Support Vector Machines With Uncertainty Option and Incremental Sampling for Kriging

Chen Xiong, Paul Honeine, Maxime Berar, Antonin van Exem

Support vector machines with uncertainty option and incremental sampling for kriging

Chen Xiong[1*], Paul Honeine[2], Maxime Berar[2], Antonin van Exem[3]

[1*]LS2N, IMT Atlantique, Nantes, 44300, France.
[2]Univ Rouen Normandie, INSA Rouen Normandie, Universite Le Havre Normandie, Normandie Univ, LITIS UR 4108, Rouen, F-76000, France.
[3]Tellux, Petit Couronne, F-76650, France.

*Corresponding author(s). E-mail(s): chen.xiong@imt-atlantique.fr;
Contributing authors: paul.honeine@univ-rouen.fr;
maxime.berar@univ-rouen.fr; antonin.vanexem@tellux.fr;

Abstract

This paper presents a novel approach to pollution assessment by investigating Support Vector Machines (SVM) with an uncertainty option to overcome the limitations of traditional kriging. While kriging is a major tool for geostatistical modelling, allowing to estimate the distribution of contaminants in a region from a small set of samples, it does not allow to extract also the uncertainty map. An uncertainty map is of great interest, as it allows to identify regions of high uncertainty where one should sample in order to reduce high level of uncertainties. In this paper, we propose two variants of the SVM with an uncertainty option, each using a different hinge loss to improve the accuracy and efficiency. These losses allow to estimate different levels of contaminations, as well as uncertainty, such as the three levels: positive, uncertain and negative, namely for pollution estimation: high-pollution, uncertain and low-pollution. In addition to the exploration of SVM variants, we propose an innovative active sample selection strategy based on the uncertainty criterion. This strategy is designed to systematically reduce uncertainties in pollution assessment, thus providing adaptability to dynamic environmental changes. An incremental SVM with an uncertainty option is introduced to further optimize the sample selection process. Furthermore, the decision-making process is refined through the introduction of a novel three-hinge loss. The corresponding optimization problem and its resolution allow for a more nuanced contamination assessment with multiple levels of estimation, providing a valuable tool for characterizing contamination levels with increased granularity. Extensive experiments on synthetic and real data validate the proposed methodology. Synthetic data simulations assess the quality of the approach, while real

data from a two-dimensional porosity measurement demonstrate practical applicability. This research contributes to the advancement of pollution assessment methodologies, providing an adaptable solution for environmental monitoring.

Keywords: Kriging, support vector machines, uncertainty, sample selection, incremental algorithm, pollution assessment

1 Introduction

Kriging and variography are major tools for geostatistical modelling and machine learning in the geosciences (Chilès and Desassis, 2018; Dramsch, 2020). Also known as Gaussian process regression, kriging is an interpolation model based on a prior covariance that controls the Gaussian process. Coupled with variography, it has been widely used in geostatistics for data distribution, allowing concentrations of substances on a map to be estimated from limited sampling information and the standard deviation to be calculated over the entire map. Examples of kriging applications include estimating pollution in groundwater and soil for various contaminants (McLean et al., 2019; Sun et al., 2019; Ouabo et al., 2020).

Sampling costs are relatively high in the geosciences. Thanks to its underlying priors, kriging can operate with a small number of samples, unlike deep learning (not to mention some recent attempts to overcome this problem with deep generative models (Rakotonirina et al., 2024a,b)). Still, due to limited sampling, data have incomplete coverage, yielding uncertainties in estimations. Uncertainties in the estimations due to incomplete and imprecise knowledge are a major issue, as demonstrated in the large literature of reservoir modeling to address subsurface heterogeneities (Pyrcz and Deutsch, 2014; Liu et al., 2021). Moreover, many diverse case studies corroborate the fact that the model is increasingly uncertain with distance away from the well-known locations, and these values cannot be cross-checked in the absence of additional data, as demonstrated for instance in the 3D hydrogeological characterization of the New Jersey Shelf (Thomas et al., 2022).

While the given examples and references highlight the issue of uncertainties in kriging estimations, it is worth noting that uncertainty underlines (statistical) machine learning. Advanced statistical studies of uncertainty distinguish between two different sources of uncertainty: statistical and systematic uncertainties, which are related to the ideas of accuracy and precision in statistics. Recently, this definition was extended to machine learning, under the categorization of aleatoric and epistemic uncertainties by Hüllermeier and Waegeman (2021) or data uncertainty and model uncertainty by Gawlikowski et al. (2023), as well as the concepts of conflict and ignorance uncertainty/ambiguity by Hüllermeier and Brinker (2008). In few words, aleatoric or statistical uncertainty depicts randomness, namely, the variability in the outcome of an experiment due to inherently random effects. The epistemic or systematic uncertainty is due to a lack of knowledge of the optimal model. In the present paper, we focus on the latter, namely epistemic or systematic uncertainty due to ignorance. The amount of such a model uncertainty reduces with an increasing number of training samples, as

it has been well-known for density estimation and was recently demonstrated for version space learning and Bayesian inference by Hüllermeier and Waegeman (2021), and more recently for deep neural networks. See Gawlikowski et al. (2023); Psaros et al. (2023) for recent surveys.

In this paper, we study the problem of estimating the contamination map, with the possibility of estimating uncertain regions, which is of great interest since it allows to assess the regions where the contamination estimates are not relevant but need more samples to reduce these uncertainties. It turns out that kriging does not allow to properly address this topic[1]. Roughly speaking, kriging is used to predict the distribution of contaminants on a map, and can also generate the variance on the entire map in order to study the map points with the greatest uncertainty. However, the variance map obtained from kriging is such that areas with large variances are often associated with areas with fewer samples. However, this information is not very useful for pollution assessment, nor to identify further sampling locations, as it does not integrate the level of pollution in the results. For instance, if areas with high variances have low contamination, it would not be relevant to sample further in those areas (See Section 6.1 and Fig. 6 for an illustration). Therefore, we need to be skeptical of locations with estimated contamination values close to the frontier of detection/classification, namely uncertain zones where further sampling would be of great interest.

In this paper, we aim to overcome these limits in kriging, by integrating an uncertainty option in the decision, which naturally provides a mechanism for future sampling to reduce the uncertainty. To this end, we revisit Support Vector Machines (SVM) in order to generate a decision function for three classes[2]: positive, uncertain and negative classes (e.g. high-pollution, uncertain and low-pollution, respectively). Such a decision is of practical interest because it provides the pollution remediation experts the 3 regions of interest: the region in the map to be decontaminated, the uncertain region where further sampling needs to be carried out, and the region that does not exceed the admissible pollution level. To this end, we consider the framework of SVM for many reasons: SVM perform better on smaller datasets and are less prone to overfitting than neural networks thanks to solving a convex optimization problem, and they are also computationally faster than deep neural networks for prediction. SVM remain central in Machine Learning (Campbell and Ying, 2022; Pisner and Schnyer, 2020; Cervantes et al., 2020; Hu et al., 2021; Menaka and Ganesh Vaidyanathan, 2023) and have been recently explored on kriging (De Caires et al., 2024; Chen et al., 2020; Wu et al., 2023) and toxicity/pollution analysis (Leong et al., 2020; Jha and Yoon, 2020; Huang et al., 2023). To the best of our knowledge, this is the first time that

[1]The current work aims to assess uncertainties in the estimated output, which is not similar to addressing uncertainties of the input data, where data is often modeled by means of intervals or fuzzy intervals. While many researchers have been tackling input data uncertainties, even though yielding mathematically debatable methods with intractable algorithms (Loquin and Dubois, 2010), providing output uncertainties seems to be novel, even though kriging metamodels were proposed to address epistemic uncertainties (Fuhg et al., 2021).

[2]The addressed problem is not a multiclass one, since in the latter the classes are not ordered, and the miss-classification cost is independent of the target-estimated classes. The problem addressed in this paper is a binary classification problem "detected pollution versus undetected pollution", with the integration of an uncertainty region between them. Moreover, in Section 5 we extend the proposed approach to multi-level estimation, beyond the binary case

an uncertainty option is investigated for estimation and classification, and that SVM with an uncertainty option is investigated for sample selection.

Our methodology relies on revisiting SVM for binary-classification, by integrating an uncertainty option, namely the classifier is allowed to predict the "uncertain" label. It turns out that this is essentially SVM with a reject option, as defined by Grandvalet et al. (2008); Wegkamp and Yuan (2011); Hanczar and Sebag (2014) and studied more recently by Franc et al. (2023). We show how two variants can be implemented, considering different double-hinge losses and regularization. Since the proposed SVM formalism integrates uncertainty within the estimation, we use this information in order to ingeniously select the future sample that would allow to highly reduce the uncertainty. Moreover, we derive an incremental algorithm for the proposed SVM with an uncertainty option.

Finally, we extend the proposed SVM with an uncertainty option, based on binary-classification, to address more levels. To this end, we introduce a triple-hinge loss, allowing to extract 5 ordered classes, and derive the corresponding optimization problem with its resolution. This is of interest in soil pollution assessment, because it allows to go beyond the 3 classes "contaminated vs uncertainty vs non-contaminated" to a more fine-grained assessment with several levels of contamination. The proposed triple-hinge loss allows to define five levels of estimation, which could be viewed in contamination assessment as: very high, high, moderate, light and very light.

