Stoch Environ Res Risk Assess (2008) 22:647–660
DOI 10.1007/s00477-007-0162-x

ORIGINAL PAPER

Multi-scale support vector algorithms for hot spot detection and modelling

Alexei Pozdnoukhov · Mikhail Kanevski

Published online: 12 June 2007

Abstract The algorithmic approach to data modelling has developed rapidly these last years, in particular methods based on data mining and machine learning have been used in a growing number of applications. These methods follow a data-driven methodology, aiming at providing the best possible generalization and predictive abilities instead of concentrating on the properties of the data model. One of the most successful groups of such methods is known as Support Vector algorithms. Following the fruitful developments in applying Support Vector algorithms to spatial data, this paper introduces a new extension of the traditional support vector regression (SVR) algorithm. This extension allows for the simultaneous modelling of environmental data at several spatial scales. The joint influence of environmental processes presenting different patterns at different scales is here learned automatically from data, providing the optimum mixture of short and large-scale models. The method is adaptive to the spatial scale of the data. With this advantage, it can provide efficient means to model local anomalies that may typically arise in situations at an early phase of an environmental emergency. However, the proposed approach still requires some prior knowledge on the possible existence of such short-scale patterns. This is a possible limitation of the method for its implementation in early warning systems. The purpose of this paper is to present the multi-scale SVR model and to illustrate its use with an application to the mapping of Cs^{137} activity given the measurements taken in the region of Briansk following the Chernobyl accident.

Keywords Machine learning · Support vector regression · Multi-scale environmental modelling · Spatial mapping · Kernel methods

A. Pozdnoukhov (✉) · M. Kanevski
Institute of Geomatics and Analysis of Risk,
University of Lausanne, 1015 Lausanne,
Switzerland
e-mail: Alexei.Pozdnoukhov@unil.ch
URL: http://www.unil.ch/igar

M. Kanevski
e-mail: Mikhail.Kanevski@unil.ch

1 Introduction

Support vector regression (SVR) has recently shown promising performances in a number of spatial mapping tasks (Kanevski et al. 2002a). SVR is a robust non-linear regression method based on the Statistical Learning Theory as defined by Vapnik (1998). This is a general framework for solving statistical Machine learning problems, such as classification, regression and probability density estimation from empirical data. SVR is a non-parametric regression method, which exploits kernel expansion. It attempts at minimizing the empirical risk (the residuals on the training data), simultaneously keeping low the complexity of the model. By doing this, the over-fitting on the training data can be avoided and one may expect promising predictive abilities.

In environmental monitoring and modelling, one often has to deal with data generated by processes that are operating at different spatial scales. This is typically the case with environmental pollutants which can show locally spotted patterns of high concentrations while these concentrations are usually lower but present more structure on scales that are closer to the one of the monitored area. These differences usually reveal several underlying physical phenomena possessing different characteristic spatial scales. The deposition of radionuclides following an acci-

dental release in the atmosphere, for example, is a process that is typically governed by both a dry deposition process that will delineate the overall contamination structure and local so called hot-spots that have been generated at shorter scales by a wet deposition process.

The usual spatial interpolators are global and smoothing since they deal with some average scale only. The methods designed for the simultaneous detection and modelling of unusual spatial phenomena in the described multi-scale conditions would be particularly interesting.

In this paper, an extension of the SVR method is considered. In the proposed multi-scale SVR, the regression estimation is based on the so-called kernel dictionaries, i.e. the linear combination of different kernel functions. The combination of Gaussian Radial Basis Functions of different bandwidths is principally considered here. The bandwidths are the hyper-parameters of the learning algorithm, which have to be adjusted by the user. The joint influence of the different scales is then tuned in an automatic way from data, providing an optimum mixture of the selected short and large scale models.

In the following sections of this contribution, the reader will first (Sect. 2) find an introduction to the Statistical Learning Theory from which the Support Vector learning is derived. Section 3 is explaining further how a multi-scale SVR model can be constructed. A real case study, presented in Sect. 4, deals with the analysis of Cs^{137} radioactive contamination of the Briansk region (Russia) that followed the Chernobyl nuclear power plant accident in 1986. In this case study, the method appears to be a powerful tool for the detection and the simultaneous modelling of the radioactive release. The tricky hot-spots patterns of the analysed data were detected and modelled by the short-scale component of the model. The performances of the multi-scale SVR model were found to be competitive to those obtained by standard geostatistical tools and a number of other Machine Learning methods for regression estimation, such as General Regression Neural Network. Final remarks and discussions will be given in Sect. 5.

2 Learning from environmental data

Geostatistics has been these last decades one of the most well-established approaches for working with spatially distributed data (Cressie 1993; Chiles and Delfiner 1999). Geostatistics, in general, is a model-dependent approach based on the exploratory analysis and modelling of spatial correlation structures.

The growing amount of multi-dimensional information coming from contemporary environmental monitoring networks asks for corresponding tools. The geometric domain of the spatial processes, usually considered as 2D or 3D space, is now extended with, for example, terrain features available from digital elevation models. Geographical Information Systems can further provide useful sources of information by allowing users to easily incorporate multi-band remote sensing images into their applications, and bringing potentially another few hundreds of input dimensions to the analysed information. Applications of contemporary approaches based on the ''learning from data'' philosophy (Cherkassky and Mullier 1998) are therefore of significant interest to the data analysts. If the challenges in learning from data in the fields of biocomputing, hyperspectral remote sensing images, data mining have led to a revolution in the statistical sciences during the last decade (Breiman 2001), much remains to be done in the analyses of geo-referenced data.

Machine learning (ML) methods present a number of advantageous features over more traditional approaches. Mainly developed for high-dimensional data such as texts and images, the ML methods aim at being independent of the dimensionality of the input space. They are furthermore designed to deal with non-linear problems in a robust and non-parametric way. Particularly tailored to overcome the curse of dimensionality are Support Vector algorithms, which were found to behave well in numerous applied problems (Meyer et al. 2003). Machine learning methods provide a way of incorporating directly additional information as an input for a learning algorithm. In geostatistics, the increasing dimensionality of the input space endows the researcher with the need for higher-dimensional variogram models.

Because the dimensionality of the space of the co variables (or ''outputs'' in the ML terminology) is increasing as well, the Machine learning approaches can also be ported to multivariable problems and provide an alternative to co-kriging for example.

Machine learning methods have thus potentially a wide and exciting field of applications and open promising perspectives for research in environmental applications. Readers can find in Cherkassky et al. (2006) a number of applications of data-driven and model-free approaches to solve environmental problems (Cherkassky et al. 2006).

A new learning paradigm called Support Vector Machine (SVM) emerged in the early nineties (Boser et al. 1992; Cortes and Vapnik 1995). It was proposed essentially to solve two-class classification problems (dichotomies) but has been generalized later on to deal with multi-class classification problems, regression tasks, as well as estimations of probability densities. For what concerns their application to spatial data, learning methods based on SVMs were applied to various tasks such as the classification of soil-types, the estimation of contamination levels, the prediction of medium porosity, the predictive mapping

of contaminant concentrations, etc. (Kanevski and Canu 2000; Kanevski et al. 2002a). More recently, environmental applications of SVMs also include landslide susceptibility modelling (Brenning 2005) snow avalanche danger prediction (Pozdnoukhov et al. 2007), chemicophysical soil analysis (Bhattacharya and Solomatine 2006) or rainfall forecasting (Pai and Hong 2007). SV-based regression models have also shown promising results when used in conjunction with geostatistics (Kanevski et al. 2002b).

In this paper, an SV-based regression algorithm is developed to allow the simultaneous modelling of environmental phenomena at several different spatial scales.

2.1 Statistical learning theory

Machine learning deals with the development of algorithms describing training data and which have good generalization abilities. This means that the successful predictive algorithms are those that provide accurate estimations at the new (validation) points, where the desired quantity is unknown (Hastie et al. 2001; Cherkassky and Mullier 1998). Statistical learning theory (SLT) is devoted to such problems as extracting knowledge from a finite number of empirical observations (Vapnik 1998). The observations are considered to be independent and identically distributed (i.i.d.).

