# SUPERVISED NONPARAMETRIC CLASSIFICATION IN THE CONTEXT OF REPLICATED POINT PATTERNS

KATEŘINA PAWLASOVÁ AND JIŘÍ DVOŘÁK✉

Department of Probability and Mathematical Statistics, Faculty of Mathematics and Physics, Charles University, Sokolovská 83, 186 75 Praha 8, Czech Republic
e-mail: dvorak@karlin.mff.cuni.cz, pawlasova@karlin.mff.cuni.cz

*(Received November 29, 2021; revised May 30, 2022; accepted June 20, 2022)*

## ABSTRACT

A spatial point pattern is a collection of points in space, representing, e.g. observed locations of trees, bird nests, centers of cells in a histological sample, etc. When several independent realizations of the underlying stochastic process are observed, these realizations are referred to as replicated point patterns. The main objective of this paper is to classify a newly observed pattern into one of the existing classes using a supervised nonparametric classification method, namely the Bayes classifier in combination with the $k$-nearest neighbors algorithms and the kernel regression method. The dissimilarity between a pair of patterns is defined using the functional summaries extracted from the point patterns via the Cramér-von Mises or Kolmogorov-Smirnov type formula. A set of simulation experiments is presented to investigate the performance of the proposed classifier with a dissimilarity measure based on functional summaries, such as the pair correlation function. The application of such a classifier to a real point pattern dataset is also illustrated.

Keywords: dissimilarity measures, kernel regression, spatial point patterns, supervised classification.

## INTRODUCTION

Spatial point processes are mathematical models that describe the arrangement of objects randomly placed in space. Such models are of particular interest in many scientific disciplines, including biology, ecology, statistical physics, or material science (Illian *et al.*, 2004, Sect. 1). We distinguish between the theoretical model, called point process, and its realization, a deterministic configuration of points, called point pattern. In practice, point patterns are observed in a bounded observation window. Three different point patterns can be seen in Fig. 1. Individual points represent locations of the centres of intramembranous particles of the mitochondrial membranes of the human HeLa cell line. These patterns were observed during the analysis of the HeLa cell line via the freeze-fracture technique (Schladitz *et al.*, 2003). It has become a standard approach to use functional summary statistics instead of univariate ones in all steps of statistical analysis of point patterns, from exploratory analysis through model fitting to hypothesis testing.

Supervised classification is one of the fundamental problems in statistics and machine learning. Early work on classification and statistical learning in general dates back to Fisher and the linear discriminant rule (Fisher, 1936; 1938). A collection of labelled observations, called a training set or training data, is available in supervised classification. The label indicates the affiliation of the given observation to one of the G possible classes. Based on the training data, the task is to assign a label to a new observation.

In the point pattern literature, the term *classification* usually refers to the procedure of labelling individual points within a single pattern generated by a superposition of several point processes (Dasgupta and Raftery, 1998; Redenbach *et al.*, 2015; Walsh and Raftery, 2005). This corresponds to the typical setting of spatial statistics, where a single point pattern, obtained by some physical measurement, is analyzed. This paper focuses on a different context: replicated point patterns. This means that the observed dataset consists of a collection of point patterns that need to be analyzed simultaneously rather than individually.

For replicated point patterns, supervised classification has been studied to a limited extent. In (Cholaquidis *et al.*, 2017), the patterns generated by inhomogeneous Poisson point processes with different intensity functions were classified. The task of classifying replicated point patterns is transformed in (Mateu *et al.*, 2015), with the help of multidimensional scaling, to the classification task in $\mathbb{R}^2$ and then solved with the help of Fisher's linear discriminant analysis. Parametric supervised classification is reviewed in (Vo *et al.*, 2018); this approach is also called model-based learning. Unsupervised classification is explored in (Ayala *et al.*, 2006). Note that we focus here only on spatial

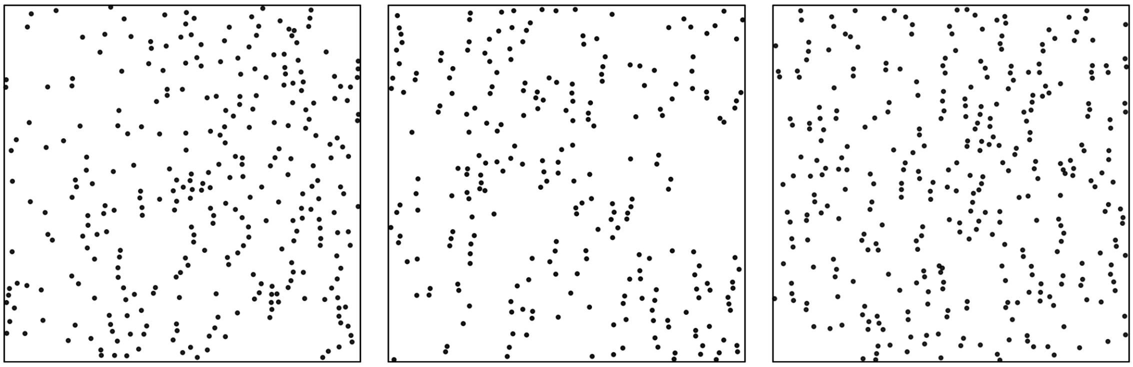

Fig. 1. *Point patterns represent centres of intramembraneous particles of mitochondrial membranes from HeLa cells under three different conditions: exposition to sodium acid (left), normal conditions (middle), and exposition to rotenone (right). The observation window is the square with a side length of* 336 *nm.*

point patterns, and thus point patterns on the real line are disregarded. For a discussion about the one-dimensional setting, see e.g. (Tranbarger Freier and Schoenberg, 2010; Victor and Purpura, 1997).

In this paper, we propose a general classification method that can be used for both stationary and nonstationary point processes in $\mathbb{R}^p$ and can be further generalized to more complicated settings (point patterns in non-Euclidean spaces, random sets, random tessellations, etc.). For ease of exposition, we present the methods only for stationary point processes in $\mathbb{R}^2$. We use the Bayes classifier in combination with the $k$-nearest neighbors algorithms and the kernel regression method. We need a mapping measuring dissimilarities between two point patterns to construct such classifiers. A summary of such mappings is given in (Mateu *et al.*, 2015; Alba-Fernández *et al.*, 2016).

We pay special attention to dissimilarity measures based on functional summaries, e.g. the pair correlation function extracted from the point patterns. So instead of comparing the patterns directly, we compare the extracted features in the form of functional data, and we can use well-established methods from functional data analysis, described, e.g. in (Ferraty and Vieu, 2006).

The complexity of the distribution of the considered point process models and their corresponding functional summary characteristics makes it highly challenging, if not impossible, to study the properties of the classifiers analytically. Thus, the behavior of the kernel regression classifier is explored through a simulation study. It is oriented towards the situation where the groups represent different parametric families of point process models (and hence different nature of spatial interactions) or the same parametric family with varying values of the model parameter.

As an example of real data, we analyze a collection of point patterns representing the intramembranous particles of the mitochondrial membranes of the human HeLa cell line. Three different classes are considered: the cell line exposed to sodium acid, the cell line under normal conditions, and the cell line exposed to rotenone. We aim to predict the class membership for a new observation based on the training set of labelled patterns. Examples of one pattern from each class can be seen in Fig. 1.

This paper is organized as follows. We start with a brief description of the three point process models that will be used in the simulation experiments. Next, we present several choices for the dissimilarity measure. The main contribution of the paper lies in the analysis of the performance of the proposed classifier on simulated point pattern data. Before describing the experiments themselves, we give the background on the Bayes classifier and the kernel regression method. In addition, we discuss some of the computational aspects of our simulations. The simulation experiments are complemented with an illustrative application to the HeLa cell line data. Finally, we close the paper with some concluding remarks.

## SPATIAL POINT PROCESSES

This section briefly describes the point process models used in the sequel. Related necessary definitions are given. For the foundations of the point process theory, see, e.g. (Daley and Vere-Jones, 2008). A comprehensive discussion about summary characteristics and feature extraction for point processes can be found in (Møller and Waagepetersen, 2004).

Throughout this paper, we restrict ourselves to point processes (random locally finite sets) in the plane. However, all the definitions and statements below can be easily reformulated for a general dimension $p$. Point process $X$ is said to be *stationary* if its distribution is invariant under translations. Moreover, $X$ is said to be *isotropic* if its distribution is invariant under rotations around the origin. We suppose that the intensity function of $X$ exists, that is, the expected number of points of $X$ in a Borel set $B \subseteq \mathbb{R}^2$ can be written as $\int_B \lambda(y)\ \mathrm{d}y$, where $\lambda$ (*the intensity function*) is nonnegative and measurable. If $X$ is stationary, then $\lambda(y) = \lambda > 0$ for all $y$, and the constant $\lambda$ is called *intensity*. The *pair correlation function* $g$ is defined by

$$g(x,y) = \frac{\lambda^{(2)}(x,y)}{\lambda(x)\lambda(y)},\ \ x,y \in \mathbb{R}^2,\ \lambda(x),\lambda(y) > 0.$$

In case $\lambda(x) = 0$ or $\lambda(y) = 0$, we set $g(x,y) = 0$. The function $\lambda^{(2)}(\cdot,\cdot)$ is the Radon-Nikodym derivative (with respect to the four-dimensional Lebesgue measure) of the second-order factorial moment measure (Møller and Waagepetersen, 2004). Loosely speaking, $\lambda^{(2)}(x,y)$ indicates how likely it is to observe two points together that occur in infinitely small neighbourhoods of $x$ and $y$, respectively. With slight abuse of the notation, we write $g(x,y) = g(x-y)$ whenever $g$ is translation invariant. If $g$ is also invariant under rotations around the origin, we write $g(x,y) = g(\|x-y\|)$, $x,y \in \mathbb{R}^2$. From now on, we suppose that $g$ is defined for $X$, and is, moreover, motion invariant. These assumptions imply that $g$ is a function of one argument $r$ that represents the Euclidean distance between two points in the process.

The *Poisson point process* is a benchmark point process model. It is used to model situations with no spatial interactions among the points. Since $\lambda^{(2)}(x,y) = \lambda(x)\lambda(y)$, $x,y \in \mathbb{R}^2$ (Møller and Waagepetersen, 2004, Sect. 4.1), the pair correlation function $g \equiv 1$ is a constant function. The value of $g$ for the Poison point process can be used as a benchmark in the following way. Let us have the formula for $g$ derived for another point process model. Then, values above 1 indicate aggregation of points, and values below 1 indicate repulsive interactions. In the sequel, the term Poisson point process always stands for a stationary Poisson point process with constant intensity $\lambda > 0$. The process will be denoted by $\Pi(\lambda)$.

The *Thomas process* is one of the basic models for aggregation of points. It was introduced in (Thomas, 1949) in the context of ecological surveys. A triplet of parameters (if we impose the stationarity) characterizes the model. First, we need to specify the intensity $\kappa$ of the underlying stationary Poisson process that models (unobserved) parental points. Then, we need to set the mean number $\mu$ of offspring points per parent. These points are the observed ones. Finally, the scale parameter $\sigma$ of the bivariate Gaussian density that controls the spatial distribution of the offspring points around a parent must be specified. The resulting process is stationary, with an intensity equal to the product $\kappa\mu$. For details, see (Møller and Waagepetersen, 2004, Sect. 5.3). The analytical formula for $g$ is known:

$$g(r) = 1 + \frac{1}{4\pi\sigma^2\kappa}\exp\left\{-\frac{r^2}{4\sigma^2}\right\},\ \ r > 0. \tag{1}$$

From equation (1), we see that for all $r$, $g(r) > 1$. In the sequel, the process will be denoted by $\Phi(\kappa,\mu,\sigma)$.

