	1396	3018	47.0	0.18	6.6	5.25
"0.2"	"0.1"	20.77	2.08	1396	14997	174.6	0.46	5.5	5.41
"0.1"	"0.5"	14.72	7.36	1985	4230	94.1	0.25	13.5	5.16
"0.1"	"0.1"	14.72	1.47	1698	21133	350.1	0.64	14.4	5.37
"0.05"	"0.5"	12.04	6.02	2442	5161	141.7	0.30	17.1	5.47
"0.05"	"0.1"	12.04	1.20	2442	25867	527.4	0.78	18.4	5.60
"0.025"	"0.5"	8.21	4.11	3643	7589	312.2	0.44	45.6	5.14
"0.025"	"0.1"	6.69	0.67	3643	37920	1155.2	1.13	47.9	5.45
"0"	"0.5"	5.39	2.69	5582	11532	728.8	0.65	119.8	5.16
"0"	"0.1"	5.39	0.54	5582	57755	2697.4	1.71	119.4	5.52

Table 1: Memory required (in megabyte) to store matrices \mathbf{Z} and \mathbf{K} when using a non-sparse (memNS) or a sparse (memS) matrix representation, mean computation time in seconds (MCT) and mean integrated squared difference (MISE, multiplied by factor 10^6) for different arguments of `delta` and `h` of the intensity fitting routine averaged over $R = 100$ simulations. The second last row corresponds to the default arguments.

MISE that can be attributed to the specification of `delta` and `h`. This result suggests that the computation time can be reduced without losing the accuracy of the estimates.

We finally discuss some details concerning the implementation of `intensity_p spline` in R. The sequence of knots on each curve of the geometric network and the bins are constructed as described above in order to create the design matrix \mathbf{Z} . The first J columns of this matrix refer to the linear B-splines defined on the network. If there appear linear covariates specified in the `formula`, the respective columns of the design matrix are computed using the `model.matrix` function from the package `stats` and the design matrix for the smooth components are computed using the function `smooth.construct.ps.smooth.spec` from the package `mgcv`. To ensure the identifiability of smooth covariate effects, the respective columns of the design matrix are centered around zero. The columns of the design matrix that refer to the linear B-spline representation of the log-baseline intensity are not centered, thus including the intercept. If there are external covariates, the design matrix will be repeated for every unique combination of external covariates with the respective values of the external covariates. The construction of the penalty matrix \mathbf{K} follows the methodology developed above. Both matrices, \mathbf{Z} and \mathbf{K} , are stored as sparse matrices since most of their entries are equal to zero. When all matrices have been stored in the memory, the model parameters $\boldsymbol{\theta}$ are estimated with a nested iterative algorithm. In the outer loop, the smoothing parameters are updated as in (7). Since the dimension of the parameter vector $\boldsymbol{\theta}$ can easily be in the range of several thousands, the update formula requires to compute the generalized inverse of a large matrix which is very expensive in terms of computation time. However, since we merely need the trace of the matrix $(\mathbf{K}_\rho)^- \mathbf{K}$, we can use the fact that

$$\text{tr}((\mathbf{K}_\rho)^- \mathbf{K}_l) = \frac{1}{\rho_l} \text{rk}(\mathbf{K}_l). \quad (10)$$

Therefore, we only need to compute the rank of \mathbf{K}_l which needs computation time in the order of $\mathcal{O}(J^2)$ instead of $\mathcal{O}(J^3)$ for the generalized inverse if $\gamma_0 \in \mathbb{R}^J$ (Trefethen and Bau III 1997). By exploiting the relation (10), updating the smoothing parameters does not significantly contribute to the overall computation time of the algorithm.

Algorithm 3: Fisher-scoring algorithm.

Data: Design Matrix \mathbf{Z} , penalty matrix \mathbf{K} , smoothing parameters $\boldsymbol{\rho}$, offsets h_m for $m = 1, \dots, M$ and threshold ϵ_θ .

Result: Maximum likelihood estimate for a given vector of smoothing parameters.

```

begin
   $\hat{\boldsymbol{\theta}}^{(0)} = \mathbf{0}$ ;
   $\Delta_\theta = \infty$ ;
   $k = 0$ ;
  while  $\Delta_\theta > \epsilon_\theta$  do
     $\hat{\boldsymbol{\theta}}^{(k+1)} = \hat{\boldsymbol{\theta}}^{(k)} + \mathbf{V}(\hat{\boldsymbol{\theta}}^{(k)}) \cdot s(\hat{\boldsymbol{\theta}}^{(k)})$ ;
     $\Delta_\theta = \|\hat{\boldsymbol{\theta}}^{(k+1)} - \hat{\boldsymbol{\theta}}^{(k)}\| / \|\hat{\boldsymbol{\theta}}^{(k)}\|$ ;
     $k = k + 1$ ;
  end
end

```

In the inner loop, the penalized log-likelihood (6) for given smoothing parameters is maximized by employing a penalized version of the Fisher-scoring algorithm. This is equivalent to the penalized iterative weighted least squares (PIRLS) algorithm which the `gam` routine of the package `mgcv` makes use of (Wood 2017). The Fisher-scoring algorithm proceeds as shown in Algorithm 3 and makes use of the (penalized) score function $s(\cdot)$ which is defined by $s(\boldsymbol{\theta}) = \frac{\partial}{\partial \boldsymbol{\theta}} \ell_{\text{penal}}(\boldsymbol{\theta})$.

Algorithm 3 repeatedly requires the computation of matrix products and inverse matrices. The dimension of these matrices is usually enormous. Therefore, we store all matrices in a sparse format making use of the `Matrix` package. As shown above, this reduces the memory needed but also lessens the computation time of basic matrix operations significantly since most of the entries of the matrices \mathbf{Z} , \mathbf{W} and \mathbf{K} are equal to zero. For the design matrix, this holds since most of the columns refer to the linear B-spline representation of the log-baseline intensity. For this reason, we do not center the respective columns of \mathbf{Z} and instead include the intercept in the log-baseline intensity.

9. Discussion

The R package `geonet` is designed for intensity estimation of point processes on geometric networks, which we understand as an advanced representation of linear networks. The methodology uses penalized spline smoothing employed in connection with the comprehensive class of generalized additive models. Our package is, for the most part, compatible with the current state-of-the-art software `spatstat`, which builds on kernel smoothing to estimate the intensity of point processes on linear networks.

Our methodology easily allows to include covariates while estimating the baseline intensity of a point process. We see this as the biggest advantage over the comparable method `density.lpp` from the package `spatstat`. Nonetheless, Poisson processes with covariates can be fitted with `spatstat`, but the baseline intensity needs to be supplied as an offset, i.e. simultaneous estimation is not possible. Moreover, the methodology, which is the basis of our software contribution, builds on a well-developed field in statistics. Another benefit of the package `geonet`

is its integration into the **tidyverse**, a bundle of R packages that share a common design and grammar. In particular, the package **dplyr** enables convenient data handling and our **plot** methods make use of the package **ggplot2**, which allows to create advanced plots compared to base R plots. However, there are also drawbacks that our method and its implementation in R inherit. Modeling the baseline intensity of a large network through B-splines requires many parameters in the model. In connection with the binning of the data, model matrices are huge and even though we choose a sparse matrix representation, this will need more resources in terms of computation time and required memory when compared to a kernel-based approach. We pursue the following extensions of our package. First, the methodology which we treated in Section 4.1 easily allows fitting the intensity of a point process on a network embedded in higher dimensional spaces. In contrast to smoothing with penalized splines in higher-dimensional Euclidean spaces, this does not generally increase the number of parameters and, therefore, the complexity of the model. Networks that are embedded in three-dimensional spaces can be plotted with the **plotly** package. However, networks in more than two dimensions are rarely available and also, the **spatstat** package is only capable of representing linear networks in the plane. Second, instead of representing a curve segment as the alignment of many straight line segments as in the current version of the **geonet** package, one could represent such a connection between two vertices as a parametric differentiable curve as generally proposed by Schneble and Kauermann (2020). Note that the current implementation allows an arbitrary approximation of such a curve, requiring much more memory with increasing precision. Third, we have considered networks so far as being undirected in general. In reality, most of the networks do have a direction. For example, car crashes can occur in either driving direction or a one-way street. In addition to these proposed extensions, which require significant changes within the package’s functionality and are sent to CRAN, we release minor changes to the package via GitHub.

References

- Ang QW, Baddeley A, Nair G (2012). “Geometrically corrected second order analysis of events on a linear network, with applications to ecology and criminology.” *Scandinavian Journal of Statistics*, **39**(4), 591–617.
- Baddeley A, Rubak E, Turner R (2015). *Spatial point patterns: Methodology and applications with R*. CRC press.
- Bivand RS, Pebesma E, Gomez-Rubio V (2013). *Applied spatial data analysis with R, Second edition*. Springer, NY. URL <https://asdar-book.org/>.
- De Boor C (1978). *A practical guide to splines*, volume 27. Springer, New York, NY.
- Devroye L (1986). *Non-Uniform Random Variate Generation*, volume 1. Springer, New York, NY.
- Eilers PH, Marx BD (1996). “Flexible smoothing with B-splines and penalties.” *Statistical science*, **11**(2), 89–121.
- Fahrmeir L, Kneib T, Lang S, Marx B (2007). *Regression*. Springer.

- Hastie T, Tibshirani R (1986). “Generalized Additive Models.” *Statistical Science*, **1**(3), 297–310.
- Kolaczyk ED, Csárdi G (2014). *Statistical analysis of network data with R*, volume 65. Springer.
- Lovelace R, Ellison R (2018). “stplanr: A Package for Transport Planning.” *The R Journal*, **10**(2). URL <https://doi.org/10.32614/RJ-2018-053>.
- McSwiggan G, Baddeley A, Nair G (2017). “Kernel density estimation on a linear network.” *Scandinavian Journal of Statistics*, **44**(2), 324–345.
- McSwiggan G, Baddeley A, Nair G (2020). “Estimation of relative risk for events on a linear network.” *Statistics and Computing*, **30**(2), 469–484.
- Okabe A, Satoh T, Sugihara K (2009). “A kernel density estimation method for networks, its computational method and a GIS-based tool.” *International Journal of Geographical Information Science*, **23**(1), 7–32.
- Okabe A, Sugihara K (2012). *Spatial analysis along networks: Statistical and computational methods*. John Wiley & Sons.
- Pebesma E (2018). “Simple Features for R: Standardized Support for Spatial Vector Data.” *The R Journal*, **10**(1), 439–446. doi:10.32614/RJ-2018-009. URL <https://doi.org/10.32614/RJ-2018-009>.
- R Core Team (2021). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. URL <https://www.R-project.org/>.
- Rakshit S, Davies T, Moradi MM, McSwiggan G, Nair G, Mateu J, Baddeley A (2019). “Fast Kernel Smoothing of Point Patterns on a Large Network using Two-dimensional Convolution.” *International Statistical Review*, **87**(3), 531–556.
- Ruppert D, Wand MP, Carroll RJ (2003). *Semiparametric regression*. 12. Cambridge University Press.
- Schneble M, Kauermann G (2020). “Intensity Estimation on Geometric Networks with Penalized Splines.” *arXiv preprint arXiv:2002.10270*.
- Scott DW (2015). *Multivariate density estimation: theory, practice, and visualization*. John Wiley & Sons.
- Trefethen LN, Bau III D (1997). *Numerical linear algebra*, volume 50. Siam.
- Ver Hoef J, Peterson E, Clifford D, Shah R (2014). “SSN: An R package for spatial statistical modeling on stream networks.” *Journal of Statistical Software*, **56**(3), 1–45.
- Wickham H (2016). *ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York. ISBN 978-3-319-24277-4. URL <https://ggplot2.tidyverse.org>.
- Wickham H, François R, Henry L, Müller K (2021). *dplyr: A Grammar of Data Manipulation*. R package version 1.0.6, URL <https://CRAN.R-project.org/package=dplyr>.

Wood SN (2017). *Generalized additive models: An introduction with R*. CRC press.

Wood SN, Fasiolo M (2017). “A generalized Fellner-Schall method for smoothing parameter optimization with application to Tweedie location, scale and shape models.” *Biometrics*, **73**(4), 1071–1081.

Chapter 4

Statistical modeling of on-street parking lot occupancy in smart cities

Contributing Article Schneble, M., Kauermann, G. (2021). Statistical modeling of on-street parking lot occupancy in smart cities. *arXiv preprint arXiv:2106.06197*

Code and data <https://github.com/MarcSchneble/OnStreetParking>

Author Contributions The idea of modeling the occupancy of on-street parking lots with semi-Markov processes stems from Marc Schneble. He also proposed to compute the interval transition probabilities by employing the (inverse) Laplace transform. Moreover, Marc Schneble formulated the time to event model to estimate the transition intensities of the semi-Markov model. Göran kauermann was involved in developing the notation and the specific formulation of the models. The implementation of the model in **R** including data preparation and visualization of the results has been performed by Marc Schneble. The major part of the manuscript has been written by Marc Schneble. Both authors were involved in extensive proofreading of the paper.

Statistical modeling of on-street parking lot occupancy in smart cities

Marc Schneble

Department of Statistics, Ludwig-Maximilians-Universität Munich
and

Göran Kauermann

Department of Statistics, Ludwig-Maximilians-Universität Munich

June 14, 2021

Abstract

Many studies suggest that searching for parking is associated with significant direct and indirect costs. Therefore, it is appealing to reduce the time which car drivers spend on finding an available parking lot, especially in urban areas where the space for all road users is limited. The prediction of on-street parking lot occupancy can provide drivers a guidance where clear parking lots are likely to be found. This field of research has gained more and more attention in the last decade through the increasing availability of real-time parking lot occupancy data. In this paper, we pursue a statistical approach for the prediction of parking lot occupancy, where we make use of time to event models and semi-Markov process theory. The latter involves the employment of Laplace transformations as well as their inversion which is an ambitious numerical task. We apply our methodology to data from the City of Melbourne in Australia. Our main result is that the semi-Markov model outperforms a Markov model in terms of both true negative rate and true positive rate while this is essentially achieved by respecting the current duration which a parking lot already sojourns in its initial state.

Keywords: (Inverse) Laplace transform; predicting parking lot occupancy; semi-Markov process; time to event analysis

1 Introduction

Finding a clear parking lot in the center of a metropolitan region is usually very time-consuming and hence an expensive affair. A recent study that estimates the “economic and non-economic impact of parking pain” is Cookson and Pishue (2017). The study is mainly concerned with 10 major cities in each, the United States, the United Kingdom and Germany, respectively, and states that the average annual search time for parking ranges from 35 to 107 hours. This induces direct and indirect costs of up to 2,243 US-Dollars per year on the individual basis. The economic costs comprise the costs for fuel and opportunity costs of wasted time whereupon the non-economic costs are, among other things, related to higher stress levels. Other studies measured the share of traffic cruising for parking and quantified the average duration until a parking lot is found (e.g. Shoup, 2017 or Cao et al., 2017). In Hampshire and Shoup (2018), the authors compare the results of 22 of these studies. The share of traffic which cruises in order to find parking ranges from 8% to 74%, where the percentages depend heavily on the location and the time of the day. However, most of the studies suffer from a selection bias as they are oftentimes focused on regions where the demand for parking is generally very high. In any case, the search for parking enhances traffic congestion which itself causes an increasing number of accidents, air pollution, noise, etc. (Goodwin, 2004).

Car drivers could reduce all the costs and harms mentioned above if they would know ahead of time where the chance of finding a free parking lot is greatest. By the use of wireless sensor technologies (e.g. Lee et al., 2008), so called “smart cities” (Lin et al., 2017) are able to collect information regarding parking lot occupancy in real time. This information can be supplied to the public via smartphone apps or a direct gateway to the car. In general, one can either measure the number of free parking lots currently available in a predefined area, e.g. in the parking garage of a mall, or one can measure the occupancy of each single parking lot, e.g. for on-street parking. In this paper, we focus on the latter. A non-exclusive list of cities which already have implemented public accessible on-street parking sensors comprises San Francisco, California (Saharan et al., 2020), Santander, Spain (Cheng et al., 2015) and Melbourne, Australia (City of Melbourne, 2021). A summary of smart parking city projects around the world is provided by Lin et al. (2017).

For both off-street and on-street parking, various methodologies have been developed which aim at the prediction of parking lot occupancy by leveraging the data collected by smart cities. Real time parking data from San Francisco are employed by Rajabioun and Ioannou (2015) in order to predict the spatio-temporal pattern of parking availability via a multivariate autoregressive model. Neural networks for

parking availability prediction are used by Zheng et al. (2015) and Vlahogianni et al. (2016). The former paper defines a set of previous observations, the calendar time and the day of the week as input variables. Regression trees and support vector regression are used as comparative methods. The latter paper makes use of a simple neural network model for time series prediction with a specified number of lagged parking occupancy rates and with application to data from the city of Santander. An extension of a parking prediction model to an online parking guidance system was designed by Liu et al. (2018). Their model can also handle multiple users looking simultaneously for a free parking lot. The availability of parking is modeled via an autoregressive model and the recommended parking lot is a linear function of both driving cost and walking cost. Deep learning with recurrent neural networks are utilized by Camero et al. (2018) to predict occupancy rates of car parks in Birmingham. They compare their results in terms of mean absolute error with already existing prediction techniques on the same data set. Among others, a time series approach lead to higher prediction accuracy.

A more statistical approach to predict off-street parking occupancy was outlined first in Caliskan et al. (2007) and revisited by Klappenecker et al. (2014). Both papers model each car park as a queue which is described by a continuous time Markov process, i.e. the duration times in each state are assumed to be exponentially distributed. In particular, the transition matrix is dependent on two parameters, the arrival rate and the parking rate which are both assumed to be constant over time. Moreover, in both papers the model is evaluated only with simulated data which does not answer the question of whether the model is actually able to reliably predict parking lot availability. In a similar manner, Monteiro and Ioannou (2018) propose to model the arrival and the departure rate at on-street parking lots via non-homogeneous Poisson processes.

In this paper, we follow up the idea of modeling parking lot occupancy as a stochastic process. Since we concern ourselves with on-street parking only, we model each parking lot as a two-state stochastic process. As it turns out, semi-Markov processes (Pyke, 1961), which allow non-exponentially distributed duration times, are an appropriate class of stochastic processes for our study. These kinds of processes have wide-ranging applications, e.g. in production systems and maintenance systems where the time spend in an operational state is of interest (Limnios and Oprisan, 2012). In order to estimate the transition intensities of the semi-Markov process we employ time to event models which are essentially used in the epidemiological field (see e.g. Klein and Moeschberger, 2006 and Kalbfleisch and Prentice, 2011). We thereby respect the spatial dependence of nearby parking lots as well as further unobserved parking lot specific heterogeneity by including random effects in the



Figure 1: Location of on-street parking lots with and without in-ground sensors in the city center of Melbourne, Australia.

model.

The remainder of this paper is organized as follows. In Section 2 we visualize the data from the City of Melbourne which we use for our analyses and already provide some descriptive statistics. In Section 3 we introduce some notation and state the problem that we tackle in this paper. Sections 4 and 5 are concerned with the methodology involving semi-Markov processes and time to event analysis. The results when applying our methodology to the Melbourne parking data are presented in Section 6. Section 7 concludes the paper while also giving an outlook on potential extensions of our work.

2 Data

We apply the model which we develop in this paper to on-street parking lot data from the City of Melbourne, Australia. The data originate from the year 2019 and are provided through the open data platform City of Melbourne (2021). This database is filled by in-ground sensors which are installed underneath around 5,000 out of more than 20,000 on-street parking lots in the city center of Melbourne. Figure 1 shows the location of these parking lots and we see that most of the sensors are located

Start	End	Duration (minutes)	State	Marker	Side of street
2019-04-30 08:24:11	2019-04-30 08:29:31	5.33	1	1155W	west
2019-04-30 08:29:31	2019-04-30 08:34:54	5.38	0	1155W	west
2019-04-30 08:34:54	2019-04-30 08:37:22	2.47	1	1155W	west
⋮	⋮	⋮	⋮	⋮	⋮
2019-06-07 08:53:44	2019-06-07 08:54:27	0.72	0	C1170	central
2019-06-07 08:54:27	2019-06-07 08:55:21	0.90	1	C1170	central
2019-06-07 09:15:11	2019-06-07 09:20:40	5.48	1	C1170	central
2019-06-07 09:20:40	2019-06-07 09:55:30	34.83	0	C1170	central

Table 1: Structure of the preprocessed on-street parking lot data.

in the central business district (CBD) of Melbourne.¹ The basic structure of the already preprocessed data is exemplified in Table 1, where every row matches to a duration in either state 0 (clear) or 1 (occupied) that is specified to the second. The first three rows of Table 1 correspond to consecutive events on the same parking lot, which can be identified through a unique marker. However, the last four rows show that the data is not complete, i.e. there are time intervals in which the sensor was either disabled or it just malfunctioned. Therefore, we advocate that these data are missing completely at random (Heitjan and Basu, 1996), i.e. not including them in the analyses does not lead to biased estimates. In other words, the available data can be considered as a random sample of the complete data. Altogether, the dataset for the year 2019 consists of more than 30 million observations in both states 0 and 1.

Figure 2 shows parking lots in the CBD along with their average time being clear (state 0, left panel) or occupied (state 1, right panel), where we only considered parking events which started after 8 am and terminated before 8 pm on each day of the week. We see that parking lots around shopping malls in the center of the map tend to be unoccupied only for several minutes, whereas parking lots in the north-western part of the map extract are usually available for more than 30 minutes before they get occupied again. Here, the parking duration is usually more than one hour compared with usually less than 30 minutes in the center of the map extract. Note that the duration of parking is also affected by several parking restrictions which might differ even along the same street section.

Looking very precisely at the plots in Figure 2 it is already evident that the duration in both states 0 and 1 depends on the relative location of the parking lot on the street. In order to quantify this we show in Figure 3 Kaplan-Meier estimators

¹All maps in this paper are created using the R package Kahle and Wickham (2013).

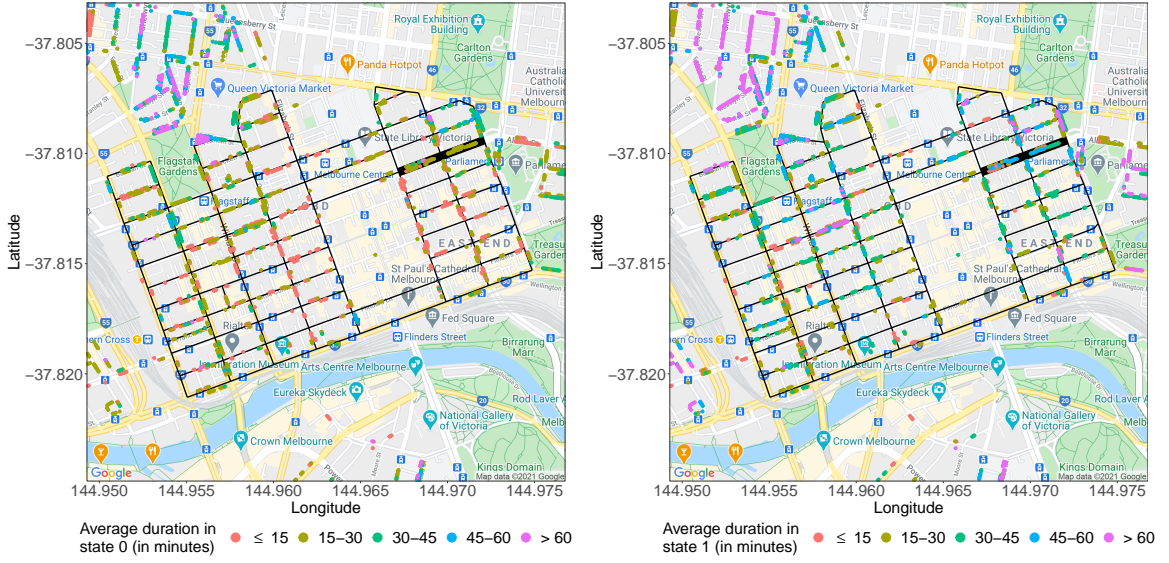


Figure 2: Average duration of parking events in the CBD of Melbourne.

(Klein and Moeschberger, 2006) of the duration in both states 0 and 1, restricted to one specific street segment which is highlighted as thicker lines in Figure 2 (Lonsdale Street between Russel Street and Exhibition Street). We find that parking lots which are located in the center of this street segment, i.e. in between the two lanes, exhibit the longest median time being occupied. Furthermore, especially the Kaplan-Meier plot for state 1 suggests the duration times being not exponentially distributed. This already hints at the necessity of employing a model which is capable of capturing also non-exponential duration times.

3 Problem and notation

We consider a set of on-street parking lots, indexed by $i = 1, \dots, N$, which are distributed along a network of streets, typically in the center of an urban area. A parking lot can be either clear or occupied which is why we model each parking lot as a continuous time two-state stochastic process $X^{(i)}$ with state space $\mathcal{S} = \{0, 1\}$. In particular, $X_t^{(i)} = 0$ if parking lot i is unoccupied at time point t and $X_t^{(i)} = 1$ otherwise. We assume knowledge of the process $X^{(i)}$ from a time point $t_p < 0$ in the past until the present moment $t = 0$, where we additionally observe K covariates $\mathbf{z}_t^{(i)} \in \mathbb{R}^K$ with $t \in [t_p, 0]$ and $i = 1, \dots, N$. Parking lots are located on a street network \mathcal{G} which is why we consider a distance measure taking the geometry of this

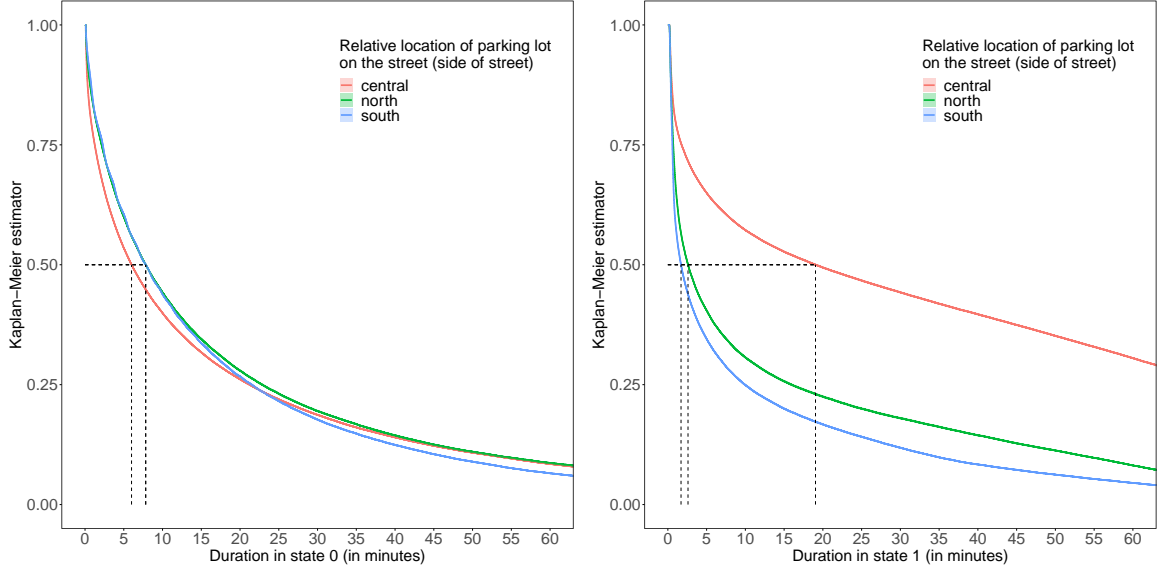


Figure 3: Kaplan Meier estimators for durations in Lonsdale Street between Russel Street and Exhibition Street.

network into account (Baddeley et al., 2015). Defining with $\mathbf{s}_i \in \mathbf{G}$ the location of the i -th parking lot we obtain with $d_{\mathbf{G}}(\mathbf{s}_{i_1}, \mathbf{s}_{i_2})$ the (street-)network-based distance between the two parking lots indexed by i_1 and i_2 , that is the driving distance between \mathbf{s}_{i_1} and \mathbf{s}_{i_2} with respect to \mathbf{G} . For simplicity we assume symmetry, i.e. one-way streets are ignored for now.