The predictive abilities of an algorithm are, obviously, one of its most important characteristics. How well an algorithm can generalize from a given training data set to predict values of the previously unseen (validation) samples can be measured with the expectation of the loss (the penalty given for an error) over the ensemble of the validation data. This value is called the *risk* in terms of SLT. This term should not be confused with the one used in environmental risk assessment. The following bounds on the generalization error or risk R are derived in SLT:

$$R(h) \leq R_{\text{emp}}(h) + R_{\text{conf}}(h), \tag{1}$$

where R_{emp} is an empirical risk found in the training data, and R_{conf} is a confidence term, which penalizes the excessively complex models. The empirical risk R_{emp} is a mean error of the algorithm applied to the training data and is measured according to the selected loss function. For example, a popular choice in regression estimation for such a loss function is the mean squared error loss.

Both terms in the bound (1) depend on the ''complexity'' h of the learning algorithm. This notion of complexity is an important one and is explained hereafter in more details. The process of learning can be seen as the choice of the most appropriate function $f(x,\lambda)$ from the available set $F(\Lambda) = \{f(x,\lambda),\ \lambda \in \Lambda\}$. The complexity of the algorithm $f(x,\lambda)$ can be controlled by the choice of the vector of hyper-parameters λ of the modelling functions in the available set, defined by the set Λ of their admissible values. To allow a comparison of the functions in the set, these need to be characterized by a single parameter defined here as the Vapnik-Chervonenkis dimension (VC-dimension) of the modelling functions (Vapnik 1998). The VC dimension is plotted on horizontal axis in Fig. 1, while the vertical axis corresponds to the value of risk. Let us consider the case where the complex model (h is large) can fit any given dataset, a situation that is typically defined as over-fitting. There is no evidence that such a model can generalize well the problem at hand, and the confidence term will here remain very large. On the other hand, a model that is overly simple can not fit the given data and capture the dependencies of the modelled process: although the confidence term of such models is low, the empirical risk is too high.

The strategy for constructing a learning machine algorithm is thus to find a trade-off between the model complexity and its fit to the data. This can be achieved by minimizing the training error while maintaining h small (see Fig. 1).

This idea, called Structural Risk Minimization (Vapnik 1998), which led to a family of Support Vector algorithms, has been further developed to solve classification tasks, regression and probability density estimations.

2.2 Support vector learning

Support vector machines provide non-linear and robust solutions by mapping the input space into a higher-dimensional feature space using kernel functions. This method has the advantage of placing into the same framework some of the most widely used models such as linear and polynomial discriminating surfaces, feed-forward neural networks, and networks composed of radial basis functions. When solving classification problems,

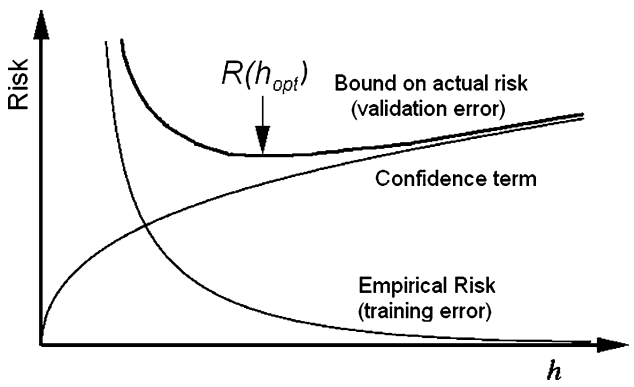

Fig. 1 Bound on the validation error which is derived in SLT. The minimum of the bound provides the optimal complexity of the predictive model for a given dataset

SVMs provide the classification directly, without solving a more general task of modelling class densities at an intermediate step. In contrast to the generative and Bayesian methods that are based on the modelling of some probability densities, SVMs are focusing on the marginal and most discriminative data samples. SVMs provide thus sparse models, i.e. only a (small) subset of data possesses non-zero weights. These data samples, called Support Vectors, usually lie close to the decision surface. They can be considered as a robust characteristic of the problem (given fixed model parameters).

The SVM classification algorithm was initially derived for the linear discriminating surfaces—hyper-planes. It was shown that in order to minimize the model complexity one has to maximize the margin between samples of different classes. More details on Support Vector for classification can be found in the tutorial of Burges (1998).

The idea of controlling the model complexity can be extended to regression problems as shown later in Vapnik (1995). Most significant examples are the Support Vectors (refer to Sect. 2.2.3; Fig. 3 below), which lie on the boundary of some ε-tube around the modelling function. The data samples lying inside the ε-tube are not taken into account as these are considered to be excessive. As a matter of fact, the use of these data samples would complicate the model too much and may lead to low generalization abilities. Let us stress that SVR is tuned in an automatic way by solving the optimization problem with a unique solution. The construction details of the SVR algorithms are given below.

2.2.1 Kernel functions

Kernel functions and the kernel ''trick'' are as much important in support vector learning as the idea of complexity control. Kernel functions are the symmetric positive-definite functions that satisfy the Mercer conditions (Aronszajn 1950). They provide a way for computing dot products in possibly infinite-dimensional feature spaces (Reproducing Kernel Hilbert Spaces, RKHS). The kernel trick consists in the substitution of dot products between the samples in the input space with the kernel function. A linear algorithm, which is formulated in terms of dot products between the samples can therefore be directly turned into its non-linear extension (Scholkopf and Smola, 2002).

Based on the training set $\{(x_1, y_1), (x_2, y_2)..., (x_N, y_N)\}$ of high-dimensional i.i.d. input vectors x_i and output measurements y_i, the basic model in Support Vector methods is a kernel expansion:

$$f(x,\alpha) = \sum_{i=1}^{N} \alpha_i K(x, x_i) + b, \tag{2}$$

where b is a constant threshold and α_i the weights to be optimized using the training data. For the sake of writing simplicity, we will denote with α the whole set of the weights $\{\alpha_i, i = 1,..., N\}$. $K(x,x_i)$ is a *kernel function.* The model (2) corresponds to some linear model $f(x, w) = wx + b$, given that w is expressed as a linear combination of training samples $w = \sum_{i=1}^{N} \alpha_i x_i$, and the dot products are substituted with the Kernel function: $(x, x_i) \rightarrow K(x, x_i)$. Consequently, the linear model in some high-dimensional feature space corresponds to the non-linear model in the input space. This duality is a remarkable property of the Support Vector algorithms.

Because the parameter(s) of the kernel are the hyper-parameter(s) of the SVM, these should be tuned using the available knowledge and data. The usual criterion to tune the parameters of the kernel function is the cross-validation or m-fold cross-validation error, or the testing error if there is enough data to split it into training and testing subsets.

Gaussian Radial Basis Functions,

$$K(x,x') = \mathrm{e}^{-\frac{(x-x')^2}{2\sigma^2}}, \tag{3}$$

are traditionally used in many practical problems. They were found to be well suited for environmental applications such as predictive spatial mapping. Its bandwidth σ, which is acting here as a hyper-parameter, is proportional to some characteristic distance implied by the data. The properties of the model will scale as shown in Fig. 1, since the model complexity increases as the value of σ decreases and visa versa. It provides thus a useful heuristic for the choice of the optimal value of this parameter.

