UNIVERSITY
OF TRENTO

DIPARTIMENTO DI INGEGNERIA E SCIENZA DELL'INFORMAZIONE

38050 Povo – Trento (Italy), Via Sommarive 14
http://www.disi.unitn.it

OPERATORS FOR TRANSFORMING KERNELS INTO QUASI-LOCAL KERNELS THAT IMPROVE SVM ACCURACY

Nicola Segata and Enrico Blanzieri

January 2008

Technical Report # DISI-08-009

Operators for transforming kernels into quasi-local kernels that improve SVM accuracy

Nicola Segata and Enrico Blanzieri *

January 19, 2008

Abstract

In the field of statistical machine learning, the integration of kernel methods with local information has been proposed through locality-improved kernels for Support Vector Machines (SVM) that make use of prior information, local kernels and local SVM that apply the SVM approach only on the subset of points close to the testing one. Here we propose a novel family of operators on kernels able to integrate the local information into any kernel without prior information obtaining *quasi-local* kernels. The quasi-local kernels maintain the possibly global properties of the input kernel and they increase the kernel value as the points get closer in the feature space of the input kernel. The operators combine the input kernel with a locality-dependent term, and accept two parameters that regulate the width of the exponential influence of points in the locality-dependent term and the balancing between the two terms. Experiments carried out with data-dependent systematic selection of the parameters of the operators (i.e. without the need for model selection phase on the obtained kernels) on a total of 33 datasets with different characteristics and application domains, achieve very good results.

Keywords: SVM, locality, kernel methods, operators on kernels, local SVM.

1 Introduction

Support Vector Machines [8] (SVM) are state-of-the-art classifiers and are now widely used and applied over a wide range of domains. Reasons for SVM's success are multiple: the presence of an elegant bound on generalization error [33], the fact that SVM is based on kernel functions $k(\cdot,\cdot)$ representing the scalar product of the sample mapped in a Hilbert space and the relative lightweight computational cost of the model in the evaluation phase. For a review on SVM and kernel methods the reader can refer to [28].

Locality in classification plays a crucial role [6]. Locality is invoked in non evenly distributed datasets and, more generally, where the properties of a sample can be more

*N. Segata and E. Blanzieri are with the Dipartimento di Ingegneria e Scienza dell'Informazione, University of Trento, Italy. E-Mail: {segata, blanzier}@disi.unitn.it.

precisely estimated by analysing only the samples of the sub-region in which it lies. For example, one of the reasons for the success of the K-Nearest Neighbors (KNN)[1] algorithm is the fact that it is deeply based on the notion of locality. In kernel methods, locality has been introduced with two meanings: i) as local relationship between the features, i.e. local feature dependence, adding prior information reflecting it, ii) as distance proximity between points, i.e. local points dependence, enhancing the kernel values for points that are close to each other and/or penalizing the points that are far from each other. The first meaning has been exploited by *locality-improved kernels*, the second by *local kernels* and *local SVM*.

Locality-improved kernels [28] take into account the prior knowledge of the local structure in data such as local correlation between pixels in images. The way the prior information is integrated into the kernel depends on the specific task but, in general, the kernel increases similarity and correlation of selected features that are considered locally related. Locality-improved kernels were successfully applied on image processing [27] and on bioinformatics tasks [35] [12].

Local kernels do not make use of prior information and when the distance between a test point and a training point tends to infinity the value of the kernel is constant and independent of the test point [2] [29]. A popular local kernel is the radial basis function (RBF) kernel that tends to zero for points whose distance is high with respect to a width parameter that regulates the degree of locality. On the other hand, distant points influence the value of global kernels (e.g. linear, polynomial and sigmoidal kernels). Local kernels and in particular the RBF kernel show very good classification capability but they can suffer from the curse of dimensionality problem [3] and they can fail with datasets that require long range extrapolation. An attempt to mix the good characteristics of local and global kernels is reported in [29] where RBF and polynomial kernels are considered for SVM regression.

Local SVM was independently proposed by Blanzieri and Melgani [4] [5] and by Zhang et al. [34] and applied respectively to remote sensing and visual recognition tasks with good results. The main idea of local SVM is to build at evaluation time a sample-specific maximal marginal hyperplane based on the set of K-neighbors. In [4] it is also proved that the local SVM has chance to have a better bound on generalization with respect to SVM. Local SVM can be seen as representative of the larger class of local learning algorithms [6] [10] that try to locally adjust the separating surface considering the characteristics of each region of the training set, the assumption being that important properties of a test point can be more precisely determined by the local neighbors rather than by the whole training set. Local SVM suffers from the high computational cost of the testing phase that comprises for each sample the selection of the K nearest neighbors and the computation of the maximal separating hyperplane, and from the problem of tuning the K parameter. The first drawback prevents the scalability of the method for large datasets, the second makes necessary complex tuning procedures.

[1] From now on, for notational reasons, we refer to the K parameter of KNN based methods with upper-case K, reserving lower-case k for denoting kernel functions.

In this work we present a family of operators that transform any existing input kernel into a kernel that integrates locality information. The idea is to balance the input kernel with a local kernel whose value increases as the points get closer in the feature space of the input kernel. A very simple example is the balancing between the linear kernel and the RBF kernel, where the RBF kernel is local in the feature space of the linear kernel coinciding in this case with the input space. The operators make it possible to systematically add locality information to kernel functions preserving the positive definite property in order to take advantage of the spatial relationships between samples. The meaning of locality we exploit is not only based on the distance on the input space but, more generally, relies on the distance in the feature space which is accessible through the scalar product, namely the kernel. This new family of kernels, opportunely tuned, maintains the original kernel behaviour for non-local regions, while increasing the values of the kernel for points in regions where the local information is more important. In this way we aim to take advantage of both locality information and the long-range extrapolation ability of global kernels, alleviating also the curse of dimensionality problem of the local kernels and balancing the compromise between interpolation and generalization capability. Moreover, being a kernel applied on normal SVM, this approach overcomes the computational limitation of local SVM.

The paper is organized as follows. After recalling in section 2 some preliminaries on SVM, kernel functions and local SVM, in section 3 we present the new family of operators that produces quasi-local kernels. The artificial example presented in section 4 illustrates intuitively how the quasi-local kernels work. In section 5 we propose a first experiment on 20 datasets with the double purpose of investigating the classification performance and of identifying the most suitable systematic settings of the quasi-local kernel parameters. The most promising quasi-local kernels with the chosen systematic parameters settings are applied in the experiment of section 6 to 13 large classification datasets. Finally, in section 7, we draw some conclusions.

2 SVM and kernel methods preliminaries

Support vector machines (SVMs) are classifiers based on statistical learning theory [33]. The decision rule of an SVM is $SVM(x) = sign(\langle w, \Phi(x)\rangle_{\mathcal{F}} + b)$ where $\Phi(x) : \mathbb{R}^p \rightarrow \mathcal{F}$ is a mapping in some transformed feature space $\mathcal{F}$ with inner product $\langle \cdot, \cdot \rangle_{\mathcal{F}}$. The parameters $w \in \mathcal{F}$ and $b \in \mathbb{R}$ are such that they minimize an upper bound on the expected risk while minimizing the empirical risk. The minimization of the complexity term is achieved by minimizing the quantity $\frac{1}{2} \cdot \|w\|^2$, which is equivalent to maximizing the margin between the classes. The empirical risk term is controlled through the following set of constraints:

$$y_i \left(\langle w, \Phi(x_i)\rangle_{\mathcal{F}} + b\right) \geq 1 - \xi_i \quad \text{with } \xi_i \geq 0 \text{ and } i = 1, \ldots, N \tag{1}$$

where $y_i \in \{-1, +1\}$ is the class label of the i-th nearest training sample. Such constraints mean that all points need to be either on the borders of the maximum margin separating hyperplane or beyond them. The margin is required to be 1 by a normalization of distances. The presence of the slack variables ξ_i's allows some misclassification on the training set. By reformulating such an optimization problem with Lagrange multipliers α_i ($i = 1, \dots, N$), and introducing a positive definite kernel function $k(\cdot, \cdot)$ that substitutes the scalar product in the feature space $\langle \Phi(x_i), \Phi(x) \rangle_{\mathcal{F}}$ it is possible to obtain a decision rule expressed as:

$$SVM(x) = sign\left(\sum_{i=1}^{N} \alpha_i y_i k(x_i, x) + b\right)$$

where training points with nonzero Lagrange multipliers are called support vectors. The introduction of the positive definite (PD) kernels avoids the explicit definition of the feature space $\mathcal{F}$ and of the mapping Φ [28] [9]. A kernel is PD if it is the scalar product in some Hilber space, i.e. the kernel matrix is symmetric and positive definite[2].

The maximal separating hyperplane defined by the SVM has been shown to have important generalization properties and nice bound on the VC dimension [33]. In particular we refer to the following theorem:

Theorem 1 (Vapnik [33] p.139)**.** *The expectation of the probability of test error for a maximal separating hyperplane is bounded by*

$$EP_{error} \leq E\left\{\min\left(\frac{m}{l}, \frac{1}{l}\left[\frac{R^2}{\Delta^2}\right], \frac{p}{l}\right)\right\}$$

where l is the cardinality of the training set, m is the number of support vectors, R is the radius of the sphere containing all the samples, $\Delta = 1/|w|$ is the margin, and p is the dimensionality of the input space.

Theorem 1 states that the maximal separating hyperplane can generalize well as the expectation on the margin is large (since a large margin minimizes the $\frac{R^2}{\Delta^2}$ ratio).

