Progress this week
March 15, 2020

- Wrote up transect sampling simulation and model fitting procedure.
- Added a figure showing the partition of the observed region used to calculate the numerical integration weights for variable sampling effort.
- Specified priors in the paper and code.
- Added posterior means and intervals for parameters in both results sections.

Previous word count: 3975
Previous page count: 19

Current word count:
Current page count:

REVIEW ARTICLE

The Integrated Nested Laplace Approximation applied to Spatial Point Process Models

Kenneth Flagg and Andrew Hoegh

Montana State University, Bozeman, MT

ARTICLE HISTORY
Compiled March 14, 2020

ABSTRACT
This template is for authors who are preparing a manuscript for a Taylor & Francis journal using the LaTeX document preparation system and the `interact` class file, which is available via selected journals' home pages on the Taylor & Francis website.

KEYWORDS
INLA, spatial prediction, log-Gaussian Cox process, spatial point process

1. Introduction

Statistical methods for spatial prediction result in a high-dimensional inference problem that can present computational challenges, particularly for large datasets. For instance, when the goal of statistical modeling is to produce a graphical map of a random variable over space, the model ultimately must be able to predict that random variable at every pixel of the image. A map image will typically be at least several hundred by several hundred pixels, so in total there can easily be hundreds of thousands of pixels requiring predictions. Thus, even when a model has only half a dozen parameters, spatial prediction may require hundreds of thousands of latent variables.

Spatial point process models further complicate the situation; in addition to computational challenges due to a large number of latent variables and parameters, point process models also require evaluating difficult likelihoods. *(Cite some computational papers — Baddely?)* Both maximum likelihood and Bayesian model fitting require integrating the intensity function over space, but the integral is generally not available in closed form and, unlike a probability distribution, the integral over the entire domain does not necessarily equal one.

Many methods have been introduced for modeling spatial point patterns, including quadrature-based approximations *(cite Baddeley)*, pseudodata approaches *(cite Baddeley/Berman/Turner etc)*, and Markov chain Monte Carlo [14]. The number of parameters and latent variables in these models still poses a major computational challenge. This computational challenge led to the development of INLA [18].

(Add a paragraph about computational challenges. This is really the point of INLA (fast albeit approximate solutions). —Andy)

CONTACT Kenneth Flagg. Email: kenneth.flagg@montana.edu

Development of the integrated nested Laplace approximation (INLA) has made accurate approximate model fitting considerably more feasible for a particular class of log-Gaussian Cox process (LGCP) models. INLA was developed to fit Bayesian hierarchical models with many latent Gaussian variables [18]. A key part of INLA's computational simplicity is that it calculates the posterior distribution of each latent Gaussian variable one at a time; that is, it provides only the posterior marginal distributions rather than the full joint distribution.

When using a LGCP for spatial mapping, two aspects make INLA a suitable approach. First, the LGCP is driven by a spatial Gaussian process (GP), so the latent variables are Gaussian. Second, even though the latent variables are expected to exhibit spatial dependence, their full joint distribution is not needed. In most situations it suffices to map their predicted values, variance, and upper and lower interval bounds pointwise across space.

We focus on spatial mapping using a hierarchical construction of the LGCP, and do so without discussing many classic spatial point process concepts such as Gibbs processes, Markov processes, point process densities, or the Papangelou conditional intensity function. For readers interested in the general spatial point process context in which the LGCP originates, we recommend other references [7, 8, 14].

(paragraph about partially observed point processes — describe both distance sampling and subwindow observation to set up for §2.3)

This article provides a review of recent advances in the fitting of spatial LGCP models via INLA, including a finite element approach for dimension reduction, a likelihood factorization that avoids gridding, and incorporation of sampling effort or false negatives. The examples use a freely-available datasets and the R code is provided in the supplementary materials.

The remainder of Section 1 provides an overview of spatial LGCP models and INLA. Section 2 describes the methodology developed to fit LGCP models in INLA, and Section 3 presents example analyses of real-world data. Finally, Section 4 presents concluding remarks.

1.1. Log-Gaussian Cox Process

The log-Gaussian Cox process is a Poisson process driven by a latent Gaussian process [15]. This section provides a brief overview of the spatial Poisson process and then presents the hierarchical construction of the spatial LGCP.

A spatial point process generates a random set of events (points) in space. The process is characterized primarily by its intensity function $\lambda(u)$, which gives the local mean number of events per unit area at any point in space. The instensity function is nonnegative and defined on a domain $\mathcal{D} \subset \mathbb{R}^2$. The point process is observed in a bounded window $\mathcal{R} \subset \mathcal{D}$. The point process is denoted $\mathbf{X}$, and its realizations are $\mathbf{x} = \{x_1, \ldots, x_n\}$. A realized (observed) $\mathbf{x}$ is called a point pattern; the elements of the point pattern are the events.

The Poisson process is fully defined by its intensity function and satisfies the following two properties [14]:

(1) The number of events in any region $\mathcal{B}$ follows a Poisson distribution with mean $\int_{\mathcal{B}} \lambda(u)\mathrm{d}u$.
(2) Given the number of events in $\mathcal{B}$, those events are independent and identically distributed with probability density $\lambda(u)/\int_{\mathcal{B}} \lambda(u)\mathrm{d}u$.

The distribution function of a Poisson process is

$$[\mathbf{X}|\lambda] = \frac{\exp\left[-\int_{\mathcal{R}} \lambda(u)\mathrm{d}u\right]}{n!} \prod_{x \in \mathbf{X}} \lambda(x)$$

as given in Cressie (1993) [7]. This can be extended to include further parameters used to define λ, most commonly coefficients to construct a log-linear model with covariates, or a polynomial or spline function of the spatial coordinates.

In a Poisson process, the intensity function is a fixed parameter (or a deterministic function of fixed parameters and covariates). In applications of spatial point process models, this is often too simplistic because spatial heterogeneity may be driven by other random processes with their own parameters about which inferences are desired.

A Cox process is a generalization of the Poisson process where the intensity function is a realization of another stochastic process. Conditional on the intensity function, a Cox process must satisfy the two Poisson process properties, but the intensity function can take any form. Cox processes are commonly used to model clustering of events, such as in the classic Neyman-Scott process where cluster centers are generated by another Poisson process [17]. Another example is where the intensity is a Dirichlet process mixture [20]. In situations where the intensity function is less structred but exhibits spatial autocorrelation, the Cox process can incorporate a geostatistical process. The log-Gaussian Cox process is an example of this.

The LGCP uses a log-linear model for the intensity

$$\log \lambda(u) = \mathbf{\Psi}(u) = \mathbf{z}(u)'\boldsymbol{\beta} + \mathbf{e}(u),$$

where $\mathbf{e}$ is a Gaussian process with mean 0 and covariance function $C(u, v)$.

Where the spatial Poisson process has a (stochastically) fixed intensity to be estimated from data, the spatial LGCP induces a hierarchical model where the intensity function is itself a random process to be predicted.

Badeley et. al. (2005) and Möller and Waagepetersen (2007) present model checking tools for LGCP and other spatial point process models [3, 14].

(insert paragraph reviewing applications and uses of LGCP)

1.2. Integrated Nested Laplace Approximation

INLA was developed with the goal of providing fast, accurate, deterministic approximations for posterior marginal distributions of the parameters in latent Gaussian models [18]. The setting is Bayesian generalized additive models but is kept very general; this includes linear models, models with nonlinear (spline, random walk, etc.) functions of predictors, and models with temporal and/or spatial dependence. Such models commonly include many Gaussian latent variables or parameters with Gaussian priors and relatively few non-Gaussian parameters. The INLA approach makes repeated use of Laplace expansion, numerical integration, and numerical search. The method has established usefulness for LGCP [11]. It is readily implemented using the standalone INLA software or in R via the R-INLA package [12].

Add a page or so walking through the algorithm.

2. Methodology

INLA provides an efficient computational framework for fitting Bayesian models with latent Gaussian variables. On top of this framework, several tools are built to further simplify the fitting of spatial LGCP models. The stochastic partial differential equation (SPDE) approach provides dimension reduction for the spatial GP [13]. The SPDE approach employs a numerical integration scheme which can also be used to approximate the LGCP likelihood and negate the need to grid the events into Poisson counts [19]. With these computational improvements, researchers are now able to efficiently fit LGCP models to large spatial point pattern datasets and even incompletely-observed point patterns [21].

2.1. The SPDE Approach

Because the LGCP includes a Gaussian process (GP), efficient computation for Gaussian processes is critical when working with LGCP models. The GP imposes a dense covariance matrix on the latent variables [4]. For GPs with a Matérn covariance function, a Gaussian Markov random field (GMRF) approximation can simplify computation, requiring only a sparse covariance structure.

The GMRF approximation is motivated by the fact that Gaussian fields with Matérn covariances are solutions to the stochastic partial differential equation (SPDE) below [13]:

$$\tau(\kappa^2 - \Delta)^{\alpha/2}\mathbf{e}(u) = \mathbf{W}(u), \qquad u \in \mathbb{R}^d, \quad \kappa > 0, \quad \alpha = \nu + d/2, \quad \nu > 0.$$

Here, $\mathbf{W}$ is a Gaussian white noise process with variance 1, and Δ is the Laplacian operator. The stationary solution $\mathbf{e}$ is a Gaussian field having a Matérn covariance function with precision (inverse variance) τ, scaling parameter κ (approximately inversly proportional to the range), and smoothness parameter ν.