The main contributions of this paper are as follows:

- We investigate SVM with an uncertainty option in order to overcome limitations of the kriging in pollution assessment. Two variants are examined with different hinge losses.
- We propose an active sample selection strategy based on this criterion and design an appropriate incremental SVM with an uncertainty option.
- We propose to refine the decision by introducing a novel three-hinge loss and deriving the corresponding optimization problem and its resolution.

To demonstrate the relevance of the proposed methodology, we conduct experiments on both synthetic and real datasets. The simulated synthetic dataset allow to assess the quality, while two real datasets are used to assess all the aforementioned contributions. The first real dataset is a two-dimensional porosity measure using 200 wells. It is worth noting that the porosity distribution is of great interest in geostatistics, as demonstrated in the wide literature on the topic, such as by Pyrcz et al. (2005) and Thomas et al. (2022). The second dataset is a well-known dataset for the pollution assessment of the Meuse river floodplains, consisting of topsoil of 4 heavy metal concentrations and organic matter.

The rest of the paper is as follows. In the next section, we provide some background material on SVM, before introducing the two variants of the SVM with an uncertainty option in Section 3. Section 4 presents sample selection and the incremental algorithm. The extension of this work to multi-level estimation is given in Section 5. Experiments are conducted in Section 6. The paper is concluded in Section 8 with a conclusion and future work.

2 Background on SVM

Let $\mathcal{X} \in \mathbb{R}^d$ be the space under investigation, with $d = 2$ for a two-dimensional area. Let $x_1, x_2, \ldots, x_n \in \mathcal{X}$ be the available samples (e.g. geographic locations) with target values $y_1, y_2, \ldots, y_n$, respectively (e.g. contaminant concentrations at these locations). For a binary classification task (also known as detection), the target labels are either -1 or $+1$ (e.g. non-contaminated vs contaminated). We aim to find a function $f(x)$ defined on $\mathcal{X}$ that computes the contamination level class for any $x \in \mathcal{X}$.

To this end, one aims to minimize the regularized empirical risk function

$$R(x) = \frac{1}{n} \sum_{i=1}^{n} \mathcal{L}\left(y_i f(x_i)\right) + \rho\, \mathcal{R}(f), \tag{1}$$

for some loss function $\mathcal{L}(\cdot)$ and regularization function $\mathcal{R}(\cdot)$, where ρ is tradeoff parameter. From the representer theorem (Unser, 2021), it is known that the optimal function that minimized the above risk function takes the form

$$f(x) = \sum_{j=1}^{n} \lambda_j\, \kappa(x, x_j), \tag{2}$$

for some kernel $\kappa(\cdot, \cdot)$, such as the Gaussian kernel defined by $\kappa(x_i, x_j) = \exp(-\|x_i - x_j\|^2/2\sigma^2)$ for a bandwidth parameter σ. The functional minimization of (1) boils down to the estimation of the n coefficients $\lambda_1, \lambda_2, \ldots, \lambda_n$.

This formulation is general and the representer theorem is valid under mild conditions on the loss function $\mathcal{L}$ and the regularization function $\mathcal{R}$. SVM with a binary classification task considers the hinge loss defined by

$$\mathcal{L}_{hinge}(z) = \max\{0, 1 - z\}.$$

The often used regularization functions are the ℓ_1 and ℓ_2 norms defined respectively by

$$\|\lambda\|_{\ell_1} = \sum_{j=1}^{n} |\lambda_j| \quad \text{and} \quad \|\lambda\|_{\ell_2}^2 = \sum_{j=1}^{n} \lambda_j^2.$$

3 SVM with an uncertainty option

In this section, we introduce SVM with an uncertainty option so that, in a detection context, the prediction of the classifier can be "detected vs uncertain vs undetected". Using mathematical expressions, we define the problem in the following.

In conjunction with the -1 and $+1$ labels, we include the label 0 in our decision function, which corresponds to the uncertainty decision. Let $decision(x_i) \in \{-1, 0, 1\}$ for $i = 1, \cdots, n$ be the labels of $x_1, x_2, \ldots, x_n$, with values -1, 0, and 1 for undetected, uncertain, and detected, respectively (e.g. non-contaminated, uncertain and contaminated). In order to integrate the uncertainty option in SVM, we investigate

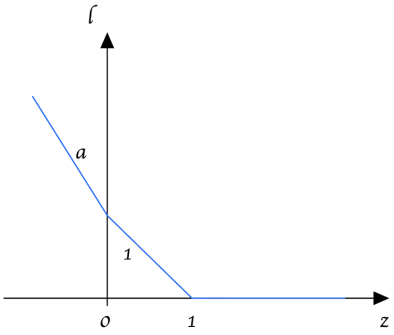

Fig. 1 The double-hinge loss

two versions using two different double-hinge losses following Wegkamp and Yuan (2011) and Grandvalet et al. (2008) with different regularizations ℓ_1 and ℓ_2.

3.1 SVM with an uncertainty option by ℓ_1 regularization (LPSVM)

The double-hinge loss $\mathcal{L}_{\text{2-hinge}}$ introduced by Wegkamp and Yuan (2011) is defined, as illustrated in Fig. 1, by:

$$\mathcal{L}_{\text{2-hinge}}(z) = \begin{cases} 1 - az & \text{if } z \leq 0 \\ 1 - z & \text{if } 0 \leq z \leq 1 \\ 0 & \text{if } z > 1 \end{cases} \tag{3}$$

where $a = (1-b)/b > 1$ and b is user-defined. By injecting this loss in (1) and considering the ℓ_1 regularization, the resulting optimization problem can be conveniently formulated as a linear program.

The above optimization problem can be solved by introducing slack variables ξ_i, leading to the following problem

$$\min_{\lambda,\xi} \xi_1 + \cdots + \xi_n + \rho(\xi_{n+1} + \cdots + \xi_{2n}) \tag{4}$$

$$\text{subject to } \begin{cases} \xi_i \geq 0 & \text{for } i = 1, \cdots, n \\ \xi_i \geq 1 - y_i h_i & \text{for } i = 1, \cdots, n \\ \xi_i \geq 1 - a y_i h_i & \text{for } i = 1, \cdots, n \\ h_i = \sum_{j=1}^{n} \lambda_j \, \kappa(x_i, x_j) & \text{for } i = 1, \cdots, n \\ \xi_{n+i} \geq \lambda_i & \text{for } i = 1, \cdots, n \\ \xi_{n+i} \geq -\lambda_i & \text{for } i = 1, \cdots, n \end{cases} \tag{5}$$

Indeed, any ξ_i (for $i = 1, \cdots, n$) that satisfies the first 3 constraints is a minimizer of $\mathcal{L}_{\text{2-hinge}}(y_i f(x_i))$, the fourth constraint is for computational convenience, and the last 2 constraints allow to represent $|\lambda_i|$, for $i = 1, \cdots, n$,

This linear programming problem can be solved using solvers such as CPLEX and Gurobi Optimizer (e.g. `cvxpy` in Python), as shown in a typical implementation in

Algorithm 1 Implementation of the LPSVM using `cvxpy` in Python

```
import cvxpy as CP
# The variables
var = cp.Variable(n+2*m)
# The optimization problem
A = np.ones(n+m)
A[n:ntm] = A[n:n+m]*r
objective = cp.Minimize(A @ var[ :n+m])
# The constraints
constraints = [np.zeros(n) <= var[:n]]
constraints += [np.ones(n) - y @ (var[n+m:] @ f) <= var[ :n]]
constraints += [np.ones(n) - a * y @ (var[ntm:] @ f) <= var[:n]] constraints
    += [var[ntm:] <= var[n:n+m], -var[n+m:] <= var[n:n+m]]
# The problem resolution
prob = cp.Problem(objective, constraints)
prob.solve( )
res = var.value
```

Algorithm 1. This variant is denoted in the following LPSVM for Linear Programming SVM.

3.2 SVM with an uncertainty option by ℓ_2 regularization (QPSVM)

A more complex double-hinge loss is introduced by Grandvalet et al. (2008), by considering the Bayes decision theory with an explicit definition of the costs of wrong decisions and of abstaining from taking any decision, namely the uncertainty option. Let c_- be the cost of a false positive (i.e., a sample labelled -1 is predicted as $+1$), and c_+ be the cost of a false negative (i.e., a sample labeled $+1$ is predicted as -1). Likewise, let r_- and r_+ be the costs of choosing the uncertain option for samples labeled -1 and $+1$.

The double-hinge loss illustrated in Fig. 2 can be defined in two parts as follows:

- If $y_i = +1$:

$$\mathcal{L}_{\text{2-hinge+}}(z) = \max\Big\{-(1-p_-)z + H(p_-), -(1-p_+)z + H(p_+), 0\Big\}$$

- If $y_i = -1$:

$$\mathcal{L}_{\text{2-hinge-}}(z) = \max\Big\{-p_+ z + H(p_+), -p_- z + H(p_-), 0\Big\}$$

where $p_+ = \frac{c_- - r_-}{c_- - r_- + r_+}$, $p_- = \frac{r_-}{c_+ - r_+ - r_-}$ and $H(p) = -p\log(p) - (1-p)\log(1-p)$. This double-hinge loss can be related to $\mathcal{L}_{\text{2-hinge}}$ defined in Section 3.1 (Wegkamp and Yuan, 2011) when considering a symmetric decision with $c_- = c_+ = 1$, and $r_- = r_+$, with the uncertainty occurring when the latter is less than 0.5.