2.1 Local and global basic kernels

Kernel functions can be divided in two classes: local and global kernels [29]. Following [2] we define the locality of a kernel as:

Definition 1 (Local kernel)**.** *A PD kernel k is a* local kernel *if, considering a test point x and a training point x_i, we have that*

$$\lim_{\|x - x_i\| \to \infty} k(x, x_i) \to c_i \tag{2}$$

with c_i constant and not depending on x. If a kernel is not local, it is considered to be global*.*

In this work we will consider as baseline and as inputs of the operators we will introduce in the next section, the linear kernel k^{lin}, the polynomial kernel k^{pol}, the radial basis function

[2] In the present work, we frequently refer to PD kernels simply as kernels.

kernel k^{rbf} and the sigmoidal kernel k^{sig}. We refer to these four kernels as *reference input kernels* and we recall their definitions:

$$\begin{array}{rclrcl} k^{lin}(x,x') & = & \langle x,x'\rangle & k^{pol}(x,x') & = & (\gamma^{pol}\cdot\langle x,x'\rangle + r^{pol})^d \\ k^{rbf}(x,x') & = & \exp(-\gamma^{rbf}\cdot ||x-x'||^2) & k^{sig}(x,x') & = & \tanh(\gamma^{sig}\cdot\langle x,x'\rangle + r^{sig}) \end{array}$$

with $\gamma^{pol}, \gamma^{rbf}, \gamma^{sig} > 0$, $r^{pol}, r^{sig} \geq 0$ and $d \in \mathbb{N}$.

It is simple to show that the only local kernel is k^{rbf} since for $\|x - x_i\| \to \infty$ we have that $k^{rbf}(x, x_i) \to 0$ (i.e. a constant that does not depend on x), whereas k^{lin}, k^{pol} and k^{sig} are global.

For the radial basis function kernel k^{rbf} we set the parameter γ^{rbf} with the inverse of the 0.1 quantile of the distribution of $\|x_i - x_j\|$, namely the Euclidean distances between every pair of samples x_i, x_j in the training set [30]. In this way the width of the k^{rbf} is of the same order of magnitude of the distance between points.

It is known that the linear, polynomial and radial basis function kernels are proper kernels since they are PD. It has been shown, however, that the sigmoidal kernel is not PD [28]; nevertheless it has been successfully applied in a wide range of domains as discussed in [25]. In [22] is showed that the sigmoidal kernel can be conditionally positive definite (CPD) for certain parameters and for specific inputs. Since CPD kernels can be safely used for SVM classification [26], the sigmoidal kernel is suitable for SVM only on a subset of the parameters and input space. In this work we use the sigmoidal kernel being aware of its theoretical limitations, which can be reflected in non-optimal solutions and convergence problems in the SVM application.

2.2 Local SVM

The method [4] combines locality and searches for a large margin separating surface by partitioning the entire transformed feature space through an ensemble of local maximal margin hyperplanes. In order to classify a given point x' of the p-dimensional input feature space, we need first to find its K nearest neighbors in the transformed feature space $\mathcal{F}$ and, then, to search for an optimal separating hyperplane only over these K nearest neighbors. In practice, this means that an SVM classifier is built over the neighborhood of each test point x'. Accordingly, the constraints in (1) become:

$$y_{r_x(i)}\left(w \cdot \Phi(x_{r_x(i)}) + b\right) \geq 1 - \xi_{r_x(i)}, \text{ with } i = 1, \ldots, K$$

where $r_{x'} : \{1, \ldots, N\} \to \{1, \ldots, N\}$ is a function that reorders the indexes of the N training points defined recursively as:

$$\begin{cases} r_{x'}(1) = \underset{i=1,\ldots,N}{\operatorname{argmin}} \|\Phi(x) - \Phi(x')\|^2 \\ r_{x'}(j) = \underset{i=1,\ldots,N}{\operatorname{argmin}} \|\Phi(x) - \Phi(x')\|^2 \quad \text{ with } i \neq r_{x'}(1), \ldots, r_{x'}(j-1) \text{ for } j = 2, \ldots, N \end{cases}$$

In this way, $x_{r_{x'}(j)}$ is the point of the set X in the j-th position in terms of distance from x' and the following holds: $j < K \Rightarrow \|\Phi(x_{r_{x'}(j)}) - \Phi(x')\| \leq \|\Phi(x_{r_{x'}(K)}) - \Phi(x')\|$ because of the monotonicity of the quadratic operator. The computation is expressed in terms of kernels as:

$$\begin{aligned} & ||\Phi(x) - \Phi(x')||^2 = \Phi^2(x) + \Phi^2(x') - 2 \cdot \langle \Phi(x), \Phi(x') \rangle_{\mathcal{F}} = \\ = \quad & \langle \Phi(x), \Phi(x) \rangle_{\mathcal{F}} + \langle \Phi(x'), \Phi(x') \rangle_{\mathcal{F}} - 2 \cdot \langle \Phi(x), \Phi(x') \rangle_{\mathcal{F}} = k(x,x) + k(x',x') - 2 \cdot k(x,x'). \end{aligned} \tag{3}$$

In the case of the linear kernel, the ordering function can be built using the Euclidean distance, whereas if the kernel is not linear, the ordering can be different. If the kernel is Gaussian the ordering function is equivalent to using the Euclidean metric.

The decision rule associated with the proposed method is:

$$K\text{NNSVM}(x) = sign\left(\sum_{i=1}^{K} \alpha_{r_x(i)} y_{r_x(i)} k(x_{r_x(i)}, x) + b \right).$$

For $K = N$, the KNNSVM method is the usual SVM whereas, for $K = 2$, the method implemented with the linear kernel corresponds to the standard 1-NN classifier. Conventionally, in the following, we assume that also 1-NNSVM is equivalent to 1-NN.

The method can be seen as a KNN classifier implemented in the input or in a transformed feature space with a SVM decision rule or as a local SVM classifier. In this second case the bound on the expectation of the probability of test error becomes:

$$EP_{error} \leq E\left\{ \min\left(\frac{m}{K}, \frac{1}{K}\left[\frac{R^2}{\Delta^2}\right], \frac{p}{K} \right) \right\}$$

where m is the number of support vectors. Whereas the SVM has the same bound with $K = N$, apparently the three quantities increase due to $K < N$. However, in the case of KNNSVM the ratio $\frac{R^2}{\Delta^2}$ decreases because: 1) R (in the local case) is smaller than the radius of the sphere that contains all the training points; and 2) the margin Δ increases or at least remains unchanged. The former point is easy to show, while the second point (limited to the case of linear separability) is stated in the following theorem [5].

Theorem 2. *Given a set of N training points $X = \{x_i \in \mathbb{R}^p\}$, each associated with a label $y_i \in \{-1, 1\}$, over which is defined a maximal margin separating hyperplane with margin Δ_X, if for an arbitrary subset $X' \subset X$ there exists a maximal margin hyperplane with margin $\Delta_{X'}$ then the inequality $\Delta_{X'} \geq \Delta_X$ holds. For the proof see [5].*

As a consequence of Theorem 2 the KNNSVM has the potential of improving over both KNN and SVM as empirically shown in [4] and [34] in remote sensing and visual applications.

Apart from the SVM parameters (C and the kernel parameters), the only parameter of KNNSVM that needs to be tuned is the number of neighbors K. K can be estimated on the training set among a predefined series of natural numbers (usually a subset of the odd numbers between 1 and the total number of points) choosing the value that shows better predictive accuracy with a 10-fold cross validation approach. In this work, when we refer to the KNNSVM classifier we assume that K is estimated in this way. In the cases in which we use a particular a-priori value of K for the KNNSVM we explicitly mention it or denote it directly with the specific number (e.g. 1-NNSVM).

3 Operators that transform kernels into quasi-local kernels

In this section we introduce the operators we use to integrate the locality information into existing kernels obtaining quasi-local kernels. An operator on kernels, generically denoted as $\mathcal{O}$, is a function that accepts a kernel as input and transforms it into another a kernel, i.e. $\mathcal{O}$ is an operator on kernels if $\mathcal{O}\, k$ is a kernel (supposing that k is a kernel). Note that we are not limiting the definition of operator to linear function as sometimes the word operator implies. A lot of operators on kernels have been defined: examples can range from the simple multiplication by a constant $(\mathcal{O}_c\, k)(x, x') = c \cdot k(x, x')$ which is a linear operator, to more complex operators such as exponentiation $(\mathcal{O}_e\, k)(x, x') = \exp(k(x, x'))$, since $c \cdot k(x, x')$ and $\exp(k(x, x'))$ are kernels [13] provided that k is a kernel. Also the identity function can be thought of as an operator on kernel such that $(\mathcal{I}\, k)(x, x') = k(x, x')$.

Our operators produce kernels that we call *quasi-local* kernels, combining the input kernel with another kernel based on the distance in the feature space of the input kernel. The formal definition of quasi-locality will be discussed in subsection 3.4. In the case of a global kernel as input of the operators, the intuitive effect of the *quasi-locality* of the resulting kernels is that they are not local for definition 1 but at the same time the kernel score is significantly increased for samples that are close in the feature space of the input kernel. In this way the kernel can take advantage from both the locality in the feature space and the long-range extrapolation ability of the global input kernel.

We first construct a kernel to capture the locality information with any kernel function; such a family of kernels takes inspiration from the RBF kernel, substituting the Euclidean distance with the distance in the feature space.

$$k^{exp}(x, x') = \exp\left(-\frac{||\Phi(x) - \Phi(x')||^2}{\sigma}\right) \qquad \sigma > 0$$

where Φ is a mapping between the input space $\mathbb{R}^p$ and the feature space $\mathcal{F}$. The feature space distance $||\Phi(x) - \Phi(x')||^2$ is dependent on the choice of kernel (see (3)):

$$||\Phi(x) - \Phi(x')||^2 = k(x, x) + k(x', x') - 2 \cdot k(x, x').$$

The k^{exp} kernel can be obtained with the first operator, named $\mathcal{E}_\sigma$, that accepts a positive parameter σ applied on a kernel k producing $\mathcal{E}_\sigma\, k = k^{exp}$. Explicitly, the $\mathcal{E}_\sigma$ operator is defined as:

$$(\mathcal{E}_\sigma\, k)(x, x') = \exp\left(\frac{-k(x, x) - k(x', x') + 2k(x, x')}{\sigma}\right) \quad \sigma > 0. \qquad (4)$$

Note that $\mathcal{E}_\sigma\, k^{lin} = k^{rbf}$ so as a special case we have the RBF kernel. However, the kernels obtained with $\mathcal{E}_\sigma$ consider only the distance in the feature space without including explicitly the input kernel. For this reason $\mathcal{E}_\sigma\, k$ is not a quasi-local kernel.

