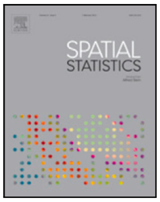
Spatial Statistics 70 (2025) 100935

Contents lists available at ScienceDirect

Spatial Statistics

journal homepage: www.elsevier.com/locate/spasta



A framework for analysing point patterns on nonconvex domains using visibility graphs and multidimensional scaling

Kabelo Mahloromela *, Inger Fabris-Rotelli

Department of Statistics, University of Pretoria, Pretoria, 0028, Gauteng, South Africa

ARTICLE INFO

Keywords:
Point pattern
Nonconvex window domain
Euclidean distance
Visibility graph
Multidimensional scaling

ABSTRACT

A point pattern is typically analysed to understand the first- and second-order properties of the underlying point process. These properties are usually inferred using estimation procedures that depend on interpoint distance and are thus sensitive to the choice of distance metric. Euclidean distance is conventionally used to quantify proximity between points, but it does not accurately reflect spatial relationships when points are constrained within irregular, nonconvex spatial domains. Herein, we propose a strategy to embed visibility graph distances into Euclidean metric space using multidimensional scaling. The aim is to simplify analyses, leverage well-developed methods based on Euclidean distance, and retain, as far as possible, the true proximity relationships on a nonconvex spatial domain. The kernel smoothed intensity estimate and the K-function are computed in this new spatial context and used to validate the effectiveness of the embedding strategy.

1. Introduction

The properties of a point pattern are traditionally determined using distance- and density-based measures (Cressie, 1993; Baddeley et al., 2015; Ripley, 1976; Diggle, 1985; Moller and Waagepetersen, 2003). The choice of distance metric is important for accurately characterising spatial relationships and dependencies between points within a given study area. While the Euclidean distance is often used to quantify proximity between objects in a point pattern, its application on nonconvex domains may lead to results that incorrectly represent the restrictions imposed by the domain's geometry (Mahloromela et al., 2023; Baddeley et al., 2022; Ferraccioli et al., 2021).

Euclidean distance measures the straight-line distance between two points, implicitly assuming a direct and unobstructed path (Longley and Batty, 2003). For convex domains, this approach is appropriate since the path between all pairs of points are unobstructed. In nonconvex domains, however, the shortest path between two points may be constrained by the boundaries of the window. Thus, the actual path between two points must circumvent these obstacles. The Euclidean distance, in this setting, will underestimate the true distance between points. For example, in ecological studies where animal movements are restricted by landscape features (e.g., rivers, mountains), the Euclidean distance does not capture the cost or difficulty of traversal, potentially leading to incorrect conclusions about habitat use or species interactions (Peck, 1999).

The challenge of incorporating the structure of a domain in point pattern analysis has been considered for non-planar cases such as points on a linear network (Baddeley et al., 2021, 2015; Yamada and Thill, 2004). Okabe and Sugihara (2012), for instance, developed techniques to address the methodological pitfalls that arise when points are confined to linear network spaces. Some adaptations of point pattern statistical summary measures have also been proposed for their appropriate use on linear network

---

* Corresponding author.
E-mail address: kabelo.mahloromela@up.ac.za (K. Mahloromela).

https://doi.org/10.1016/j.spasta.2025.100935
Received 18 December 2024; Received in revised form 14 September 2025; Accepted 17 September 2025

spaces. These include the network kernel smoothed intensity estimate and network K-function (Okabe and Yamada, 2001; Okabe and Sugihara, 2012; Baddeley et al., 2021).

Techniques have been developed to estimate density and intensity over planar domains with complex irregular boundaries. Ferraccioli et al. (2021), in their paper, present a method that combines a likelihood approach with a regularisation based on a differential operator. Baddeley et al. (2022) reconcile and extend the statistical theory of diffusion kernel estimation into a practical methodology for analysing point pattern data. More broadly, beyond density estimation, there has been growing interest in modelling data distributed over complex planar domains; this can be seen for example in Bakka et al. (2019). In their work, Bakka et al. (2019) develop a barrier model that defines spatial dependence by blocking paths across physical barriers within a simultaneous autoregressive framework, formulated as a stochastic partial differential equation to ensure a well-defined Gaussian field with a sparse, positive-definite precision matrix.

Empirical studies have demonstrated the advantages of using distance metrics alternative to the Euclidean distance in nonconvex domains. One example is in epidemiological research, where accounting for the actual paths of human movement within urban environments improved the understanding of disease spread and transmission dynamics (Eubank et al., 2004).

The problem of finding a representative distance metric between points, one that respects the window boundaries and circumvents obstacles in the window domain, can be solved by computing the length of the Euclidean shortest path. The Euclidean shortest path (Li and Klette, 2011) is the shortest path that can be traversed between two points in a polygon whilst avoiding obstacles (i.e., boundaries and holes). The Euclidean shortest path is computed using a visibility graph (Ghosh, 2007; Li and Klette, 2011). A visibility graph of a polygon is a graph that connects vertices that are mutually visible. Vertices are considered mutually visible if the line segment connecting them lies entirely within the polygonal domain, without intersecting any edges or holes. Each edge of the visibility graph can be labelled with the length of the edge connecting the intervisible vertices of the polygon. A shortest path finding algorithm can then be applied to the visibility graph to find the Euclidean shortest path (Li and Klette, 2011).

Capturing proximity relationships using a visibility graph in point patterns constrained on nonconvex windows is valuable and has been applied in Mahloromela et al. (2023) to modify the kernel smoothed intensity estimate. The visibility graph provides a means to handle complex geometric structures in real-world applications, particularly in settings where traditional Euclidean distance is not valid; especially when correctly describing spatial interactions is important for understanding movement or spatial connectivity.

Despite its usefulness in appropriately quantifying proximity relationships, constructing the visibility graph and using it in point pattern analysis algorithms can be computationally intensive. Performing a visibility query (i.e., determining whether a line of sight exists between two points) for a set of fixed coordinates in a grid during the process of kernel smoothing in a domain with many obstacles can be computationally expensive as the complexity, number of pixel cells, and domain obstacles increase (Ghosh, 2007). Techniques developed to try address the computational challenges of optimising the visibility graph construction include (Ghosh, 2007; Overmars and Welzl, 1988; Ghosh and Mount, 1991; Hershberger, 1989). While visibility graph construction optimisation algorithms are invaluable for reducing computational resources in some route planning applications (Shah and Gupta, 2016; Wu et al., 2021), their utility in point pattern analysis is constrained by scalability issues.

Owing to the scalability issues, we are interested in embedding the distances computed on the visibility graph in a Euclidean metric space to leverage the computationally optimal techniques to calculate distance in such spaces. The technique usually employed to do this is multidimensional scaling (MDS) (Torgerson, 1952; Borg and Groenen, 2007). MDS is used to represent the pairwise distances among q elements as a configuration of q points positioned in a geometric space (typically Euclidean). The embedding seeks to minimise the error between the original distances and the new distances produced in the embedded space.

Herein, we provide a framework to analyse point patterns within nonconvex spatial domains. We apply a graph theoretic approach using a weighted visibility graph to determine distance along a path-connected space in the spatial window. The resultant distances are embedded in Euclidean metric space using multidimensional scaling. The embedding strategy allows the use of the available statistical summary measures that are well established for convex spatial windows.

The remainder of this paper is organised as follows. Section 2 provides some definitions and background theory. Section 3 gives a presentation of the proposed framework for analysing point patterns on nonconvex windows. Section 4 is dedicated to a simulation experiment that compares our proposed strategy to other established methods in the case of density estimation. Following this, we consider a real world application in Section 5, and a discussion and conclusion in Section 6.

2. Background theory

We begin with the definition of a point pattern data set and its first- and second-order properties in Sections 2.1 and 2.2, respectively. Sections 2.3 and 2.4 follow with a discussion on weighted visibility graphs and shortest-path distances, respectively. Multidimensional scaling is outlined in Section 2.5. The proposed framework for analysis, using visibility-graph based distances and multidimensional scaling, is presented in Section 3.

2.1. Spatial point pattern data

A point pattern (Baddeley et al., 2015; Cressie, 1992; Illian et al., 2008) refers to a set of points, $x = \{x_1, \ldots, x_n\}$, in a defined region of space $W \subset \mathbb{R}^2$, where $n \geq 0$ is not fixed in advance. The points represent the locations of objects, entities, or events. The main aim of an analysis is typically to understand the underlying random mechanism, termed a point process and denoted by X, that produced the point configuration. A scientific enquiry into the properties of a point pattern could involve assessing its departure from stationarity and isotropy. A point process is considered stationary if it is invariant under translations and isotropic if it is invariant under rotations.

To analyse point patterns, the data must comprise the specific locations $\{x_i\}$ of points and the window W where points are observed. These components are important since the properties of point patterns inherently rely on the definition of the window and how proximity relationships are determined on it (Baddeley et al., 2015; Diggle, 1985; Ripley, 1976; Cressie, 1992).

2.2. First- and second-order properties

Point processes are commonly characterised using their first- and second-order intensity measures. The first-order intensity (Cressie, 1993; Baddeley et al., 2015; Moller and Waagepetersen, 2003; Ibe, 2013) at a location $u \in W$ is defined as

$$\lambda(u) = \lim_{|du|\to 0} \frac{E[N(du)]}{|du|}, \tag{1}$$

where du is a small region around u, $E[\cdot]$ is the expected value operator, $|du|$ is the area of the small region, and $N(du)$ is the number of points in the region du. The intensity is the average number of points per unit area. It is a constant λ for stationary isotropic point processes. The kernel smoothed intensity estimate (Cressie, 1993; Baddeley et al., 2015; Moller and Waagepetersen, 2003) is commonly used as an estimator of varying intensity. The intensity estimate is given by

$$\hat{\lambda}(u) = \sum_{i=1}^{n} \frac{1}{e_\sigma(u)} \kappa_\sigma(\|u - x_i\|), \tag{2}$$

where $\|\cdot\|$ is the Euclidean distance, σ is the bandwidth that controls the smoothness of the estimator, $\kappa_\sigma(u) = \sigma^{-2}\kappa(\frac{u}{\sigma})$, and $\kappa(\cdot)$ is a kernel function, usually a probability density on $\mathbb{R}^2$, and $e_\sigma(u) = \int_W \kappa(u - y)dy$ is a bias correction term for boundary effects.