Considering the ℓ_2 regularization, we get a quadratic programming problem, as derived in the following. Let $D = \frac{1}{\rho}(p_+ - p_-)$, $C_i = \frac{1}{\rho}(1 - p_+)$ for positive samples, and $C_i = \frac{1}{\rho}p_-$ for negative samples. With the introduction of slack variables ξ_i and η_i

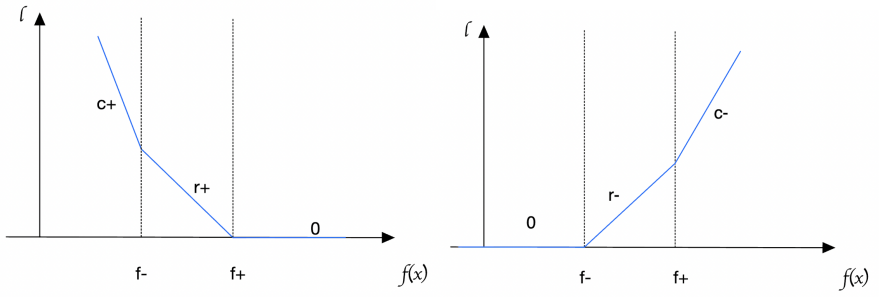

Fig. 2 The double-hinge loss of QPSVM for positive and negative cases

for $i = 1, \cdots, n$, the optimization problem can be written as follows

$$\min_{f,\xi,\eta} \tfrac{1}{2}\|f\|^2_{\ell_2} + \sum_{i=1}^{n} C_i \xi_i + D \sum_{i=1}^{n} \eta_i \tag{6}$$

$$\text{subject to } \begin{cases} y_i f(x_i) \geq t_i - \xi_i & \text{for } i = 1, \cdots, n \\ y_i f(x_i) \geq \tau_i - \eta_i & \text{for } i = 1, \cdots, n \\ \xi_i \geq 0 & \text{for } i = 1, \cdots, n \\ \eta_i \geq 0 & \text{for } i = 1, \cdots, n \end{cases} \tag{7}$$

where $t_i = H(p_+)/(1-p_+)$, $\tau_i = (H(p_-) - H(p_+))/(p_- - p_+)$ for x_i in the positive class, and $t_i = H(p_-)/p_-$, $\tau_i = (H(p_-) - H(p_+))/(p_- - p_+)$ for x_i in the negative class.

Following the representer theorem giving the general (2), and by using the variable change $\gamma_i y_i = \lambda_i$, we get the dual optimization formulation

$$\min_{\alpha,\gamma,\zeta} \tfrac{1}{2}\gamma^\top G\gamma - \tau^\top \gamma - (t-\tau)^\top \alpha + \zeta y^\top \gamma, \tag{8}$$

where $y = (y_1, \cdots, y_n)^\top$, $t = (t_1, \cdots, t_n)^\top$, $\tau = (\tau_1, \cdots, \tau_n)^\top$ et $G_{ij} = y_i\, y_j\, \kappa(x_i, x_j)$. To solve this optimization problem, Grandvalet et al. (2008) use an active variable algorithm proposed by Vishwanathan and Murty (2002), as described in the following.

The training dataset is partitioned into five subsets designated by the active box constraints of the optimization problem (8). The training samples are indexed by

$$\begin{cases} I_0 = \{i \mid \gamma_i = 0\} & \text{such that } y_i f(x_i) > t_i \\ I_t = \{i \mid 0 < \gamma_i < C_i\} & \text{such that } y_i f(x_i) = t_i \\ I_C = \{i \mid \gamma_i = C_i\} & \text{such that } \tau_i \leq y_i f(x_i) \leq t_i \\ I_\tau = \{i \mid C_i < \gamma_i = C_i + D\} & \text{such that } y_i f(x_i) = \tau_i \\ I_D = \{i \mid \gamma_i = C_i + D\} & \text{such that } y_i f(x_i) < \tau_i \end{cases} \tag{9}$$

With this partitioning, we need only to compute γ_i for the samples indexed in $I_T = I_t \cup I_\tau$, which means that the dual formulation of the optimization problem can be

transformed into the following form

$$\begin{array}{l} \min\limits_{0\leq\gamma_i\leq C_i+D,\gamma_i\neq C_i} \frac{1}{2}\sum\limits_{i,j\in I_T}\gamma_i\gamma_j G_{ij} - \sum\limits_{i\in I_T}\gamma_i s_i \\ \text{subject to } \sum\limits_{i\in I_T} y_i\gamma_i + \sum\limits_{i\in I_C} C_i y_i + \sum\limits_{i\in I_D}(C_i+D)y_i = 0 \end{array} \tag{10}$$

where $s_i = t_i - \sum_{j\in I_C} C_j G_{ji} - \sum_{j\in I_D}(C_j + D)G_{ji}$ for $i \in I_t$ and $s_i = \tau_i - \sum_{j\in I_C} C_j G_{ji} - \sum_{j\in I_D}(C_j + D)G_{ji}$ for $i \in I_\tau$. We can therefore solve this problem by solving the following linear system

$$\begin{cases} \sum\limits_{j\in I_T} G_{ij}\gamma_j + y_i\zeta = s_i, \text{ for } i \in I_T \\ \sum\limits_{i\in I_T} y_i\gamma_i = -\sum\limits_{i\in I_C} C_i y_i - \sum\limits_{i\in I_D}(C_i+D)y_i \end{cases} \tag{11}$$

When the optimum function $f(\cdot)$ is obtained, the following decision function is used to determine the classification:

$$decision(x) = \begin{cases} +1 & \text{if } f(x) \geq f_+ \\ 0 & \text{if } f_- < f(x) < f_+ \\ -1 & \text{if } f(x) \leq f_- \end{cases} \tag{12}$$

where $f_+ = \log(p_+/(1-p_+))$ and $f_- = \log(p_-/(1-p_-))$.

This variant is denoted in this paper by QPSVM for Quadratic Programming SVM.

4 Incremental SVM with active sample selection

Since sampling is expensive in general, the initial number of samples may be too small. In order to increase the accuracy of the prediction, after each estimate, we want to find a best location to be used for the next sampling, where best should be assessed from the information provided by this estimate so far. By sampling at this location, we get the new data (geographic location, measure), allowing to augment the available dataset for training. From this new data, we adapt the SVM model accordingly. The Iteration of these steps stops when the amount of information is rich enough for the problem at hand, namely the estimation of the pollution in the area of interest.

Therefore, the two main ingredients to enhance the model precision are:

- The selection of the next location to be sampled in an optimal way, based on the information provided by the current estimate.
- The training of the SVM with an uncertainty option presented in Section 3 in an incremental way, namely adapting the model with each new sample.

We address these two ingredients in this section and illustrate the proposed approach in Fig. 3.

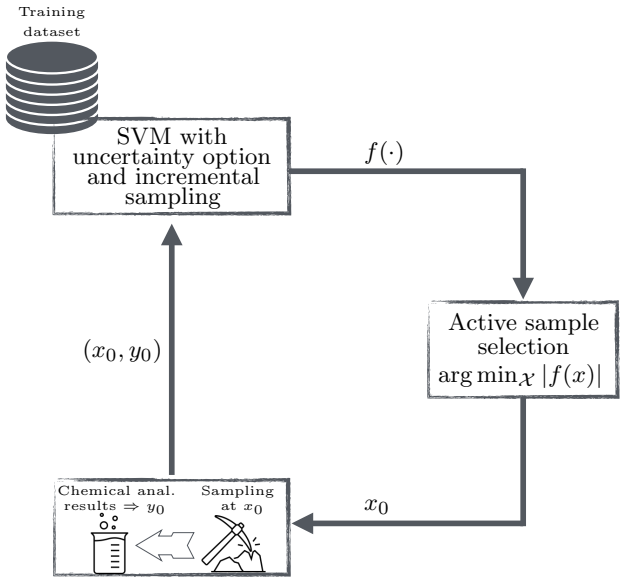

Fig. 3 Schematic illustration of the proposed incremental SVM with active sample selection. Initially, the proposed SVM with uncertainty option described in Section 3 is trained on an initial training dataset. And then, the active sample selection (13) determines the next sampling location x_0, with optimality in the sense of minimizing the uncertainty. The label of x_0 is then determined (e.g., for the soil assessment application, a soil sample is extracted from the location x_0 and chemical analyses are conducted to measure its pollution concentration y_0). The new data (x_0, y_0) is fed to the incremental algorithm described in Section 4.2.

4.1 Choice of the next sample location

We formulate the problem of estimating the location of the next sampling as follows. We aim to find the next location, denoted the following x_0, that leads to the "most modification" on the model if this novel information is included.

This problem is essentially related to active learning, which is within the human-in-the-loop concept. Active learning has been largely investigated in the literature (Settles, 1995), including some recent advances (Yoo and Kweon, 2019; Ren et al., 2021). In active learning, the learning algorithm can interactively ask the user to label new data points with real labels. The basic idea is that if a machine learning algorithm is allowed to select the data it wants to learn, it can achieve greater accuracy while using less training data. Different approaches have been proposed in the literature to address active learning (Yoo and Kweon, 2019; Ren et al., 2021). For instance, the samples that are the closest to the decision frontier are selected by Jan Kremer and Igel (2014), while the samples that are the farthest from the class are chosen by Huang and Lin (2016), and the entropy measure is used by Jing et al. (2004).

Considering our Machine Learning model, which is SVM with an uncertainty option, we propose to consider the choice of the sample that reduces the uncertainty.

