Journal of Statistical Software

July 2025, Volume 113, Issue 10. *doi: 10.18637/jss.v113.i10*

stopp: An R Package for Spatio-Temporal Point Pattern Analysis

Nicoletta D'Angelo
University of Palermo

Giada Adelfio
University of Palermo

Abstract

stopp is a novel R package specifically designed for the analysis of spatio-temporal point patterns which might have occurred in a subset of the Euclidean space or on some specific linear network, such as roads of a city. It represents the first package providing a comprehensive modeling framework for spatio-temporal Poisson point processes. While many specialized models exist in the scientific literature for analyzing complex spatio-temporal point patterns, we address the lack of general software for comparing simpler alternative models and their goodness of fit. The package's main functionalities include modeling and diagnostics, together with exploratory analysis tools and the simulation of point processes. A particular focus is given to local first-order and second-order characteristics. The package aggregates existing methods within one coherent framework, including those we proposed in recent papers, and it aims to welcome many further proposals and extensions from the R community.

Keywords: point patterns, simulation, model fitting, diagnostics, local analyses, spatial statistics, space-time point processes, R.

1. Introduction

stopp (D'Angelo and Adelfio 2025) is a new package in the R language for analyzing point patterns in three dimensions. The first two dimensions represent spatial components, while the third dimension is regarded as temporal. The **stopp** package has been published on the Comprehensive R Archive Network (CRAN) and is available from `https://CRAN.R-project.org/package=stopp`, version 1.0.0.

The research literature on spatial statistics provides a large body of techniques for analyzing spatio-temporal point patterns, most of which are summarized in González, Rodríguez-Cortés, Cronie, and Mateu (2016). Still, only a few of them have been implemented in software for

	Simulations	Exploratory analysis	Model fitting	Diagnostics	Linear networks	Local analyses
stpp	✓	✓				✓
stppSim	✓					
splancs		✓	✓			
stlnpp	✓	✓			✓	

Table 1: List of R packages for spatio-temporal point processes and their main functionalities.

general use. Some packages dealing with spatio-temporal point pattern exploratory analysis include **stpp** (Gabriel, Diggle, Rowlingson, and Rodriguez-Cortes 2022; Gabriel, Rowlingson, and Diggle 2013), **stppSim** (Adepeju 2022), **splancs** (Rowlingson and Diggle 2022), and **stlnpp** (Moradi, Cronie, and Mateu 2020) whose main functionalities are summarized in Table 1. These include all the tools also provided by **stopp**.

While **stpp** allows for the simulation of Poisson, inhibitive and clustered patterns, the **stppSim** package generates artificial spatio-temporal point patterns through the integration of microsimulation and agent-based models. Moreover, **splancs** fosters many tools for the analysis of both spatial and spatio-temporal point patterns, including three-dimensional kernel estimation, Monte-Carlo tests of space-time clustering, and the estimation of homogeneous spatial and temporal K functions. Regarding model fitting functions, it is only possible to fit the Diggle-Rowlingson raised incidence model. Moving to spatio-temporal point patterns on linear networks, the package **stlnpp** provides tools to visualize and analyze such patterns, implementing network-tailored kernel densities and first- and second-order summary statistics. Among those, **stpp** stands out as the most comprehensive spatio-temporal point process devoted package, furnishing statistical tools for analyzing the global and local second-order properties of spatio-temporal point processes, including estimators of the space-time inhomogeneous K function and pair correlation function. All in all, none of the spatio-temporal point process packages allows for the diagnostics of a general fitted model.

Specifically, methods for fitting both separable and non-separable spatio-temporal point process models have emerged in many disciplines, including epidemiology (Jalilian and Mateu 2021; Briz-Redón, Iftimi, Mateu, and Romero-García 2023; Schoenberg 2023), seismicity (Xiong and Zhuang 2023; Adelfio and Chiodi 2015; Siino, Mateu, and Adelfio 2016) and fire mapping (Raeisi, Bonneu, and Gabriel 2021) in the classical Euclidean space, and GPS data (D'Angelo, Adelfio, Abbruzzo, and Mateu 2022), crimes (D'Angelo, Payares, Adelfio, and Mateu 2024a), and traffic accidents (Kalair, Connaughton, and Alaimo Di Loro 2021; Chaudhuri, Juan, and Mateu 2023; Gilardi, Borgoni, and Mateu 2024; Alaimo Di Loro, Mingione, and Fantozzi 2024) in the context of linear networks. Some also included variables external to the point pattern under analysis as spatio-temporal covariates assumed to influence the occurrence of points (Adelfio and Chiodi 2021). However, most of these methods were very specific to the chosen model, and there are no software implementations of sufficient generality to fit realistic models to a real dataset. Packages dealing with spatio-temporal point process model fitting include **etasFLP** (Chiodi and Adelfio 2023, 2017; Adelfio and Chiodi 2021), mainly devoted to the estimation of the components of an ETAS (epidemic type aftershock sequence) model for earthquake description with the non-parametric background seismicity estimated through FLP (forward likelihood predictive), **ETAS** (Jalilian 2024, 2019) which fits the space-time ETAS model to earthquake catalogs using a stochastic "declustering" approach, and **stelfi** (Jones-Todd and Van Helsdingen 2023), which allows for the fitting of spatio-temporal self-exciting models and LGCPs (log-Gaussian Cox processes). Another worth-to-mention package that implements routines to simulate and fit LGCPs include **lgcp** (Taylor, Davies, Rowlingson, and Diggle 2015), which allows the fitting using methods of the moments and Bayesian inference for spatial, spatio-temporal, multivariate and aggregated point processes. This package, however, does not handle non-separable (and anisotropic) correlation structures of the covariance structure of the GRF (Gaussian random field). Turning to the context of the most simple spatio-temporal Poisson point processes, only the package **ppgam** (Youngman and Economou 2020; Wood, Li, Shaddick, and Augustin 2017) allows for the fitting of this kind of processes, but restricting the possibility to generalized additive models (GAMs), excluding more simple models like the homogeneous and inhomogeneous Poisson process. Finally, playing an important role in the R spatial statistics community outside the CRAN, the R-**INLA** package (Rue, Martino, and Chopin 2009) allows LGCP estimation within the framework of Bayesian inference for latent Gaussian models.

All the aforementioned packages leave no doubt about the widespread usage of spatio-temporal point process theory and its application by the spatial statistics community working with spatio-temporal data. However, as noted, none of those packages allows for a complete analysis of real datasets, including exploratory analysis, model fitting, and diagnostics. In particular, a considerable lack is the possibility of fitting spatio-temporal models permitting the inclusion of the dependence on external covariates. The main contribution of the **stopp** package is the collection of standard tools for a complete analysis of a spatio-temporal point pattern while also fostering functions for more detailed issues. Among the latter, we highlight some spatio-temporal local tools, which are becoming more and more used in real spatio-temporal data analysis. The **stopp** package further allows for the integration with the previously mentioned packages by only requesting the estimated intensity to be diagnosed.

One of the main contributions of **stopp** is embodied in the `stppm()` function, which provides the first choice in R to fit general spatio-temporal Poisson point process models. These models include both homogeneous and inhomogeneous processes, with options for parametric and non-parametric specifications of coordinates, external covariates, and multitype cases. This is achieved following a cubature scheme (D'Angelo, Adelfio, and Mateu 2023b; D'Angelo and Adelfio 2024a), which extends Berman and Turner (1992)'s and Baddeley, Coeurjolly, Rubak, and Waagepetersen (2014)'s algorithm from the purely spatial to the spatio-temporal context.

Another important contribution of **stopp** lies in the second-order based diagnostic techniques, which only utilize fitted intensities, making them applicable to any fitted model (whether Euclidean or network-based), even to those beyond the scope of **stopp**. This versatility is a significant strength of **stopp** and enhances the linkage to other point process packages. As far as we are aware, there is currently no software implementation of any technique for fitting spatio-temporal point process models at the level of generality and flexibility that we propose. This is only achieved by **spatstat** (Baddeley and Turner 2005) in the purely spatial point process framework.

stopp also provides codes related to methods and models for analyzing complex spatio-temporal point processes proposed in the papers Siino, Adelfio, and Mateu (2018a); Siino, Rodríguez-Cortés, Mateu, and Adelfio (2018b); Adelfio, Siino, Mateu, and Rodríguez-Cortés (2020); D'Angelo, Adelfio, and Mateu (2021, 2023a); D'Angelo *et al.* (2023b). A particular focus is given to both first-order and second-order *local* characteristics. Regarding first-order estimation, **stopp** allows for the estimation of both local spatio-temporal Poisson and local log-Gaussian Cox processes models, that is, with spatio-temporal varying parameters. As previously mentioned, an R package that implements routines to fit spatio-temporal LGCPs is **lgcp**, where the *minimum contrast* method is used to estimate parameters assuming a separable structure of the covariance of the Gaussian random field. In addition, **stopp** also handles non-separable correlation structures of the covariance structure of the GRF by means of the *joint minimum contrast* procedure (Siino *et al.* 2018a), with the further advantage of giving the possibility of estimating both (or either) first-order and second-order parameters locally (D'Angelo *et al.* 2023b).

The level of generality achieved by **stopp** is due to the integration with other well-established point processes R packages. The main dependencies of the **stopp** package are indeed **spatstat**, **stpp**, and **stlnpp**. We exploit many functions from **spatstat** when purely spatial tools are needed while performing spatio-temporal analyses. Furthermore, we rely on **stpp**'s both global and local K functions and pair correlation functions (pcfs) estimators, to perform diagnostics based on second-order summary statistics (Gabriel and Diggle 2009; Adelfio *et al.* 2020). From **stlnpp**, we borrow the linear networks estimators counterparts (Moradi and Mateu 2020).

The ambitious aim of this package is to contribute to the existing literature by gathering many of the most widespread methods for the analysis of spatio-temporal point processes into a unique package, which is intended to host many further extensions.

The outline of the paper conceptually follows the package structure, illustrated in Table 1.

First, in Section 2, we introduce the main classes of objects for handling spatio-temporal point pattern objects. Some available datasets are introduced in Section 3. Then, we present some novel functions to simulate specific classes of point processes in Section 4. We then move to Section 5 with exploratory analysis carried out through the local indicators of spatio-temporal association (LISTA) functions on linear networks, newly available in R. In the same exploratory context, we illustrate the function to perform a local test for assessing the presence of local differences in two point patterns. Then, in Section 6, a large body of functions available for fitting models is presented, including the general Poisson model, which includes both homogeneous or inhomogeneous specification of the first-order intensity function that can depend on semiparametric effects of both coordinates or external covariates. The multitype point process is also available. There is also the possibility of fitting a separable Poisson process model on either the Euclidean space and networks, and LGCPs. Moreover, we illustrate some functions to fit local models, including the generic Poisson process and LGCPs. Finally, methods to perform global and local diagnostics on both models for point patterns on planar and linear network spaces are presented in Section 7. The paper ends with some future developments in Section 8.

2. Data types

2.1. Spatio-temporal point patterns

The `stp()` function creates a '`stp`' object as a dataframe with three columns: `x`, `y`, and `t`. If the linear network `L`, of class '`linnet`' of the **spatstat** package, is also provided, a '`stlp`' object is created instead. This class of objects are equipped with the `print`, `summary`, and `plot` methods. The creation of these two types of objects comes as follows,

	Data types
`stcov()`	Create and interpolate spatio-temporal covariates on a regular grid
`stp()`	Create '`stp`' and '`stlp`' objects for point patterns storage*
`stpm()`	Create '`stpm`' and '`stlpm`' objects for marked point patterns storage*
	Datasets
`chicagonet`	Rescaled roads of Chicago (Illinois, USA)
`greececatalog`	Catalog of Greek earthquakes
`valenciacrimes`	Crimes in Valencia in 2019
`valencianet`	Roads of Valencia, Spain
	Simulations
`rETASlp()`	Simulate a spatio-temporal ETAS process on a linear network
`rETASp()`	Simulate a spatio-temporal ETAS process
`rstlpp()`	Simulate spatio-temporal Poisson point patterns on a linear network
`rstpp()`	Simulate spatio-temporal Poisson point patterns
	Eploratory analysis
`localSTLginhom()`	Estimate the local inhomogeneous spatio-temporal pcfs on a linear network
`localSTLKinhom()`	Estimate the local inhomogeneous spatio-temporal K functions on a linear network
`localtest()`	Perform the test of local structure for spatio-temporal point processes*
	Model fitting
`locstppm()`	Fit a local spatio-temporal Poisson process
`sepstlppm()`	Fit a separable spatio-temporal Poisson process on a linear network
`sepstppm()`	Fit a separable spatio-temporal Poisson process
`stlgcppm()`	Fit global or local spatio-temporal log-Gaussian Cox processes
`stppm()`	Fit a spatio-temporal Poisson process
	Diagnostics
`globaldiag()`	Perform global diagnostics of a spatio-temporal point process models*
`infl()`	Display outlying LISTA functions*
`localdiag()`	Perform local diagnostics of spatio-temporal point process models*