The *Gaussian determinantal point process* is a member of the family of the determinantal point processes (DPPs), which have been studied in mathematical physics, combinatorics, and random matrix theory for several decades. The general notion was introduced in 1975 in (Macchi, 1975). A detailed overview of the theory of DPPs is given in (Lavancier *et al.*, 2015). A DPP models repulsive interactions. Roughly speaking, the process is defined by specifying the Radon-Nikodym derivatives for the factorial moment measures of all orders with the help of a Gaussian covariance function $C_0(u) = \theta\exp\left\{-\|u/\alpha\|^2\right\}$, $u \in \mathbb{R}^2$. Here, $\theta > 0$ and $0 < \alpha \leq \alpha_{max}$, where $\alpha_{max}$ is a known constant given by $\alpha_{max} = 1/\sqrt{\pi\theta}$. Again, the formula for $g$ is known:

$$g(r) = 1 - \exp\left\{-\frac{2r^2}{\alpha^2}\right\},\ \ r > 0. \tag{2}$$

Equation (2) shows that the pair correlation function is always below the benchmark value for the Poisson point process. In the sequel, the process will be denoted by $\Psi(\theta,\alpha)$. Sample realizations of the three models can be found in Fig. 2.

# DISSIMILARITY MEASURES

In this section, we are looking for a map d that takes two point patterns and returns a number quantifying how dissimilar the two point patterns are. It has to meet the following conditions:

(i) $\mathrm{d}(\mathcal{X},\mathcal{Z}) \geq 0$,
(ii) $\mathrm{d}(\mathcal{X},\mathcal{Z}) = \mathrm{d}(\mathcal{Z},\mathcal{X})$,
(iii) $\mathrm{d}(\mathcal{X},\mathcal{Z}) \leq \mathrm{d}(\mathcal{X},\mathcal{U}) + \mathrm{d}(\mathcal{U},\mathcal{Z})$,

where $\mathcal{U},\mathcal{X}$ and $\mathcal{Z}$ represent different point patterns. An overview of the dissimiliraty measures for point patterns is given in (Mateu *et al.*, 2015; Alba-Fernández *et al.*, 2016).

Starting with the dissimilarity measures that are based on pattern matching, the Hausdorff distance is

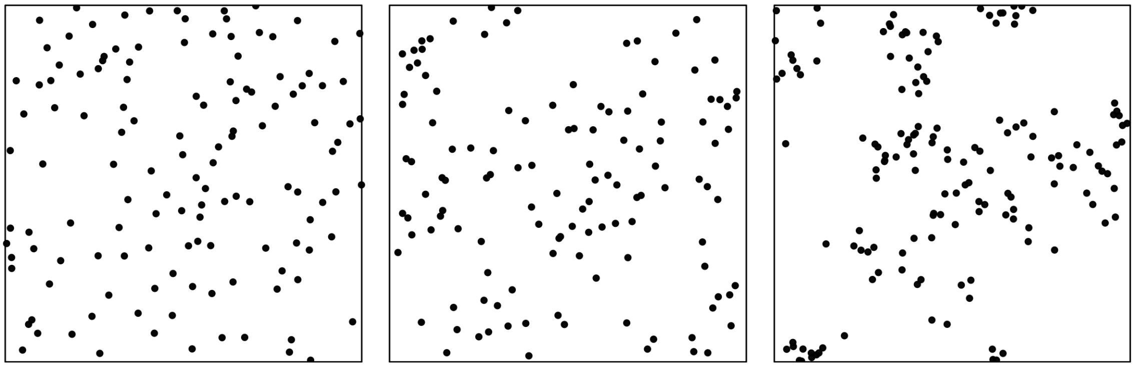

Fig. 2. *Realization of* $\Psi(\theta,\alpha)$, $\Pi(\lambda)$ *and* $\Phi(\kappa,\mu,\sigma)$ *respectively. The observation window W is the unit square* $[0,1]^2$, $\lambda = 120$, $\kappa = 20$, $\mu = 6$, $\sigma = 0.04$, $\theta = 120$ *and* $\alpha = 0.05$. *Parameters are chosen so that the three models have the same intensity.*

defined as

$$\begin{aligned} \mathrm{d}_H(\mathcal{X},\mathcal{Z}) &= \max\{\Delta(\mathcal{X},\mathcal{Z}),\Delta(\mathcal{Z},\mathcal{X})\}, \\ \Delta(\mathcal{X},\mathcal{Z}) &= \sup_{x\in\mathcal{X}}\inf_{z\in\mathcal{Z}} \|x-z\|, \\ \Delta(\mathcal{Z},\mathcal{X}) &= \sup_{z\in\mathcal{Z}}\inf_{x\in\mathcal{X}} \|z-x\|. \end{aligned}$$

In particular, it takes the maximum of the maximal Euclidean distance from a point in $\mathcal{X}$ to its nearest neighbour in $\mathcal{Z}$ and vice versa. In this case, $\mathrm{d}_H$ is a metric, so $\mathrm{d}_H(\mathcal{X},\mathcal{Z}) = 0$ if and only if the two point patterns coincide. However, its use is reasonable only if the two investigated patterns are observed in the same observation window. To achieve a low value, $\mathrm{d}_H$ forces the two configurations to have points at very similar locations, and hence high values can be seen even for two realisations coming from the same model. Modifications of $\mathrm{d}_H$ have been proposed in (Hoffman and Mahler, 2004; Schuhmacher *et al.*, 2008; Cholaquidis *et al.*, 2017), but none of them is the actual remedy for this problem. In special cases and assuming stationarity and isotropy, it may be relevant to consider $\mathrm{d}_H(\mathcal{X},\mathcal{Z}^\star)$ instead of $\mathrm{d}_H(\mathcal{X},\mathcal{Z})$, where $\mathcal{Z}^\star$ is the element of the set of all translations and rotations of $\mathcal{Z}$ such that $\mathrm{d}_H(\mathcal{X},\mathcal{Z}^\star)$ is minimal.

Another group of dissimilarity measures is based on feature matching. Important information (called a feature) about the distribution of the stochastic process that generated the pattern at hand is extracted from the pattern using a point process summary characteristic. In what follows, we focus on functional summaries such as the pair correlation function.

Let us now fix the functional characteristic $f$. For $r > 0$, let $\widehat{f}(\mathcal{X},r)$ be the value of $f(r)$ estimated from the point pattern $\mathcal{X}$. We define the integral dissimilarity measure for two point patterns $\mathcal{X}$, $\mathcal{Z}$ based on the functional characteristic $f$ as

$$\mathrm{d}_{int}(f,\mathcal{X},\mathcal{Z}) = \int_0^R |\widehat{f}(\mathcal{X},r) - \widehat{f}(\mathcal{Z},r)|^2 \,\mathrm{d}r, \qquad (3)$$

where $R$ is a given constant depending on the size and shape of the observation window $W$. If $W$ is the unit square, a popular rule of thumb leads to the choice $R = 0.25$. In real-life applications, the observation window $W$ can be rather complicated. Then, any general recommendation for the choice of $R$ would be counterproductive. Expert knowledge of the problem at hand should play the primary role in deciding which ranges of $r$ are relevant for distinguishing the groups of patterns.

The expression (3) resembles the Cramér-von Mises statistic from the goodness-of-fit tests in the classical statistics with i.i.d. observations. In the point process literature, similar expressions appear in the theory of parameter estimation (Møller and Waagepetersen, 2004, Sect. 10.1). Moreover, (3) is used to quantify the dissimilarities between point patterns in the stochastic reconstruction procedure for point patterns (Koňasová and Dvořák, 2021; Tscheschel and Stoyan, 2006). In (Mateu *et al.*, 2015), a dissimilarity matrix with entries computed as in (3) is plugged into a multidimensional scaling procedure, resulting in a representation of the collection of observed point patterns by a collection of points in $\mathbb{R}^2$.

The maximum absolute deviation counterpart of $\mathrm{d}_{int}$, resembling the Kolmogorov-Smirnov statistic, is then defined as

$$\mathrm{d}_{sup}(f,\mathcal{X},\mathcal{Z}) = \sup_{r\in[0,R]} |\widehat{f}(\mathcal{X},r) - \widehat{f}(\mathcal{Z},r)|.$$

Both $\mathrm{d}_{int}$ and $\mathrm{d}_{sup}$ are derived from semi-metrics commonly used in functional data analysis. They can be used even when the two investigated patterns are observed in different observation windows. The use of edge correction factors in the estimators of the functional summary characteristics reduces the impact of the size and shape of the observation window on the value of the estimate. However, the constant $R$ has to be chosen with care.

An ideal dissimilarity measure $\mathrm{d}(\mathcal{X},\mathcal{Z})$ would have small values whenever the probability distributions of the stochastic processes that generated $\mathcal{X}$ and $\mathcal{Z}$, respectively, are very similar. Our dissimilarity measures $\mathrm{d}_{int}$ and $\mathrm{d}_{sup}$ are based on matching the second-order properties of the point processes that generated $\mathcal{X}$ and $\mathcal{Z}$. One should have in mind that two point processes with different probability distributions can have the same form of the second-order characteristics; see e.g. Baddeley and Silverman (1984). In other words, two (visibly different) point patterns generated by models with different probability distributions can have zero dissimilarity $\mathrm{d}_{int}$ or $\mathrm{d}_{sup}$. This is a deeper issue than the fact that $\mathrm{d}_{int}$ and $\mathrm{d}_{sup}$ are not metrics. However, if such a situation would be encountered in a practical application, the user would choose a different summary characteristic instead, e.g. one based on interpoint distances, which would be able to distinguish the different groups in the training dataset.

For some applications, the need to use more than one characteristic to extract the essential features of the probability distribution may arise. Combining multiple integral or maximum absolute deviation terms into a weighted sum is possible. Choosing appropriate weights is a complex problem that usually deserves some preliminary exploratory analysis and expert knowledge. Another possibility is to use the dissimilarity measure described in (Dai *et al.*, 2021) which allows combining multiple characteristics without the need to weigh the individual terms.

# SUPERVISED CLASSIFICATION

This section gives an overview of the two classification methods used in the sequel. A list of related theoretical results available in the literature is included in Sect. S10 of the Supplementary material accompanying this paper.

Suppose that a point pattern $\mathcal{X}$ is observed in a bounded observation window $W$, $|W| > 0$. We assume that this pattern was generated by a stationary point process $X$, for which we can define the pair correlation function. Take $\overline{\mathrm{G}} \in \mathbb{N}$. Let $Y$ be the label, i.e., a random variable with values in $\overline{\mathrm{G}} = \{1,2,\ldots,\mathrm{G}\}$ representing the affiliation to one of the G possible groups. We consider $X$ and $Y$ as a random pair $(X,Y)$ and aim at predicting the value of the label variable $Y$, given the realization $\mathcal{X}$ of the point process $X$. Since we are talking about supervised classification, our decision about the value of $Y$ is based on the knowledge of training data, i.e., a collection of point patterns with known labels. In other words, let $\mathcal{T}_N = \{(\mathcal{X}_i,\mathcal{Y}_i),\ i=1,2,\ldots,N\}$ be a set of $N \geq 1$ independent realizations of $(X,Y)$. We call $\mathcal{T}_N$ the training set or the training data. From now on, we suppose that the dissimilarity measure d in hand is chosen so that ties do not occur. Suppose the expert knowledge about the analysed data indicates that the choice of d can lead to ties. In that case, one should consider changing the dissimilarity measure, e.g. using a different functional summary.