All available information from t_p up to the present moment $t = 0$ is formally contained in $\mathcal{F} = \sigma(X_t^{(i)}, \mathbf{z}_t^{(i)}; t \in [t_p, 0], i = 1, \dots, N)$, where \mathcal{F} denotes the observed history of the processes related to parking lot occupancy and covariate information including the present moment. Our goal is to predict the probability that a parking lot is unoccupied at a future time point $t_f > 0$. Since $X_0^{(i)}$ is included in the history \mathcal{F} , this probability is given by

$$\mathbb{P}(X_{t_f}^{(i)} = 0 \mid \mathcal{F}) = P_{00}^{(i)}(t_f) \cdot \mathbb{1}_{\{0\}}(X_0^{(i)}) + P_{10}^{(i)}(t_f) \cdot \mathbb{1}_{\{1\}}(X_0^{(i)}),$$

where

$$P_{jk}^{(i)}(t_f) = \mathbb{P}(X_{t_f}^{(i)} = k \mid \mathcal{F}, X_0^{(i)} = j), \quad j, k = 0, 1, \quad (1)$$

are transition probabilities to be determined. In order to predict (1), we first need to characterize the stochastic processes $X^{(i)}$ in more depth. Therefore, we denote with

$D_{j,t}^{(i)}$ the random duration time that parking lot i remains in state $j \in \mathcal{S}$ with index t indicating some time point. Furthermore, we define with

$$\lambda_{j,t}^{(i)}(d | \mathbf{z}_t^{(i)}) = \lim_{\Delta d \downarrow 0} \frac{\mathbb{P}(d \leq D_{j,t}^{(i)} < d + \Delta d | D_{j,t}^{(i)} \geq d, \mathbf{z}_t^{(i)})}{\Delta d} \quad (2)$$

the transition intensity from state j to state $1-j$ depending on the current duration d in state j and for time point t . Note that in the context of time to event analysis, (2) is usually denoted as the hazard rate and can thus be interpreted as the instantaneous rate at which a parking lot is changing its state. For fixed t we obtain the relation between the hazard function and the distribution function of the duration $D_{j,t}^{(i)}$ (see e.g. Kalbfleisch and Prentice, 2011)

$$\mathbb{P}(D_{j,t}^{(i)} \leq d) = 1 - \exp\left(-\int_0^d \lambda_{j,t}^{(i)}(x | \mathbf{z}_t^{(i)}) dx\right), \quad (3)$$

where the integral in (3) represents the cumulative hazard for duration d . It follows immediately that constant transition intensities imply exponentially distributed duration times D , i.e. the memorylessness property $\mathbb{P}(D > d + s | D > d) = \mathbb{P}(D > s)$ holds for $s, d > 0$. Allowing the transition intensities $\lambda(\cdot)$ to also depend on the duration time d in the current state leads to non-exponentially distributed duration times in general. We derive in the next section the transition probabilities (1) for the above defined process.

4 Prediction of transition probabilities

4.1 Transition probabilities in semi-Markov processes

Though our process has just two states, we provide a general description with multiple states here. We will see that this is advantageous as it will allow us to incorporate information about the duration from the most previous change of state before $t = 0$. More details are given in the next subsection. We define with $0 = t_{(0)} < t_{(1)} < t_{(2)} < \dots$ the time points of status changes of the stochastic process $X = (X_t)_{t \geq 0}$ whose finite state space is denoted as \mathcal{S} . Bear in mind that we consider the current time point as $t = 0$. Further, we denote with $X_{t_{(n)}} \in \mathcal{S}$ the state of the process at the time of the n -th transition in which X stays for the duration $D_{(n)}$. In other words, $D_{(n)}$ is the random length of the interval $[t_{(n)}, t_{(n+1)})$. We assume that X is a semi-Markov process which is fully characterized by an initial distribution $\mathbf{p} = [p_j(0) | j \in \mathcal{S}]$ with $\sum_{j \in \mathcal{S}} p_j(0) = 1$ and the renewal kernel

$\mathbf{Q}(d) = [\mathcal{Q}_{jk}(d) \mid j, k \in \mathcal{S}]$, where

$$\begin{aligned} \mathcal{Q}_{jk}(d) &= \mathbb{P}(X_{t_{(n+1)}} = k, D_{(n)} \leq d \mid X_{t_{(n)}} = j, D_{(n-1)}, X_{t_{(n-1)}}, \dots) \\ &= \mathbb{P}(X_{t_{(n+1)}} = k, D_{(n)} \leq d \mid X_{t_{(n)}} = j) \end{aligned} \quad (4)$$

for $d \geq 0$. Note that X has right-continuous sample paths and by definition we have $\mathcal{Q}_{jj}(\cdot) \equiv 0$ for $j \in \mathcal{S}$. The cumulative conditional distribution function of $D_{(n)}$ is given by $F_j(d) = \mathbb{P}(D_{(n)} \leq d \mid X_{t_{(n)}} = j) = \sum_{k \in \mathcal{S}} \mathcal{Q}_{jk}(d)$ for $j \in \mathcal{S}$ and $n \in \mathbb{N}_0$.

From (3) it follows that in case of non-constant transition intensities, $D_{(n)}$ satisfies the memorylessness property only in the instant $t_{(n)}$ of the transition into state $X_{t_{(n)}}$. Under consideration of this property it can be shown that for $t \geq 0$ the interval transition probabilities $P_{jk}(t) = \mathbb{P}(X_t = k \mid X_0 = j)$ are the solutions of the following integral equations

$$P_{jk}(t) = (1 - F_j(t))\delta_{jk} + \sum_{m \in \mathcal{S}} \int_0^t P_{mk}(t-x)q_{jm}(x) dx \quad (5)$$

with initial condition $P_{jk}(0) = \delta_{jk}$ (Kronecker delta) and $q_{jk}(\cdot)$ denotes the derivative of $\mathcal{Q}_{jk}(\cdot)$ with respect to the duration time. Note that if the subsequent state of each state $j \in \mathcal{S}$ is deterministic, say $k \in \mathcal{S}$, then $\mathcal{Q}_{jk}(\cdot)$ is a probability distribution function and with $f_j(\cdot) = q_{jk}(\cdot)$ we denote the corresponding density. This will simplify matters later when we are specifically concerned with the parking lot problem.

Following Grabski (2014), the systems of integral equations (5) can be solved via Laplace transforms (Widder, 2015). The Laplace transform $\tilde{f} := \mathcal{L}\{f\} : \mathbb{C} \supset C \rightarrow \mathbb{C}$ of a real valued function $f : [0, \infty) \rightarrow \mathbb{R}$ is given by

$$\tilde{f}(u) := \mathcal{L}\{f\}(u) = \int_0^\infty f(t)e^{-ut} dt, \quad (6)$$

where the integral in (6) converges for $u \in C$. The set C is called the region of convergence and consists of all $u \in \mathbb{C}$ which satisfy $\Re(u) > \gamma$, the so-called abscissa of convergence (Hall et al., 1992), and $\Re(u)$ denotes the real part of u . In particular, it holds that $\mathcal{L}\{1\}(u) = \frac{1}{u}$ and the Laplace transform of $\int_0^t P_{mk}(t-x)q_{jm}(x) dx$ is given by $\tilde{P}_{mk}(u)\tilde{q}_{jm}(u)$, i.e. convolution in the real time domain corresponds to multiplication in the complex frequency domain. Consequently, the linearity of the Laplace transform easily allows to represent the system of integral equations (5) as the following system of linear equations in the frequency domain

$$\tilde{P}_{jk}(u) = \left(\frac{1}{u} - \tilde{F}_j(u) \right) \delta_{jk} + \sum_{m \in \mathcal{S}} \tilde{q}_{jm}(u)\tilde{P}_{mk}(u). \quad (7)$$

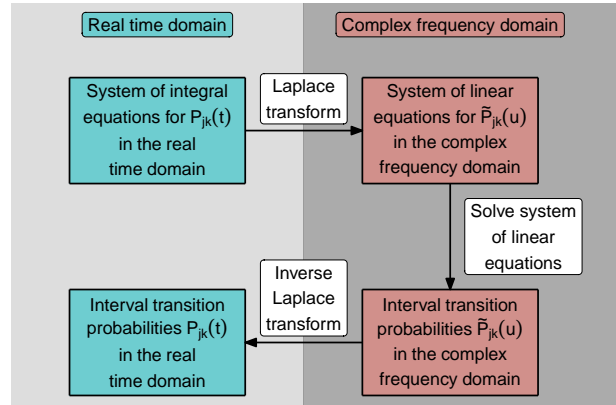


Figure 4: Visualization of the procedure of obtaining the interval transition probabilities $P_{jk}(t)$ in a semi-Markov process from the system of integral equations (5).

Defining with $\tilde{\mathbf{q}}(u) = [\tilde{q}_{jk}(u) \mid j, k \in \mathcal{S}]$ and $\tilde{\mathbf{F}}(u) = [\delta_{jk}\tilde{F}_j(u) \mid j, k \in \mathcal{S}]$ matrices of the same dimension as $\mathbf{Q}(u)$, we obtain the solution of (7) as

$$\tilde{\mathbf{P}}(u) = (\mathbf{I} - \tilde{\mathbf{q}}(u))^{-1} \left(\frac{1}{u} \mathbf{I} - \tilde{\mathbf{F}}(u) \right), \quad (8)$$

where \mathbf{I} denotes the identity matrix and $\tilde{\mathbf{P}}(u) = [\tilde{P}_{jk}(u) \mid j, k \in \mathcal{S}]$. The interval transition probabilities $P_{jk}(t)$ in the real time domain can then be calculated by applying the inverse Laplace transform element-wise to the solution (8) in the complex frequency domain. The inverse Laplace transform \mathcal{L}^{-1} of a Laplace-transformed function $\mathcal{L}\{f\} : \mathbb{C} \rightarrow \mathbb{C}$ is generally defined through the following Bromwich integral (Weideman and Trefethen, 2007)

$$f(t) = \mathcal{L}^{-1}\{\mathcal{L}\{f\}(u)\}(t) = \frac{1}{2\pi i} \lim_{T \rightarrow \infty} \int_{\gamma_0 - iT}^{\gamma_0 + iT} e^{ut} \mathcal{L}\{f\}(u) du,$$

where $\gamma_0 > \gamma$ from above, i.e. γ_0 must be in the region of convergence C of $\mathcal{L}\{f\}$. The whole procedure for determining the interval transition probabilities (1) which we formally described above is summarized in Figure 4.

4.2 Solving the parking lot problem

Suppose that we find ourselves at the current time point $t = 0$ and for simplicity of notation, we drop superscript (i) related to the parking lot index in this section. In

order to predict the transition probabilities (1) in the time span ranging from $t = 0$ until a future time point $t_f > 0$ we need $\lambda_{j,t}(d)$ for $t > 0$. Certainly, the future evolution of time dependent covariates is unknown at $t = 0$, i.e. $\lambda_{j,t}(d)$ might not be available for $t > 0$. However, since in our specific application to parking lot data the forecast horizon is usually in the range of several minutes up to an hour, we assume that $\lambda_{j,t}(d) = \lambda_{j,t=0}(d)$ which we denote in short with $\lambda_j(d) = \lambda_{j,t=0}(d)$ for $j = 0, 1$. Estimation of $\lambda_j(\cdot)$ is covered in the following Section 5.

We can now adopt the theoretical concepts outlined in Section 4.1 to our problem. First, the history \mathcal{F} contains the state of each parking lot at the present moment $t = 0$, i.e. the initial distribution \mathbf{p} is deterministic. However, formula (5) derived above is based on the assumption that the process X jumps in $t = 0$ into its initial state, i.e. $t_{(0)} = 0$. In other words, the duration in the initial state at $t = 0$ is zero. This is certainly not the case with regards to our parking lot problem since at $t = 0$ a parking lot has already been in state $X_0 = j$ for a known duration, say d_0 . This is shown in Figure 5 where, in view of the present moment $t = 0$, the parking lot has lastly changed its state at time point $t_{(0)} < 0$. Therefore, for the random duration time $D_{(0)}^* = t_{(1)}$ from being in state j at $t = 0$ until the first transition to state $1 - j$ it holds that

$$\mathbb{P}(D_{(0)}^* > d \mid X_0 = j) = \mathbb{P}(D_{(0)} > d + d_0 \mid D_{(0)} > d_0, X_0 = j) = \exp\left(-\int_{d_0}^{d_0+d} \lambda_j(x) dx\right). \quad (9)$$

Consequently, if $t_{(0)} < 0$ the distribution of the duration $D_{(0)}^*$ in the initial state differs from the distribution of the duration $D_{(0)} = t_{(1)} - t_{(0)}$. This is illustrated in Figure 5 where it holds for the duration times in the occupied state that $D_{(0)}^* \stackrel{d}{\neq} D_{(k)} = t_{(k+1)} - t_{(k)}$ for $k = 2, 4, 6, \dots$. In order to respect this finding in our model we add two initial states 0^* and 1^* to the updated state space $\mathcal{S}^* = \{0^*, 1^*, 0, 1\}$, where j^* is the state of X at $t = 0$.

Further, it holds that $F_{j^*}(\cdot) = \mathcal{Q}_{j^*, 1-j}(\cdot)$ and $F_j(\cdot) = \mathcal{Q}_{j, 1-j}(\cdot)$ for $j = 0, 1$ with

$$F_{j^*}(d) = 1 - \exp\left(-\int_{d_0}^{d_0+d} \lambda_j(x) dx\right) \quad \text{and} \quad F_j(d) = 1 - \exp\left(-\int_0^d \lambda_j(x) dx\right).$$

With $f_{j^*}(\cdot) = q_{j^*, 1-j}(\cdot)$ and $f_j(\cdot) = q_{j, 1-j}(\cdot)$ we denote the densities corresponding to the distribution functions F_{j^*} and F_j , respectively. We can now employ the general solution (8) in the setting with state space $\mathcal{S}^* = \{0^*, 1^*, 0, 1\}$ which yields a matrix $\tilde{\mathbf{P}}(u) = [\tilde{P}_{jk} \mid j, k \in \mathcal{S}^*] \in \mathbb{C}^{4 \times 4}$ of interval transition probabilities represented in the

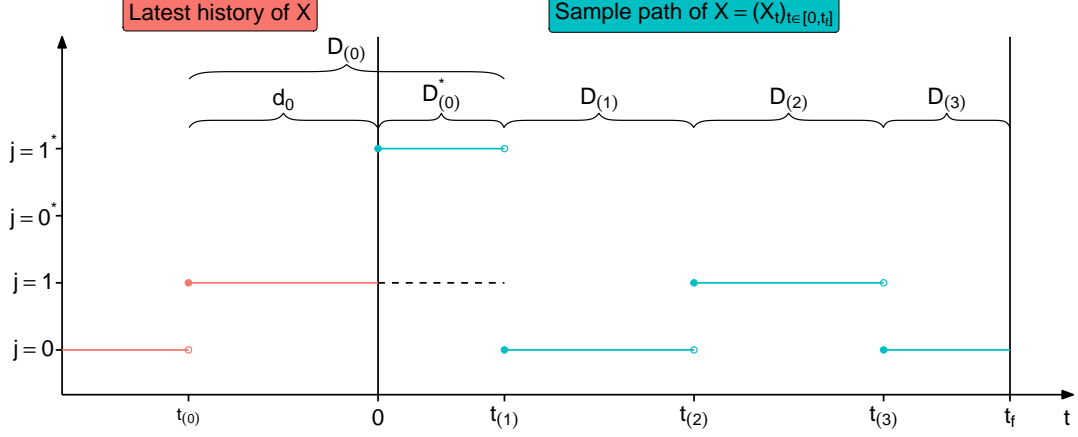


Figure 5: Visualization of the process $X = (X_t)_{t \in [0, t_f]}$ in view of the current time point $t = 0$.

complex frequency domain. The matrix $\mathbf{I} - \tilde{\mathbf{q}}(u)$ in (8) and its inverse are given by

$$\mathbf{I} - \tilde{\mathbf{q}}(u) = \begin{pmatrix} 1 & 0 & 0 & -\tilde{f}_{0^*}(u) \\ 0 & 1 & -\tilde{f}_{1^*}(u) & 0 \\ 0 & 0 & 1 & -\tilde{f}_0(u) \\ 0 & 0 & -\tilde{f}_1(u) & 1 \end{pmatrix}, \quad (\mathbf{I} - \tilde{\mathbf{q}}(u))^{-1} = \begin{pmatrix} 1 & 0 & \frac{\tilde{f}_{0^*}(u)\tilde{f}_1(u)}{1-\tilde{f}_0(u)\tilde{f}_1(u)} & \frac{\tilde{f}_{0^*}(u)}{1-\tilde{f}_0(u)\tilde{f}_1(u)} \\ 0 & 1 & \frac{\tilde{f}_{1^*}(u)}{1-\tilde{f}_0(u)\tilde{f}_1(u)} & \frac{\tilde{f}_{1^*}(u)\tilde{f}_0(u)}{1-\tilde{f}_0(u)\tilde{f}_1(u)} \\ 0 & 0 & 1 & \frac{\tilde{f}_0(u)}{1-\tilde{f}_0(u)\tilde{f}_1(u)} \\ 0 & 0 & \frac{\tilde{f}_1(u)}{1-\tilde{f}_0(u)\tilde{f}_1(u)} & 1 \end{pmatrix}$$

respectively. Now, using formula (8) and denoting $\tilde{f}(u) = (1 - \tilde{f}_0(u)\tilde{f}_1(u))^{-1}$ finally yields the solution of (7) as

$$\tilde{\mathbf{P}}(u) = \begin{matrix} & \mathbf{0}^* & \mathbf{1}^* & \mathbf{0} & \mathbf{1} \\ \mathbf{0}^* & \left(\frac{1}{u} - \tilde{F}_{0^*}(u) \right) & 0 & \tilde{f}(u)\tilde{f}_{0^*}(u)\tilde{f}_1(u) \left(\frac{1}{u} - \tilde{F}_0(u) \right) & \tilde{f}(u)\tilde{f}_{0^*}(u) \left(\frac{1}{u} - \tilde{F}_1(u) \right) \\ \mathbf{1}^* & 0 & \frac{1}{u} - \tilde{F}_{1^*}(u) & \tilde{f}(u)\tilde{f}_{1^*}(u) \left(\frac{1}{u} - \tilde{F}_0(u) \right) & \tilde{f}(u)\tilde{f}_{1^*}(u)\tilde{f}_0(u) \left(\frac{1}{u} - \tilde{F}_1(u) \right) \\ \mathbf{0} & 0 & 0 & \tilde{f}(u) \left(\frac{1}{u} - \tilde{F}_0(u) \right) & \tilde{f}(u)\tilde{f}_0(u) \left(\frac{1}{u} - \tilde{F}_1(u) \right) \\ \mathbf{1} & 0 & 0 & \tilde{f}(u)\tilde{f}_1(u) \left(\frac{1}{u} - \tilde{F}_0(u) \right) & \tilde{f}(u) \left(\frac{1}{u} - \tilde{F}_1(u) \right) \end{matrix} \quad (10)$$

The interval transition probabilities $P_{jk}(t)$ can be obtained by applying the inverse Laplace transform to the entries of $\tilde{\mathbf{P}}(u)$, i.e. $P_{jk}(t) = \mathcal{L}^{-1}\{\tilde{P}_{jk}(u)\}(t)$ for $j, k \in \mathcal{S}^*$ and $t \geq 0$. Since $\mathcal{L}^{-1}\{1/u\}(t) = \mathbb{1}_{[0, \infty)}(t)$, it particularly holds that $P_{0^*0^*}(t) = 1 - F_{0^*}(t)$ and $P_{1^*1^*}(t) = 1 - F_{1^*}(t)$ for $t \geq 0$. This is an expected result since $P_{j^*j^*}(t)$ is the probability that a parking lot's state will remain unchanged from time

point $t = 0$ until time $t > 0$ and $1 - F_{j^*}(t) = \mathbb{P}(D_{(0)^*}^* > t \mid X_0 = j)$ represents the probability that the duration in the initial state j will exceed time point t . For the remaining entries of the transition probability matrix $\mathbf{P}(t)$ we have to build on numerical inversion techniques here, where we make use of the approach proposed by Valsa and Brančik (1998). Details about the algorithm can be found in Appendix B. In the end, the probability that a parking lot is clear at a prospective time point $t_f > 0$, conditional on the history \mathcal{F} of parking lot occupation and covariate information, is given by

$$\mathbb{P}(X_{t_f} = 0 \mid \mathcal{F}) = \begin{cases} 1 - F_{0^*}(t) + \mathcal{L}^{-1}\{\tilde{P}_{0^*0}(u)\}(t_f), & X_0 = 0 \\ \mathcal{L}^{-1}\{\tilde{P}_{1^*0}(u)\}(t_f), & X_0 = 1 \end{cases}. \quad (11)$$

4.3 Exponentially distributed duration times

Using the techniques from above, explicit formulae can be derived for the transition probabilities (1) if the durations $D_{(n)}$ are exponentially distributed, i.e. if $X = (X_t)_{t \in [0, t_f]}$ is in fact a Markov process. With the transition intensities λ_j being constant, it holds that $F_{j^*}(d) = F_j(d) = 1 - e^{-\lambda_j d}$ and $f_j(d) = \lambda_j e^{-\lambda_j d}$. Therefore, we can omit the states 0^* and 1^* and for the interval transition probabilities it follows that

$$\mathbf{P}(t) = \frac{1}{\lambda_0 + \lambda_1} \begin{pmatrix} \lambda_1 + \lambda_0 e^{-t(\lambda_0 + \lambda_1)} & \lambda_0 [1 - e^{-t(\lambda_0 + \lambda_1)}] \\ \lambda_1 [1 - e^{-t(\lambda_0 + \lambda_1)}] & \lambda_0 + \lambda_1 e^{-t(\lambda_0 + \lambda_1)} \end{pmatrix}. \quad (12)$$

Usually, the result (12) is derived by solving the Kolomogorov forward differential equations as e.g. in Ross et al. (1996), Example 5.4(A).

5 Estimation of transition intensities

5.1 Modeling transition intensities

We now discuss the estimation of transition intensities $\lambda_{j,t}^{(i)}(d \mid \mathbf{z}_t^{(i)})$ as defined in (2) which can be interpreted as hazard rates in a time to event model. We consider covariates $\mathbf{z}_t^{(i)} = (z_{1,t}^{(i)}, \dots, z_{K,t}^{(i)})^\top$ through a model of the form

$$\lambda_{j,t}^{(i)}(d \mid \mathbf{z}_t^{(i)}) = \lambda_{j,0}(d) \exp \left(\beta_{0,j} + \sum_{k=1}^K g_{j,k}(z_{k,t}^{(i)}) + u_j^{(i)} \right). \quad (13)$$

Here, $\lambda_{j,0}(\cdot)$ is the common baseline intensity for the transition from state j to state $1 - j$. Furthermore, $\eta_{j,t}^{(i)} = \beta_{0,j} + \sum_{k=1}^K g_{j,k}(z_{k,t}^{(i)})$ is the linear predictor of the model (including the intercept $\beta_{0,j}$) which will be treated in more depth later. Coefficients $u_j^{(i)}$ are random effects, which account for unobserved parking lot specific heterogeneity. Hereby, we assume that $v_j^{(i)} = \log u_j^{(i)}$ follow independently a $\text{Gamma}(\frac{1}{\gamma_j}, \frac{1}{\gamma_j})$ prior distribution with $\mathbb{E}(v_j^{(i)}) = 1$ and $\text{Var}(v_j^{(i)}) = \gamma_j$. In the context of time to event analysis, these kinds of models are known as (gamma) shared frailty models (Therneau et al., 2003), since a common multiplicative frailty $v_j^{(i)}$ on the baseline hazard is shared among observations for parking lot i in state j .

5.2 Choosing an appropriate baseline intensity

Under consideration of (3) and (13), the distribution function $F_j^{(i)}(\cdot)$ of the random duration $D_j^{(i)}$ that parking lot i stays in state j is given by

$$F_j^{(i)}(d) = \mathbb{P}(D_j^{(i)} \leq d) = 1 - \exp\left(-\exp(\eta_{j,t=0}^{(i)} + u_j^{(i)}) \int_0^d \lambda_{j,0}(x) dx\right) \quad (14)$$

for $j = 0, 1$. For $j = 0^*, 1^*$ the integral boundaries in (14) need to be shifted by the current duration d_0 as motivated above. Employing the numerical algorithm proposed by Valsa and Brančik (1998) in order to compute the inverse Laplace transformation of (10) involves repeatedly evaluation of both $F_j^{(i)}(\cdot)$ and its derivative $f_j^{(i)}(\cdot)$. Considering (14), it is evident that the form of the baseline intensity $\lambda_{j,0}(\cdot)$ determines the numerical effort therefore, which is why an easy-to-integrate $\lambda_{j,0}(\cdot)$ is preferred. Often, the baseline intensity is not explicitly modeled as in the Cox-Model. Here, the (cumulative) baseline intensity can be obtained via the Breslow estimator (Lin, 2007). Alternatively, the baseline can be modeled semiparametrically, e.g. with piece-wise exponential additive mixed models (PAMMs, Bender et al., 2018). However, with both approaches the cumulative hazard can only be evaluated numerically which has the consequence that the inversion of the Laplace transform suffers from numerical instability. The employment of a fully-parametric model for $\lambda_{j,0}(\cdot)$ allows to circumvent numerical integration if the integral of $\lambda_{j,0}(\cdot)$ has an explicit representation. This is pursued in the following.