Table 2: List of functions in **stopp**, excluding S3 methods. The symbol * indicates the functions implemented to work both on point patterns in Euclidean spaces and linear networks.

```
R> install.packages("stopp")
R> library("stopp")
R> set.seed(2)
R> df <- data.frame(runif(100), runif(100), runif(100))
R> stp1 <- stp(df)
R> stp1

Spatio-temporal point pattern
100 points
Enclosing window: rectangle = [0.007109, 0.9889022] x [0.0136249, 0.9806] units
Time period: [0.013, 0.991]
```

The following command produces Figure 1.

```
R> plot(stp1)
```

The left and central panels produced by `plot.stp` and `plot.stlp` show the spatio-temporal and the purely spatial locations of the points. The right panel displays the cumulative sum

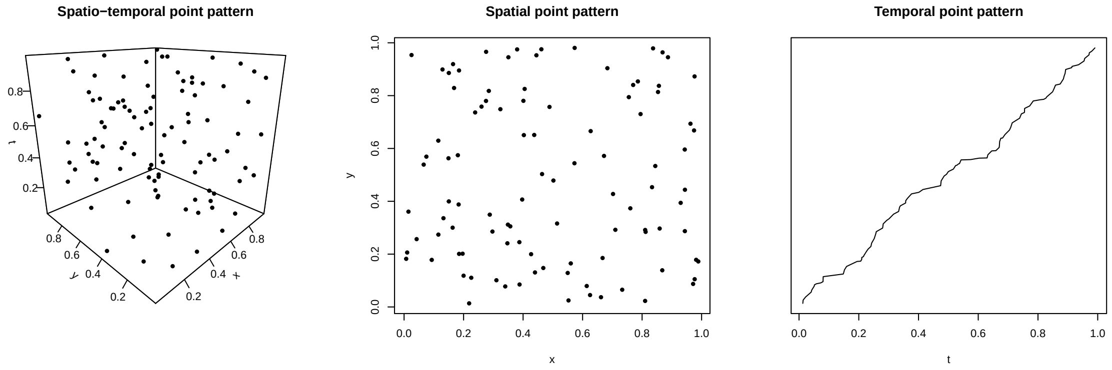

Figure 1: Output of the `plot.stp()` function applied to a simulated spatio-temporal point pattern.

of the temporal locations ordered in time. For this reason, the temporal cumulative plot of a homogeneous point pattern will be quadratic, and not linear as the intensity trend would be instead. By setting the argument `tcum` equal to `FALSE`, the temporal pattern is displayed instead (Figure 2), only advisable when dealing with few points.

```
R> set.seed(2)
R> df_net <- data.frame(runif(100, 0, 0.85), runif(100, 0, 0.85),
+    runif(1000.85), runif(100))
R> stlp1 <- stp(df_net, L = chicagonet)
R> stlp1

Spatio-temporal point pattern on a linear network
100 points
Linear network with 338 vertices and 503 lines
Enclosing window: rectangle = [0, 0.9996963] x [0, 0.8763407] units
(one unit = 1281.98625717162 feet)
Time period: [0.013, 0.991]
```

The following command produces Figure 2.

```
R> plot(stlp1, tcum = FALSE)
```

2.2. Marked point processes

If additional variables are attached to the points of the pattern, it is possible to build a spatio-temporal marked point pattern as a '`stpm`' object (or '`stlpm`', if occurred on a linear network). For the multitype point process, we choose the same approach of continuous marks, that is, collecting all the points together in one point pattern and labeling each point by the type to which they belong. An advantage of this approach is that it is easy to deal with multitype point patterns with more than two types.

Below is an example of a point pattern characterized by both a continuous mark and a categorical mark, rendering it a multitype point pattern, as shown in Figure 3.

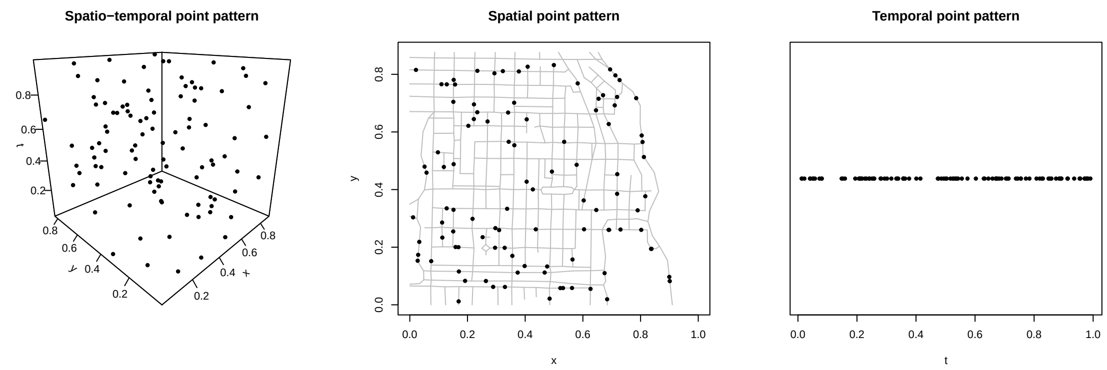

Figure 2: Output of the `plot.stlp()` function applied to a simulated spatio-temporal point pattern on a linear network.

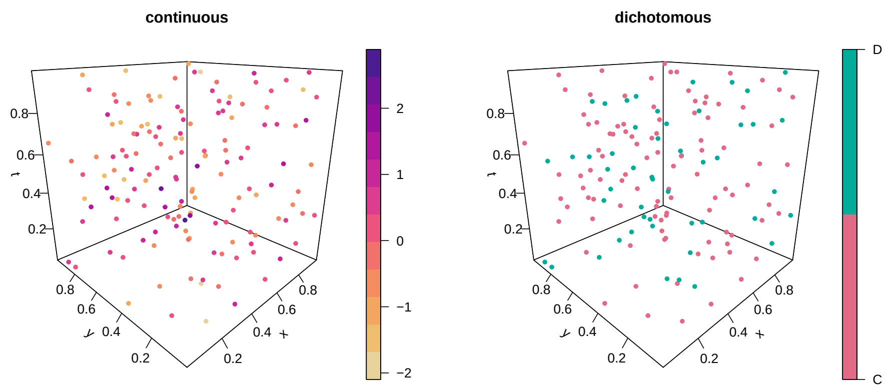

Figure 3: Output of the `plot.stpm()` function applied to a simulated spatio-temporal point pattern marked by a continuous and a categorical mark.

```
R> set.seed(2)
R> dfA <- data.frame(x = runif(100), y = runif(100), t = runif(100),
+    m1 = rnorm(100), m2 = rep(c("C"), times = 100))
R> dfB <- data.frame(x = runif(50), y = runif(50), t = runif(50),
+    m1 = rnorm(25), m2 = rep(c("D"), times = 50))
R> stpm2 <- stpm(rbind(dfA, dfB), names = c("continuous", "dichotomous"))
R> plot(stpm2)
```

2.3. Spatio-temporal covariates

The class ‘`stcov`’ is reserved to be used for creating and interpolating potential spatio-temporal covariates, intended to be included in the `formula` of the main function of **stopp**: `stppm()`.

Figure 4 displays an example of a simulated spatio-temporal covariate (on the left panel) and the interpolated covariate resulting from the application of the `stcov()` function (right panel).

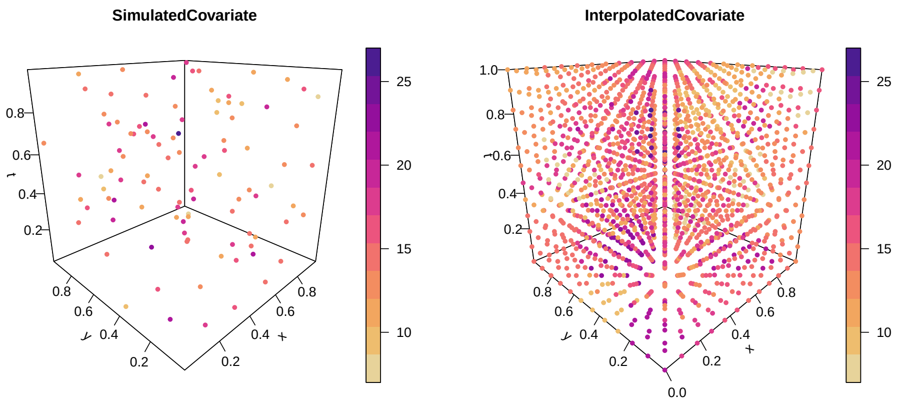

Figure 4: Simulated and interpolated covariate with the `stcov()` function.

This preliminary procedure is a device to speed estimation in `stppm()`. Indeed, since the covariate values must be known at every data and dummy point, an advisable approach is to use interpolation (Tarantino, D'Angelo, and Adelfio 2024; D'Angelo and Adelfio 2024a). We employ a spatial smoothing of the numeric values observed at the covariate locations $\hat{Z}(x, y, t) = \sum_{j=1}^{J} w_j(x, y, t)Z(x_j, y_j, t_j)/\sum_{j=1}^{J} w_j(x, y, t)$, where $\hat{Z}(x, y, t)$ is the interpolated value at new location (x, y, t), J is the number of covariate locations, and $Z(x_j, y_j, t_j)$ is the covariate value at the observed location (x_j, y_j, t_j). Particularly, we set $w_j(x, y, t) = \left(\sqrt{(x - x_j)^2 - (y - y_j)^2 - (t - t_j)^2}\right)^{-p}$, meaning that we employ inverse-distance weighting (Shepard 1968), where p is the power of the Euclidean distance between (x, y, t) and (x_j, y_j, t_j). To avoid a different interpolation at each model fit, we, therefore, interpolate only once when employing the `stcov()` function, making a very fine regular grid, and then just attribute to the data or dummy point the covariate value of the closest grid point in `stppm()`.

```
R> set.seed(2)
R> df <- data.frame(runif(100), runif(100), runif(100), rpois(100, 15))
R> sim_cov <- stcov(df, interp = FALSE, names = "SimulatedCovariate")
R> interp_cov <- stcov(df, mult = 20, names = "InterpolatedCovariate")
R> plot(sim_cov)
R> plot(interp_cov)
```

3. Datasets

The package is furnished with the `greececatalog` dataset from the Hellenic Unified Seismic Network (HUSN). in the '`stp`' format containing the catalog of Greek earthquakes of magnitude at least 4.0 from 2005 to 2014 (Figure 5).

```
R> data("greececatalog", package = "stopp")
R> plot(greececatalog)
```

A dataset of crimes that occurred in Valencia, Spain, in 2019 is also available as a '`stpm`' object, together with the linear network of class '`linnet`' of the Valencian roads, named

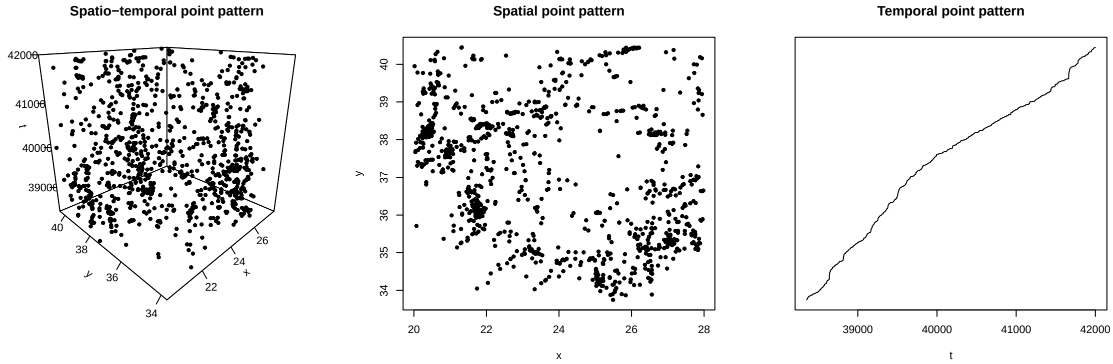

Figure 5: Plots of `greececatalog` data provided in '`stp`' format in the **stopp** package.

`valenciacrimes` (Figure 6), and `valencianet` (right panel of Figure 7), respectively. The marks of this dataset include the month, week, day, and hour of crime occurrences, and many distances to the closest points of interest, which can be assumed to have influenced the occurrence of crimes.

```
R> data("valenciacrimes", package = "stopp")
R> plot(valenciacrimes)
R> data("chicagonet", package = "stopp")
R> data("valencianet", package = "stopp")
R> plot(chicagonet)
R> plot(valencianet)
```

Finally, the linear network of class '`linnet`' of the roads of Chicago (Illinois, USA) close to the University of Chicago is also available (left panel of Figure 7). It represents the linear network of the Chicago dataset published and analyzed in Ang, Baddeley, and Nair (2012). The window has been rescaled to be enclosed in a unit square.

4. Simulations

Stochastic simulation of spatio-temporal point process models is another area where the richness of the theoretical literature contrasts with the scarcity of stable public domain software.