Lindgren, Rue, and Lindström (2011) investigate the limit as $\nu \to 0$ for $d = 2$, finding that the solution is a GMRF on a unit lattice [13]. They then construct approximations for positive integer values of ν by ν-fold convolution of $\mathbf{e}$ with itself. Finally, they use a finite element method to generalize the approximation to arbitrary triangulations of the support. This approximation has considerable computational benefits because the GMRF has a sparse covariance structure; the only nodes with nonzero covariances are those directly connected by edges in the triangulation.

The core of the SPDE approach is the finite-element GMRF representation of the GP. This technique can approximate complicated surfaces as relatively low-dimensional vectors via a piecewise-linear approximation. Figure 1 shows an example. The surface is modeled at the nodes of the mesh. Values at points other than the nodes are found by linear interpolation, resulting in a surface constructed from triangular faces. When using the finite element representation to compute averages or for numerical integration, the values at the nodes are used, and are weighted by the area represented by the nodes (Figure 2). This area is one of third the total area of the triangles of which the node is a vertex. *(Add linear algebra and basis function stuff here.)*

In practice, the SPDE approach goes as follows. Choose nodes s_i at which to model $\mathbf{\Psi}(s_i) = \mathbf{z}(s_i)'\boldsymbol{\beta} + \mathbf{e}(s_i)$, then build a triangular mesh using these nodes. Typically the nodes will include locations where data or covariates are available, then the rest will

be filled in with a Delaunay triangulation under some edge length constraints. The $\mathbf{e}(s_i)$ are modeled as a GMRF where the distribution of each $\mathbf{e}(s_i)$ depends only on the $\mathbf{e}(s_j)$ where s_i and s_j are connected by an edge. The GMRF representation is assumed to be a piecewise linear approximation of the continuous Gaussian field; values of $\mathbf{e}(s)$ for s not in the set of nodes are predicted by linear interpolation using the barycentric coordinates of s (Figure 3). The mathematical form of this interpolation is very simple; for a point u inside the triangle with vertices s_1, s_2, s_3, the approximation is simply

$$\mathbf{e}(u) \approx \alpha\mathbf{e}(s_1) + \beta\mathbf{e}(s_2) + \gamma\mathbf{e}(s_3).$$

(define the dual region)

The SPDE approach is implemented in the `INLA` package alongside tools for constructing meshes. A wrapper for easily specifying LGCP models is provided in the `inlabru` package [1].

2.2. Going Off the Grid

Point pattern data are sometimes known as presence-only data, underscoring the fact that information about where events did *not* occur is both important and often overlooked. There have been many proposed methods to account for regions that were observed to contain no events (in contrast to unobserved regions where it is unknown if any events are present). Many of these methods involve imputation of dummy points or discretization. In the world of maximum likelihood, perhaps the most well-developed of these use approximations based on logistic regression on presence/absence information in small disjoint regions *(cite a bunch of Baddely etc papers).*

Another alternative, which is probably the most popular approach to Bayesian fitting of LGCP models but is also common in frequentist analyses, is to grid the domain and model the induced Poisson counts. It has long been understood that results are sensitive to the discretization scheme [5]. Simpson et. al. (2016) explain that this is also computationally wasteful [19].

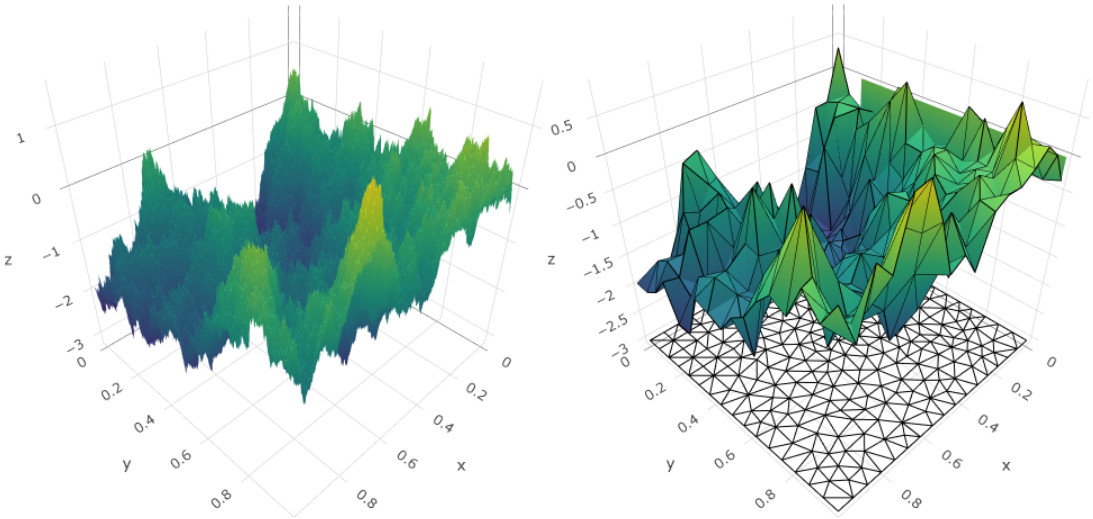
Simpson et. al. took LGCP inference "off grid" by intoducing a computationally-efficient approximation to the Poisson process likelihood that requires the intensity function only to be evaluated at the locations of observed events and at the nodes of a mesh. Thus, the SPDE approach can be employed to model the intensity surface and the same nodes reused in evaluation of the Poisson process likelihood. The result is a substantial improvement in both computing time and accuracy of the approximation compared to gridding.

The approximation arises from a factorization of the Poisson processes likelihood. For notational clarity, assume $\log[\lambda(u)] = \mathbf{\Psi}(u)$. The exact log-likelihood is

$$\ell(\lambda) = C - \int \lambda(u)\mathrm{d}u + \sum_{i=1}^{n} \log\left[\lambda(x_i)\right]$$

where C is a normalizing constant. The log-intensity is projected into the space spanned by a finite set of basis functions,

$$\log\left[\lambda(u)\right] \approx \sum_{j=1}^{m} \psi_j \phi_j(u)$$



(a) Example Gaussian process realization.

(b) Finite element approximation.

Figure 1. A realization of a spatial Gaussian process and an approximation of that realization over a triangular mesh.

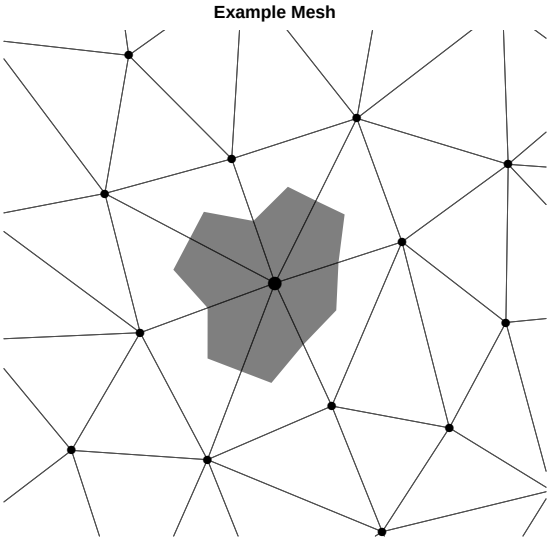
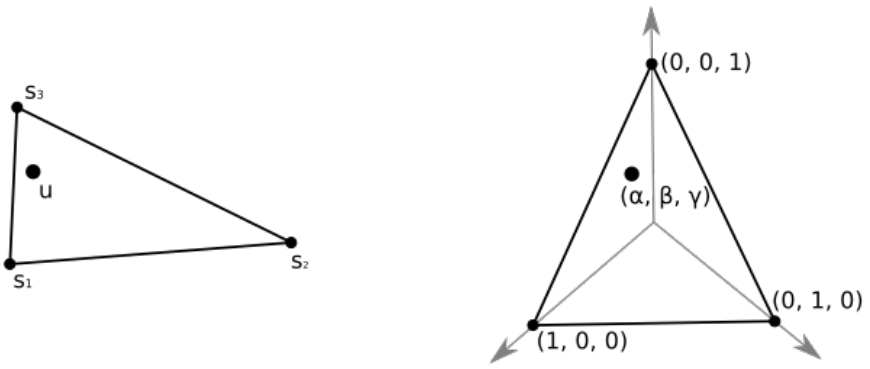


Figure 2. An illustration of the nodes (dots) and the weighting scheme for one node (large dot). In the SPDE approach, this node represents the shaded region, so its weight is proportional to the shaded area. The shaded region is constructed by connecting the midpoints of the node's edges to the midpoints of the adjacent triangles.



(a) A point u in a triangle with vertices s_1, s_2, s_3.

(b) A point (α, β, γ) in a simplex with vertices $(1,0,0), (0,1,0), (0,0,1)$.

Figure 3. An illustration of the linear transformation from a mesh triangle to the 2-simplex. The transformation is defined from $\mathbb{R}^2$ to $\mathbb{R}^3$ such that the vertices are mapped to a unit vector along each axis. This transformation takes the point u to (α, β, γ), its barycentric coordinates. The barycentric coordinates have the property that $\alpha + \beta + \gamma = 1$.

with $\boldsymbol{\psi} = (\psi_1, \ldots, \psi_m)'$ a multivariate normal random vector and $\{\phi_1, \ldots, \phi_m\}$ a set of linearly independant basis functions. *(The basis function representation is actually the SPDE representation. Say so and move this to the previous section.)* This defines a numerical integration scheme at nodes s_i with weights α_i, where the weight is the observed area represented by the node (Figure 2), and yields the approximation,

$$\begin{aligned}\ell(\lambda) &\approx C - \sum_{i=1}^{m} \alpha_i \exp\left[\sum_{j=1}^{m} \psi_j \phi_j(s_i)\right] + \sum_{i=1}^{n}\sum_{j=1}^{m} \psi_j \phi_j(x_i) \\ &= C - \boldsymbol{\alpha}' \exp\left[\mathbf{A}\boldsymbol{\psi}\right] + \mathbf{y}'\mathbf{A}_2\boldsymbol{\psi}.\end{aligned}$$

This is equivalent to the likelihood of independent Poisson random variables with means $\alpha_i \eta_i$ *(translate the confusing algebra from the appendix)*, and with observations $y_i = 0$ at the nodes s_i and $y_i = 1$ at the events x_i. Thus, the SPDE approach, which can be easily fit in R-INLA, can be combined with a Poisson GLM to rapidly fit a LGCP model. There is no need to compute grid counts, only to define dummy points at the mesh nodes and construct a pseudodata vector $\mathbf{y}$.