The second-order intensity (Cressie, 1993; Baddeley et al., 2015; Ripley, 1976; Moller and Waagepetersen, 2003) of a point process is defined as

$$\lambda_2(u, v) = \lim_{\substack{|du|\to 0\\ |dv|\to 0}} \frac{E[N(du) \times N(dv)]}{|du| \times |dv|}, \tag{3}$$

where du and dv are small regions around the points u and v respectively, and their areas are denoted by $|du|$ and $|dv|$. The second-order intensity is an approximate probability that there is at least one point of the point process in each of the regions du and dv. For a stationary isotropic spatial point process, the reduced second-moment measure, termed the K-function, is related to the second order intensity through the expression

$$K(r) = \frac{1}{\lambda} \int_0^{2\pi} \int_0^r u\lambda_2(u, \theta)dud\theta, \tag{4}$$

where r is the distance (usually Euclidean) from points in the pattern and θ, the angle of the direction (Cressie, 1993; Ibe, 2013). The K-function measures the relative expected number of points within a distance r of a randomly chosen point. If a point process exhibits clustering, each point is typically surrounded by more points from the same cluster, resulting in a relatively high value of $K(r)$ for small values of r. Conversely, if a point process is randomly dispersed in space, each point is more likely to be isolated, leading to a lower value of $K(r)$ for small r. Under complete spatial randomness, $K(r) = \pi r^2$. When $K(r) > \pi r^2$, we have an indication of clustering, while $K(r) < \pi r^2$ suggests that points are more regularly spaced or exhibit repulsion. The edge-corrected empirical K-function, $\hat{K}(r)$, is commonly used as an estimator of $K(r)$ and is defined as

$$\hat{K}(r) = \frac{|W|}{n(n-1)} \sum_{i=1}^{n} \sum_{\substack{j=1\\ j\neq i}}^{n} I(\|x_i - x_j\| \leq r)e_{ij}(r), \tag{5}$$

where $I(\cdot)$ is an indicator function and $e_{ij}(r)$ is a weight to correct for edge effects, e.g., a border or isotropic correction (Ripley, 1976; Moller and Waagepetersen, 2003; Baddeley et al., 2015). For the case of an isotropic correction, the contributions are reweighted using the proportion of the circumference of a circle centred at x_i, passing through x_j, and contained in W. Additional second order summary quantities are available that make use of the nearest-neighbour and empty-space distances. The reader is referred to Baddeley et al. (2015), Moller and Waagepetersen (2003) for a discussion of these.

Eqs. (2) and (5) are commonly used summary measures (among others) that use the Euclidean distance to characterise the first- and second-order nature of a point pattern. The Euclidean distance, however, is not a representative measure of proximity on nonconvex domains since it assumes an unobstructed connected path between points. Consequently, spurious estimates are produced and artefacts arise. Owing to this, we propose using a weighted visibility graph to find representative proximity relationships.

2.3. Weighted visibility graph

A weighted visibility graph (Ghosh, 2007) of a polygon P, denoted by $G(P)$, is an ordered triple (V, E, w) comprising

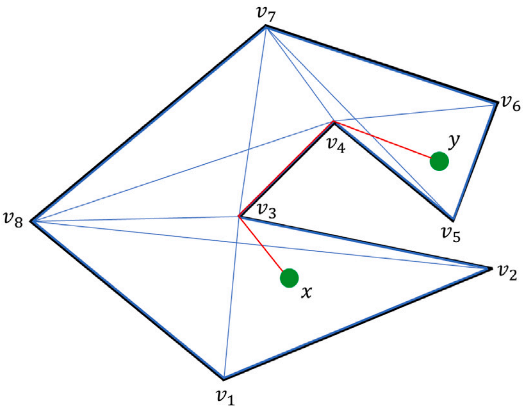


Fig. 1. Shortest path (red line) between two points (green dots) and visibility graph (blue lines) of a nonconvex polygon.

- $V = \{v_1, \dots, v_m\}$, a set of the vertices[1] of P,
- $E = \{(v_i, v_j) | v_i, v_j \in V \text{ and } v_i \neq v_j \text{ for } i, j = 1, \dots, m\}$, a set of edges between vertices that are intervisible (in a geometric context, two vertices v_i and v_j, $v_i \neq v_j$, are visible to each other if the line segment connecting them does not intersect the outer boundary of P or hole boundaries, that is $\alpha v_i + (1-\alpha) v_j \in P,\ \forall \alpha \in (0,1)$), and
- $w : E \mapsto \mathbb{R}^+$, a function mapping edges to the value of the Euclidean distance between the vertices it connects, i.e., $w(v_i, v_j) = \|v_i - v_j\|$ for $(v_i, v_j) \in E$.

A representative distance measure between two points in a point pattern, one that quantifies the length of a path that does not cross the window's boundary and holes in its domain, can be found by determining the shortest path on the weighted visibility graph of the window.

2.4. Shortest path distance

A path (Ghosh, 2007; Li and Klette, 2011) from x to y is a finite sequence of vertices $Q = (v_0 = x, v_1, \dots, v_{k-1}, v_k = y)$ such that $(v_h, v_{h+1}) \in E\ \forall h = 0, \dots, k-1$; the path starts at x and follows adjacent vertices via edges in the graph until it reaches y. The shortest path from x to y, denoted by Q^*, is defined as

$$Q^* = \arg\min_{Q} \sum_{h=0}^{k-1} w(v_h, v_{h+1}). \tag{6}$$

The shortest path distance,

$$d(x, y) = \min_{Q} \sum_{h=0}^{k-1} w(v_h, v_{h+1}), \tag{7}$$

provides a measure of distance within the spatial domain that avoids obstacles, such as boundaries and holes. Fig. 1 shows the shortest path (red line) between two points (green dots) in a nonconvex polygon. The visibility graph (blue lines) of the polygon is also shown.

To incorporate visibility into point pattern analysis, we embed the metric space[2] with a shortest path distance metric into Euclidean space using multidimensional scaling. This method simplifies the interpretation of complex, path-dependent, distances while preserving the original space's geometric and topological properties. It also allows for the application of point pattern techniques suited for convex spaces whilst improving estimation of first- and second-order properties of point patterns constrained on nonconvex domains.

2.5. Multidimensional scaling (MDS)

Multidimensional scaling (MDS) (Torgerson, 1952; Borg and Groenen, 2007) is a technique used to convert the distances between each pair of q elements in a set into a configuration of q points positioned within an abstract Cartesian space.

Given a distance matrix $D = [d_{ij}]$, where $d_{ij} = d(x_i, x_j)$ is the distance metric between the points x_i and x_j in the set, and a specified number of dimensions, $p < q$, the MDS algorithm positions each object within a p-dimensional space (a reduced-dimensional

---

[1] In the case of a polygon with holes, the vertices of the hole boundaries are included in the visibility graph vertex set.

[2] A metric space (X, d) is a set X with a metric d that defines a notion of distance between its elements.

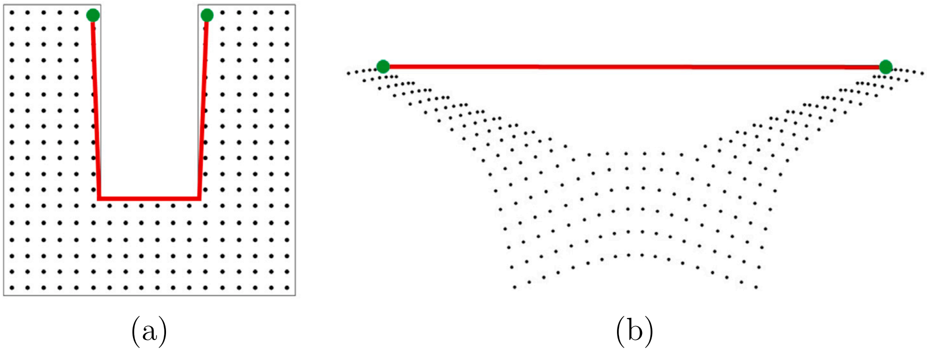


Fig. 2. An illustration of the result of embedding the shortest path distances on a weighted visibility graph in a Euclidean space. (a) shows the shortest path in the original nonconvex domain and (b) shows the result after applying MDS.

representation) to maintain the original distances between objects as closely as possible. Metric MDS is applied to quantitative dissimilarities and non-metric to qualitative (ordinal) dissimilarities (Borg and Groenen, 2007). We deal with the former.

In classical MDS the aim is to minimise the strain loss function (Borg and Groenen, 2007; Torgerson, 1952; Mardia, 1978),

$$\text{Strain} = \sum_{i<j} (\delta_{ij}^2 - d_{ij}^2), \tag{8}$$

where δ_{ij} are the low-dimensional distances produced from the solution. The solution can be obtained analytically for $\delta_{ij} = \|y_i - y_j\|$, $y_i, y_j \in \mathbb{R}^p$ via a matrix eigendecompositions (Mardia, 1978). Usual metric MDS seeks to find an optimal configuration of points that minimises the stress loss function

$$\text{Stress} = \left(\frac{\sum_{i<j} (\delta_{ij} - d_{ij})^2}{\sum_{i<j} d_{ij}^2} \right)^{\frac{1}{2}}. \tag{9}$$

A solution can then be obtained using numerical optimisation methods. A thorough treatment of these techniques along with other loss functions is provided in Borg and Groenen (2007). In this research we consider Sammon mapping (Sammon, 1969; Borg and Groenen, 2007). Sammon mapping (Sammon, 1969) aims to minimise the error

$$\text{Sammon Stress} = \frac{1}{\sum_{i<j} d_{ij}} \sum_{i<j} \frac{(\delta_{ij} - d_i j)^2}{d_{ij}}, \tag{10}$$

known as Sammon's stress or Sammon's error.

Fig. 2 is an illustrative example that shows the effect of applying classical MDS on the shortest path distances of the weighted visibility graph. The new shape produced within the abstract space is not necessarily convex, but the distances are approximately Euclidean and preserved the original distances as much as possible. Thus, there will be slight deviations between the distances computed from the original space and the distances produced in the embedded space.

3. Methodology

In this section, a methodology is provided for computing spatial measures in a setting where the points in a point pattern are constrained by the boundaries of a nonconvex window polygon.

3.1. An outline of the proposed framework

The MDS procedure works on a finite set of points. To get a discrete representation of the spatial domain, we subdivide the window polygon using a triangular mesh. A mesh is a discretised representation of a geometric domain, often used in computational modelling to approximate complex shapes. Fig. 3(a) depicts a triangular mesh created on a nonconvex window and Fig. 3(b) shows the transformed mesh after applying MDS. The centres or vertices of the triangles in the mesh are used as the finite set of points over which the weighted visibility graph is constructed.