We contribute to the framework of simulating spatio-temporal point process models with novel designed functions. The first contribution is given by the possibility of simulating Poisson patterns as '`stp`' objects, with inhomogeneous intensity by means of the `rstpp()` function, as follows.

```
R> rstpp(lambda = 500)
R> rstpp(lambda = function(x, y, t, a) {exp(a[1] + a[2] * x)}, par = c(2, 6))
```

The above code simulates two spatio-temporal point patterns. The first one follows the homogeneous intensity $\lambda(x, y, t) = 500$, while the second one is generated from the inhomogeneous intensity $\lambda(x, y, t) = \exp(2+6x)$. In the former case, the simulated pattern will be completely random, while the second one will show a trend increasing along the x coordinate.

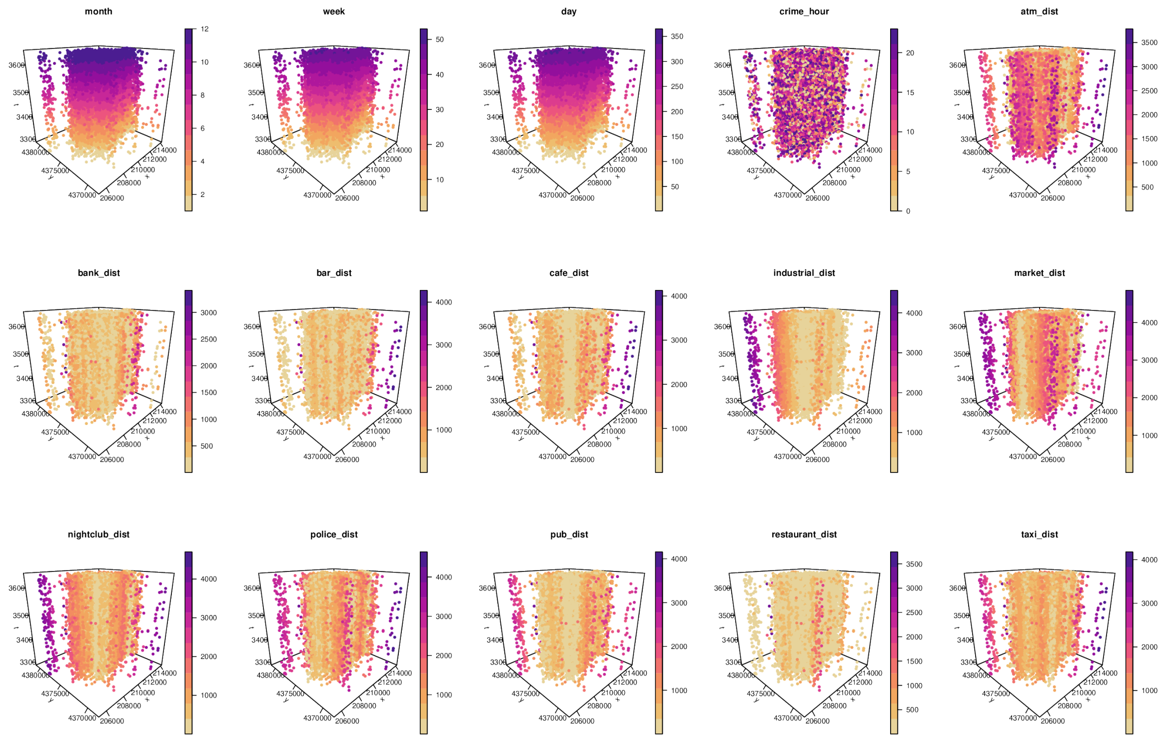

Figure 6: Plot of `valenciacrimes` data provided in '`stpm`' format in the **stopp** package.

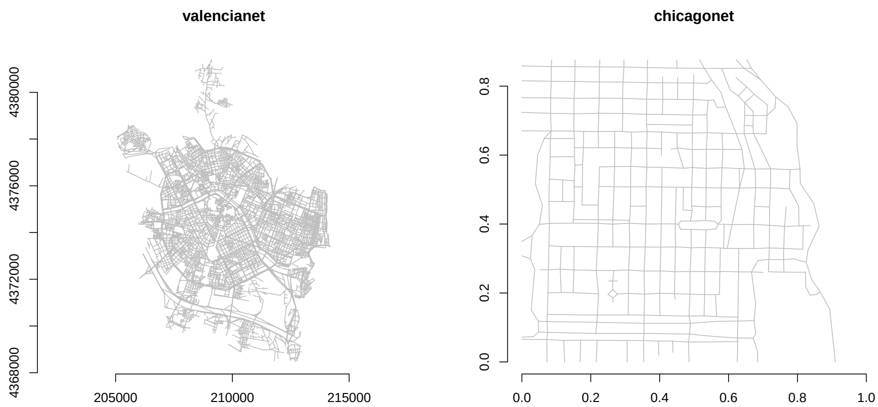

Figure 7: Plot of `valencianet` and `chicagonet` linear networks provided in the **stopp** package.

The `rstlpp()` function creates a '`stlp`' object instead, simulating a spatio-temporal Poisson point pattern on a linear network.

Then, `rETASp()` simulates a spatio-temporal point pattern following an ETAS process as in Adelfio and Chiodi (2021). Figure 8 shows an example.

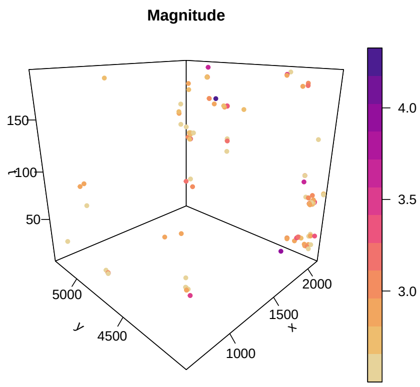

Figure 8: Plot of a spatio-temporal ETAS point pattern simulated by the `rETASp()` function.

```
R> set.seed(95)
R> X <- rETASp(c(0.1293688525, 0.003696, 0.013362, 1.2, 0.424466, 1.164793),
+    betacov = 0.5, xmin = 600, xmax = 2200, ymin = 4000, ymax = 5300)
R> plot(X)
```

Finally, `rETAStlp()` function creates a '`stlp`' object, simulating a spatio-temporal ETAS process on a linear network. The simulation scheme in this case is adapted for the space location of events to be constrained on a linear network, being firstly introduced and employed for simulation studies by D'Angelo *et al.* (2021).

All the simulation functions are equipped with a `seed` argument, allowing to specify the seed for reproducing the same simulation. Note that we have set specific seeds throughout the paper to ensure the reproducibility of the codes.

5. Exploratory analysis

The exploratory analysis tools of **stopp** build upon the local indicators of spatio-temporal association (LISTA) functions, which are defined as a set of functions that are individually associated with each one of the points of the point pattern, and can provide information about the local behavior of the pattern (Anselin 1995; Siino *et al.* 2018b).

In particular, the package implements the local spatio-temporal K functions and pair correlation functions on linear networks, introduced in D'Angelo *et al.* (2021). These are estimated by means of the function `localSTLKinhom()` and `localSTLginhom()`, respectively, and can be displayed through the `plot()` function. Since any of `localSTLKinhom()` and `localSTLginhom()` will produce a list of K (or pcf) functions, one for each point in the observed point pattern, it is not possible to display them all together. Therefore, the argument `id` is reserved for a vector for identifying which points to display the LISTA function of. Below is an example to display the local K functions of the first three points stored in the '`stp`' object passed to the `localSTLKinhom()` function, as shown in Figure 9.

```
R> set.seed(2)
R> df_net <- data.frame(runif(25, 0, 0.85), runif(25, 0, 0.85), runif(25))
```

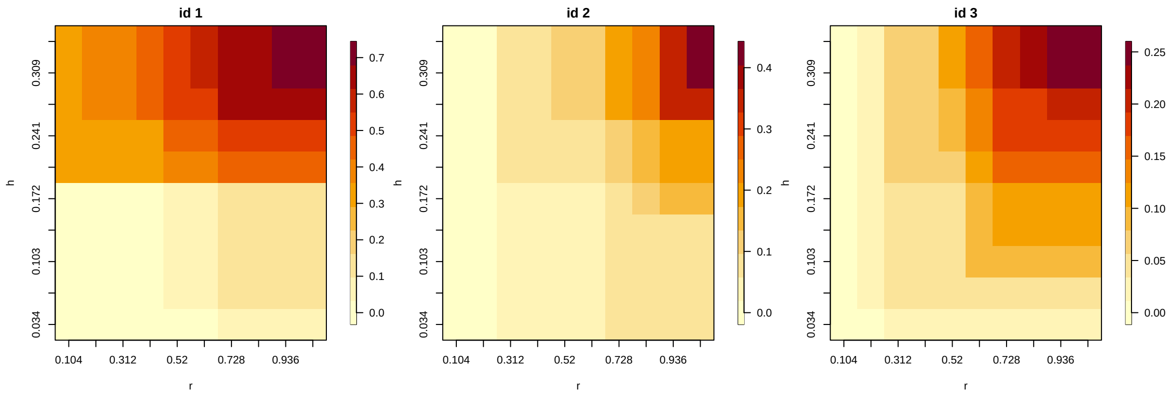

Figure 9: Output of the `plot.lista()` function.

```
R> stlp1 <- stp(df_net, L = chicagonet)
R> lambda <- rep(diff(range(stlp1$df$x)) * diff(range(stlp1$df$y)) *
+    diff(range(stlp1$df$t)) / spatstat.geom::volume(stlp1$L),
+    nrow(stlp1$df))
R> k <- localSTLKinhom(stlp1, lambda = lambda, normalize = TRUE)
R> plot(k, id = 1:3)
```

5.1. Local test

The function `localtest()` performs the permutation test of the local structure of spatio-temporal point pattern data proposed in Siino *et al.* (2018b). The network counterpart is also implemented, following D'Angelo *et al.* (2021). This test detects local differences in the second-order structure of two observed point patterns **x** and **z** occurring in the same space-time region. The test is performed for spatio-temporal point patterns, as in Siino *et al.* (2018b), on two objects of class '`stp`'. The employed LISTA functions $\hat{L}^{(i)}$ are the local K functions introduced in Adelfio *et al.* (2020) and computed by the function `KLISTAhat()` of the **stpp** package (Gabriel *et al.* 2013). If `localtest()` is applied to '`stlp`' objects, that is, on two spatio-temporal point patterns observed on the same linear network `L`, the local K functions used are the ones proposed in D'Angelo *et al.* (2021), implemented in the `localSTLKinhom()` function of **stopp**. Details on the performance of the test are found in Siino *et al.* (2018b) and D'Angelo *et al.* (2021) for Euclidean and network spaces, respectively. Alternative LISTA functions that can be employed to run the test are `LISTAhat()` of **stpp** and `localSTLginhom()` of **stopp**, that is, the pcfs on Euclidean space and linear networks, respectively, fixing the argument `method = "g"`. The class of these objects is called '`localtest`', and it is equipped with the methods `print`, `summary`, and `plot`, working as follows. In Figure 10, an output example of the function `plot.localtest()` is reported. A background and an alternative patterns can be obtained, and the local test can be run as follows:

```
R> set.seed(2)
R> X <- rstpp(lambda = function(x, y, t, a) {exp(a[1] + a[2] * x)},
+    par = c(.005, 5))
R> set.seed(2)
```

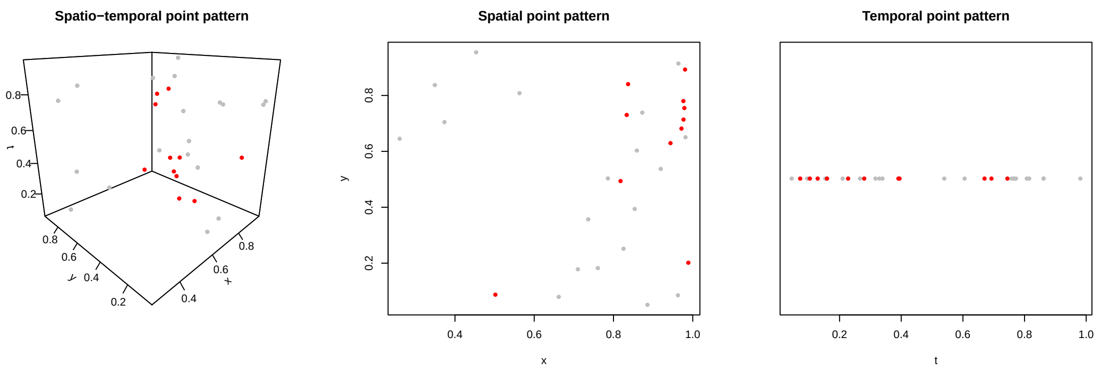

Figure 10: Output of the `plot.localtest()` function.

```
R> Z <- rstpp(lambda = 30)
R> test <- localtest(X, Z, method = "K", k = 3)
R> test

Test for local differences between two
spatio-temporal point patterns
--------------------------------------
Background pattern X: 30
Alternative pattern Z: 25

11 significant points at alpha = 0.05

R> plot(test)
```

6. Model fitting

In this section, we outline the main functions to fit different specifications of inhomogeneous spatio-temporal Poisson process models.