2.2.2 Regularization and complexity

Traditionally, SV algorithms are introduced starting from their linear versions. In the case of SV classification, an optimal large margin separation hyper-plane is introduced and then extended to non-linear SVM using the Kernel trick, as shown in Burge's tutorial (Burges 1998). In regression, the flattest hyper-plane with the ε-tube which best fits the data is constructed, and then extended into the non-linear kernel expansion (2), as shown for example in (Smola and Scholkopf 2004). There is, however, an equivalent way to introduce SV algorithms, which involves the construction of a regularized risk functional (Tikhonov and Arsenin 1977) exploiting specific cost functions and regularizer types. This approach, which is exploited later on here, is often used to construct in two steps a kernel-based algorithm with some specifically desired properties. First, an appropriate cost function (a penalty for misfit of the model to the given training sample) that implies the

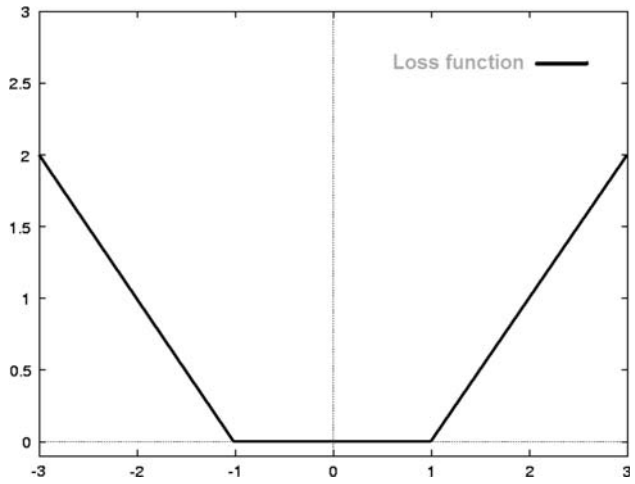

Fig. 2 The cost function of Support Vector Regression is a linear ε-insensitive function. In this figure, ε is set to one and no penalty is attributed to the samples that deviate from the regression function for more than ε = 1

sparse solution, i.e. a lot of the weights α in the expansion (2) are zero, is selected. The complexity of the decision/regression function is then penalized using regularization in RKHS. Both criteria contribute to the development of the model of the optimal complexity for a given task.

ε-Insensitive cost function The cost function of SV regression is a linear ε-insensitive one (see also Fig. 2):

$$D(y,x,f) = \begin{cases} |y - f(x)| - \varepsilon \text{ if } |y - f(x)| > \varepsilon, \\ 0 \text{ otherwise.} \end{cases} . \quad (4)$$

It provides sparseness of the model since points with values inside the ε-tube have no penalty and obtain zero weights. Note that an asymmetric cost function can be taken if the costs of over and underestimations are not equivalent in the applied problem at hand. The cost functions which are used nowadays in SVR were originally exploited in the framework of robust regression estimation (Huber 1964; Vapnik 1995).

Optimization problem The general optimization problem for finding the weights in the expansion (2) is the following:

$$\min_{\alpha} Q(\alpha) + C\sum_{i=1}^{N} D(y_i, x_i, \alpha_i), \quad (5)$$

where $Q(\alpha)$ is a *regularization term*, which we will define later on, and C is a constant defining the trade-off between the model complexity and the fit to the given training data. This minimization is usually solved given the constraints $C > \alpha_i > 0$, $i = 1,...N$. For the details on the equivalence of this approach to original ideas of SLT such as controlling the complexity of the model, please refer to Scholkopf and Smola (2002) and references therein, or to the tutorial Smola and Scholkopf (2004) (Sects. 4, 6). Intuitively, the regularization term penalizes the weights α to force the resulting model to be smooth or, in other words, not ''excessively complex''.

2.2.3 *Support vectors*

The optimal solutions provided by SV algorithms are sparse. It means that a larger part of the weights α_i take zero values (due to the specific cost functions) and only those that are strictly positive will contribute to the decision function. The training data samples that correspond to $\alpha_i > 0$ are called *Support Vectors*. In classification problems, the Support Vectors with $C > \alpha_i > 0$ are the samples that are the closest to the decision boundary between different classes. In a regression point of view, these are the samples that lie at the boundary of the ε-tube around the model. Note that if one is removing all other points except the SV from the training data set and training SVM on the Support Vectors only, one would obtain the same model, i.e. SV play the determinant role in the given learning task. This also means that the number of SV, their locations and the corresponding weights can provide a basis for the criteria to be used for the search for locations where additional measurements would change (improve) the current model (Pozdnoukhov and Kanevski 2006).

The meaning of parameter C has not been discussed yet although it plays an important function. The parameter is an upper bound for weights, which defines the trade-off between complexity of the model and the tolerance to training errors. If C is set to a sufficiently large value (infinity), the model is forced to describe the training data without errors. It can be a doubtful choice if the data are known to be noisy. Noisy data are thus often better modelled with lower values of C, which will account for training errors.

3 Linear programming support vector machines

The regularization functional (5) has still to be specified. The choice of $Q(\alpha)$ determines the type of the optimization problem which has to be solved to find the optimal weights α_i. In traditional SVR settings, the quadratic regularization is used, leading to quadratic programming optimization problem. The Linear Programming SVR (LP-SVR) is of our interest in this paper. The regularizer for LP-SVM's is defined as follows:

$$Q_{LP}(\alpha) = \sum_{i=1}^{N} (\alpha_i + \alpha_i^*), \quad (6)$$

where the summation by i corresponds to the training data. The reasons for the choice of the LP formulation will be

explained hereafter. For computational reasons, the weights α_i^* and α_i expressing, respectively, a positive and a negative impact of the training samples, need to be introduced. One of the weights is always zero, i.e. the contribution of every data sample may be either positive or negative. The kernel expansion of the model stays unchanged, hence one will define

$$f(x,\alpha) = \sum_{i=1}^{N} (\alpha_i^* - \alpha_i) K(x, x_i) + b.$$

The standard linear ε-insensitive loss function (4) is used in this formulation as it is a common choice of a loss function for the majority of SVR methods.

The resulting optimization problem is a Linear Programming problem in which

$$\min_{\alpha,\xi} \quad Q_{LP}(\alpha) + C\sum_{i=1}^{N} (\xi_i + \xi_i^*), \text{ subject to} \tag{7}$$

$$y_i - \varepsilon - \xi_i \le \sum_{i=1}^{N} (\alpha_i^* - \alpha_i) K(x_i, x_j) + b \le y_i + \varepsilon + \xi_i^*, \; i = 1, \ldots N \tag{8}$$

$$C \ge \alpha_i^* \ge 0, \quad C \ge \alpha_i \ge 0, \quad \xi_i^* \ge 0, \quad \xi_i \ge 0. \tag{9}$$

In this formulation, the non-negative variables ξ, ξ^* were introduced to substitute the non-differentiable cost function in the regularized functional. The condition implied by the ε-insensitive cost function (an allowance for the modelling function to lie inside the ε-tube without giving any penalty, illustrated in Fig. 2) is now taken into account in the constraints.

This problem can be solved in the present form using some standard Linear Programming solvers. The kernel function and ε parameter have to be specified by a user before defining the optimal weights α^*_i, α_i by solving the problem (7)–(9). The obtained weights are then used for prediction with the kernel expansion model.

3.1 Multi-scale kernels

A linear combination of the simpler basic kernels can be used to construct prediction models that are spatially adaptive as we will see hereafter. The general idea of using the kernel dictionaries and the linear regularizer (6) was introduced in (Weston et al. 1999), where it was applied to probability density estimation with SV algorithm. The idea was to build a kernel-based model which would use different kernels selected from a user-defined ''kernel dictionary'', and combine them in a data-driven way. This can be considered as a multi-kernel decomposition of functions. In the context of spatial data, this method will select the kernels from the dictionary adapting in space in a data-driven way.

The final model is provided with the following kernel expansion:

$$f(x,\alpha) = \sum_{i=1}^{N} \left[(\alpha_i^{*(1)} - \alpha_i^{(1)}) K_1(x, x_i) + \ldots + (\alpha_i^{*(k)} - \alpha_i^{(k)}) K_k(x, x_i) \right] + b, \tag{10}$$

where we denote $\alpha_i^{(p)}$ as the weight corresponding to ith training point and pth kernel.