The shortest paths between all pairs of nodes are computed on the resulting weighted visibility graph, and the sums of the edge weights along these paths are used to construct the distance matrix. The distance matrix provided from this step is used to embed the points in a Euclidean metric space using metric MDS. The results of applying this procedure to a U-shaped, S-shaped, and irregular window is depicted in Figs. 4, 5, and 6. For each window, a scree plot is shown with the number of dimensions on the horizontal axis and the stress (i.e., Eq. (9)) on the vertical axis. It can be observed in Figs. 4(b), 5(b), and 6(b) that stress decreases as the number of dimensions increases. There are noticeable "elbows" in the curves, where the decrements in stress begin to be less pronounced. In the case of the U-shaped window the elbow appears at three-dimensions, the S-shaped window at five-dimensions,

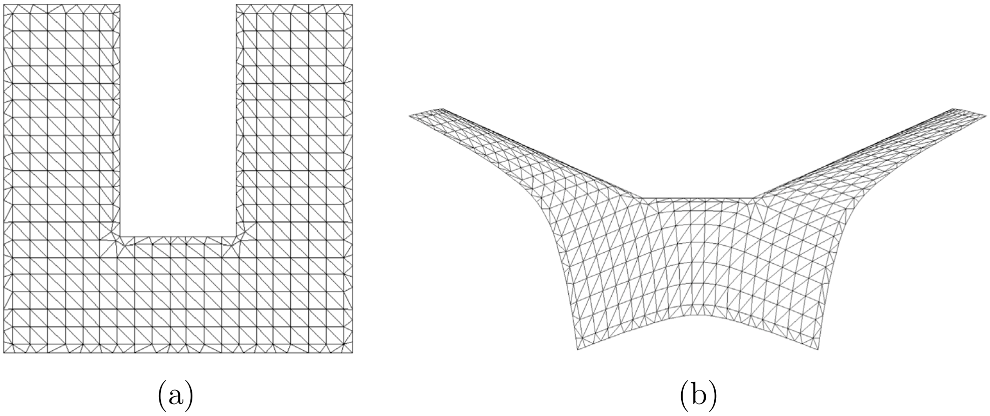


Fig. 3. A discretisation of a nonconvex polygon using a triangular mesh in (a) and its corresponding representation in Euclidean space after the application of MDS in (b).

and the irregular window at four-dimensions. The resulting two- and three-dimensional embeddings from the MDS procedure are shown in panels (c) and (e) of each figure, respectively. The embeddings are transformed so that their orientations align with that of the original spatial window.[3]

A Shepard diagram (Borg and Groenen, 2007) for the two- and three-dimensional cases are provided in (d) and (f) of each figure, respectively. The Shepard diagram is a plot of the weighted visibility graph distances, i.e., the d_{ij}'s, on the horizontal axis against the corresponding MDS distances δ_{ij} on the vertical axis. The line where $d_{ij} = \delta_{ij}$ is shown in red. Observable from these diagrams is the reduction in the spread around the line $d_{ij} = \delta_{ij}$ when the dimensions increase from two to three. This is the typical observed behaviour as the number of dimensions increase in all cases we considered. Thus, it is valuable to project to dimensions higher than three-dimensions in terms of optimising precision of estimates based on these distances; even though we may not be able to visualise the points in the new space.

Once a decision is made regarding the number of dimensions p in which to project, the observed points $x = \{x_1, \dots, x_n\}$ in the original space, $W \subset \mathbb{R}^2$, are transformed to the new metric space using the transformation

$$T_l(x_i) = A_l x_i, \tag{11}$$

where $T_l : \mathbb{R}^2 \mapsto \mathbb{R}^p$ and A_l is a transformation matrix that maps a point x_i from the triangle in which it lies on the triangular mesh of the window W to its counterpart in the new metric space.

To compute the spatial summary measures for the first- and second-order properties of the point pattern, we apply tradition statistical methodology based on the Euclidean distance on this new space and transform it to the original space. An illustration is provided for the kernel smoothed intensity estimate and the K-function in the following section.

3.2. Illustrating the framework with kernel smoothed intensity estimation and the empirical K-function

In this section, we illustrate the use of our proposed framework by considering kernel smoothing and Ripley's K-function.

Consider the point pattern depicted in Fig. 7. The points are realised from a process with inhomogeneous intensity, i.e., there is a higher density of points in the upper right corner of the nonconvex window as opposed to the bottom left corner. The points are simulated using the approach described in the simulation study detailed later in this document, i.e., from a Gaussian density that is a function of the Euclidean shortest path distance. We apply the methodology outlined in Section 3.1 as follows. The window is discretised using an irregular[4] triangular mesh. The result is shown in Fig. 8(a). A weighted visibility graph of the window polygon is constructed. The inter-point distances between the vertices of the triangles in the mesh are then determined using the length of the shortest path in the weighted visibility graph. The resultant distances are embedded in a Euclidean metric space using metric MDS. The result of which was already shown in Fig. 6 We perform multidimensional scaling using four-dimensions since this was the dimension at which the elbow on the scree plot was observed. For illustrative purposes, the result in two-dimensions is shown in Fig. 8(b). The new locations of the observed points in the Euclidean metric space are then determined by using a linear mapping from the triangle in which it originally lies to the corresponding triangle in four-dimensional Euclidean metric space. This new domain now forms the basis on which analysis will be performed.

---

[3] While MDS preserves the relative distances between points, the absolute orientation of the configuration is arbitrary. To compare the MDS embedding with the original spatial configuration, a post hoc alignment is required, typically involving optimal rotations and translations or manual adjustments based on visual inspection.

[4] The choice between using a regular or irregular triangular mesh has no observable influence on the result of the final estimates provided.

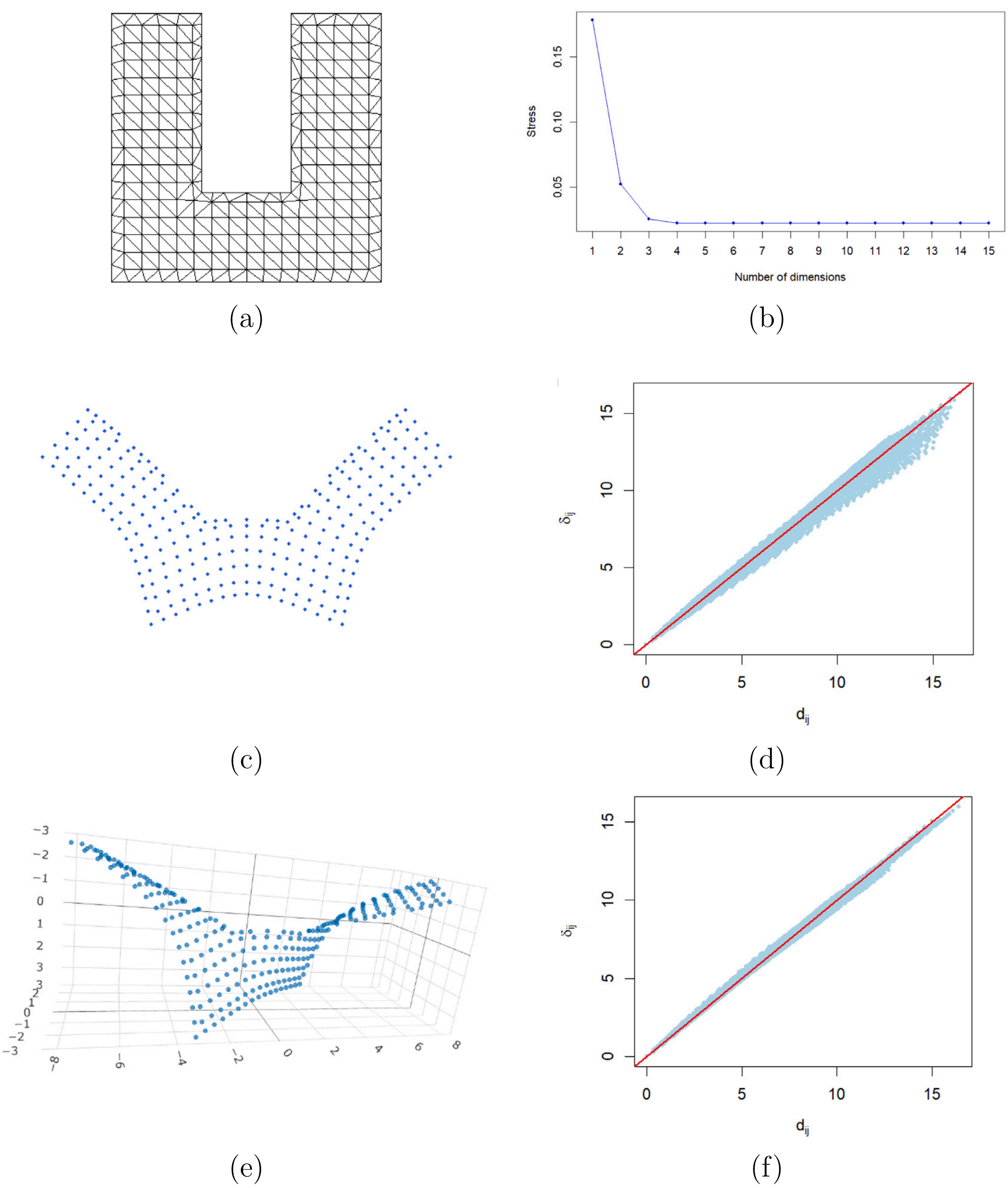


Fig. 4. The stress values for metric MDS in different dimensions are shown in (b) for the vertices extracted from the triangular mesh of the U-shaped window in (a). The representation of the points in two-dimensional and three-dimensional space is provided in (c) and (e), respectively. A plot of d_{ij} against δ_{ij} is provided in each case in (d) for the 2D case and (f) for the 3D case.

In MDS, the transformation applied to the points is not uniform nor globally consistent across the entire dataset. Instead, the location of each point in the embedded space is determined by its pairwise distances to all other points, as part of a global optimisation process that seeks to minimise the overall stress. Consequently, the transformation cannot be described as a single rigid mapping applied uniformly to all points, i.e., the way points within one triangle of the mesh are transformed can differ from the way that points in another triangle are transformed. Thus, we perform local linear mappings between corresponding triangles in the original and embedded spaces, allowing us to map points within a triangle in a locally coherent way. Although minor non-affine distortions may persist locally, their impact is expected to be negligible in the context of the broader spatial analysis. The use of scree plots and Shepard diagrams to select an optimal embedding dimension does help in reducing both the global and local distance