Therefore, we query the sample x_0 that has the smallest absolute estimate, namely.

$$x_0 = \arg\min_{x \in \mathcal{X}} |f(x)|. \tag{13}$$

It turns out that this criterion is similar to the one proposed by Jan Kremer and Igel (2014) for binary classification SVM and is called simple margin.

4.2 Incremental algorithm for SVM with an uncertainty option

Once the new location is determined and its corresponding sample obtained, we aim to integrate this new data within this model, by adapting it in an incremental way, namely operating the updates of the model when having a novel sample at each instance.

To this end, we update the classifier by examining in detail the Karush-Kuhn-Tucker (KKT) conditions. Our approach follows the same idea of Cauwenberghs and Poggio (2000) proposed for conventional SVM (see also Karasuyama and Takeuchi (2009) and Laskov et al. (2006) for a survey), and we extend it to the proposed SVM with an uncertainty option. In the following, we derive the expressions for the QPSVM.

4.2.1 The KKT conditions

Using the stationarity condition and the KKT complementarity condition of the dual formulation (8) of QPSVM, namely the minimization of $W = \frac{1}{2}\gamma^\top G\gamma - \tau^\top\gamma - (t-\tau)^\top\alpha + \zeta y^\top\gamma$, we obtain the following conditions:

$$\begin{aligned}
\frac{\partial W}{\partial \gamma_i} &= \sum_j G_{ij}\gamma_j + y_i\zeta - \tau_i = y_i(f(x_i)) - \tau_i \\
\frac{\partial W}{\partial \alpha_i} &= t_i - \tau_i \\
\frac{\partial W}{\partial \zeta} &= \sum_j y_j\gamma_j = 0
\end{aligned}$$

In the following, we denote $g_i = \frac{\partial W}{\partial \gamma_i}$. When having a novel sample defined by (x_0, y_0), its addition to the training data leads to a modification of the coefficients γ. Let $\Delta\gamma_0$ be the incremental modification. Then to satisfy the KKT conditions, the coefficients must be updated as follows:

$$\begin{cases} \Delta g_i = G_{ic}\Delta\gamma_0 + \sum_j G_{ij}\Delta\gamma_j + y_i\Delta\zeta \quad \text{for all } i \in D \cup \{c\} \\ 0 = y_0\Delta\gamma_0 + \sum_j y_j\Delta\gamma_j \end{cases} \tag{14}$$

Following the approach given by Cauwenberghs and Poggio (2000), we can conclude that the new sample added to the QPSVM with an uncertainty option must satisfy the following conditions.

For the set $I_T = I_t \cup I_\tau$, $g_i = 0$, we have for all $i \in I_T$:

$$\underbrace{\begin{bmatrix} 0 & y_{s_1} & \cdots & y_{s_{l_{I_T}}} \\ y_{s_1} & G_{s_1 s_1} & \cdots & G_{s_1 s_{l_{I_T}}} \\ \vdots & \vdots & \cdots & \vdots \\ y_{s_{l_{I_T}}} & G_{s_{l_{I_T}} s_1} & \cdots & G_{s_{l_{I_T}} s_{l_{I_T}}} \end{bmatrix}}_{\mathbf{G}} \begin{bmatrix} \Delta\zeta \\ \Delta\gamma_{s_1} \\ \vdots \\ \Delta\gamma_{s_{l_{I_T}}} \end{bmatrix} = -\begin{bmatrix} y_0 \\ G_{s_1 c} \\ \vdots \\ G_{s_{l_{I_T}} c} \end{bmatrix} \Delta\gamma_0$$

For all the training data, we have

$$\begin{cases} \Delta\zeta = \beta\Delta\gamma_0 \\ \Delta\gamma_j = \beta_j\Delta\gamma_0 & \text{for all } j \in D \\ \Delta g_j = \theta_j\Delta\gamma_0 & \text{for all } j \in D \cup \{c\} \end{cases} \tag{15}$$

For the entries of I_T, the vector $[\beta, \beta_{s_1}, \cdots, \beta_{s_{l_{I_T}}}]^T$ can be computed by

$$\begin{bmatrix} \beta \\ \beta_{s_1} \\ \vdots \\ \beta_{s_{l_{I_T}}} \end{bmatrix} = -\mathbf{R} \begin{bmatrix} y_0 \\ G_{s_1 c} \\ \vdots \\ G_{s_{l_{I_T}} c} \end{bmatrix}$$

with $\mathbf{R} = \mathbf{G}^{-1}$ and

$$\theta_i = \begin{cases} G_{ic} + \sum_{j \in S} G_{ij}\beta_j + y_i\beta & \text{for all } i \notin I_T \\ 0 & \text{for all } i \in I_T \end{cases} \tag{16}$$

Once the entries of I_T are updated, the matrix $\mathbf{R}$ should be updated. To add an entry in $\mathbf{R}$, we operate as follows

$$\mathbf{R} \leftarrow \begin{bmatrix} & & 0 \\ & \mathbf{R} & \vdots \\ & & 0 \\ 0 & \cdots\ 0 & 0 \end{bmatrix} + \frac{1}{\theta_0} \begin{bmatrix} \beta \\ \beta_{s_1} \\ \vdots \\ \beta_{sl_{I_T}} \\ 1 \end{bmatrix} \begin{bmatrix} \beta & \beta_{s_1} & \cdots & \beta_{sl_{I_T}} & 1 \end{bmatrix}$$

To remove the k-th entry in $\mathbf{R}$, we operate as follows:

$$\mathbf{R}_{ij} \leftarrow \mathbf{R}_{ij} - \mathbf{R}_{kk}^{-1}\mathbf{R}_{ik}\mathbf{R}_{kj},$$

for all $i, j \in I_T \cup \{0\}$ and $i, j \neq k$.

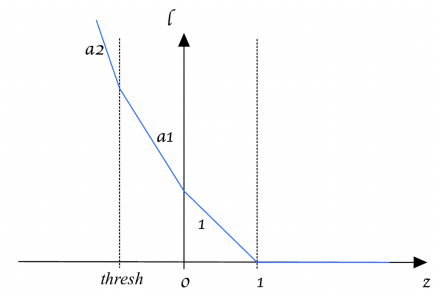

Fig. 4 The proposed triple-hinge loss

5 Extension to a multi-level estimation

While in the parts of the paper we have considered three levels of assessment, namely "contaminated vs uncertainty vs non-contaminated", we provide in this section an extension to a more fine-grained assessment with five levels, namely "very high vs high vs moderate vs light vs very light contaminated". To this end, we define the proper triple-hinge loss and derive the corresponding optimization problem and its resolution.

5.1 Triple-hinge loss

The study of double-hinge loss of Wegkamp and Yuan (2011) has inspired us to extend this work to a triple-hinge loss. By using the concept of Section 3.1, we define the triple-hinge loss as follows:

$$\mathcal{L}_{\text{3-hinge}}(z) = \begin{cases} 1 + (a_2 - a_1)thresh - a_2 z & \text{if } z \leq thresh \\ 1 - a_1 z & \text{if } thresh \leq z \leq 0 \\ 1 - z & \text{if } 0 \leq z \leq 1 \\ 0 & \text{if } z > 1 \end{cases} \tag{17}$$

The triple-hinge loss is illustrated in Fig. 4 with its parameters a_1, a_2 et $thresh$. With this definition of the loss function, the decision function can be written in the following form:

$$decision(x) = \begin{cases} -1 & \text{if } f(x) \leq b_1 \\ -0.5 & \text{if } b_1 < f(x) < b_2 \\ 0 & \text{if } b_2 \leq f(x) \leq 1 - b_2 \\ 0.5 & \text{if } 1 - b_2 < f(x) < 1 - b_1 \\ 1 & \text{if } f(x) \geq 1 - b_2 \end{cases}$$

where the labels 1, 0.5, 0, -0.5 and -1 are the very high, high, moderate, light and very light contamination classes respectively. The relationship between a_1, a_2, $thresh$, b_1 and b_2 is $a_1 = (1 - b_1)/b_1$ and $a_2 = (1 - b_2 - a_1 thresh)/(b_2 - thresh)$.

Algorithm 2 Implementation of the LPSVM with the triple-hinge loss in Python

```
import cvxpy as cp
# The variables
varl = cp.Variable(n+m)
var2 = cp.Variable(m)
# The optimization problem
A = np.ones(ntm)
A[n:n+m] = A[n:n+m]*r
objective = cp.Minimize(A @ varl)
# The constraints
constraints = [np.zeros(n) <= var1[:n]]
constraints += [np.ones(n) - y @ (var2 @ f) <= varl[ :n]]
constraints += [np.ones(n) - al * y @ (var2 @ f) <= var1[ :n] ]
constraints += [np.ones(n) * (1 + a2 * s- al * s) - a2 * y @ (var2 @ f) <=
    varl[ :n]]
constraints += [var2 <= varl[n:], -var2 <= var1[n: ]]
# The problem resolution
prob = cp.Problem(objective, constraints)
prob.solve(qcp=True)
res = var2.value
```

5.2 Optimization problem and resolution

By applying the triple-hinge loss, the optimization problem can be written as follows, where slack variables are used:

$$\min_{\lambda,\xi} \xi_1 + \cdots + \xi_n + \rho\,(\xi_{n+1} + \cdots + \xi_{2n}) \tag{18}$$

$$\text{subject to} \begin{cases} \xi_i \geq 0 & \text{for } i = 1, \cdots, n \\ \xi_i \geq 1 - y_i h_i & \text{for } i = 1, \cdots, n \\ \xi_i \geq 1 - a_1 y_i h_i & \text{for } i = 1, \cdots, n \\ \xi_i \geq 1 + (a_2 - a_1) thresh - a_2 y_i h_i & \text{for } i = 1, \cdots, n \\ h_i = \sum_{j=1}^{n} \lambda_j\, \kappa(x_i, x_j) & \text{for } i = 1, \cdots, n \\ \xi_{n+i} \geq \lambda_j & \text{for } i = 1, \cdots, n \\ \xi_{n+i} \geq -\lambda_j & \text{for } i = 1, \cdots, n \end{cases} \tag{19}$$

In order to understand this constrained optimization problem (18)-(19) resulting from the triple-hinge loss (17), we compare it to the optimization problem (4)-(5) obtained from the double-hinge loss (3). It is easy to see that the only difference is the novel fourth constraint, which is a linear inequality constraint. This is where the piece-wise addition to the triple-hinge loss comes into play, by refining the classification in adding 2 classes for a total of 5, compared to the 3 classes with the double-hinge loss. This result can be extended to a multiple-hinge loss, where each additional piece-wise addition in the hinge induces an additional linear inequality constraint.