The classification task can also be viewed as the search for a classification rule $\varphi$, which assigns a label $\varphi(\mathcal{X})$ to a point pattern $\mathcal{X}$. If we know the conditional probabilities $p_{\mathrm{g}}(\mathcal{X}) = \mathbb{P}[Y=\mathrm{g} \mid X=\mathcal{X}],\ \mathrm{g} \in \overline{\mathrm{G}}$, we can construct the so-called naive Bayes classifier

$$\varphi_{Bayes}(\mathcal{X}) = \arg\max_{\mathrm{g}\in\overline{\mathrm{G}}} p_{\mathrm{g}}(\mathcal{X}).$$

However, $\{p_{\mathrm{g}},\ \mathrm{g}\in\overline{\mathrm{G}}\}$ are usually not known in practical applications. The crucial step when building up a classification rule is thus the estimation of these conditional probabilities, based on our knowledge of the training data $\mathcal{T}_N$. Let $\mathcal{X}$ be a new pattern whose label is to be predicted. In the following, we will restrict our attention to the estimators of $\{p_{\mathrm{g}},\ \mathrm{g}\in\overline{\mathrm{G}}\}$ in the form

$$\widehat{p}_{\mathrm{g}}(\mathcal{X}) = \sum_{\{i:\ \mathcal{Y}_i=\mathrm{g}\}} \omega_i(\mathcal{X})\ , \tag{4}$$

where $\{\omega_i(\mathcal{X}),\ i=1,2,\ldots,N\}$ are nonnegative weights, depending on the new observation $\mathcal{X}$ and the training set $\mathcal{T}_N$. The weights should be chosen in such a way that for all $\mathrm{g}\in\overline{\mathrm{G}}$ and $\mathcal{X}$

$$0 \leq \widehat{p}_{\mathrm{g}}(\mathcal{X}) \leq 1,\ \sum_{\mathrm{g}\in\overline{\mathrm{G}}} \widehat{p}_{\mathrm{g}}(\mathcal{X}) = 1. \tag{5}$$

In this paper, we focus on the weights derived from the *kernel regression method*. The method is presented in the context of functional data analysis in (Ferraty and Vieu, 2006, Sect. 8.2). Let $\mathrm{K}:\mathbb{R}\longrightarrow\mathbb{R}_+$ be a symmetric function such that $\int_{\mathbb{R}}\mathrm{K}(u)\ \mathrm{d}u = 1$. We will call such a function a kernel. Moreover, let the support of K be $[-1,1]$. Note that our assumptions imply that $\int_{\mathbb{R}} u\mathrm{K}(u)\ \mathrm{d}u = 0$. One of the classical examples of this function is the *Epanechnikov kernel* $\mathrm{K}(u) = (3/4)(1-u^2), u\in[-1,1]$ and $\mathrm{K}(u)=0$ otherwise. For $i\in\{1,2,\ldots,N\}$ we set

$$\omega_{i,h}(\mathcal{X}) = \frac{\mathrm{K}\left(h^{-1}\mathrm{d}(\mathcal{X},\mathcal{X}_i)\right)}{\sum_{j=1}^{N}\mathrm{K}\left(h^{-1}\mathrm{d}(\mathcal{X},\mathcal{X}_j)\right)}, \tag{6}$$

where $h>0$ is a smoothing parameter, also called bandwidth. Note that if $\mathrm{d}(\mathcal{X},\mathcal{X}_i) > h$, then the weight corresponding to the pattern $\mathcal{X}_i$ is 0. Equation (6) in fact corresponds to the Nadaraya-Watson estimate for a categorical response variable in the context of local polynomial regression. The classification rule based on the Nadaraya-Watson type of weights will be denoted by $\varphi_{NW}(\cdot \mid \mathcal{T}_N, \mathrm{d}, \mathrm{K}, h)$. The corresponding weights $\left\{\omega_{i,h}(\mathcal{X}),\ i = 1,2,\ldots,N\right\}$ sum up to 1 and the conditions in (5) are met.

Three different features have to be chosen by the user; the kernel K, the dissimilarity measure d, and the value of the smoothing parameter $h$. As is typical for kernel methods, the choice of the kernel function K is not crucial. Regarding the dissimilarity measure, we will restrict our attention to the three examples listed in the previous section. Note that the integral dissimilarity measure $\mathrm{d}_{int}$ corresponds to the framework of discrepancy measures for functional data presented in (Ferraty and Vieu, 2006, Sect. 3.4).

For the choice of bandwidth, one needs to deal with the problem of selecting $h$ among an infinite set of positive values. To overcome this issue, we replace $h$ with $h_k$ such that only $k$ terms in (4) contribute to the final sum with nonzero weights. In other words, we want to choose $h_k$ so that there are exactly $k$ elements $\mathcal{X}_{j_1}, \ldots \mathcal{X}_{j_k}$ in the training set $\mathcal{T}_N$ such that

$$\mathrm{d}\left(\mathcal{X},\mathcal{X}_{j_l}\right) h_k,\ l \in \{1,\ \ldots\ ,k\}, \qquad (7)$$
$$\mathrm{d}\left(\mathcal{X},\mathcal{X}_t\right) h_k,\ t \in \{1,\ \ldots ,N\} \setminus \{j_1,\ \ldots, j_k\}.$$

For the choice of $k$, three different approaches can be considered.

*Fixed value* The parameter $k$ is considered fixed and the decision about its value has to be based on our prior knowledge of the problem at hand. For each new observation $\mathcal{X}$, we order the dissimilarities $d_1 = \mathrm{d}(\mathcal{X},\mathcal{X}_1),\ldots,d_N = \mathrm{d}(\mathcal{X},\mathcal{X}_N)$ so that $d_{(1)} < d_{(2)} < \ldots < d_{(N)}$. We set $h_k = (d_{(k)} + d_{(k+1)})/2$. This choice of $h_k$ fulfils condition (8).

*Global cross-validation* This approach is based on the leave-one-out cross-validation procedure. For $i = 1,2,\ldots,N$, denote by $\mathcal{T}_N^{(-i)}$ a modified set of the training data $\mathcal{T}_N$, where we omit the $i$-th observation $(\mathcal{X}_i,\mathcal{Y}_i)$. The point pattern $\mathcal{X}_i$ is considered as a new observation, and we predict its label. We build $\left\{\widehat{p}^{(-i)}_{\mathrm{g},h^i_k}(\mathcal{X}_i),\ \mathrm{g} \in \overline{\mathrm{G}}\right\}$, using formulas (4), (6) and the new training data set $\mathcal{T}_N^{(-i)}$. The bandwidth $h^i_k$ is chosen so that $h^i_k = (d^i_{(k)} + d^i_{(k+1)})/2$, where $d^i_{(1)} < \ldots < d^i_{(N-1)}$ is the ordered collection of dissimilarities

$$\left\{\mathrm{d}\left(\mathcal{X}_i,\mathcal{X}_j\right),\ j \in \{1,\ \ldots\ ,N\} \setminus \{i\}\right\}.$$

Then, we estimate the label of $\mathcal{X}_i$ as

$$\varphi_{NW}(\mathcal{X}_i \mid \mathcal{T}_N^{(-i)}, \mathrm{d}, h^i_k) = \underset{\mathrm{g}\in\overline{\mathrm{G}}}{\arg\max}\ \widehat{p}^{(-i)}_{\mathrm{g},h^i_k}(\mathcal{X}_i).$$

We define the *global loss function GCV* : $\{1,2,\ldots,N-1\} \longrightarrow [0,1]$ by

$$GCV(k) = \frac{1}{N}\sum_{i=1}^{N} \mathbf{1}\left\{\mathcal{Y}_i \neq \varphi_{NW}(\mathcal{X}_i \mid \mathcal{T}_N^{(-i)}, \mathrm{d}, h^i_k)\right\}.$$

The parameter $k$ is chosen as a solution of an optimization problem

$$k_{GCV} = \underset{k\ \in\ \{1,2,\ldots,N-1\}}{\arg\min}\ GCV(k).$$

This choice of $k$ is called global since $k_{GCV}$ does not depend on the new observation $\mathcal{X}$. Finally, the new observation $\mathcal{X}$ is classified using the estimated conditional probabilities $\left\{\widehat{p}_{\mathrm{g},h_{k_{GCV}}}(\mathcal{X}),\ \mathrm{g} \in \overline{\mathrm{G}}\right\}$, where $h_{k_{GCV}} = (d_{(k_{GCV})} + d_{(k_{GCV}+1)})/2$. This procedure is described in (Ferraty and Vieu, 2006, Sect. 7.1.1).

*Local cross-validation* We now include the new observation $\mathcal{X}$ in the procedure of finding the optimal value of $k$. Let us define the *local loss function*

$$LCV : \{1,2,\ldots,N-1\} \times \{1,2,\ldots,N\} \longrightarrow \mathbb{R}_+$$

as

$$LCV(k,i) = \sum_{\mathrm{g}\in\overline{\mathrm{G}}} \left[\mathbf{1}\{\mathcal{Y}_i = \mathrm{g}\} - \widehat{p}^{(-i)}_{\mathrm{g},h^i_k}(\mathcal{X}_i)\right]^2.$$

For each observation $\mathcal{X}_i$ in the training set $\mathcal{T}_N$, the local optimal value $k_{LCV}(\mathcal{X}_i)$ is then found as a solution to the optimization problem

$$k_{LCV}(\mathcal{X}_i) = \underset{k\ \in\ \{1,2,\ldots,N-1\}}{\arg\min}\ LCV(k,i),\ i \in \{1,2,\ldots,N\}.$$

Let $i_0 = i_0(\mathcal{X})$ be the index of the nearest neighbour of $\mathcal{X}$ in the training set $\mathcal{T}_N$, that is,

$$i_0 = \underset{i\ =\ 1,2,\ldots,N}{\arg\min}\ \mathrm{d}(\mathcal{X},\mathcal{X}_i).$$

Then, we use the value $k_{LCV}(\mathcal{X}) = k_{LCV}(\mathcal{X}_{i_0})$ to build up the probabilities $\left\{\widehat{p}_{\mathrm{g},h_{k_{LCV}(\mathcal{X})}}(\mathcal{X}),\ \mathrm{g} \in \overline{\mathrm{G}}\right\}$. The local cross-validation procedure is described in (Ferraty and Vieu, 2006, Sect. 8.3).

Note that if we set K as the uniform kernel $\mathrm{K}(u) = 1/2,\ u \in [-1,1]$, instead of the Epanechnikov kernel, then the weights $\left\{\omega_{i,h_k}(\mathcal{X}),\ i = 1,2,\ldots,N\right\}$ are of the form

$$\omega_{i,h_k}(\mathcal{X}) = \omega_{i,k}(\mathcal{X}) = \begin{cases} \dfrac{1}{k}, & \text{if } \mathrm{d}(\mathcal{X},\mathcal{X}_i) \leq d_{(k)}, \\ 0, & \text{otherwise.} \end{cases} \qquad (8)$$

The classification rule based on this choice of weights is called the $k$-nearest neighbors classifier. It can be easily seen that the estimated probabilities $\widehat{p}_{g,k}(\mathcal{X})$, obtained by plugging these weights in (4), meet the conditions (5). The parameter $k$ can be fixed, or it can be learned from the training set $\mathcal{T}_N$ using global or local cross-validation. Together with the dissimilarity measure $\mathrm{d}_H$, the $k$-nearest neighbours classifier is used in (Cholaquidis *et al.*, 2017) to classify realizations of inhomogeneous Poisson point processes with different intensity functions.

# SIMULATION EXPERIMENTS

In the next sections, the performance of the Bayes classifier in combination with the $k$-nearest neighbors algorithms and the kernel regression method will be examined in the context of replicated point patterns. Three simulation experiments will be presented. First, we compare the performance of the proposed classifier with respect to the different choices of the underlying dissimilarity measure. Second, the proposed method is compared with the approach suggested in (Mateu *et al.*, 2015), which uses multidimensional scaling to transform the problem into a classification task in $2D$. Finally, we explore the situation where the groups correspond to the realizations from the models of the same parametric family but with different values of the model parameter. For ease of exposition, our attention is restricted solely to binary classification.

This section serves as a detailed description of the practical aspects of our simulation experiments, such as the methodology for assessing the quality of performance of individual classification rules or the exact setting of the computational environment.