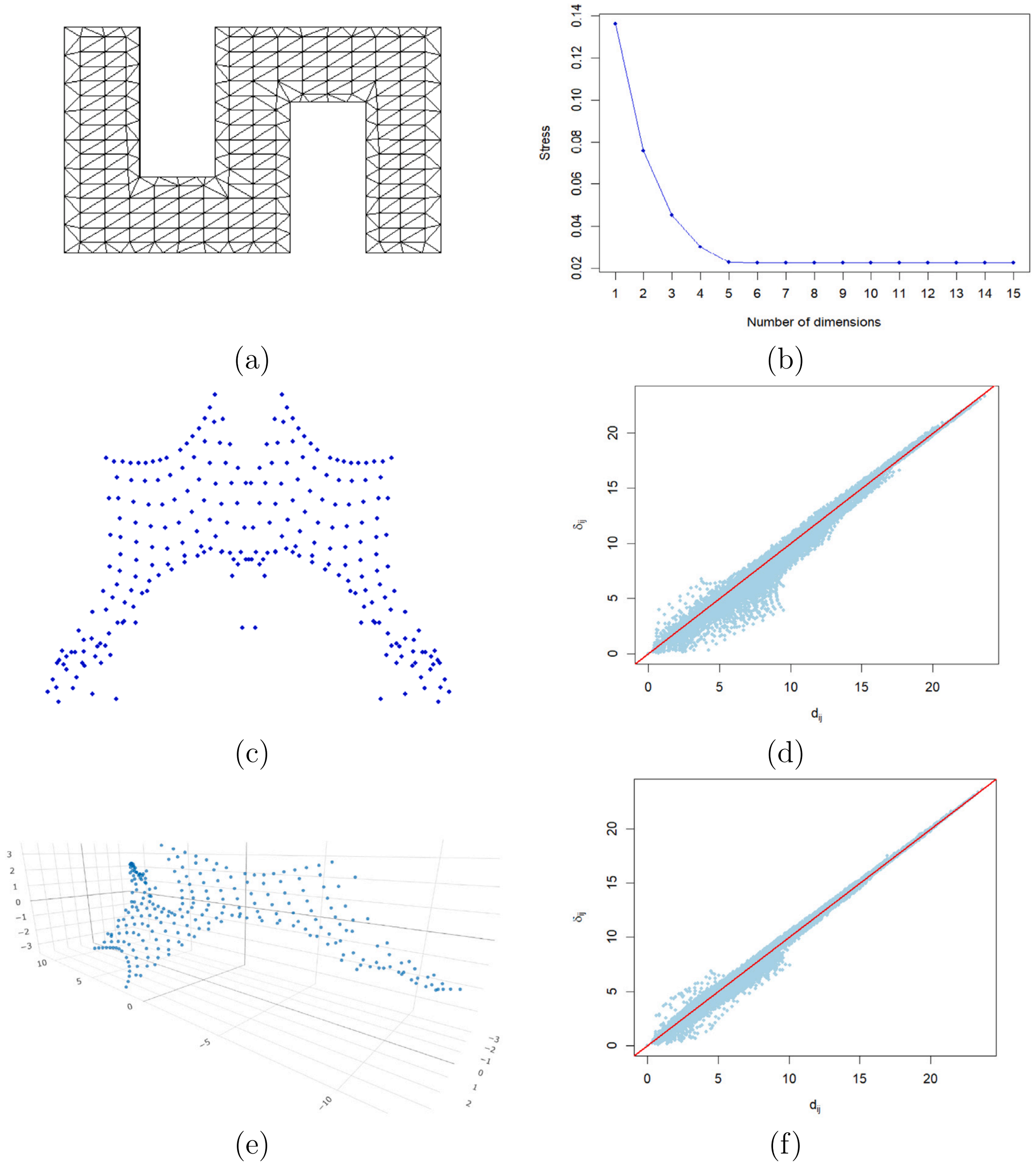


Fig. 5. The stress values for metric MDS in different dimensions are shown in (b) for the vertices extracted from the triangular mesh of the S-shaped window in (a). The representation of the points in two-dimensional and three-dimensional space is provided in (c) and (e), respectively. A plot of d_{ij} against δ_{ij} is provided in each case in (d) for the 2D case and (f) for the 3D case.

discrepancies. Visual inspections of the mapped triangles and the results return by their use in the simulation study that follows in the next section further support the conclusion that the local geometric integrity is sufficiently maintained for the purposes of this type of analysis.

In the case of the smoothed estimate of intensity, an estimate at $u \in W$ is determined using the expression,

$$\hat{\lambda}_T(u) = \sum_y \frac{1}{e_\sigma(u)} \kappa_\sigma(\|T_u(u) - T_y(y)\|), \tag{12}$$

where $T_y(\cdot)$ and $T_u(\cdot)$ are mappings from the triangle t_y (in which y lies) and t_u (in which u lies), respectively, to the corresponding triangles in Euclidean metric space. The result of using this as an estimator of intensity is depicted in Fig. 9(b) and the result of

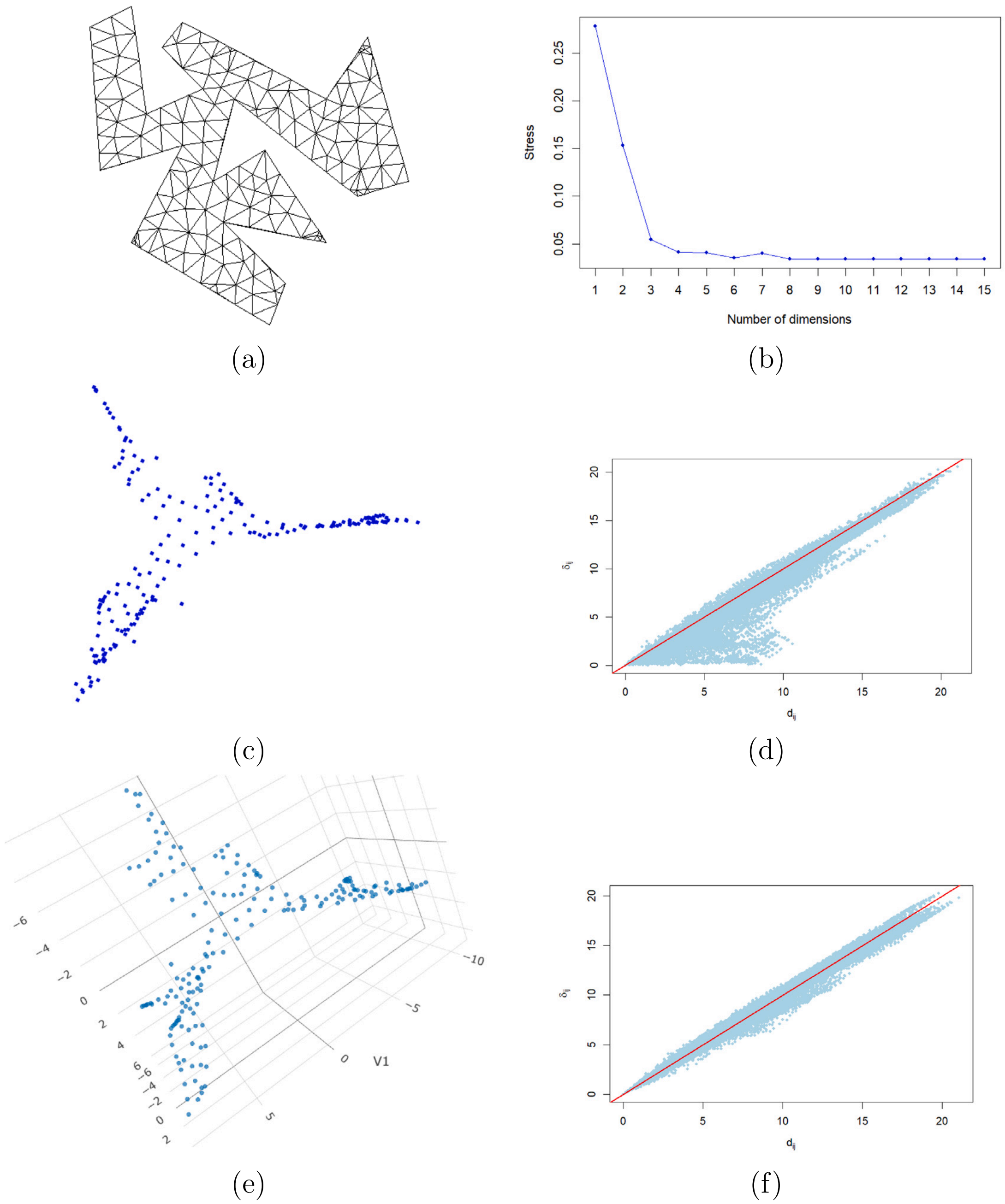


Fig. 6. The stress values for metric MDS in different dimensions are shown in (b) for the vertices extracted from the triangular mesh of the irregular window in (a). The representation of the points in two-dimensional and three-dimensional space is provided in (c) and (e), respectively. A plot of d_{ij} against δ_{ij} is provided in each case in (d) for the 2D case and (f) for the 3D case.

using Euclidean distance on the original nonconvex window is show in Fig. 9(a). For comparison purposes, an uncorrected kernel estimator with a bandwidth value of 1 is used for both estimates. In the case of the estimate computed using Eq. (12), the influence exerted by a point spreads through a continuous path within the domain. When Euclidean distance is used in this domain, the influence of a point diffuses radially based on a straight line that disregards window boundaries. Thus, the contribution of the point to the intensity estimate remains substantial, even when movement between points is restricted by the domain's boundary.

In the case of the K-function, the empirical function is determined using the expression,

$$\hat{K}_T(r) = \frac{|W|}{n(n-1)} \sum \sum\nolimits_{i \neq j} I(\|T_i(x_i) - T_j(x_j)\| \le r), \tag{13}$$

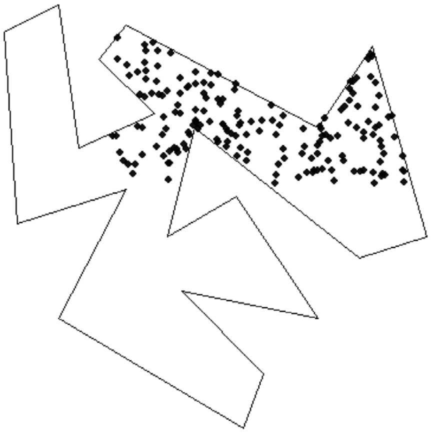
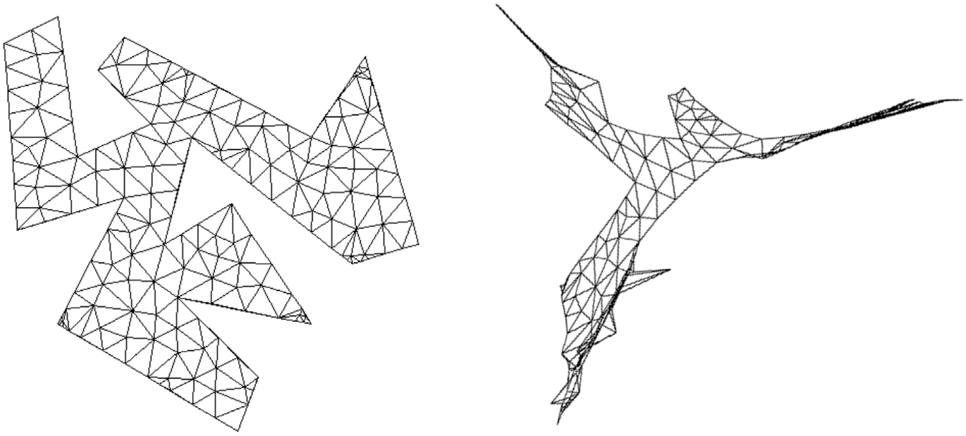
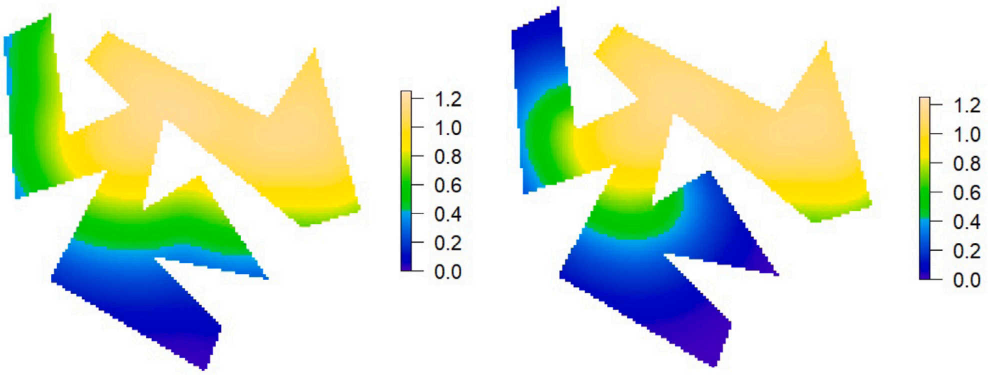


Fig. 7. A point pattern realised from a point process with inhomogeneous intensity.