To solve this constrained optimization problem, we use the same techniques used in Section 3.1 of LPSVM with an uncertainty option, where the added constraint can be easily integrated in the linear programming with solvers like CPLEX and Gurobi Optimizer (e.g. `cvxpy` in Python). A typical implementation is given in Algorithm 2.

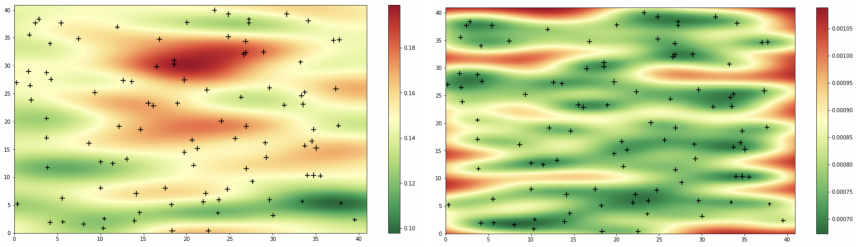

Fig. 5 Illustration of the kriging results on the 2D data, with the estimation map (left) and the uncertainty map (right).

6 Experimental results

In order to assess the relevance of the proposed methodology and the derived methods, we evaluate this work on a dataset in two dimensions of the porosity measure using 200 wells (i.e., sampling points)[3]. It is worth noting that oil contamination affects soil porosity, as oil tends to force soil particles together, thereby decreasing porosity (Ndimele et al., 2018; Zhang et al., 2019)

As a baseline, we investigate the ordinary kriging model. Fig. 5 shows the estimated map and the uncertainty map, the latter is computed from the standard deviation estimation on each location.

6.1 On the limits of kriging

Since the cost of sampling is relatively high in geostatistics, one may aim to be able to further refine the location of the boundaries of the different regions. To this end, it would be ideal that the largest uncertainties appear at the frontier of the regions, thus guiding us to sample at these locations. However, the kriging does not allow this properly, since both estimation map and uncertainty map are not "conditional" to one another.

We illustrate this limit of kriging in Fig. 6. On one hand, if we examine the estimation map in order to have new samples at the boundaries, such as marker ■ in the left figure (at the limit value of 0.15), the uncertainty of such a point is low as given in the right figure. On the other hand, if we use the uncertainty map (right figure) to find the point with the highest uncertainty value, such as marker ▲, it turns out that such a point lies in the middle of the green zone of the estimation map (left figure); Therefore, we can be almost certain that this point is one of low contamination and thus less interesting to sample. These two examples show that kriging and its variogram do not give the information we need, because the uncertainty map only gives us spatial uncertainties and does not take into account the pollution concentration at each location.

[3] https://github.com/GeostatsGuy/GeoDataSets/blob/master/2D MV 200wells.csv

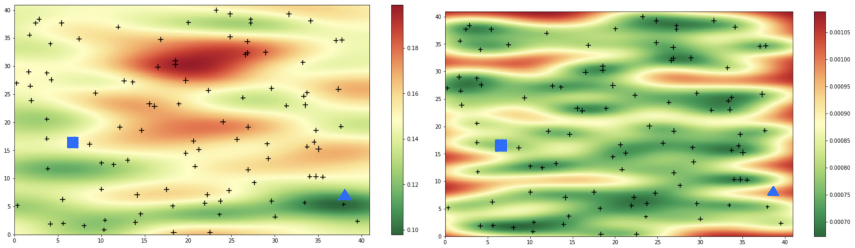

Fig. 6 Same figures as in Fig. 5 with marker ■ showing a boundary limit and marker ▲ showing a large uncertainty.

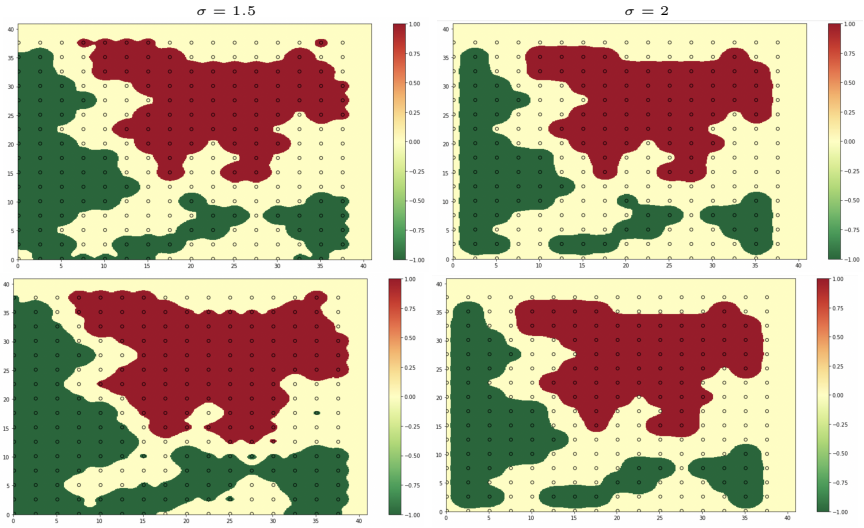

Fig. 7 Classification maps of LPSVM with $\sigma = 1.5$ (first column) or $\sigma = 2$ (second column), and $b = 0.4$ (first row) or $b = 0.45$ (second row).

6.2 Results using the proposed methods

To overcome these limits, we should be more concerned with classifying different regions on the map and treating points between two regions as having higher uncertainty, as proposed in Section 3 with SVM with an uncertainty option. We therefore have 3 target values. We set high pollution to $+1$ for higher contamination density (color red in figures), low pollution to -1 for lower contamination density (green color in figures), and 0 for the uncertainty region (yellow).

First of all, we examine the influence of the hyperparameters on the results. For both LPSVM and QPSVM, we can see in Fig. 7 and 8 that as the value of bandwidth parameter σ increases, the range of influence of each point widens. For LPSVM, as the value of b decreases, the area of the uncertainty class becomes smaller. For QPSVM, where we set $c_+ = c_- = 1$, we can see that, as the values of r_+ and r_- decrease, the area of the uncertainty class becomes larger. In practice, we can adjust these parameters accordingly to our real needs.

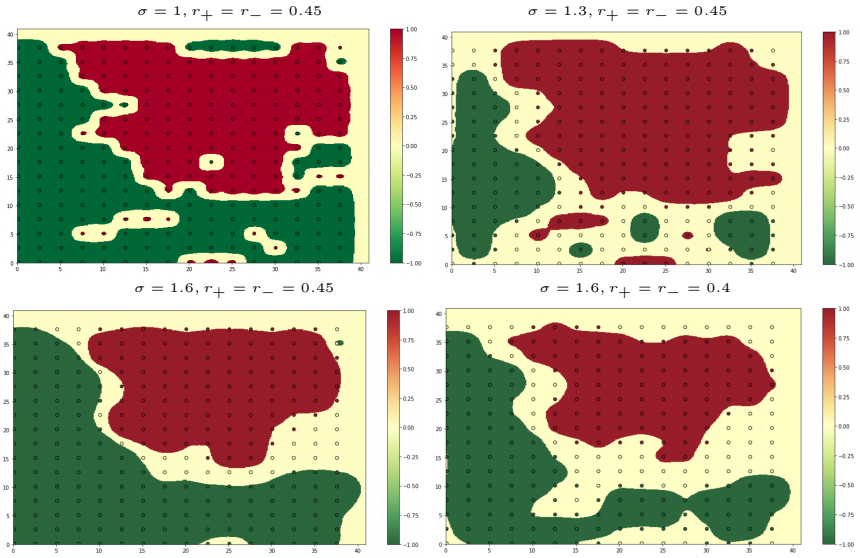

Fig. 8 Classification maps of QPSVM with several values of bandwidth parameter σ and (r_+, r_-).

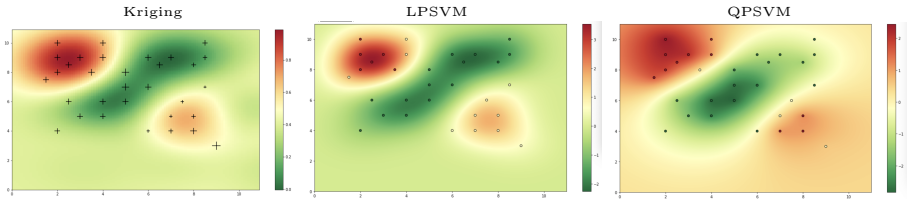

Fig. 9 The results of Kriging interpolation (left), LPSVM (middle) and QPSVM (right)

6.3 Assessing the quality of interpolation using synthetic data

To confirm that LPSVM and QPSVM can achieve similar performance to Kriging in terms of interpolation results, we generated a 10-by-10 map with values following a Gaussian distribution, and selected 31 random samples in the map as the training dataset. We use this dataset to compare the results of the proposed methods with kriging interpolation.