2.3. Variable Sampling Effort

There has long been a gap between point process modeling theory and practice, where point process models are only fit to data from completely-observed domains under the assumption that every event was detected perfectly. In practice, this is not the case as it may be impossible or impractical to census the entire region of interest. For example, line-transect surveys routinely generate point pattern data but are analyzed after aggregation rather than using a point process model. Another issue is that of false negatives, in other words events which exist but are not detected during the survey. False negatives are an accepted part of species abundance surveys, where the species of interest may be camouflaged or hidden in thick cover *(citation?)*.

The idea of the incompletely-observed domain has been around for some time. Brix and Möller (2001) fit an LGCP model to weed data observed in rectangular frames and used a Metropolis-adjusted Langevin algorthim to predict the intensity outside of the observed frames [5]. Chakraborty et. al (2011) discuss nonhomogeneous Poisson process modeling as a richer alternative to ecological presence/absence models and describe their data as "degraded" in the sense that sampling bias prevented the entire region from being fully observed; they fit their model by aggregating the point pattern to counts in grid cells and using MCMC [6]. Gabriel et. al. (2016, 2017) derived a relationship between the Kriging predictor and the LGCP to predict the LGCP intensity across an unobserved subset of the region of interest [9, 10].

With INLA, the SPDE approach, and the "off grid" approximation facilitating the routine fitting of LGCP models, there is now renewed interest in accounting for variable sampling effort in spatial point process models [19, 21].

Sampling effort is accounted for using the theory of thinned point processes. Thinning refers to the events of one point process being kept or discarded probabilistically. Let $\lambda(u)$ be the intensity of the parent process, and let an event $x = u$ (if it exists) be observed with probability $p(u)$. The observed point process is a thinned point process with intensity

$$\lambda_p(u) = p(u)\lambda(u),$$

and if the parent process is a Poisson process then the observed process is also a Poisson process [16].

We seek to make posterior inferences about the parent intensity, $\lambda(u)$. If $p(u)$ is known, its value at each node is used to adjust the SPDE integration weights. Most usefully, if it is known that $p(s) = 0$ at certain nodes s because they were outside the surveyed domain, the weights become zero so those nodes do not contribute to the integral. More generally, when a known subdomain is surveyed, $p(u)$ is an indicator function for u being in the surveyed region. When $p(u)$ is known, the integration weight for node s_i is multiplied by

$$\int_{\text{dual}(s_i)} p(u)\mathrm{d}u$$

which, for indicator functions, is simply the proportion of the dual region that is inside the surveyed region.

If $p(u)$ is unknown, it can be modeled. Taking the logarithm of the thinned intensity and substituting in the basis function representation, we have the log-linear model

$$\log\left[\lambda_p(u)\right] = \log\left[p(u)\right] + \sum_{j=1}^{m} \psi_j \phi_j(u).$$

Thus, any log-linear model for $p(u)$ that can be fit by INLA and the SPDE approach can be incorporated into the LGCP model. This provides a convenient way to estimate the detection function from distance sampling data; the model for $\log(p(u))$ includes the distance to the nearest transect as a covariate. For example, Yuan et. al (2017) fit such a model to data from a line-transect survey using a spline model to account for the unknown detection function [21].

3. Applications

3.1. Beilschmiedia Pendula Lauraceae Dataset

For an example analysis of real-world data, we consider the locations of $n = 3605$ *Beilschmiedia pendula Lauraceae* trees in a 1000 meter by 500 meter plot in a tropical rainforest on Barro Colorado Island, Panama [14]. The data are available as the `bei` dataset in the `spatstat` R package [2]. The point pattern is accompanied by elevation and gradient covariates measured on a square lattice with 5 m spacing. The point pattern exhibits inhomogeneity which appears to be associated with elevation and the elevation gradient (Figure 4).

3.1.1. Model Specification

We use the log-Gaussian Cox process model with intensity

$$\log\left[\lambda(u)\right] = \beta_0 + \beta_1 z_1(u) + \beta_2 z_2(u) + \mathbf{e}(u)$$

where $z_1(u)$ is the elevation at u, $z_2(u)$ is the magnitude of the elevation gradient at u, and $\mathbf{e}(u)$ is a zero-mean Gaussian process with Matérn covariance and $\alpha = 2$.

The prior distributions are set as follows. β_0, β_1, and β_2 are independent $\mathrm{N}(0, \infty)$. For the Matérn covariance parameters, R-INLA provides a PC prior defined in terms of the standard deviation σ and the effective range ρ. The PC prior implements is specified by defining a tail probability for each parameter. We set $\Pr(\sigma > 1) = 0.5$ to be relatively uninformative for the SD of a log-scale random effect, and $\Pr(\rho < 45) = 0.5$ similar to the prior used by Möller and Waagepetersen [14].

3.1.2. Fitting in R-INLA

We fit the model in R using the `INLA` package and its implementation of the SPDE approach. Some manual preprocessing was needed to employ the Poisson factorization. Our R code can be found in the supplementary materials. This section provides a conceptual explanation of the model-fitting process.

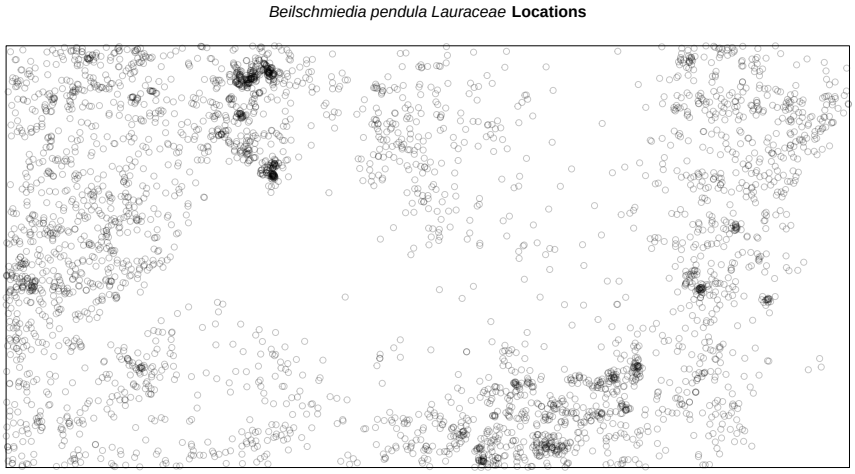
The first task is to construct a mesh for the piecewise linear approximations. A 25 meter resolution appears adequate to accurately represent the covariate surfaces. We used tools in the `INLA` package to create a Delaunay triangulation with a maximum edge length of 25 m. The resulting mesh has $m = 2145$ nodes $s_1, \ldots, s_{2145}$ and 4096 triangles (Figure 5). We used linear interpolation to approximate the covariates $z_1(s_i)$ and $z_2(s_i)$ at each node. After creating the mesh, we specifiy the priors for the covariance parameters and `INLA` can initialize an object to represent the covariance structure on the mesh.

The next step is to define the pseudodata and calculate the numerical integration weights. The pseudodata vector is

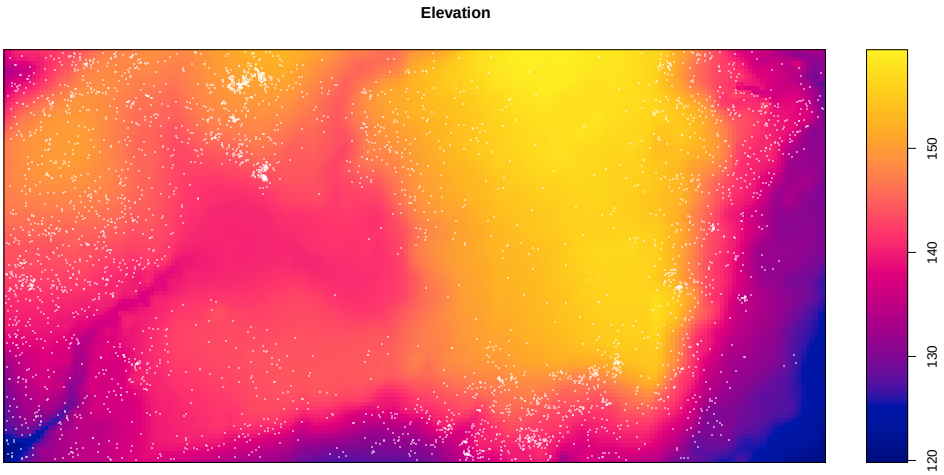
$$\mathbf{y} = (0, \ldots, 0, 1, \ldots, 1)'$$

consisting of 2145 zeros corresponding to the mesh nodes and 3604 ones representing the events. The weight vector is

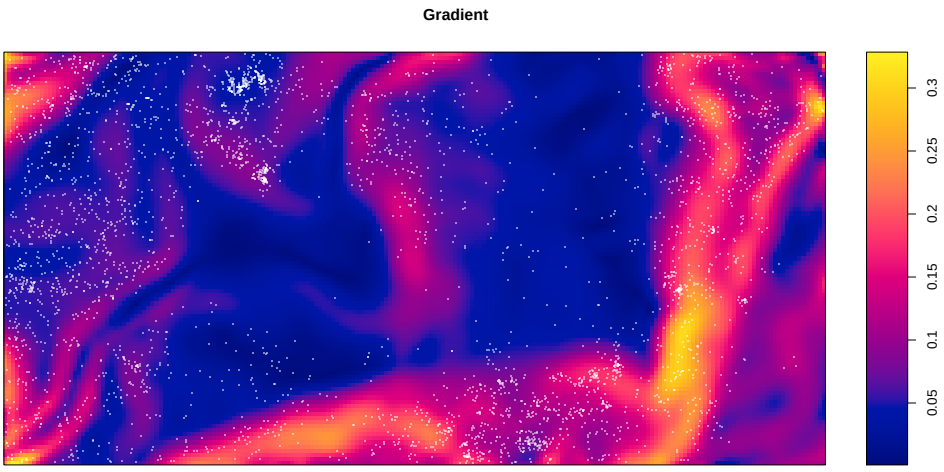
$$\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_{2145}, 0, \ldots, 0)'.$$



(a) Locations of trees (events).