6.1. Inhomogeneous spatio-temporal Poisson point processes

The primary fitting function of **stopp** is the function `stppm()`. It fits a Poisson process model (Diggle 2013) to an observed spatio-temporal point pattern stored in a '`stp`' object, assuming the template Poisson process model with a parametric first-order intensity function

$$\lambda(x, y, t; \boldsymbol{\theta}), \qquad (x, y) \in W, \quad t \in T, \quad \boldsymbol{\theta} \in \Theta,$$

where (x, y) and t are the spatial and temporal coordinates in the spatial and temporal regions W and T, and $\boldsymbol{\theta}$ are the parameters to be estimated.

For the homogeneous case, we can fit

$$\lambda(x, y, t) = \lambda = \exp(\theta_0)$$

as follows:

```
R> set.seed(2)
R> ph <- rstpp(lambda = 200)
R> hom1 <- stppm(ph, formula = ~ 1, seed = 2)
R> hom1

Homogeneous Poisson process
with Intensity: 202.093

Estimated coefficients:
(Intercept)
      5.309
```

Therefore, the only mandatory arguments are the spatio-temporal point pattern '`stp`', and the formula specifying the linear predictor to consider. Note that the function `stppm()` is also equipped with the argument `seed` since the generation of the dummy points depends on the `rstpp()` function in turn. To make the code results reproducible, we set the seed in the examples illustrated with `stppm()`, and in all the functions based on the generation of some dummy points.

In point process theory, it is common not to have available auxiliary covariates, so many point process models only resort to the Cartesian coordinates.

For the inhomogeneous case, we can simulate:

```
R> set.seed(2)
R> pin <- rstpp(lambda = function(x, y, t, a) {exp(a[1] + a[2] * x)},
+     par = c(2, 6))
```

The following code fits a model with the following intensity specification

$$\lambda(x, y, t) = \exp(\theta_0 + \theta_1 x)$$

estimating $\hat{\theta}_0 = 2.18$ and $\hat{\theta}_1 = 5.783$.

```
R> inh1 <- stppm(pin, formula = ~ x, seed = 2)
R> inh1

Inhomogeneous Poisson process
with Trend: ~x

Estimated coefficients:
(Intercept)           x
      2.180       5.783
```

Estimation is performed by fitting a generalized linear mixed model (Breslow and Clayton 1993), in which the linear predictor can contain random effects in addition to the usual fixed effects, employing a spatio-temporal cubature scheme (D'Angelo *et al.* 2023b; D'Angelo and Adelfio 2024a). The `stppm()` function has an argument `method` which selects the parameter estimation technique. Another option is `method = "lsr"` representing the spatio-temporal

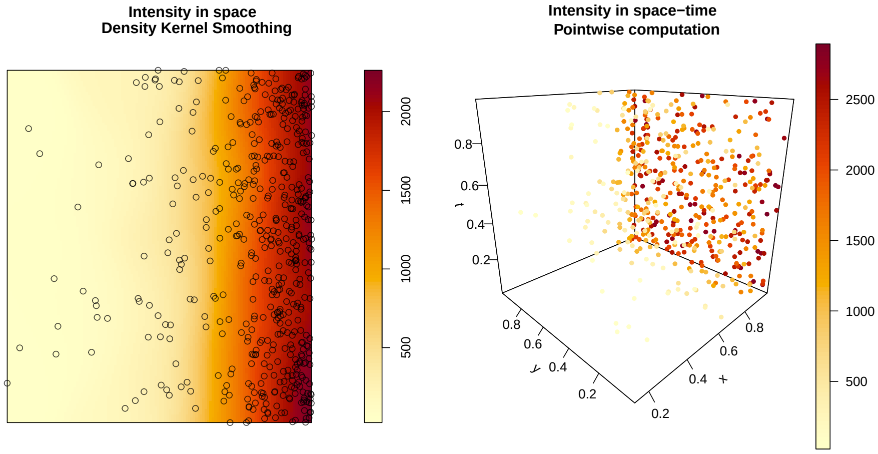

Figure 11: Output of the `plot.stppm()` function applied to a fitted non-parametric model.

extension of logistic spatial regression (Baddeley *et al.* 2014). The choice of the `gam()` function of the **mgcv** package (Wood *et al.* 2017) is due to the possibility of including both smooth terms of the covariates (typical in point process theory for the spatio-temporal coordinates) and random effects. The latter comes in aid when wishing to fit a multitype point pattern, where basically each type of the categorical mark believed to represent the type will have its own set of fitted parameters (D'Angelo and Adelfio 2024a).

For instance, the following code fits an inhomogeneous Poisson point process of the form

$$\lambda(x, y, t) = \exp(f(x, y))$$

with $f(\cdot)$ a non-parametric function for the spatial coordinates estimated through thin plate regression splines (Wood 2003) with 30 knots.

Figure 11 shows the estimated intensity in space (left panel) and in space and time (right panel).

```
R> inh2 <- stppm(pin, formula = ~ s(x, y, bs = "tp",  k = 30), seed = 2)
R> plot(inh2)
```

6.2. Spatio-temporal Poisson point processes with external covariates

Another peculiar capability in **stopp** is the possibility of fitting Poisson point process models with a first-order intensity function depending on external spatio-temporal covariates as

$$\lambda(x, y, t; \boldsymbol{\theta}) = \exp(\boldsymbol{\theta}^\top \mathbf{Z}(x, y, t)),$$

where $\mathbf{Z}(x, y, t) = \{Z_1(x, y, t), \ldots, Z_p(x, y, t)\}$ are p known spatio-temporal covariate functions, and $\boldsymbol{\theta}$ their associated unknown parameters to estimate.

It is very uncommon to have the covariate values observed at the point pattern locations. Nevertheless, their values must be known at points and some other locations in the analyzed region for inferential purposes. This is achieved by preliminary interpolating the covariate values through the `stcov()` function, as shown in the example below.

Let's first simulate some covariates.

```
R> set.seed(2)
R> df1 <- data.frame(runif(100), runif(100), runif(100), rpois(100, 15))
R> df2 <- data.frame(runif(100), runif(100), runif(100), rpois(100, 15))
```

Next, it is advisable to interpolate them along a finer and more regular grid with `stcov()`, which will return a '`stcov`' object.

```
R> obj1 <- stcov(df1, names = "cov1")
R> obj2 <- stcov(df2, names = "cov2")
```

Then, we have to store all of the covariates into a unique list.

```
R> covariates <- list(cov1 = obj1, cov2 = obj2)
```

Note that this is necessary because, often, the covariate's sites are not the same among different covariates. To then fit a spatio-temporal Poisson point process model depending on a spatial coordinate and a spatio-temporal covariate, such as

$$\lambda(x, y, t) = \exp(\theta_0 + \theta_1 x + \theta_2 \texttt{cov2}(x, y, t)),$$

we have to input the list of '`stcov`' objects into the `covs` argument of `stppm()` and specify `spatial.cov = TRUE`, as the following code illustrates.

```
R> inh3 <- stppm(pin, formula = ~ x + cov2, covs = covariates,
+    spatial.cov = TRUE, seed = 2)
R> inh3

Inhomogeneous Poisson process
with Trend: ~x + cov2

Estimated coefficients:
(Intercept)           x        cov2
      2.116       5.791       0.004
```

6.3. Multitype spatio-temporal Poisson point processes

Finally, `stppm()` offers the capability to fit multitype Poisson point process models.

If the multitype point process has $m = 1, 2, \ldots, M$ types, the (marginal) intensity is

$$\lambda(x, y, t) = \sum_{m=1}^{M} \lambda(x, y, t, m)$$

where $\lambda(x, y, t, m)$ is the intensity function for locations (x, y, t) and mark type m.

As an example, the following codes simulate a multitype point pattern with points belonging to two different types, named A and B, with 100 and 50 points each (Figure 12).

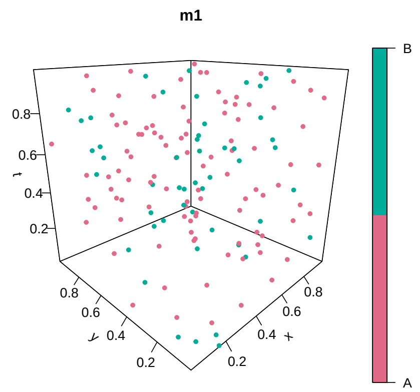

Figure 12: Plot of a simulated multitype point pattern with only two types of points.

```
R> set.seed(2)
R> dfA <- data.frame(x = runif(100), y = runif(100), t = runif(100),
+    m1 = rep(c("A"), times = 100))
R> dfB <- data.frame(x = runif(50), y = runif(50), t = runif(50),
+    m1 = rep(c("B"), each = 50))
R> stpm1 <- stpm(rbind(dfA, dfB))
R> plot(stpm1)
```

To fit a multitype Poisson point process model, therefore, an object of ‘stpm’, with a categorical mark, must be provided to stppm(). The multitype model is fitted by setting marked = TRUE, and by calling the mark with a formula like s(mark, bs = "re"), exactly following the random effects specifications of the gam() function. In brief, this is because multitype point process fitting is based on a cubature scheme replicated for each mark type. For instance, the following code fits a multitype Poisson process model with inhomogeneous intensity depending on the x coordinate and a random intercept θ_{0m}, as follows

$$\lambda(x, y, t) = \exp(\theta_0 + \theta_{0m} + \theta_1 x).$$

```
R> inh4 <- stppm(stpm1, formula = ~ x + s(m1, bs = "re"), marked = TRUE,
+    seed = 2)
```

In point process terms, this means that the average number of points will differ between the two types, but the x coordinate is believed to have a common effect on the intensities of the two subpatterns. The right panel of Figure 13 clearly illustrates these results, showing a consistently low intensity for the points belonging to the subpattern with fewer points.

```
R> plot(inh4)
```

Note that any combination of the presented model specifications is allowed. For instance, multitypes point processes can be fitted, with semi-parametric specifications of the first-order intensity, depending on both coordinates and external spatio-temporal covariates.

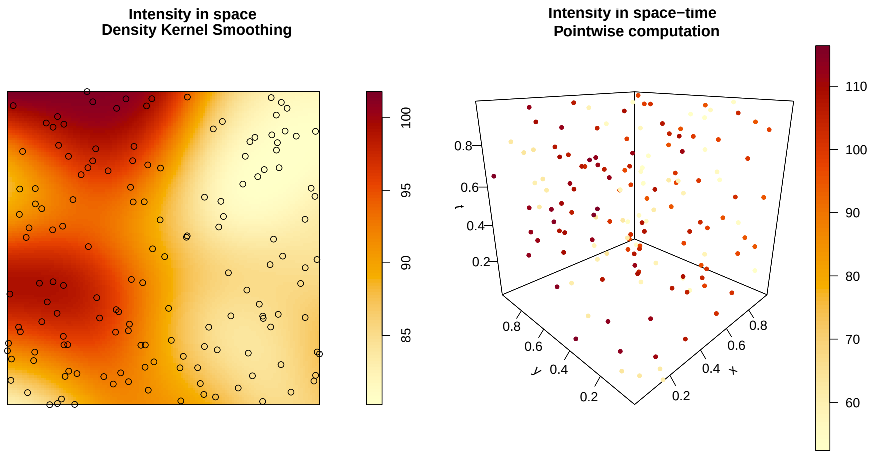

Figure 13: Output of the `plot.stppm()` function applied to a fitted multitype model.