From Fig. 9, we can clearly see that the interpolation results of the two variants of SVM behave in the same way as the kriging results, while at the same time our SVM is able to take into account the uncertainty class and compute the boundary equations for each class.

6.4 Comparative analysis with different classifiers

In order to provide a comparative analysis with several classifiers, we operate as follows. We use the 2D dataset of porosity with 200 wells and use the kriging to interpolate the whole region. For the training dataset, we generated our training dataset using a regular-grid sampling on the contamination distribution map obtained using

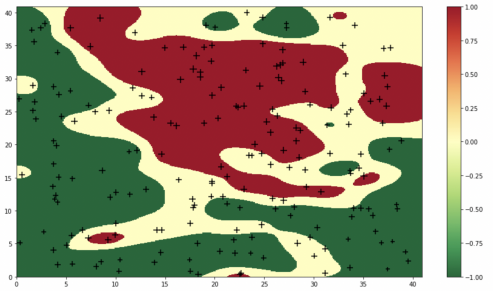

Fig. 10 The target classification map

kriging. For the evaluation dataset, we took 100 random points on the contamination distribution map produced using kriging. We treat sample points with a degree of contamination equal to or greater than 1.5 as contaminated and the others as uncontaminated. The resulting map is given in Fig. 10.

For comparative analysis, we considered Random Forest (RF), AdaBoost (ADA) and Gradient Boosting with Decision Tree (GBDT), where the number of trees was 500 and a maximum depth of 2. In order to provide a fair comparison with other classifiers that are binary classifiers but without an uncertainty class, we need to design a setting to use them in our context of two-class plus an uncertainty class. To this end, inspired by the one-versus-rest procedure in multiclass classification, we operate as follows. We divide the training dataset into two categories according to the contamination value of each sample using two strategies. The first strategy seeks to discriminate the contaminated data versus all the other (i.e., uncertain and uncontaminated data), and the second strategy to discriminate uncontaminated data versus all the rest (i.e., uncertain and contaminated data). By applying these two strategies to each traditional classifier, we get two sets of labels. We combine the results, with samples where predicted labels differ are treated as uncertain samples. Lastly, the final accuracy and uncertainty rate are computed.

As given in Table 1, we can see that the classification accuracy of both LPSVM and QPSVM is higher than all the conventional classifiers. Moreover, the proposed methods provide lower uncertainty rates compared to the other methods, with 2 to 3 folds if we compare LPSVM with all conventional methods. These results demonstrate that the proposed methodology provides algorithms that are able to meet the expected requirements.

6.5 Assessing incremental learning with sample selection

In this section, we assess the sample selection and incremental learning algorithm of QPSVM with an uncertainty option. To this end, we consider the same real 2D data of 200 wells and use kriging to interpolate the whole region in order to constitute the map regarded as groundtruth, as described in Section 6.4 and illustrated in Fig. 10.

As in practical geoscience sampling, samples are selected on a regular grid. Consequently, when selecting the next sampling, we choose the location with the greatest uncertainty among all the points of the regular grid, following the discussion conducted

Table 1 Comparative analysis

Model	Accuracy	Uncertainty Rate
LPSVM (this paper)	97%	10%
QPSVM (this paper)	92%	13%
RF	72%	21%
ADA	82%	28%
GBDT	88%	29%

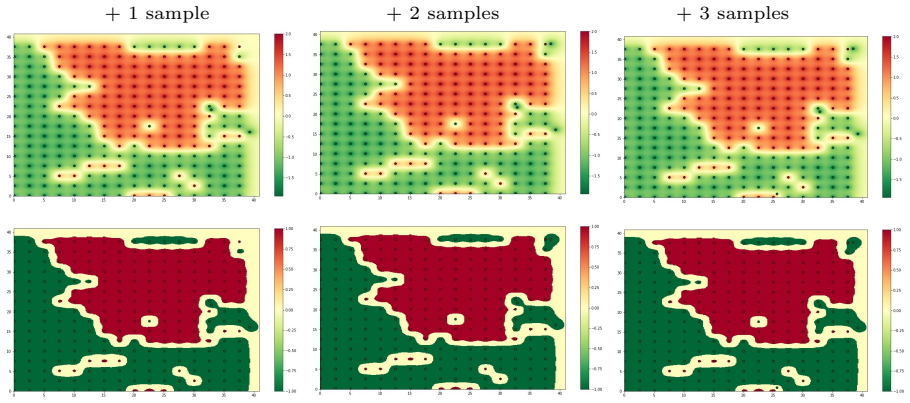

Fig. 11 The evolution of the estimated maps (estimation in the upper row, classification in the lower row) after adding 1, 2 and 3 samples

Table 2 Change rate between two consecutive iterations

Iteration $1 \to 2$	Iteration $2 \to 3$	Iteration $3 \to 4$
0.59%	0.50%	0%

in Section 4.1. The final mathematical expressed mathematically becomes

$$x_0 = \arg \min_{x \in Grid} |f(x)|,$$

where *Grid* is the set of sampleable locations not sampled so far. This sampling is iterated with the incremental update described in Section 4.2. As a stopping criterion, we consider the stability of the estimation.

For these experiments, we set $\sigma = 1$, $r_+ = r_- = 0.45$, and $c_+ = c_- = 1$ (see Section 6.2 for the influence of these parameters). The evolution of the estimation map, as well as the classification map, is shown in Fig. 11 for four consecutive iterations, illustrating the stability of the map throughout iterations. the change rate reduces with iterations, as given in Table 2

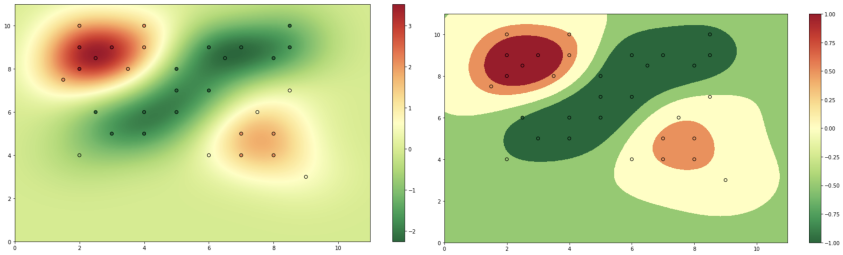

Fig. 12 The estimation map (left) and classification map (right) obtained by LPSVM with the triple-hinge loss

6.6 Increasing the assessment levels

As in the work process of soil contamination assessment, one might be interested in assessing different areas of the soil for the level of contamination. In this section, we evaluate the method proposed in Section 5, considering the classification of the soil into five levels of contamination: very high, high, moderate, light and very light.

By considering the same data as in the previous sections, we set $thresh = -1$, $b_1 = 0.25$, $b_2 = 0.4$ and $\sigma = 2$. The obtained results are given in Fig. 12 in terms of estimation map and classification map. Depending on the application at hand, the user can define one of these levels as "uncertain", allowing to refine-and-reduce it by integrating active sample selection and incremental learning in the process, as proposed in Section 4. This part is beyond the scope of this paper.

7 Experiments on the Meuse river dataset

In this section, we demonstrate the relevance of the proposed methods on another real well-known dataset: the Meuse river dataset, available from the `R sp package`[4] (Middelkoop, 2000; Bivand et al., 2008). The dataset was collected in the Meuse river floodplains west of the town Stein, southeastern Netherland. It consists of topsoil of 4 heavy metal concentrations and organic matter (OM) at different locations, as well as several soil and landscape features at the observation locations. These concentrations, in ppm (parts per million), are bulk sampled from an area of approximately 15 m $\times$ 15 m. The dataset has 156 samples from different locations of cadmium, copper, lead, zinc and OM concentrations. The distributions of all the heavy metal and OM concentrations are shown in Fig. 13, as well their spatial distributions in Fig. 14, illustrating their diversities and the difficulties in addressing such non-Gaussian distributions with nonlinear variabilities.