*Classification rules* In the following experiments, we consider the classification rule to be the $\varphi_{NW}$. We fix K as the Epanechnikov kernel and use the automatic choice of the bandwidth; $h = h_{k_{LCV}}$ is found by the $k$-nearest neighbors algorithm with the local choice of $k$. We write

$$\varphi(\cdot \mid \mathcal{T},\mathrm{d}) = \varphi_{NW}\left(\cdot \mid \mathcal{T},\mathrm{d},\mathrm{K},h_{k_{LCV}}\right)$$

to highlight the impact of the training set $\mathcal{T}$ and the dissimilarity measure d. If, for example, we fix $\mathcal{T}$ and consider two different dissimilarity measures $\mathrm{d}_1$ and $\mathrm{d}_2$, then $\varphi(\cdot \mid \mathcal{T},\mathrm{d}_1)$ and $\varphi(\cdot \mid \mathcal{T},\mathrm{d}_2)$ are referred to as two different classification scenarios.

*Misclassification rate* The performance of a given classification rule $\varphi$ is determined using the misclassification rate. Take $\mathcal{T}_N$, $N \in \mathbb{N}$, the training set. To calculate the misclassification rate, another set of labelled patterns is needed. Denote the testing set, that is, a collection of patterns with known labels, by $\Gamma_M = \left\{(\widetilde{\mathcal{X}}_j,\widetilde{\mathcal{Y}}_j),\ j = 1,2,\ldots M\right\}, M \in \mathbb{N}$. Note that $\mathcal{T}_N$ and $\Gamma_M$ should include different observations. Given the training and the testing set, the misclassification rate $\gamma\big(\varphi \mid \mathcal{T}_N,\Gamma_M\big)$ is computed as

$$\gamma(\varphi \mid \mathcal{T}_N,\Gamma_M) = \frac{1}{M}\sum_{j=1}^{M}\mathbf{1}\left\{\varphi\big(\widetilde{\mathcal{X}}_j \mid \mathcal{T}_N\big) \neq \widetilde{\mathcal{Y}}_j\right\},$$

where $\varphi\big(\widetilde{\mathcal{X}}_j \mid \mathcal{T}_N\big)$ is the estimated value of the label based on the classification rule $\varphi$ and the training data $\mathcal{T}_N$. Point pattern $\widetilde{\mathcal{X}}_j$ is misclassified if its estimated label $\varphi\big(\widetilde{\mathcal{X}}_j \mid \mathcal{T}_N\big)$ does not correspond to its true label $\widetilde{\mathcal{Y}}_j$. The misclassification rate thus gives the ratio of the number of misclassified patterns from the testing set and the size of the testing set itself. For binary classification, $\gamma\big(\varphi \mid \mathcal{T}_N,\Gamma_M\big) = 0.5$ corresponds to the situation where the labels are assigned randomly, regardless of the value of the new observation.

*Quality of performance* Different classification scenarios are compared using the average misclassification rate computed from the set of $I \in \mathbb{N}$ replications of the specific simulation experiment. Scenario with a lower average misclassification rate is then referred to as preferable. For all the simulation experiments, we set $I$ at 100.

*Computational aspects* All simulation experiments are performed in the statistical software `R` (R Core Team , 2017), with packages `doParallel` (Wallig *et al.*, 2019), `pracma` (Borchers, 2019), and `spatstat` (Baddeley *et al.*, 2015). The code for the Hausdorff metric $\mathrm{d}_H$ is taken from the package `pracma`. The pair correlation function is computed with the default estimator in `spatstat`. Details about the estimation of functional summary characteristics can be found in (Møller and Waagepetersen, 2004). The unit square $[0,1]^2$ is taken as the observation window $W$. The constant $R$, appearing in the formulas for $\mathrm{d}_{int}$ and $\mathrm{d}_{sup}$, is set to 0.25. When implementing the kernel regression classifiers, we use the code accompanying the book (Ferraty and Vieu, 2006). We have made some minor modifications to adapt the original code to the context of replicated point patterns. The automatic procedure for the choice of the smoothing parameter $h$ is based on leave-one-out cross-validation. In the simulation experiments, we use a small number of patterns in the training set (which is often the case in practical applications). In this setting, the computation of the dissimilarities (namely, the estimation of the pair correlation function) is the computational bottleneck. In contrast, the classification, including cross-validation, runs rather quickly. However, in the case of a large training set, it can be beneficial to consider $j$-fold cross-validation to save some computational time. This approach requires the specification of the number of folds $j$.

Further simulation experiments (Sections S4, S5, S6) and the extension of the ones presented here (Sections S2, S3, S7, S8) can be found in the Supplementary material accompanying the paper. An illustration of the code for our experiments can be found in a repository on Github at `https://github.com/kpawlasova/Sup_nonparam_clas_pp.git`

## EXPERIMENT 1

This simulation experiment provides a basic overview of the performance of the proposed classifiers. The pair correlation function $g$ is considered here, given its simple interpretation and widespread use in practical applications.

*Models* We fix the intensity $\lambda = 120$ and denote $\Pi(\lambda)$ the stationary Poisson point process with intensity $\lambda$. The stationary Thomas process is denoted by $\Phi(\kappa,\mu,\sigma)$ and we set $\kappa = 20$ and $\mu = 6$. Parameter $\sigma$ takes values in $\Sigma = \{0.02, 0.03, \ldots, 0.20\}$. To stress the dependence on the varying value of $\sigma$ we write $\Phi(\sigma) = \Phi(20,6,\sigma)$, $\sigma \in \Sigma$. For the Gaussian DPP, we fix $\theta = 120$ and $\mathcal{A} = \{0.0025, 0.0050, \ldots, 0.0500, \alpha_{max}\}$ where $\alpha_{max} = 1/\sqrt{120\pi} \doteq 0.0515$. To emphasize the dependence on $\alpha$, we write $\Psi(\alpha) = \Psi(120,\alpha), \alpha \in \mathcal{A}$. All three models have the same intensity 120. The observation window is $W = [0,1]^2$ in all cases.

*Training and testing data* For each $\sigma \in \Sigma$, the training set $\mathcal{T}(\sigma)$ consists of 20 realizations of $\Phi(\sigma)$ and 20 realizations of $\Pi$. The testing set $\Gamma(\sigma)$ contains 100 realizations of $\Phi(\sigma)$ and 100 realizations of $\Pi$. The same applies for each $\alpha \in \mathcal{A}$ for the training set $\mathcal{T}(\alpha)$ and the testing set $\Gamma(\alpha)$, with $\Psi(\alpha)$ in place of $\Phi(\sigma)$.

*Dissimilarity measure* Three different dissimilarity measures are considered: the Hausdorff metric $\mathrm{d}_H$ and two dissimilarity measures based on the pair correlation function $g$: $\mathrm{d}_{int}[g]$ (shorter notation for $\mathrm{d}_{int}(g,\cdot,\cdot)$) and $\mathrm{d}_{sup}[g]$ (shorter notation for $\mathrm{d}_{sup}(g,\cdot,\cdot)$).

*Classification scenarios* For each $\sigma \in \Sigma$, three different classification scenarios are considered:

$$\begin{aligned}
\varphi_H[\sigma](\cdot) &= \varphi_{NW}\big(\cdot \mid \mathcal{T}(\sigma), \mathrm{d}_H, \mathrm{K}, h_{k_{LCV}}\big),\\
\varphi_{g,int}[\sigma](\cdot) &= \varphi_{NW}\big(\cdot \mid \mathcal{T}(\sigma), \mathrm{d}_{int}[g], \mathrm{K}, h_{k_{LCV}}\big),\\
\varphi_{g,sup}[\sigma](\cdot) &= \varphi_{NW}\big(\cdot \mid \mathcal{T}(\sigma), \mathrm{d}_{sup}[g], \mathrm{K}, h_{k_{LCV}}\big),
\end{aligned}$$

where the choice of K and $h_{k_{LCV}}$ was described in the previous section of this paper. Values of average misclassification rates are reported. For $\alpha \in \mathcal{A}$, the corresponding scenarios $\varphi_H[\alpha]$, $\varphi_{g,int}[\alpha]$ and $\varphi_{g,sup}[\alpha]$ are considered.

*Results for Thomas process* In the following comments, values of $\sigma$ in $[0.02, 0.1)$ are considered small and correspond to strong clustering, values in $[0.1, 0.15)$ are considered moderate, and values in $[0.15, 0.2]$ are considered large and correspond to weak clustering. The Poisson point process $\Pi$ can then be considered a limiting case of $\Phi(\sigma)$ as $\sigma$ goes to infinity. The categorization to strong, moderate, and weak clustering is related to the size and shape of the observation window. Recall that the theoretical formula for $g$ is given in (1). For an illustration of how the values of $g$ depend on the model parameter $\sigma$, see Fig. S2.1 in the Supplementary material.

For small values of $\sigma$, which indicates strong clustering, the average misclassification rate is expected to be close to 0. On the other hand, for large values of $\sigma$, the realizations from $\Phi(\sigma)$ are hardly distinguishable from those from $\Pi$ (given our observation window), and the average misclassification rate is expected to be close to 0.5.

The observed average misclassification rates are, in fact, increasing functions of $\sigma$, regardless of the dissimilarity measure; see Fig. 3 (top left). In terms of the average misclassification rate, both $\varphi_{g,int}$ and $\varphi_{g,sup}$ outperform $\varphi_H$, especially for $\sigma < 0.1$. The small difference between the performance of $\varphi_{g,int}$ and $\varphi_{g,sup}$ is caused by the high variability of the estimator of $g(r)$ for very small values of $r$, which influences the maximum absolute deviation counterpart of the dissimilarity measure more than the integral one.

The realizations of the Poisson point process $\Pi$ have a very similar structure, which leads to small variability in the values of $\mathrm{d}(\mathcal{X}_i, \mathcal{X}_j)$, where $\mathcal{X}_i$ and $\mathcal{X}_j$ are realizations of $\Pi$, regardless of the dissimilarity measure in use. On the other hand, for small values of $\sigma$, the realizations of the Thomas process $\Phi(\sigma)$ show higher variability in terms of point configurations (clusters of points are placed in arbitrary locations, leaving gaps between clusters) and hence higher variability in the dissimilarities measured between two elements of this group. The dissimilarities between elements of different groups are smaller for $\varphi_{g,int}[\sigma]$ and $\varphi_{g,sup}[\sigma]$ than for $\varphi_H[\sigma]$. See Fig. 4 for illustration. These observations correspond to the fact that most of the misclassified patterns in this experiment were realizations of $\Phi(\sigma)$ (Thomas process), erroneously labeled as realizations of $\Pi$ (Poisson process) (see Fig. S2.3 and Fig. S2.2 in the Supplementary material). The overall performance of $\varphi_{g,int}[\sigma]$ and $\varphi_{g,sup}[\sigma]$ is