(a) (b)

Fig. 8. A discretisation of a nonconvex polygon using a triangular mesh in (a) and its corresponding representation in Euclidean space after the application of MDS in (b).



(a) (b)

Fig. 9. A kernel smoothed intensity estimate produced using (a) Eq. (2) and (b) Eq. (12).

where $T_i(\cdot)$ and $T_j(\cdot)$ here again denotes the mapping of the points x_i and x_j from the triangles in which they lie in W to the corresponding triangles in p-dimensional Euclidean metric space. The left panel of Fig. 11 shows the results obtained using Eq. (5), while the right panel of the figure presents the results obtained using Eq. (13) for a clustered, random, and regular point pattern depicted in Fig. 10. Simulation envelopes (Baddeley et al., 2015) (blue lines) are computed under the assumption that the pattern is random. The red curves represent the empirical K-function values for the observed point patterns.

The Euclidean-based K-function and the weighted visibility graph-based K-function produce broadly similar results in detecting clustering and regularity. While both methods reach comparable conclusions regarding spatial structure, very subtle differences are observable. In the given examples, although extremely modest, the weighted visibility graph-based K-function exhibits a slightly enhanced sensitivity to both pattern regularity and clustering. Although these differences are not stark, they are consistent with the premise that path-based distances do provide a representative measure of spatial relationships in constrained environments. This perspective is further motivated by findings in non-planar settings, such as point patterns on linear networks, where adapting the K-function to a linear network context has proven useful (see for example (Yamada and Thill, 2004)). This effect in the planar case may be less exaggerated and future work will further explore the advantages of this approach for point patterns confined within nonconvex window domains.

In this section, we have outlined a framework to embed distances calculated on a weighted visibility graph in Euclidean space. The method starts with a discretisation of the original nonconvex window using a triangulation or triangle mesh. The inter-point distances between the vertices on the triangular mesh are computed along the weighted visibility graph. Multidimensional scaling is then applied to the resultant distances to produce a point configuration in a dimensional space admitted by observing the dimension at which the elbow appears on the scree plot. The triangles in the new space are matched with their counterparts in the original space and a local mapping between triangles performed to preserve local visibility. For the purposes of exposition, we provide a discussion of both the techniques of first- and second-order intensity estimation in our given framework. The remainder of this document, however, is dedicated to first-order intensity estimation, and second-order properties will be further detailed in future work.

4. Simulation experiments on density estimation

In this section, a simulation study is conducted to compare the performance of our proposed framework in the case of density estimation (henceforth referred to as VG-KS, i.e., visibility-graph based kernel smoothing), traditional kernel-smoothing (KDE), and other established methods designed to estimate density (or intensity) of phenomenon that is influenced by the shape of the spatial domain. In particular, we consider the diffusion smoother (DF) of Baddeley et al. (2022), in which the smoothing kernel is the heat kernel on the spatial domain, and the non-parametric density estimator proposed by Ferraccioli et al. (2021) that combines a likelihood approach with a regularisation based on a differential operator (DE-PDE). When a technique yields an intensity estimate, we scale it so that it integrates to one over the spatial domain, yielding a density. The implementation of the proposed methodology is facilitated via the use of the R packages `spatstat` (Baddeley et al., 2015), `inlabru` (Bachl et al., 2019), `MASS` (Venables and Ripley, 2002), and `igraph` (Csardi and Nepusz, 2006).[5]

In the simulation study, we consider two different spatial domains, namely an irregularly shaped spiral domain and a complex domain. For each domain, we define a simple density obtained by using a Gaussian density that is a function of the Euclidean shortest path distance and a non-trivial density that is a mix of these Gaussian densities for a set of given points in the window. Each density is scaled to integrate to one over the windows. The densities are shown in the left panel of Fig. 12. The corresponding point patterns generated for a single realisation with 500 independent points is shown on the right panel. Case 1 in Fig. 12(a) shows a Gaussian density defined on a spiral domain. Case 2, depicted in Fig. 12(b), also uses a spiral domain but features a mix of three Gaussian densities. Case 3, shown in Fig. 12(c), presents a Gaussian density on a more complex domain. Finally, Case 4 illustrated in Fig. 12(d) shows a mix of three Gaussian densities on the complex irregular window. In all four cases, the density is computed based on the Euclidean shortest-path distance.

In each of the different cases, we simulate 500 independent points. The simulations are repeated for 300 iterations. For each iteration, the KDE, DF, DE-PDE, and VG-KS methods are applied. In the case of the KDE, DF, and VG-KS, the bandwidth is selected as two for the spiral window, and three for the complex window. For DE-PDE, we tune the smoothing parameter so that the average effective smoothing range of this estimator is approximately equal to that of the chosen bandwidths in the kernel smoothing setting. The effective smoothing range (ESR) is defined as

$$\mathrm{ESR} = \sup\{r : f(x) > \epsilon \text{ for some } \|x - u_0\| = r\},$$

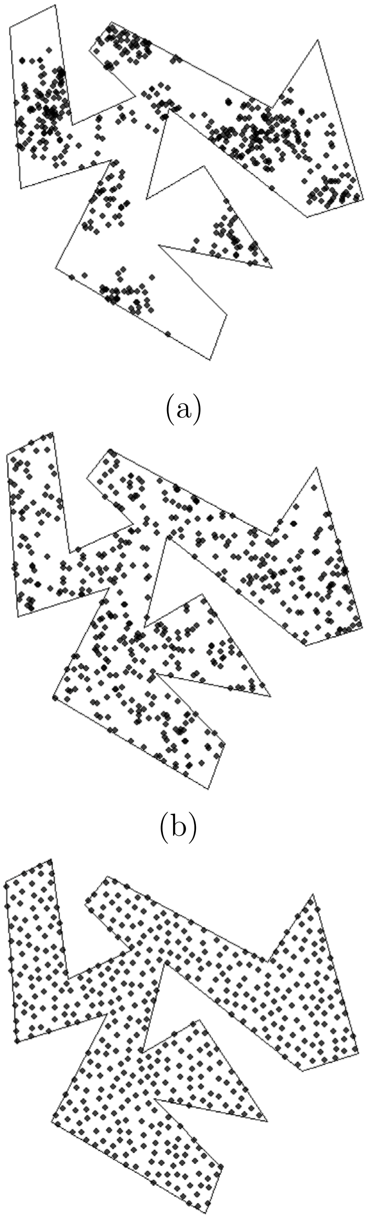
where $f(x)$ is the smoothed density, u_0 is the location of the point source that generates an impulse, and ϵ is a fixed small value. A numerical approximation of the integrated squared error (ISE),

$$ISE(\hat{f}) = \int_W (\hat{f}(u) - f(u))^2 du,$$

is then calculated for each estimator, where $f(\cdot)$ denotes the true density and $\hat{f}(\cdot)$ the estimated density. The results are presented in Figs. 13–16. Panel (a) in these figures show the true density, while panels (c)–(f) depict the mean estimates obtained by the various

---

[5] The reader may contact the authors for the code used to implement the methodology.



(a)

(b)

(c)

Fig. 10. Simulated point patterns: (a) clustered, (b) random, and (c) regular.

methods across the 300 point pattern realisations. Panel (b) shows the boxplots of the ISEs of each of the competing methods over the 300 point pattern realisations.

In Figs. 13 and 15, for the case of the trivial densities, it can be observed that KDE struggles to capture the underlying structure of the true density, and blends high-density and low-density regions that are separated by the domain boundaries. In both instances, KDE attains the worst performance with respect to the ISE. The results for the cases of non-trivial densities, depicted in Figs. 14 and 16, however, show that KDE outperforms DE-PDE at the given effective smoothing range, but still performs worse than the results produced by DF and VG-KS. In all cases, the estimates produced by VS-KS tend to outperform KDE and DE-PDE. The VG-KS method is, however, slightly outperformed by the DF method. The relatively poorer performance is attributed to the higher bias of the VG-KS technique near the boundary edges, which is better accounted for by the DF method. For future work, a method to

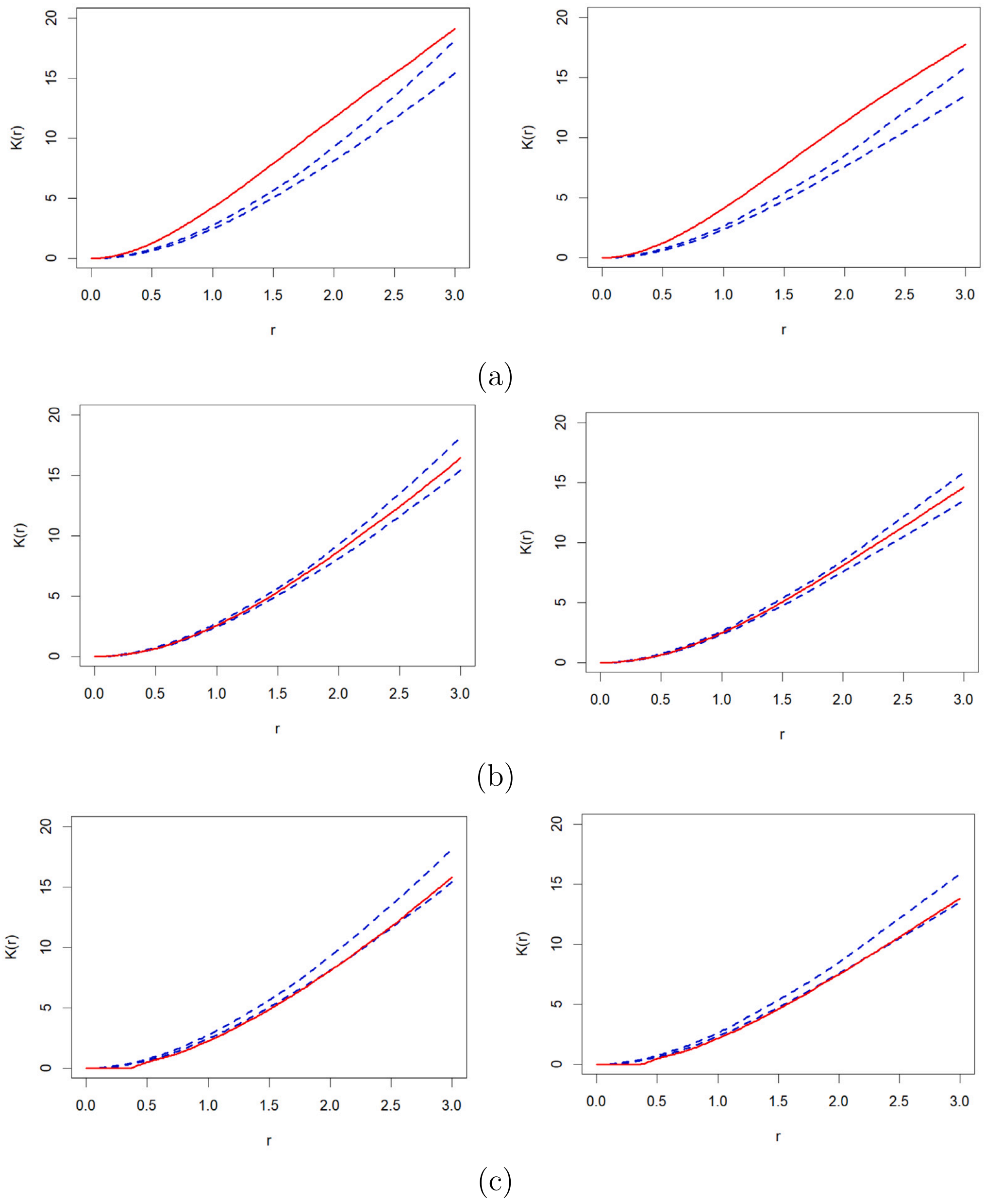
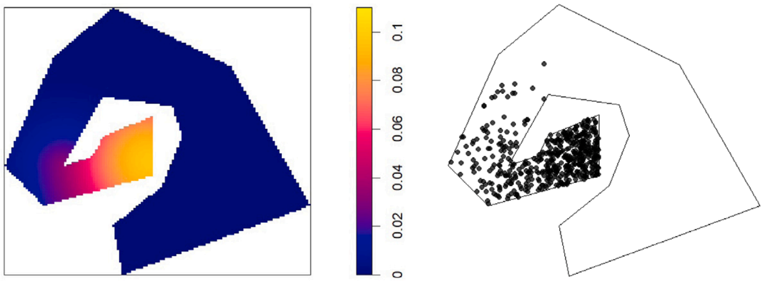