In order to overcome the limitation of $\mathcal{E}_\sigma$ which completely drops the global information, the idea is to weight the input kernel with the local information to obtain a real quasi-local kernel. So we include explicitly the input kernel in the output of the following operator:

$$(\mathcal{P}_\sigma\, k)(x, x') = k(x, x') \cdot (\mathcal{E}_\sigma k)(x, x') \qquad \sigma > 0.$$

Observing that the $\mathcal{E}_\sigma\, k$ kernel can assume values only between 0 and 1 (since it is an exponential with negative exponent) and that the higher the distance in the feature space between samples the lower the value of the $\mathcal{E}_\sigma\, k$ kernel, the idea of $\mathcal{P}_\sigma$ is to exponentially penalize the basic kernel k with respect to the feature space distance between x and x'.

An opposite possibility is to amplify the values of input kernels in the cases in which the samples contain local information. This can be done simply by adding the $\mathcal{E}_\sigma\, k$ kernel to the input one.

$$(\mathcal{S}_\sigma\, k)(x,x') = k(x,x') + (\mathcal{E}_\sigma k)(x,x') \qquad \sigma > 0.$$

However, since $\mathcal{E}_\sigma$ gives kernels that can assume at most the value of 1 while the input kernel in the general case does not have an upper bound, it is reasonable to weight the $\mathcal{E}_\sigma$ operator with a constant reflecting the order of magnitude of the values that the input kernel can assume in the training set. We call this parameter η and the new operator is:

$$(\mathcal{S}_{\sigma,\eta}\, k)(x,x') = k(x,x') + \eta \cdot (\mathcal{E}_\sigma k)(x,x') \quad \sigma > 0, \eta \geq 0.$$

A different formulation of the $\mathcal{P}_\sigma$ operator that maintains the product form but adopts the idea of amplifying the local information is:

$$(\mathcal{PS}_\sigma\, k)(x,x') = k(x,x')\left[1 + (\mathcal{E}_\sigma k)(x,x')\right] \quad \sigma > 0, \eta \geq 0.$$

Also in this case the parameter η that controls the weight of the $\mathcal{E}_\sigma\, k$ kernel is introduced:

$$(\mathcal{PS}_{\sigma,\eta}\, k)(x,x') = k(x,x')\left[1 + \eta \cdot (\mathcal{E}_\sigma k)(x,x')\right] \quad \sigma > 0, \eta \geq 0.$$

The quasi-local kernels are more complicated then the corresponding input kernels, since it is necessary to evaluate $k(x,x)$, $k(x',x')$, $k(x,x')$ and to perform a couple of addition/multiplication operation and an exponentiation instead of the evaluation of $k(x,x')$ only. However, this is a constant computational overhead in the kernel evaluation phase, that does not affect the complexity of the SVM algorithm either in the training or in the testing phase.

Intuitively all the kernels produces by $\mathcal{S}_\sigma$, $\mathcal{S}_{\sigma,\eta}$, $\mathcal{PS}_\sigma$ and $\mathcal{S}_{\sigma,\eta}$ are quasi-local since they combine the original kernel with the locality information in its feature space. We will formalise this in subsection 3.4, while in the following subsection we will prove that the operators preserve the PD property of the input kernel.

3.1 The operators preserve the PD property of the kernels

We recall three well-known properties of PD kernels (for a comprehensive discussion of PD kernels refer to [28] or [9]):

Proposition 1 (Some properties of PD kernels)**.**

(i) *the class of PD kernels is a convex cone, i.e. if $\alpha_1, \alpha_2 \geq 0$ and k_1, k_2 are PD kernels then $\alpha_1 k_1 + \alpha_2 k_2$ is a PD kernel;*

(ii) *the class of PD kernels is closed under pointwise convergence, i.e. if* $k(x, x') := \lim_{n\to\infty} k_n(x, x')$ *exists for all* x, x', *then* k *is a PD kernel;*

(iii) *the class of PD kernels is closed under pointwise product, i.e. if* k_1, k_2 *are PD kernels, then* $(k_1k_2)(x, x') := k_1(x, x') \cdot k_2(x, x')$ *is a PD kernel.*

The introduced operators preserve the PD property of the kernels on which they are applied, as stated in the following theorem.

Theorem 3. *If* k *is a PD kernel, then* $\mathcal{O}\, k$ *with* $\mathcal{O} \in \{\mathcal{E}_\sigma,\, \mathcal{P}_\sigma,\, \mathcal{S}_\sigma,\, \mathcal{S}_{\sigma,\eta},\, \mathcal{PS}_\sigma,\, \mathcal{PS}_{\sigma,\eta}\}$ *is a PD kernel.*

Proof. It is straightforward to see that, for a PD kernel k, all the kernels resulting from the introduced operators can be obtained using properties (i) and (iii) of Proposition 1, provided that $\mathcal{E}_\sigma\, k$ is a PD kernel. So the only thing that remains to prove is that $\mathcal{E}_\sigma\, k$ is PD. Decomposing the definition of $(\mathcal{E}_\sigma\, k)(x, x')$ into three exponential functions we obtain:

$$(\mathcal{E}_\sigma\, k)(x, x') = \exp\left(\tfrac{2k(x,x')}{\sigma}\right) \exp\left(\tfrac{-k(x,x)}{\sigma}\right) \exp\left(\tfrac{-k(x',x')}{\sigma}\right)$$

that can be written as:

$$(\mathcal{E}_\sigma\, k)(x, x') = (\mathcal{O}_e\, 2k/\sigma)(x, x') \cdot f(x)f(x')$$

where $\mathcal{O}_e\, 2k/\sigma$ is the exponentiation of the $2k/\sigma$ kernel, and f is a real valued function such that $f(x) = \exp(-k(x, x)/\sigma)$. The first term is the exponentiation of a kernel multiplied by a non-negative constant and, since the kernel exponentiation can be seen as the limit of the series expansion of the exponential function which is the infinite sum of polynomial kernels, for property (ii) we conclude that $\mathcal{O}_e\, 2k/\sigma$ is a PD kernel. Moreover, recalling from the definition of PD kernels, that the product $f(x)f(x')$ is a PD kernel for all the real-valued functions f defined in the input space [9] we conclude that $\mathcal{E}_\sigma\, k$ is a PD kernel. □

Obviously, if the input of $\mathcal{E}_\sigma$ is a non PD kernel, also the resulting function cannot be, in the general case, a PD kernel since the exponentiation operator is valid only for PD kernels. So, in the case of the sigmoidal kernel as input kernel, the resulting kernel is still not ensured to be PD.

3.2 Properties of the operators

In order to understand how the operators modify the original feature space of the input kernel we study the distances in the feature space of the quasi-local kernels. The new feature space introduced by kernels produced by the operators is denoted with $\mathcal{F}_\mathcal{O}$, the corresponding mapping function with $\Phi_\mathcal{O}$ and the distance between two input points mapped in $\mathcal{F}_\mathcal{O}$ with $dist_{\mathcal{F}_\mathcal{O}}(x, x') = m(\Phi_\mathcal{O}(x), \Phi_\mathcal{O}(x'))$ where m is a metric in $\mathcal{F}_\mathcal{O}$. Applying the kernel trick for distances, we can express the squared distances in $\mathcal{F}_\mathcal{O}$ as:

$$dist^2_{\mathcal{F}_\mathcal{O}}(x, x') = \|\Phi_\mathcal{O}(x) - \Phi_\mathcal{O}(x')\|^2 = (\mathcal{O}\, k)(x, x) + (\mathcal{O}\, k)(x', x') - 2(\mathcal{O}\, k)(x, x'). \quad (5)$$

For $\mathcal{O} = \mathcal{E}_\sigma$, since it is clear that $dist_{\mathcal{F}}(x,x) = 0$ for every x, we can derive $dist_{\mathcal{F}_{\mathcal{E}_\sigma}}$ as follows:

$$\begin{aligned} dist^2_{\mathcal{F}_{\mathcal{E}_\sigma}}(x,x') &= \exp\left(-\tfrac{dist^2_{\mathcal{F}}(x,x)}{\sigma}\right) + \exp\left(-\tfrac{dist^2_{\mathcal{F}}(x',x')}{\sigma}\right) - 2\exp\left(-\tfrac{dist^2_{\mathcal{F}}(x,x')}{\sigma}\right) = \\ &= 2\left[1 - exp\left(-\tfrac{dist^2_{\mathcal{F}}(x,x')}{\sigma}\right)\right]. \end{aligned} \tag{6}$$

Note that $dist^2_{\mathcal{F}_{\mathcal{E}_\sigma}}(x,x') \leq 2$ for every pair of samples, and so the distances in $\mathcal{F}_{\mathcal{E}_\sigma}$ are bounded even if they are not bounded in $\mathcal{F}$.