The algorithm which would tune the $\alpha_i^{(p)}$, $\alpha^*{}_i^{(p)}$ parameters uses the linear regularizer, which is analogous to (6):

$$Q_{\mathrm{LP}}^{\mathrm{Multi}}(\alpha) = \sum_{p=1}^{k} \sum_{i=1}^{N} \left(\alpha_i^{(p)} + \alpha_i^{*(p)} \right), \tag{11}$$

where the summation by i corresponds to the training data and the summation by p corresponds to the kernels. Compared to (7)–(9), the summation by kernels is included. The optimization problem becomes therefore

$$\min_{\alpha,\xi} Q_{\mathrm{LP}}^{\mathrm{Multi}}(\alpha) + C\sum_{i=1}^{N} (\xi_i + \xi_i^*) \text{ subject to} \tag{12}$$

$$y_i - \varepsilon - \xi_i \le \sum_{i=1}^{N} \sum_{p=1}^{k} (\alpha_i^{*(p)} - \alpha_i^{(p)}) K_p(x_i, x_j) + b \le y_i + \varepsilon + \xi_i^*, \tag{13}$$

$$\alpha_i^{*(p)} \ge 0, \quad \alpha_i^{(p)} \ge 0, \quad \xi_i^* \ge 0, \quad \xi_i \ge 0. \tag{14}$$

Thus, the core of the optimization problem remains the Linear Programming, and the kernel representation of the modelling function is preserved.

Considering the spatial modelling problem, the multi-scale RBF functions can be used

$$f(x,\alpha) = \sum_{i=1}^{N} \sum_{p=1}^{k} (\alpha_i^{*(p)} - \alpha_i^{(p)}) e^{-\frac{(x-x_i)^2}{2\sigma_p^2}} + b. \tag{15}$$

The choice of the number of components in (15) has to be made by user. The choice of k components increases the dimension of the optimization problem (12)–(14), which is $2N(k+1)$. Moreover, k bandwidths σ_p have to be tuned. The two-scale Gaussian RBF is a practical choice for the case studies

$$f(x,\alpha)=\sum_{i=1}^{N}\left[(\alpha_i^{*(1)}-\alpha_i^{(1)})\mathrm{e}^{-\frac{(x-x_i)^2}{2\sigma_1^2}}+(\alpha_i^{*(2)}-\alpha_i^{(2)})\mathrm{e}^{-\frac{(x-x_i)^2}{2\sigma_2^2}}\right]+b. \quad (16)$$

3.2 Related approaches

Generally, the presented problem and its solution are related to the task of model selection which received particular attention in many fields and the readers will find that considerable amount of work has already been done in this direction. Among the most related approaches, we will cite the work of (Weston 1999) which includes the popular multiple kernel learning methods (Sonnenburg et al. 2006). These aim at exploring the (convex) combinations of kernel functions that are leading to the relevant optimization problems such as QP and LP. The main target of the latter methods is the automatic feature selection, and not some modelling that is spatially adaptive.

Another related group of methods is dealing with mixture and ensemble models (Kuncheva 2004). Approaches like boosting are also good candidates that can deal with mixtures of kernels (Bi et al. 2004): here, the final prediction is obtained from a combination of the outputs of ''weak learners'', which are the building blocks of the boosting methods.

4 Case study

The following case study aims to highlight the main properties of the developed method. Particularly, the spatial distribution of the weights α is presented. It demonstrates that the model adapts to the data spatially, meaning that the α weights in the mixture (16) are tuned automatically by solving the LP. A hot spot in Cs^{137} activity is detected and modelled with a short-scale component of (16).

4.1 Hot spot detection and modelling in Cs^{137} fallout

In the present section the problem of interpolating spatial data using multi-scale SVR is explored by means of a case study. A set of 683 observations of deposited radiocaesium (Cs^{137}, in kBq/m^2) measured in the western part of the Briansk region, Russia, will be here analysed (Savelieva et al. 2005). The data were collected following the Chernobyl nuclear power plant accident of April 1986. The first objective of the analysis is obviously to generate some predictions of the radioactivity levels at unsampled locations. The particular problem of hot-spot detection and its modelling is of particular interest. Details about data collection and other relevant information can be found in the report (Chernobyl Accident Results 2001).

4.1.1 General methodology

The case study follows the traditional approach to spatial data analysis used with geostatistics and Machine learning algorithms. Starting with an analysis of the monitoring network and the identification of possible clusters, the measurements are then analysed using statistics and geostatistics for identifying outliers and spatial correlations. These first stages are common to all environmental mapping tasks and we refer to (Kanevski and Maignan 2004) for a comprehensive description of the methodology.

The next step concerns the data preparation for the training of the algorithm. The data are split into training, testing and validation subsets. The validation set is used strictly only for checking the residuals obtained from the outcome of the selected regression model defined by the optimal parameters selected during the training phase. Because the validation subset is never used for model training or tuning, the results provide reliable information about the quality of the obtained model. The testing set is used for the prior selection of the SVR hyper-parameters. The investigation of the residuals is made according to various statistical and geostatistical criteria (e.g. statistical distribution and spatial correlation of the residuals...). It is only when the residual statistics of the validation subset are considered to be satisfactory that the mapping of the whole data set is applied with the optimal SVR model to generate the final map of the investigated variable.

The Lambert-Azimuthal projection of the spatial coordinates was used. The measured values of Cs^{137} activity were linearly scaled into [0, 1] interval. From the 683 measurements, a set of 200 validation points was extracted after some declustering procedure, which aimed to get a representative dataset over the investigated spatial domain. Note that, in the presence of clustered data, a random selection would have been inappropriate because of the risk to get a validation set with an overrepresentation of measurements from the same cluster.

4.1.2 Parameters of support vector regression

The parameters of SVR model (16) are the bandwidths of kernel functions σ_1 and σ_2, the trade-off parameter C and the width of insensitive tube ε. How these user-defined parameters are tuned and how they influence the regression estimation is explained below.

- The *RBF kernel bandwidth*(*s*) σ is defined in kilometres and acts as a hyper-parameter of the learning algorithm. For values of σ that are much smaller than the average distance between samples, the model shows some trend towards overfitting while for values of σ that are closer to the size of the spatial domain, the model will show

too much smoothing. From an SLT perspective, these observations can be explained as follows: small values of σ lead to a VC-dimension that is too high, the model becomes too complex and tends to fit any data, including outliers. On the other hand, large values of σ will lead to a low VC dimension and low model complexity, therefore the dependencies of the analysed processed will be lost. The choice of an optimal value of σ depends thus mainly on the topology of the monitoring network and on the data variability. For the multi-scale kernel models (16), thus means that different values of σ parameters can provide an elegant way for the modelling of complex phenomena observed at different spatial scales.

- The *Trade-off parameter C* is defining the trade-off between the training error and the model complexity. In a dual formulation, C defines the upper-bound of the multipliers α_i; hence, it defines the maximal influence the sample can exert on the solution. Practically, one will seek a value of C that will not be much less than the maximum values found within the training data to fit the extreme values but also not too low in view to avoid too much smoothing of the data.
- The *Insensitivity parameter* ε represents the width of the region that is insensitive to the cost function (see Figs. 2, 3 above). The parameter is thus the one that mainly defines the sparseness of the SVR solution—the points lying inside the ε-tube have zero weights. It is consequently also the main parameter that incorporates some information about the quality of the measurements. It should be of the same order as the measurement's accuracy, or as the square root of the so-called nugget value used in geostatistics, that is the sum of the variances attributed to the measurement errors and the microscale variability. Hence, ε influences the smoothness of the mapping and the larger its value, the smoother the result.

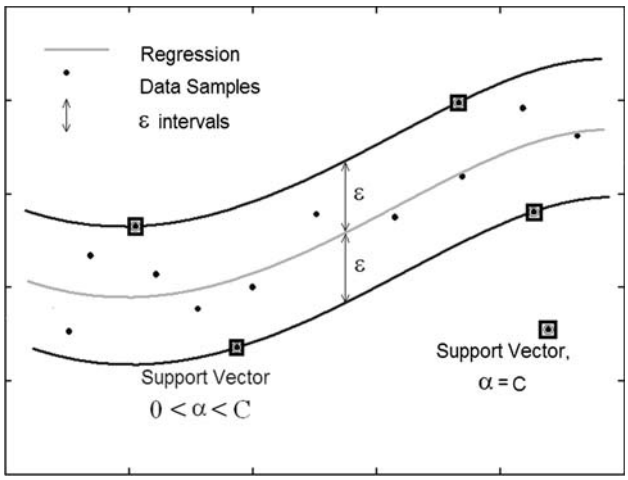

Fig. 3 SVR solution is ε-insensitive, no penalty is attributed to the samples found inside the ε-tube. Support Vectors are defined by the samples that are found on the edges of the tube. The samples found outside of the tube are likely to be some noise and their ''influence'' is bounded with C

Tuning of the parameters Two widely used approaches for tuning the parameters can be encountered. One is based on cross-validations while the second one, which is the approach that is adopted in this case study, is based on the splitting of the data into training and testing subsets and the errors are analysed for all the possible combinations of the parameters tested. The distribution of the observations into the training and the testing subsets is shown with a post plot in Fig. 4a.