(b) Elevation in meters, with events overlaid.



(c) Magnitude of the elevation gradient, with events overlaid.

Figure 4. Plots of the event and covariate data.

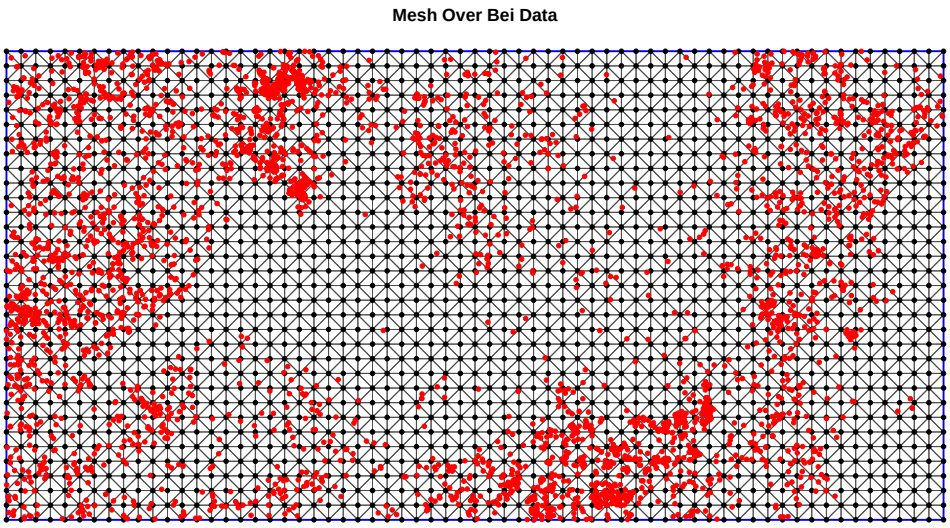


Figure 5. Triangular mesh with pseudodata overlaid. Black dots represent pseudodata with values of 0 at the mesh nodes. Red dots represent pseudodata with values of 1 at the events.

The alphas equal one-third of the total area of all triangles of which each node is a vertex, calculated by the `INLA` package. The zeros correspond to the 3604 events, which do not contribute to the numerical integration.

Talk about the $\mathbf{A}$ *matrix.*

The final bit of setup is to combine the pseudodata, covariates at the nodes, and a node index variable into a data list for the `inla()` function. This function is then called with arguments specifying a formula for the linear predictor, a Poisson family model (which defaults to a log link), the data list, the $\mathbf{A}$ matrix, the priors for the coefficients, and the $\boldsymbol{\alpha}$ vector as the exposure parameter.

3.1.3. Results

Posterior means and central 95% posterior intervals: β_0: 10.9, $(-14.2, -7.61)$; β_1: 0.0327, $(0.0111, 0.0547)$; β_2: 4.43, $(2.41, 6.45)$; σ: 1.31, $(1.06, 1.65)$; Range: 178, $(138, 234)$

decide which of figures 8, 9, 10, 11 are needed, then combine into a single multi-panel figure.

The posterior mean of the fixed effect component of the linear predictor,

$$E(\beta_0|\mathbf{x}) + E(\beta_1|\mathbf{x})z_1(u) + E(\beta_2|\mathbf{x})z_2(u),$$

appears as an average of the elevation and gradient surfaces (Figure 8). The fixed effects do not explain all of the heterogeneity of the point pattern; some spatial structure remains to be described by the posterior predicted GP (Figure 9).

3.1.4. Model Checking

For a more general discussion of residuals, see [3]. The classic Pearson residual,

$$\frac{\text{Observed count} - \text{Expected count}}{\sqrt{\text{Expected count}}},$$

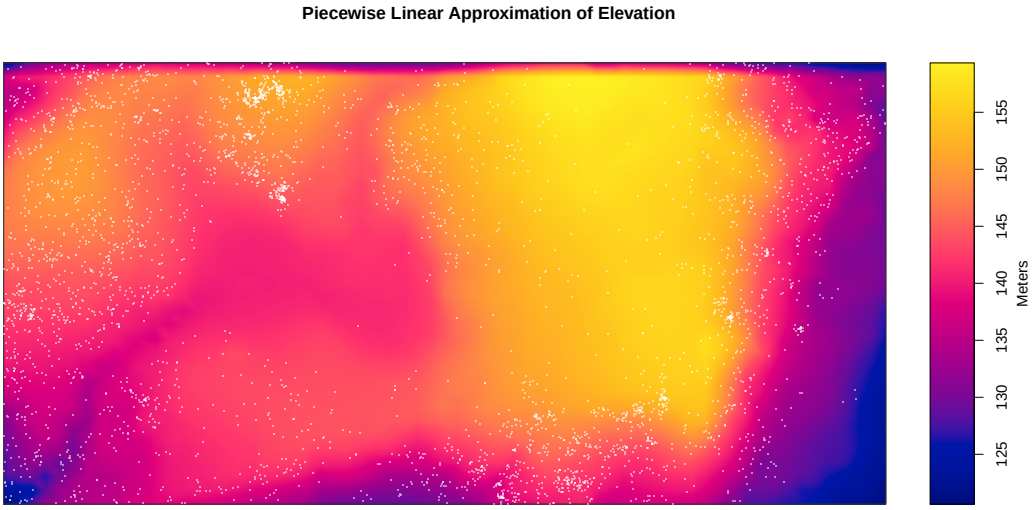


Figure 6. Piecewise linear approximation of the elevation surface.

is a special case.

Should make the whole four-panel plot if citing Baddeley et. al.

Pearson residuals (Figure 12): small magnitude in the region with no observed events and low posterior intensity (not possible to see much variation), the model overestimates the intensity in the canyon (too small to be accurately represented on the mesh), random with no obvious spatial patterns elsewhere, a few large-magnitude positive residuals (maybe there is some clustering?), tried a few different grids (not shown) and saw the same patterns

3.2. Transect Sampling Example

In this section, we illustrate the analysis of incompletely-observed point pattern data using the *Beilschmiedia pendula Lauraceae* and a simulated transect sampling scheme. Suppose part of the study requires every observed tree to be visited so time-consuming measurements can be made on each. For practical reasons, the researchers decide to take a systematic sample of transects and limit data collection to events within 5 m of a transect. Sections of the site farther than 5 m from a transect are unobserved, meaning events in those regions cannot be observed and nothing is known about the number of events or their locations. We assume the elevation and gradient data were available from another source so that the full covariate data can be used to predict the intensity in the unobserved parts of the site.

We simulate this by choosing a random starting point and defining transects running north-south spaced 50 m apart. All events within 5 m of a transect are saved; all other events are discarded. The result is $n = 710$ events observed in 20 rectangles sized 500 m by 10 m (Figure 13).

(the coding is more involved for the window approach than for distance sampling — mention a distance sampling example is in the online supplement but the preprocessing and model-fitting procedure is not much more complicated than for the fully-observed situation)

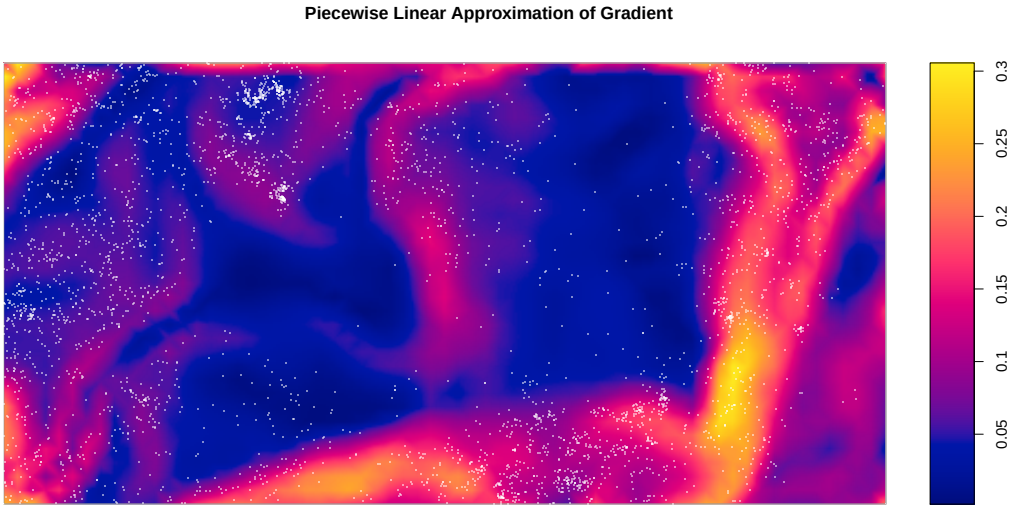


Figure 7. Piecewise linear approximation of the gradient surface.