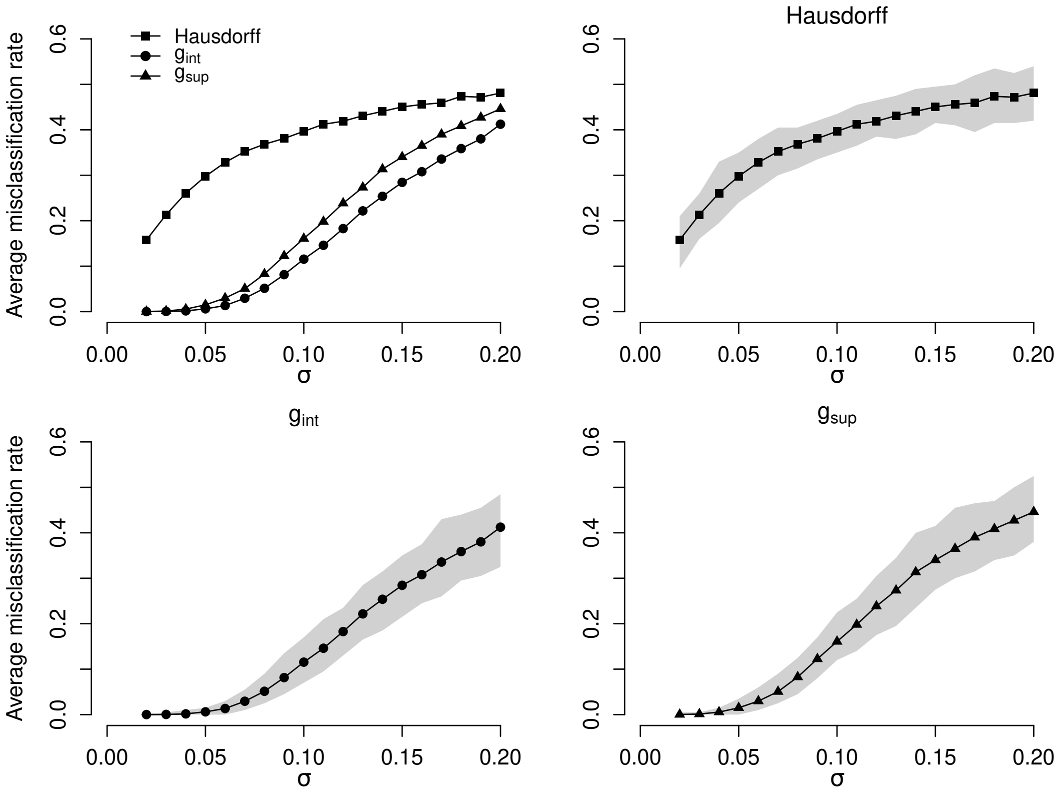

Fig. 3. *Average misclassification rates $\bar{\gamma}(\varphi_H[\sigma])$, $\bar{\gamma}(\varphi_{g,int}[\sigma])$ and $\bar{\gamma}(\varphi_{g,sup}[\sigma])$ are plotted as functions of parameter $\sigma$ (top-left). To illustrate the variability of the individual misclassification rates, the* 90% *pointwise envelopes are plotted for each classification scenario.*

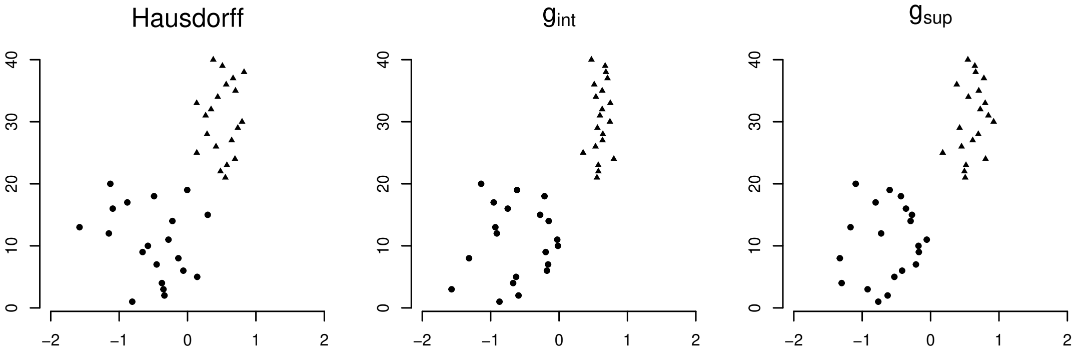

Fig. 4. *Visualization of dissimilarities in a set of 40 patterns (20 patterns generated from $\Phi(\sigma)$ with $\sigma = 0.05$, indices 1 to 20, denoted by circles, and 20 patterns from $\Pi$, indices 21 to 40, denoted by triangles). The plots correspond to $\mathrm{d}_H$ (left), $\mathrm{d}_{int}[g]$ (middle) and $\mathrm{d}_{sup}[g]$ (right). Vertical axis – index of the pattern. Horizontal axis – the position of the points of the plot on horizontal axis are determined by the multidimensional scaling so that the distance of a pair of points in the plot is approximately proportional to the dissimilarity between the underlying pair of point patterns.*

satisfactory, but the average misclassification rate of $\varphi_H[\sigma]$ is much higher for small values of $\sigma$.

For higher values of $\sigma$, the realizations of $\Phi(\sigma)$ resemble those of $\Pi$, and the dissimilarities between the realizations from different groups become smaller. This implies a higher number of misclassified patterns from both groups and higher average misclassification rates for all classifiers considered in this experiment.

Note that the pair correlation function itself is estimated with the help of a kernel-based estimator. Therefore, it requires the user to choose a smoothing parameter. The quality of the estimates is highly dependent on the choice of this smoothing parameter. Fig. 5 illustrates the impact of this choice on the performance of the classifier based on $\mathrm{d}_{int}[g]$ and $\mathrm{d}_{sup}[g]$, respectively. We study three choices of the smoothing parameter: the default value from the `spatstat` package, 0.5× the default, and 1.5× the default. Our simulations show that the appropriate choice of the smoothing parameter (in this case, higher than the default) can slightly improve the results. On the contrary, a wrong choice (too small value) can severely disrupt the classification. Hence, we recommend keeping the default value to reduce the risk of choosing a too small value.

*Results for Gaussian determinantal process* In this case, the values of $\alpha$ in $[0.0025, 0.002)$ are considered small and correspond to weak repulsion, the values in $[0.002, 0.004)$ are considered moderate and the values in $[0.04, \alpha_{max}]$ are considered large and correspond to strong repulsion. With the value of $\alpha$ approaching 0, the repulsive interactions become weaker, and $\Pi$ can be considered a limiting case of $\Psi(\alpha)$ for $\alpha \to 0$. Recall that the theoretical formula for $g$ is given in (2), illustration of the dependence of $g$ on $\alpha$ can be found in Fig. S2.4 in the Supplementary material. For small values of $\alpha$, the average misclassification rate is expected to be close to 0.5. Then, it is expected to decrease with $\alpha$ increasing.

Fig. 6 shows that $\bar{\gamma}(\varphi_H[\alpha])$ is between 0.4 and 0.5 for all $\alpha \in \mathcal{A}$, meaning that the realizations from $\Psi(\alpha)$ and $\Pi$ are practically indistinguishable using the Hausdorff metric, regardless of the value of $\alpha$. Scenarios $\varphi_{g,int}[\alpha]$ and $\varphi_{q,sup}[\alpha]$ produce similar average misclassification rates, which are very satisfactory for high values of $\alpha$.

The repulsive interactions in $\Psi(\alpha)$ imply that the realizations have a very similar structure, with smaller dissimilarities between two realizations from the same model than in the case of two realizations of $\Pi$. Similar considerations about the number of misclassified patterns in each group that we have made in the previous paragraphs also apply to the classification $\Psi(\alpha)$ vs $\Pi$. However, note that now $\Pi$ has more variable configurations and larger dissimilarities between two realizations from the same model, see Fig. 7. More details can be found in Fig. S2.5 and Fig. S2.6 in the Supplementary material.

*Summary* In both situations, $\Phi(\sigma)$ vs $\Pi$ and $\Psi(\alpha)$ vs $\Pi$, the classifiers $\varphi_{g,int}$ and $\varphi_{g,sup}$ outperform $\varphi_H$ for all values of the model parameter. If the model parameters are set such that the observation window $W$ does not provide enough information to distinguish between the realizations of the two models (high value of $\sigma$ or small value of $\alpha$), the average misclassification rate is close to 0.5 for all classifiers. However, even in these cases, it is beneficial to choose $\varphi_{g,int}$ or $\varphi_{g,sup}$ over $\varphi_H$. We conclude that the choice of the dissimilarity measure greatly affects the performance of the classifiers. Further computations (Sect. S3 in the Supplementary material) favour the use of the second-order summary characteristics such as $g$ or $L$ when constructing the dissimilarity measure.

Given this basic framework (binary classification, simple models), $\varphi_{NW}$ together with $\mathrm{d}_{int}[g]$ or $\mathrm{d}_{sup}[g]$ provides satisfactory results. However, the appropriate choice of the summary characteristic is not obvious. Prior expert knowledge of the problem at hand should always be taken into account.

## EXPERIMENT 2

This simulation experiment extends Experiment 1 from the previous section. It compares the performance of the classifiers studied in Experiment 1 with the classifier introduced in (Mateu *et al.*, 2015). Models, training and testing data are precisely the same as in Experiment 1.

*Classification scenarios* For each $\sigma \in \Sigma$, we consider the three classification scenarios $\varphi_H[\sigma]$, $\varphi_{g,int}[\sigma]$ and $\varphi_{g,sup}[\sigma]$ introduced in Experiment 1. The performance of these classifiers will be compared with $\varphi^{\star}_H[\sigma]$, $\varphi^{\star}_{g,int}[\sigma]$ and $\varphi^{\star}_{g,sup}[\sigma]$ introduced in (Mateu *et al.*, 2015). In detail, $\varphi^{\star}_H[\sigma]$ uses the dissimilarities based on the Hausdorff metric and multidimensional scaling (MDS) to represent the realizations of $\Pi$ and $\Phi(\sigma)$ as points in $\mathbb{R}^2$. Then, Fisher's linear discriminant analysis (LDA) is used to solve the substitutive classification task in $\mathbb{R}^2$. Analogously, $\varphi^{\star}_{g,int}[\sigma]$ and $\varphi^{\star}_{g,sup}[\sigma]$ use $\mathrm{d}_{int}[g]$ and $\mathrm{d}_{sup}[g]$, respectively, to calculate the dissimilarities that enter multidimensional scaling. Multidimensional scaling is performed using the function `smacofSym` from the `R` package `smacof`. The function `lda` from the `R` package `MASS` is then used to perform the classification. We report the values of the average misclassification rates, as in Experiment 1.

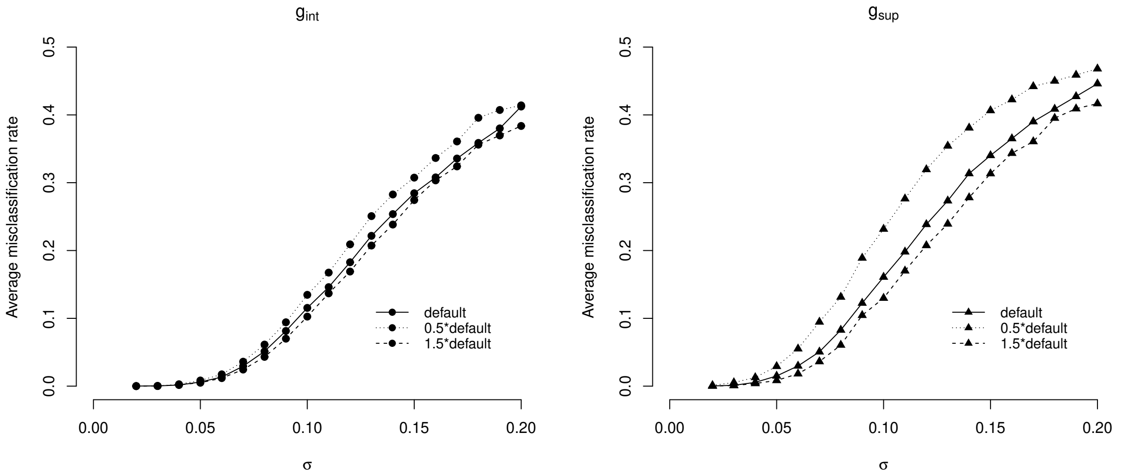

Fig. 5. *Average misclassification rates for different choices of the smoothing parameter used while estimating the pair correlation function. The solid line corresponds to the default setting in* `spatstat` *package (this values are the same as in Fig. 3), the dotted line corresponds to the value* $0.5\times$ *the default and the dashed line corresponds to the value* $1.5\times$ *the default.*