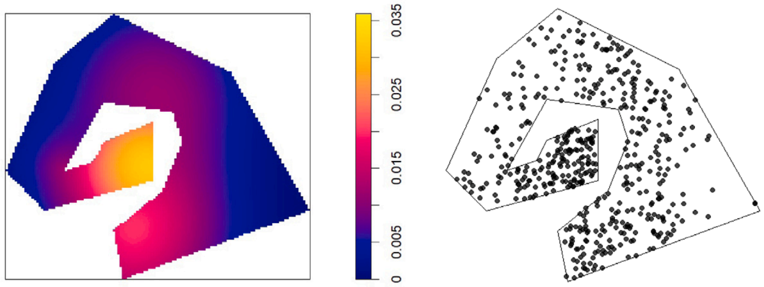
Fig. 11. Empirical K-function estimates computed using Eq. (5)(left) and Eq. (13)(right) for simulated patterns: (a) clustered, (b) random, and (c) regular.

better account for the edge bias in the proposed framework will be investigated. Despite its poorer performance relative to the DF method, the density estimator developed under the proposed framework is a good candidate for density (or intensity) estimation in settings where points are confined to nonconvex spatial domains which is demonstrated through the relatively low ISE.

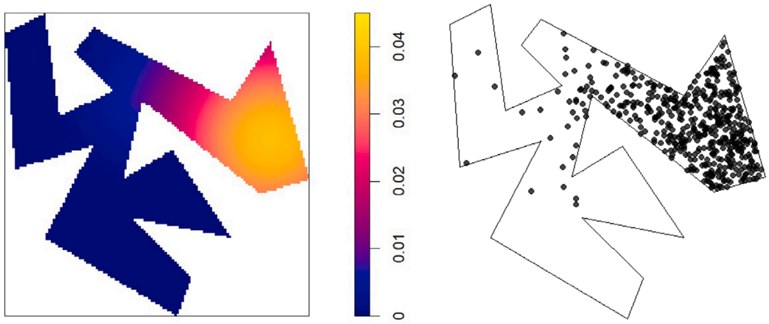
The computational cost of the algorithm warrants careful consideration, particularly the visibility graph construction, which is the most computationally intensive component. Building a visibility graph is expensive because it involves checking the line of sight between all pairs of vertices in the set, typically comprising the polygon boundary and any internal holes. This pairwise visibility check forms the dominant factor in the overall computational complexity. Consequently, optimising the visibility graph construction is critical for ensuring the algorithm's practical scalability, especially in complex environments where window domains with many vertices may arise. In the naïve approach, this results in a time complexity of $O(n^2)$, where n is the number of vertices (Ghosh, 2007). To address this, more computationally efficient algorithms have been developed, with the most optimal, based on our knowledge,



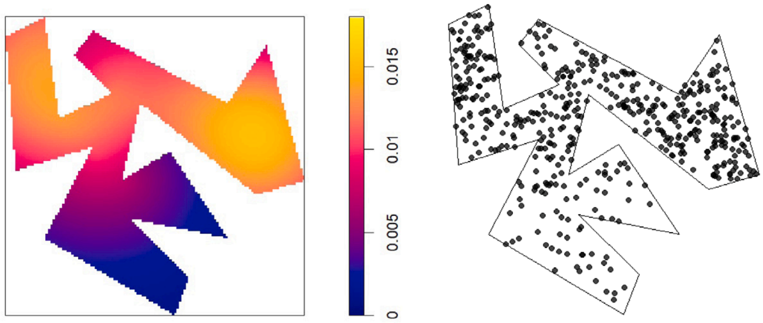
(a) Case 1



(b) Case 2

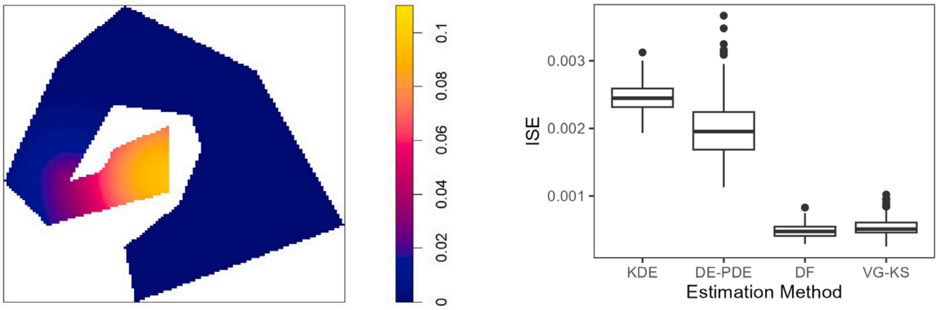


(c) Case 3

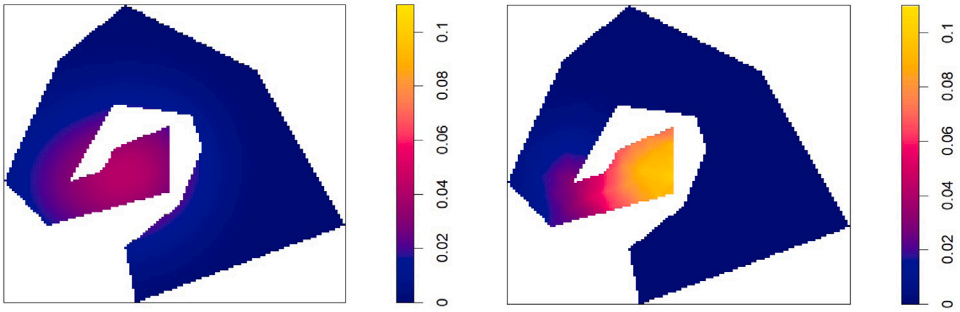


(d) Case 4

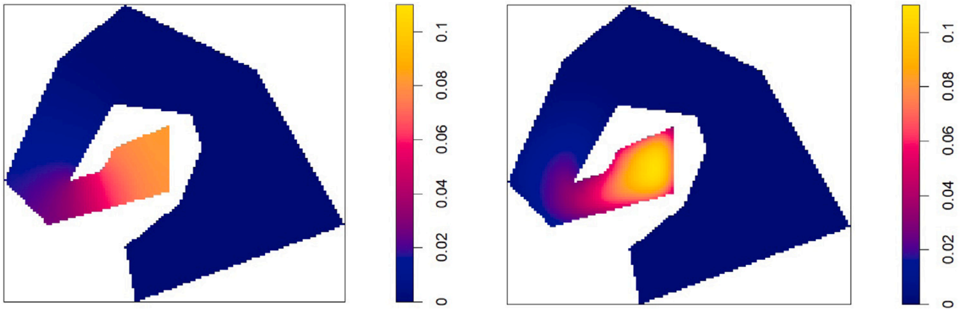
Fig. 12. An illustration of the density functions used for simulation and an example of the resulting point patterns based on four scenarios. The left panel shows the density functions defined on irregular, nonconvex spatial domains. The right panel displays a single realisation of the corresponding point patterns, each generated with 500 independent points according to the respective density.



(a) True density (b) ISEs relative to the true density



(c) KDE (d) DE-PDE



(e) DF (f) VG-KS

Fig. 13. Results of the simulation experiment for Case 1. The true density function is shown in panel (a). Panel (b) displays a boxplot of the ISEs of each method, relative to the true density. Panels (c)–(f) show the mean density estimates obtained by the competing methods across the 300 point pattern realisations.

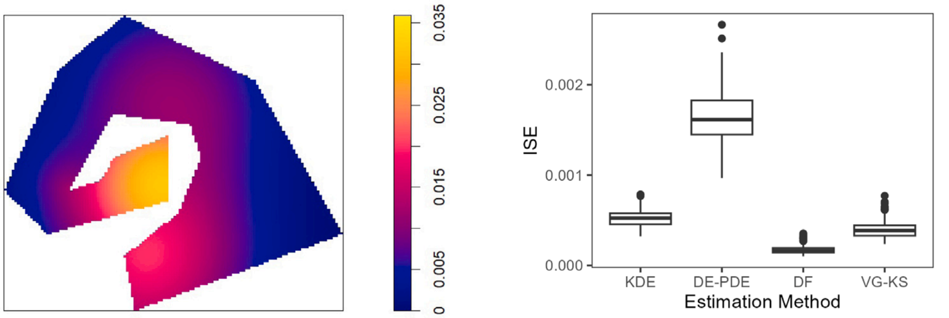
achieving a time complexity of $O(|E|\log^2 n)$, where $|E|$ is the number of edges in the visibility graph (see Overmars and Welzl (1988), Ghosh and Mount (1991), Ghosh (2007), and Hershberger (1989)).