6.4. Spatio-temporal Poisson point processes with separable intensity

The function `sepstppm()` fits a separable parametric spatio-temporal Poisson process model (Diggle 2013) to point patterns observed on a subset of the Euclidean space, according to the following generic form

$$\lambda(x, y, t) = \lambda(x, y)\lambda(t),$$

where $\lambda(x, y)$ and $\lambda(t)$ are non-negative functions on W and T, respectively. This formulation can include a combination of a parametric spatial point pattern model, potentially depending on the spatial coordinates and/or spatial covariates, and a parametric log-linear model for the temporal component. The spatio-temporal intensity is therefore obtained by multiplying the spatial and temporal intensities fitted separately. This has the advantage of giving the possibility to include purely spatial and purely temporal covariates, denoted by $\mathbf{Z}_S(x, y)$ and $\mathbf{Z}_T(t)$, with the following general formulation

$$\lambda(x, y, t) = \lambda(x, y)\lambda(t) = \exp(\theta_0 + \boldsymbol{\theta}_S^\top \mathbf{Z}_S(x, y) + \boldsymbol{\theta}_T^\top \mathbf{Z}_T(t)).$$

The function `sepstlppm()` implements the network counterpart of the spatio-temporal Poisson point process with separable intensity and fully parametric specification. Concerning linear network point patterns, only non-parametric estimators of the intensity function have been suggested in the literature (Mateu, Moradi, and Cronie 2020; Moradi and Mateu 2020). The functions `plot.sepstppm()` and `plot.sepstlppm()` show the fitted intensities, displayed both in space and in space and time. Next, we perform an example on a subset of the Valencia dataset, including the linear network in the inferential procedure. See Figure 14 for the plot of the carried-out example.

```
R> crimesub <- stpm(valenciacrimes$df[101:200, ],
+    names = colnames(valenciacrimes$df)[-c(1:3)], L = valencianet)
R> mod1 <- sepstlppm(crimesub, spaceformula = ~x , timeformula = ~ day)
R> plot(mod1)
```

6.5. Spatio-temporal Poisson point processes with non-separable intensity

When separability of the spatial and temporal component is not plausible for the data, a non-separable specification of the intensity function is more advisable. This is obtained through

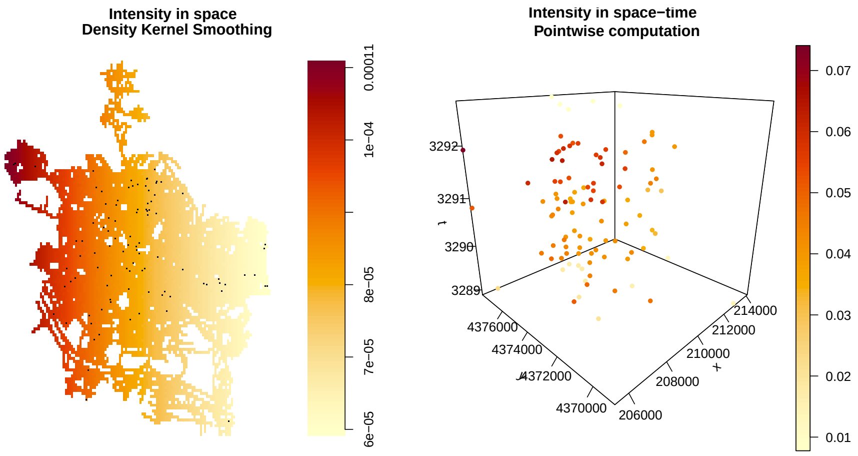

Figure 14: Plot of the separable model fitted produced by the `plot.sepstlppm()` function.

the `stppm()` function when including proper spatio-temporal covariates or specifying any kind of interaction between spatial and temporal variables.

As an example, the following code fits an inhomogeneous non-separable spatio-temporal Poisson model with dependence on the spatio-temporal coordinates and some of their polynomials and interactions specified as follows

$$\lambda(x, y, t) = \exp(\theta_0 + \theta_1 x + \theta_2 y + \theta_3 t + \theta_4 xy + \theta_5 yt + \theta_6 x^2 + \theta_7 y^2 + \theta_8 t^2 + \theta_9 x^2 y^2).$$

```
R> nonsepmod <- stppm(greececatalog, formula = ~ x + y + t + x:y + y:t +
+     I(x^2) + I(y^2) + I(t^2) + I(x^2):I(y^2), seed = 2)
```

As any other model fitted through `stppm()`, both the `print()` and `summary()` functions will return the estimated coefficients, and the `plot()` function will display the estimated intensity in space and in space and time.

```
R> summary(nonsepmod)

Inhomogeneous Poisson process
with Trend: ~x + y + t + x:y + y:t + I(x^2) + I(y^2) + I(t^2) + I(x^2):I(y^2)

Estimated coefficients:
  (Intercept)             x             y             t        I(x^2)
     -967.872        54.323        41.785        -0.007        -0.528
       I(y^2)        I(t^2)           x:y           y:t I(x^2):I(y^2)
       -0.343         0.000        -1.481         0.000         0.000
```

Figure 15 is produced by the following command:

```
R> plot(nonsepmod)
```

Furthermore, since the model is fitted altogether employing a GLM, the significance of the parameters can be inspected by checking the `summary()` of the `mod_global` element of the object returned by the `stppm()` function.

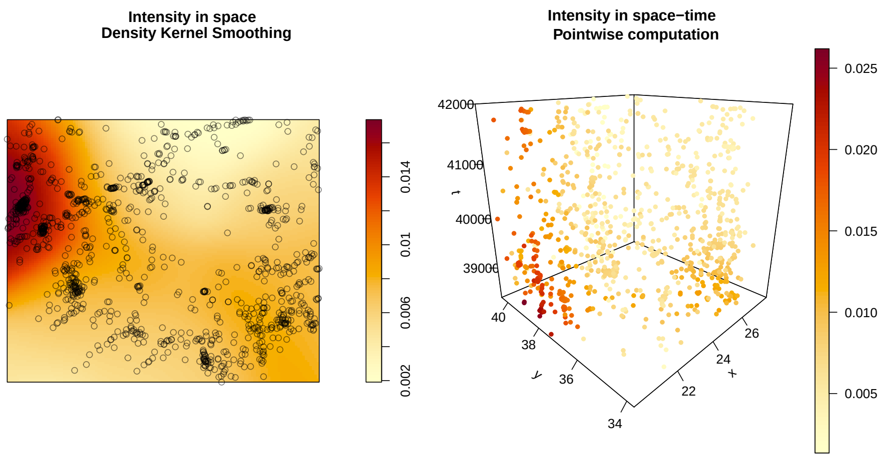

Figure 15: Plot of the non-separable model fitted produced by the `plot.stppm()` function.

```
R> summary(nonsepmod$mod_global)

Family: poisson
Link function: log

Formula:
y_resp ~ x + y + t + x:y + y:t + I(x^2) + I(y^2) + I(t^2) + I(x^2):I(y^2)

Parametric coefficients:
Estimate Std. Error z value Pr(>|z|)
(Intercept)   -9.679e+02  1.344e+02  -7.202 5.93e-13 ***
x              5.432e+01  6.687e+00   8.124 4.50e-16 ***
y              4.179e+01  4.597e+00   9.090  < 2e-16 ***
t             -7.294e-03  2.326e-03  -3.136 0.001712 **
I(x^2)        -5.282e-01  6.989e-02  -7.557 4.12e-14 ***
I(y^2)        -3.432e-01  3.328e-02 -10.311  < 2e-16 ***
I(t^2)         6.193e-08  2.869e-08   2.159 0.030867 *
x:y           -1.481e+00  1.782e-01  -8.313  < 2e-16 ***
y:t            5.823e-05  1.622e-05   3.590 0.000331 ***
I(x^2):I(y^2)  3.920e-04  4.953e-05   7.914 2.49e-15 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Note that even though `mod_global` is of class '`glm`', it is not advisable to rely on standard classical GLM tools, such as the AIC or the R^2, since they depend on the chosen structure of the cubature scheme, not explored here in detail.

6.6. Log-Gaussian Cox processes

The `stlgcppm()` function estimates the covariance parameters of a spatio-temporal log-Gaussian Cox process (LGCP, Diggle, Moraga, Rowlingson, and Taylor 2013) with random intensity

$$\Lambda(x, y, t) = \lambda(x, y, t) \exp(S(x, y, t)),$$

following the joint minimum contrast procedure introduced in Siino *et al.* (2018a). LGCPs are hierarchical Poisson processes, where the dependence in the point pattern is modeled through the common latent Gaussian variable S (Rue *et al.* 2009). Here S is a Gaussian process with $\mathsf{E}(S(x,y,t)) = \mu = -0.5\sigma^2$ and so $E(\exp(S(x,y,t))) = 1$ and with variance and covariance matrix $\sigma^2\gamma(r,h)$ under the stationary assumption, with $\gamma(\cdot)$ the correlation function of the Gaussian random field, and r and h some spatial and temporal distances.

The covariances available are separable exponential, Gneiting (Gneiting, Genton, and Guttorp 2006; Schlather, Malinowski, Menck, Oesting, and Strokorb 2015), and Iaco-Cesare (De Cesare, Myers, and Posa 2002; De Iaco, Myers, and Posa 2002). The function works by assuming a homogeneous first-order intensity as default. Different inhomogeneous specifications of the first-order intensity function are implemented as well.

```
R> catsub <- stp(greececatalog$df[1:200, ])
R> lgcp1 <- stlgcppm(catsub, seed = 2)
```

As a default, the package fits a LGCP model with a separable structure for the covariance function of the GRF (Brix and Diggle 2001) that has exponential form for both the spatial and the temporal components,

$$\mathbb{C}(r, h) = \sigma^2 \exp\left(\frac{-r}{\alpha}\right) \exp\left(\frac{-h}{\beta}\right),$$

where σ^2 is the variance parameter, α is the scale parameter for the spatial distance and β is the scale parameter for the temporal one.

The `print()` and `summary()` functions give the main information on the fitted model.

```
R> lgcp1

Joint minimum contrast fit
for a log-Gaussian Cox process with
global first-order intensity and
global second-order intensity
--------------------------------------------------
Homogeneous Poisson process
with Intensity: 0.00849

Estimated coefficients of the first-order intensity:

(Intercept)
     -4.769
--------------------------------------------------
Covariance function: separable

Estimated coefficients of the second-order intensity:
```

```
 sigma  alpha   beta
15.389  0.239 15.275
--------------------------------------------------
Model fitted in 0.014 minutes
```

The `plot.sepstlppm()` function shows the fitted intensity displayed both in space (by means of a density kernel smoothing) and in space and time, similar to what we have seen so far with the other classes of models.

6.7. Local models

Local spatio-temporal Poisson point processes

The `locstppm()` function fits a spatio-temporal local Poisson process model (D'Angelo *et al.* 2023b) to an observed spatio-temporal point pattern stored in a '`stp`' object, that is, a Poisson model with a vector of parameters $\boldsymbol{\theta}_i \in \Theta$ for each point (x_i, y_i, t_i). In local likelihood estimation of Poisson processes (Loader 1999) the estimated intensity at (x, y, t) is taken to be the plug-in value

$$\hat{\lambda}(x, y, t) = \lambda(x, y, t; \hat{\boldsymbol{\theta}}(x, y, t))$$

associated with the fitted parameter vector at (x, y, t).

The `print()` and `summary()` functions will provide information of the estimated local parameters by means of the summary of their distributions.

```
R> set.seed(2)
R> inh <- rstpp(lambda = function(x, y, t, a) {exp(a[1] + a[2] * x)},
+    par = c(0.005, 5))
R> inh_local <- locstppm(inh, formula = ~ x, seed = 2)
R> inh_local

Inhomogeneous Poisson process
with Trend: ~x

Summary of estimated coefficients
(Intercept)           x
Min.   :0.3075   Min.   :2.803
1st Qu.:0.9073   1st Qu.:3.652
Median :1.4415   Median :4.264
Mean   :1.4360   Mean   :4.291
3rd Qu.:2.0157   3rd Qu.:4.975
Max.   :2.7504   Max.   :5.637
```

Inference is performed through the fitting of a GLM using a localized version of the cubature scheme, firstly introduced in the spatio-temporal framework by D'Angelo *et al.* (2023b). Moreover, the `localplot()` function displays the local coefficients overlapped to the observed points in some three-dimensional plots (Figure 16).

```
R> localplot(inh_local)
```

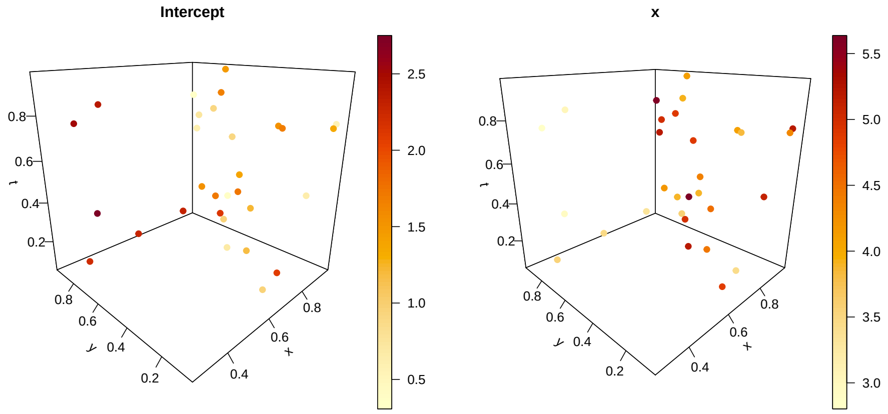

Figure 16: Output of the `localplot()` function applied on a '`locstppm`' object.

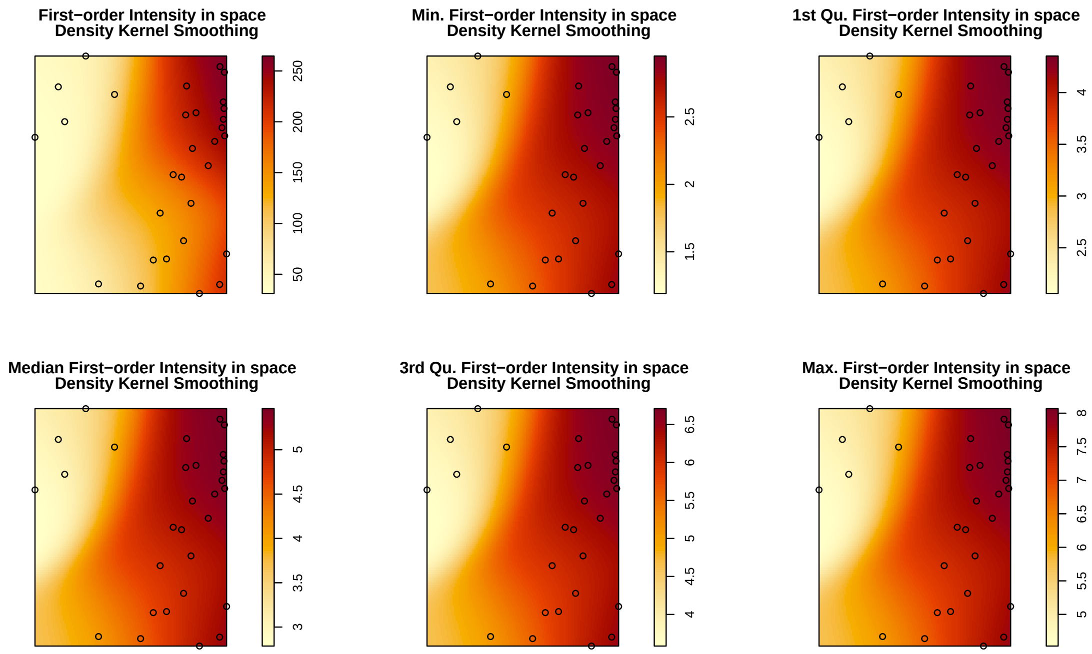

Figure 17: Output of the `localsummary()` function applied on a '`locstppm`' object.

Finally, we also implemented the `localsummary()` function, to break up the contribution of the local estimates to the fitted intensity by plotting the overall intensity and the density kernel smoothing of some artificial intensities obtained by imputing the quartiles of the local parameters' distributions (Figure 17).