A frequently used parametric time to event model is the Weibull model, in which case (13) for $t = 0$ has the form $\lambda_j^{(i)}(d) = b_j^{(i)} \alpha_j d^{\alpha_j - 1}$, where $\alpha_j d^{\alpha_j - 1} = \lambda_{j,0}(d; \alpha_j)$ is the common baseline intensity and $b_j^{(i)} = \exp(\eta_{j,t=0}^{(i)} + u_j^{(i)})$. This allows to express

both the distribution function $F_j^{(i)}(d) = 1 - \exp(-b_j^{(i)}d^{\alpha_j})$ and density $f_j^{(i)}(d) = b_j^{(i)}\alpha_j d^{\alpha_j-1} \exp(-b_j^{(i)}d^{\alpha_j})$ for $d \geq 0$ explicitly. Finally, note that the baseline intensity $\lambda_{j,0}(\cdot)$ is shaped by a single parameter α_j , with $\alpha_j < 1$ ($\alpha_j > 1$) resulting in strictly decreasing (increasing) transition intensities.

5.3 The linear predictor

The linear predictor $\eta_{j,t}^{(i)} = \beta_{0,j} + \sum_{k=1}^K g_{k,j}(z_{k,t}^{(i)})$ is independent of the duration time d and multiplicatively takes the covariate effects in the time to event model (13) into account. We either include the k -th covariate linearly in the model, i.e. $g_{k,j}(z_{k,t}^{(i)}) = \beta_{k,j}z_{k,t}^{(i)}$ or through nonlinear modeling achieved by applying B-splines. In the latter case, the k -th covariate has a B-Spline basis representation $g_{k,j}(z_{k,t}^{(i)}) = \sum_{m=1}^{M_k} \beta_{k,j,m} B_{k,j,m}^l(z_{k,t}^{(i)})$ of order $l \in \mathbb{N}$ with parameter vector $\boldsymbol{\beta}_{k,j} = (\beta_{k,j,1}, \dots, \beta_{k,j,M_k})^\top$ (see Ruppert et al., 2003 or Fahrmeir et al., 2007). For reasons of identifiability of smooth effects, these functions are centered around zero as proposed in Wood (2017). We collect all regression parameters for state j in a single vector which we denote with $\boldsymbol{\theta}_j$. Estimation of the model parameters is shown in Appendix A.

6 Results

6.1 Fitting intensities

We first show an exemplary fit of the time to event model from Section 5 to the Melbourne parking data which we introduced in Section 2. The linear predictor for state $j = 0, 1$ is specified as

$$\eta_{j,t}^{(i)} = \beta_{0,j} + \beta_{1,j} \cdot \text{weekday}_t + \beta_{2,j} \cdot \text{sideofstreet}^{(i)} + \beta_{3,j} \cdot \text{nearby}_{1-j,t}^{(i)} + g_{4,j}(\text{hour}_t), \quad (15)$$

where time point t corresponds to the start time of each observed, possibly censored, duration. Temporarily, we here restrict the data to all days of June 2019 between 8 am in the morning and 8 pm in the evening while on the spatial scale we only include parking lots which are located in the eastern section of Lonsdale Street, which is shown as a thicker network segment in Figure 2. The covariate $\text{nearby}_{j,t}^{(i)}$ is defined as

$$\text{nearby}_{j,t}^{(i)} = \sum_{k \neq i} \mathbb{1}\{X_t^{(i)} = j, d_G(\mathbf{s}_i, \mathbf{s}_k) \leq h\} / \sum_{k \neq i} \mathbb{1}\{d_G(\mathbf{s}_i, \mathbf{s}_k) \leq h\}$$

	Model state 0		Model state 1	
	Effect (s.e.)	Relative risk	Effect (s.e.)	Relative Risk
Intercept	-2.721 (0.177)	0.066	-1.751 (0.146)	0.174
Tuesday	0.050 (0.012)	1.052	0.037 (0.012)	1.037
Wednesday	0.091 (0.012)	1.095	0.025 (0.012)	1.025
Thursday	0.124 (0.012)	1.132	0.005 (0.012)	1.005
Friday	0.098 (0.012)	1.103	0.024 (0.012)	1.024
Saturday	-0.139 (0.013)	0.870	-0.032 (0.013)	0.968
Sunday	0.126 (0.014)	1.134	-0.371 (0.016)	0.690
central	0.159 (0.199)	1.172	-0.764 (0.164)	0.466
south	0.129 (0.265)	1.138	0.218 (0.218)	1.244
nearby _{1-j}	1.451 (0.022)	4.266	1.078 (0.023)	2.940

Table 2: Estimates $\widehat{\beta}_{k,j}$ of fixed linear covariate effects in the time to event model with linear predictor as specified in (15), standard errors in brackets. The relative risk is given as $\exp(\widehat{\beta}_{k,j})$.

which is the fraction of all parking lots in state j with driving distance less than h from parking lot i at time point t . For all our analyses we set $h = 50$ meters. The smooth functions $g_{4,j}(\text{hour}_t)$ which model the effect of the time of the day are build with quadratic B-splines with 10 degrees of freedom.

We fit the time to event model which we specified above employing the `survival` package for the statistical software **R** (R Core Team, 2013). For the estimated shape parameter α_j of the Weibull baseline intensity it holds that $\widehat{\alpha}_1 = 0.55 < \widehat{\alpha}_0 = 0.65 \ll 1$ which mimics strictly decreasing transition intensities. This suits the Kaplan-Meier estimators shown in Figure 3 and legitimates the use of a Weibull baseline hazard instead of an exponential baseline hazard. Estimates of linear covariate effects and their standard errors are shown in Table 2. The proportional hazards assumption allows to quantify the relative risk of covariate effects. The effect of the weekly variation is fairly weak except for Sunday, where we see a significant negative effect in the model for the transition from state 1 to state 0. Therefore, the average duration of parking on Sundays is longer which might be caused by more relaxed parking restrictions on Sundays. The effect of the relative location of a parking lot mirrors the conclusion which we drew from Figure 3, i.e. there is no significant effect in the model for state 0 and the parking duration in between two lanes is expected

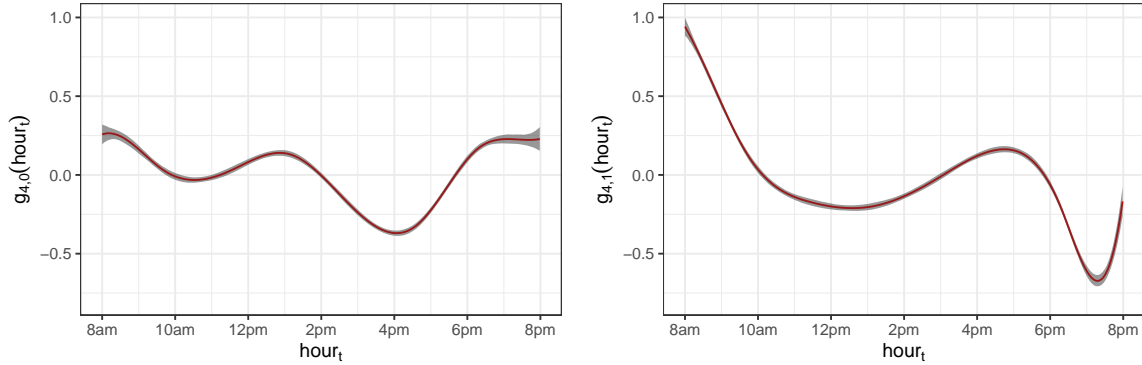


Figure 6: Smooth effect of the time of the day in state 0 (left panel) and state 1 (right panel), 95% confidence bands are shown in grey.

to be significantly higher than at the curbside. The significant positive effect of the covariate nearby_1 (nearby_0) in the model for state 0 (1) means that the duration a parking lot is clear (occupied) decreases with increasing occupancy (availability) of nearby parking lots. Finally, we show in Figure 6 the smooth effect of daytime. We see that if a parking lot is cleared in the afternoon the expected duration in state 0 is least. Overall, the effect of the time of the day for state 0 is much weaker when compared to state 1 where the parking duration tendentially rises during the course of the day, i.e. short-term parking occurs mainly in the morning.

6.2 Predicting parking lot occupancy

As we can see, the results of the time to event model are valuable in their own right. However, the main purpose of those was to use them as plug-in estimates for a model in order to predict $\mathbb{P}(X_{t_f}^{(i)} = 0 \mid \mathcal{F})$. Therefore, we next assess the performance of the following three prediction models which were generally treated in Section 4. For the first two models we fit a time to event model with Weibull baseline hazard and linear predictor as specified in (15). The fitted intensities are used in order to estimate the distribution function as derived in (14) and the probabilities of interest $\mathbb{P}(X_{t_f}^{(i)} = 0 \mid \mathcal{F})$ are computed according to (11) where, however, in the first case d_0 corresponds to the actually observed value and in the second case d_0 will be generally set to zero. Thus, the first model is a semi-Markov model with state space \mathcal{S}^* of cardinality four while the second model essentially reduces to a semi-Markov model with two states, i.e. state space \mathcal{S} . For the third model, we fit the time to event model with auxiliary condition $\alpha_j = 1$ which leads to exponentially

distributed duration times. Consequently, the predictions can be made according to the closed-form solution (12). We refer to this model as the two-state Markov model.

In order to evaluate the performance of the different models under a preferably realistic scenario we consider the following setting. We randomly choose a time point $t = 0$ as well as a location \mathbf{s} on the street network \mathbf{G} (see Figure 2) where we put a higher sampling weight on areas with more parking lots, see Schneble and Kauermann (2020). Now, we predict the availability of parking lots being clear for all parking lots i which satisfy $d_{\mathbf{G}}(\mathbf{s}, \mathbf{s}_i) \leq 250$ meters. The time point $t = 0$ is chosen to be either between 10 am and 12 pm or between 4 pm and 6 pm of each day in June 2019. The prediction horizon shall be $t_f = 10$ minutes or $t_f = 30$ minutes, respectively. In each case, the transition intensities are determined by making use of data restricted to 30 days in the past from the perspective of $t = 0$.

We repeat each scenario described above $R = 100$ times and measure the prediction performance by making use of receiver operating characteristic (ROC) curves (Robin et al., 2011). Thereby, we first specify a set $\{c_p\}$ of $P + 2$ thresholds with $-\infty = c_0 < 0 < c_1 < \dots < c_P < 1 < \infty = c_{P+1}$. Then, we determine for each c_p the binary estimate

$$\widehat{X}_t^{(i)}(c_p) = \begin{cases} 0, & \mathbb{P}(X_{t_f}^{(i)} = 0 \mid \mathcal{F}) \geq c_p \\ 1, & \mathbb{P}(X_{t_f}^{(i)} = 0 \mid \mathcal{F}) < c_p \end{cases}.$$

Next, we compute the specificity $\text{TNR}(c_p)$ (true negative rate) and the sensitivity $\text{TPR}(c_p)$ (true positive rate) in dependence of the threshold c_p as

$$\text{TNR}(c_p) = \frac{\#\{X_{t_f}^{(i)} = 1 \text{ and } \widehat{X}_{t_f}^{(i)}(c_p) = 1\}}{\#\{X_{t_f}^{(i)} = 1\}}, \quad \text{TPR}(c_p) = \frac{\#\{X_{t_f}^{(i)} = 0 \text{ and } \widehat{X}_{t_f}^{(i)}(c_p) = 0\}}{\#\{X_{t_f}^{(i)} = 0\}}.$$

Note that in contrast to the habitual convention, “0” refers to the positive class and “1” refers to the negative class. Finally, a ROC curve is a function of the sensitivity in dependence of $1 - \text{specificity}$. An index which measures the overall prediction performance of a binary predictor is the area under the (ROC) curve (AUC), which is the integral of the ROC curve. An AUC equal to one corresponds to a perfect predictor where the random predictors AUC, highlighted through a diagonal line in Figure 7, is always equal to 0.5.

In Figure 7 we show ROC curves of the predictions related to each of the four possible scenarios described above, i.e. with prediction horizon equal to $t_f = 10$ minutes (upper panels) or $t_f = 30$ minutes (lower panels) and setting the present moment to $t = 0$ in the morning (left panels) or in the late afternoon (right panels).

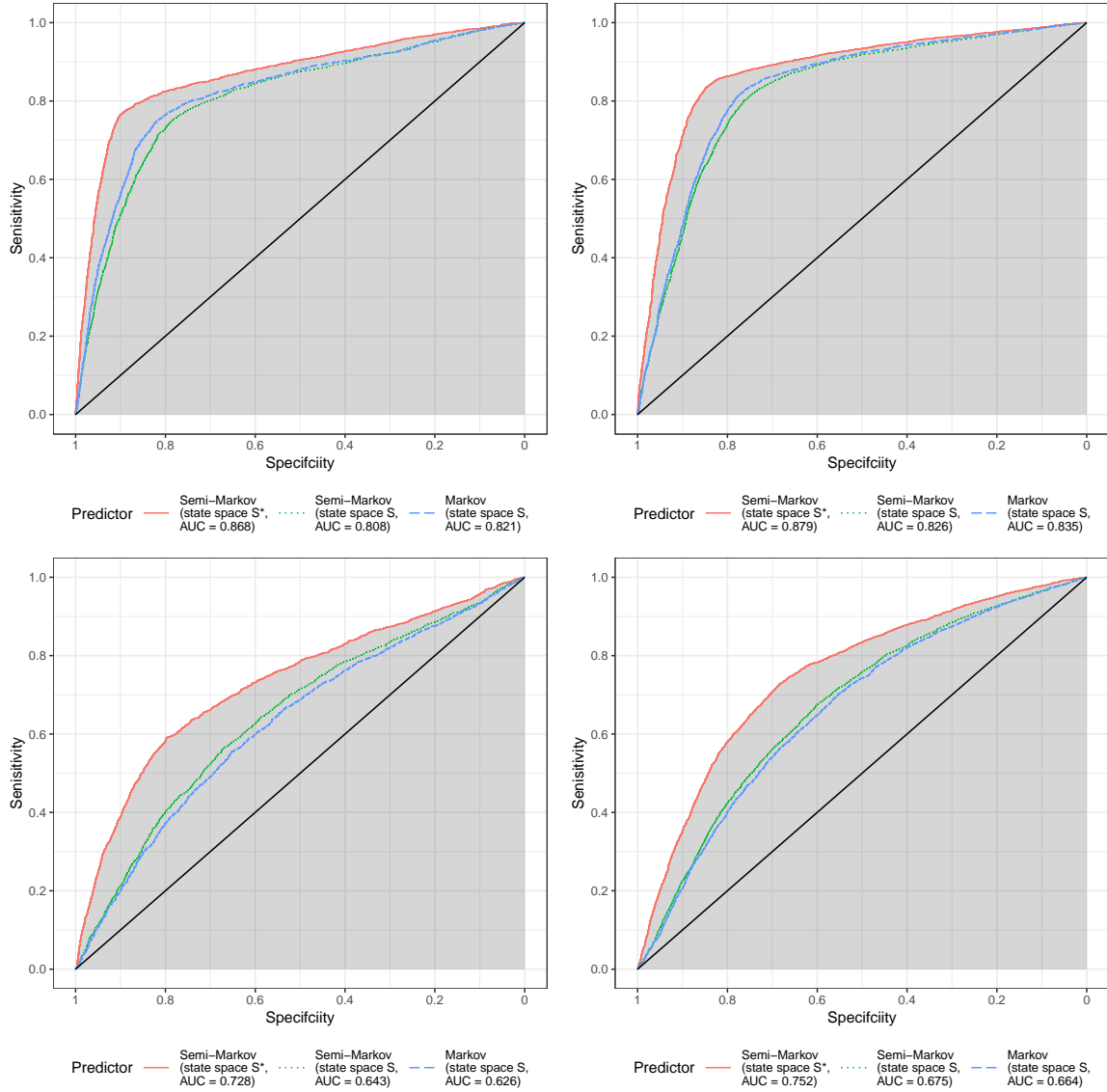


Figure 7: ROC curves showing the performance of three predictors when daily predicting free parking lots in June 2019. In the left (right) panels, $t = 0$ ranges between 10 am and 12 pm (4 pm and 6 pm). In the top (bottom) panels, time point t_f is 10 minutes (30 minutes) after $t = 0$.

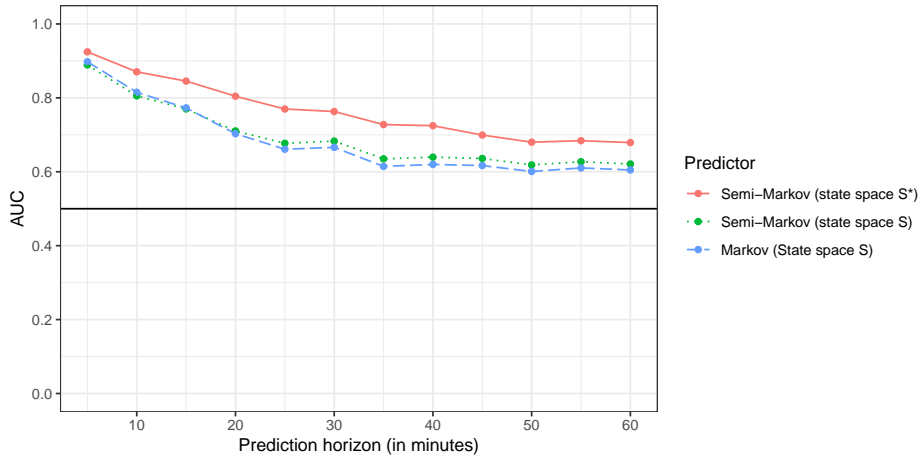


Figure 8: AUC of three binary predictors depending on the prediction horizon.

We can generally observe that for a fixed specificity, the semi-Markov predictor with state space \mathcal{S}^* outperforms the (semi-)Markov predictor with state space \mathcal{S} in terms of sensitivity, where the same holds vice versa. Accordingly, the AUC of the semi-Markov predictor is larger when compared to the AUC of the Markov predictor. This difference is minor if the prediction horizon is very short-term (10 minutes). If the prediction horizon amounts to 30 minutes, the performance of both models worsens distinctly. However, on a relative scale the prediction accuracy of the (semi-)Markov model with two states diminishes stronger when compared to the semi-Markov model with four states. Summarizing, we conclude that the benefit of our methodology mainly comes from the adding of the additional states 0^* and 1^* to the state space \mathcal{S} . However, when opposing the two predictors which make use of a two-state stochastic process, it is not apparent that one outperforms the other.

Finally, we show the AUC for the same predictors as considered in Figure 7 when the prediction horizon ranges between five minutes and one hour. Since a large effect of the time of the day on the prediction accuracy can not be observed we draw $t = 0$ from the interval between 4 pm and 6 pm. Figure 7 confirms what we have seen before, the semi-Markov predictor with the extended state space \mathcal{S}^* performs superior in terms of AUC when compared to the (semi-)Markov predictor with two states, i.e. when the current duration d_0 is not respected in the model. This is most apparent when the prediction horizon is between 20 and 40 minutes. When opposing the semi-Markov predictor with two states to the Markov predictor, we observe only few discrepancy between the two predictors for predictors horizons 20 minutes and less. If the prediction horizon is longer the semi-Markov model can

be favored as the AUC is marginally larger. Summing up, the semi-Markov model distinctly outperforms the Markov model in terms of AUC. Again, this is largely due to the involvement of the two additional states.

7 Discussion

In this paper we have presented a general framework which can be used to predict the individual short-term probability of on-street parking lots being unoccupied. A time to event model is employed in order to estimate the transition intensities and consequently the distribution of the duration in each state. Besides the usage as plug-in estimates for the (semi-)Markov prediction model, the results of the time to event model already provide valuable insight into the patterns of on-street parking dynamics in the City of Melbourne. On the other hand, the semi-Markov model is solely designed as a prediction model and we have seen that the prediction accuracy, measured in terms of AUC of a ROC curve, is distinctly larger when compared to a model which is restricting duration times to be exponentially distributed. If the response in a binary prediction model is greatly unbalanced, i.e. most of the data belong either to the positive or negative class, Saito and Rehmsmeier (2015) propose to employ precision-recall curves in order to evaluating a binary predictor. However, we do not see such a strong imbalance in our data such that we consider these results reliable.

The performance of the prediction model is highly dependent on the quality of the data. Usually, not all sensors in a chosen area are active at the same time and not all parking lots are equipped with a sensor. If a sensor does not send the current state of the related parking lot, it is much harder to predict its short-term availability since the initial distribution is not known and particularly not deterministic. Consequently, the initial distribution \mathbf{p} of the parking lot occupancy would have to be estimated as well and then $\mathbf{P}(t_f)\mathbf{p}$ yields a vector of transition probabilities from time $t = 0$ to time $t = t_f$, where $\mathbf{P}(t_f)$ can be obtained by applying the inverse Laplace transform to (10).

The time to event model which we employ to estimate the distribution functions of the semi-Markov model is rather unsophisticated. However, this has the advantage that standard software can be used to fit the model. The performance of the prediction model could certainly be further improved by incorporating covariates into the time to event model in a (duration) time varying manner. Moreover, the usage of further external covariates such as weather conditions might increase the prediction performance as well. However, our main intention with this paper was to show that a semi-Markov model clearly outperforms a Markov model in terms of prediction

accuracy. Since we use the same estimates for both kinds of prediction models, this finding should be rather independent from the goodness of the chosen time to event model.

A natural extension of our model is its integration into an individual parking guidance system. Here, a car driver might choose a destination \mathbf{s} in the center of an urban area and a navigation system computes the estimated time of arrival t_f until reaching the target \mathbf{s} . Our model is able to yield predictions of on-street parking availability at time point t_f in the surrounding area of \mathbf{s} and could guide the driver into a street section in which the likelihood of finding a clear parking lot is greatest. We consider this computational considerable optimization problem to be beyond the scope of this paper where we focused on the statistical point of view of our modeling approach.

References

- Baddeley, A., E. Rubak, and R. Turner (2015). *Spatial point patterns: Methodology and applications with R*. CRC press.
- Bender, A., A. Groll, and F. Scheipl (2018). A generalized additive model approach to time-to-event analysis. *Statistical Modelling* 18(3-4), 299–321.
- Caliskan, M., A. Barthels, B. Scheuermann, and M. Mauve (2007). Predicting parking lot occupancy in vehicular ad hoc networks. In *2007 IEEE 65th Vehicular Technology Conference-VTC2007-Spring*, pp. 277–281. IEEE.
- Camero, A., J. Toutouh, D. H. Stolfi, and E. Alba (2018). Evolutionary deep learning for car park occupancy prediction in smart cities. In *International Conference on Learning and Intelligent Optimization*, pp. 386–401. Springer.
- Cao, J., M. Menendez, and R. Waraich (2017). Impacts of the urban parking system on cruising traffic and policy development: the case of Zurich downtown area, Switzerland. *Transportation* 46(3), 883–908.
- Cheng, B., S. Longo, F. Cirillo, M. Bauer, and E. Kovacs (2015). Building a big data platform for smart cities: Experience and lessons from Santander. In *2015 IEEE International Congress on Big Data*, pp. 592–599. IEEE.
- City of Melbourne (2021). City of Melbourne open data. <https://data.melbourne.vic.gov.au>. Accessed: June 14, 2021.

- Cookson, G. and B. Pishue (2017, July). The impact of parking pain in the US, UK and Germany. INRIX Research. <https://www2.inrix.com/research-parking-2017>.
- Duchateau, L. and P. Janssen (2007). *The frailty model*. Springer Science & Business Media.
- Fahrmeir, L., T. Kneib, S. Lang, and B. Marx (2007). *Regression*. Springer.
- Goodwin, P. (2004). The economic costs of road traffic congestion. Technical report, UCL (University College London), The Rail Freight Group.
- Grabski, F. (2014). *Semi-Markov processes: Applications in system reliability and maintenance*. Elsevier.
- Hall, P., J. L. Teugels, and A. Vanmarcke (1992). The abscissa of convergence of the laplace transform. *Journal of applied probability*, 353–362.
- Hampshire, R. C. and D. Shoup (2018). What share of traffic is cruising for parking? *Journal of Transport Economics and Policy (JTEP)* 52(3), 184–201.
- Heitjan, D. F. and S. Basu (1996). Distinguishing “missing at random” and “missing completely at random”. *The American Statistician* 50(3), 207–213.
- Kahle, D. and H. Wickham (2013). ggmap: Spatial visualization with ggplot2. *The R Journal* 5(1), 144–161.
- Kalbfleisch, J. D. and R. L. Prentice (2011). *The statistical analysis of failure time data*, Volume 360. John Wiley & Sons.
- Klappenecker, A., H. Lee, and J. L. Welch (2014). Finding available parking spaces made easy. *Ad Hoc Networks* 12, 243–249.
- Klein, J. P. (1992). Semiparametric estimation of random effects using the Cox model based on the EM algorithm. *Biometrics*, 795–806.
- Klein, J. P. and M. L. Moeschberger (2006). *Survival analysis: Techniques for censored and truncated data*. Springer Science & Business Media.
- Lee, S., D. Yoon, and A. Ghosh (2008). Intelligent parking lot application using wireless sensor networks. In *CTS*, pp. 48–57.

- Limnios, N. and G. Oprisan (2012). *Semi-Markov processes and reliability*. Springer Science & Business Media.
- Lin, D. (2007). On the Breslow estimator. *Lifetime data analysis* 13(4), 471–480.
- Lin, T., H. Rivano, and F. Le Mouél (2017). A survey of smart parking solutions. *IEEE Transactions on Intelligent Transportation Systems* 18(12), 3229–3253.
- Liu, K. S., J. Gao, X. Wu, and S. Lin (2018). On-street parking guidance with real-time sensing data for smart cities. In *2018 15th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON)*, pp. 1–9. IEEE.
- Monteiro, F. V. and P. Ioannou (2018). On-street parking prediction using real-time data. In *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, pp. 2478–2483. IEEE.
- Nielsen, G. G., R. D. Gill, P. K. Andersen, and T. I. Sørensen (1992). A counting process approach to maximum likelihood estimation in frailty models. *Scandinavian journal of Statistics*, 25–43.
- Pyke, R. (1961). Markov renewal processes: Definitions and preliminary properties. *The Annals of Mathematical Statistics*, 1231–1242.
- R Core Team (2013). *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing.
- Rajabioun, T. and P. A. Ioannou (2015). On-street and off-street parking availability prediction using multivariate spatiotemporal models. *IEEE Transactions on Intelligent Transportation Systems* 16(5), 2913–2924.
- Robin, X., N. Turck, A. Hainard, N. Tiberti, F. Lisacek, J.-C. Sanchez, and M. Müller (2011). proc: an open-source package for R and S+ to analyze and compare ROC curves. *BMC bioinformatics* 12(1), 1–8.
- Ross, S. M., J. J. Kelly, R. J. Sullivan, W. J. Perry, D. Mercer, R. M. Davis, T. D. Washburn, E. V. Sager, J. B. Boyce, and V. L. Bristow (1996). *Stochastic processes*, Volume 2. Wiley New York.
- Ruppert, D., M. P. Wand, and R. J. Carroll (2003). *Semiparametric regression*. Number 12. Cambridge University Press.