3.2.1. Model Specification

We use the same model as fit to the full data, a log-Gaussian Cox process model with intensity

$$\log\left[\lambda(u)\right] = \beta_0 + \beta_1 z_1(u) + \beta_2 z_2(u) + \mathbf{e}(u).$$

As before, $z_1(u)$ is the elevation at u, $z_2(u)$ is the magnitude of the elevation gradient at u, and $\mathbf{e}(u)$ is a zero-mean Gaussian process with Matérn covariance and $\alpha = 2$. The prior distributions are again β_0, β_1, and β_2 independent $\mathrm{N}(0, \infty)$, $\Pr(\sigma > 1) = 0.5$, and $\Pr(\rho < 45) = 0.5$.

3.2.2. Fitting in R-INLA

The procedure for fitting the LGCP model to variable sampling effort data is the same as for fully-observed data except that the weight vector $\boldsymbol{\alpha}$ is different. Our code is available in the supplementary materials.

The mesh (Figure 14) is the same as used with the fully-observed point pattern, with $m = 2145$ nodes. We interpolated the covariates at the nodes and had `INLA` can initialize the covariance structure as before. The pseudodata vector changes dimension, becoming

$$\mathbf{y} = (0, \ldots, 0, 1, \ldots, 1)'$$

with 2145 zeros corresponding to the mesh nodes and 710 ones for the observed events.

The computation of the weight vector is more involved. The notation is again

$$\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_{2145}, 0, \ldots, 0)',$$

now with 710 zeros representing the observed events. The alphas equal the observed area represented by each node. We used tools from `spatstat` to calculate these areas.

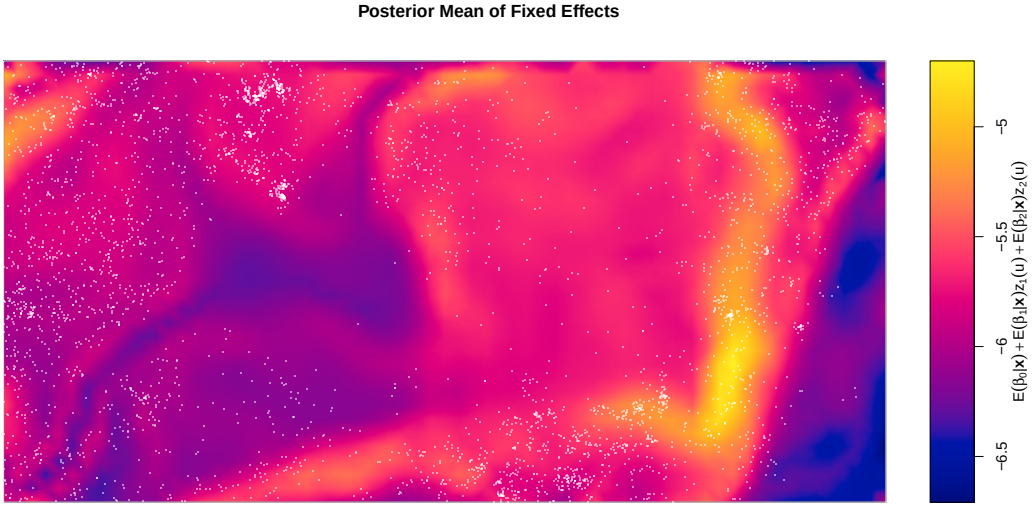


Figure 8. Posterior mean surface for the fixed effects.

First, we created an `owin` object defining the observed region along the transects. Then, we created a list of `owin` objects with each representing one triangle of the mesh. Next, we intersected the observed region with each triangle to partition the observed region into many small shards, each contained within a single triangle (Figure 17). Then we calculated the area of each shard. These areas are each associated with a triangle, but the need to be allocated to the nodes at the vertices of the triangle. To do this, we found the centroid of each shard and allocated the area proportional to the barycentric coordinates of the centroid. This way, the area of each shard is distributed to the three nodes which define the triangle, and the closest node gets the most weight while the farthest node get the least weight. Finally, the alpha value for each node is computed by summing the weights contributed by each shard to that node.

Is there a better (or more formal) term than "shard"? It has an intuitive meaning to me, but it feels weird to create nonstandard terminology for use in exactly one paragraph. —Kenny

Finally, the pseudodata, covariates, and an index are combined into a data list and the `inla()` function is called to fit the model.

3.2.3. Results

Posterior means and central 95% posterior intervals: β_0: -12.7, $(-18.7, -7.07)$; β_1: 0.0445, $(0.00676, 0.0839)$; β_2: 6.19, $(3.21, 9.19)$; σ: 1.29, $(0.967, 1.76)$; Range: 237, $(162, 352)$

decide which of figures 18, 19, 20, 21 are needed, then combine into a single multi-panel figure.

The posterior mean of the fixed effect component of the linear predictor, again appears very as an average of the elevation and gradient surfaces (Figure 18), very similar to the previous figure. As before, there is some additional spatial structure described by the GP (Figure 19). However, the predicted GP surface is smoother compared to the model fit to the fully data. *(compare posterior dists of covariance params)*

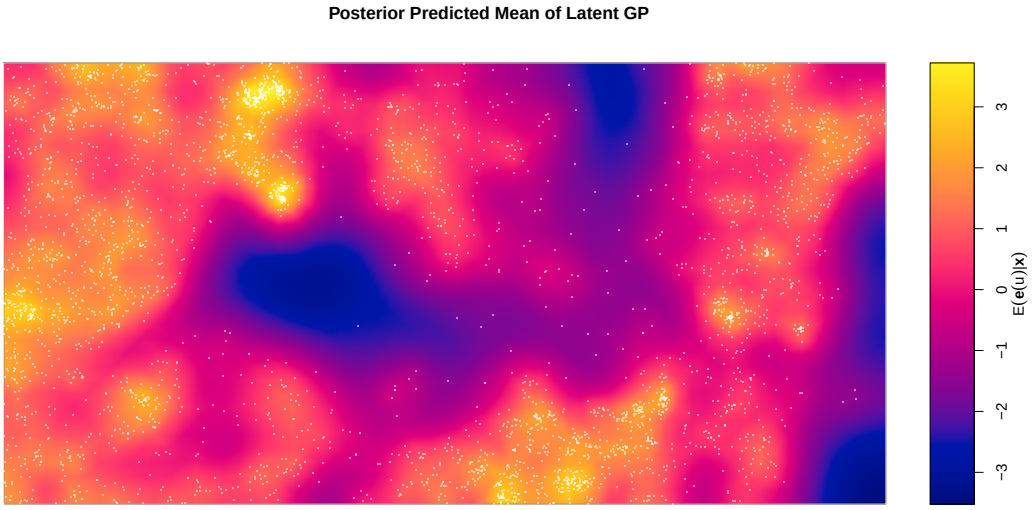


Figure 9. Posterior mean of prediction surface.

3.2.4. Model Checking

Pearson residuals (Figure 22): undefined where grid cells have 0 observed area (because 0 events expected), where defined the residuals appear similar to those from the model fit to the full data, small magnitude in the observed region with no events and low posterior intensity, more large positive residuals than large-magnitude negative residuals, random with no obvious spatial patterns

4. Conclusion and Discussion

not really possible to get a posterior distribution of the intensity function because it is a nonlinear transformation of several parameters and we do not have their joint posterior

the mesh used for `bei` for is finer than needed to describe the GP — think about coarsening and refining based on the detail needed to represent the covariates (is there literature on this?)

comment on similarities in posterior between model fit to full data and model fit to sampled data

need further work on model checking when prediction is desired over a different region than observed

shows that mapping based on Bayesian LGCP models is now practical with useful accuracy in short amount of time

References

[1] F.E. Bachl, F. Lindgren, D.L. Borchers, and J.B. Illian, *inlabru: an R package for bayesian spatial modelling from ecological survey data*, Methods in Ecology and Evolution 10 (2019), pp. 760–766.

[2] A. Baddeley and R. Turner, *Spatstat: An R package for analyzing spatial point patterns*, Journal of Statistical Software 12 (2005), pp. 1–42.

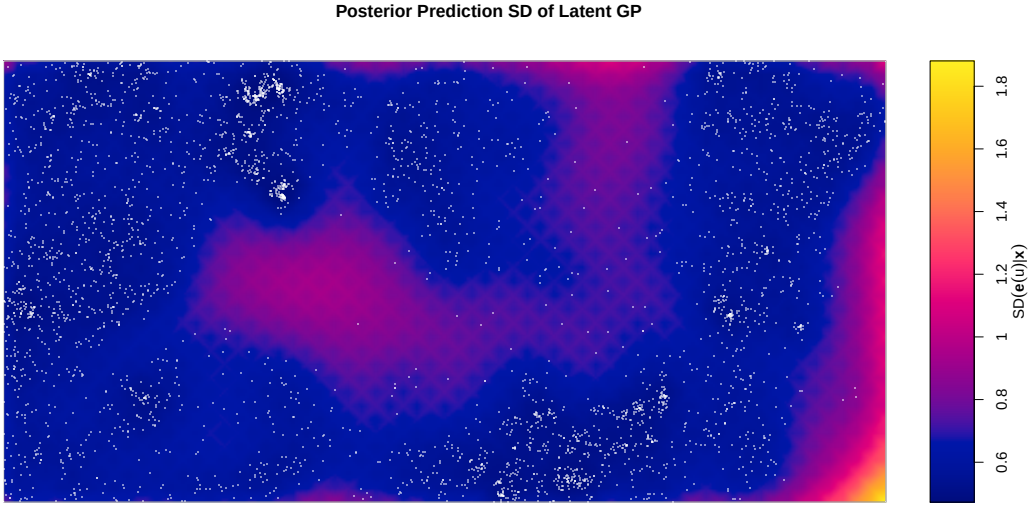


Figure 10. Pointwise posterior standard deviation of predictions.