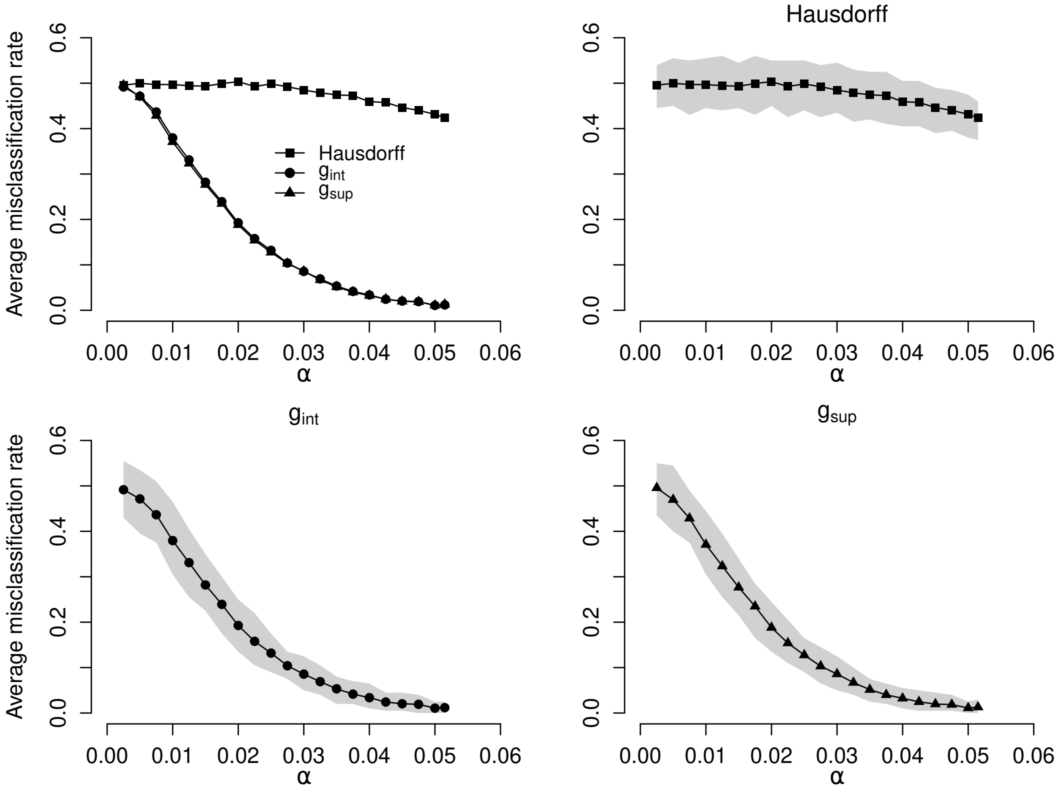

Fig. 6. *Average misclassification rates* $\bar{\gamma}(\varphi_H, \alpha)$, $\bar{\gamma}(\varphi_{g,int}, \alpha)$ *and* $\bar{\gamma}(\varphi_{g,sup}, \alpha)$ *are plotted as functions of parameter* $\alpha$ *(top left). To illustrate the variability of the individual misclassification rates, the* 90% *pointwise envelopes are plotted for each classification scenario.*

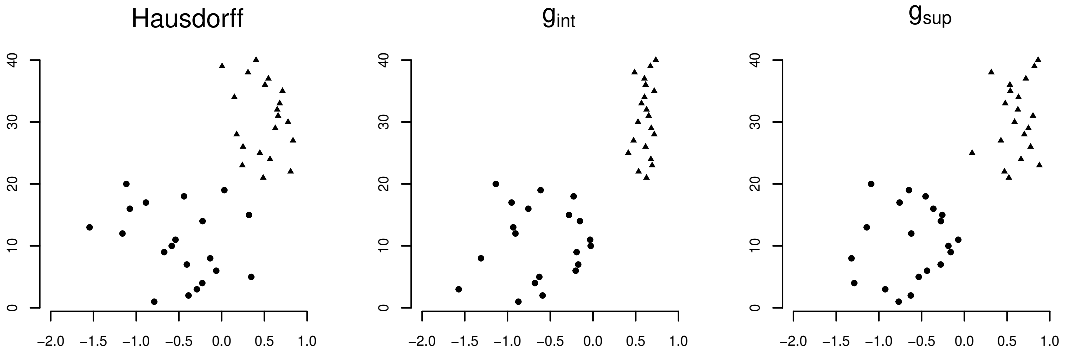

Fig. 7. *Same as Fig. 4, with circles now corresponding to realizations from* $\Pi$ *and triangles corresponding to realizations from* $\Psi(\alpha_{max})$.

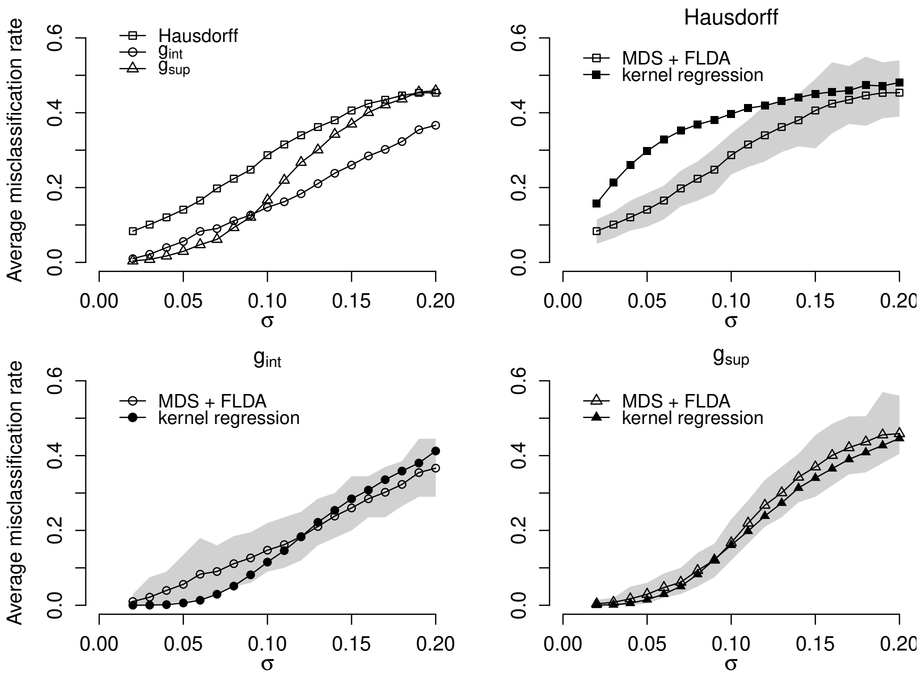

Fig. 8. *Average misclassification rates corresponding to* $\varphi_H^\star[\sigma]$, $\varphi_{g,int}^\star[\sigma]$ *and* $\varphi_{g,sup}^\star[\sigma]$ *are plotted as functions of the model parameter* $\sigma$. *For each* $\mathrm{d} \in \{\mathrm{d}_H, \mathrm{d}_{int}[g], \mathrm{d}_{sup}[g]\}$, *the average misclassification rates corresponding to* $\varphi_\mathrm{d}^\star$ *(including the* 90% *pointwise envelope based on the individual misclassification rates) are compared to average misclassification rates corresponding to* $\varphi_\mathrm{d}$ *(from Experiment 1).*

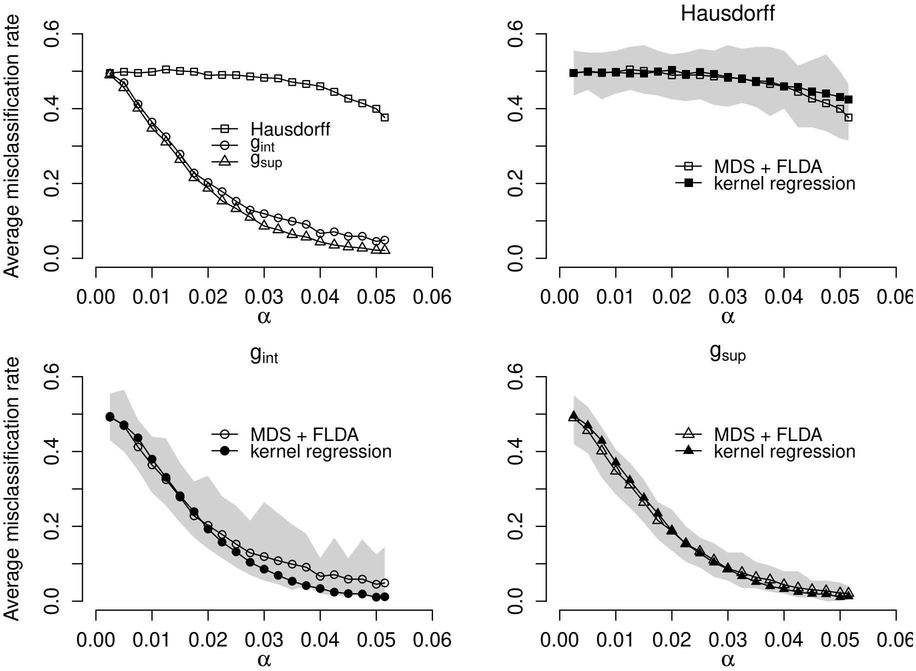

Fig. 9. *Average misclassification rates corresponding to $\varphi^{\star}_{H}[\alpha]$, $\varphi^{\star}_{g,int}[\alpha]$ and $\varphi^{\star}_{g,sup}[\alpha]$ are plotted as functions of the model parameter alpha. For each* $\mathrm{d} \in \{\mathrm{d}_H, \mathrm{d}_{int}[g], \mathrm{d}_{sup}[g]\}$*, the average misclassification rates corresponding to $\varphi^{\star}_{\mathrm{d}}$ (including the* 90% *pointwise envelope based on the individual misclassification rates) are compared to average misclassification rates corresponding to $\varphi_{\mathrm{d}}$ (from Experiment 1).*

For $\alpha \in \mathcal{A}$, the corresponding scenarios $\varphi_H[\alpha]$, $\varphi_{g,int}[\alpha]$, $\varphi_{g,sup}[\alpha]$, $\varphi^{\star}_{H}[\alpha]$, $\varphi^{\star}_{g,int}[\alpha]$ and $\varphi^{\star}_{g,sup}[\alpha]$ are considered.

*Results for Thomas process* Fig. 8 shows that $\varphi^{\star}_{H}[\sigma]$ is clearly outperformed by the two classification rules $\varphi^{\star}_{g,int}[\sigma]$ and $\varphi^{\star}_{g,sup}[\sigma]$. For small values of $\sigma$, $\varphi^{\star}_{g,sup}[\sigma]$ gives the lowest values (among the three classifiers based on multidimensional scaling and linear discriminant analysis) of the average misclassification rate. On the other hand, for $\sigma > 0.1$, $\varphi^{\star}_{g,int}[\sigma]$ has the best performance. When comparing the “MDS + LDA” classifiers with those based on kernel regression, we see that $\varphi^{\star}_{H}[\sigma]$ produces a lower average misclassification rate than $\varphi_H[\sigma]$ with the greatest difference between the average misclassification rates observed for $\sigma$ between 0.05 and 0.1. The classification rule $\varphi_{g,int}[\sigma]$ outperforms $\varphi^{\star}_{g,int}[\sigma]$ for small values of $\sigma$, for $\sigma > 0.1$ the choice of the “MDS + LDA” classifier leads to a slightly lower average misclassification rate. The differences in the performance of $\varphi_{g,sup}[\sigma]$ and $\varphi^{\star}_{g,sup}[\sigma]$ are almost negligible, with a small favor towards the use of $\varphi_{g,sup}[\sigma]$.

*Results for Gaussian determinantal process* For classification $\Pi$ vs $\Psi(\alpha)$, the situation with the three classifiers based on MDS and LDA is very similar to the results reported in Experiment 1. For $\alpha > 0.02$, $\varphi^{\star}_{g,sup}[\alpha]$ gives slightly lower average misclassification rates than $\varphi^{\star}_{g,int}[\alpha]$, but both clearly outperform $\varphi^{\star}_{H}[\alpha]$. The difference between the “MDS + LDA” classifiers and those based on kernel regression is negligible, except for the dissimilarity measure $\mathrm{d}_{int}[g]$, where the kernel regression classifier gives lower average misclassification rates while $\alpha > 0.03$.