Substituting $\mathcal{P}_\sigma$, $\mathcal{S}_{\sigma,\eta}$ and $\mathcal{PS}_{\sigma,\eta}$ in equation (5), an taking into account equation (6), the distances in $\mathcal{F}_{\mathcal{O}}$ for the quasi-local kernels are:

$$\begin{aligned} dist^2_{\mathcal{F}_{\mathcal{P}_\sigma}}(x,x') &= dist^2_{\mathcal{F}}(x,x') + k(x,x')\, dist^2_{\mathcal{F}_{\mathcal{E}_\sigma}}(x,x'); \\ dist^2_{\mathcal{F}_{\mathcal{S}_{\sigma,\eta}}}(x,x') &= dist^2_{\mathcal{F}}(x,x') + \eta \cdot dist^2_{\mathcal{F}_{\mathcal{E}_\sigma}}(x,x'); \\ dist^2_{\mathcal{F}_{\mathcal{PS}_{\sigma,\eta}}}(x,x') &= (1+\eta)\, dist^2_{\mathcal{F}}(x,x') + \eta \cdot k(x,x')\, dist^2_{\mathcal{F}_{\mathcal{E}_\sigma}}(x,x') = \\ &= dist^2_{\mathcal{F}}(x,x') + \eta \cdot dist^2_{\mathcal{F}_{\mathcal{P}_\sigma}}(x,x'). \end{aligned} \tag{7}$$

We can notice that the distances in $\mathcal{F}_{\mathcal{E}_\sigma}$ and in $\mathcal{F}_{\mathcal{S}_{\sigma,\eta}}$ do not contain explicitly the kernel function but they are based only on the distances in $\mathcal{F}$. So we can further analyse the behaviour of the distances in $\mathcal{F}_{\mathcal{E}_\sigma}$ and $\mathcal{F}_{\mathcal{S}_{\sigma,\eta}}$ with the following proposition.

Proposition 2. *The operators $\mathcal{E}_\sigma$ and $\mathcal{S}_{\sigma,\eta}$ preserve the ordering on distances in $\mathcal{F}$. Formally*

$$dist_{\mathcal{F}}(x,x') < dist_{\mathcal{F}}(x,x'') \Rightarrow dist_{\mathcal{F}_{\mathcal{O}}}(x,x') < dist_{\mathcal{F}_{\mathcal{O}}}(x,x'')$$

for $\mathcal{O} \in \{\mathcal{E}_\sigma, \mathcal{S}_{\sigma,\eta}\}$ and for every sample x, x', x''.

Proof. It follows directly from the observations that $dist_{\mathcal{F}_{\mathcal{E}_\sigma}}(x,x')$ and $dist_{\mathcal{F}_{\mathcal{S}_{\sigma,\eta}}}(x,x')$ are defined with strictly increasing monotonic functions, equations (6) and the second equation in (7) respectively, and that $dist_{\mathcal{F}}$ is always non-negative. □

3.3 The operator parameters

There are two parameters for the operators on kernels through which we obtain the quasi-local kernels: σ, which is present in $\mathcal{E}_\sigma$ and consequently in all the operators, and η, which is present in $\mathcal{S}_{\sigma,\eta}$ and $\mathcal{PS}_{\sigma,\eta}$ (notice that $\mathcal{S}_\sigma$ and $\mathcal{PS}_\sigma$ can be seen as special cases of $\mathcal{S}_{\sigma,\eta}$ and $\mathcal{PS}_{\sigma,\eta}$ with $\eta = 1$).

The role of these two parameters will be illustrated in the next section. Here we propose some data-dependent settings that do not require cross validation on the training set. In other words, the σ and η parameters are chosen on the basis of statistical properties of the datasets rather then tuning them with an expensive model selection phase. This choice privileges the reduction of the computational effort of applying the quasi-local kernels on an input kernel instead of the potential gain in terms of classification accuracy provided by model selection.

The dataset-dependent estimation of σ take inspiration from the γ^{rbf} estimation, since σ and γ^{rbf} play a similar role of controlling the width of the kernel. However, differently from the k^{rbf} kernel, the $\mathcal{E}_\sigma$ operator uses distances in the feature space $\mathcal{F}$ (except for the special case $k = k^{lin}$). So two families of estimated values for σ are possible: one based on the distances in the feature space (we call this family $\sigma^{\mathcal{F}}$) and the other based on the distances in the input space (we call this second family $\sigma^{\mathbb{R}^p}$). In particular, denoting with $q_h[\|x - x'\|^{\mathcal{Z}}]$ the h quantile of the distribution of the distance in the $\mathcal{Z}$ space between every pair of points x, x' in the training dataset, we consider two possibilities for σ: $\sigma^{\mathbb{R}^p}_{.1} = q_{.1}[\|x - x'\|^{\mathbb{R}^p}]$ and $\sigma^{\mathcal{F}}_{.1} = q_{.1}[\|x - x'\|^{\mathcal{F}}]$. In the following we will omit the "$\mathbb{R}^p$" apex for denoting the input space, so $\sigma^{\mathbb{R}^p}_{.1}$ will be denoted simply by $\sigma_{.1}$.

For η we choose a broad spectrum of possibility:

$$\eta_{.1} = q_{.1}[\|x - x'\|^{\mathbb{R}^p}] \qquad \eta_{.1r} = \sqrt{\frac{\eta_{.1}}{2}} \qquad \eta^{\mathcal{F}}_{.1} = q_{.1}[\|x - x'\|^{\mathcal{F}}]$$

$$\eta_{.5} = q_{.5}[\|x - x'\|^{\mathbb{R}^p}] \qquad \eta_{.5r} = \sqrt{\frac{\eta_{.5}}{2}} \qquad \eta^{\mathcal{F}}_{.5} = q_{.5}[\|x - x'\|^{\mathcal{F}}]$$

$$\eta_{.9} = q_{.9}[\|x - x'\|^{\mathbb{R}^p}] \qquad \eta_{.9r} = \sqrt{\frac{\eta_{.9}}{2}} \qquad \eta^{\mathcal{F}}_{.9} = q_{.9}[\|x - x'\|^{\mathcal{F}}]$$

Note that also for η we omit the apex $\mathbb{R}^p$, similarly to the convention for $\sigma_{.1}$.

3.4 Quasi-local kernels

In this section we formally introduce the notion of quasi-local kernels showing that kernels produced by the $\mathcal{S}_\sigma$, $\mathcal{S}_{\sigma,\eta}$, $\mathcal{PS}_\sigma$ and $\mathcal{S}_{\sigma,\eta}$ are quasi-local kernels. Firstly we introduce the concept of locality with respect to a function:

Definition 2. *Given a PD kernel k with implicit mapping function $\Phi : \mathbb{R}^p \mapsto \mathcal{F}$ (namely $k(x, x') = \langle\Phi(x), \Phi(x')\rangle$), and a function $\Psi : \mathbb{R}^p \mapsto \mathcal{F}_\Psi$, k* is local with respect to Ψ *if there exists a function $\Omega : \mathcal{F}_\Psi \mapsto \mathcal{F}$ such that the following holds:*

1. *$\langle\Phi(x), \Phi(x_i)\rangle = \langle\Omega(\Psi(x)), \Omega(\Psi(x_i))\rangle$*
2. *$\lim_{\|u - v_i\|_{\mathcal{F}_\Psi} \to \infty} \langle\Omega(u), \Omega(v_i)\rangle = c_i$ with $u = \Psi(x)$, $v_i = \Psi(x_i)$ for some $x, x_i \in \mathbb{R}^p$ and c_i constant and not depending on u.*

In other terms, the notion of locality referred to samples in input space (Definition 1), is modified here in order to consider the locality in any space accessible from the input space through a corresponding mapping function. Notice that, as particular cases, we have that every local kernel is local with respect to the identity function and with respect to its own implicit mapping function.

With the next theorem we see that the $\mathcal{E}_\sigma$ formally respect the idea of producing kernels that are local with respect to the feature space of the input kernel.

Theorem 4. *If k is a PD kernel with the implicit mapping function $\Phi : \mathbb{R}^p \mapsto \mathcal{F}$, then $\mathcal{E}_\sigma\, k$ is local with respect to Φ.*

Proof. We have already shown that $\mathcal{E}_\sigma\, k$ is a PD kernel given that k is a PD kernel (see Theorem 3). It remains to show that $\mathcal{E}_\sigma\, k$ is local with respect to Φ.

First we need to show that (Definition 2 point 1), denoted with $\Phi' : \mathbb{R}^p \mapsto \mathcal{F}'$ the implicit mapping function of $\mathcal{E}_\sigma\, k$, there exists a function $\Omega : \mathcal{F} \mapsto \mathcal{F}'$ such that $\Phi'(x) = \Omega(\Phi(x))$.

Taking as $\Omega : \mathcal{F} \mapsto \mathcal{F}'$ the implicit mapping of the kernel $\exp\left(-\frac{\|u-v_i\|}{\sigma}\right)$ with $u = \Phi(x)$, $v_i = \Phi(x_i)$ with $x, x_i \in \mathbb{R}^p$ we have

$$\langle \Omega(u), \Omega(v_i) \rangle = \exp\left(-\frac{\|u - v_i\|}{\sigma}\right). \tag{8}$$

Using the hypothesis on u and v_i it becomes:

$$\exp\left(-\frac{\|\Phi(x) - \Phi(x_i)\|}{\sigma}\right) = \langle \Omega(\Phi(x)), \Omega(\Phi(x_i)) \rangle. \tag{9}$$

The implicit mapping function of $\mathcal{E}_\sigma\, k$ is Φ' and so

$$\langle \Phi'(x), \Phi'(x_i) \rangle = (\mathcal{E}_\sigma\, k)(x, x_i) \tag{10}$$

Moreover since $(\mathcal{E}_\sigma\, k)(x, x_i) = \exp\left(-\frac{\|\Phi(x)-\Phi(x_i)\|}{\sigma}\right)$ for definition of $\mathcal{E}_\sigma$ (see equation (4)), substituting equation (9) into (10) we conclude that

$$\langle \Phi'(x), \Phi'(x_i) \rangle = \langle \Omega(\Phi(x)), \Omega(\Phi(x_i)) \rangle.$$

Second, we need to show that (Definition 2 point 2) $\langle \Omega(u), \Omega(v_i) \rangle \to c_i$ with c_i constant for $\|\Omega(u) - \Omega(v_i)\| \to \infty$. From the equation (8), it is clear that, as the distance between $u = \Phi(x)$ and $v_i = \Phi(x_i)$ tend to infinity, the kernel value is equal to the constant 0 regardless of x. □

Now we can define the quasi-locality property of a kernel.