[3] A. Baddeley, R. Turner, J. Møller, and M. Hazelton, *Residual analysis for spatial point processes*, Journal of the Royal Statistical Society: Series B (Statistical Methodology) 67 (2005), pp. 617–666.

[4] M. Blangiardo and M. Cameletti, *Spatial and Spatio-temporal Bayesian Models with R-INLA*, Wiley, 2015.

[5] A. Brix and J. Møller, *Space-time multi type log gaussian cox processes with a view to modelling weeds*, Scandinavian Journal of Statistics 28 (2001), pp. 471–488.

[6] A. Chakraborty, A.E. Gelfand, A.M. Wilson, A.M. Latimer, and J.A. Silander, *Point pattern modelling for degraded presence-only data over large regions*, Journal of the Royal Statistical Society: Series C (Applied Statistics) 60 (2011), pp. 757–776.

[7] N. Cressie, *Statistics for Spatial Data*, Wiley, 1993.

[8] P. Diggle, *Statistical Analysis of Spatial and Spatio-Temporal Point Patterns*, 3rd ed., CRC Press, 2013.

[9] E. Gabriel, F. Bonneu, P. Monestiez, and J. Chadœuf, *Adapted kriging to predict the intensity of partially observed point process data*, Spatial Statistics 18 (2016), pp. 54–71.

[10] E. Gabriel, J. Coville, and J. Chadœuf, *Estimating the intensity function of spatial point processes outside the observation window*, Spatial Statistics 22 (2017), pp. 225–239.

[11] J.B. Illian, S.H. Sørbye, and H. Rue, *A toolbox for fitting complex spatial point process models using integrated nested laplace approximation (INLA)*, The Annals of Applied Statistics (2012), pp. 1499–1530.

[12] F. Lindgren and H. Rue, *Bayesian spatial modelling with R-INLA*, Journal of Statistical Software 63 (2015), pp. 1–25.

[13] F. Lindgren, H. Rue, and J. Lindström, *An explicit link between gaussian fields and gaussian markov random fields: the stochastic partial differential equation approach*, Journal of the Royal Statistical Society: Series B (Statistical Methodology) 73 (2011), pp. 423–498.

[14] J. Møller and R. Waagepetersen, *Modern spatial point process modelling and inference*, Scandinavian Journal of Statistics 34 (2007), pp. 643–711.

[15] J. Møller, A.R. Syversveen, and R.P. Waagepetersen, *Log gaussian cox processes*, Scandinavian journal of statistics 25 (1998), pp. 451–482.

[16] J. Møller and R.P. Waagepetersen, *Statistical Inference and Simulation for Spatial Point Processes*, CRC Press, 2003.

[17] J. Neyman and E.L. Scott, *Statistical approach to problems of cosmology*, Journal of the Royal Statistical Society. Series B (Methodological) (1958), pp. 1–43.

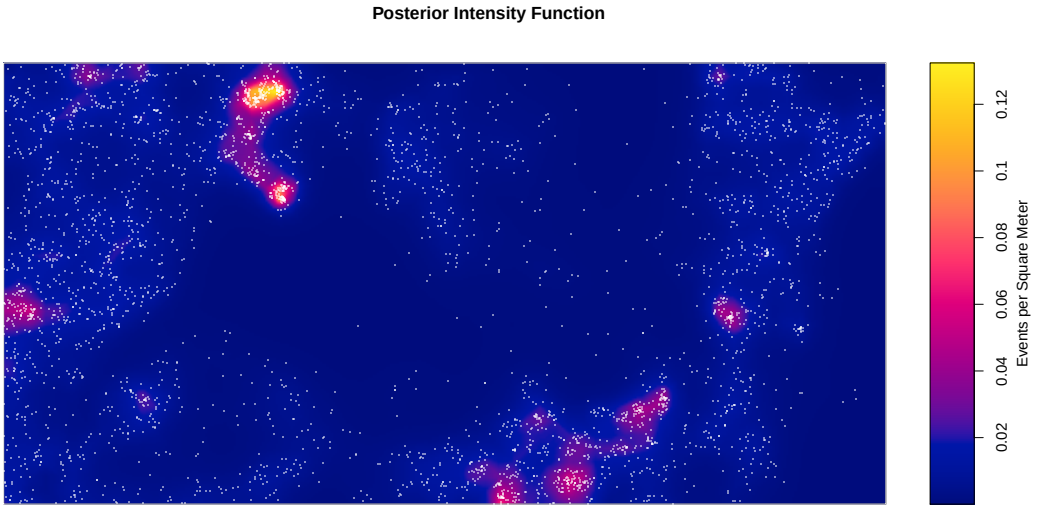


Figure 11. Posterior intensity surface in events per square meter, calculated using the piecewise linear approximate covariate surfaces and the posterior means of the intercept, coefficients, and GP.

[18] H. Rue, S. Martino, and N. Chopin, *Approximate bayesian inference for latent gaussian models by using integrated nested laplace approximations*, Journal of the royal statistical society: Series b (statistical methodology) 71 (2009), pp. 319–392.

[19] D. Simpson, J.B. Illian, F. Lindgren, S.H. Sørbye, and H. Rue, *Going off grid: Computationally efficient inference for log-gaussian cox processes*, Biometrika 103 (2016), pp. 49–70.

[20] M. Taddy, *Autoregressive mixture models for dynamic spatial poisson processes: Application to tracking intensity of violent crime*, Journal of the American Statistical Association 105 (2010), pp. 1403–1417.

[21] Y. Yuan, F.E. Bachl, F. Lindgren, D.L. Borchers, J.B. Illian, S.T. Buckland, H. Rue, T. Gerrodette, *et al.*, *Point process models for spatio-temporal distance sampling data from a large-scale survey of blue whales*, The Annals of Applied Statistics 11 (2017), pp. 2270–2297.

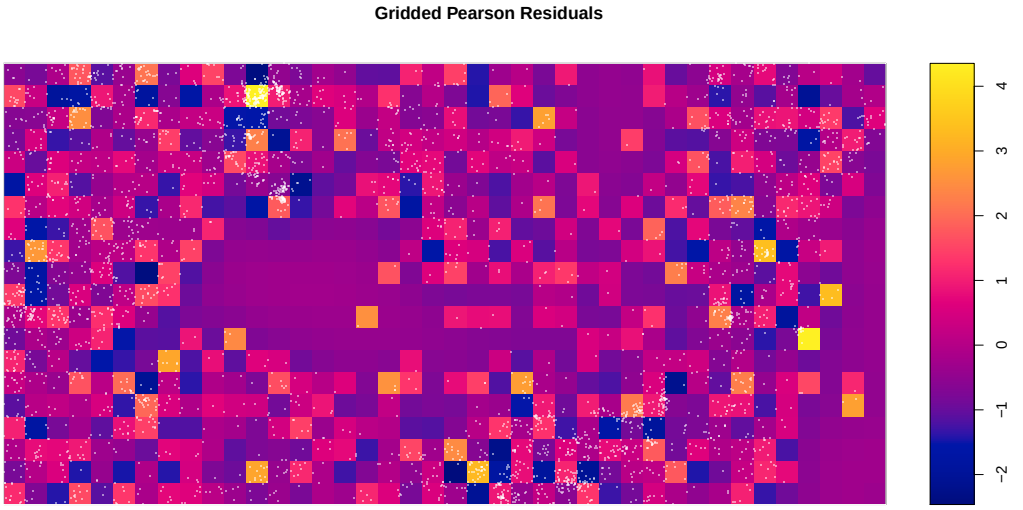
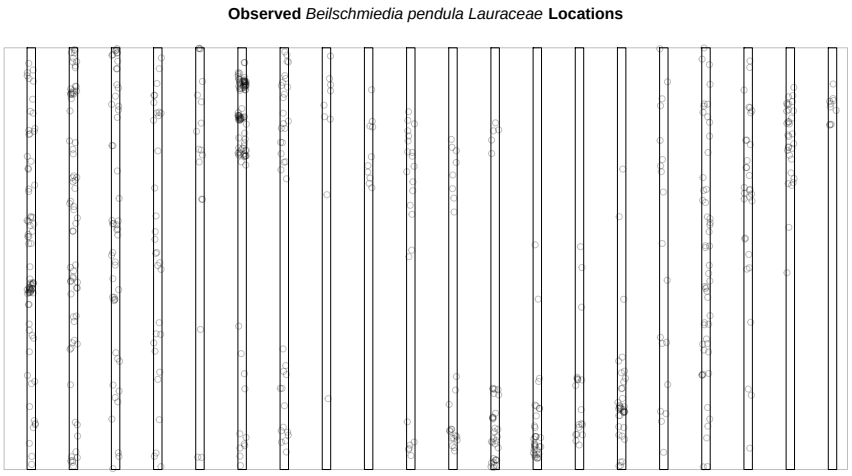
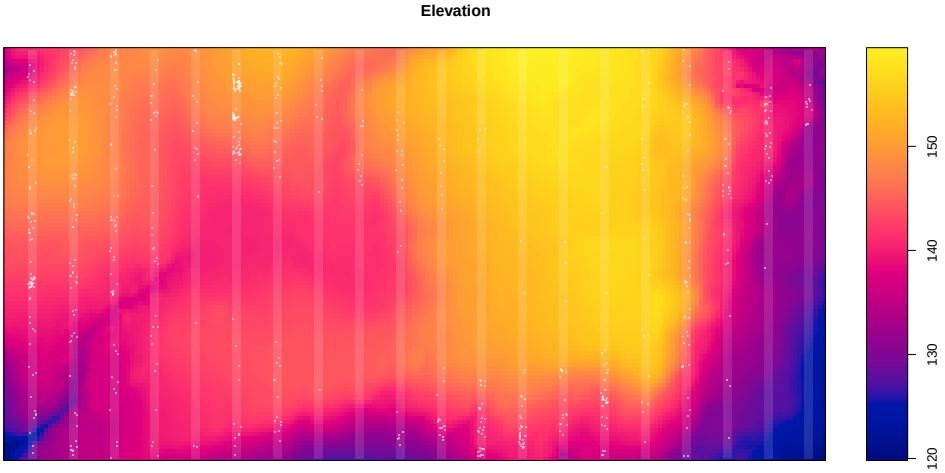