A comprehensive search in a hyper-parameter space (ε, C) was performed. C was set to 25, and the values for ε were ranging between 0.02 and 0.04. The value of $\varepsilon = 0.04$ was used for the overall prediction mapping.

The search for optimal values for parameters σ_1 and σ_2 is the key to the successful outcome of the presented method. Figure 5 shows that the lowest testing error values for both parameters σ_1 and σ_2 are found in two distinct regions of the plot of the error surface (σ_1, σ_2), underlying so that the investigated phenomenon is presenting different characteristics at different spatial scales. The error surfaces shown in Fig. 5 clearly highlight symmetry along the line $\sigma_1 = \sigma_2$. This diagonal corresponds to the single-scale model. The minima of the testing error for the single-scaled model can be found for $\sigma = 5$ and 7. However, the multi-scaled model has better performance according to the testing error. Hence, the following values were chosen for the predictive mapping and validation of the model: $\sigma_1 = 1.5$ and $\sigma_2 = 6$.

4.1.3 Analysis and validation of the model

Before a model can be applied to generate the prediction maps, it has to be analysed and validated first. Figure 6 (left) shows the scatterplot of the training data versus their prediction according to the model. The values fall into the ε-tube of the width 0.04, according to the constraints (14). This underlines that the optimization problem (12)–(14) was solved correctly and that the model can be considered as being trained properly.

The weights of the trained two-scale model are displayed in Fig. 7 using Voronoi polygons. The short-scale component of the model ($\sigma = 1.5$) focuses mainly on the hot-spot found in the centre of the western part of the investigated area as well as on some other short-scale variations. The component of the bandwidth for $\sigma = 6$ mainly models large-scale structures and trends. The presence of these two scales can be further underlined

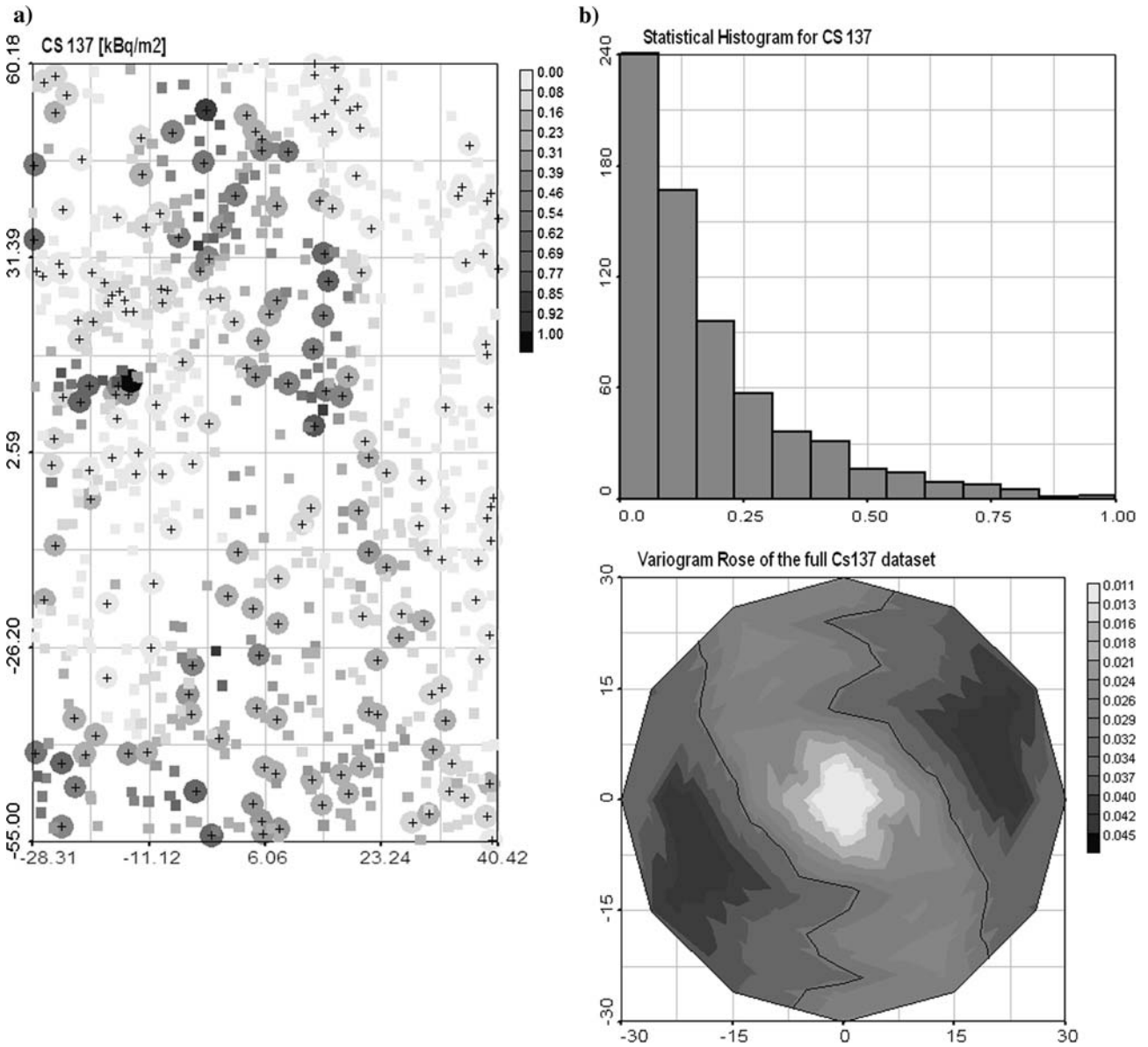

Fig. 4 Cs^{137}data. **a** Training data (183 samples) are shown with *crossed circles*, testing data are indicated with *squares*. The intensity value is shown using a normalized scale. The *X* and *Y* coordinates are given in kilometre. **b** Frequency histogram and variogram rose of the Cs^{137} data

when comparing Fig. 7 with the final predictive maps shown in Figs. 8 and 9.

The validation set, which was kept aside till now, is expected to provide us with an efficient mean to test the reliability of our models. The validation scatterplots of the single-scale and two-scale SVR models are presented on the right side of Fig. 6. One can see that both the single and the two-scale models tend to over-estimate the (linearly scaled) levels of radioactivity. If overall improvements in terms of Root Mean Squared Errors (RMSE) and correlation coefficient are found when using the two-scale model (Table 1), the last, however, will not show better estimates of the values falling within the upper 75% quantile (Q3/4 = 0.23) of the validation dataset. Still, the use of a short-scale component improved the whole model since the trade-off between different spatial scales was avoided.

Regarding the reconstruction of the spatial structures of the investigated variable, Fig. 10 shows that the omnidirectional variograms of the validation residuals of the models are close to pure nugget effect, especially when the two-scale model is used. Comparing the latter to the variogram of validation data, one can conclude that most of the spatially structured information was extracted from data. However, one will warn the readers that such an analysis may not be thoroughly correct because of the possibility to have non-stationarity within the data.