Definition 3 (Quasi-local kernel). *A PD kernel k is a quasi-local kernel if $k = f(k^{inp}, k^{loc})$ where k^{inp} is a PD kernel with implicit mapping function $\Phi : \mathbb{R}^p \mapsto \mathcal{F}$, k^{loc} is a PD kernel which is local with respect to Φ and f is a function involving legal and non trivial operations on PD kernels.*

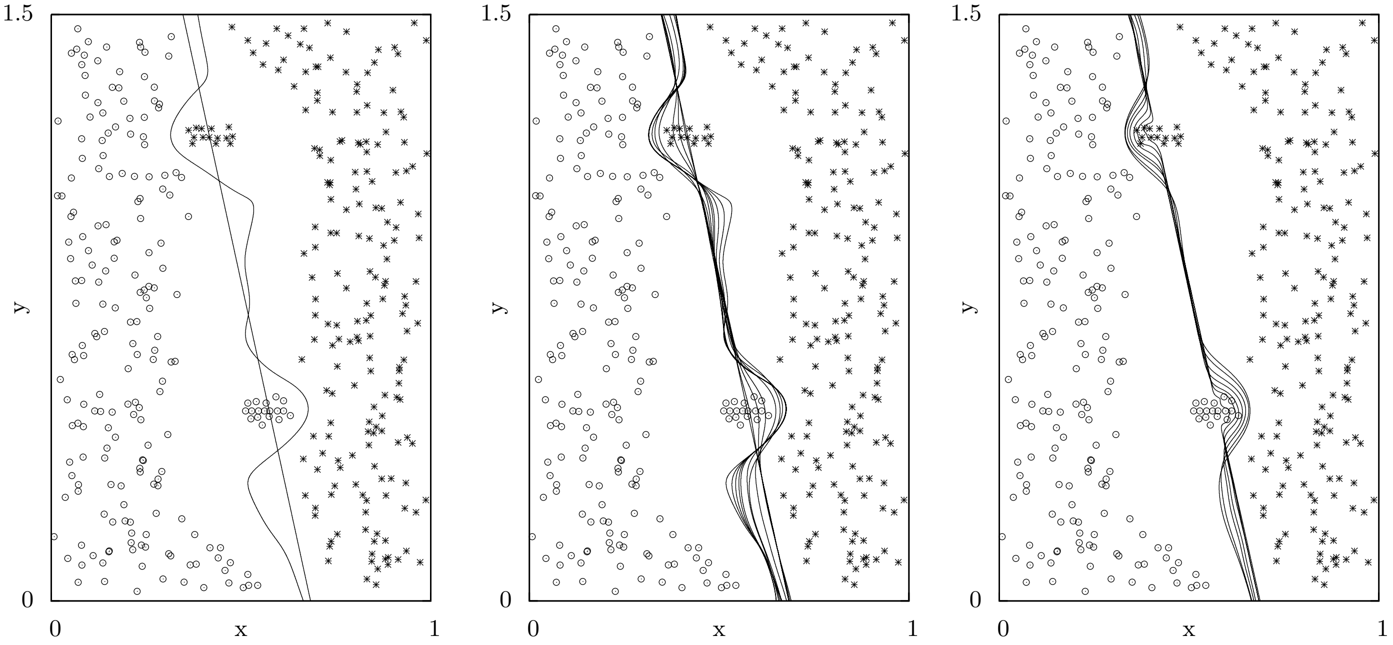
For legal operations on kernels we mean operations preserving the PD property. For non trivial operations we intend operations that always maintain the influence of all the input kernels in the output kernel; more precisely a function $f(k_1, k_2)$ does not introduce trivial operations if there exists two kernels k' and k'' such that $f(k', k_2) \neq f(k_1, k_2)$ and $f(k_1, k'') \neq f(k_1, k_2)$. Notice that the k^{inp} kernel of the definition corresponds to the input kernel of the operator that produces the quasi-local kernel k.

Theorem 5. *If k is a PD kernel, then $\mathcal{S}_\sigma\, k$, $\mathcal{S}_{\sigma,\eta}\, k$, $\mathcal{PS}_\sigma\, k$ and $\mathcal{S}_{\sigma,\eta}\, k$ are quasi-local kernels.*

Proof. Theorem 4 already states that $\mathcal{E}_\sigma\, k$ is a PD kernel which is local with respect to the implicit mapping function Φ of the kernel k which is PD for hypothesis. It is easy to see that all the kernels resulting from the introduced operators can be obtained using properties (i) and (iii) of Proposition 1 starting from the two PD kernels k and $\mathcal{E}_\sigma\, k$, and thus $\mathcal{S}_\sigma\, k$, $\mathcal{S}_{\sigma,\eta}\, k$, $\mathcal{PS}_\sigma\, k$ and $\mathcal{S}_{\sigma,\eta}\, k$ are PD kernels obtained with legal operations. Moreover the properties (i) and (iii) of Proposition 1 introduce multiplications and sums between kernels and between kernels and constant. The sums introduced by the operators are always non trivial because they always consider positive addends, and so it is for the multiplications because they never consider null factors (the introduced constants are non null for definition). □

Both quasi-local kernels and KNNSVM classifiers are based on the notion of locality in the feature space. However, two main theoretical differences can be found between them. The first is that in KNNSVM locality is included directly, considering only the points that are close to the testing point, while for the quasi-local kernels the information of the input kernel is only balanced with the local information. The second consideration concerns the fact that KNNSVM has a variable but hard boundary between the local and non local points, while $\mathcal{S}_{\sigma,\eta}$ and $\mathcal{PS}_{\sigma,\eta}$ produce kernels whose locality decreases exponentially but in a continuous way.

4 Intuitive behaviour of quasi-local kernels



(a) k^{lin} and k^{rbf}

(b) $\mathcal{S}_{\sigma,\eta}\, k^{lin}$ varying η

(c) $\mathcal{S}_{\sigma,\eta}\, k^{lin}$ varying σ

Figure 1: The separating hyperplanes for a two-feature hand-built artificial datasets defined by the application of the SVM (all with $C = 3$) with (a) linear kernel k^{lin} and RBF kernel k^{rbf} (with $\gamma^{rbf} = 150$), (b) the $\mathcal{S}_{\sigma,\eta}\, k^{lin}$ quasi-local kernel with fixed σ ($\sigma = 1/150 = 1/\gamma^{rbf}$) and variable η ($\eta = 10^6, 50, 10, 1, 0.5, 0.1, 0.05, 0.03, 0.01, 0.005, 0.001, 0.000001$), and (c) the $\mathcal{S}_{\sigma,\eta}\, k^{lin}$ quasi-local kernel with fixed η ($\eta = 0.05$) and variable σ ($\sigma = 1/5000, 1/2000, 1/1000, 1/500, 1/300, 1/150, 1/100$).

The operators on kernels defined in the previous section aim to modify the behaviour of an input kernel k in order to produce a kernel more sensitive to local information in the feature space, maintaining however the original behaviour for regions in which the locality is not important. In addition the η and σ parameters control the balance between the input kernel k and its local reformulation $\mathcal{E}_\sigma\, k$, in other words the effects of the local information.

These intuitions are highlighted in Figure 1 with an example that illustrates the effects of the $\mathcal{S}_{\sigma,\eta}$ operator on the linear kernel k^{lin} using a two-feature hand-built artificial dataset. We chose the linear kernel because its effects are easily recognizable in plots. The transformed kernel is:

$$(\mathcal{S}_{\sigma,\eta}\, k^{lin})(x,x') = k^{lin}(x,x') + \eta \cdot (\mathcal{E}_\sigma\, k^{lin})(x,x') = k^{lin}(x,x') + \eta \cdot k^{rbf}(x,x') \qquad (11)$$

with $\gamma^{rbf} = 1/\sigma$. So the $\mathcal{S}_{\sigma,\eta}$ operator on the k^{lin} kernel gives a linear combination of k^{lin} and k^{rbf}. Figure 1(a) shows the behaviour of only the global term k^{lin} and of only the local term $\mathcal{E}_\sigma\, k^{lin} = k^{rbf}$. Figure 1(b) illustrates what happens when the local and the global terms are balanced with different values of η and a fixed σ. Figure 1(c) shows the behaviour of the separating hyperplane with a fixed balancing factor η but varying the σ parameter.

The η parameter regulates the influence on the separating hyperplane of the local term of the quasi-local kernel. In fact, in Figure 1(b), we see that all the planes lie between the input kernel (k^{lin}, obtained with $\eta \to 0$ from $\mathcal{S}_{\sigma,\eta}\, k^{lin}$) and the local reformulation of the same kernel (obtained with $\eta = 10^6$ from $\mathcal{S}_{\sigma,\eta}\, k^{lin}$ which behaves as k^{rbf} since the high value of η partially hides the effect of the global term). Moreover, since σ is low, the modifications induced by different values of η are global, influencing all the regions of the separating hyperplane.

We can observe in Figure 1(c), on the other hand, that σ regulates the magnitude of the distortion from the linear hyperplane for the region containing points close to the plane itself. The σ parameter in the $\mathcal{E}_\sigma\, k^{lin}$ term of $\mathcal{S}_{\sigma,\eta}\, k^{lin}$ has a similar role to the K parameter in the local SVM approach (i.e. it regulates the range of the locality), even though K defines an hard boundary between local and non local points instead of a negative exponential one. It is important to underline that in the central region of the dataset the separating hyperplane remains linear, highlighting that the kernel resulting from the $\mathcal{S}_{\sigma,\eta}$ operator is able to modify the input kernel only where the information is local.

The example simply illustrates the intuition behind the proposed family of quasi-local kernels, and in particular how the input kernel behaviour is maintained for the regions in which the information is not local, so it is not important here to analyse the classification accuracy. However, kernels that are sensitive to important local information but retain properties of global input kernels, can also be obtained from very elaborated and well tuned kernels defined on high-dimensionality input spaces. Notice that in this toy example we are considering locality in input space since, using the linear kernel, the kernel trick is not applied. In the following two sections we investigate the accuracy performances of the quasi-local kernels in a number of real datasets using a data-dependent method of choosing η and σ parameters.