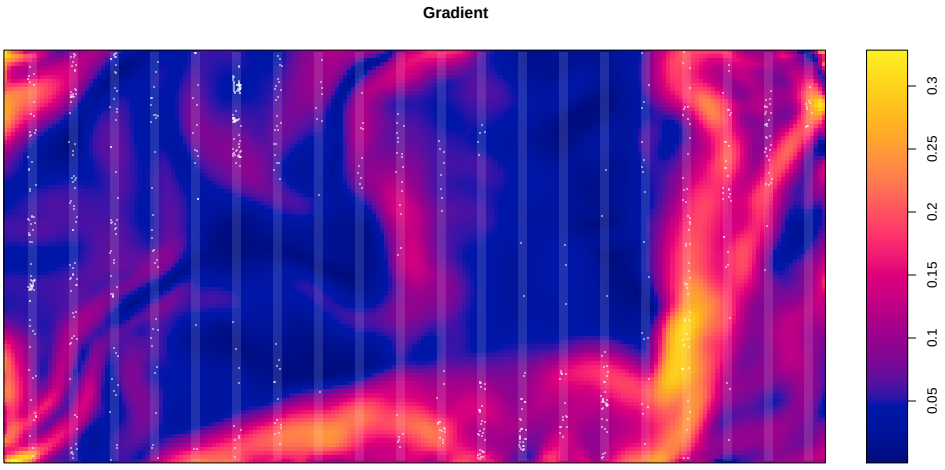
Figure 12. Pearson residuals calculated on a grid.



(a) Locations of observed trees (events).



(b) Elevation in meters, with observed region and events overlaid.



(c) Magnitude of the elevation gradient, with observed region and events overlaid.

Figure 13. Plots of the event and covariate data.

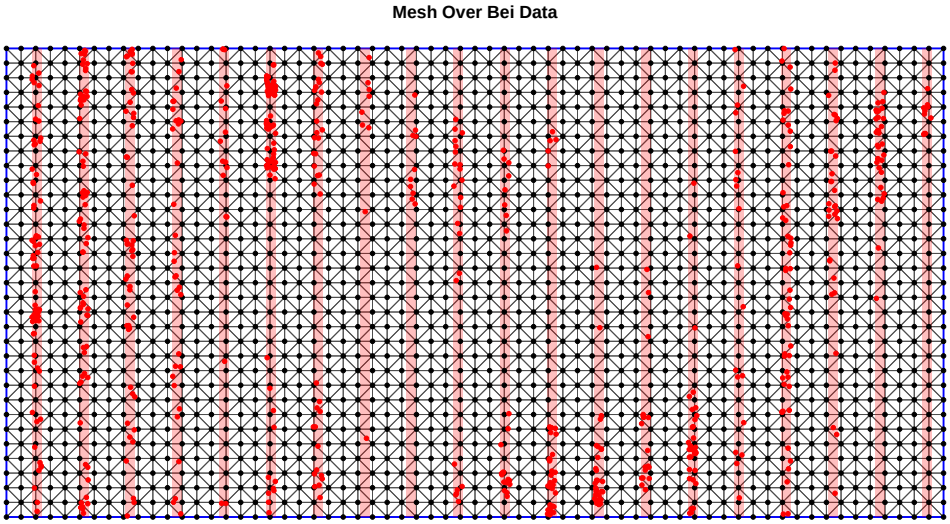


Figure 14. Triangular mesh with observed region and events overlaid.

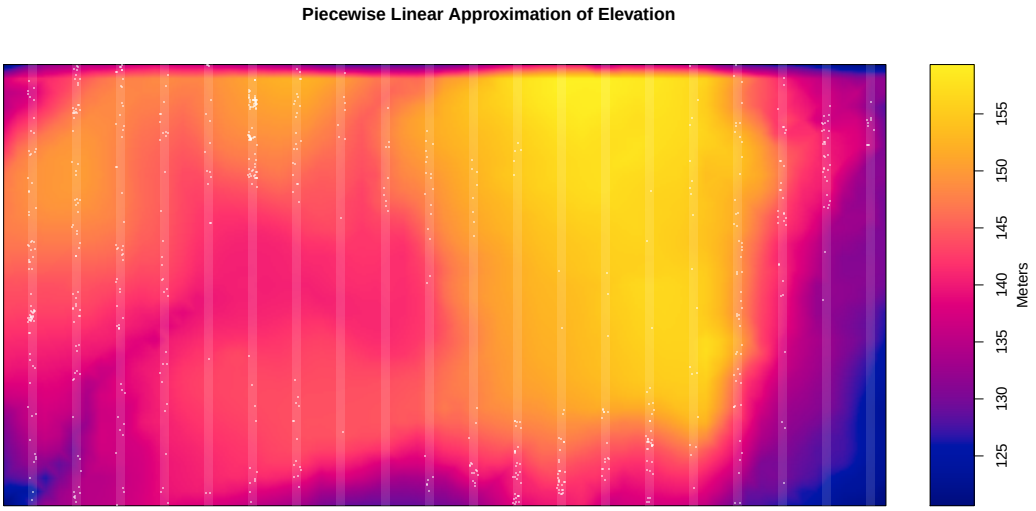


Figure 15. Piecewise linear approximation of the elevation surface.

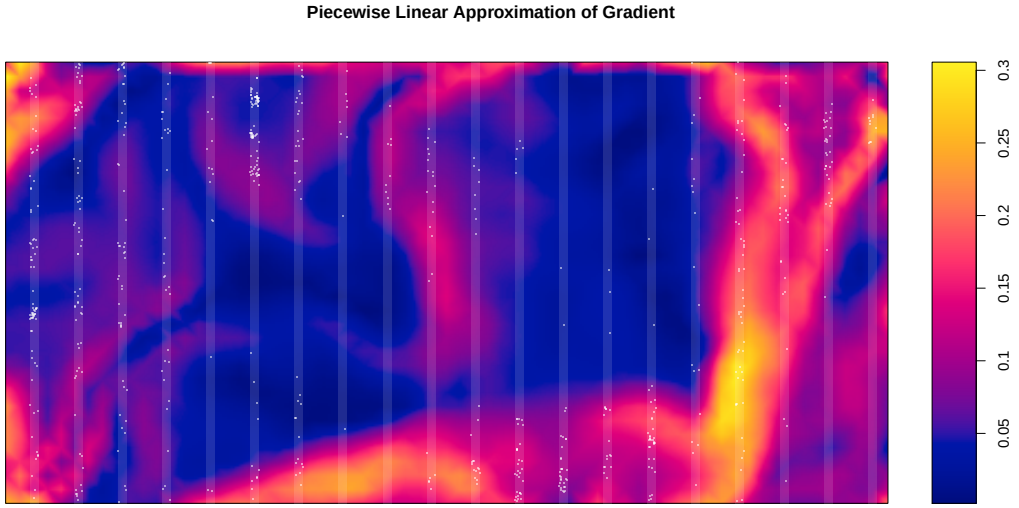


Figure 16. Piecewise linear approximation of the gradient surface.

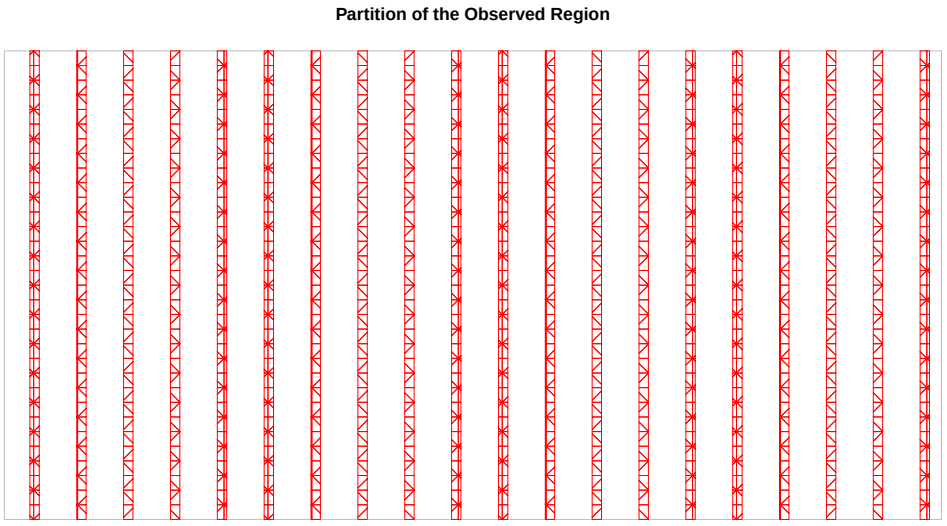


Figure 17. Partition of the observed region created by intersection with each mesh triangle.