```
R> localsummary(inh_local)
```

Local spatio-temporal log-Gaussian Cox processes

If the `second` argument of the `stlgcppm()` function is set to `"local"`, it allows to estimate local second-order parameters of a spatio-temporal LGCP, following the *locally weighted minimum contrast* procedure introduced in D'Angelo *et al.* (2023b). In particular, we employ the minimum contrast procedure based on the local spatio-temporal pair correlation function (Gabriel *et al.* 2013) documented in `LISTAhat()` of **stpp**. If also `first` is set to `"local"`, also the first-order intensity parameters will be fitted locally, obtaining the same achieved by `locstppm()`. In the case of local parameters (either first, second-order, or both), the `print()` and `summary()` functions contain information on their distributions.

```
R> lgcp2 <- stlgcppm(catsub, second = "local", seed = 2)
R> lgcp2

Joint minimum contrast fit
for a log-Gaussian Cox process with
global first-order intensity and
local second-order intensity
--------------------------------------------------
Homogeneous Poisson process
with Intensity: 0.00849

Estimated coefficients of the first-order intensity:
(Intercept)
     -4.769
--------------------------------------------------
Covariance function: separable

Summary of estimated coefficients of the second-order intensity
sigma            alpha             beta
Min.   : 4.867   Min.   :0.1212   Min.   : 7.174
1st Qu.: 6.740   1st Qu.:0.1776   1st Qu.: 8.528
Median :13.178   Median :0.3546   Median :12.861
Mean   :15.638   Mean   :1.0904   Mean   :14.229
3rd Qu.:18.946   3rd Qu.:1.3206   3rd Qu.:16.433
Max.   :40.859   Max.   :6.1096   Max.   :31.786
--------------------------------------------------
Model fitted in 0.88 minutes
```

In the even more specific case of local covariance parameters, the `plot()` function returns the mean of the random intensity, instead of the first-order intensity, displayed both in space (by means of a density kernel smoothing) and in space and time (Figure 18).

```
R> plot(lgcp2)
```

Finally, the `localplot()` and `localsummary()` functions also work on '`stlgcppm`' objects, if the LGCP has local first- or second-order fitted parameters. In the particular case of local covariance parameters, `localplot()` applied on a '`stlgcppm`' object further displays the local estimates of the chosen covariance function (Figure 19).

```
R> localplot(lgcp2)
```

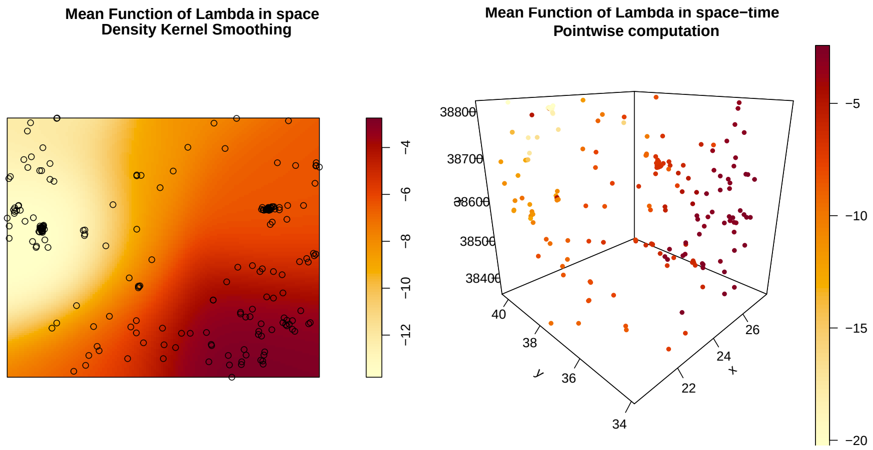

Figure 18: Output of the `plot()` function applied to an estimated LGCP with local covariance parameters.

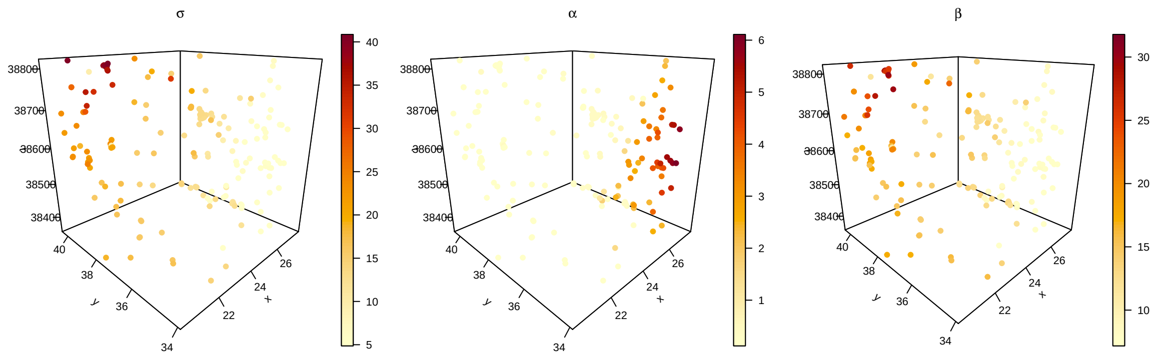

Figure 19: Output of the `localplot()` function applied on a '`stlgcppm`' object in the case of local covariance parameters.

7. Diagnostics

This section is devoted to the presentation of general diagnostic tools based on second-order summary statistics, both globally and locally.

7.1. Global diagnostics

The `globaldiag()` function performs global diagnostics of a model fitted for the first-order intensity of a spatio-temporal point pattern, using the spatio-temporal inhomogeneous K function (Gabriel and Diggle 2009) documented by the function `STIKhat()` of the **stpp** package (Gabriel *et al.* 2022). It can also perform global diagnostics of a model fitted for the first-order intensity of a spatio-temporal point pattern on a linear network by means of the spatio-temporal inhomogeneous K function on a linear network (Moradi and Mateu 2020) documented by the function `STLKinhom()` of the **stlnpp** package (Moradi *et al.* 2020). Both versions return the plots of the inhomogeneous K function weighted by the provided intensity to diagnose, its theoretical value, and their difference (Figure 20). Next, an example of a simulated point pattern on the unit cube.

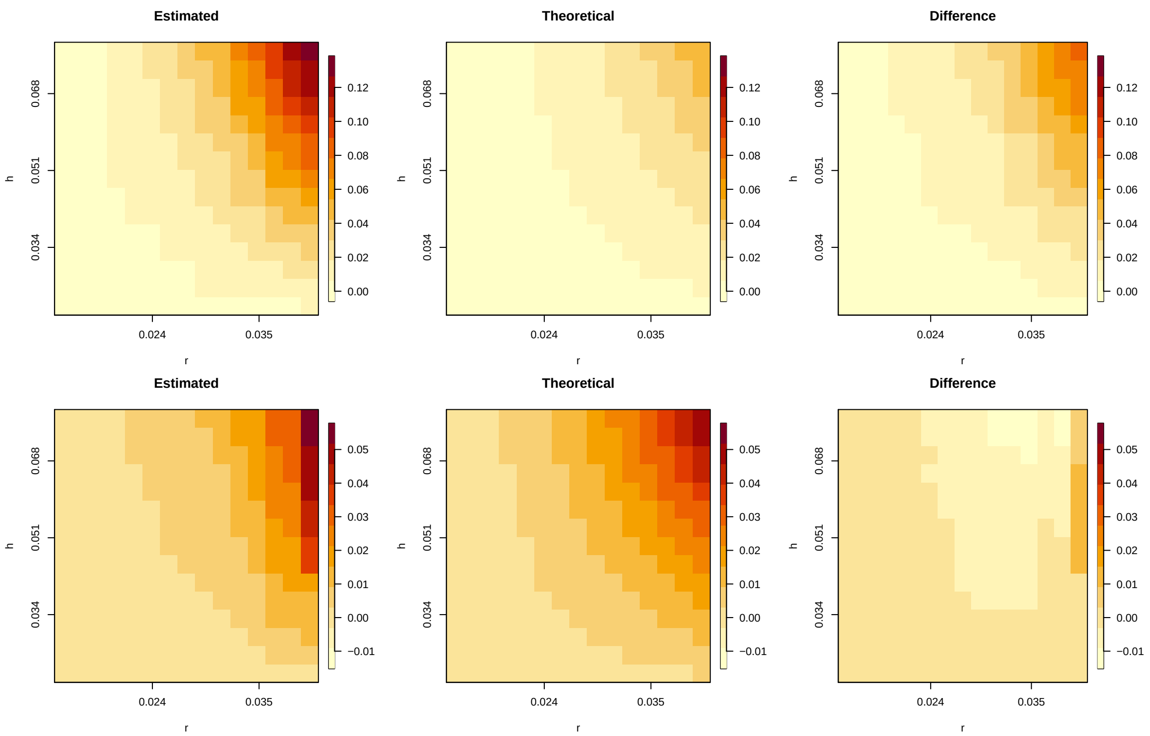

Figure 20: Output of the `globaldiag()` function: K functions weighted by the constant and therefore wrong intensity function (top panels), and K functions weighted by the true intensity function (bottom panels).

```
R> set.seed(2)
R> inh <- rstpp(lambda = function(x, y, t, a) {exp(a[1] + a[2] * x)},
+    par = c(.3, 6))
R> mod1 <- stppm(inh, formula = ~ 1, seed = 2)
R> mod2 <- stppm(inh, formula = ~ x, seed = 2)
R> (g1 <- globaldiag(mod1))

Sum of squared differences : 2.036

R> (g2 <- globaldiag(mod2))