- Saharan, S., N. Kumar, and S. Bawa (2020). An efficient smart parking pricing system for smart city environment: A machine-learning based approach. *Future Generation Computer Systems* 106, 622–640.
- Saito, T. and M. Rehmsmeier (2015). The precision-recall plot is more informative than the roc plot when evaluating binary classifiers on imbalanced datasets. *PloS one* 10(3), e0118432.
- Schneble, M. and G. Kauermann (2020). Intensity estimation on geometric networks with penalized splines. *arXiv preprint arXiv:2002.10270*.
- Shoup, D. (2017). *The high cost of free parking: Updated edition*. Routledge.
- Therneau, T. M., P. M. Grambsch, and V. S. Pankratz (2003). Penalized survival models and frailty. *Journal of Computational and Graphical Statistics* 12(1), 156–175.
- Valsa, J. and L. Brančik (1998). Approximate formulae for numerical inversion of laplace transforms. *International Journal of Numerical Modelling: Electronic Networks, Devices and Fields* 11(3), 153–166.
- Vlahogianni, E. I., K. Kepaptsoglou, V. Tsetsos, and M. G. Karlaftis (2016). A real-time parking prediction system for smart cities. *Journal of Intelligent Transportation Systems* 20(2), 192–204.
- Weideman, J. and L. Trefethen (2007). Parabolic and hyperbolic contours for computing the bromwich integral. *Mathematics of Computation* 76(259), 1341–1356.
- Widder, D. V. (2015). *Laplace transform (PMS-6)*. Princeton University Press.
- Wood, S. N. (2017). *Generalized additive models: An introduction with R*. CRC press.
- Zheng, Y., S. Rajasegarar, and C. Leckie (2015). Parking availability prediction for sensor-enabled car parks in smart cities. In *2015 IEEE Tenth International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP)*, pp. 1–6. IEEE.

A Parameter estimation in the time to event model

When estimating the parameters of the time to event model (13), we assume the duration times in states 0 and 1 to be independent, which allows for independent estimation of the model parameters $\zeta_j = (\alpha_j, \boldsymbol{\theta}_j^\top, \gamma_j)^\top$ for $j = 0, 1$. Recall that γ_j is the parameter which determines the variance of the frailties $v_j^{(i)}$. This seems plausible as the duration of a parking lot being clear should not be related to the duration a parking lot being occupied. This allows for a break of notation in this appendix. We drop state index j , parking lot index i is now a subscript instead of a superscript and time index t is understood to be explicitly contained in the linear predictor.

Many procedures for estimating the parameters in a shared frailty model have been discussed. An EM-Algorithm was proposed by Klein (1992) where the model parameters ζ and the frailty terms v_i are iteratively estimated. A penalized and likewise iterative estimation algorithm is suggested by Therneau et al. (2003). However, estimates of ζ can also be obtained directly by maximizing the marginal log-likelihood of the model. Following the arguments in Kalbfleisch and Prentice (2011), for the conditional likelihood related to the i -th parking lot it holds that

$$L_i(\alpha, \boldsymbol{\theta} \mid v_i) = \prod_{k=1}^{n_i} [\alpha d_{ik}^\alpha v_i \exp(\eta_{ik})]^{\delta_{ik}} \exp(-\exp(\eta_{ik}) v_i (d_{ik})^\alpha). \quad (16)$$

Here, index k refers to the k -th observed duration time of a parking lot and n_i is the number of duration times observed for parking lot i . Moreover, $\delta_{ik} = 1$ if the corresponding event is observed and $\delta_{ik} = 0$ if the duration time d_{ik} is right censored. Note that we censor all duration times which are longer than 60 minutes which has the desired consequence that d_{ik} and δ_{ik} are independent. The marginal likelihood of the model can be obtained by integrating the frailty terms out of (16) for every i

$$L_{\text{marg}}(\zeta) = \prod_{i=1}^N \int_0^\infty L_i(\alpha, \boldsymbol{\theta} \mid v_i) h(v_j^{(i)}; \gamma) dv_i \quad (17)$$

with

$$h(v; \gamma) = \frac{v^{\frac{1}{\gamma}-1}}{\gamma^{\frac{1}{\gamma}} \Gamma(\frac{1}{\gamma})} \exp\left(-\frac{v}{\gamma}\right) \mathbb{1}_{(0, \infty)}(v)$$

denoting the density of the Gamma($\frac{1}{\gamma}, \frac{1}{\gamma}$) distribution. In this special case the integral in (17) is analytically tractable and the marginal log-likelihood of the model results

to

$$\ell_{\text{marg}}(\zeta) = \sum_{i=1}^N \left[\Delta_i \log \gamma + \log \frac{\Gamma\left(\frac{1}{\gamma} + \Delta_i\right)}{\Gamma\left(\frac{1}{\gamma}\right)} - \left(\frac{1}{\gamma} + \Delta_i\right) \log \left(1 + \gamma \sum_{k=1}^{n_i} (d_{ik})^\alpha\right) + \sum_{k=1}^{n_i} \delta_{ik} (\eta_{ik} + \log \alpha + (\alpha - 1) \log d_{ik}) \right],$$

where $\Delta_i = \sum_{k=1}^{n_i} \delta_{ik}$ and the maximum likelihood estimate of the fixed parameters is given by $\hat{\zeta} = \text{argmax} \ell_{\text{marg}}(\zeta)$ (Duchateau and Janssen, 2007). Finally, we need predicted values for the random effects u_i which are based on the posterior mean of the frailties $v_i = \log u_i$ given the observations $\mathbf{d}_i = (d_{i1}, \dots, d_{in_i})^\top$. Using Bayes' theorem, it can be shown that the posterior density $h(v_i | \mathbf{d}_i; \hat{\zeta})$ of v_i is equal to the density of a gamma distribution with shape parameter $a_i = \frac{1}{\hat{\gamma}} + \Delta_i$ and scale parameter $b_i = \frac{1}{\hat{\gamma}} + \sum_{k=1}^{n_i} \hat{\alpha} (d_{ik})^{\hat{\alpha}-1} \exp(\hat{\eta}_{ik})$ and therefore, $\hat{v}_i = \mathbb{E}(v_i | \mathbf{d}_i; \hat{\zeta}) = \frac{a_i}{b_i}$ (Nielsen et al., 1992).

B Numerical inversion of the Laplace transform

We here provide details about the algorithm which we employ to numerically compute the inverse Laplace transform of a Laplace transformed function $\tilde{f} = \mathcal{L}\{f\} : C \rightarrow \mathbb{C}$, where $C \in \mathbb{C}$ is the region of convergence. The complete derivation of the underlying theoretical results can be found in Valsa and Brančik (1998). Recall that the inverse Laplace transform is defined as the following Bromwich integral

$$f(t) = \mathcal{L}^{-1}\{\tilde{f}(u)\}(t) = \frac{1}{2\pi i} \int_{\gamma-i\infty}^{\gamma+i\infty} \tilde{f}(u) e^{ut} du, \quad (18)$$

where $u = x + iy$ and $\gamma > \min\{\Re(u) \mid u \in C\}$. The basic idea of the algorithm is to approximate the complex exponential in (18) as

$$e^{ut} \approx \frac{e^a}{2 \sinh(a - ut)} = \frac{e^{ut}}{1 - e^{-2(a-ut)}}$$

such that

$$\begin{aligned}
 f(t) &\approx f(t; a) = f(t) + \sum_{n=1}^{\infty} e^{-2na} f((2n+1)t) \\
 &= \frac{e^a}{t} \sum_{n=0}^{\infty} (-1)^n 2^{-[\mathbb{1}_{\{0\}}(n)]} \Re \left[\tilde{f} \left(\frac{a + in\pi}{t} \right) \right]
 \end{aligned} \tag{19}$$

where $a > 0$ is a tuning parameter. It is recommended to choose $a = 6$ (Valsa and Brančik, 1998). Naturally, the sum in (19) needs to be truncated at some positive integer n_t . In order to increase the speed of convergence, Valsa and Brančik (1998) propose to keep n_t rather low, but adding an Euler approximation of the subsequent n_e summands. Consequently, a finite numerical approximation for (18) is

$$\begin{aligned}
 f(t) &\approx f(t; a) \approx \frac{e^a}{t} \sum_{n=0}^{n_t} (-1)^n 2^{-[\mathbb{1}_{\{0\}}(n)]} \Re \left[\tilde{f} \left(\frac{a + in\pi}{t} \right) \right] \\
 &\quad + \frac{e^a 2^{-n_e}}{t} \sum_{n=n_t+1}^{n_t+n_e} (-1)^n \left[\sum_{k=n-n_t}^{n_e} \binom{n_e}{k} \right] \Re \left[\tilde{f} \left(\frac{a + in\pi}{t} \right) \right].
 \end{aligned}$$

Part II.

Statistical modeling of COVID-19 data

Chapter 5

Nowcasting fatal COVID-19 infections on a regional level in Germany

Contributing Article Schneble, M., De Nicola, G., Kauermann, G., Berger, U. (2021). Nowcasting fatal COVID-19 infections on a regional level in Germany. *Biometrical Journal*. 63: 471–489. <https://doi.org/10.1002/bimj.202000143>

Code and data <https://doi.org/10.1002/bimj.202000143>

Copyright 2020 The Authors. *Biometrical Journal* published by Wiley-VCH GmbH. This is an open access article under the terms of the Creative Commons Attribution License (<https://creativecommons.org/licenses/by/4.0/>), which permits use, distribution and reproduction in any medium, provided the original work is properly cited.

Author Contributions The idea of modeling fatal COVID-19 infections due to less strong dependency on testing strategies can be attributed to Göran Kauermann and Giacomo De Nicola. Marc Schneble had the idea of nowcasting fatal infections, and Göran Kauermann substantiated the respective model. The further contribution of Marc Schneble is given by the implementation of the model in the **R** language including the preparation of all figures in the paper. Furthermore, Marc Schneble wrote significant parts of the manuscript, especially Sections 2, 4.2-4.4, 5 and 6. All authors contributed to the manuscript writing and were involved in extensive proofreading.

Nowcasting fatal COVID-19 infections on a regional level in Germany

Marc Schneble¹  | Giacomo De Nicola¹ | Göran Kauermann¹ | Ursula Berger²

¹ Department of Statistics,
Ludwig-Maximilians-University Munich,
Munich, Germany

² Institute for Medical Information
Processing, Biometry, and Epidemiology,
Ludwig-Maximilians-University Munich,
Munich, Germany

Correspondence

Marc Schneble, Department of Statistics,
Ludwig-Maximilians-University Munich,
Ludwigstr. 33, 80539 Munich, Germany.
Email: marc.schneble@stat.uni-muenchen.de



This article has earned an open data badge “**Reproducible Research**” for making publicly available the code necessary to reproduce the reported results. The results reported in this article could fully be reproduced.

Abstract

We analyse the temporal and regional structure in mortality rates related to COVID-19 infections, making use of the openly available data on registered cases in Germany published by the Robert Koch Institute on a daily basis. Estimates for the number of present-day infections that will, at a later date, prove to be fatal are derived through a nowcasting model, which relates the day of death of each deceased patient to the corresponding day of registration of the infection. Our district-level modelling approach for fatal infections disentangles spatial variation into a global pattern for Germany, district-specific long-term effects and short-term dynamics, while also taking the age and gender structure of the regional population into account. This enables to highlight areas with unexpectedly high disease activity. The analysis of death counts contributes to a better understanding of the spread of the disease while being, to some extent, less dependent on testing strategy and capacity in comparison to infection counts. The proposed approach and the presented results thus provide reliable insight into the state and the dynamics of the pandemic during the early phases of the infection wave in spring 2020 in Germany, when little was known about the disease and limited data were available.

KEYWORDS

COVID-19, disease mapping, generalized regression model, nowcasting

1 | INTRODUCTION

In March 2020, COVID-19 became a global pandemic. From Wuhan, China, the virus spread across the whole world, and with its diffusion more and more data became available to scientists for analytical purposes. In daily reports, the WHO provides the number of registered infections as well as the daily death toll globally (<https://www.who.int/>). It is inevitable for the number of registered infections to depend on the testing strategy in each country (see, e.g., Cohen & Kupferschmidt, 2020). This has a direct influence on the number of undetected infections (see, e.g., Li et al., 2020), and first empirical analyses aim to quantify how detected and undetected infections are related (see, e.g., Niehus, De Salazar, Taylor, & Lipsitch, 2020). Though similar issues with respect to data quality hold for the reported number of fatalities

This is an open access article under the terms of the [Creative Commons Attribution](https://creativecommons.org/licenses/by/4.0/) License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited.

© 2020 The Authors. *Biometrical Journal* published by Wiley-VCH GmbH.

(see, e.g., Baud et al., 2020), the number of deaths can overall be considered a more reliable source of information than the number of registered infections. The results of the ‘Heinsberg study’ in Germany point in the same direction (Streeck et al., 2020). A thorough analysis of death counts can in turn generate insights on changes in infections as proposed in Flaxman et al. (2020) (see also Ferguson et al., 2020). In this paper, we pursue the idea of directly modelling registered death counts related to COVID-19 instead of registered infections. In other words, we restrict our analysis to fatal COVID-19 cases only, omitting recovered or symptom-free infections. We analyse data from Germany and break down the analyses to a regional level. Such regional view is apparently immensely important, considering the local nature of some of the outbreaks, for example in Italy (see, e.g. Grasselli, Pesenti, & Cecconi, 2020; Grasselli, Zangrillo, & Zanella, 2020), France (see, e.g., Massonnaud, Roux, & Crépey, 2020) or Spain and can assist local health authorities in monitoring the disease and planning infection control measures.

The analysis of fatalities has, however, an inevitable time delay and requires to take the course of the disease of COVID-19 patients into account. In particular, in this paper we consider the timespan between the registration of the infection through local health authorities and the report of its deadly outcome by the Robert Koch Institute (RKI). A first approach on modelling and analysing the time from illness and onset of symptoms to reporting and further to death is given in Jung et al. (2020) (see also Linton et al., 2020). Understanding the delay between onset and registration of an infection and, for severe cases, the time between registered infection and death, can be of vital importance. Knowledge on those timespans allows us to obtain estimates for the number of infections that are expected to be fatal based on the number of infections registered on the present day. The statistical technique to obtain such estimates is called nowcasting (see, e.g., Höhle & an der Heiden, 2014) and traces back to Zeger, See, and Diggle (1989) or Lawless (1994). Nowcasting in COVID-19 data analyses is not novel and is, for instance, used in Günther, Bender, Küchenhoff, Katz, and Höhle (2020) for nowcasting daily infection counts in Germany, that is to adjust daily reported new infections to include infections which occurred the same day but were not yet reported. Altmeyd, Rocklöv, and Wallin (2020) apply nowcasting techniques to Swedish data and Bird and Nielsen (2020) provide nowcasting fatalities in English hospitals. We extend this approach to model the duration between the registration date of an infection and its fatal outcome, accounting for additional covariates. To do so, we combine a nowcasting model with a spatio-temporal regression model.

We analyse the number of fatal cases of COVID-19 infections in Germany using district-level data. The data are provided by the RKI (www.rki.de), the German federal government agency and scientific institute responsible for health reporting, disease control and prevention in humans. They report the cumulative number of deaths in different gender and age groups for each of the 412 administrative districts in Germany, together with the date of registration of the infection. The data are available in dynamic form through daily downloads of the updated cumulated numbers of deaths. Comparing two consecutive daily downloads allows to construct a new dataset which contains both the date at which a COVID-19 disease is registered and the date at which a fatality is reported to the RKI, with the latter usually being reported at a later time point. We employ flexible statistical models with smooth components (see, e.g., Wood, 2017), assuming the district-specific number of fatalities to be negatively binomial distributed, which permits to also account for possible overdispersion in the data. The spatial structure in the death rate is incorporated in two ways: First, we assume a spatial correlation of the number of deaths by including a long-range smooth spatial death intensity. This allows to map a general pattern of the spread of the disease over Germany, which shows that regions of Germany are affected to different extents. On top of this long-range effect, we include two types of unstructured region-specific effects. An overall region-specific effect reflects the situation of a district as a whole, while a short-term effect mirrors region-specific variations of fatalities over time and captures local outbreaks as happened in, for example Heinsberg (North-Rhine-Westphalia) or Tirschenreuth (Bavaria). This effect can be seen as an unstructured time-space interaction. In addition to the spatial components, we include an overall temporal effect to capture dynamic changes in the number of fatal infections for Germany. The latter effect mirrors the overall flattening of the infectious situation in the considered time period, that is spring 2020. Besides the spatio-temporal character, our modelling approach further adjusts for the district-specific age and gender structure.

Modelling infectious diseases is a well-developed field in statistics, and we refer to Held, Meyer, and Bracher (2017) for a general overview of the different models. We also refer to the powerful R package *surveillance* (Meyer, Held, & Höhle, 2017). Since our focus is on analysing district-specific dynamics, both structured and unstructured, as well as dynamic behaviour of fatal infections, we prefer to make use of generalized additive regressions implemented in the *mgcv* package in R, which also allows to decompose the spatial component in more depth.

The paper is organized as follows. In Section 2 we describe the data. Section 3 introduces our model, while Section 4 discusses the necessity of incorporating a nowcasting model. Section 5 shows the results of our analysis which are then refined to subgroups of the data in Section 6. Section 7 concludes the paper by also discussing the limitations of our modelling exercise.

TABLE 1 Illustration of the data structure, showing downloads of the data from April 25 and April 26, 2020 as an example. To facilitate reproducibility, the original column names used in the RKI datasets are given in brackets below our English notation

	District (Landkreis)	Age Group (Altersgruppe)	Gender (Geschlecht)	Infections (Anzahl Fall)	Fatal Infections (Anzahl Todesfall)	Registration Date (Meldedatum)	Reporting Date (Datenstand)
Data downloaded on April 25, 2020	⋮	⋮	⋮	⋮	⋮	⋮	⋮
	Munich City	60–79	F	3	1	April 22, 2020	April 25, 2020
	Munich City	60–79	M	5	1	April 22, 2020	April 25, 2020
	⋮	⋮	⋮	⋮	⋮	⋮	⋮
Data downloaded on April 26, 2020	⋮	⋮	⋮	⋮	⋮	⋮	⋮
	Munich City	60–79	F	6	2	April 22, 2020	April 26, 2020
	Munich City	60–79	M	5	1	April 22, 2020	April 26, 2020
	⋮	⋮	⋮	⋮	⋮	⋮	⋮

2 | DATA

We make use of the COVID-19 dataset (Esri Deutschland GmbH, 2020) provided by the RKI on a daily basis for the 412 districts in Germany (which also include the 12 districts of Berlin separately). The data are collected by the RKI, but originate from the district-based health authorities (*Gesundheitsämter*). Due to different population sizes in the districts, and certainly also because of different local situations, some health authorities transmit the daily numbers to the RKI with a delay. This happens in particular over the weekend, a fact that we need to take into account in our model. We have daily downloads of the data since March 27, 2020. We here choose to focus on a phase of the COVID-19 pandemic in which the death toll in Germany was high. The subsequent analysis was thus conducted with data up to May 14, 2020, and was performed considering only deadly infections with registration dates from March 26, 2020 until May 13, 2020 (the day before that of the analysis).

Table 1 illustrates an exemplary extract of the data that are available. For each of the 412 districts, the data contain the cumulated number of laboratory-confirmed COVID-19 infections as well as the cumulated number of deaths related to COVID-19 for each district of Germany, stratified by age group (15–34, 35–59, 60–79 or 80+), gender, and the date of registration of the infection by the local public health authorities. The time stamp for a fatal outcome always refers to the registration date of the infection and *not* to the individual's date of death. Therefore, the numbers in the column 'Fatal infections' cannot exceed the numbers shown in the column 'Infections'. Even though the time point of infection obviously precedes that of death, registration of an infection can also occur after death, for example when a post-mortem test is conducted, or when test results arrive after the patient has passed away. In the former case, the registration date are set to the day of death by the local health authority. Also note that it is not indicated in the dataset whether a fatal infection resulted from a post-mortem test, and that no information on whether the patient has died *with* or *because* of a COVID-19 infection is included.

The cumulative numbers are reported on a daily basis by the RKI, which is mirrored in the column 'Reporting date' in Table 1. The reporting date always corresponds to the query date and the download date of the data. In Table 1, we see that the number of reported infections with registration date April 22, 2020, which relate to females in the age group 60–79 living in the city of Munich, increases by three from April 25, 2020 to the following day. In the same period, the number of fatal infections increased by one. Thus, we can deduce that three registered infections in this sub-population were reported with a delay of 4 days. The single newly reported fatal infection belongs to an individual of this sub-population for which the time between registration by the local health authorities and reported death amounts to 4 days. In this paper, we are especially interested in the latter quantity, which we model as a duration time. It is of importance to note that we can derive such information only due to daily downloads of the dataset, which are not being provided retrospectively.

We refrain from providing general descriptive statistics on the spatio-temporal distribution of confirmed COVID-19 infections here, since these numbers are already visualized on the RKI dashboard (Robert Koch-Institut, 2020; see also StaBLab, LMU Munich, 2020). However, the number of fatal infections is less often taken into account. Thus, in Figure 1 we show the empirical duration between the day of registration as COVID-19 infected by the local health authorities and the day on which the death has been reported by the RKI (based on the data until May 14, 2020). Due to the aforementioned reporting delay, the minimum duration is 1 day. Note that these plots show stapled bar charts, highlighting the counts by gender. We see that considerably more fatal infections originate from the age group 80+. Regarding the age group 80– (aggregated age groups 15–39, 40–59 and 60–79), we see that males are much more affected than females, whereas in the age group 80+ the counts are more balanced. Finally, in both age groups there are a small number of deaths,

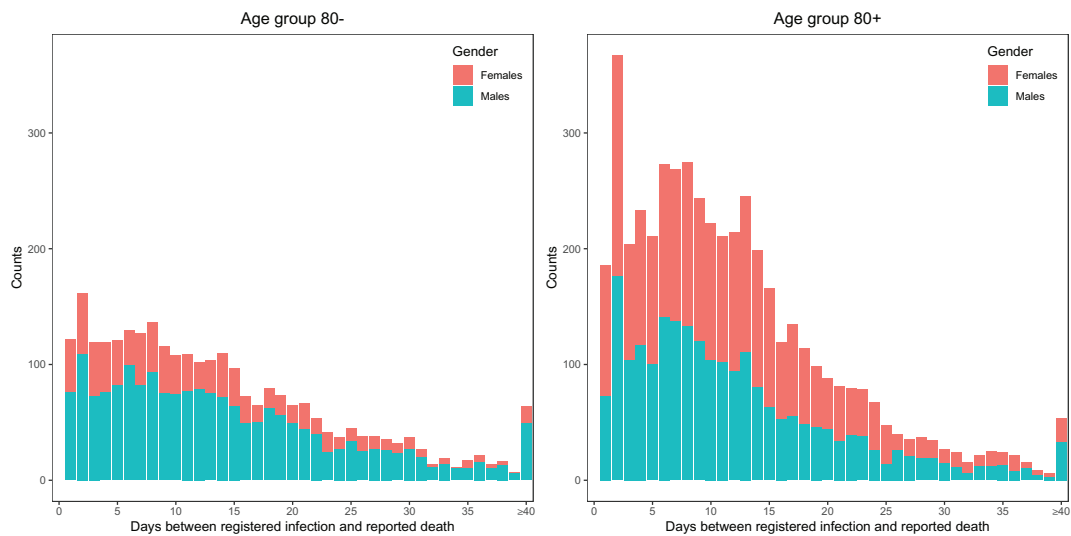


FIGURE 1 Stapled bar chart of the counts of fatal infections depending on days between registered infection and reported death. Only data reported until May 14, 2020 is considered here (left panel: age group 80– (less than 80 years), right panel: age group 80+ (80 years or older))

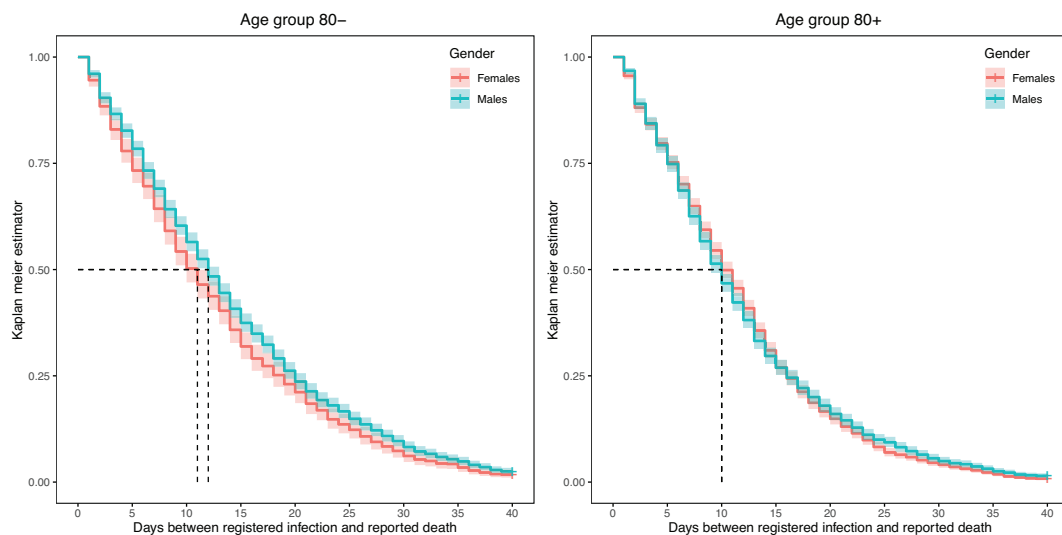


FIGURE 2 Kaplan–Meier estimators of the data shown in Figure 1 with 95% confidence intervals

which were reported 40 or more days after the registration of the COVID-19 infection. Kaplan–Meier estimators of the duration between registered infection and reported death are shown in Figure 2 for age groups 80– and 80+ by gender. Here we especially see that the median duration time of elderly patients is slightly shorter when compared to the younger age groups.

3 | MORTALITY MODEL

Let $Y_{t,r,c}$ denote the number of deaths due to COVID-19 with time point of registration $t = 0, \dots, T$ in district/region r and cohort c , where the cohort c is characterized by age group and gender of the deceased. Time index $t = T$ corresponds to the day of analysis, which is May 14, 2020, and $t = 0$ corresponds to March 26, 2020. Not all fatalities with registered infection at time point t have been observed at time T , as some deaths will occur later. We therefore need a model for nowcasting, which is discussed in the next section.