5 Experiment 1

The first experiment consists in the evaluation of the accuracy of the quasi-local kernels with different but systematic choices of σ and η parameters on quite a large number of datasets. The aim of the experiment is to understand which quasi-local kernels achieve better results with respect to the four reference input kernels. The implementation of the classifiers used in this work is based on the LibSVM library [7] version 2.84.

5.1 Experimental procedure

Table 1 lists the 20 datasets from different sources and scientific fields used in this experiment; some datasets are multiclass and the number of features ranges from 2 to 7129. All the datasets are collected and freely available online at the homepage of LibSVM [7]. They are small- or medium-size datasets permitting the Leave-One-Out (Loo) evaluation strategy for the classifiers which is a special case of the Cross Validation technique [31] consisting in testing every sample of the dataset with the classifier built on all the other points and averaging the correct classification cases with the dataset cardinality. The Loo is computationally expensive, but as shown in [32] it gives a good bound on the expectation of error for SVM. We denote with $A^{Loo}(\omega, D)$ the Loo accuracy obtained by the ω classifier on the D dataset.

The reference input kernels for the quasi-local operators considered are the linear kernel k^{lin}, the polynomial kernel k^{pol}, the radial basis function kernel k^{rbf} and the sigmoidal kernel k^{sig}. The quasi-local kernels we tested are those resulting from the application of the $\mathcal{E}_\sigma$, $\mathcal{P}_\sigma$, $\mathcal{S}_\sigma$, $\mathcal{S}_{\sigma,\eta}$, $\mathcal{PS}_\sigma$ and $\mathcal{PS}_{\sigma,\eta}$ operators on the reference input kernels. We also evaluated the accuracy of the KNNSVM classifier with the same reference input kernels.

Dataset name	source	# classes	train. size	# features
iris	UCI [1]	3	150	4
wine	UCI [1]	3	178	13
leukemia	TG99 [14]	2	38	7129
glass	UCI [1]	6	214	9
heart	Statlog [18]	2	270	13
sonar	UCI [1]	2	208	60
liver-disorders	UCI [1]	2	345	6
ionosphere	UCI [1]	2	351	34
svmguide2	CWH03a [16]	3	391	20
breast-cancer	UCI [1]	2	683	10
vowel	UCI [1]	11	528	10
fourclass	TKH96a [15]	2	862	2
australian	Statlog [18]	2	690	14
diabetes	UCI [1]	2	768	8
vehicle	Statlog [18]	4	846	18
splice	UCI [1]	2	1000	60
german-numer	Statlog [18]	2	1000	24
a1a	UCI [1]	2	1605	123
w1a	JP98a [23]	2	2477	300
segment	Statlog [18]	7	2310	19

Table 1: The 20 datasets for Experiment 1.

For the reference input kernels we set the parameters in the following way: for the polynomial kernel $d = 3$, $\gamma^{pol} = p$ and $r^{pol} = 0$, for the sigmoidal kernel $r^{sig} = 0$. For γ^{rbf} we used the estimation described in section 2 based on the distribution of the Euclidean distance between the samples of the datasets. For the quasi-local kernels obtained with the operators, the parameters to set are η and σ and we use the data-dependent estimation described in subsection 3.3; in particular we set $\eta \in \{\eta_{.1},\ \eta_{.5},\ \eta_{.9},\ \eta_{.1r},\ \eta_{.5r},\ \eta_{.9r},\ \eta^{\mathcal{F}}_{.1},\ \eta^{\mathcal{F}}_{.5},\ \eta^{\mathcal{F}}_{.9}\}$ and $\sigma = \sigma_{.1}$ without extensively use $\sigma = \sigma^{\mathcal{F}}_{.1}$ since preliminary tests gave bad results for this setting. The C parameter of SVM is set to 1. Finally, the value of K in the KNNSVM classifier is automatically chosen on the training set between $\mathcal{K} = \{1, 3, 5, 7, 9, 11, 15, 23, 39, 71, 135, 263, 519, 1031\}$ (the first 5 odd natural numbers followed by the ones obtained with a base-2 exponential increment from 9) as described in section 2.2. We also evaluate KNNSVM without automatic K choice, i.e. with a-priori fixed values of $K \in \mathcal{K}$.

In order to compare the quasi-local kernel results with the best potentially achievable results of the KNNSVM locality-based classifier, we compute the K that maximizes the Loo accuracy, denoted as K^*. Formally, K^* is:

$$K^* = \operatorname*{argmin}_{K \in \mathcal{K}} A^{Loo}(K\text{NNSVM}, D).$$

So K^*NNSVM has the best Loo accuracy among the local SVM with fixed value of K, but we remark that this a-posteriori choice of the best K for KNNSVM to obtain K^*NNSVM is not a classification method because it uses information of test samples.

As stated above, the classification capability of an SVM with a kernel k on a specific dataset D, is evaluated considering the Loo accuracy, denoted by $A^{Loo}(SVM_k, D)$ or simply, assuming that we use a kernel always through the SVM algorithm, $A^{Loo}(k, D)$. To assess the accuracy difference between two kernels in the same dataset, we can calculate the absolute difference between the Loo accuracies (both expressed in percentage):

$$\Delta_{A^{Loo}}(k_1, k_2, D) = (A^{Loo}(k_1, D) - A^{Loo}(k_2, D)) \cdot 100$$

In order to make the $\Delta_{A^{Loo}}$ independent from the absolute values of the accuracies, we also introduce the relative percentage difference of Loo accuracy:

$$\delta_{A^{Loo}}(k_1, k_2, D) = \frac{\Delta_{A^{Loo}}(k_1, k_2, D)}{A^{Loo}(k_2, D)}$$

Applying $\delta_{A^{Loo}}(k_1, k_2, D)$ (or $\Delta_{A^{Loo}}(k_1, k_2, D)$) on a considerable number of datasets D with $k_1 = \mathcal{O}\,k$ the kernel obtained with the application of the operator $\mathcal{O}$ on the input kernel k and $k_2 = k$, we can obtain a distribution of relative (or absolute) Loo accuracy differences between $\mathcal{O}\,k$ and k. On this distribution we compute descriptive statistics; the mean, the standard deviation (sd) and the skewness (skew). By means of box-plot diagrams, we show the median, the first quartile, the third quartile, the whiskers (i.e. the maximum value within the third quartile plus 1.5 times the interquartile range and the minimum value within the first quartile minus 1.5 times the interquartile range), and the extreme or outlier values (i.e. values that are over the third quartile plus 1.5 times the interquartile range or under the first quartile minus 1.5 times the interquartile range). Moreover, we define $\nu_{\delta>t}$

to be the percentage of datasets in which the relative percentage difference $\delta_{A^{Loo}}$ between $\mathcal{O}\,k$ and k is greater than a fixed threshold t:

$$\nu_{\delta>t} = \frac{|\{D_i \mid \delta_{A^{Loo}}(\mathcal{O}\,k, k, D_i) > t\}|}{n_D} \cdot 100 \quad i = 1, \ldots, n_D$$

where n_D is the total number of datasets considered. Similarly the percentage of datasets in which the relative percentage difference between $\mathcal{O}\,k$ and k is negative and lower than a threshold -t is:

$$\nu_{\delta<\text{-}t} = \frac{|\{D_i \mid \delta_{A^{Loo}}(\mathcal{O}\,k, k, D_i) < \text{-}t\}|}{n_D} \cdot 100 \quad i = 1, \ldots, n_D.$$

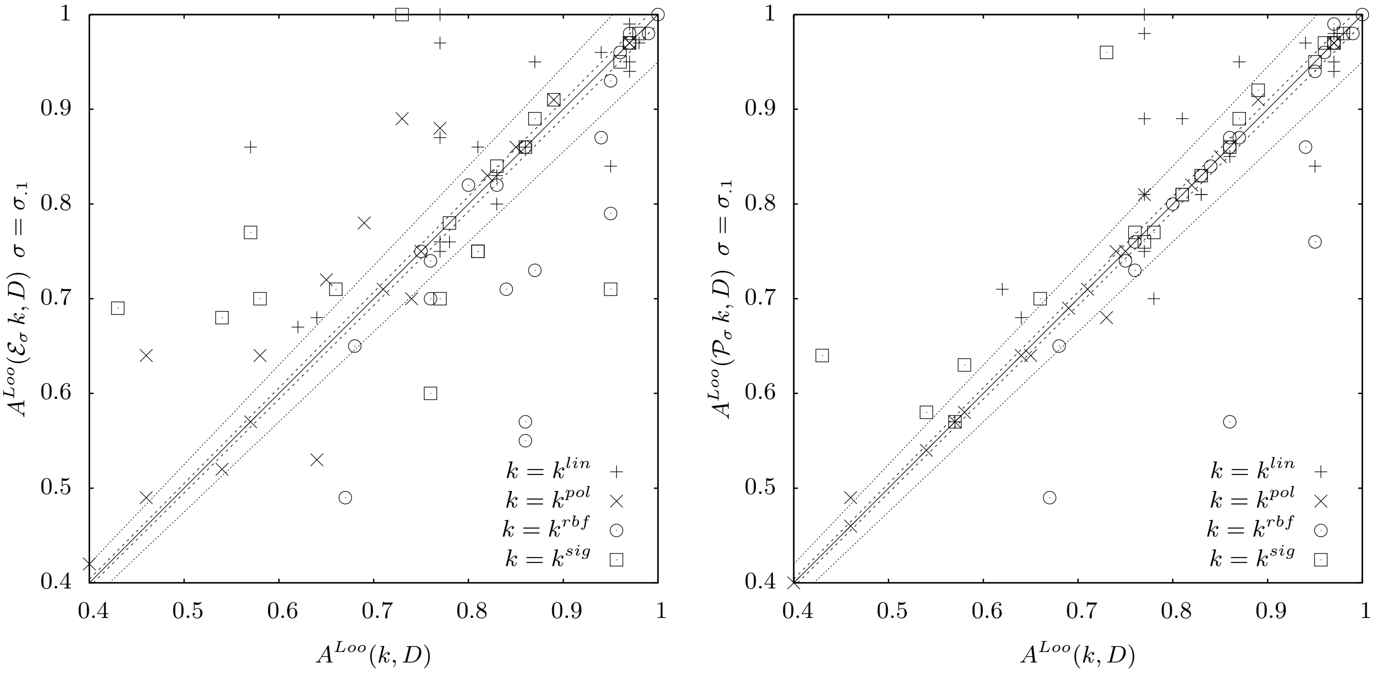
Representing the accuracy values of $\mathcal{O}\,k$ and k on a scatter plot, $\nu_{\delta>t}$ and $\nu_{\delta<\text{-}t}$ with $t = 0$ can be seen as the number of points lying over and under the bisector line ($y = x$), i.e. the number of datasets in which the quasi-local kernels perform better and worse with respect to the input kernels. With non-zero values for t, the graphical meaning of $\nu_{\delta>t}$ and $\nu_{\delta<\text{-}t}$ is the number of points lying over $y = x \cdot (1 + t/100)$ and under $y = x \cdot (1 - t/100)$.