Sum of squared differences : 0.486

R> plot(g1)
R> plot(g2)
```

Figure 20 displays the result of `globaldiag()` applied to two different fitted intensities: the constant and wrong intensity, and the true one, on the top and bottom panels, respectively. It is evident that the difference between the estimated inhomogeneous K function and its theoretical value is considerably smaller when weighted by the true intensity function.

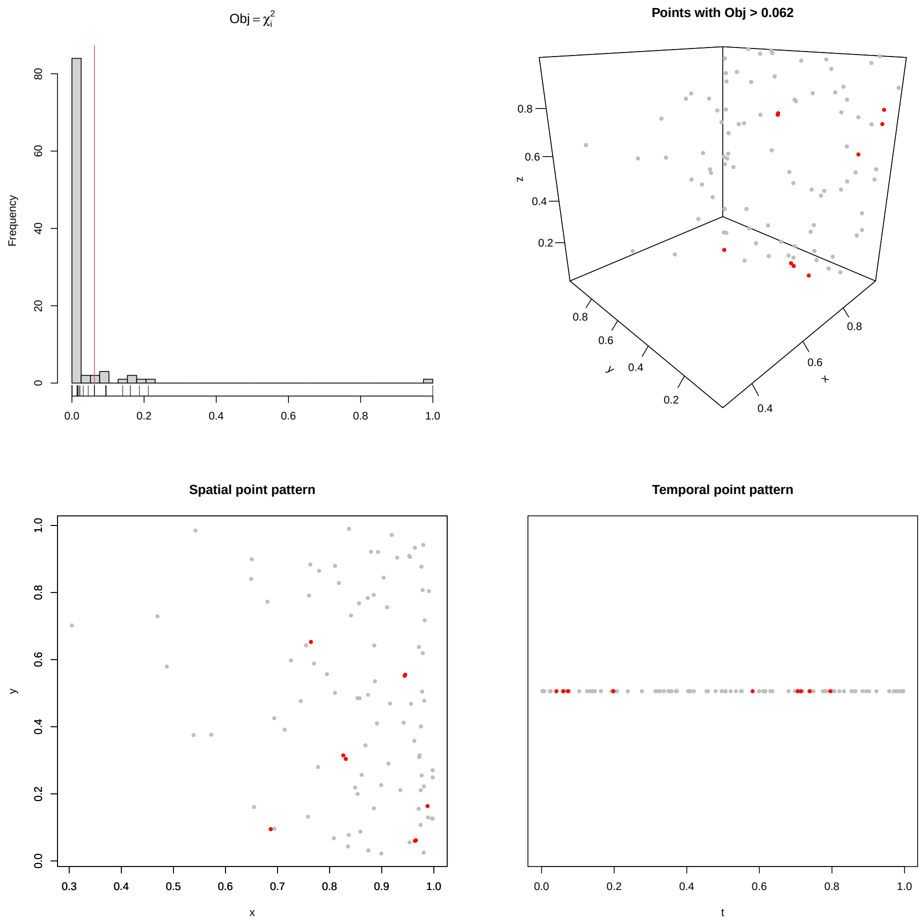

Figure 21: Output of the `plot.localdiag()` function.

7.2. Local diagnostics

The `localdiag()` function performs local diagnostics of a model fitted for the first-order intensity of a spatio-temporal point pattern by means of the local spatio-temporal inhomogeneous K functions (Adelfio *et al.* 2020) documented by function `KLISTAhat()` of **stpp**. It returns the points identified as outlying following the diagnostics procedure on individual points of an observed point pattern, as introduced in Adelfio *et al.* (2020) and then extended by D'Angelo *et al.* (2023a) to the linear network case. `localdiag()` is indeed also able to perform local diagnostics of a model fitted for the first-order intensity of a spatio-temporal point pattern on a linear network by the local spatio-temporal inhomogeneous K functions on linear networks D'Angelo *et al.* (2021) documented by the function `localSTLKinhom()` of this package. The points resulting from the local diagnostic procedure provided by this function can be inspected via the `plot()` (Figure 21), `print()`, `summary()`, and `infl()` (Figure 22) functions, as illustrated in the following.

```
R> res <- localdiag(inh, mod1$l, p = .9)
R> res

Points outlying from the 0.9 percentile
```

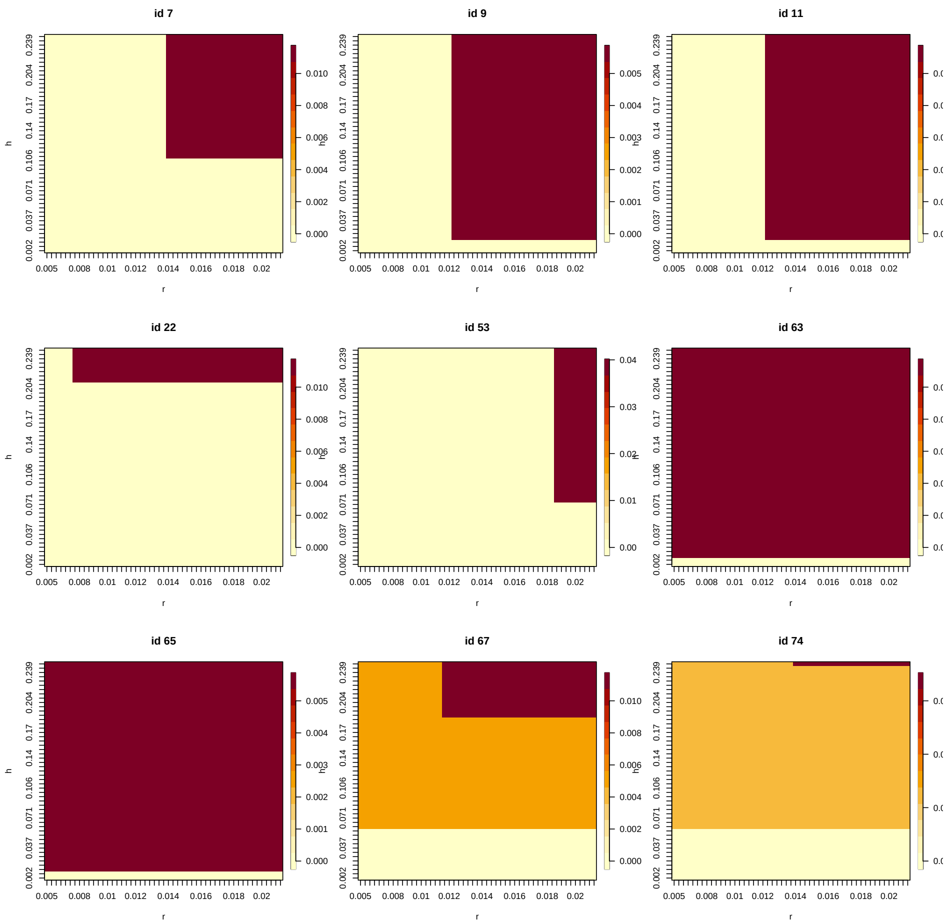

Figure 22: Output of the `infl()` function.

```
of the analysed spatio-temporal point pattern
--------------------------------------

Analysed pattern X: 97 points
9 outlying points

R> plot(res)
```

In particular, the `infl()` function plots the K functions of all those points identified as outlying by `localdiag()`. Alternatively, one can show only some specific K functions by imputing a vector to the argument `id`.