As opposed to the dataset studied in Section 6 where samples are scattered in all the 2D space under study, the current dataset has samples from the river floodplains with pollution on both sides of the river; Therefore, an increased uncertainty area is expected. Fig. 15 illustrates the classification maps generated by QPSVM using different parameters for all 4 heavy metals and OM. These results demonstrate the uncertainty region (yellow color in figures) beyond the river and within the river floodplains between the higher (red) and low (green) levels of contamination. These

[4] https://cran.r-project.org/web/packages/sp/

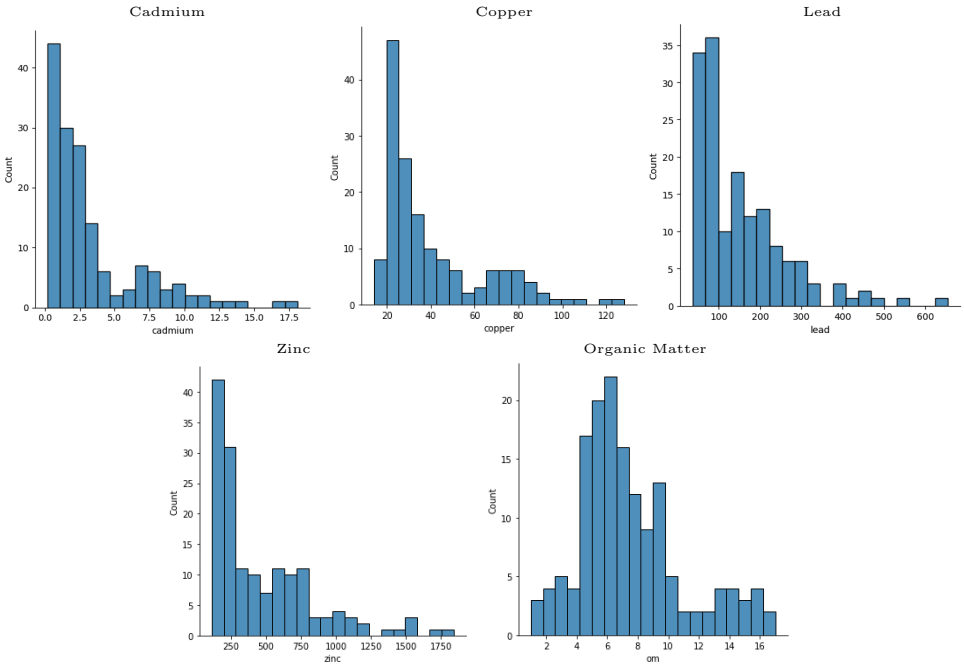

Fig. 13 Distribution of the heavy metal and OM concentrations

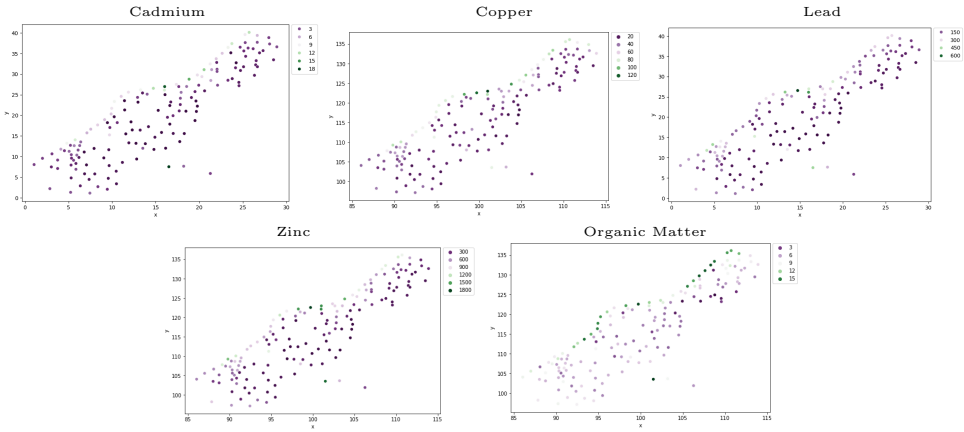

Fig. 14 Locations of samples and the spatial distribution of the heavy metal and OM concentrations

results illustrate the impact of the bandwidth parameter on the spatial granularity of the results. Similar results can be drawn from LPSVM, omitted here due to space limit.

Finally, we study the performance of the proposed methods LPSVM and QPSVM. To this end, we compare them to the different ML methods given in the previous section, and use the same setting used in that section. Table 3 presents the obtained results on all 4 heavy metal and OM concentrations, demonstrating that both proposed

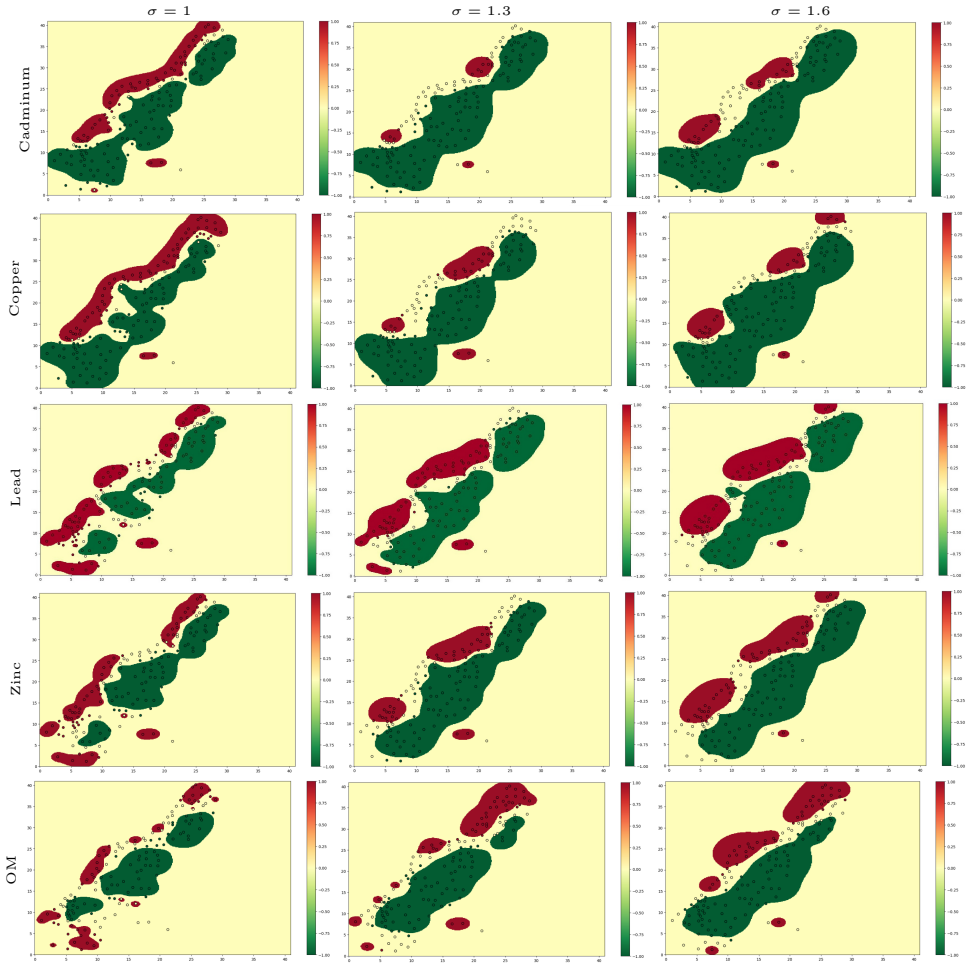

Fig. 15 Classification maps of QPSVM with several values of bandwidth parameter σ with $r_+ = r_- = 0.45$, for the Cadmium (first row), Copper (second row), Lead (third row), Zinc (fourth row), and OM (last row).

methods LPSVM and QPSVM outperform the other methods from the literature. Moreover, LPSVM slightly outperforms QPSVM. All these results corroborate the results obtained in Section 6 on the 200-well dataset, demonstrating the relevance of the proposed methods.

8 Conclusion and future work

This paper proposed an original way to overcome some drawbacks of the kriging, by investigating SVM with an uncertainty option. As this is essentially related to the literature of the so-called SVM with reject option, we explored two variants leading to LPSVM and QPSVM. Moreover, we explored the concept of active sampling, which

Table 3 Comparative analysis in terms of accuracy of different methods for the 4 heavy metal and OM

Method	Ca	Cu	Zn	Pd	OM
LPSVM (this paper)	92%	99%	99%	99%	99%
QPSVM (this paper)	91%	93%	97%	98%	99%
RF	86%	88%	90%	90%	86%
ADA	83%	88%	92%	86%	85%
GBDT	82%	88%	90%	90%	86%

turned out to be relevant for our proposed methodology, as we can select the sample that allows to reduce the uncertainty. Having a sampling selection, we designed a relevant incremental learning algorithm for the SVM with an uncertainty option. Finally, we demonstrated that it is easy to extend the SVM with an uncertainty option to a multi-level estimation, by introducing a triple-hinge loss and deriving the corresponding optimization problem and resulting algorithm. We conducted extensive experiments that demonstrated the relevance of these methodological and algorithmic developments on different real data. As of future work, one could be interested in defining a multi-hinge loss, beyond the triple hinge proposed in this paper. It is worth noting that this extension seems straightforward, as explained in this paper. Integrating sample selection and incremental learning for such multi-hinge loss seems also to be straightforward.

Acknowledgments. The authors would like to thank the agency of ecological transition ADEME in France for the funding of this research work.