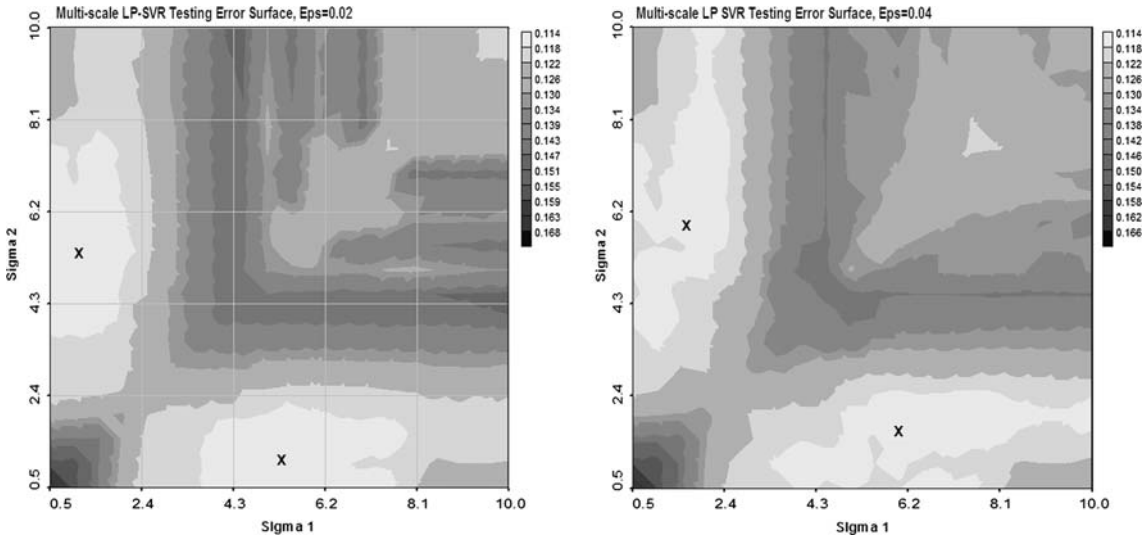

Fig. 5 Testing error surfaces in the (σ_1 σ_2) plane. The figures show a clear symmetry along the *line* $\sigma_1 = \sigma_2$. The optimal values (*low error areas*) of the two-band width parameters differ, highlighting the existence of different (short and large) spatial scales in data. The locations of the optimal parameters σ_1–σ_2 are shown with a cross

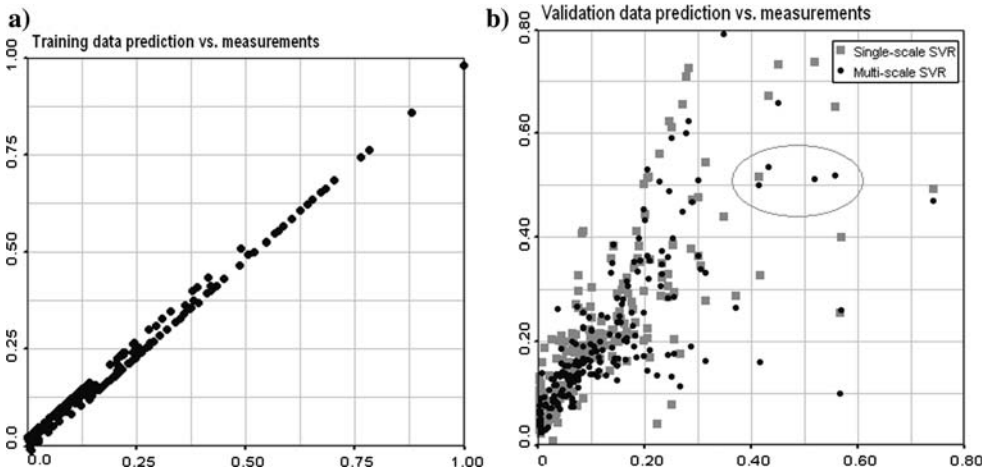

Fig. 6 **a** Scatter plot of the predicted values of the training data prediction versus observations for the multi-scale model. The residuals lie in the tube of $\varepsilon = 0.04$, according to constraints (13). **b** Scatter plot of the predicted validation data versus measurements for both the single and the two-scale SVR models. Values corresponding to the hot spot are *highlighted*

4.1.4 Multi-scale mapping

The prediction maps obtained for the short and the large-scale components of the multi-scale LP-SVR are shown in Fig. 8. While large-scale SVR component mainly models the trend, the short-scale component highlights local variations and the hot spot, which is further highlighted in the post plot of the full dataset shown in Fig. 11.

The hot spot was thus captured by the short-scale part of the model quite well. For what concerns the standard single-scaled approach, it always provides some trade-off, choosing the averaged parameters, which may not always be the best compromise. For example, the ordinary SVR with optimal C and ε parameters provided the minimum validation error of 0.125 for $\sigma = 5$. In the case of the double-scaled model, the obtained validation error was of 0.11. This improvement of the two-scale model in the presented case study is therefore twofold. It provided first a more accurate model of the short-scale dependencies as well as of the hot spot, while this hot spot is smoothed when applying the single-scale SVR model. Secondly, the two-scale model allowed avoiding the trade-off and finding optimal values of the spatial kernel bandwidths for the modelling of the data. This is further underlined by the lower RMSE obtained on the whole validation data.

5 Discussion and conclusions

A number of state-of-the-art methods that can be used for the task of spatial prediction mapping exist, among which

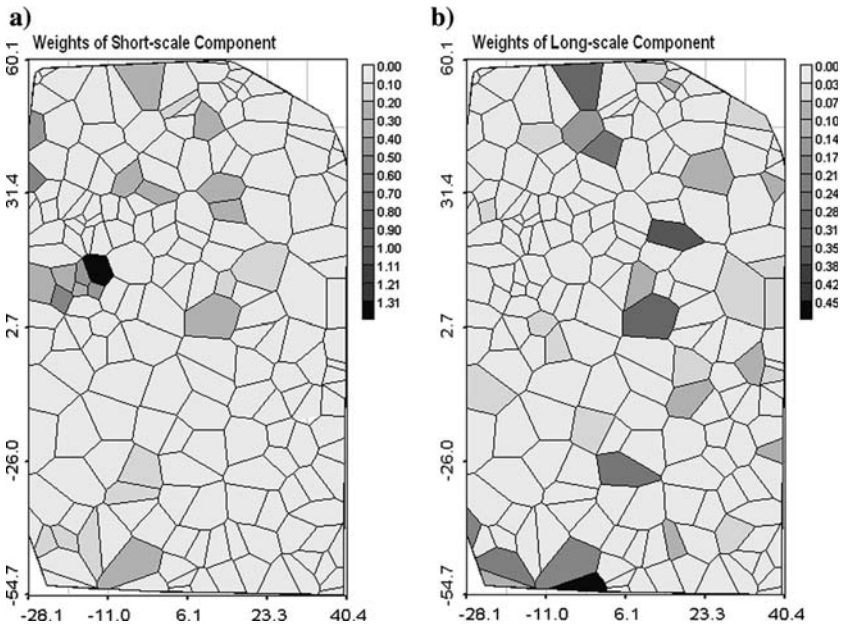

Fig. 7 Weights of the multi-scale kernel expansion (6) found for the selected parameters $\sigma_1 = 1.5$ (**a**), $\sigma_2 = 6$ (**b**). Note that while the large-scale component models large-scale variations and trends, the short-scale part of the model concentrates mainly on the hot spot. X and Y coordinates are given in km