For now, we assume all $Y_{t,r,c}$ to be known. A family of discrete distributions which is supported on the set of nonnegative integers and also allows to account for possible overdispersion in the data is the negative binomial distribution. Therefore, we model those numbers as according to

$$Y_{t,r,c} \sim \text{NB}(\lambda_{t,r,c}, \phi), \quad (1)$$

where $\mathbb{E}(Y_{t,r,c}) = \lambda_{t,r,c}$ and the constant dispersion parameter ϕ relates to the variance by $\text{Var}(Y_{t,r,c}) = \lambda_{t,r,c} + \phi\lambda_{t,r,c}^2$. We model the mean $\lambda_{t,r,c}$ of the response $Y_{t,r,c}$ through a regression model and specify

$$\begin{aligned} \lambda_{t,r,c} = \exp\{ & \beta_0 + \text{age}_c \beta_{\text{age}} + \text{gender}_c \beta_{\text{gender}} \\ & + \text{age}_c \text{gender}_c \beta_{\text{age, gender}} + \text{weekday}_t \beta_{\text{weekday}} \\ & + m_1(t) + m_2(s_r) + u_{r0} + \mathbb{1}_{\{t \geq T-14\}} u_{r1} + \log(\text{pop}_{r,c})\}, \end{aligned} \quad (2)$$

where the linear predictor is composed as follows:

- β_0 is the intercept.
- β_{age} and β_{gender} are the age- and gender-related regression coefficients, and $\beta_{\text{age, gender}}$ is the coefficient that models the interaction between age and gender.
- β_{weekday} are regression coefficients, which relate to the weekday of the registration date as COVID-19 infected.
- $m_1(t)$ is an overall smooth time trend, with no prior structure imposed on it.
- $m_2(s_r)$ is a smooth spatial effect, where s_r is the geographical centroid of district/region r .
- u_{r0} and u_{r1} are district-/region-specific random effects, which are independently and identically distributed (i.i.d.) and follow a normal prior probability model. While u_{r0} specifies an overall level of the death rate for district r over the entire observation time, u_{r1} is a spatio-temporal effect that reveals region-specific dynamics by allowing the regional effects to differ for the last 14 days.
- $\text{pop}_{r,c}$ is the gender and age group-specific population size in district/region r and serves as an offset in our model.

We here emphasize that we fit spatial effects of different types: We model a smooth spatial effect, that is $m_2(s_r)$, which takes the correlation between the fatal infections of neighbouring districts/regions into account and gives a global overview of the spatial distribution of fatal infections. In addition to that we also have unstructured district-/region-specific effects $\mathbf{u}_r = (u_{r0}, u_{r1})^\top$, which capture local behaviour related to single districts only. While u_{r0} captures the corresponding long-term effect, u_{r1} captures the short-term effect of the last 14 days; see (2). This means that we also model a dichotomous and unstructured interaction of space with time. The district-specific effects \mathbf{u}_r are considered as random, with prior structure

$$\mathbf{u}_r \sim \mathcal{N}(\mathbf{0}, \Sigma_u) \text{ i.i.d} \quad (3)$$

for $r = 1, \dots, 412$. The prior variance matrix Σ_u is estimated from the data. The predicted values $\hat{\mathbf{u}}_r$ (i.e. the posterior mode) exhibit districts that show unexpectedly high or low death tolls when adjusted for the global spatial structure and for age- and gender-specific population sizes.

While model (2) is complex and highly structured, note that no autoregressive components are included in the linear predictor in (2). We will demonstrate in Section 6.4 below that auto-correlation is of negligible size, and that time dependence is fully captured by $m_1(t)$ as well as the unstructured effects u_{r1} .

The mortality model defined through (1) and (2) belongs to the model class of generalized additive mixed model (see, e.g., Wood, 2017). The smooth functions are estimated by penalized splines without restrictions on the number of degrees of freedom, with a quadratic penalty that can be comprehended as a normal prior (see, e.g., Wand, 2003). The same type of prior structure holds for the region-specific random effects \mathbf{u}_r . In other words, smooth estimation and random effect estimation can be accommodated in one fitting routine, which is implemented in the R package `mgcv`. This package has been used to fit the model, so that no extra software implementation was necessary. This demonstrates the practicability of the proposed method. Our analysis is completely reproducible, with code and data openly available and downloadable from our GitHub repository.¹

¹ <https://github.com/MarcSchneble/Nowcasting-Fatal-COVID-19-Infections>

4 | NOWCASTING MODEL

4.1 | Model description

The above model cannot be fitted directly to the available data, since we need to take the course of the disease on the individual level into account. This means that the final number of fatal outcomes for infections registered on date $t < T$ is not known at the time point of analysis $t = T$, since not all patients with a fatal outcome of the disease have died yet. This requires the implementation of nowcasting. Due to the sparsity of the data, we perform the nowcast on a national level, that is we cumulate the numbers over district/region r . For reasons of notation, we temporarily drop the gender and age-related subscript g , and we simply notate the cumulated number of deaths with registered infections at day t with Y_t .

Let $N_{t,d}$ denote the number of deaths reported on day $t + d$ for infections registered on day t . Assuming that the true date of death is at $t + d$, or at least close to it, we ignore any time delays between time of death and its notification to the health authorities. We call d the duration in days between the registration date as a COVID-19 patient and the reported day of death, where $d = 1, \dots, d_{\max}$. Here, d_{\max} is a fixed reasonable maximum duration, which we set to 40 days (see, e.g., Wilson, Kvalsvig, Barnard, & Baker, 2020). This is also motivated by the means of Figure 1. The minimum duration is one day, since the RKI daily reports the new numbers, which they have received from the public health departments the day before. In nowcasting, we are interested in the cumulated number of deaths for infections registered on day t , which we define as

$$Y_t = \sum_{d=1}^{d_{\max}} N_{t,d}.$$

Therefore, the total number of deaths with a registered infection at t becomes available only after d_{\max} days. In other words, only after d_{\max} days we know exactly how many deaths occurred due to an infection which was registered on day t . We define the partial cumulated sum of deaths as

$$C_{t,d} = \sum_{l=1}^d N_{t,l}$$

so that by definition $C_{t,d_{\max}} = Y_t$.

On day $t = T$, when the nowcasting is performed, we are faced with the following data constellation, where NA stands for not (yet) available:

t	d				Reported deaths
	1	2	...	d_{\max}	
0	$N_{0,1}$	$N_{0,2}$...	$N_{0,d_{\max}}$	Y_0
1	$N_{1,1}$	$N_{1,2}$...	$N_{1,d_{\max}}$	Y_1
⋮	⋮	⋮	⋮	⋮	⋮
$T - d_{\max}$	$N_{T-d_{\max},1}$	$N_{T-d_{\max},2}$...	$N_{T-d_{\max},d_{\max}}$	$Y_{T-d_{\max}}$
$T - d_{\max} + 1$	$N_{T-d_{\max}+1,1}$	$N_{T-d_{\max}+1,2}$...	NA	$C_{T-d_{\max}-1,d_{\max}-1}$
⋮	⋮	⋮	⋮	⋮	⋮
$T - 2$	$N_{T-2,1}$	$N_{T-2,2}$	NA	NA	$C_{T-2,2}$
$T - 1$	$N_{T-1,1}$	NA	NA	NA	$C_{T-1,1}$

We may consider the timespan between registered infection and (reported) death as a discrete duration time taking values $d = 1, \dots, d_{\max}$. Let D be the random duration time, which by construction is a multinomial random variable. In principle, for each death we can consider the pairs (D_i, t_i) as i.i.d. and we aim to find a suitable regression model for D_i given t_i , including potential additional covariates $x_{t_i,d}$. We make use of the sequential multinomial model (see Agresti, 2010) and define

$$\pi(d; t, x_{t,d}) = P(D = d | D \leq d; t, x_{t,d}).$$

Let $F_t(d)$ denote the corresponding cumulated distribution function of D which relates to probabilities $\pi(\cdot)$ through

$$\begin{aligned} F_t(d) &= P_t(D \leq d) = P(D \leq d | D \leq d+1) \cdot P(D \leq d+1) \\ &= (1 - \pi(d+1; \cdot)) \cdot (1 - \pi(d+2; \cdot)) \cdot \dots \cdot (1 - \pi(d_{\max}; \cdot)) \\ &= \prod_{k=d+1}^{d_{\max}} (1 - \pi(k; \cdot)) \end{aligned} \quad (4)$$

for $d = 1, \dots, d_{\max} - 1$ and $F_t(d_{\max}) = 1$.

We generalize notation again by including the subscript g , which in the nowcasting model only distinguishes between the two age groups 80– and 80+. The available data on cumulated death counts now allow us to estimate the conditional probabilities $\pi(d; \cdot)$ for $d = 2, \dots, d_{\max}$. In fact, the sequential multinomial model allows to look at binary data such that

$$N_{t,d,c} \sim (\text{quasi-})\text{Binomial}(C_{t,d,c}, \pi(d; t, c, x_{t,d})) \quad (5)$$

with

$$\text{logit}(\pi(d; t, c, x_{t,d})) = s_1(t) + s_2(d) + s_3(d) \cdot \mathbb{1}_{\text{age}\{80+\}} + x_{t,d}\gamma, \quad (6)$$

where

- $s_1(t)$ is an overall smooth time trend over calendar days.
- $s_2(d)$ is a smooth duration effect, capturing the course of the disease.
- $s_3(d)$ is a varying smooth duration effect, capturing interaction between the dynamics of the disease and age, particularly for the age group 80+. Note that with effect $s_3(d)$ we take into account that for infections with a fatal outcome, the individual course of the disease for elderly patients might differ compared to younger patients.
- $x_{t,d}$ are covariates which may be time and duration specific.

By utilizing a quasi-likelihood model (Fahrmeir, Kneib, Lang, & Marx, 2007) as in (5), we account for possible overdispersion in the data, which results in adjusted standard errors of the parameter estimates, while, however, the estimates themselves are the same when compared to the fit of a binomial model.

Assuming that D , the duration between a registered fatal infection and its reported death, is independent of the number of fatal COVID-19 infections, we obtain the relationship

$$\mathbb{E}(C_{t,d,c}) = F_{t,c}(d) \cdot \mathbb{E}(Y_{t,c}). \quad (7)$$

Note further that if we model $Y_{t,c}$ with a negative binomial model as presented in the previous section, we have no final observation $Y_{t,c}$ for time points $t > T - d_{\max}$. Instead, we have observed $C_{t,T-t,c}$, which relates to the mean of $Y_{t,c}$ through (7) by $C_{t,T-t,c} = F_{t,c}(T-t) \cdot \mathbb{E}(Y_{t,c})$. Including therefore $\log F_{t,c}(T-t)$ as additional offset in model (2) allows to fit the model as before, but with the nowcasted number of fatal infections included. That means, instead of $\lambda_{t,r,c}$ as in (2), the expected number of fatal infections are now parameterized by $\lambda_{t,r,c}^* = \lambda_{t,r,c} \exp(\log F_{t,c}(T-t))$, where the latter multiplicative term is included as additional offset in the model.

4.2 | Results for nowcasting

We fit the nowcasting model (5) with parameterization (6). We include a weekday effect for the registration date of the infection with reference category ‘Monday’. The estimates of the fixed linear effects are shown in Table 2. The fitted smooth effects are shown in Figure 3. The top panel shows the effect over calendar time, which is very weak and confirms that the individual course of the disease hardly varies over time. This is supported by the fact that the German healthcare

TABLE 2 Estimated fixed linear effects (standard errors in brackets) in the nowcasting model (6). Parameters and their standard errors are given on the log scale. The relative risk is given together with 95% confidence intervals. The reference for the weekdays is Monday

	Effect (SE)	exp(Effect) Relative risk	95% Confidence interval of relative risk
Intercept	−3.12 (0.045)	0.04	[0.04, 0.05]
Tuesday	0.06 (0.060)	1.06	[0.94, 1.19]
Wednesday	0.11 (0.059)	1.12	[0.99, 1.25]
Thursday	0.20 (0.058)	1.23	[1.09, 1.38]
Friday	0.26 (0.059)	1.30	[1.16, 1.45]
Saturday	0.27 (0.063)	1.31	[1.16, 1.48]
Sunday	0.20 (0.068)	1.22	[1.07, 1.40]

system remained stable over the considered period, and hence survival did not depend on the date on which the infection was notified.

The bottom panel of Figure 3 shows the course of the disease as a smooth effect over the time between registration of the infection and death. We see that the probabilities $\pi(d; \cdot)$ decrease in d , where this effect is the strongest in the first days after registration. Thus, most of the COVID-19 patients with fatal infections are expected to die not long after their registration date. We also see no overall significant difference in the duration effect between the age groups 80– and 80+, since the fitted curves $s_2(d)$ and $s_2(d) + s_3(d)$ hardly differ. To some extent, this was already visible from Figure 1. This shows that, given that a registered case ends with a fatal outcome, the individual’s course of the disease does not depend on the age group. The effect of d becomes easier to interpret by visualizing the resulting distribution function $F_{t,c}(d)$, where here g refers to the age group 80+. This is shown in Figure 4 for two different values of t , that is April 13 and May 13. The plot also shows how the course of the disease hardly varies over calendar time: In fact, the small differences between the two distribution functions is dominated by the weekday effect, since the red curve is related to a Monday while the blue one is from a Wednesday.

4.3 | Nowcasted number of fatal infections

On the day of analysis, we do not observe the total counts of deaths for recently registered infections. This means that there are an unknown number of currently infected people which will die at a future point in time. We therefore nowcast those numbers, that is we predict the prospective deaths which can be attributed to all registration dates up to today. This is done on a national level, and the resulting nowcast of fatal infections for Germany is shown in Figure 5. For example, on May 14, 2020 there are 25 deaths reported where the infection was registered on May 5 (red bullets on May 5). We expect this number to increase to about 50 when all deaths due to COVID-19 for this registration date will have been reported (green triangles on May 5). Naturally, the closer a date is to the present, the larger the uncertainty in the nowcast will be. This is shown by the shaded bands. Details on how the statistical uncertainty has been quantified are provided below. In Section 5, we incorporate the nowcasting results into the mortality model as discussed before, but the nowcast results are interesting in their own right. The curve confirms that the number of fatal infections is decreasing since the beginning of April. Note that the curve also mirrors the ‘weekend effect’ in registration, as less infections are reported on Sundays.

Since we are now more than $d_{\max} = 40$ days after the day of analysis (May 14, 2020), we can assess the predictive accuracy of our nowcast. Therefore, we also show in Figure 5 the counts of fatal infections, which we observe 40 days after the respective registration date. We see that our nowcast performs in general very well. However, there are a handful of registration dates for which the nowcasted values were clearly outside of the prediction intervals. Most remarkably, the cumulative number of fatal infections for registered infections on April 8, 2020 has dropped after May 14, 2020. This happens in the rare case in which the database has been modified retrospectively by the local health authorities.

4.4 | Uncertainty quantification in nowcasting

In Figure 5, we have shown the nowcasting results along with uncertainty intervals shaded in grey. These were constructed using a bootstrap approach as follows. Given the fitted model, we simulate $n = 10,000$ times from the asymptotic joint

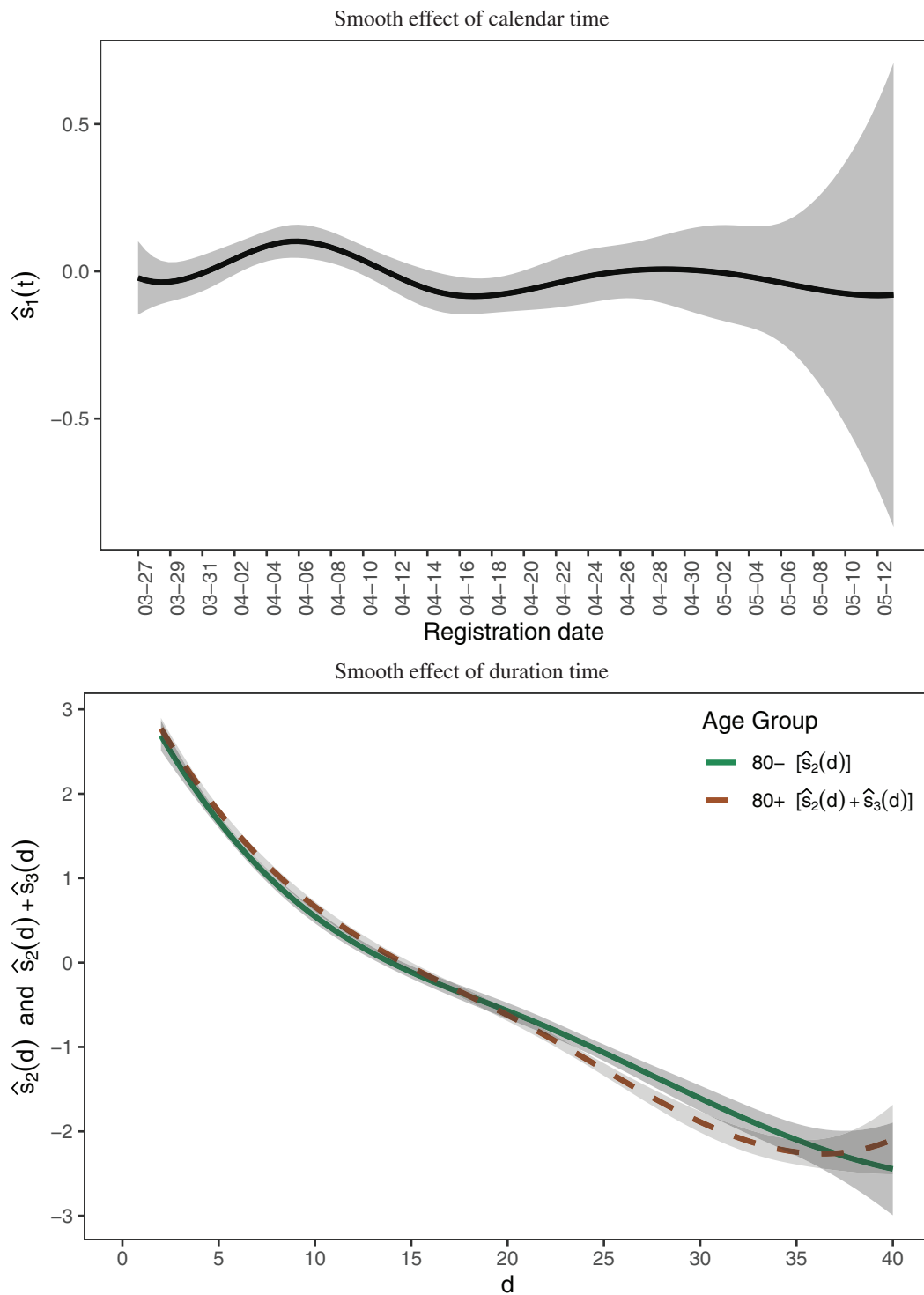


FIGURE 3 Estimates of smooth effects in the nowcasting model

normal distribution of the estimated model parameters which results through (4). This leads to a set of bootstrapped distribution functions $\mathcal{F} = \{\hat{F}_t^{(i)}(T - t), i = 1, \dots, n; t = T - d_{\max} + 1, \dots, T - 1\}$. This set is used to compute the simulated nowcasts $\hat{Y}_t^{(i)} = C_{t, T-t} / \hat{F}_t^{(i)}(T - t)$ applying (7), where $C_{t, T-t}$ is the observed partial cumulated sum of deaths at time point $T - t$ with registration date t . The point-wise lower and upper bounds of the 95% prediction intervals for the nowcast for Y_t are then given by the 2.5 and the 97.5 quantiles of the set $\{\hat{Y}_t^{(i)}, i = 1, \dots, n\}$, respectively.

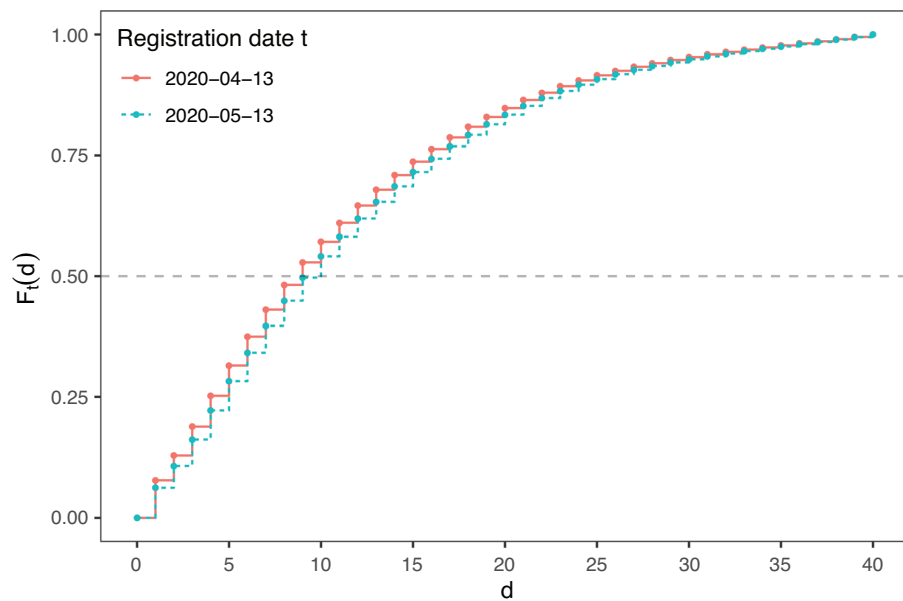


FIGURE 4 Fitted distribution function $F_t(d)$ for the age groups 80+ and 80–, where t corresponds to Wednesday, May 13, 2020

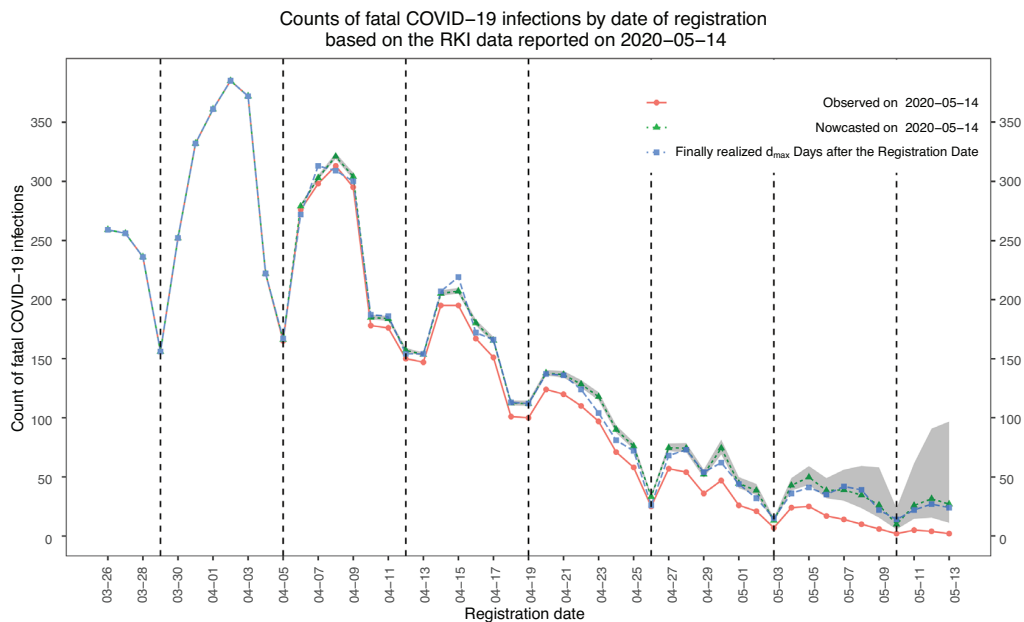


FIGURE 5 Observed (red line) and nowcasted (blue line) of daily death counts due to a COVID-19 infection on May 14, 2020 including 95% prediction intervals (shaded areas). Sundays are marked by a dashed vertical line. Finally realized death counts (d_{\max} after the respective registration date) are shown as blue squares

5 | RESULTS OF THE MORTALITY MODEL

We first discuss the estimates of the fixed linear effects included in model (2), which are shown in Table 3. We see that both age and gender play a major role when estimating the numbers of fatal infections. Elderly people exhibit a much higher death rate from COVID-19, which is, for males (females) in the age group 80+, around 80 times ($148 \approx \exp(4.39 + 0.61)$ times) higher than in the reference age group 35–59. This already hints at a remarkable difference between genders, where the expected death rate for females in the reference age group is around 60% ($\approx 1 - \exp(-0.94)$) lower than the

TABLE 3 Estimated fixed linear effects (standard errors in brackets) in the mortality model (2). Parameters and their standard errors are given on the log scale. The relative risk is given together with 95% confidence intervals. The reference category for age is the age group 35–59. The reference for the weekdays is Monday

		Effect (S.E.)	exp(Effect) Relative risk	95% Confidence interval of relative risk
	Intercept	−15.90 (0.095)	$1.27 \cdot 10^{-7}$	$[1.05 \cdot 10^{-7}, 1.53 \cdot 10^{-7}]$
Patient related	Female	−0.94 (0.142)	0.39	[0.29, 0.51]
	Age 15–34	−2.53 (0.325)	0.08	[0.04, 0.15]
	Age 15–34 Female	−0.18 (0.674)	0.84	[0.22, 3.18]
	Age 60–79	2.61 (0.081)	13.58	[11.60, 15.90]
	Age 60–79 Female	0.07 (0.151)	1.07	[0.80, 1.45]
	Age 80+	4.41 (0.080)	81.9	[70.20, 95.90]
	Age 80+ Female	0.61 (0.147)	1.83	[1.38, 2.45]
Reporting related	Tuesday	0.20 (0.051)	1.22	[1.10, 1.35]
	Wednesday	0.23 (0.052)	1.26	[1.14, 1.39]
	Thursday	0.24 (0.050)	1.28	[1.16, 1.41]
	Friday	0.10 (0.051)	1.10	[1.00, 1.22]
	Saturday	−0.12 (0.054)	0.88	[0.79, 0.98]
	Sunday	−0.41 (0.058)	0.66	[0.59, 0.74]

corresponding death rate for males. When considering the total gender-related numbers of fatal infections in the age group 80+ (see Figure 1), the difference between the genders is seemingly very small. However, by respecting the district-, gender- and age-specific population sizes in our model we see that the death rate of females in the age group 80+ is still around 28% ($\approx 1 - \exp(-0.94 + 0.61)$) lower when compared to the male population in this age group. Furthermore, we see that significantly less deaths are attributed to infections registered on Sundays compared to weekdays, due to the existing reporting delay during weekends.

Our model includes a global smooth time trend representing changes in the death rate since March 26. This is visualized in Figure 6. The plotted death rate is scaled to give the expected number of deaths per 100,000 people in an average district for the reference group, that is males in the age group 35–59. Overall, we see a peak in the death rate on April 3 and a downwards slope until the end of April. However, our nowcast reveals that the rate remains constant since beginning of May. Note that such developments cannot be seen by simply displaying the raw death counts of these days. The nowcasting step inevitably carries statistical uncertainty, which is taken into account in Figure 6 by including best and worst case scenarios. The latter are based on bootstrapped confidence intervals, where details are provided in Section 6.3 later in the paper.

Our aim is to investigate spatial variation and regional dynamics. To do so, we combine a global geographic trend for Germany with unstructured region-specific effects, where the latter uncovers local behaviour. In Figure 7, we combine these different components and map the fitted nowcasted death counts related to COVID-19 for the different districts of Germany, cumulated over the last 14 days before the day of analysis, that is May 14, 2020. While in most districts of Germany, the death rate is relatively low, some hotspots can be identified. Among those, Traunstein and Rosenheim (in the south-east part of Bavaria) are the most evident, but Greiz and Sonneberg (east and south part of Thuringia) stand out as well, to mention a few. A deeper investigation of the spatial structure is provided in Section 6, where we show the global geographic trend and provide maps that allow to detect new hotspot areas, after correcting for the overall spatial distribution of the infection.