5. Application

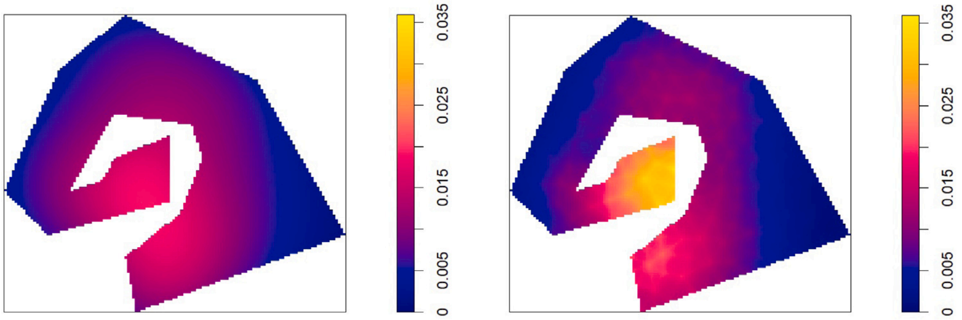
We now consider an application of our proposed methodology to crime incident reports in Boston, in the state of Massachusetts, United States of America (USA). The data is provided online by the Boston Police Department (BPD)[6] and contains records from a crime incident report system which includes the type of incident being reported as well as the location where it occurred. The records date back to June of 2015 and are regularly updated. For the purposes of this section, we use data records from 2023 to the present. In particular, consideration is given to crime incident reports that are classified as vandalism in the South Boston and South Boston Waterfront. A Google Earth plot of the these regions along with the locations of vandalism crime incident reports are

---

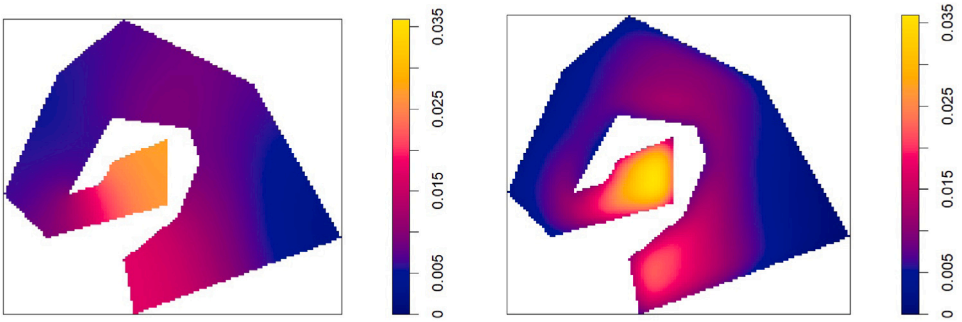
[6] https://data.boston.gov/



(a) True density (b) ISEs relative to the true density



(c) KDE (d) DE-PDE

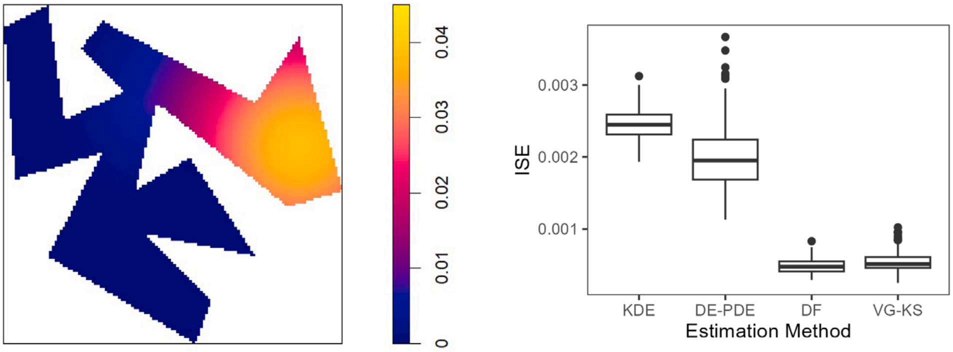


(e) DF (f) VG-KS

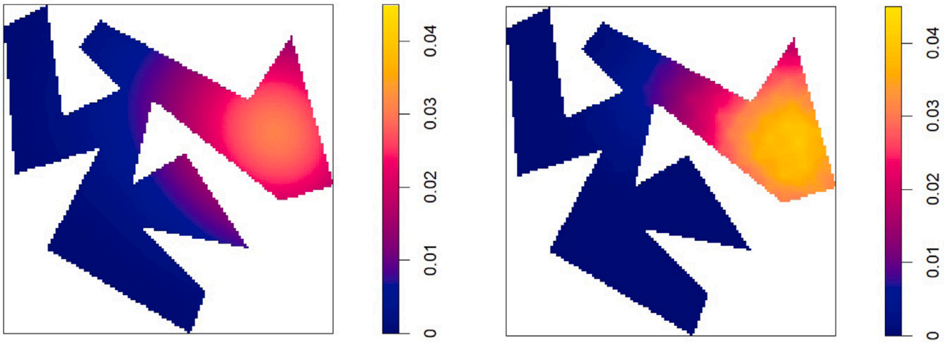
Fig. 14. Results of the simulation experiment for Case 2. The true density function is shown in panel (a). Panel (b) displays a boxplot of the ISEs of each method, relative to the true density. Panels (c)–(f) show the mean density estimates obtained by the competing methods across the 300 point pattern realisations.

depicted in Fig. 17(a). In the figure, it can be observed that there is a body of water that creates discontinuities in the domains of the subregions of Boston. Owing to this, the pathways for typical travel and movement are constrained to connected land masses.

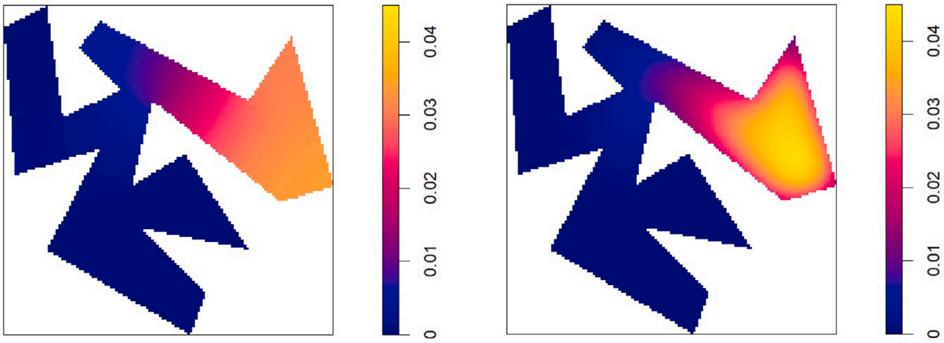
In applying our methodology to this real-world example, we simplify the polygon to reduce the number of extraneous vertices on the boundary, whilst preserving the shape of the window. The point pattern of the reported crime incidents within the simplified window is shown in Fig. 17(b). A triangular mesh is determined on the simplified window and illustrated in Fig. 17(b). To determine the optimal number of dimensions for the embedding, we consider the scree plot of stress against the number of dimensions, as depicted in Fig. 17(c). The lowest dimension at which increasing the number of dimensions yields diminishing returns in reducing stress is observed at three; therefore, we select this as the number of dimensions. Fig. 17(e) shows the Shepard diagram that compares the weighted visibility graph distances with the corresponding distances produced by applying MDS using Sammon's error. The points lie relatively close to the identity line (i.e., the line along which the embedded distances are exactly equal to the original distances), which suggests that the embedded distances correspond reasonably well to the original ones. The resultant VG-KS estimate is shown in Fig. 17(f). The density plot reveals distinct clusters of crime incident reports. These high-density regions indicate that local spatial proximities captured by the visibility graph and preserved in the embedding correspond to meaningful concentration



(a) True density (b) ISEs relative to the true density



(c) KDE (d) DE-PDE



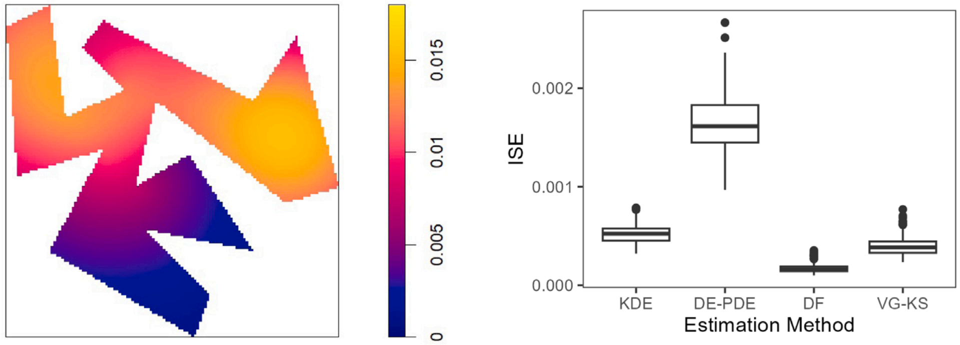
(e) DF (f) VG-KS

Fig. 15. Results of the simulation experiment for Case 3. The true density function is shown in panel (a). Panel (b) displays a boxplot of the ISEs of each method, relative to the true density. Panels (c)–(f) show the mean density estimates obtained by the competing methods across the 300 point pattern realisations.

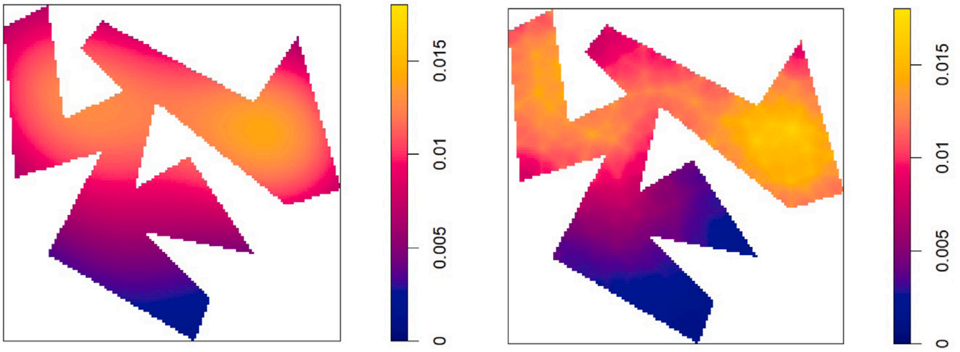
areas in the data, suggesting that the method captures aspects of the underlying spatial structure and provides a reasonable estimate of how the distribution of crime locations may be influenced by the area's layout and movement constraints.

6. Discussion and conclusion

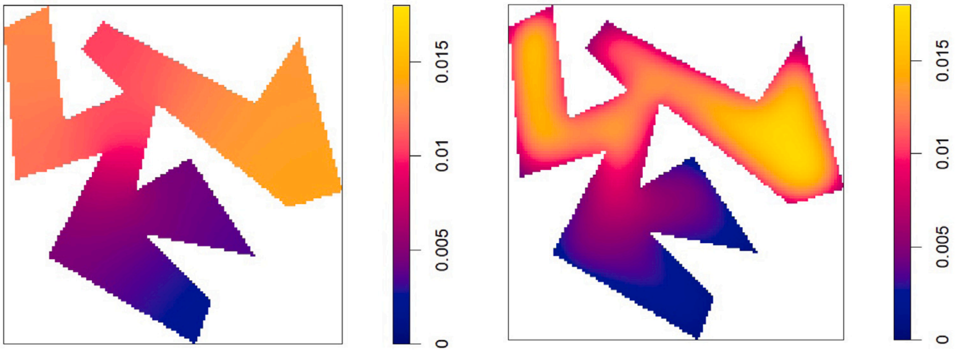
In spatial point pattern analysis, first- and second-order properties are typically defined as functions of interpoint distances. In this research, we address the need to adjust point pattern summary measures for appropriate use in nonconvex spatial domains. The choice of metric used to quantify proximity between points is important, as it directly influences estimation and prediction. Traditional approaches often make use of the Euclidean distance which assumes an unobstructed linear path between points. However, this assumption fails in real-world scenarios where the spatial domain includes barriers, gaps, or other forms of nonconvexity that constrain movement and interaction. In practical applications, point distributions are frequently shaped by the underlying geometric structure of the domain, resulting in more intricate spatial structures.