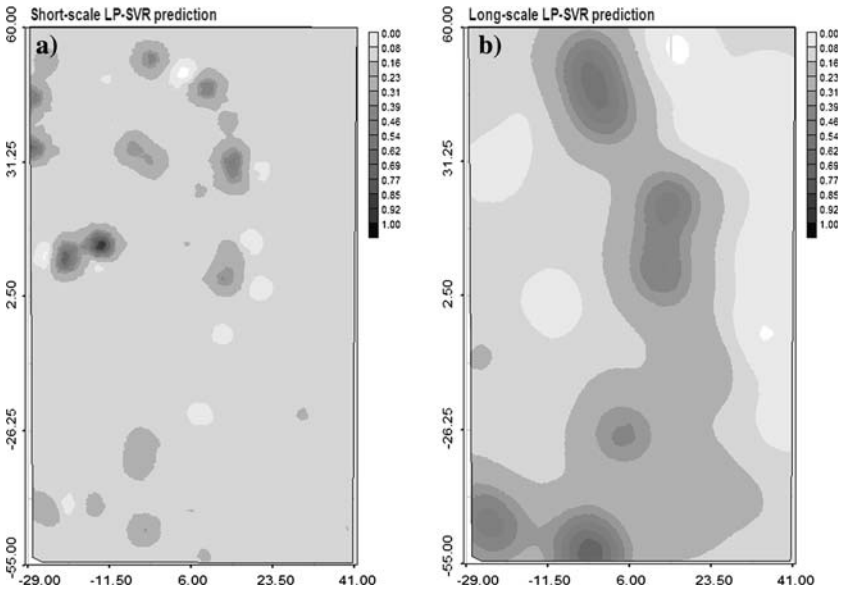

Fig. 8 Prediction maps of the multi-scale SVR components of different scales, **a** $\sigma_1 = 1.5$, **b** $\sigma_2 = 6$. X and Y coordinates are given in kilometre

deterministic interpolators (Nearest Neighbours, Inverse Weighted Distance), geostatistical estimators (ordinary kriging, simulations) and artificial neural networks are regularly encountered (Kanevski et al. 1996). Geostatistical estimators (Deutsch and Journel 1997) are probably the most widely used functions nowadays because of their aptitude to effectively benefit from the information extracted from the data about the spatial correlation of the analysed variable. Geostatistics treat the measurements as the realization of some spatial random process and the estimation method is based on a model of the spatial covariance function, the variogram. This dependence on the variogram is known to be one of the most challenging obstacles to the development of automated mapping systems built around geostatistical algorithms. This is particularly true when a few observations only are available and/or when the condition of stationarity is not verified. These limitations have been motivating new developments based on other foundations, among others the Machine Learning and Statistical Learning Theory. A useful link between Machine Learning and geostatistics has already been established in the field of Gaussian Processes (Rasmussen and Williams 2006).

Two main types of algorithms based on the Machine Learning and Statistical Learning Theory have been particularly studied by the authors: kernel-based machine learning algorithms such as Support Vector methods and General Regression Neural Network (GRNN) (Specht 1991;

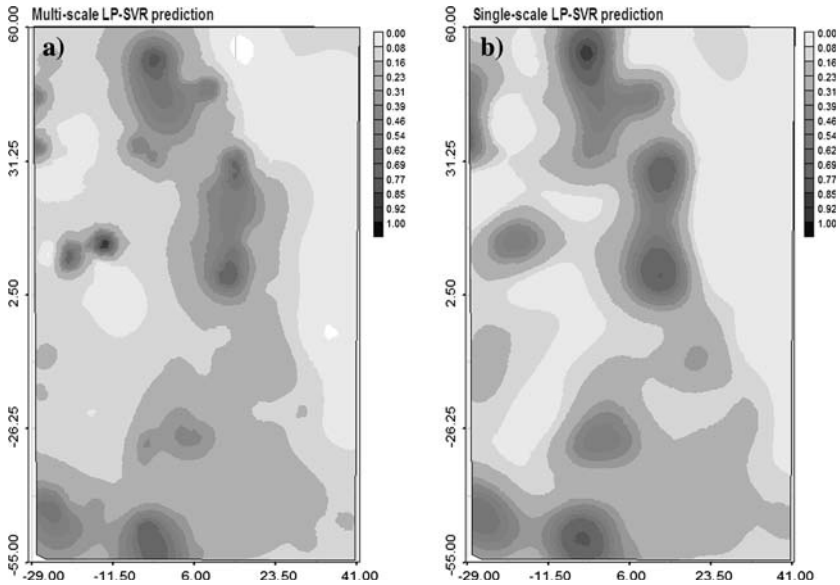

Fig. 9 **a** Multi-scale predictive mapping with the developed LP-SVR model. **b** Single scale (standard SVR) predictive mapping at scale $\sigma = 5$. Note that the hot-spot in the western part has been considerably smoothed in the single-scaled prediction. The X and Y coordinates are given in kilometre

Table 1 Validation RMSE and correlation coefficient ρ obtained for four spatial interpolation models tested

	Training RMSE	Training ρ	Validation RMSE	Validation ρ	Validation Q3/4 RMSE	Validation Q3/4 Ro
SVR	0.022	0.96	0.125	0.74	0.17	0.36
Multi-scale SVR	0.017	0.98	0.110	0.76	0.172	0.35
GRNN	0.075	0.93	0.121	0.74	0.164	0.40
Ordinary kriging	0	1	0.130	0.73	0.172	0.34

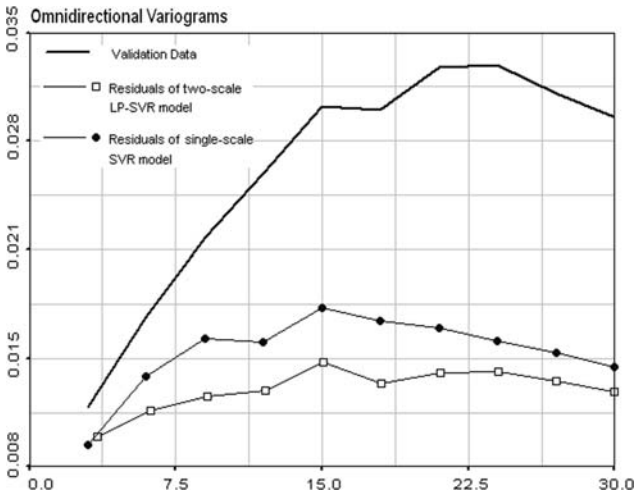

Fig. 10 Experimental omnidirectional variograms of the validation data, and of the validation residuals for both the single-scale and the two-scale SVR models. The variogram of the residuals of the single scale model reveals some short-scale structures, while the variogram of the residuals of the two-scale model is closer to a pure nugget, showing that most of the spatial structure could be extracted. The lag distances are in kilometre

Timonin and Savelieva 2005). In the General Regression Neural Network (GRNN), the predictions are obtained by taking the weighted sum of the adjacent measurements. Compared to the SVR approach, GRNN is a faster method, which can be trained and tuned in a more effective way.

The SVR method whish has been discussed in detail in this paper is a robust regression estimator allowing for the development of new extensions. The extension developed here, called ''the multi-scale approach'', showed, by means of a case study involving radioactivity measurements, that processes operating simultaneously but at different scales could be identified and that the handling of situations of non-stationarity was facilitated. These advantages over geostatistical estimators like ordinary kriging are particularly interesting when designing environmental monitoring systems conceived for the surveillance of critical variables which values can rapidly fluctuate in time and space (Pozdnoukhov 2005).

A number of open questions still remain for the use of the methods for their implementation in automated environmental monitoring and decision support systems. First of all, like in geostatistics, Machine Learning based methods rely very much on some training process and are thus very much depending on the quantity of the data that can be used for the training as well as on their quality. We have seen that the multi-scale SVR offers some means to adapt to the spatial scale of the data and that it can so provide an interesting possibility for the detection of local anomalies that may typically arise in situations of