*Summary* This experiment shows that the classifiers $\varphi_{g,int}, \varphi_{g,sup}$ performs (in the straightforward situation presented in Experiment 1) at least as well as the “MDS + LDA” classifiers proposed in (Mateu *et al.*, 2015). Furthermore, for small values of $\sigma$ and large values of $\alpha$ (i.e. situations where we expect low misclassification rates), the results favour the classifiers that use the kernel regression method. The only situation where the classification rules from (Mateu *et al.*, 2015) show better performance is in combination with the Hausdorff metric. However, classifiers based on the Hausdorff metric exhibit significantly higher average misclassification rates than those based on $\mathrm{d}_{int}[g]$ or $\mathrm{d}_{sup}[g]$. Note that in the following experiment, the Hausdorff metric $\mathrm{d}_H$ will not be considered due to its inferior performance in experiments 1 and 2.

## EXPERIMENT 3

This experiment studies the proposed method provided that the two models at hand belong to the same parametric family and differ in the value of the model parameter.

*Models, training and testing data* Let $\sigma_1 \in \{0.05, 0.1, 0.15\}$ and $\sigma_2 \in \Sigma$. We consider the classification $\Phi(\sigma_1)$ vs $\Phi(\sigma_2)$. Similarly, let $\alpha_1 \in \{0.02, 0.03, \alpha_{max}\}$, $\alpha_2 \in \mathcal{A}$, and consider the classification $\Psi(\alpha_1)$ vs $\Psi(\alpha_2)$. For each combination of $\sigma_1$ and $\sigma_2$, the training sets $\mathcal{T}(\sigma_1, \sigma_2, 20, 20)$ are composed of 20 realizations of $\Phi(\sigma_1)$ and 20 realizations of $\Phi(\sigma_2)$. The testing sets $\Gamma(\sigma_1, \sigma_2, 100, 100)$ are then composed of 100 + 100 realizations from the given models. Training and testing sets $\mathcal{T}(\alpha_1, \alpha_2, 20, 20)$, $\Gamma(\alpha_1, \alpha_2, 100, 100)$ (for each combination of $\alpha_1$ and $\alpha_2$) are defined correspondingly.

*Dissimilarity measures and classification scenarios* Following the notation in Experiment 1, the dissimilarity measure $\mathrm{d}_{int}[g]$ is considered. For $\sigma_1 \in \{0.05, 0.1, 0.15\}$ and $\sigma_2 \in \Sigma$, we denote by $\varphi_{g,int}[\sigma_1, \sigma_2]$ the classification rule

$$\varphi_{NW}\left(\cdot \mid \mathcal{T}(\sigma_1, \sigma_2, 20, 20), \mathrm{d}_{int}[g], \mathrm{K}, h_{k_{LCV}}\right),$$

where K and $h_{k_{LCV}}$ are as in Experiment 1. For $\alpha_1 \in \{0.02, 0.03, \alpha_{max}\}$ and $\alpha_2 \in \mathcal{A}$, the corresponding scenarios $\varphi_{g,int}[\alpha_1, \alpha_2]$ and $\varphi_{g,sup}[\alpha_1, \alpha_2]$ are considered. The performance of a classification scenario will be referred to as satisfactory if the corresponding average misclassification rate is $\leq 0.1$.

*Results for Thomas process* For each value of $\sigma_1$, we expect the average misclassification rate to be 0.5 for $\sigma_2 = \sigma_1$ and to decrease with increasing distance $|\sigma_1 - \sigma_2|$. Fig. 10 shows that the average misclassification rate corresponding to $\varphi_{g,int}[\sigma_1, \sigma_2], \sigma_1 = 0.05$, is below 0.1 expect for the few values of $\sigma_2$ that are the closest neighbors of $\sigma_1 = 0.05$. This is a consequence of the behaviour of the (theoretical) pair correlation function in the model with such strong clustering. The values of the pair correlation function change significantly with even small changes of $\sigma$. In addition, the 90% pointwise envelope is very narrow and the changes in its width with respect to $\sigma_2$ are negligible. For $\sigma_1 = 0.1$ and $\sigma_1 = 0.15$, the average misclassification rate is close to 0 for very small values of $\sigma_2$ but increases significantly towards 0.5 for $\sigma_2 = \sigma_1$ and decreases afterwards. The value of the average misclassification rate is below 0.1 for $\sigma_2 \in [0.02, 0.05]$ in the first case and $\sigma_2 \in [0.02, 0.7]$ in the second case. The 90% pointwise envelopes are narrow for the smallest values of $\sigma_2$, and their width increases as $\sigma_2$ is growing towards 0.2. Loosely speaking, the realizations of $\Phi(\sigma)$ with the value of $\sigma$ corresponding to mild or weak clustering can be distinguished successfully from the realizations of $\Phi(\sigma)$ with $\sigma$ sufficiently small (representing strong clustering). Similar observations can be made for the maxima absolute deviation counterpart $\mathrm{d}_{sup}[g]$ of the dissimilarity measure (see Fig. S7.14 and Fig. S7.15 in the Supplementary material).

*Results for Gaussian determinantal process* We again expect the average misclassification rate to be 0.5 for $\alpha_2 = \alpha_1$, and to decrease with $|\alpha_1 - \alpha_2|$ increasing. For $\alpha_1 = 0.02$, Fig. 11 shows that the average misclassification rate corresponding to $\varphi_{g,int}$ starts at 0.2, then increases to 0.5 and then decreases as $\alpha_2$ grows towards its maximal values. For $\alpha_2 > 0.04$, the average misclassification rate is below 0.1 Similarly, for $\alpha_1 = 0.03$ the average misclassification rate is below 0.1 for the two smallest values of $\alpha_2$ as well as for the two largest values. For $\alpha_1 = \alpha_{max}$, the situation is different. In this case, we start with a nearly perfect classification, the average misclassification rate stays below 0.1 for $\alpha_2 \leq 0.03$, then increases sharply towards 0.5. This is consistent with the fact that $\alpha_{max}$ represents the most repulsive model whereas $\alpha < 0.02$ represents weak repulsion. The difference in performance for the classifier based on the maximum absolute deviation counterpart $\mathrm{d}_{sup}[g]$ of the dissimilarity measure is negligible.

*Summary* For binary classification $\Phi(\sigma_1)$ vs $\Phi(\sigma_2)$ or $\Psi(\alpha_1)$ vs $\Psi(\alpha_2)$, the average misclassification rates corresponding to $\varphi_{g,int}$ decrease with increasing distance $|\sigma_1 - \sigma_2|$ or $|\alpha_1 - \alpha_2|$. Classification between models with strong interactions and weak interactions is very successful, but for models with similar properties (similar values of model parameters), the average misclassification rates are high. Recall that all of the realizations in this experiment are observed on the unit square. Sect. S8 in the Supplementary material presents a repetition of this experiment with a larger observation window, studying the impact of the increasing number of points per realization on the performance of the kernel regression classifier.

## REAL-DATA EXAMPLE

To illustrate the proposed methodology, we apply our classification procedure to a collection of 68 point patterns representing the centers of intramembranous particles located in the mitochondrial membranes of the HeLa cell line. These data were collected using the freeze-fracture technique (Schladitz *et al.*, 2003).

During data collection, the cell line was observed in three different environments to study mitochondrial metabolism: under normal conditions, after exposure to sodium acid, and after exposition to rotenone. Therefore, we distinguish three groups of patterns: a control group corresponding to standard

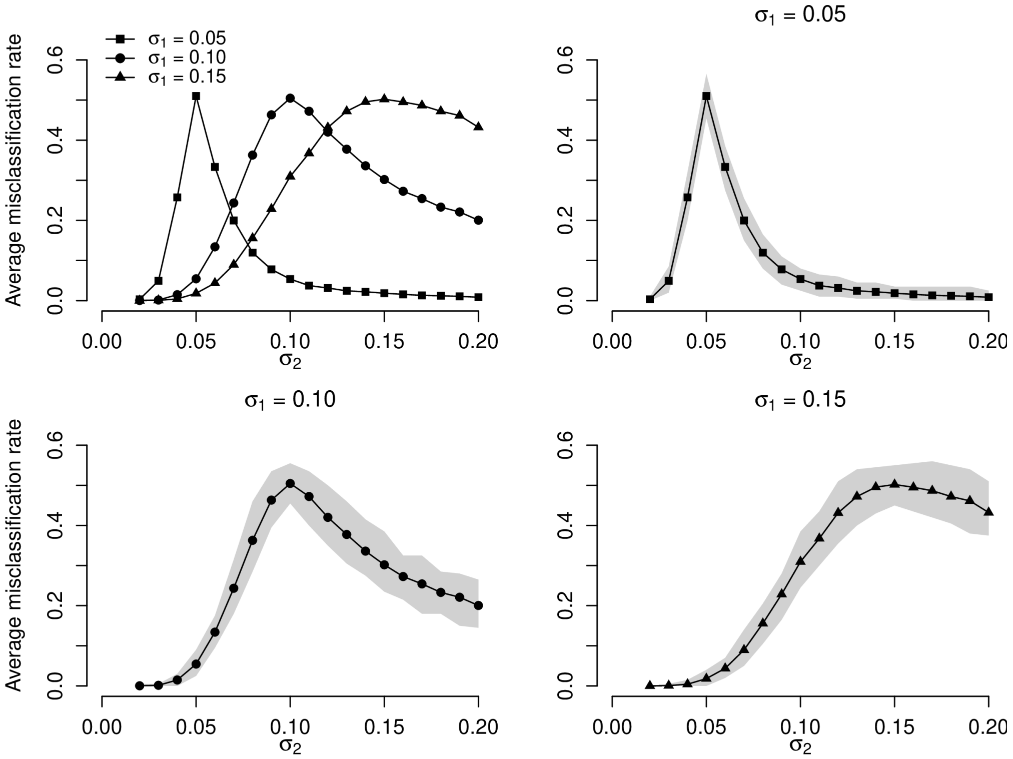

Fig. 10. *The average misclassification rates* $\bar{\gamma}(\varphi_{g,int}[\sigma_1,\sigma_2])$, $\sigma_1 \in \{0.05, 0.1, 0.15\}$, *are plotted as functions of the model parameter* $\sigma_2$. *The variability of the sequences of the individual misclassification rates are illustrated with the* 90% *pointwise envelopes.*

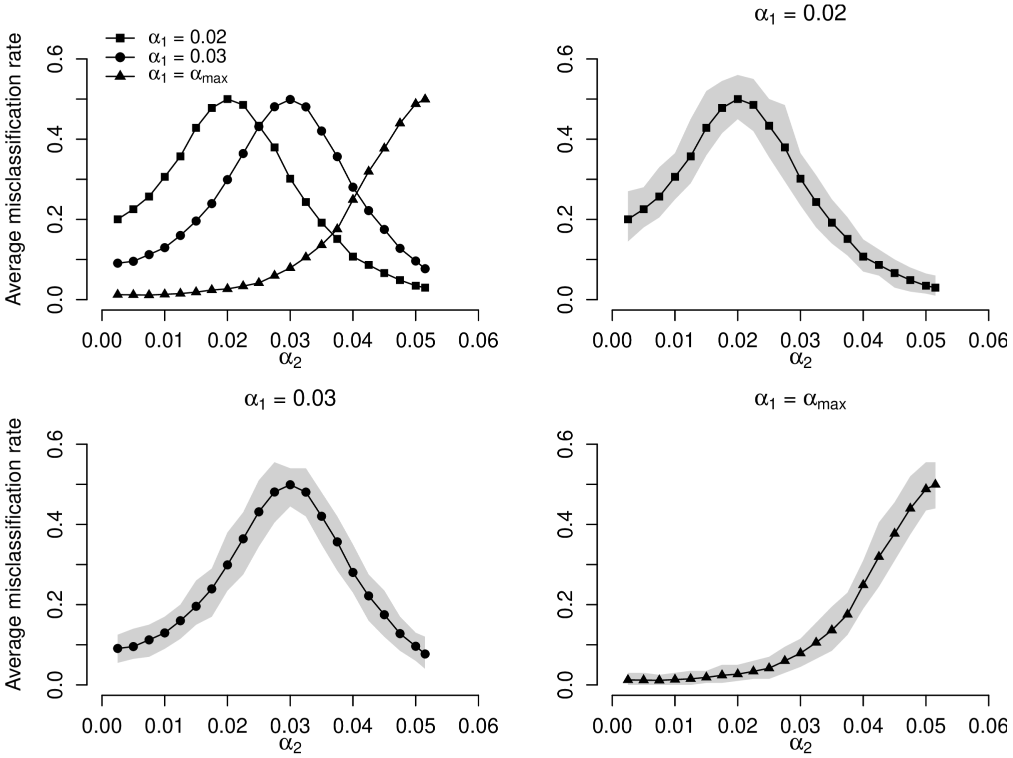

Fig. 11. *The average misclassification rates* $\bar{\gamma}(\varphi_{g,int}[\alpha_1,\alpha_2])$, $\alpha_1 \in \{0.02, 0.03, \alpha_{max}\}$, *are plotted as functions of the model parameter* $\alpha_2$. *The variability of the sequences of the individual misclassification rates are illustrated with the* 90% *pointwise envelopes.*

conditions (33 patterns), the first group corresponding to the sodium acid environment (14 patterns), and the second group corresponding to the rotenone environment (21 patterns). One (randomly chosen) pattern from each group is plotted in Fig. 1.