References

Chilès, J.-P., Desassis, N.: In: Daya Sagar, B.S., Cheng, Q., Agterberg, F. (eds.) Fifty years of kriging, pp. 589–612. Springer, Cham (2018)

Dramsch, J.S.: 70 years of machine learning in geoscience in review. Advances in geophysics **61**, 1–55 (2020)

McLean, M., Evers, L., Bowman, A., Bonte, M., Jones, W.: Statistical modelling of groundwater contamination monitoring data: A comparison of spatial and spatiotemporal methods. Science of The Total Environment **652**, 1339–1346 (2019)

Sun, X.-L., Wu, Y.-J., Zhang, C., Wang, H.-L.: Performance of median kriging with robust estimators of the variogram in outlier identification and spatial prediction for soil pollution at a field scale. Science of the Total Environment **666**, 902–914 (2019)

Ouabo, R.E., Sangodoyin, A.Y., Ogundiran, M.B.: Assessment of ordinary kriging and inverse distance weighting methods for modeling chromium and cadmium soil pollution in e-waste sites in douala, cameroon. Journal of Health and Pollution **10**(26), 200605 (2020)

Rakotonirina, H., Honeine, P., Atteia, O., Exem, A.V.: Spatial interpolation and conditional map generation using deep image prior for environmental applications. Mathematical Geoscience **56**, 949–974 (2024)

Rakotonirina, H., Honeine, P., Atteia, O., Exem, A.V.: Estimating contaminated soil volumes using a generative neural network: A hydrocarbon case in france. In: Proc. 15th International Conference on Geostatistics for Environmental Applications (geoENV). Springer, Chania, Crete, Greece (2024)

Pyrcz, M.J., Deutsch, C.V.: Geostatistical Reservoir Modeling. Oxford University Press, USA (2014)

Liu, W., Ikonnikova, S., Scott Hamlin, H., Sivila, L., Pyrcz, M.J.: Demonstration and mitigation of spatial sampling bias for machine-learning predictions. SPE Reservoir Evaluation & Engineering **24**(01), 262–274 (2021)

Thomas, A.T., Harten, J., Jusri, T., Reiche, S., Wellmann, F.: An integrated modeling scheme for characterizing 3d hydrogeological heterogeneity of the new jersey shelf. Marine Geophysical Research **43**(2), 11 (2022)

Hüllermeier, E., Waegeman, W.: Aleatoric and epistemic uncertainty in machine learning: An introduction to concepts and methods. Machine learning **110**(3), 457–506 (2021)

Gawlikowski, J., Tassi, C.R.N., Ali, M., Lee, J., Humt, M., Feng, J., Kruspe, A., Triebel, R., Jung, P., Roscher, R., *et al.*: A survey of uncertainty in deep neural networks. Artificial Intelligence Review **56**(Suppl 1), 1513–1589 (2023)

Hüllermeier, E., Brinker, K.: Learning valued preference structures for solving classification problems. Fuzzy Sets and Systems **159**(18), 2337–2352 (2008)

Psaros, A.F., Meng, X., Zou, Z., Guo, L., Karniadakis, G.E.: Uncertainty quantification in scientific machine learning: Methods, metrics, and comparisons. Journal of Computational Physics **477**, 111902 (2023)

Loquin, K., Dubois, D.: In: Jeansoulin, R., Papini, O., Prade, H., Schockaert, S. (eds.) Kriging and epistemic uncertainty: a critical discussion, pp. 269–305. Springer, Berlin, Heidelberg (2010)

Fuhg, J.N., Fau, A., Nackenhorst, U.: State-of-the-art and comparative review of adaptive sampling methods for kriging. Archives of Computational Methods in Engineering **28**, 2689–2747 (2021)

Campbell, C., Ying, Y.: Learning with Support Vector Machines. Springer, Switzerland (2022)

Pisner, D.A., Schnyer, D.M.: Support vector machine. In: Mechelli, A., Vieira, S. (eds.)

Machine Learning, pp. 101–121. Academic Press, (2020). Chap. 6

Cervantes, J., Garcia-Lamont, F., Rodríguez-Mazahua, L., Lopez, A.: A comprehensive survey on support vector machine classification: Applications, challenges and trends. Neurocomputing **408**, 189–215 (2020)

Hu, K., Jiang, H., Ji, C.-G., Pan, Z.: A modified butterfly optimization algorithm: An adaptive algorithm for global optimization and the support vector machine. Expert Systems **38**(3), 12642 (2021)

Menaka, D., Ganesh Vaidyanathan, S.: A hybrid convolutional neural network-support vector machine architecture for classification of super-resolution enhanced chromosome images. Expert Systems **40**(3), 13186 (2023)

De Caires, S.A., Keshavarzi, A., Bottega, E.L., Kaya, F.: Towards site-specific management of soil organic carbon: Comparing support vector machine and ordinary kriging approaches based on pedo-geomorphometric factors. Computers and Electronics in Agriculture **216**, 108545 (2024)

Chen, L., Ren, C., Zhang, B., Wang, Z.: Multi-sensor prediction of stand volume by a hybrid model of support vector machine for regression kriging. Forests **11**(3), 296 (2020)

Wu, X., Lin, Q., Lin, W., Ye, Y., Zhu, Q., Leung, V.C.: A kriging model-based evolutionary algorithm with support vector machine for dynamic multimodal optimization. Engineering Applications of Artificial Intelligence **122**, 106039 (2023)

Leong, W., Kelani, R., Ahmad, Z.: Prediction of air pollution index (API) using support vector machine (SVM). Journal of Environmental Chemical Engineering **8**(3), 103208 (2020)

Jha, S.K., Yoon, T.H.: Toxicity modelling of nanomaterials by origin evaluation of their physicochemical descriptors using a combination of principal component analysis and support vector machine methods. Expert Systems **37**(2), 12492 (2020)

Huang, H., Huang, S., Du, Q.: Evaluation of soil heavy metal pollution based on k-means and SVM. International Journal of Environmental Science and Technology **20**(11), 12015–12024 (2023)

Grandvalet, Y., Rakotomamonjy, A., Keshet, J., Canu, S.: Support vector machines with a reject option. In: Koller, D., Schuurmans, D., Bengio, Y., Bottou, L. (eds.) Advances in Neural Information Processing Systems, vol. 21. Curran Associates, Inc., Red Hook, NY, USA (2008)

Wegkamp, M., Yuan, M.: Support vector machines with a reject option. Bernoulli, 17(4), 1368–1385 (2011)

Hanczar, B., Sebag, M.: Combination of one-class support vector machines for classification with reject option. In: Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2014, Nancy, France, September 15-19, 2014. Proceedings, Part I 14, pp. 547–562 (2014). Springer

Franc, V., Prusa, D., Voracek, V.: Optimal strategies for reject option classifiers. Journal of Machine Learning Research **24**(11), 1–49 (2023)

Pyrcz, M., Gringarten, E., Frykman, P., Deutsch, C.: Representative input parameters for geostatistical simulation. Stochastic Modeling and Geostatistics: Principles, Methods, and Case Studies, Vol. II, AAPG Computer Applications in Geology 5 **5**, 123 (2005)

Unser, M.: A unifying representer theorem for inverse problems and machine learning. Foundations of Computational Mathematics **21**(4), 941–960 (2021)

Vishwanathan, S.V.M., Murty, M.N.: SSVM: a simple SVM algorithm. In: Proceedings of the International Joint Conference on Neural Networks (IJCNN), vol. 3, pp. 2393–2398. IEEE, Honolulu, HI, USA (2002)

Settles, B.: Active learning literature survey. Science **10**(3), 237–304 (1995)

Yoo, D., Kweon, I.S.: Learning loss for active learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, pp. 93–102 (2019)

Ren, P., Xiao, Y., Chang, X., Huang, P.-Y., Li, Z., Gupta, B.B., Chen, X., Wang, X.: A survey of deep active learning. ACM computing surveys (CSUR) **54**(9), 1–40 (2021)

Jan Kremer, K.S.P., Igel, C.: Active learning with support vector machines. Data Mining and Knowledge Discovery **4**, 269–340 (2014)

Huang, K.-H., Lin, H.-T.: A novel uncertainty sampling algorithm for cost-sensitive multiclass active learning. In: 16th IEEE International Conference on Data Mining (ICDM), Barcelona, Spain, pp. 925–930 (2016)

Jing, F., Li, M., Zhang, H.-J., Zhang, B.: Entropy-based active learning with support vector machines for content-based image retrieval. In: IEEE International Conference on Multimedia and Expo (ICME), vol. 1. Taipei, Taiwan, pp. 85–88 (2004)

Cauwenberghs, G., Poggio, T.: Incremental and decremental support vector machine learning. In: Leen, T., Dietterich, T., Tresp, V. (eds.) Advances in Neural Information Processing Systems, vol. 13. MIT Press, Cambridge, MA, USA (2000)

Karasuyama, M., Takeuchi, I.: Multiple incremental decremental learning of support

vector machines. In: Bengio, Y., Schuurmans, D., Lafferty, J., Williams, C., Culotta, A. (eds.) Advances in Neural Information Processing Systems, vol. 22. Curran Associates, Inc., Red Hook, NY, USA (2009)

Laskov, P., Gehl, C., Krüger, S., Müller, K.-R.: Incremental support vector learning: Analysis, implementation and applications. Journal of Machine Learning Research **7**(69), 1909–1936 (2006)

Ndimele, P.E., Saba, A.O., Ojo, D.O., Ndimele, C.C., Anetekhai, M.A., Erondu, E.S.: Chapter 24 - remediation of crude oil spillage. In: Ndimele, P.E. (ed.) The Political Ecology of Oil and Gas Activities in the Nigerian Aquatic Ecosystem, pp. 369–384. Academic Press, (2018)

Zhang, S., Zhu, X., Zhou, S., Shang, H., Luo, J., Tsang, D.C.W.: Chapter 15 - Hydrothermal carbonization for hydrochar production and its application. In: Ok, Y.S., Tsang, D.C.W., Bolan, N., Novak, J.M. (eds.) Biochar from Biomass and Waste, pp. 275–294. Elsevier, Amsterdam, Netherlands (2019)

Middelkoop, H.: Heavy-metal pollution of the river Rhine and Meuse floodplains in the netherlands. Netherlands journal of geosciences **79**(4), 411–427 (2000)

Bivand, R.S., Pebesma, E.J., Gómez-Rubio, V., Pebesma, E.J.: Applied Spatial Data Analysis with R vol. 747248717. Springer, New York, NY, USA (2008)