5.2 Results

		segment				*sonar*				*vehicle*				*splice*			
	$\eta \backslash k$	k^{lin}	k^{pol}	k^{rbf}	k^{sig}	k^{lin}	k^{pol}	k^{rbf}	k^{sig}	k^{lin}	k^{pol}	k^{rbf}	k^{sig}	k^{lin}	k^{pol}	k^{rbf}	k^{sig}
$\mathcal{I}\,k$		.94	.89	.96	.89	.77	.64	.87	.76	.77	.46	.75	.66	.81	.54	.86	.81
$\mathcal{E}_\sigma\,k$		.96	.91	.96	.91	.87	.53	.73	.60	.75	.49	.75	.71	.86	.52	.55	.75
$\mathcal{P}_\sigma\,k$		.97	.91	.96	.92	.89	.64	.87	.77	.81	.46	.74	.70	.89	.54	.87	.81
$\mathcal{S}_\sigma\,k$		.97	.93	.97	.92	.84	.68	.87	.79	.79	.54	.76	.72	.85	.54	.86	.80
$\mathcal{S}_{\sigma,\eta}\,k$	$\eta_{.1}$	.97	.93	.97	.92	.88	.71	.91	.79	.78	.54	.77	.73	.89	.80	.89	.80
$\mathcal{S}_{\sigma,\eta}\,k$	$\eta_{.5}$	.97	.94	.97	.94	.89	.74	.90	.77	.82	.69	.79	.76	.89	.81	.89	.79
$\mathcal{S}_{\sigma,\eta}\,k$	$\eta_{.9}$	.97	.96	.97	.94	.89	.78	.90	.77	.82	.74	.80	.80	.89	.82	.88	.81
$\mathcal{S}_{\sigma,\eta}\,k$	$\eta_{.1r}$	.96	.92	.97	.92	.88	.70	.88	.78	.79	.52	.76	.72	.88	.57	.87	.80
$\mathcal{S}_{\sigma,\eta}\,k$	$\eta_{.5r}$	.97	.93	.97	.93	.88	.72	.89	.77	.79	.56	.77	.73	.89	.57	.87	.80
$\mathcal{S}_{\sigma,\eta}\,k$	$\eta_{.9r}$	.97	.94	.97	.93	.88	.71	.89	.80	.81	.65	.77	.74	.89	.58	.87	.81
$\mathcal{S}_{\sigma,\eta}\,k$	$\eta^{\mathcal{F}}_{.1}$	.97	.90	.97	.90	.88	.64	.88	.77	.78	.46	.76	.67	.89	.54	.86	.81
$\mathcal{S}_{\sigma,\eta}\,k$	$\eta^{\mathcal{F}}_{.5}$	.97	.90	.97	.90	.89	.64	.88	.77	.82	.47	.77	.69	.89	.54	.86	.80
$\mathcal{S}_{\sigma,\eta}\,k$	$\eta^{\mathcal{F}}_{.9}$	.97	.91	.97	.92	.89	.65	.88	.77	.82	.47	.77	.70	.89	.54	.86	.81
$\mathcal{PS}_\sigma\,k$		.97	.92	.97	.92	.89	.71	.88	.80	.83	.51	.76	.73	.88	.71	.88	.81
$\mathcal{PS}_{\sigma,\eta}\,k$	$\eta_{.1}$	.97	.92	.97	.92	.89	.79	.90	.76	.82	.52	.76	.72	.89	.84	.89	.80
$\mathcal{PS}_{\sigma,\eta}\,k$	$\eta_{.5}$	.97	.94	.98	.94	.89	.83	.90	.75	.81	.66	.79	.77	.89	.84	.89	.80
$\mathcal{PS}_{\sigma,\eta}\,k$	$\eta_{.9}$	.97	.95	.98	.95	.89	.87	.90	.81	.82	.71	.79	.80	.89	.84	.89	.79
$\mathcal{PS}_{\sigma,\eta}\,k$	$\eta_{.1r}$	.98	.91	.97	.92	.90	.72	.90	.79	.82	.50	.76	.71	.89	.82	.89	.79
$\mathcal{PS}_{\sigma,\eta}\,k$	$\eta_{.5r}$	.97	.93	.97	.93	.89	.73	.90	.77	.82	.53	.76	.73	.89	.83	.89	.78
$\mathcal{PS}_{\sigma,\eta}\,k$	$\eta_{.9r}$	.97	.94	.98	.93	.90	.74	.90	.79	.81	.57	.77	.74	.89	.83	.89	.79
$\mathcal{PS}_{\sigma,\eta}\,k$	$\eta^{\mathcal{F}}_{.1}$	.97	.90	.97	.90	.89	.65	.89	.77	.82	.46	.76	.67	.89	.56	.89	.81
$\mathcal{PS}_{\sigma,\eta}\,k$	$\eta^{\mathcal{F}}_{.5}$	.97	.90	.97	.90	.89	.67	.91	.78	.81	.46	.77	.69	.89	.60	.89	.81
$\mathcal{PS}_{\sigma,\eta}\,k$	$\eta^{\mathcal{F}}_{.9}$	.97	.91	.97	.92	.89	.67	.91	.77	.82	.47	.77	.71	.89	.65	.89	.81

Table 2: Experiment 1. Loo accuracy of input kernels $A^{Loo}(\mathcal{I}\,k, D)$ and of quasi-local kernels $A^{Loo}(\mathcal{O}\,k, D)$ on *segment*, *sonar*, *vehicle* and *splice* datasets.

Table 2 presents the Loo accuracy values for the *segment*, *sonar*, *vehicle* and *splice* datasets. These four datasets were arbitrarily selected for their representativeness. For space reasons, the accuracy results of the remaining datasets are available in the Additional Material. Formally the table shows the $A^{Loo}(\mathcal{O}\,k, D)$ values using $\mathcal{O} \in \{\mathcal{I}, \mathcal{E}_\sigma, \mathcal{P}_\sigma, \mathcal{S}_\sigma, \mathcal{S}_{\sigma,\eta}, \mathcal{PS}_\sigma, \mathcal{PS}_{\sigma,\eta}\}$ with $\sigma = \sigma_{.1}$ and $\eta \in \{\eta_{.1}, \eta_{.5}, \eta_{.9}, \eta_{.1r}, \eta_{.5r}, \eta_{.9r}, \eta^{\mathcal{F}}_{.1}, \eta^{\mathcal{F}}_{.5}, \eta^{\mathcal{F}}_{.9}\}$, $k \in \{k^{lin}, k^{pol}, k^{rbf}, k^{sig}\}$, $D \in \{segment, sonar, vehicle, splice\}$.



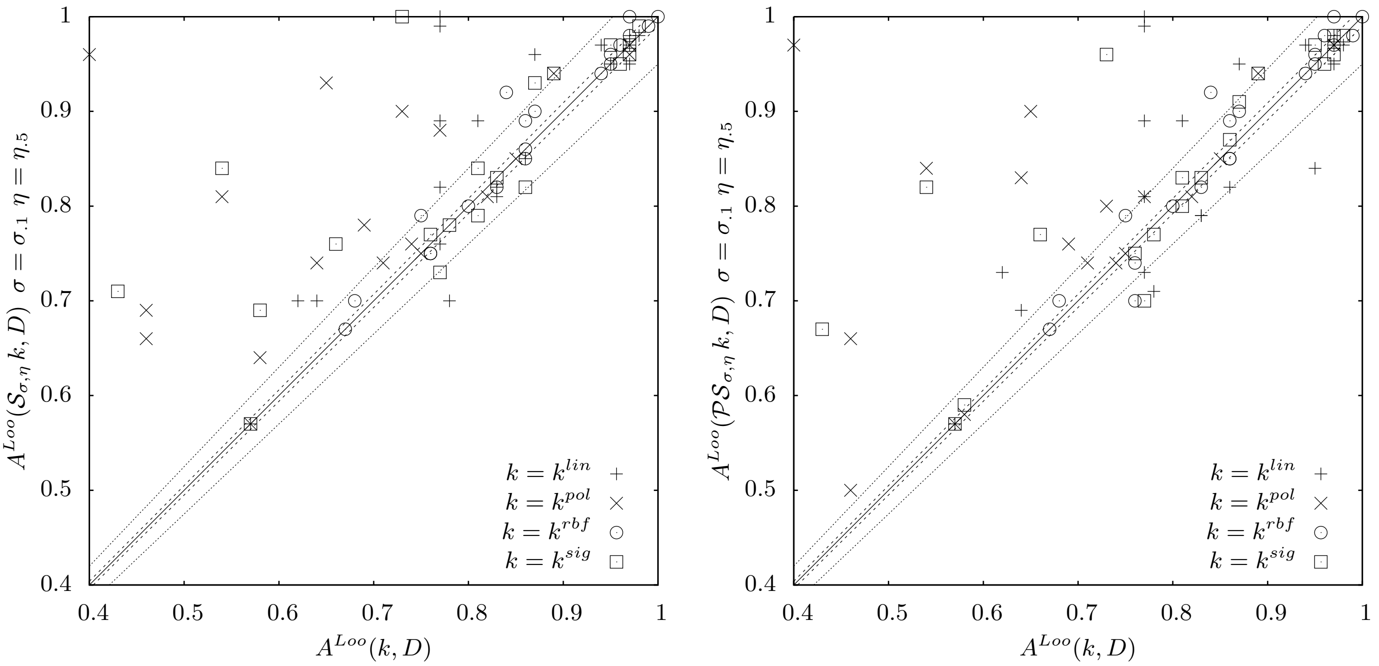
(a) $\mathcal{E}_\sigma\,k$ with $\sigma = \sigma_{.1}$ vs. k

(b) $\mathcal{P}_\sigma\,k$ with $\sigma = \sigma_{.1}$ vs. k

Figure 2: Experiment 1. Scatter plots for the Loo accuracy comparison between $\mathcal{E}_\sigma\,k$ and $\mathcal{P}_\sigma\,k$ quasi-local kernels and the corresponding input kernels k with $k \in \{k^{lin}, k^{pol}, k^{rbf}, k^{sig}\}$ for all the 20 datasets.