For all the observed patterns, we fix a squared observation window with an edge length of 336 *nm*. According to the analysis in (Schladitz *et al.*, 2003), observations in all three groups exhibit a hard-core property for very small distances. It means that pairs of points that are very close to each other do not occur in the patterns – the intramembranous particles, whose centers are recorded, do not overlap. Weak repulsion between points of the process occurs on the scale from 10 *nm* to 20 *nm*. Weak aggregation can be observed for interpoint distances greater than 20 *nm*.

We perform ternary classification using the Bayes classifier in combination with the $k$-nearest neighbors algorithm and the kernel regression method (including the local choice of the optimal $k$). After some preliminary observations, we have decided to use $g$, $L$ and $G$ to build the dissimilarity measures. While $L$ is derived from the pair correlation function, the characteristic $G$ is based on interpoint distances; see (Møller and Waagepetersen, 2004) for further details. We set $R(g) = R(L) = 84$ *nm*, that is, 1/4 of the edge length of the observation window, and $R(G) = 66$ *nm*. The classification is performed as follows: we fix one of the 68 patterns, consider the remaining 67 patterns as training data, and predict the label of the fixed pattern. We then compare the predicted labels to the true ones to compute the misclassification rate, see Table 1. Furthermore, we report the number of misclassified patterns in each group.

Table 1 shows that the use of $G$ leads to the lowest misclassification rate (for both versions of the dissimilarity measure). However, more than one-quarter of the patterns are misclassified, even in the best scenario. Note that from the sodium acid group, 9 resp. 8 patterns are not classified correctly. This group contains the smallest number of patterns (14). For $g$ and $L$, the misclassification rates based on $\mathrm{d}_{sup}$ are the same. With $\mathrm{d}_{sup}[L]$, 60% of the patterns in the rotenone group are misclassified. With $\mathrm{d}_{sup}[g]$, the sodium acid group is the most problematic. The highest misclassification rate is observed for $\mathrm{d}_{int}[L]$, with almost all patterns in the noncontrol groups labelled wrongly. For $\mathrm{d}_{int}[g]$, one-third of the patterns in the control group (the largest group, containing 33 patterns) are misclassified. Visualisation of the dissimilarities between the elements of this dataset can be seen in Fig. S9.20 and Fig. S9.21 in the Supplementary material.

In conclusion, none of the three summary characteristics considered in this section provides a satisfactory ternary classification. Suppose that we select the control and sodium acid groups and consider binary classification. In that case, we expect good performance from the classifier based on $\mathrm{d}_{sup}[L]$, see the number of misclassified patterns from individual groups in Table 1. Similarly, we expect that the classifier based on $\mathrm{d}_{int}[G]$ will provide satisfactory results for binary classification between the control and rotenone groups. The same applies to the rotenone and sodium acid groups. To improve the ternary classification, we need to tune up the classifiers, e.g. by identifying another summary characteristic, better capturing the differences between the three groups.

## DISCUSSION

This paper proposes a methodology for the supervised classification of point patterns based on their representation by a selected functional summary characteristic. The presented simulation experiments confirm that the Bayes classifier in combination with the $k$-nearest neighbors algorithms and the kernel regression method is successful in solving the problem.

The simulation experiments cover the three main classes of models: aggregation, complete spatial randomness, and repulsion. The particular models considered in this paper represent the typical behavior in their respective classes and are often used in practice, thus providing a good picture of the problem. Of course, other models could also be considered.

The simulation study focuses mainly on the pair correlation function, selected for its simple interpretation and popularity in the applied literature. However, many other functional summary characteristics are available, and to make an appropriate choice, one should use the expert knowledge of the problem at hand.

In a specific application, choosing an appropriate version of the classifier with all tuning constants is a difficult task. For that reason, seeking generally applicable recommendations is useless. Our decisions should be guided by expert knowledge about the particular dataset. When several candidate (versions of) classifiers are assumed to be relevant, we suggest investigating their performance in the training dataset using an appropriate cross-validation scheme.

Finally, we remark that the proposed method can be directly extended to more complicated settings, such as random sets, provided that relevant summary characteristics are available.

Table 1. *Ternary classification problem: the Bayes classifier in combination with the k-nearest neighbors algorithm and the kernel regression method is applied to the point pattern data from (Schladitz et al., 2003). The misclassification rate is reported, as well as the number of misclassified patterns in each group. The left-hand side of every column corresponds to the integral version of the dissimilarity measure, and the right-hand side corresponds to its maximum absolute deviation counterpart.*

| | **Avg. m. rate** | | **Control** | | **Sodium acid** | | **Rotenone** | |
|---|---|---|---|---|---|---|---|---|
| $g$ | 0.324 | 0.353 | 11 | 8 | 5 | 9 | 6 | 7 |
| $L$ | 0.500 | 0.353 | 6 | 7 | 12 | 4 | 16 | 13 |
| $G$ | 0.294 | 0.265 | 6 | 6 | 9 | 8 | 5 | 4 |

**Acknowledgements** We are very grateful to Frédéric Ferraty and Philippe Vieu for making their code for the kernel regression classifier accompanying the book (Ferraty and Vieu, 2006) available on their website. In the real-data example, we use the data from (Schladitz *et al.*, 2003) with the kind permission of Prof. Schladitz. We are grateful to the whole team of Prof. Mattfeld for their effort during the collection of the data. This work has been supported by the Charles University Grant Agency, project no. 1198120, and The Czech Science Foundation, project no. 19-04412S.

## REFERENCES

Alba-Fernández MV, Ariza-López FJ, Dolores Jiménez-Gamero M, Rodríguez-Avi J (2016). On the similarity analysis of spatial patterns. Spat Stat 18:352–62.

Ayala G, Epifanio I, Simo A, Zapater V (2006). Clustering of spatial point patterns. Comput Stat Data Anal 50:1016–32.

Baddeley A, Gill R (1997). Kaplan-Meier estimators of distance distributions for spatial point processes. Ann Stat 25:263–92.

Baddeley A, Rubak E, Turner R (2015). Spatial point patterns: methodology and applications with R. Chapman and Hall/CRC Press, Boca Raton.

Baddeley A, Silverman BW (1984). A cautionary example for the use of the second-order methods for analysing point patterns. Biometrics 40:1089–94.

Borchers H (2019). pracma: practical numerical math functions, R package version 2.2.9. https://CRAN.R-project.org/package=pracma

Cholaquidis A, Forzani L, Llop P, Moreno L (2017). On the classification problem for Poisson point processes. J Multivar Anal 153:1–15.

Dai W, Athanasiadis S, Mrkvička T (2021). A new functional clustering method with combined dissimilarity sources and graphical interpretation. In Computational Statistics and Applications (ed. R. López-Ruiz), IntechOpen, London. https://www.intechopen.com/chapters/79248 doi: 10.5772/intechopen.100124

Daley D, Vere-Jones D (2008). An introduction to the theory of point processes. Volume II: general theory and structure. 2nd edn. Springer, New York.

Dasgupta A, Raftery AE (1998). Detecting features in spatial point processes with clutter via model-based clustering. J Am Stat Assoc 93:294–302.

Ferraty F, Vieu P (2006). Nonparametric functional data analysis. Theory and practice. Springer-Verlag, New York.

Fisher RA (1936). The use of multiple measurements in taxonomic problems. Ann Eugen 7:179–88.

Fisher RA (1938). The statistical utilization of multiple measurements. Ann Eugen 8:376–86.

Hoffman JR, Mahler RPS (2004). Multitarget miss distance via optimal assignment. IEEE Trans Syst Man Cybern Syst Part A 34:327–36.

Illian J, Penttinen A, Stoyan H, Stoyan D (2004). Statistical analysis and modelling of spatial point patterns. Wiley, Chichester.

Koňasová K, Dvořák J (2021). Stochastic reconstruction for inhomogeneous point patterns. Methodol Comput Appl Probab 23:527-–47.

Lavancier, F, Møller J, Rubak, E (2015). Determinantal point process models and statistical inference. J R Stat Soc 77:853–77.

Macchi, O (1975). The coincidence approach to stochastic point processes. Adv Appl Probab 7:83–122.

Mateu J, Schoenberg FP, Diez DM, González JA, Lu W (2015). On measures of dissimilarity between point patterns: classification based on prototypes and multidimensional scaling. Biom J 57:340–58.

Wallig M, Weston S, Tenenbaum D (2019). doParallel: foreach parallel adaptor for the 'parallel' Package, R package version 1.0.15. https://CRAN.R-project.org/package=doParallel

Møller J, Waagepetersen RP (2004). Statistical inference and simulation for spatial point processes. Chapman & Hall/CRC, Boca Raton.

R Core Team (2017). R: a language and environment for statistical computing. R Foundation for Statistical Computing. Vienna, Austria. https://www.R-project.org

Redenbach C, Särkkä A, Sormani M (2015). Classification of points in superpositions of Strauss and Poisson processes. Spat Stat 12:81–95.

Ripley, BD (1976) The second-order analysis of stationary point processes. J Appl Prob 13:255–66.

Schladitz K, Särkkä A, Pavenstädt I, Haferkamp O, Mattfeldt T (2003). Statistical analysis of intramembraneous particles using freeze fracture specimens. J Microsc 211:137–53.

Schuhmacher D, Vo BT, Vo BN (2008). A consistent metric for performance evaluation of multi-object filters. IEEE Trans Signal Process 56:3447–57.

Tranbarger Freier KE, Schoenberg FP (2010). On the computation and application of prototype point patterns. Open Appl Informat J 4:1–9.

Thomas M (1949). A generalization of Poisson's binomial limit for use in ecology. Biometrika 36:18–25.

Tscheschel A, Stoyan D (2006). Statistical reconstruction of random point patterns. Comput Stat Data Anal 51:859–871.

Victor JD , Purpura K (1997). Metric-space analysis of spike trains: theory, algorithms and application. Netw Comput Neural Syst 8:127–64.

Vo BN, Dam N, Phung D, Tran QN, Vo BT (2018). Model-based learning for point pattern data. Pattern Recognit 84:136–51.

Walsh D, Raftery AE (2005). Classification of mixtures of spatial point processes via partial Bayes factors. J Comput Graph Stat 15:139–54.