6 | MORE RESULTS AND MODEL EVALUATION

6.1 | Spatial effects

It is of general interest to disentangle the two spatial components that we introduced in Section 3. We visualize the fitted global geographic trend $m_2(\cdot)$ for Germany in Figure 8. The plot confirms that, up to mid May 2020, the northern parts

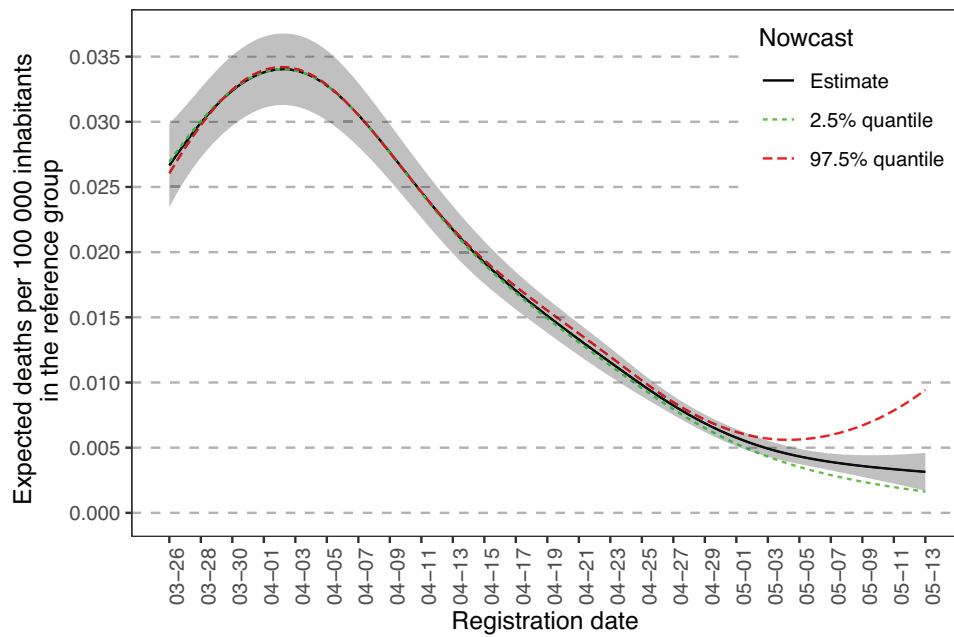


FIGURE 6 Fitted smoothed expected fatal COVID-19 infections per 100,000 inhabitants in the reference group (males aged between 35 and 59 in an average district) by registration date including 95% confidence bands as shaded area. Uncertainty resulting from the nowcast model is shown as dashed coloured lines

of the country are less affected by the disease in comparison to the southern states. The two plots in Figure 9 map the region-specific effects, that is the predicted long-term level of a district u_{r0} (left-hand side) and the predicted short-term dynamics u_{r1} (right-hand side). Both plots uncover quite some region-specific variability. In particular, the short-term dynamics u_{r1} (right plot) pinpoint districts with unexpectedly high nowcasted death rates in the last two weeks, after correcting for the global geographic trend and the long-term effect of the district. Some of the noticeable districts have already been highlighted in Section 3 above, but we can here detect further districts which are less evident in Figure 6: For instance, Steinfurt (in the north-west of North Rhine-Westphalia), Olpe (southern North Rhine-Westphalia) and Gotha (center of Thuringia) all show a relatively high rate of fatal infections.

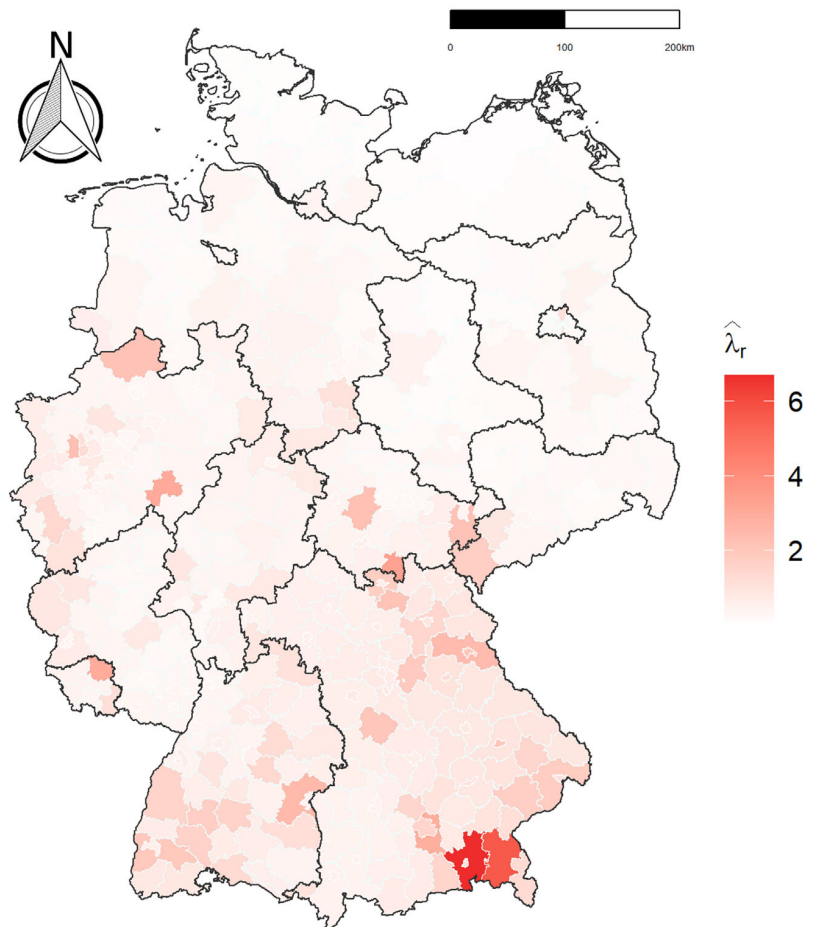
6.2 | Age group-specific analyses

A large portion of the registered fatal infections related to COVID-19 stems from people in the age group 80+. Locally, high numbers are often caused by an outbreak in a retirement home. Such outbreaks apparently have a different effect on the spread of the disease, and the risk of an epidemic infection caused by outbreaks in this age group is limited. Thus, the death rate among elderly people could vary differently across districts when compared to regional peaks in the death rate of the rest of the population. In order to respect this, we decompose the district-specific effects \mathbf{u}_r in (2) into $\mathbf{u}_r^{80-} = (u_{r0}^{80-}, u_{r1}^{80-})^\top$ for the age group 80– and $\mathbf{u}_r^{80+} = (u_{r0}^{80+}, u_{r1}^{80+})^\top$ for the age group 80+, where the age group 80– consists of the aggregated age groups 15–34, 35–59 and 60–79. We put the same prior assumption on the random effects as we did in (3), but now the variance matrix that needs to be estimated from the data has dimension 4×4 .

The fitted age group-specific random effects are shown in Figure 10, where the \mathbf{u}_r^{80-} are shown in the top panel and the \mathbf{u}_r^{80+} in the bottom panel. Most evidently, the variation of the random effects is much higher in the age group 80+ when compared to the younger age groups, as more districts occur which are coloured dark blue or dark red, respectively. When comparing the district-specific short-term dynamics of the last 14 days (u_{r1}) in Figure 10 to those in Figure 9, we recognize that in most of the districts which recently experienced very high death intensities (with respect to the whole period of analysis), these stem from the age group 80+. As mentioned before, this can often be explained by outbreaks in retirement homes.

FIGURE 7 Nowcasted fatal COVID-19 infections per 100,000 inhabitants in each district in the timespan from Thursday, April 30 until Wednesday, May 13, 2020

Nowcasted fatal infections per 100 000 inhabitants with registration dates from 2020-04-30 until 2020-05-13



Based on data reported up to 2020-05-14.
Model includes registration dates from
2020-03-26 until 2020-05-13.

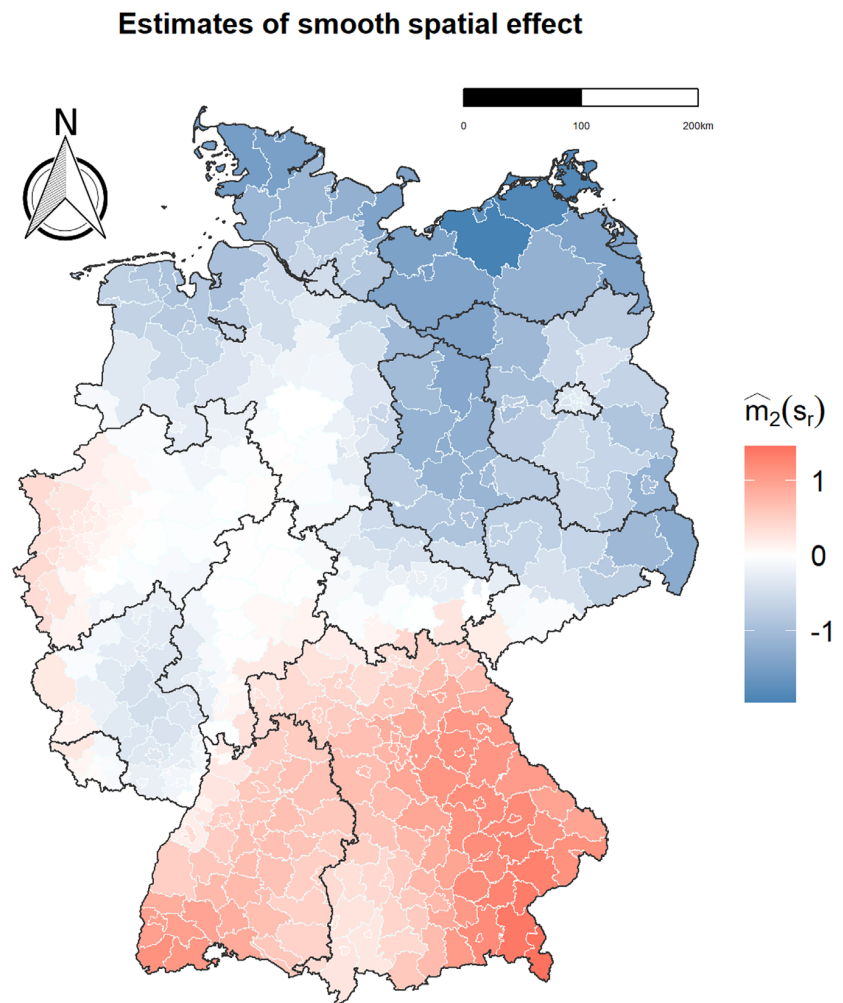
6.3 | Additional uncertainty in the mortality through the nowcast

When fitting the mortality model (1), we included the fitted nowcast model as offset parameter. This apparently neglects the estimation variability in the nowcasting model, which we explored via bootstrap as explained in Section 4.4 and visualized in Figure 5. In order to also incorporate this uncertainty in the fit of the mortality model, we refitted the model using (a) the upper end and (b) the lower end of the prediction intervals shown in Figure 5. It appears that there is little (and hardly any visible) effect on the spatial components, which is therefore not shown here. But the time trend shown in Figure 6 does change, which is visualized by including the two fitted functions corresponding to the 2.5% and 97.5% quantile of the offset function. We can see that the estimated uncertainty of the nowcast model mostly affects the last 10 days, with a strong potential increase in the death rate mirroring a possible worst case scenario.

6.4 | Auto-correlation of residuals in the mortality model

In the mortality model (2), we did not include an epidemic component accounting for possible temporal auto-correlation, as it is often done in endemic-epidemic models (see, e.g., Meyer et al., 2017). To check for possibly omitted auto-correlation

FIGURE 8 Smooth spatial effect of the death rate in Germany



Based on data reported up to 2020-05-14.
Model includes registration dates from
2020-03-26 until 2020-05-13.

in our model, we explore the temporal correlation of the Pearson residuals in the mortality model (2). To do so, we compute the auto-correlation function (ACF) for all lags $k = 0, \dots, T - 1$. The corresponding ACF plot is shown in Figure 11. Apparently, the results do not show any pattern of auto-correlation and support the suitability of our model. We emphasize, however, that infection dynamics are included in the model through the time trend $m_1(t)$. Moreover, even if we ignore possibly existing auto-correlation, this time trend $m_1(t)$ is still estimated unbiased with penalized spline smoothing, which is robust against misspecification of the auto-correlation structure (Krivobokova & Kauermann, 2007).

We also think that the epidemic component is generally less impactful when modelling fatal infections in comparison to modelling the number of registered infections. The time between person-to-person transmission of the virus and a fatal outcome of a COVID-19 infection is much larger than the time until the registration of the infection, as shown in Figure 1, and hence any auto-correlation is rather indistinct for fatal cases.

7 | DISCUSSION

The paper presents a general approach for monitoring the dynamic behaviour of COVID-19 infections on a small-area level purely based on the analysis of the number of observed death counts. This in turn means that the results are less dependent on testing strategies, which may vary by region and over time.

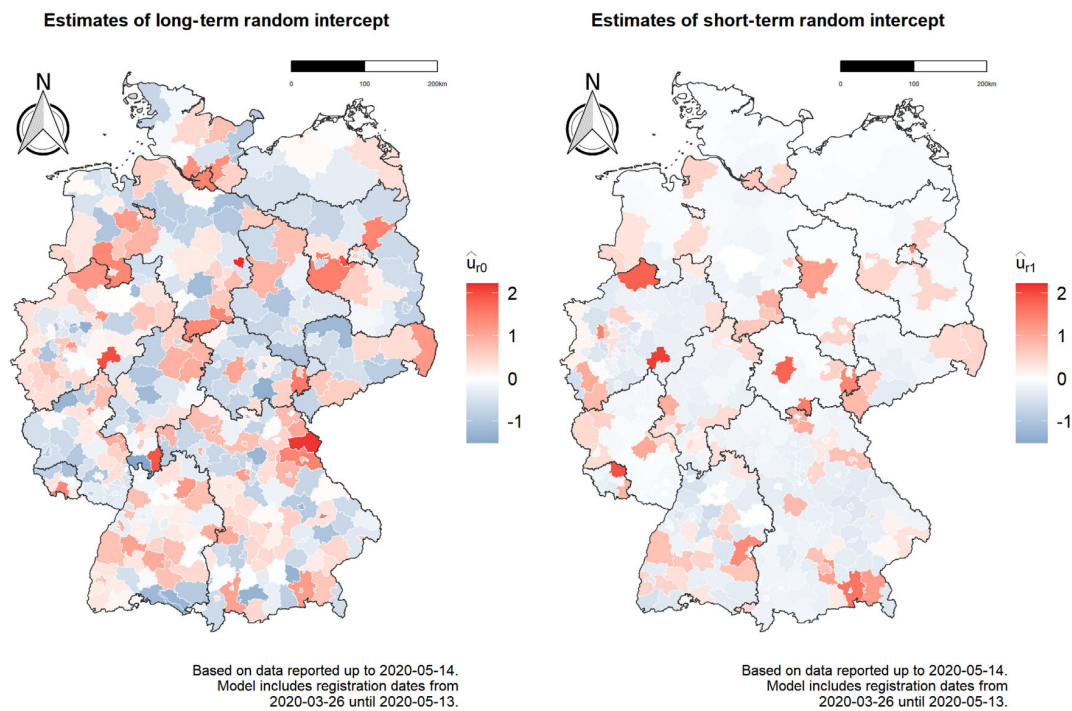


FIGURE 9 Region-specific long-term level (left-hand side) and short-term dynamics of the 14 days prior to May 14, 2020 (right-hand side) of fatal COVID-19 infections

In addition, patients with fatal infections typically require intensive medical care and are therefore relevant in the planning of clinical capacities of the local health system. An analysis of fatal infections is especially interesting in situations in which reliable information on hospitalization is not available, as in the considered timespan of the COVID-19 pandemic in Germany.

The described nowcasting approach enables us to estimate the number of deaths following a registered infection even if the fatal outcome has not occurred yet, providing an up-to-date picture of the situation. The results of the nowcasting model confirm that the individual course of the disease for fatal infections did not change over calendar time nor did it differ by gender. More in particular, it uncovers that in Germany, during the considered timespan, elderly patients had, in the case of fatal infections, about the same course of the disease as younger patients.

Our analysis of the nowcasted number of fatal infections on a regional level allows to draw conclusions on the current dynamics of the disease on the spatial dimension. By separately estimating, for each district, a long-range effect which mirrors the overall situation as well as a short-term dynamic effect, we can timely identify districts with unexpectedly high nowcasted death rates. An additional interaction for elderly people allows us to distinguish between outbreaks which might be attributed to activity in retirement homes and those due to unexpected activities in the general population. Mapping the general pattern of the spread of the disease in Germany confirms that different regions are affected to different extents, with southern and western regions being generally more affected than northern states. In addition, a global smooth time trend captures the changes in death rate, showing the peak at the beginning of April and a constant decrease since then. Thanks to the implemented nowcasting, the time trend can be estimated up to the date of analysis. This spatially differentiated picture would not be achievable through a simple monitoring of district-specific observed deaths.

A natural next step would now be to consider the nowcasted fatal infections in relation to the number of newly registered infections, which is, in contrast, highly dependent on both testing strategy and capacity. We consider this as possible future research, but the proposed model allows to explore data in this direction. This might ultimately help us in shedding light on the relationship between registered and undetected infections as well as on the effectiveness of different testing strategies.

There are several limitations to this study, which we want to address as well. First and foremost, even though death counts are, with respect to cases counts, less dependent on testing strategies, they are not completely independent from them. This applies in particular to the handling of post-mortem tests. We therefore do not claim that our analysis of death counts is completely unaffected by testing strategies. Second, a fundamental assumption in the model is the independence between

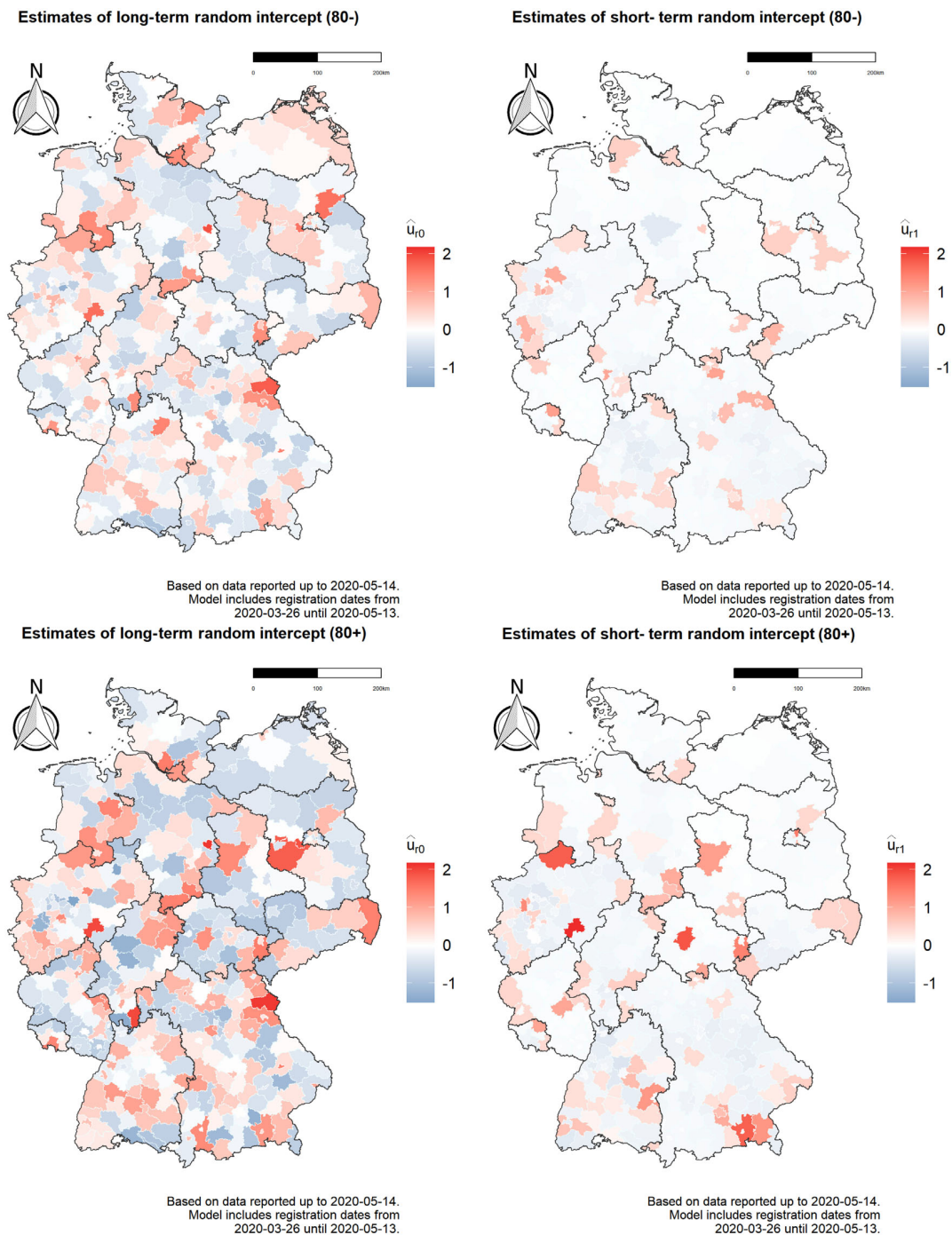


FIGURE 10 Region-specific long-term level (left-hand column) and short-term dynamics of the 14 days prior to May 14, 2020 (right-hand column) of fatal COVID-19 infections for the age groups under 80 (80–, upper row) and above 80 (80+, bottom row)

the course of the disease (on the population level) and the number of infections. Overall, if the local health systems have sufficient capacity and triage can be avoided, this assumption seems plausible, but it is difficult or even impossible to prove the assumption formally. However, the results of the nowcasting model empirically show a rather stable course of the disease, supporting our assumption. Furthermore, the registration of a COVID-19 case is related to the district of residence, while the infection does not necessarily occur in the district where the infected person resides. However, due to a lack of data we cannot explore this point further. Also, in the considered timespan, the mobility in the population has

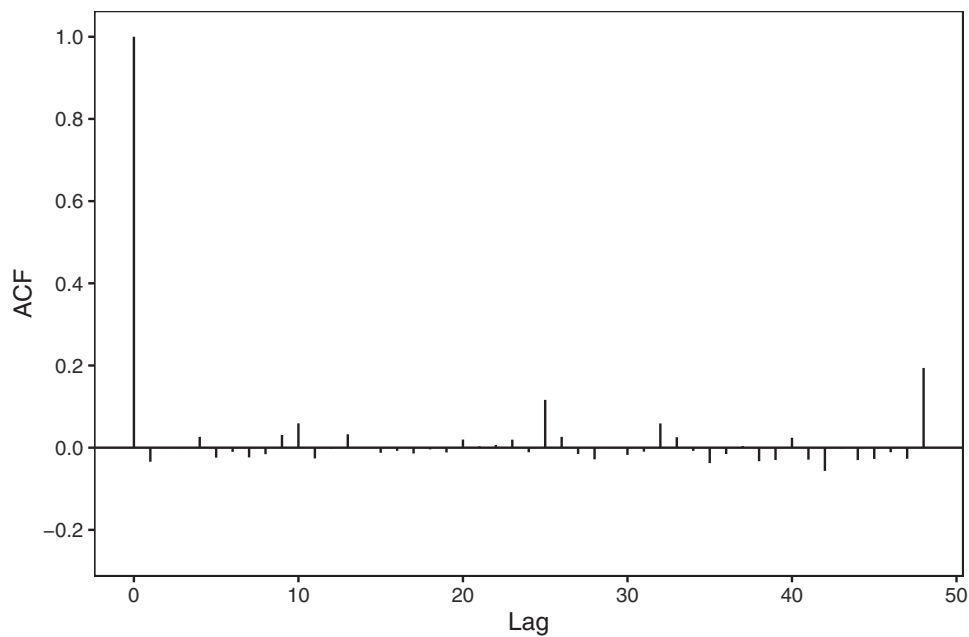


FIGURE 11 ACF plot of the Pearson residuals in the mortality model

been rather low due to governmental restrictions. Even though the model focuses on regional aspects of the pandemic, the nowcasting itself is carried out on a national level, due to sparse data. Given that our results show that the course of the disease from registration to death in Germany did not notably depend on age or gender, we do not expect it to depend on place of residence either.

A general limitation results through the availability of information. Our analyses are based on available data of all registered COVID-19 infections in Germany together with the information on fatalities, which is published daily by the RKI. While these data allow for an analysis of the occurrence of the disease in Germany, it lacks further detailed patient-specific information, for example on clinical aspects or on the differentiation between death with or because of COVID-19. This issue is shared with many other public disease registers. Note also that the methods we are proposing in this paper are not necessarily restricted to the use case of German COVID-19 data. For the purpose of applying our methodology to other countries, the data need to be in the same format as illustrated in Table 1, that is death counts need to be available in an aggregated form stratified by age (group), gender and district. For an appropriate interpretation of the results, it is critical that the reference date of every infection with a fatal outcome (here: registration date) corresponds to a time point at the early stages of the course of the disease. This could also be the date of infection with COVID-19, if known. The second date, which is needed for our nowcasting approach, is the reporting day of each fatal infection. While in Germany, this information can be deduced by considering the COVID-19 database daily over a longer period, the health authorities in other countries might supply historical reporting dates in a consecutively updated database.

Finally, the proposed approach demonstrates that valuable insight into the state and the dynamic of the disease can be obtained by disentangling spatial variation into a global pattern, district-specific long-term effects and current short-term dynamics in a spatio-temporal model. A particular virtue of the presented modelling approach over other proposals is that it also adjusts for the age and gender structure of the local population. This can provide relevant support for the monitoring of this new disease and can assist local health authorities in the planning of infection control measures as well as healthcare system capacities, in a further step towards the understanding and control of the COVID-19 pandemic.

ACKNOWLEDGEMENTS


We want to thank Maximilian Weigert and Andreas Bender for introducing us to the art of producing geographic maps with **R**. Moreover, we would like to thank all members of the **Corona Data Analysis Group (CoDAG)** at LMU Munich for fruitful discussions.

Open access funding enabled and organized by Projekt DEAL.

CONFLICT OF INTEREST

The authors have declared no conflict of interest.

OPEN RESEARCH BADGES

 This article has earned an Open Data badge for making publicly available the digitally-shareable data necessary to reproduce the reported results. The data is available in the [Supporting Information](#) section.

This article has earned an open data badge “**Reproducible Research**” for making publicly available the code necessary to reproduce the reported results. The results reported in this article could fully be reproduced.