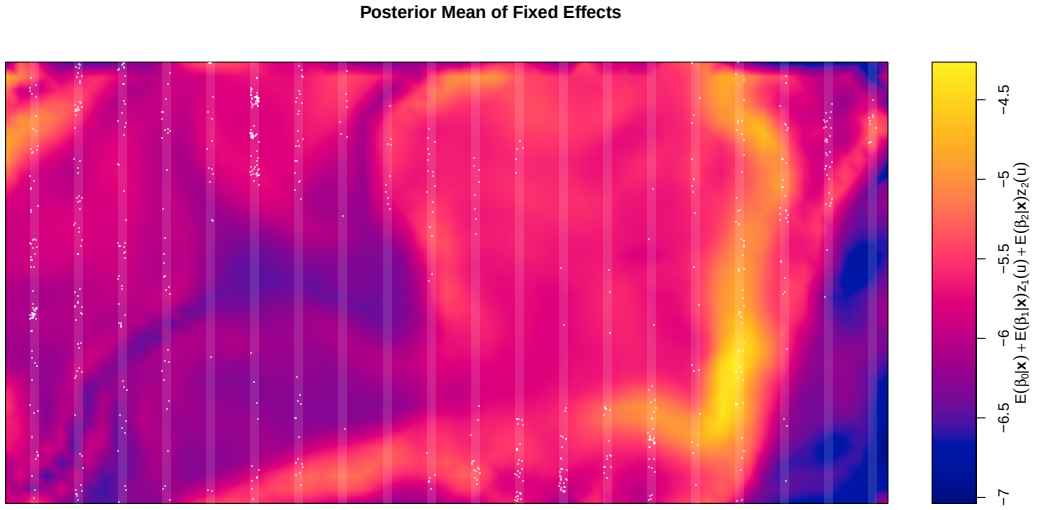


Figure 18. Posterior mean surface for the fixed effects.

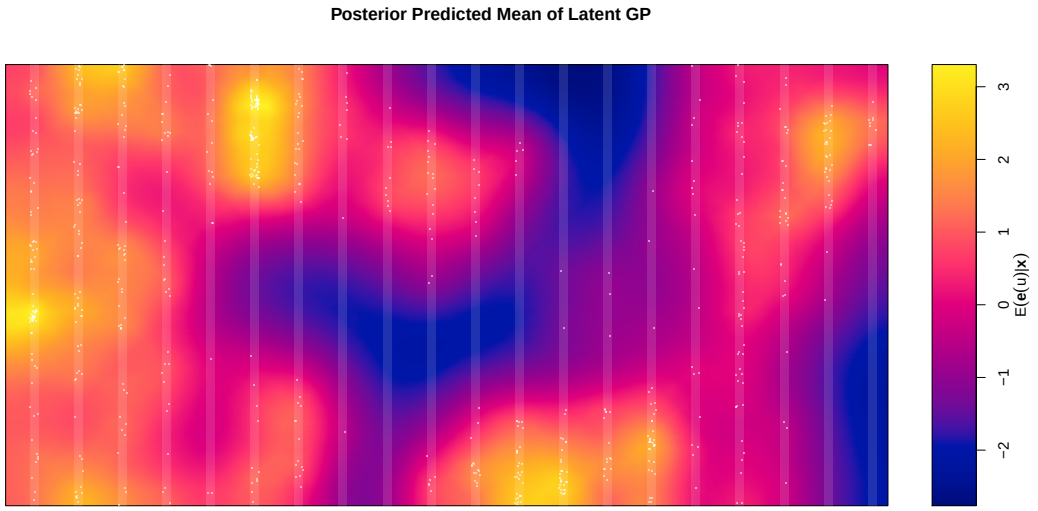


Figure 19. Posterior mean of prediction surface.

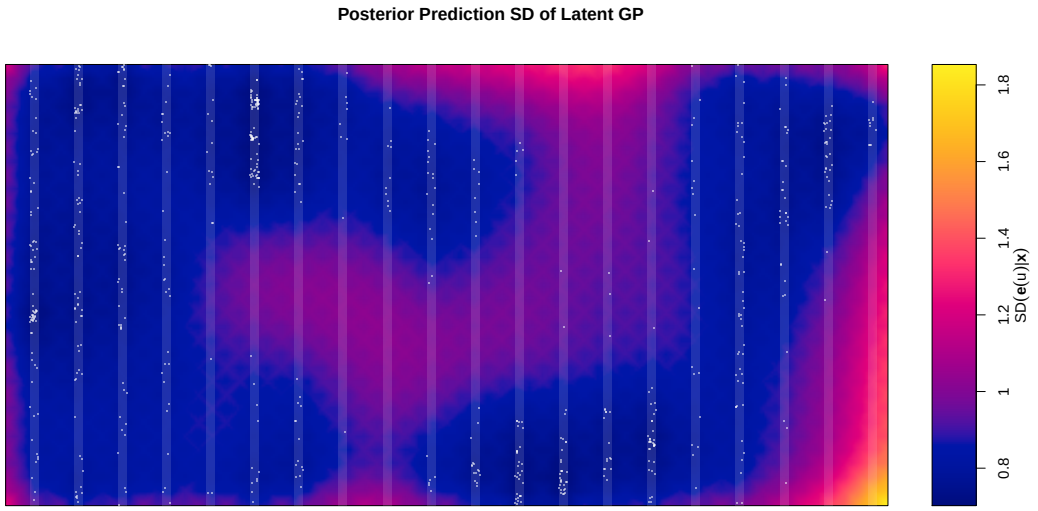


Figure 20. Pointwise posterior standard deviation of predictions.

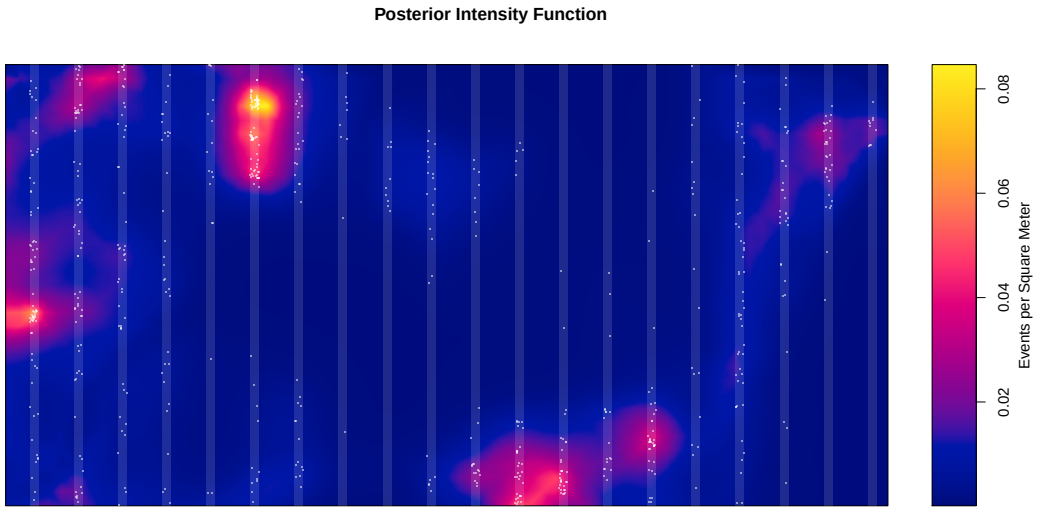


Figure 21. Posterior intensity surface in events per square meter, calculated using the piecewise linear approximate covariate surfaces and the posterior means of the intercept, coefficients, and GP.

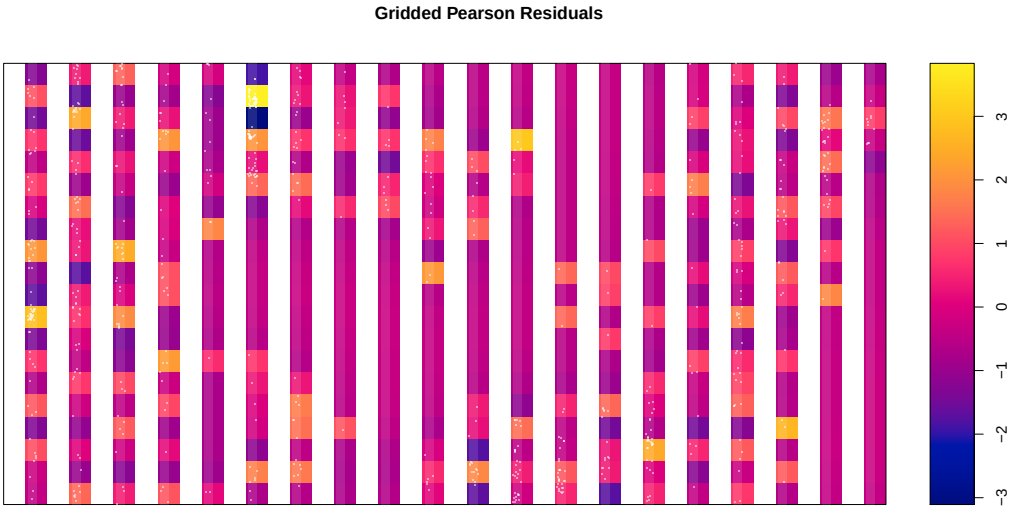


Figure 22. Pearson residuals calculated on a grid.