```
R> infl(res)
```

8. Future developments

The **stopp** package represents the creation of a toolbox for different spatio-temporal analyses to be performed on observed point patterns, following the growing stream of literature on point process theory. We contribute to the existing literature by framing many of the most widespread methods for the analysis of spatio-temporal point processes into a unique package, which is intended to foster many further extensions.

The **stopp** package tools are not exhaustive. Some current developments that will be available in future include the possibility of handling irregular spatial windows for purely spatial components and three-dimensional spatial point patterns, fundamental in geology (Li, Zhuang, Chen, Guo, and Xiong 2024) and astronomy (Babu and Feigelson 1996; Stoica, Martínez, and Saar 2007). It would be useful to be able to simulate multitype point patterns as well as patterns with intensity depending on external covariates. Also, the next versions of the package could generalize the function `stppm()`, allowing for the inclusion of non-continuous covariates. Alternatives to the inverse-distance weighting for the continuous covariate interpolation could be implemented, including some spatio-temporal smoothing using a Gaussian kernel weighting, which would lead to the Nadaraya-Watson smoother (Nadaraya 1964, 1989; Watson 1964), in addition to the most used kriging (Matheron 1963) and nearest neighbors interpolation. In addition, the very general Poisson point process model implementation currently available could serve as the basis for the estimation of the first-order intensity function, like the LGCPs. Also, other Cox process models relying on the minimum contrast procedure could be implemented, providing the possibility of fitting global and local parameter estimation.

Moreover, already published research on local characteristics of point processes needing a general software implementation includes: Siino, Rodríguez-Cortés, Mateu, and Adelfio (2020); D'Angelo, Adelfio, Mateu, and Cronie (2023c); D'Angelo (2024). Regarding alternative fitting methods, it is our intention to give the possibility to use that provided in the recent paper D'Angelo and Adelfio (2024b).

Currently, the cubature scheme (D'Angelo *et al.* 2023b; D'Angelo and Adelfio 2024a) is being numerically explored. Indeed, the number of dummy points should be enough for accurate likelihood estimation and experimental results are expected to give guidelines on the number of dummy points to generate. Note that the cubature scheme is already being applied in some work-in-progress analyses of real data in D'Angelo, Sharma, Tarantino, and Adelfio (2024b), for the spatio-temporal analysis of lightning point process data in severe storms, and in Tarantino *et al.* (2024), for the study of a real three-dimensional purely spatial observed point pattern of young stars of the Gaia Archive.

Other interesting implementations which still need some underlying methodological development include the fitting of some general marked point processes with continuous marks, in addition to the already established fitting of general multitype point processes.

Acknowledgments

This work was supported by the Targeted Research Funds 2025 (FFR 2025) of the University of Palermo (Italy), the PRIN 2022: Spatio-Temporal Functional Marked Point Processes for Probabilistic Forecasting of Earthquakes 2022BN7CJP P. I. Giada Adelfio. CUP B53C24006340006, and the European Union–NextGenerationEU, in the framework of the GRINS–Growing Resilient, INclusive and Sustainable project (GRINS PE00000018–CUP C93C22005270001). The views and opinions expressed are solely those of the authors and do not necessarily reflect those of the European Union, nor can the European Union be held responsible for them.

References

Adelfio G, Chiodi M (2015). "FLP Estimation of Semi-Parametric Models for Space-Time Point Processes and Diagnostic Tools." *Spatial Statistics*, **14**, 119–132. doi:10.1016/j.spasta.2015.06.004.

Adelfio G, Chiodi M (2021). "Including Covariates in a Space-Time Point Process with Application to Seismicity." *Statistical Methods & Applications*, **39**, 947–971. doi:10.1007/s10260-020-00543-5.

Adelfio G, Siino M, Mateu J, Rodríguez-Cortés FJ (2020). "Some Properties of Local Weighted Second-Order Statistics for Spatio-Temporal Point Processes." *Stochastic Environmental Research and Risk Assessment*, **34**(1), 149–168. doi:10.1007/s00477-019-01748-1.

Adepeju M (2022). ***stppSim**: Spatiotemporal Point Patterns Simulation*. doi:10.32614/CRAN.package.stppSim. R package version 1.2.7.

Alaimo Di Loro P, Mingione M, Fantozzi P (2024). "Semi-Parametric Spatio-Temporal Hawkes Process for Modelling Road Accidents in Rome." *Journal of Agricultural, Biological and Environmental Statistics*. doi:10.1007/s13253-024-00615-z.

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. doi:10.1111/j.1467-9469.2011.00752.x.

Anselin L (1995). "Local Indicators of Spatial Association – LISA." *Geographical Analysis*, **27**(2), 93–115. doi:10.1111/j.1538-4632.1995.tb00338.x.

Babu GJ, Feigelson ED (1996). "Spatial Point Processes in Astronomy." *Journal of Statistical Planning and Inference*, **50**(3), 311–326. doi:10.1016/0378-3758(95)00060-7.

Baddeley A, Coeurjolly JF, Rubak E, Waagepetersen R (2014). "Logistic Regression for Spatial Gibbs Point Processes." *Biometrika*, **101**(2), 377–392. doi:10.1093/biomet/ast060.

Baddeley A, Turner R (2005). "**spatstat**: An R Package for Analyzing Spatial Point Patterns." *Journal of Statistical Software*, **12**(6), 1–42. doi:10.18637/jss.v012.i06.

Berman M, Turner TR (1992). "Approximating Point Process Likelihoods with **GLIM**." *Journal of the Royal Statistical Society C*, **41**(1), 31–38. doi:10.2307/2347614.

Breslow NE, Clayton DG (1993). "Approximate Inference in Generalized Linear Mixed Models." *Journal of the American Statistical Association*, **88**(421), 9–25. doi:10.1080/01621459.1993.10594284.

Brix A, Diggle PJ (2001). "Spatiotemporal Prediction for Log-Gaussian Cox Processes." *Journal of the Royal Statistical Society B*, **63**(4), 823–841. doi:10.1111/1467-9868.00315.

Briz-Redón Á, Iftimi A, Mateu J, Romero-García C (2023). "A Mechanistic Spatio-Temporal Modeling of COVID-19 Data." *Biometrical Journal*, **65**(1), 2100318. doi:10.1002/bimj.202100318.

Chaudhuri S, Juan P, Mateu J (2023). "Spatio-Temporal Modeling of Traffic Accidents Incidence on Urban Road Networks Based on an Explicit Network Triangulation." *Journal of Applied Statistics*, **50**(16), 3229–3250. doi:10.1080/02664763.2022.2104822.

Chiodi M, Adelfio G (2017). "Mixed Non-Parametric and Parametric Estimation Techniques in R Package **etasFLP** for Earthquakes' Description." *Journal of Statistical Software*, **76**(3), 1–29. doi:10.18637/jss.v076.i03.

Chiodi M, Adelfio G (2023). ***etasFLP**: Mixed FLP and ML Estimation of ETAS Space-Time Point Processes for Earthquake Description*. doi:10.32614/CRAN.package.etasFLP. R package version 2.2.2.

D'Angelo N (2024). "Advances in Kth Nearest-Neighbour Clutter Removal." *Environmental and Ecological Statistics*, **31**(2), 537–554. doi:10.1007/s10651-023-00588-1.

D'Angelo N, Adelfio G (2024a). "Cubature Scheme for Spatio-Temporal Poisson Point Processes Estimation." *arXiv 2403.10878*, arXiv.org E-Print Archive. doi:10.48550/arXiv.2403.10878.

D'Angelo N, Adelfio G (2024b). "Minimum Contrast for the First-Order Intensity Estimation of Spatial and Spatio-Temporal Point Processes." *Statistical Papers*, **65**(6), 3651–3679. doi:10.1007/s00362-024-01541-5.

D'Angelo N, Adelfio G (2025). ***stopp**: Spatio-Temporal Point Pattern Methods, Model Fitting, Diagnostics, Simulation, Local Tests*. doi:10.32614/CRAN.package.stopp. R package version 1.0.0.

D'Angelo N, Adelfio G, Abbruzzo A, Mateu J (2022). "Inhomogeneous Spatio-Temporal Point Processes on Linear Networks for Visitors' Stops Data." *The Annals of Applied Statistics*, **16**(2), 791–815. doi:10.1214/21-aoas1519.

D'Angelo N, Adelfio G, Mateu J (2021). "Assessing Local Differences between the Spatio-Temporal Second-Order Structure of Two Point Patterns Occurring on the Same Linear Network." *Spatial Statistics*, **45**, 100534. ISSN 2211-6753. doi:10.1016/j.spasta.2021.100534.

D'Angelo N, Adelfio G, Mateu J (2023a). "Local Inhomogeneous Second-Order Characteristics for Spatio-Temporal Point Processes Occurring on Linear Networks." *Statistical Papers*, **64**, 779–805. doi:10.1007/s00362-022-01338-4.

D'Angelo N, Adelfio G, Mateu J (2023b). "Locally Weighted Minimum Contrast Estimation for Spatio-Temporal Log-Gaussian Cox Processes." *Computational Statistics & Data Analysis*, **180**, 107679. doi:10.1016/j.csda.2022.107679.

D'Angelo N, Adelfio G, Mateu J, Cronie O (2023c). "Local Inhomogeneous Weighted Summary Statistics for Marked Point Processes." *Journal of Computational and Graphical Statistics*, **33**(2), 588–602. doi:10.1080/10618600.2023.2206441.

D'Angelo N, Payares D, Adelfio G, Mateu J (2024a). "Self-Exciting Point Process Modelling of Crimes on Linear Networks." *Statistical Modelling*, **24**(2), 139–168. doi:10.1177/1471082x221094146.

D'Angelo N, Sharma M, Tarantino M, Adelfio G (2024b). "Spatio-Temporal Analysis of Lightning Point Process Data in Severe Storms." In A Plaia, L Egidi, A Abbruzzo (eds.), *Proceedings of the Statistics and Data Science 2024 Conference*, pp. 302–307. Università degli Studi di Palermo, Palermo. URL `https://tinyurl.com/SDS2024-Proceedings`.

De Cesare L, Myers DE, Posa D (2002). "Fortran Programs for Space-Time Modeling." *Computers & Geosciences*, **28**(2), 205–212. doi:10.1016/s0098-3004(01)00040-1.

De Iaco S, Myers DE, Posa D (2002). "Nonseparable Space-Time Covariance Models: Some Parametric Families." *Mathematical Geology*, **34**(1), 23–42. doi:10.1023/a:1014075310344.

Diggle PJ (2013). *Statistical Analysis of Spatial and Spatio-Temporal Point Patterns*. 3rd edition. Chapman & Hall/CRC.

Diggle PJ, Moraga P, Rowlingson B, Taylor BM (2013). "Spatial and Spatio-Temporal Log-Gaussian Cox Processes: Extending the Geostatistical Paradigm." *Statistical Science*, **28**(4), 542–563. doi:10.1214/13-sts441.

Gabriel E, Diggle PJ (2009). "Second-Order Analysis of Inhomogeneous Spatio-Temporal Point Process Data." *Statistica Neerlandica*, **63**(1), 43–51. doi:10.1111/j.1467-9574.2008.00407.x.

Gabriel E, Diggle PJ, Rowlingson B, Rodriguez-Cortes FJ (2022). ***stpp**: Space-Time Point Pattern Simulation, Visualisation and Analysis*. doi:10.32614/CRAN.package.stpp. R package version 2.0-5.

Gabriel E, Rowlingson BS, Diggle PJ (2013). "**stpp**: An R Package for Plotting, Simulating and Analyzing Spatio-Temporal Point Patterns." *Journal of Statistical Software*, **53**(2), 1–29. doi:10.18637/jss.v053.i02.

Gilardi A, Borgoni R, Mateu J (2024). "A Non-Separable First-Order Spatio-Temporal Intensity for Events on Linear Networks: An Application to Ambulance Interventions." *The Annals of Applied Statistics*, **18**(1), 529–554. doi:10.1214/23-aoas1800.

Gneiting T, Genton MG, Guttorp P (2006). "Geostatistical Space-Time Models, Stationarity, Separability, and Full Symmetry." In B Finkenstadt, L Held, V Isham (eds.), *Statistical Methods for Spatio-Temporal Systems*, chapter 4, pp. 151–175. Chapman & Hall/CRC, New York.

González JA, Rodríguez-Cortés FJ, Cronie O, Mateu J (2016). "Spatio-Temporal Point Process Statistics: A Review." *Spatial Statistics*, **18**(B), 505–544. doi:10.1016/j.spasta.2016.10.002.

Jalilian A (2019). "**ETAS**: An R Package for Fitting the Space-Time ETAS Model to Earthquake Data." *Journal of Statistical Software, Code Snippets*, **88**(1), 1–39. doi:10.18637/jss.v088.c01.

Jalilian A (2024). ***ETAS**: Modeling Earthquake Data Using ETAS Model*. doi:10.32614/CRAN.package.ETAS. R package version 0.6.1.

Jalilian A, Mateu J (2021). "A Hierarchical Spatio-Temporal Model to Analyze Relative Risk Variations of COVID-19: A Focus on Spain, Italy, and Germany." *Stochastic Environmental Research and Risk Assessment*, **35**, 797–812. doi:10.1007/s00477-021-02003-2.

Jones-Todd CM, Van Helsdingen A (2023). ***stelfi**: Hawkes and Log-Gaussian Cox Point Processes Using Template Model Builder*. doi:10.32614/CRAN.package.stelfi. R package version 0.0.2.

Kalair K, Connaughton C, Alaimo Di Loro P (2021). "A Non-Parametric Hawkes Process Model of Primary and Secondary Accidents on a UK Smart Motorway." *Journal of the Royal Statistical Society C*, **70**(1), 80–97. doi:10.1111/rssc.12450.

Li Y, Zhuang J, Chen S, Guo Y, Xiong Z (2024). "A 3D Spherical ETAS Model: A Case Study in the Alaska-Aleutian Region." *Seismological Research Letters*. doi:10.1785/0220230440.

Loader CR (1999). "Bandwidth Selection: Classical or Plug-In?" *The Annals of Statistics*, **27**(2), 415–438. doi:10.1214/aos/1018031201.

Mateu J, Moradi M, Cronie O (2020). "Spatio-Temporal Point Patterns on Linear Networks: Pseudo-Separable Intensity Estimation." *Spatial Statistics*, **37**, 100400. doi:10.1016/j.spasta.2019.100400.

Matheron G (1963). "Principles of Geostatistics." *Economic Geology*, **58**(8), 1246–1266. doi:10.2113/gsecongeo.58.8.1246.

Moradi M, Cronie O, Mateu J (2020). ***stlnpp**: Spatio-Temporal Analysis of Point Patterns on Linear Networks*. doi:10.32614/CRAN.package.stlnpp. R package version 0.3-7.

Moradi MM, Mateu J (2020). "First- And Second-Order Characteristics of Spatio-Temporal Point Processes on Linear Networks." *Journal of Computational and Graphical Statistics*, **29**(3), 432–443. doi:10.1080/10618600.2019.1694524.

Nadaraya EA (1964). "On Estimating Regression." *Theory of Probability & Its Applications*, **9**(1), 141–142. doi:10.1137/1109020.

Nadaraya EA (1989). *Nonparametric Estimation of Probability Densities and Regression Curves*. Springer-Verlag. doi:10.1007/978-94-009-2583-0.

Raeisi M, Bonneu F, Gabriel E (2021). "A Spatio-Temporal Multi-Scale Model for Geyer Saturation Point Process: Application to Forest Fire Occurrences." *Spatial Statistics*, **41**, 100492. doi:10.1016/j.spasta.2021.100492.

Rowlingson B, Diggle P (2022). ***splancs**: Spatial and Space-Time Point Pattern Analysis*. doi:10.32614/CRAN.package.splancs. R package version 2.01-43.

Rue H, Martino S, Chopin N (2009). "Approximate Bayesian Inference for Latent Gaussian Models by Using Integrated Nested Laplace Approximations." *Journal of the Royal Statistical Society B*, **71**(2), 319–392. doi:10.1111/j.1467-9868.2008.00700.x.

Schlather M, Malinowski A, Menck PJ, Oesting M, Strokorb K (2015). "Analysis, Simulation and Prediction of Multivariate Random Fields with Package **RandomFields**." *Journal of Statistical Software*, **63**(8), 1–25. doi:10.18637/jss.v063.i08.

Schoenberg F (2023). "Estimating COVID-19 Transmission Time Using Hawkes Point Processes." *The Annals of Applied Statistics*, **17**(4), 3349–3362. doi:10.1214/23-aoas1765.

Shepard D (1968). "A Two-Dimensional Interpolation Function for Irregularly-Spaced Data." In *Proceedings of the 1968 23rd ACM National Conference*, pp. 517–524. doi:10.1145/800186.810616.

Siino M, Adelfio G, Mateu J (2018a). "Joint Second-Order Parameter Estimation for Spatio-Temporal Log-Gaussian Cox Processes." *Stochastic Environmental Research and Risk Assessment*, **32**(12), 3525–3539. doi:10.1007/s00477-018-1579-0.

Siino M, Mateu J, Adelfio G (2016). "Spatio-Temporal Log-Gaussian Cox Processes on Earthquake Events." In *Proceedings of the Eight International Workshop on Spatio-Temporal Modelling (METMA8)*. URL https://hdl.handle.net/10447/244356.

Siino M, Rodríguez-Cortés FJ, Mateu J, Adelfio G (2018b). "Testing for Local Structure in Spatiotemporal Point Pattern Data." *Environmetrics*, **29**(5-6), e2463. doi:10.1002/env.2463.

Siino M, Rodríguez-Cortés FJ, Mateu J, Adelfio G (2020). "Spatio-Temporal Classification in Point Patterns Under the Presence of Clutter." *Environmetrics*, **31**(2), e2599. doi:10.1002/env.2599.

Stoica RS, Martínez VJ, Saar E (2007). "A Three-Dimensional Object Point Process for Detection of Cosmic Filaments." *Journal of the Royal Statistical Society C*, **56**(4), 459–477. doi:10.1111/j.1467-9876.2007.00587.x.

Tarantino M, D'Angelo N, Adelfio G (2024). "Modelling Three-Dimensional Point Patterns." In *Methodological and Applied Statistics and Demography IV*, Italian Statistical Society Series on Advances in Statistics. Springer-Verlag, Cham. doi:10.1007/978-3-031-64447-4_100.

Taylor BM, Davies TM, Rowlingson BS, Diggle PJ (2015). "Bayesian Inference and Data Augmentation Schemes for Spatial, Spatiotemporal and Multivariate Log-Gaussian Cox Processes in R." *Journal of Statistical Software*, **63**(7), 1–48. doi:10.18637/jss.v063.i07.

Watson GS (1964). "Smooth Regression Analysis." *Sankhyā: The Indian Journal of Statistics A*, pp. 359–372. doi:10.1002/9781119432036.ch8.

Wood SN (2003). "Thin Plate Regression Splines." *Journal of the Royal Statistical Society B*, **65**(1), 95–114. doi:10.1111/1467-9868.00374.

Wood SN, Li Z, Shaddick G, Augustin NH (2017). "Generalized Additive Models for Gigadata: Modeling the UK Black Smoke Network Daily Data." *Journal of the American Statistical Association*, **112**(519), 1199–1210. doi:10.1080/01621459.2016.1195744.

Xiong Z, Zhuang J (2023). "SETAS: A Spherical Version of the Space-Time ETAS Model." *Seismological Research Letters*, **94**(3), 1676–1688. doi:10.1785/0220220198.

Youngman B, Economou T (2020). ***ppgam**: Generalised Additive Point Process Models*. doi:10.32614/CRAN.package.ppgam. R package version 1.0.1.

Affiliation:

Nicoletta D'Angelo, Giada Adelfio
University of Palermo
Department of Economics, Business, and Statistics
Palermo, Italy
E-mail: nicoletta.dangelo@unipa.it

Journal of Statistical Software https://www.jstatsoft.org/
published by the Foundation for Open Access Statistics https://www.foastat.org/
July 2025, Volume 113, Issue 10 *Submitted:* 2024-04-18
doi:10.18637/jss.v113.i10 *Accepted:* 2024-08-27