ORCID

Marc Schneble  <https://orcid.org/0000-0001-9523-4173>

REFERENCES

- Agresti, A. (2010). *Analysis of ordinal categorical data* (2nd ed.). Hoboken, NJ: Wiley.
- Altmejd, A., Rocklöv, J., & Wallin, J. (2020). Nowcasting COVID-19 statistics reported with delay: A case-study of Sweden. Preprint arXiv:2006.06840v1.
- Baud, D., Qi, X., Nielsen-Saines, K., Musso, D., Pomar, L., & Favre, G. (2020). Real estimates of mortality following COVID-19 infection. *The Lancet. Infectious Diseases*, 20(7), 773. [https://doi.org/10.1016/S1473-3099\(20\)30195-X](https://doi.org/10.1016/S1473-3099(20)30195-X).
- Bird, S., & Nielsen, B. (2020). Now-casting of COVID-19 deaths in English hospitals. Retrieved from <http://users.ox.ac.uk/nuff0078/Covid/index.htm>.
- Cohen, J., & Kupferschmidt, K. (2020). Countries test tactics in “war” against COVID-19. *Science*, 367(6484), 1287–1288.
- Esri Deutschland GmbH (2020). Daily COVID-19 case numbers provided by the Robert-Koch-Institute. Retrieved from <https://www.arcgis.com/home/item.html?id=f10774f1c63e40168479a1feb6c7ca74>. Accessed: 30/09/2020.
- Fahrmeir, L., Kneib, T., Lang, S., & Marx, B. (2007). *Regression*. Berlin, Germany: Springer.
- Ferguson, N., Laydon, D., Nedjati-Gilani, G., Imai, N., Ainslie, K., Baguelin, M., ... Ghani, A. C. (2020). Report 9 - Impact of non-pharmaceutical interventions (NPIs) to reduce COVID-19 mortality and healthcare demand. London: Imperial College London. <https://doi.org/10.25561/77482>.
- Flaxman, S., Mishra, S., Gandy, A., Unwin, H., Coupland, H., Mellan, T., ... Bhatt, S. (2020). Report 13: Estimating the number of infections and the impact of non-pharmaceutical interventions on COVID-19 in 11 European countries. Retrieved from <https://spiral.imperial.ac.uk/8443/handle/10044/1/77731>.
- Grasselli, G., Pesenti, A., & Cecconi, M. (2020). Critical care utilization for the COVID-19 outbreak in Lombardy, Italy: Early experience and forecast during an emergency response. *JAMA*, 323(16), 1545–1546.
- Grasselli, G., Zangrillo, A., & Zanella, A. (2020). Baseline characteristics and outcomes of 1591 patients infected with SARS-CoV-2. *JAMA*, 323(16), 1574–1581.
- Günther, F., Bender, A., Katz, K., Küchenhoff, H., & Höhle, M. (2020). Nowcasting the COVID-19 pandemic in Bavaria. *Biometrical Journal*, 1–13. <https://doi.org/10.1002/bimj.202000112>.
- Held, L., Meyer, S., & Bracher, J. (2017). Probabilistic forecasting in infectious disease epidemiology: the 13th Armitage lecture. *Statistics in Medicine*, 36(22), 3443–3460.
- Höhle, M., & an der Heiden, M. (2014). Bayesian nowcasting during the STEC O104:H4 outbreak in Germany, 2011. *Biometrics*, 70, 993–1002.
- Jung, S.-M., Akhmetzhanov, A., Hayashi, K., Linton, N., Yang, Y., Yuan, B., ... Nishiura, H. (2020). Real-time estimation of the risk of death from novel coronavirus (COVID-19) infection: Inference using exported cases. *Journal of Clinical Medicine*, 9, 523.
- Krivobokova, T., & Kauermann, G. (2007). A note on penalized spline smoothing with correlated errors. *Journal of the American Statistical Association*, 102(480), 1328–1337.
- Lawless, J. (1994). Adjustment for reporting delays and the prediction of occurred but not reported events. *Canadian Journal of Statistics*, 22(1), 15–31.
- Li, R., Pei, S., Chen, B., Song, Y., Zhang, T., Yang, W., & Shaman, J. (2020). Substantial undocumented infection facilitates the rapid dissemination of novel coronavirus (SARS-CoV-2). *Science*, 368(6490), 489–493.
- Linton, N. M., Kobayashi, T., Yang, Y., Hayashi, K., Akhmetzhanov, A. R., Jung, S.-m., ... Nishiura, H. (2020). Incubation period and other epidemiological characteristics of 2019 novel coronavirus infections with right truncation: A statistical analysis of publicly available case data. *Journal of Clinical Medicine*, 9(2), 538.
- Massonnaud, C., Roux, J., & Crépey, P. (2020). COVID-19: Forecasting short term hospital needs in France. *medRxiv 2020.03.16.20036939*. <https://doi.org/10.1101/2020.03.16.20036939>.
- Meyer, S., Held, L., & Höhle, M. (2017). Spatio-temporal analysis of epidemic phenomena using the R package surveillance. *Journal of Statistical Software, Articles*, 77(11), 1–55.
- Niehus, R., De Salazar, P. M., Taylor, A., & Lipsitch, M. (2020). Quantifying bias of COVID-19 prevalence and severity estimates in Wuhan, China that depend on reported cases in international travelers. *medRxiv 2020.02.13.20022707*. <https://doi.org/10.1101/2020.02.13.20022707>.

- Robert Koch-Institut (2020). COVID-19-dashboard. Retrieved from <https://experience.arcgis.com/experience/478220a4c454480e823b17327b2bfd4>.
- StaBLab, LMU Munich (2020). CoronaMaps. Retrieved from <https://corona.stat.uni-muenchen.de/maps/>.
- Streeck, H., Schulte, B., Kümmerer, B. M., Richter, E., Höller, T., Fuhrmann, C., ... Hartmann, G. (2020). Infection fatality rate of SARS-CoV-2 infection in a German community with a super-spreading event. *medRxiv* 2020.05.04.20090076. <https://doi.org/10.1101/2020.05.04.20090076>.
- Wand, M. (2003). Smoothing and mixed models. *Computational Statistics*, 18(2), 223–249.
- Wilson, N., Kvalsvig, A., Barnard, L. T., & Baker, M. G. (2020). Case-fatality risk estimates for COVID-19 calculated by using a lag time for fatality. *Emerging Infectious Diseases*, 20(6), 1339–1441.
- Wood, S. N. (2017). *Generalized additive models: an introduction with R*. Boca Raton, FL: CRC Press.
- Zeger, S. L., See, L. C., & Diggle, P. J. (1989). Statistical methods for monitoring the AIDS epidemic. *Statistics in Medicine*, 8, 3–21.

SUPPORTING INFORMATION

Additional supporting information may be found online in the Supporting Information section at the end of the article.

How to cite this article: Schneble M, De Nicola G, Kauermann G, Berger U. Nowcasting fatal COVID-19 infections on a regional level in Germany. *Biometrical Journal*. 2021;63:471–489. <https://doi.org/10.1002/bimj.202000143>

Chapter 6

A statistical model for the dynamics of COVID-19 infections and their case detection ratio in 2020

Contributing Article Schneble, M., De Nicola, G., Kauermann, G., Berger, U. (2021). A statistical model for the dynamics of COVID-19 infections and their case detection ratio in 2020. *Biometrical Journal*. 1-10. <https://doi.org/10.1002/bimj.202100125>

Code and data <https://doi.org/10.1002/bimj.202100125>

Copyright 2020 The Authors. *Biometrical Journal* published by Wiley-VCH GmbH. This is an open access article under the terms of the Creative Commons Attribution License (<https://creativecommons.org/licenses/by/4.0/>), which permits use, distribution and reproduction in any medium, provided the original work is properly cited and is not used for commercial purposes.

Author Contributions The idea of relating registered COVID-19 infections and deaths related to COVID-19 can be attributed to Göran Kauermann. He also formulated the model to estimate the change of the case detection ratio over time. Furthermore, Marc Schneble was involved in elaborating the methodology presented in the paper. The major parts of the paper were written by Göran Kauermann and Marc Schneble, where Section 2, Section 4 and the supplementary material were mainly written by Marc Schneble. Besides, Giacomo De Nicola and Ursula Berger contributed especially to Section 1 and Section 5. All authors were involved in extensive proof-reading of the manuscript. Marc Schneble was responsible for implementating the model in **R** and for the visualization of the results.

A statistical model for the dynamics of COVID-19 infections and their case detection ratio in 2020

Marc Schneble¹  | Giacomo De Nicola¹ | Göran Kauermann¹ | Ursula Berger²

¹ Department of Statistics, LMU Munich, Munich, Germany

² Institute for Medical Information Processing, Biometry and Epidemiology, LMU Munich, Munich, Germany

Correspondence

Marc Schneble, Department of Statistics, LMU Munich, Ludwigstr. 33, 80539 Munich, Germany.

Email:

marc.schneble@stat.uni-muenchen.de



This article has earned an open data badge “**Reproducible Research**” for making publicly available the code necessary to reproduce the reported results. The results reported in this article could fully be reproduced.

Abstract

The case detection ratio of coronavirus disease 2019 (COVID-19) infections varies over time due to changing testing capacities, different testing strategies, and the evolving underlying number of infections itself. This note shows a way of quantifying these dynamics by jointly modeling the reported number of detected COVID-19 infections with nonfatal and fatal outcomes. The proposed methodology also allows to explore the temporal development of the actual number of infections, both detected and undetected, thereby shedding light on the infection dynamics. We exemplify our approach by analyzing German data from 2020, making only use of data available since the beginning of the pandemic. Our modeling approach can be used to quantify the effect of different testing strategies, visualize the dynamics in the case detection ratio over time, and obtain information about the underlying true infection numbers, thus enabling us to get a clearer picture of the course of the COVID-19 pandemic in 2020.

KEYWORDS

case detection ratio, COVID-19, dark figure of infections, generalized additive models, penalized splines

1 | INTRODUCTION

Originating from Wuhan, China, coronavirus disease 2019 (COVID-19) developed to become a worldwide pandemic in the spring of 2020 (Velavan & Meyer, 2020). Starting from the very beginning of this unprecedented health crisis, the issue of case detection, while always being at the center of scientific and public discourse, has been all but transparent. Knowing how many infections are really present in the population would be of paramount importance, and researchers have tried to tackle the problem in several different ways. Early in the epidemic wave, the ratio of undetected COVID-19 cases was likely to be high, that is, 5–20 times higher than the number of confirmed cases (e.g., Li et al., 2020 or Wu et al., 2020). The problem of discovering the case detection ratio (CDR) is tightly intertwined with the issue of uncovering the true fatality ratio of the disease, as knowledge on one of those two unknown quantities would provide information about the other. A natural experiment that allowed to obtain initial estimates of both the fatality ratio and the CDR occurred with the outbreak on the cruise ship “Diamond Princess” (Mizumoto et al., 2020). During the early stages of the pandemic, the actual percentage of the population infected for 11 European countries was deduced from early estimates of the mortality

This is an open access article under the terms of the [Creative Commons Attribution-NonCommercial License](https://creativecommons.org/licenses/by-nc/4.0/), which permits use, distribution and reproduction in any medium, provided the original work is properly cited and is not used for commercial purposes.

© 2020 The Authors. *Biometrical Journal* published by Wiley-VCH GmbH.

rates (Flaxman et al., 2020). Moreover, Aspelund et al. (2020) used Bayes arguments applied to testing data from Ireland to estimate the CDR in the order of 7–11% at the beginning of the pandemic, and in the order of 10–20% after that. The argument is based on relating the number of tests and the share of positive tests. A similar approach has been pursued making use of Canadian data (Benatia et al., 2020). The problem of estimating the true numbers of COVID-19 infections has also been discussed from a purely statistical point of view, where the CDR was related to the fatality ratio (Manski & Molinari, 2020). A capture–recapture approach to estimate the total number of COVID-19 cases was proposed by Böhning et al. (2020) and Rocchetti et al. (2020), where the latter derive an upper bound for the cumulative number in mid-April for 10 European countries. The ratio of the upper bound and the observed number of cases ranges from around 4 (Greece) to around 8 (France). The capture–recapture method makes only use of publicly available data on COVID-19 cases and deaths, which also holds for the method that we present in this note. Here, we assume that the number of infected can be split into detected and undetected infections. In SIDARTHE models (Giordano et al., 2020), there is additional distinction into either asymptomatic or symptomatic cases, which we ignore here since the database that we use does not reliably contain these numbers. However, it should be noted that pre- and asymptomatic individuals have a significant impact on the spread of a pandemic disease, especially in the younger population (Stella et al., 2020). Thereby, presymptomatic individuals play a more significant role than asymptomatic ones (Buitrago-Garcia et al., 2020). Nonetheless, the number of asymptomatic cases can reduce the reproduction value of a disease because a background immunity is established, as shown for influenza transmission (Mathews et al., 2007).

Overall, underreporting appears to be an overarching problem, which plays a central role when estimating the CDR for COVID-19 (Russell et al., 2020). The importance of assessing the detection ratio and its effect on predictions of future infections has been demonstrated in mathematical simulation studies (Fuhrmann & Barbarossa, 2020). In this context, different national underreporting ratios have been compared (e.g., Rahmandad et al., 2020 or Jagodnik et al., 2020) and a general discussion and survey on assessing the infection fatality ratio (IFR) was conducted (Levin et al., 2020). In general, it is clear that the CDR changes greatly over time depending on testing strategy and capacities, which vary over time and across different regions. In Germany, the number of tests has increased considerably since the pandemic outbreak in March 2020. The testing strategy has also been adjusted several times: In the beginning, mainly individuals with symptoms were being tested, whereas in later phases, a very high number of tests have been performed on travelers returning from foreign countries and contact persons of COVID-19-positive individuals.

In this note, we explore the dynamics in the CDR using publicly available registry data on COVID-19 infections in Germany from March to December 2020 provided by the Robert-Koch-Institute (RKI). It is important to mention that in Germany's first months of the pandemic, no mass or systematic testing of the population had taken place. Our model therefore only makes use of a limited amount of information. We propose to jointly model fatal and nonfatal infections using a dynamic generalized linear mixed model with smooth random effects (see, e.g., Durbán et al., 2005; Durban & Aguilera-Morillo, 2017; Wood, 2017). The major advantage of our approach is that it only relies on the assumption that age-specific COVID-19 fatality ratios, while unknown, have not substantially changed over time. Whether this assumption is valid is currently discussed (Harris, 2020; Kip et al., 2020) and the possibility of differing fatality ratios in the second wave has been considered as well (Aspelund et al., 2020; Kenyon, 2020). To assess the impact of this assumption on our results, we provide sensitivity analyses and a simulation study in the Supporting Information, which demonstrate that our approach is sufficiently robust if there is no abrupt change in the infection fatality ratio.

Overall, our approach allows investigating the following. First, we explore how the case detection rate has changed over time, how it varies among different age groups, and if and how it changes in different regions of Germany, depending on infection dynamics and different testing strategies. Second, the model also provides an estimate of the dynamics in the true number of infections, regardless of whether they have been detected or not. All in all, this provides insight into the course of the COVID-19 pandemic, built exclusively on registry data.

The remainder of the paper is structured as follows. We describe the data constellation in depth in Section 2, and we propose our model in Section 3. In Section 4, we show the results of our analyses and provide extensive interpretations, whereas Section 5 concludes the paper with some implications and limitations of our study.

2 | DATA

We make use of COVID-19 data openly provided by the RKI, the German federal government agency and scientific institute responsible for health reporting, disease control, and prevention in humans (Esri Deutschland GmbH, 2020). The data, exemplified in Table 1, contain cumulated counts of newly registered, laboratory-confirmed COVID-19 cases in Germany

TABLE 1 Illustration of the data structure. To facilitate reproducibility, the original column names used in the RKI dataset are given in brackets below our English notation

District (Landkreis)	Age group (Altersgruppe)	Gender (Geschlecht)	Cases (Anzahl Fall)	Deaths (Anzahl Todesfall)	Registration date (Meldedatum)
⋮	⋮	⋮	⋮	⋮	⋮
Munich City	60–79	F	26	0	September 8, 2020
Munich City	60–79	M	21	1	September 8, 2020
⋮	⋮	⋮	⋮	⋮	⋮

for each calendar day stratified by age group (0–4, 5–14, 15–34, 35–59, 60–79, or 80+ years), gender (male/female), and district (412 in total). Furthermore, for all registration dates and strata, the number of deaths associated with COVID-19 transmitted to the RKI by the local health authorities of the respective district is recorded. Note that the date of death is not provided, but for each death, we have the date when the infection was detected and confirmed by a (PCR) test. The database of the RKI is updated every morning with the new numbers transmitted to it from the local health authorities.

In this study, we only consider data entries with registration dates ranging from calendar week (CW) 10 (mid-March) to CW 53 (end of December) of the year 2020. For earlier weeks, the number of tests being positive was not large enough to draw conclusive results. On the other hand, the German vaccination campaign started at the very end of 2020. As this increasingly reduces the IFR, we only include infections that were registered in 2020. Consequently, the final outcome of almost all of these infections is known today. Moreover, although the data are given on a daily resolution, we here aggregate it into weekly data, which renders reporting delays occurring over the weekends and weekly reporting cycles irrelevant to our analysis, leading to more stable results. Since for children aged 14 years and younger, barely, any fatalities have been recorded, we excluded these age groups from our analysis.

To give a first insight into the data at hand, we plot in Figure 1 the raw numbers of cases reported by the official health authorities over time together with the raw number of fatalities stratified by age group. This is shown in the top four plots on a log-scale. Both the number of registered cases and that of fatal cases (indexed by registration date of the infection, and not by day of death) peak in CW 13 for the two younger age groups and in CW 14 for the two oldest age groups, respectively. Over the following weeks, these numbers decrease. The small peak in CW 25 was caused by an outbreak in the district of Gütersloh, which is explored in more depth later on in the paper. From CW 28 onward, we resume seeing an exponential increase of registered cases, whereas the numbers of registered fatal cases only start to rise 7 weeks later, also exponentially. By the end of the year 2020, we see a slight decrease in registered infections.

The raw case fatality ratio, calculated as the ratio of fatal cases over total registered cases, stratified by age group, is shown at the bottom of Figure 1. The raw case fatality ratio for the age group 80+ generally dropped from CW 10 onward and fluctuated mostly between 10% and 15% from week 25 onward. However, since CW 40 the case fatality ratio in this age group steadily climbed up to more than 20%. For the age group 60–79, the case fatality ratio has peaked in CW 16 and gradually decreased to 2.5%. Here, we also observe a steady increase toward the end of 2020, which results in more than a doubling of the case fatality ratio within 10 weeks. All other age groups exhibit relatively low raw case fatality ratios throughout.

Note that the raw data do not contain undetected cases, and therefore cannot provide a complete picture of the actual infection numbers, nor do these plots provide any information about the CDR. In the following, we develop a statistical model that enables us to estimate the relative changes in the CDR and the true infection numbers over time.

3 | METHODS

When describing the dynamics of the COVID-19 pandemic, the number of interest is the true count of newly infected persons in a cohort, which shall be denoted by I_t for week $t = 1, \dots, T$. Note that I_t remains unobservable. However, the number can be decomposed into the number of detected and reported cases D_t and the unknown number of newly infected persons, who have not been tested and remain undetected, which we can call the “dark number,” U_t . Hence, we have $I_t = D_t + U_t$, and D_t/I_t defines the CDR, which, however, remains unknown due to U_t being unknown.

Note that the index t indicates the time point on which the infection took place, which is usually unknown. The infection is eventually detected through a positive test at a later time point $\tilde{t} = t + d$. As d is often unknown, in particular, if the spread of the disease is diffuse, we will conceptually omit d in the following, which means that we set t equal to the

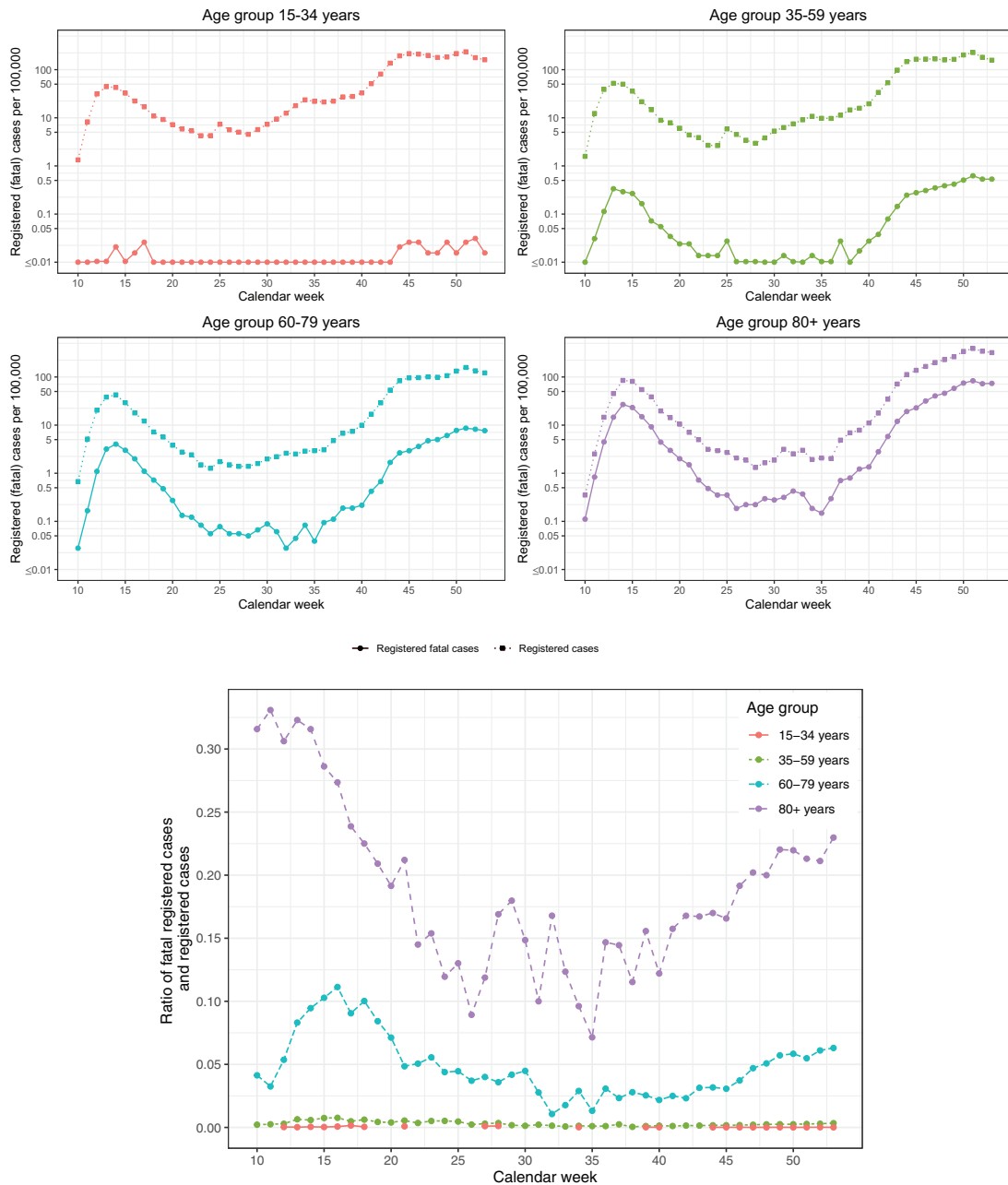


FIGURE 1 Raw data: registered cases of COVID-19 infections and registered fatal cases on a weekly basis for Germany. Top figure: Absolute numbers on a log-scale stratified by age group. Bottom figure: Case fatality ratios (= fatal cases / registered cases) stratified by age group

registration date when an infection is confirmed through a test. This time point is the registration date described in the previous section. Generally, this approach is justifiable for COVID-19 infections because the range of delay d is small compared to the time range T of our data analysis (Mallett et al., 2020).

From today’s perspective, we have uncensored knowledge on the outcomes of all reported cases D_t . That is, we know if they ended fatally or if they recovered. Consequently, the reported cases are composed of recovered (nonfatal) outcomes R_t and fatal outcomes F_t , that is, $D_t = R_t + F_t$. Given this, the total number of infected persons splits into $I_t = R_t + F_t + U_t$.

The expected number of reported fatal cases F_t as well as the expected number of recovered cases R_t are fractions of the total number of infections I_t . This leads to

$$\mathbb{E}(F_t | I_t) = I_t a \text{ and } \mathbb{E}(R_t | I_t) = I_t c_t, \tag{1}$$

where $0 < (a + c_t) < 1$. Here, quantity a defines the infection fatality ratio (IFR), whereas c_t is the CDR of nonfatal (recovered) infections. Note that these nonfatal infections also include mild and symptom-free cases. Thus, if testing capacities are increased or the testing strategy is changed, c_t will change as well, which is incorporated in the notation by time index t . In contrast, the IFR a will be assumed to remain constant over time. This can be justified by the fact that fatal cases, due to their severeness, are likely to be detected independently of any testing policy. This also includes, to some extent, postmortem tests.

With this notation, we obtain the time-dependent case detection ratio $CDR_t = a + c_t$. Note that for the dark number, that is, the latent number of undetected infections U_t , it holds that $\mathbb{E}(U_t | I_t) = (1 - CDR_t)I_t$. It would, of course, be favorable to estimate the number of undetected infections U_t via estimation of a and c_t . However, when only the reported fatal and nonfatal cases F_t and R_t are known, these two ratios cannot be estimated due to nonidentifiability issues, which we will demonstrate below. Nonetheless, with the data at hand, we are able to estimate the ratio c_t/a . To see this, we rewrite the above model in an equivalent form by defining a binary covariate $x \in \{0, 1\}$ and by specifying the response variable Y_t through

$$Y_t | x = \begin{cases} F_t & \text{for } x = 0 \\ R_t & \text{for } x = 1. \end{cases}$$

This notational trick allows us to rewrite the above relations (1) as a regression model

$$\mathbb{E}(Y_t | I_t, x = 0) = \mathbb{E}(F_t | I_t) = \exp\{\log(I_t a)\} = \exp\{V_t + \alpha\}, \quad (2)$$

$$\mathbb{E}(Y_t | I_t, x = 1) = \mathbb{E}(R_t | I_t) = \exp\{V_t + \gamma_t\}, \quad (3)$$

where $V_t = \log(I_t)$, $\alpha = \log(a)$, and $\gamma_t = \log(c_t)$. Equations (2) and (3) can, in turn, be summarized into a single regression model formula

$$\mathbb{E}(Y_t | V_t, x) = \exp\{V_t + \alpha + x(\gamma_t - \alpha)\}. \quad (4)$$

Note that I_t and hence $V_t = \log(I_t)$ remain unobserved. We employ a Bayesian view and model V_t as normally distributed random effects $V_t \sim N(\mu_t, \sigma^2)$. Still, the parameters in model (4) are not identifiable, because any shift in μ_t and a matching negative shift in α and γ_t , respectively, results in the same model. This demonstrates the identifiability problem, which we have mentioned above. Hence, we are neither able to estimate the fatality ratio $a = \exp(\alpha)$ nor the time-dependent ratio $c_t = \exp(\gamma_t)$ with the data at hand. However, we can shift μ_t such that the integral of $\tilde{\mu}_t = \mu_t - k$ is equal to zero and define the global intercept $\beta_0 = \alpha + k$, which allows to rewrite (4) in an identifiable form (see Wood, 2017) to obtain the final regression model

$$\mathbb{E}(Y_t | V_t, x) = \exp(V_t + \beta_0 + x\beta_t) \text{ and } V_t \sim N(\tilde{\mu}_t, \sigma^2) \text{ for } t = 1, \dots, T, \quad (5)$$

where $\beta_t = \gamma_t - \alpha$ and $\exp(\beta_t) = c_t/a$. With this model, we can now explore the dynamics in the CDR. For two different time points t_1 and t_2 , we have using the small $o()$ notation

$$\frac{CDR_{t_2}}{CDR_{t_1}} = \frac{c_{t_2} + a}{c_{t_1} + a} = \frac{c_{t_2}}{c_{t_1}} \{1 + o(a)\} = \frac{\exp(\beta_{t_2})}{\exp(\beta_{t_1})} \{1 + o(a)\} \approx \frac{\exp(\beta_{t_2})}{\exp(\beta_{t_1})}. \quad (6)$$

The latter approximation in (6) holds as long as the fatality rate a is small, which holds for COVID-19. Consequently, $\beta_{t_2} - \beta_{t_1}$ can serve as a proxy for $\log(CDR_{t_2}) - \log(CDR_{t_1})$, and $\exp(\beta_{t_2} - \beta_{t_1})$ is a proxy for the relative change in the case detection ratio CDR_{t_2}/CDR_{t_1} .