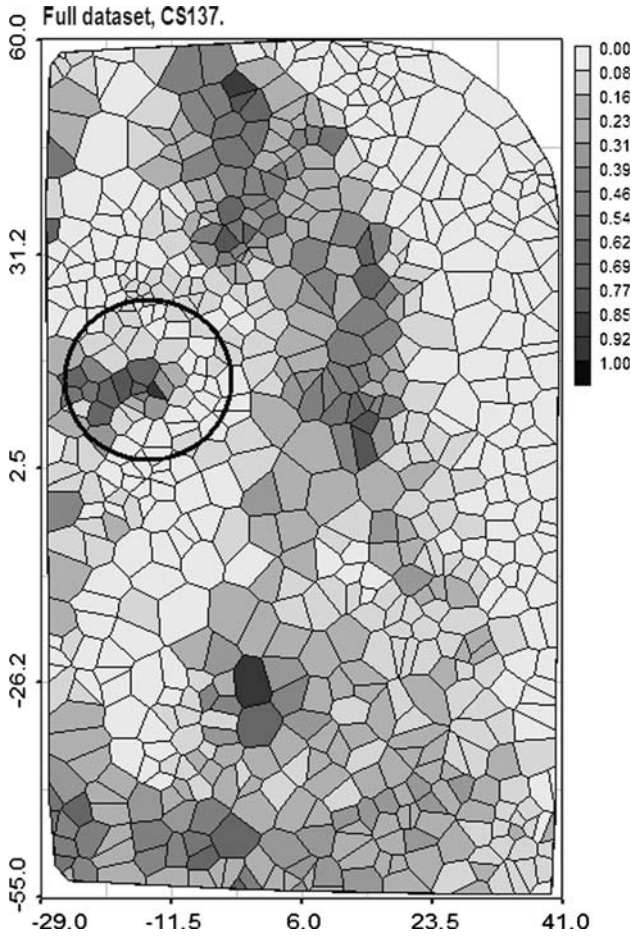

Fig. 11 Post-plot of the full dataset of Cs137 activity visualized using Voronoi polygons. The location of the hot spot is highlighted with a *circle*. The *X* and *Y* coordinates are given in kilometre

an early phase of an environmental emergency. Nevertheless, the method requires some prior knowledge on the possible existence of such short-scale patterns, knowledge that can be difficult to get in the early phase of an environmental accident presenting extreme events. Hence, the algorithms would have to be trained with the anticipation of an event presenting possible known patterns. Decision-makers would also want to rely on information regarding the uncertainties of the SVR predictions, which are not yet available. The traditional approach of mapping the variance of the predictions could be a solution. Other approaches are under development and approaches as those discussed in Nix and Weigend (1995) for example, offer interesting possibilities. Another important research direction in SVR is the incorporation of additional information (soft data, prior knowledge and physical models) about the investigated process, which may improve the prediction performance of the model. In the case study, described in Savelieva et al. (2005), mapping methods based on the Bayesian Maximum Entropy (Christakos 2000), which incorporated in the estimation process ''hard'' data as well as ''soft'' information, that is intervals or histograms obtained after repetitive measurements, showed improved prediction performances of the model.

Acknowledgments The research was supported by Swiss National Science Foundation projects ''GeoKernels: Kernel-Based Methods for Geo- and Environmental Sciences'' (project No. 200021-113944). The authors would like to thank Gregoire Dubois and an anonymous reviewer for helpful comments and suggestions.

References

Aronszajn N (1950) Theory of reproducing kernels. Trans Am Math Soc 68:337–404

Bhattacharya B, Solomatine D (2006) Machine learning in soil classification. Neural Netw 19(2):186–195

Bi J, Zhang T, Bennett K (2004) Column-generation boosting methods for mixture of kernels. In: Proceedings of ACM SIGKDD international conference on knowledge discovery and data mining (SIGKDD'04), pp 521–526

Boser B, Guyon I, Vapnik V (1992) A training algorithm for optimal margin classifiers. Fifth annual workshop on computational learning theory. ACM Press, Pittsburgh

Breiman L (2001) Statistical modeling: the two cultures. Stat Sci 16(3):199–231

Brenning A (2005) Spatial prediction models for landslide hazards: review, comparison and evaluation. Nat Hazards Earth Syst Sci 5:853–862

Burges C (1998) A tutorial on support vector machines for pattern recognition. Data Min Knowl Discov 2(2):121–167

Cherkassky V, Mullier F (1998) Learning from data. Wiley, New York

Cherkassky V, Krasnopolsky M, Solomatine D, Valdés J (2006) Introduction to special issue: earth sciences and environmental applications of computational intelligence. Neural Netw 19(2):111

Chernobyl Accident Results and Problems in Eliminating Its Consequences in Russia 1986–2001, Russian National Report (2001). http://www.ibrae.ac.ru/english/natrep-2001.htm

Chiles J-P, Delfiner P (1999) Geostatistics: modeling spatial uncertainty. In: Probability and statistics, Wiley Series

Christakos G (2000) Modern Spatiotemporal Geostatistics. Oxford University Press, New York

Cressie N (1993) Statistics for spatial data (Revised edition). Wiley, New York

Cristianini N, Shawe-Taylor J (2000) An introduction to support vector machines and other kernel-based learning methods. Cambridge University Press, Cambridge

Cortes C, Vapnik V (1995), support vector networks. Mach Learn 20:273–297

Deutsch CV, Journel AG (1997) GSLIB. Geostatistical software library and user's guide. Oxford University Press, New York

Hastie T, Tibshirani R, Friedman J (2001) The elements of statistical learning. Springer, Heidelberg

Huber P (1964) Robust estimation of location parameter. Ann Math Stat 35(1)

Kanevski M, Canu S (2000) Spatial data mapping with support vector regression, IDIAP Research Report, RR-00-09

Kanevski M, Maignan M (2004) Analysis and modelling of spatial environmental data. EPFL Press

Kanevski M, Arutyunyan R, Bolshov L, Demyanov V, Maignan M (1996) Artificial neural networks and spatial estimations of Chernobyl fallout. Geoinformatics 7(1):5–11

Kanevski M, Pozdnoukhov A, Canu S, Maignan M (2002a) Advanced spatial data analysis and modelling with support vector machines. Int J Fuzzy Syst 4(1):606–616

Kanevski M, Parkin R, Pozdnoukhov A, Timonin V, Maignan M, Yatsalo B, Canu S (2002b) Environmental data mining and modelling based on machine learning algorithms and geostatistics. In: International environmental modelling and software society conference (iEMSs2002) Lugano, Switzerland, pp 414–419

Kuncheva L (2004) Combining pattern classifiers. Wiley, New Jersey

Meyer D, Leisch F, Hornik K (2003) The support vector machine under test. Neurocomputing, 55:169–186

Nix DA, Weigend AS (1995) Learning local error bars for non-linear regression. In: Proceedings of NIPS 7, pp 489–496

Pai P-F, Hong W-C (2007) A recurrent support vector regression model in rainfall forecasting. Hydrol Process 21(6):819–827

Pozdnoukhov A (2005) Support vector regression for automated robust spatial mapping of natural radioactivity. Applied GIS 1(2):21-01–21-10

Pozdnoukhov A, Kanevski M (2006) Monitoring network optimisation for spatial data classification using support vector machines. Int J of Environ Pollut 28(3/4):465–484

Pozdnoukhov A, Kanevski M, Purves RS (2007) Avalanche danger forecasting with machine learning methods. Geophys Res Abstr 9:01917, EGU

Rasmussen CE, Williams C (2006) Gaussian processes for machine learning. MIT press, Cambridge

Savelieva E, Demyanov V, Kanevski M, Serre M, Christakos G (2005) BME-based uncertainty assessment of the Chernobyl fallout. Geoderma 128:312–324

Smola A, Scholkopf B (2004) A tutorial on support vector regression. Stat Comput 14:199–222

Scholkopf B, Smola A (2002) Learning with kernels: support vector machines, regularization, optimization, and beyond. MIT Press, Cambridge

Specht DF (1991) A generalized regression neural network. IEEE Trans Neural Netw 2:568–576

Sonnenburg S, Raetsch G, Schaefer C, Scholkopf B (2006) Large scale multiple kernel learning. In: Bennett K, Hernandez EP (eds) J Mach Learn Res 7:1531–1565

Tikhonov AN, Arsenin VY (1977) Solutions of Ill-posed problems, Wiley, New York

Timonin V, Savelieva E (2005) Spatial prediction of radioactivity using general regression neural network. Appl GIS 1(2):19-01–19-14

Vapnik V (1995) The nature of statistical learning theory. Springer, Heidelberg

Vapnik V (1998) Statistical learning theory. Wiley, New York

Weston J (1999) Extensions to the support vector method. PhD Thesis, Royal Holloway University of London

Weston J, Gammerman A, Stitson M, Vapnik V, Vovk V, Watkins C (1999) Support vector density estimation. In: Schölkopf B, Burges CJC, Smola AJ (Eds) Advances in kernel methods-support vector learning. MIT Press, Cambridge, pp 293–306