(a) True density (b) ISEs relative to the true density



(c) KDE (d) DE-PDE



(e) DF (f) VG-KS

Fig. 16. Results of the simulation experiment for Case 4. The true density function is shown in panel (a). Panel (b) displays a boxplot of the ISEs of each method, relative to the true density. Panels (c)–(f) show the mean density estimates obtained by the competing methods across the 300 point pattern realisations.

We have proposed a framework to account for the nonconvex structure of spatial domains using the notion of visibility and a path connected space determined by using the edge weights of the shortest path on a weighted visibility graph. Multidimensional scaling is used to allow the use of traditional statistical methodology that is based on the Euclidean distance. This alternative approach allows for a reduction in computational time that would be required for multiple queries to a visibility graph by preserving local visibility through the use of points in a triangular mesh. We used these ideas to analyse point patterns on nonconvex domains and illustrated their usefulness by using the kernel-smoothed estimator and K-function as examples.

Our findings show that when points are confined to nonconvex spatial domains, kernel-smoothed intensity estimates that apply weights as a function of Euclidean distance do not account for boundaries and gaps within the window. We addressed this issue by using the shortest path distance on the visibility graph, which provides a more appropriate measure of proximity within a connected space on a nonconvex window. In the case of the K-function, we have also obtained good preliminary empirical results.

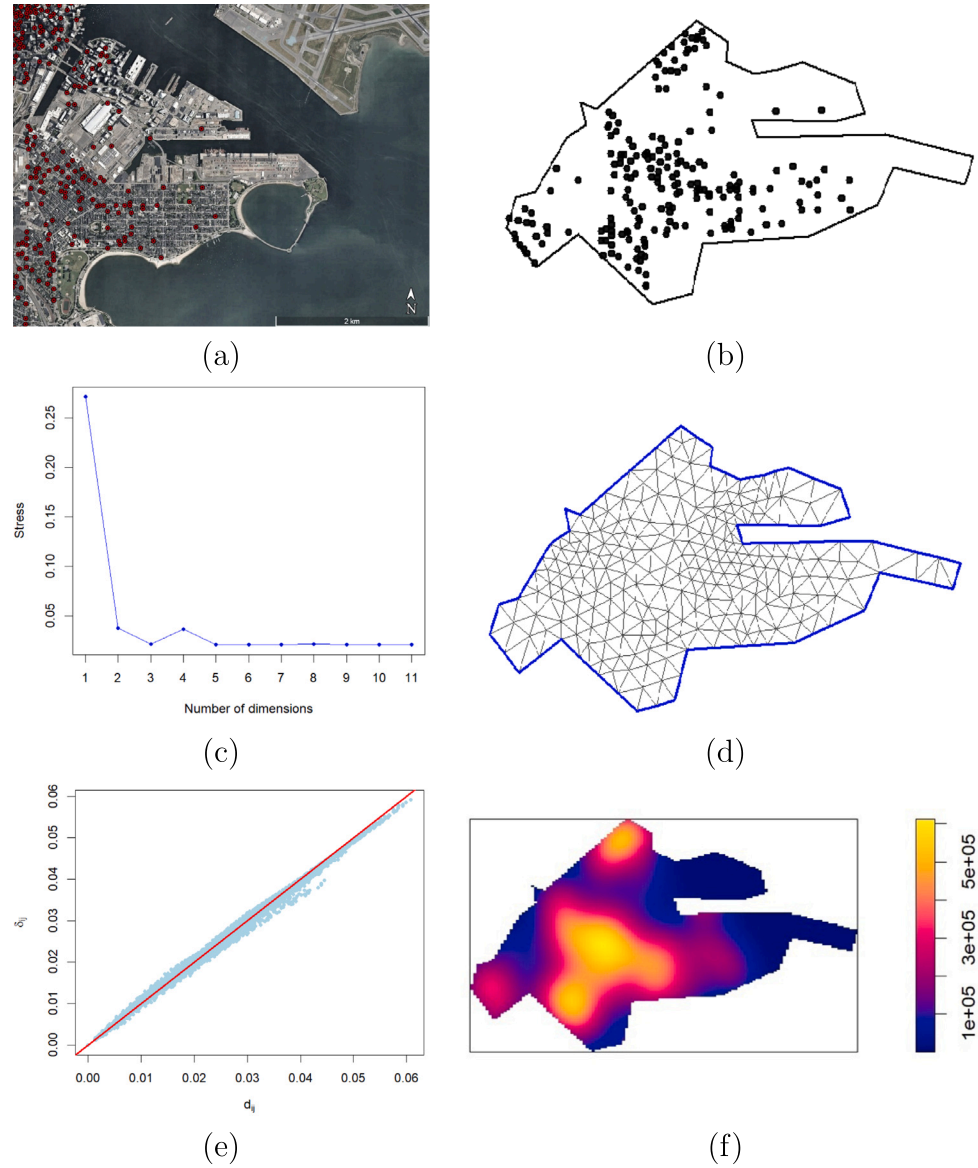


Fig. 17. Real world application of VG-KS. A Google Earth of South Boston and South Boston Waterfront is shown in (a). A point pattern observed within the boundaries of South Boston and South Boston Waterfront is depicted in (b). (c) shows the stress determined for metric MDS in different dimensions. The mesh used for MDS is shown in (d). In (e), a plot of d_{ij} against δ_{ij} is provided for the 3D case of MDS, and (e) shows the intensity estimate derived using VG-KS.

A drawback of the proposed method is its ability to handle bias near the boundary relative to other frameworks, i.e., the diffusion smoother of Baddeley et al. (2022). Future research will extend the current methodology to allow for a Euclidean characterisation that appropriately represents polygons with holes. In the absence of such methodology, the visibility graph can directly be used to compute distances.

Acknowledgments

This research received support from the National Research Foundation of South Africa (Grant Number 137785), and the South Africa National Research Foundation and South Africa Medical Research Council (South Africa DST-NRF-SAMRC SARChI Research Chair in Biostatistics, Grant number 114613). The content and opinions expressed herein are the sole responsibility of the authors and do not necessarily represent the official views of the NRF.

References

Bachl, F.E., Lindgren, F., Borchers, D.L., Illian, J.B., 2019. Inlabru: an R package for Bayesian spatial modelling from ecological survey data. Methods Ecol. Evol. 10 (6), 760–766.

Baddeley, A., Davies, T.M., Rakshit, S., Nair, G., McSwiggan, G., 2022. Diffusion smoothing for spatial point patterns. Statist. Sci. 37 (1), 123–142.

Baddeley, A., Nair, G., Rakshit, S., McSwiggan, G., Davies, T.M., 2021. Analysing point patterns on networks—A review. Spat. Stat. 42, 100435.

Baddeley, A., Rubak, E., Turner, R., 2015. Spatial Point Patterns: Methodology and Applications with R. CRC Press.

Bakka, H., Vanhatalo, J., Illian, J.B., Simpson, D., Rue, H., 2019. Non-stationary Gaussian models with physical barriers. Spat. Stat. 29, 268–288.

Borg, I., Groenen, P.J., 2007. Modern Multidimensional Scaling: Theory and Applications. Springer Science & Business Media.

Cressie, N., 1992. Statistics for spatial data. Terra Nova 4 (5), 613–617.

Cressie, N., 1993. Statistics for Spatial Data. John Wiley & Sons.

Csardi, G., Nepusz, T., 2006. The igraph software. Complex Syst. 1695, 1–9.

Diggle, P., 1985. A kernel method for smoothing point process data. J. R. Stat. Soc. Ser. C. Appl. Stat. 34 (2), 138–147.

Eubank, S., Guclu, H., Anil Kumar, V., Marathe, M.V., Srinivasan, A., Toroczkai, Z., Wang, N., 2004. Modelling disease outbreaks in realistic urban social networks. Nature 429 (6988), 180–184.

Ferraccioli, F., Arnone, E., Finos, L., Ramsay, J.O., Sangalli, L.M., 2021. Nonparametric density estimation over complicated domains. J. Roy. Stat. Soc. Ser. B: Stat. Methodol. 83 (2), 346–368.

Ghosh, S.K., 2007. Visibility Algorithms in the Plane. Cambridge University Press.

Ghosh, S.K., Mount, D.M., 1991. An output-sensitive algorithm for computing visibility graphs. SIAM J. Comput. 20 (5), 888–910.

Hershberger, J., 1989. An optimal visibility graph algorithm for triangulated simple polygons. Algorithmica 4 (1), 141–155.

Ibe, O., 2013. Markov Processes for Stochastic Modeling. Newnes.

Illian, J., Penttinen, A., Stoyan, H., Stoyan, D., 2008. Statistical Analysis and Modelling of Spatial Point Patterns, vol. 70, John Wiley & Sons.

Li, F., Klette, R., 2011. Euclidean shortest paths. In: Euclidean Shortest Paths. Springer, pp. 3–29.

Longley, P., Batty, M., 2003. Advanced Spatial Analysis: the CASA Book of GIS. ESRI, Inc..

Mahloromela, K., Fabris-Rotelli, I.N., Kraamwinkel, C., 2023. Covariate construction of nonconvex windows for spatial point patterns. South African Statist. J. 57 (2), 65–87.

Mardia, K.V., 1978. Some properties of classical multi-dimesional scaling. Commun. Stat. - Theory Methods 7 (13), 1233–1241.

Moller, J., Waagepetersen, R.P., 2003. Statistical Inference and Simulation for Spatial Point Processes. Chapman and Hall/CRC.

Okabe, A., Sugihara, K., 2012. Spatial Analysis Along Networks: Statistical and Computational Methods. John Wiley & Sons.

Okabe, A., Yamada, I., 2001. The K-function method on a network and its computational implementation. Geogr. Anal. 33 (3), 271–290.

Overmars, M.H., Welzl, E., 1988. New methods for computing visibility graphs. In: Proceedings of the Fourth Annual Symposium on Computational Geometry. pp. 164–171.

Peck, S.L., 1999. Quantitative analysis of movement: Measuring and modeling population redistribution in animals and plants. Ecology 80 (4), 1451–1453.

Ripley, B.D., 1976. The second-order analysis of stationary point processes. J. Appl. Probab. 13 (2), 255–266.

Sammon, J.W., 1969. A nonlinear mapping for data structure analysis. IEEE Trans. Comput. 100 (5), 401–409.

Shah, B., Gupta, S., 2016. Speeding up A* search on visibility graphs defined over quadtrees to enable long distance path planning for unmanned surface vehicles. In: Proceedings of the International Conference on Automated Planning and Scheduling, vol. 26, pp. 527–535.

Torgerson, W.S., 1952. Multidimensional scaling: I. Theory and method. Psychometrika 17 (4), 401–419.

Venables, W.N., Ripley, B.D., 2002. Modern Applied Statistics with S, fourth ed. Springer, New York, ISBN: 0-387-95457-0, URL https://www.stats.ox.ac.uk/pub/MASS4/.

Wu, G., Atilla, I., Tahsin, T., Terziev, M., Wang, L., 2021. Long-voyage route planning method based on multi-scale visibility graph for autonomous ships. Ocean. Eng. 219, 108242.

Yamada, I., Thill, J.-C., 2004. Comparison of planar and network K-functions in traffic accident analysis. J. Transp. Geogr. 12 (2), 149–158.