Based on these considerations, we see that it is necessary to model the dynamics in time t more appropriately to derive stable estimates for the CDR. It is natural to assume that changes in the CDR over time do not occur suddenly but gradually. For instance, test capacities are slowly increased and test strategies are gradually changed. To accommodate this in our model (5), we fit β_t by a smooth function in time leading to a time-varying coefficient model (Hastie & Tibshirani, 1993). We also induce smooth dynamics on the random component, leading to a time-varying random effect (Durban &

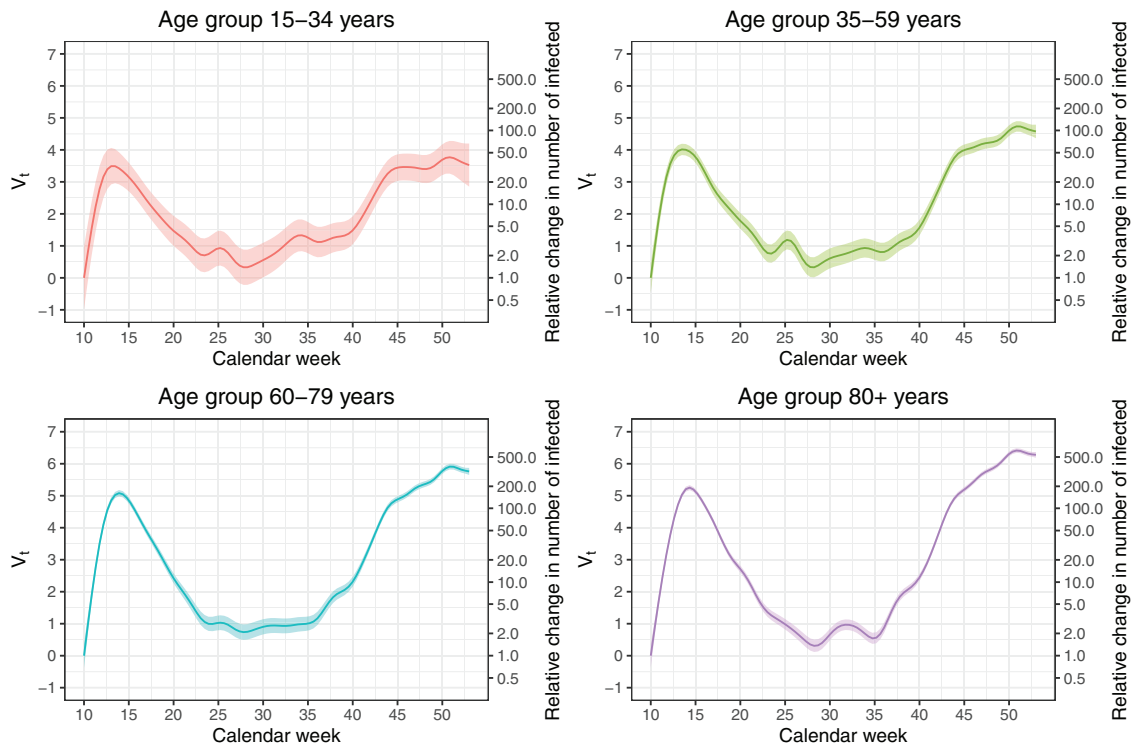


FIGURE 2 Dynamics of the true infection numbers on the log-scale for different age groups: The smooth random effects V_t . The shaded areas represent 95% confidence bands

Aguilera-Morillo, 2017). These modifications lead to an identifiable and dynamic mixed regression model, for which we use a negative-binomial distribution for Y_t with a constant dispersion factor. The entire model can be fitted with standard software: All of our analyses were performed in **R** (R Core Team, 2013) and the dynamic mixed regression model is fitted using the **R**-package **mgcv** (Wood, 2017).

We apply this modeling approach using the reported data from CW 10 (beginning of March) up to CW 53 (final week of 2020), stratified by different age groups, to visualize the dynamics in the real infection numbers and the CDR from the beginning of the pandemic up to the beginning of the second wave. To assess the robustness of the approach concerning the assumption of time-constant and age-specific fatality ratios, we also refit the model when subdividing the data into different time frames. The results of this analysis are shown in the Supporting Information.

4 | RESULTS

4.1 | Model estimates

As the IFR a depends on age, we fit separate models for each of the relevant age groups defined by the RKI, that is, 15–34, 35–59, 60–79, and 80+ years. The dynamics in the true infection numbers on the log-scale, represented by the fitted smooth dynamic random effects V_t , are displayed in Figure 2. These curves mirror the relative change in the actual number of infected (detected and undetected) over time. Note that the absolute numbers cannot be interpreted on their own due to the mentioned identifiability issues. We therefore shift the curves such that $V_{CW10} = 0$. We can see that the relative course of the pandemic was very similar across all age groups, where a peak is reached around CW 14. However, the peak for the younger age groups is estimated to be around 1 week earlier than for the older age groups, that is, in CW 13. An explanation for this finding is that the younger age groups have been more affected by the lockdown, which started in Germany in CW 12. Looking at the difference between the maximum $\max_t V_t$ and the minimum of V_t during the summer months, that is, $\min_{20 \leq t \leq 40} V_t$, we see that this difference increases with age, that is, the relative decline in true infections numbers after the first wave and the relative increase toward the second wave, respectively, was less pronounced in the younger age groups. Also eye-catching is the increase in infections around CW 25 for people below 60 years of age. This is

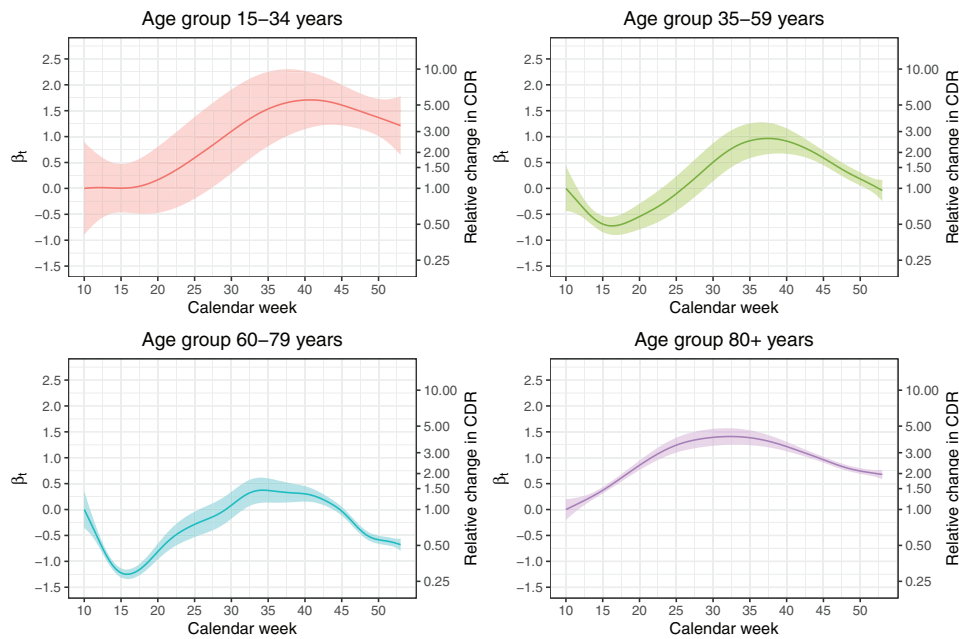


FIGURE 3 Dynamics in the case-detection ratio for different age groups: The normalized time-varying coefficients β_t . The function values on the exp-scale (right y-axes) are the relative change in the case-detection ratio (CDR) with respect to calendar week 10

the aforementioned outbreak in the district of Gütersloh, which occurred in an industrial slaughterhouse and has mainly affected people of the working age. From CW 35 (end of August), all curves start rising steadily, where the steepest rise is seen for the oldest age group, whereas the rise is flatter for the younger age group. This shows that the second wave of the pandemic had already begun around CW 35. Moreover, Figure 2 shows that in all age groups but the youngest one, the peak of the second wave has surpassed the peak of the first wave.

Next, we look at the dynamics in the CDR. Figure 3 shows the fitted time-varying coefficients β_t together with corresponding 95% confidence bands. Again, the absolute level is not identifiable, so these curves are normalized such that $\beta_{CW10} = 0$. Hence, the function values on the exp-scale (right y-axes) give the relative change in the CDR with respect to CW 10. The CDR in the age group 80+ has risen monotonically since the beginning of the pandemic up to CW 33, where our model estimates the CDR to be more than four times higher as in mid-March. Note that in later weeks, the CDR among the elderly decreased again to the level of April/May. In contrast, for people aged 60–79, the CDR first dropped by about 70%, reaching its bottom as the pandemic passed its peak in Germany in CW 16. We subsequently see a monotonic increase, with the CDR becoming 1.5 times higher compared to the beginning of the pandemic. However, in this age group, the CDR has been more than halved from CW 40 up to the end of 2020 again. The dynamics in the CDR in the population aged 35–59 years are similar to those of the 60–79 years old: After a drop during March and April (CW 10–CW 16), the CDR increases, in mid-September, to nearly three times what it was in CW 10. For the youngest age group (aged 15–34), we also see a rise in the CDR over time, which seems substantial. However, the confidence bands in this age group are relatively wide because this age group is not as prone to fatal outcomes as older age groups.

4.2 | Interpretations

For the population aged 80 years and older, the CDR had increased until late summer, when it started to stagnate before slightly decreasing again. As the CDR can be at most 100%, and given that the relative change in this age group was about as high as a factor of 4 in CW 33 compared to March, we can conclude that at the beginning of the pandemic, the CDR among the population of 80 years and older could not have been more than 25%. Moreover, considering the relative change in the CDR, we can adjust the numbers from the peak in the first wave to be comparable, for example, to the numbers in week 40. To exemplify this, note that in week 40, the CDR for the age group 80+ was 2.3 times higher as in CW 15, at the peak of the first wave. This ratio results from the plot in Figure 3 (bottom right) by taking $\beta_{CW15} = 0.4$ and $\beta_{CW40} = 1.25$ and calculating the ratio $\exp(1.25 - 0.4) = 2.3$. In week 40, we had about 11 new infections per week per 100,000 reported

in this age group. In CW 15, this number had become 80. However, in week 15, the CDR was much lower as in CW 40, and thus, we would have seen $2.3 \cdot 80 = 184$ cases per 100,000 in this age group 80+ if we had the same CDR in CW 15 as in CW 40.

For the population aged 60–79 years, the CDR between the minimum in CW 16 and its maximum in calendar week 34 changed by a factor of around 5. From this, we can deduce that around the peak of the first wave in Germany, at most 20% of the infections were detected, whereas at least 80% remained unseen. To be able to compare numbers from the first wave to those in autumn, we apply a similar calculation as above. This results in an estimated number of at least $5 \cdot 17 = 85$ cases per 100,000, where only 16 cases per 100,000 have been observed in CW 16.

In the age group 35–59, the relative change of CDR during the minimum in CW 16 and the maximum in CW 36 was as high as a factor of 5 as well. Again, the same calculation shows that the 22 detected infections per 100,000 in week 16 would increase to $5 \cdot 22 = 110$ cases per 100,000 if we would have had the CDR in week 16 as it was in week 36.

A general question in the pandemic is whether extensive testing leads to a high CDR. Applying our model to regional data allows us to investigate this question. The Supporting Information compares separate model fits for the two most populous German states, North-Rhine-Westphalia and Bavaria. The two states implemented different testing strategies over the summer months. Although in Bavaria, public test stations were opened in summer, particularly at the borders on the motorways, such fine screening of holiday returnees was not pursued in North-Rhine-Westphalia. Our model allows assessing and, in particular, quantifying how such different testing strategies lead to different CDRs in these two regions. The results quantify by how much the dark figure was reduced in relationship with the Bavarian testing strategy.

5 | DISCUSSION

Raw reported case numbers and measures derived from them, such as the case fatality ratio, are prone to changes in testing strategies and test capacities, which also influence the CDR. Comparisons between raw case numbers over time therefore need to be interpreted with care. The case-fatality ratio, calculated from the raw number of reported deaths related to COVID-19 divided by reported cases, is also impaired because deaths occur with a time delay after registration, meaning that deaths registered today correspond to infections that have been reported up to several weeks ago. Our method allows us to uncover relative changes in the CDR over different pandemic phases. Moreover, by shedding light on the number of undetected cases, we can describe the dynamics in the true number of COVID-19 infections for Germany from March 2020 until December 2020. The approach is based on publicly available data on registered cases and does not rely on simulations or additional survey data. We make use of the fact that, for each fatal outcome, the registration date of the infection is included in the data. This allows us to jointly model the number of registered nonfatal cases and that of fatal infections in a dynamic mixed model, leading to an assessment of the dynamics taking place in real infection numbers. Based on the available information on the relative change in the CDR over time, we are able to compare numbers from the first wave of the pandemic in spring with numbers from the second wave in autumn, adjusting for the difference in the proportion of undetected cases.

A general limitation of our approach is that it suffers from an identifiability issue and hence does not derive absolute values of the CDR. One may, however, combine our results with findings from seroepidemiological studies, which aim to assess the prevalence of COVID-19 in the general population by screening a representative sample. A list of current seroepidemiological studies in Germany is provided by the RKI (Robert-Koch-Institute, 2020). Although these studies provide crucial information on the current situation of the spread of the disease, they can only give a snapshot of the instantaneous situation when the study was conducted. With the knowledge of the dynamics in new infections given by our approach, the findings of such studies can be used to estimate the situation at other time points. For example, we look at the Prospective Covid-19 Cohort Study Munich (KoCo19, Radon et al., 2020). They report a CDR of about 25%, where the survey was run between May and June 2020 in the city of Munich. We can deduce that the CDR for October to be about three times higher for the 35–59 age group. More precise calculations would require age-specific numbers in the study as well as a regional refit of our model. A nationwide seroprevalence study was conducted between the beginning of July and mid-August of 2020, which yielded a CDR of around 55% in the adult population (ifo Institut & forsa, 2020). Nonetheless, the authors admit that the fading of COVID-19 antibodies could influence their findings sometime after the infection. A seroprevalence study, which is also nationwide but on a larger scale, is currently being carried out, but the results are not yet available.¹ In principle, however, this demonstrates that the combination of seroepidemiological studies and our approach allows obtaining estimates for absolute numbers of the CDR instead of relative comparisons only.

¹ https://www.rki.de/DE/Content/Gesundheitsmonitoring/Studien/lid/lid_node.html;jsessionid=02C6FAB6F407B92315BDA5C1650F4D3A.internet158

A critical assumption of our model is that we assume the IFR a to be constant over time for a given age group and negligibly small compared to the detection ratio of nonfatal cases. The latter is certainly valid for the numbers we looked at. Staerk et al. (2021) show that most of the dynamics in the effective IFR of the German population can be explained by the varying age distribution of COVID-19 cases. As the age distribution within the RKI age categories varies as well, the IFR a within each age group might slightly change over time that, however, occurs not abruptly but smoothly over time. The sensitivity analysis, which can be found in the Supporting Information, provides evidence that our assumption of a being constant is, for the most part, fulfilled. With increasing vaccination levels in the population starting from January 2021, the assumption of a constant case fatality ratio becomes invalid. This eventually prevents the application of our model to later stages of the pandemic.


CONFLICT OF INTEREST

The authors have declared no conflict of interest.

DATA AVAILABILITY STATEMENT

The data that support the findings of this study are openly available at <https://www.arcgis.com/home/item.html?id=f10774f1c63e40168479a1feb6c7ca74>.

OPEN RESEARCH BADGES

 This article has earned an Open Data badge for making publicly available the digitally-shareable data necessary to reproduce the reported results. The data is available in the [Supporting Information](#) section.

This article has earned an open data badge “**Reproducible Research**” for making publicly available the code necessary to reproduce the reported results. The results reported in this article could fully be reproduced.

ORCID

Marc Schneble  <https://orcid.org/0000-0001-9523-4173>

REFERENCES

- Aspelund, K., Droste, M., Stock, J. H. & Walker, C. D. (2020). Identification and estimation of undetected COVID-19 cases using testing data from Iceland. NBER Working Paper w27528.
- Benatia, D., Godefroy, R., & Lewis, J. (2020). Estimates of COVID-19 cases across four Canadian provinces. *Canadian Public Policy*, 46(S3), S203–S216.
- Böhning, D., Rocchetti, I., Maruotti, A., & Holling, H. (2020). Estimating the undetected infections in the COVID-19 outbreak by harnessing capture–recapture methods. *International Journal of Infectious Diseases*, 97, 197–201.
- Buitrago-Garcia, D., Egli-Gany, D., Counotte, M. J., Hossmann, S., Imeri, H., Ipekci, A. M., Salanti, G., & Low, N. (2020). Occurrence and transmission potential of asymptomatic and presymptomatic SARS-CoV-2 infections: A living systematic review and meta-analysis. *PLoS Medicine*, 17(9), e1003346.
- De Boor, C. (1972). On calculating with B-splines. *Journal of Approximation Theory*, 6(1), 50–62.
- Drikvandi, R., Verbeke, G., & Molenberghs, G. (2017). Diagnosing misspecification of the random-effects distribution in mixed models. *Biometrics*, 73(1), 63–71.
- Durban, M., & Aguilera-Morillo, M. C. (2017). On the estimation of functional random effects. *Statistical Modelling*, 17(1–2), 50–58.
- Durbán, M., Harezlak, J., Wand, M., & Carroll, R. (2005). Simple fitting of subject-specific curves for longitudinal data. *Statistics in Medicine*, 24(8), 1153–1167.
- Eilers, P. H., & Marx, B. D. (1996). Flexible smoothing with b-splines and penalties. *Statistical Science*, 11(2), 89–121.
- Esri Deutschland GmbH. (2020). Daily COVID-19 case numbers provided by the Robert-Koch-Institute. <https://www.arcgis.com/home/item.html?id=f10774f1c63e40168479a1feb6c7ca74>
- Flaxman, S., Mishra, S., Gandy, A., Unwin, H. J. T., Mellan, T. A., Coupland, H., Whittaker, C., Zhu, H., Berah, T., Eaton, J. W., Monod, M., Ghani, C. A., Donnelly, A. C., Riley, S., Vollmer, M. A. C., Ferguson, N. M., Okell, L. C., & Bhatt, S. (2020). Estimating the effects of non-pharmaceutical interventions on COVID-19 in Europe. *Nature*, 584(7820), 257–261.
- Fuhrmann, J., & Barbarossa, M. V. (2020). The significance of case detection ratios for predictions on the outcome of an epidemic - A message from mathematical modelers. *Archives of Public Health*, 78(63).
- Giordano, G., Blanchini, F., Bruno, R., Colaneri, P., Di Filippo, A., Di Matteo, A., & Colaneri, M. (2020). Modelling the COVID-19 epidemic and implementation of population-wide interventions in Italy. *Nature Medicine*, 26(6), 855–860.
- Harris, J. E. (2020). COVID-19 case mortality rates continue to decline in Florida. *medRxiv*.
- Hastie, T., & Tibshirani, R. (1993). Varying-coefficient models. *Journal of the Royal Statistical Society: Series B (Methodological)*, 55(4), 757–779.

- ifo Institut, & forsa. (2020). Die Deutschen und Corona - Schlussbericht der BMG-“Corona-BUND-Studie”. <https://www.ifo.de/publikationen/2020/monographie-autorenschaft/die-deutschen-und-corona>
- Jagodnik, K. M., Ray, F., Giorgi, F. M., & Lachmann, A. (2020). Correcting under-reported COVID-19 case numbers: Estimating the true scale of the pandemic. *medRxiv*.
- Kenyon, C. (2020). Flattening-the-curve associated with reduced COVID-19 case fatality rates-an ecological analysis of 65 countries. *Journal of Infection*, 81(1), e98–e99.
- Kip, K. E., Snyder, G., Yealy, D. M., Mellors, Minnier, T., Donahoe, M. P., McKibben, J., Collins, K., & Marroquin, O. C. (2020). Temporal changes in clinical practice with COVID-19 hospitalized patients: Potential explanations for better in-hospital outcomes. *medRxiv*.
- Levin, A., Hanage, W., Owusu-Boaitey, N., Cochran, B., Walsh, S. P., & Meyerowitz-Katz, G. (2020). Assessing the age specificity of infection fatality rates for COVID-19: Systematic review, meta-analysis, and public policy implications. *European Journal of Epidemiology*, 35, 1123–1138.
- Li, R., Pei, S., Chen, B., Song, Y., Zhang, T., Yang, W., & Shaman, J. (2020). Substantial undocumented infection facilitates the rapid dissemination of novel coronavirus (SARS-CoV-2). *Science*, 368(6490), 489–493.
- Mallett, S., Allen, A. J., Graziadio, S., Taylor, S. A., Sakai, N. S., Green, K., Suklan, J., Hyde, C., Shinkins, B., Zhelev, Z., Peters, J., Turner, P. J., Roberts, N. W., di Ruffano, L. F., Wolff, R., Whiting, P., Winter, A., Bhatnagar, G., Nicholson, B. D., & Halligan, S. (2020). At what times during infection is SARS-CoV-2 detectable and no longer detectable using rt-pcr-based tests? A systematic review of individual participant data. *BMC Medicine*, 18.
- Manski, C. F., & Molinari, F. (2020). Estimating the COVID-19 infection rate: Anatomy of an inference problem. *Journal of Econometrics*, 220, 181–192.
- Mathews, J. D., McCaw, C. T., McVernon, J., McBryde, E. S., & McCaw, J. M. (2007). A biological model for influenza transmission: pandemic planning implications of asymptomatic infection and immunity. *PLoS One*, 2(11), e1220.
- Mizumoto, K., Kagaya, K., Zarebski, A., & Chowell, G. (2020). Estimating the asymptomatic proportion of coronavirus disease 2019 (COVID-19) cases on board the Diamond Princess cruise ship, Yokohama, Japan, 2020. *Eurosurveillance*, 25(10), 2000180.
- R Core Team. (2013). R: A language and environment for statistical computing.
- Radon, K., Saathoff, E., Pritsch, M., Guggenbühl, N., Jessica, M., Kroidl, I., Olbrich, L., Thiel, V., Diefenbach, M., Riess, F., Forster, F., Theis, F., Wieser, A., Hoelscher, M., Bakuli, A., Eckstein, J., Froeschl, G., Geisenberger, O., Geldmacher, C. ... Schwettmann, L. (2020). Protocol of a population-based prospective COVID-19 cohort study Munich, Germany (KoCo19). *medRxiv*.
- Rahmandad, H., Lim, T. Y., & Sterman, J. (2020). Estimating COVID-19 under-reporting across 86 nations: Implications for projections and control. Available at SSRN 3635047.
- Robert-Koch-Institute. (2020). Seroepidemiological studies in the general population. https://www.rki.de/EN/Content/infections/epidemiology/outbreaks/COVID-19/AK-Studien-english/Sero_General.html
- Rocchetti, I., Böhning, D., Holling, H., & Maruotti, A. (2020). Estimating the size of undetected cases of the COVID-19 outbreak in Europe: An upper bound estimator. *Epidemiologic Methods*, 9(s1).
- Russell, T. W., Hellewell, J., Abbott, S., Jarvis, C., van Zandvoort, K., Ratnayake, R., CMMID nCov working group, Flasche, S., Eggo, R., Edmunds, W. J., & Kucharski, A. J. (2020). *Using a delay-adjusted case fatality ratio to estimate under-reporting*. Centre for Mathematical Modeling of Infectious Diseases Repository.
- Staerk, C., Wistuba, T., & Mayr, A. (2021). Estimating effective infection fatality rates during the course of the COVID-19 pandemic in Germany. *BMC Public Health*, 21(1073).
- Stella, L., Martínez, A. P., Bauso, D., & Colaneri, P. (2020). *The role of asymptomatic individuals in the covid-19 pandemic via complex networks*. arXiv preprint arXiv:2009.03649.
- Velavan, T. P., & Meyer, C. G. (2020). The COVID-19 epidemic. *Tropical Medicine & International Health*, 25(3), 278–280.
- Wood, S. (2017). *Generalized additive models: An introduction with R* (2nd ed.). Chapman & Hall/CRC Texts in Statistical Science. CRC Press.
- Wood, S. N., Pya, N., & Säfken, B. (2016). Smoothing parameter and model selection for general smooth models. *Journal of the American Statistical Association*, 111(516), 1548–1563.
- Wu, S. L., Mertens, A. N., Crider, Y. S., Nguyen, A., Pokpongkiat, N. N., Djajadi, S., Seth, A., Hsiang, M. S., Colford, J. M., Reingold, A., Arnold, B. F., Hubbard, A., & Benjamin-Chung, J. (2020). Substantial underestimation of SARS-CoV-2 infection in the United States. *Nature Communications*, 11(1), 1–10.

SUPPORTING INFORMATION

Additional supporting information may be found online in the Supporting Information section at the end of the article.

How to cite this article: Schneble, M., De Nicola, G., Kauermann, G., & Berger, U. A statistical model for the dynamics of COVID-19 infections and their case detection ratio in 2020. *Biometrical Journal*. 2021;1–10. <https://doi.org/10.1002/bimj.202100125>

Eidesstaatliche Versicherung

(Siehe Promotionsordnung vom 12.07.11, § 8, Abs. 2 Pkt. 5.)

Hiermit erkläre ich an Eidesstatt, dass die Dissertation von mir selbstständig, ohne unerlaubte Beihilfe angefertigt ist.

München, 05.08.2021

Ort, Datum

Marc Schneble

Unterschrift