Considering all the 20 datasets of experiment 1, the scatter plots in Figure 2 and Figure 3 summarize the results for the $\mathcal{E}_\sigma\,k$, $\mathcal{P}_\sigma\,k$, $\mathcal{S}_{\sigma,\eta}\,k$ and $\mathcal{PS}_{\sigma,\eta}\,k$ quasi-local kernels with $\eta = \eta_{.5}$ and $\sigma = \sigma_{.1}$ comparing their Loo accuracies with those of the corresponding input kernels. Points over the bisector line mean accuracy improvements of quasi-local kernels with respect to the input kernels, the opposite for points under the bisector line. The dotted lines denote accuracy deviations of 1% and 5% from the bisector line, i.e. the limits defined by $\nu_{\delta>t}$ and $\nu_{\delta<\text{-}t}$ with $t = 1$ and $t = 5$.

Table 3 and the box-plots of Figure 4 report the statistics on the differences between the Loo accuracies of the quasi-local kernels and the corresponding input kernels. Table 3 presents the mean and the standard deviation (sd) of the distribution of the absolute differences $\Delta_{A^{Loo}}(\mathcal{O}\,k, k, D)$, and the mean, the standard deviation (sd) and the skewness (skew) of the distribution of the relative differences $\delta_{A^{Loo}}(\mathcal{O}\,k, k, D)$ for every dataset D



(a) $\mathcal{S}_{\sigma,\eta}\,k$ with $\sigma = \sigma_{.1}$ and $\eta = \eta_{.5}$ vs. k

(b) $\mathcal{PS}_{\sigma,\eta}\,k$ with $\sigma = \sigma_{.1}$ and $\eta = \eta_{.5}$ vs. k

Figure 3: Experiment 1. Scatter plots for the Loo accuracy comparison between $\mathcal{S}_{\sigma,\eta}\,k$ and $\mathcal{PS}_{\sigma,\eta}\,k$ quasi-local kernels and the corresponding input kernels k with $k \in \{k^{lin}, k^{pol}, k^{rbf}, k^{sig}\}$ for all the 20 datasets.

of Experiment 1. Figure 4 shows graphically, by means of box-plots, the median, the first quartile, the third quartile, the whiskers and the extreme values of the relative Loo differences.

Table 4 reports the percentages of datasets in which the quasi-local kernels achieve substantially better results, in terms of Loo accuracy relative differences, with respect to the corresponding input kernel, formally $\nu_{\delta>t}$, and in which they perform substantially worse, formally $\nu_{\delta<\text{-}t}$. For example, for $t = 5$ and for the kernel resulting from $\mathcal{S}_{\sigma,\eta}$ with $\sigma = \sigma_{.1}$ and $\eta = \eta_{.5}$, we have that the accuracy is never worse than 5% of the input kernels (except for one dataset in the linear kernel case) while the percentages of datasets in which we have a gain in accuracy of at least 5% are 40% for the linear kernel, 55% for the polynomial kernel, 10% for the RBF kernel and 30% for the sigmoidal kernel.

The input-space estimated values of the dataset-dependent parameter used in this experiment ($\sigma_{.1} = \eta_{.1}$, $\eta_{.5}$, $\eta_{.9}$, $\eta_{.1r}$, $\eta_{.5r}$, $\eta_{.9r}$) are reported in the Additional Material for every dataset of Experiment 1; here we notice that the ratios between $\eta_{.5}$ and $\eta_{.1r}$ are about of one order of magnitude. In particular the median of the ratios is 6.5, the first quartile is 4.5, the third quartile is 9.4; in only one case (*svmguide2* dataset) is $\eta_{.5}$ lower than $\eta_{.1r}$ and

		Statistics on $\Delta_{A_{Loo}}$				Statistics on $\delta_{A_{Loo}}$							
	k	k^{lin}	k^{pol}	k^{rbf}	k^{sig}	k^{lin}		k^{pol}		k^{rbf}		k^{sig}	
	η	mean±sd	mean±sd	mean±sd	mean±sd	mean±sd	skew	mean±sd	skew	mean±sd	skew	mean±sd	skew
$\mathcal{E}_\sigma\, k$		4.2±9.7	2.8±6.6	-7.0±10.0	2.4±12.6	6.3±14.4	2.0	4.9±11.8	1.3	-8.4±12.0	-1.4	6.0±20.4	6.0
$\mathcal{P}_\sigma\, k$		3.1±8.5	0.1±1.6	-4.0±8.2	3.0±6.6	4.3±10.8	1.0	0.1±2.4	-0.9	-5.0±10.0	-2.1	5.2±12.2	5.2
$\mathcal{S}_\sigma\, k$		5.1±8.7	5.3±6.3	0.1±0.8	6.0±9.4	7.4±13.5	2.3	9.8±12.9	1.4	0.1±1.0	0.2	10.5±17.8	10.5
$\mathcal{S}_{\sigma,\eta}\, k$	$\eta_{.1}$	3.7±7.1	5.7±7.5	0.9±2.1	3.4±7.3	4.9±9.2	1.5	10.6±15.6	2.0	1.0±2.4	2.3	5.8±12.6	5.8
$\mathcal{S}_{\sigma,\eta}\, k$	$\eta_{.5}$	4.1±7.8	10.8±14.1	1.1±2.2	5.6±10.4	5.4±10.1	1.2	21.3±33.2	2.5	1.2±2.6	1.8	9.9±19.5	9.9
$\mathcal{S}_{\sigma,\eta}\, k$	$\eta_{.9}$	4.2±8.1	12.6±14.7	0.6±2.7	5.6±11.6	5.7±10.7	1.1	24.1±34.2	2.2	0.6±3.3	1.1	10.0±21.0	10.0
$\mathcal{S}_{\sigma,\eta}\, k$	$\eta_{.1r}$	3.9±6.7	3.9±4.5	0.3±0.6	3.8±7.3	5.2±8.7	1.7	6.9±8.5	1.1	0.3±0.7	1.1	6.4±13.2	6.4
$\mathcal{S}_{\sigma,\eta}\, k$	$\eta_{.5r}$	4.3±7.1	6.5±7.9	0.4±0.9	4.9±8.9	5.9±9.4	1.4	12.5±17.8	2.2	0.4±1.1	1.1	8.6±16.7	8.6
$\mathcal{S}_{\sigma,\eta}\, k$	$\eta_{.9r}$	4.8±7.6	8.5±11.8	0.3±1.1	5.5±9.6	6.7±10.5	1.2	17.0±28.2	2.8	0.3±1.3	0.6	9.7±18.2	9.7
$\mathcal{S}_{\sigma,\eta}\, k$	$\eta^{\mathcal{F}}_{.1}$	3.7±7.1	0.3±0.4	0.2±0.9	0.9±2.4	4.9±9.2	1.5	0.4±0.6	2.0	0.2±1.0	0.3	1.7±4.1	1.7
$\mathcal{S}_{\sigma,\eta}\, k$	$\eta^{\mathcal{F}}_{.5}$	4.1±7.8	0.9±1.3	0.1±1.2	2.7±6.6	5.4±10.1	1.2	1.4±2.3	1.9	0.1±1.5	0.3	4.5±10.6	4.5
$\mathcal{S}_{\sigma,\eta}\, k$	$\eta^{\mathcal{F}}_{.9}$	4.2±8.1	2.2±3.5	0.1±1.2	3.8±8.0	5.7±10.7	1.1	3.4±5.4	1.6	0.1±1.5	0.2	6.6±14.7	6.6
$\mathcal{PS}_\sigma\, k$		4.3±7.7	4.0±5.4	0.7±1.9	4.5±7.8	5.8±10.2	1.2	7.1±9.5	1.7	0.8±2.3	2.9	7.8±14.8	7.8
$\mathcal{PS}_{\sigma,\eta}\, k$	$\eta_{.1}$	3.0±7.8	5.1±8.6	0.8±2.3	2.4±5.3	4.2±9.9	0.8	9.6±15.8	1.9	0.9±2.7	1.1	4.1±9.7	4.1
$\mathcal{PS}_{\sigma,\eta}\, k$	$\eta_{.5}$	3.2±8.9	8.8±14.5	0.7±2.7	4.6±9.4	4.4±11.5	1.0	18.0±33.9	2.8	0.8±3.3	0.1	8.0±17.4	8.0
$\mathcal{PS}_{\sigma,\eta}\, k$	$\eta_{.9}$	3.0±9.0	10.9±14.6	0.2±2.9	6.2±11.2	4.2±11.5	1.0	21.5±33.8	2.4	0.1±3.6	0.5	10.8±20.8	10.8
$\mathcal{PS}_{\sigma,\eta}\, k$	$\eta_{.1r}$	3.1±8.0	4.3±7.4	0.7±2.0	2.6±4.8	4.4±10.2	0.7	7.6±13.3	2.6	0.8±2.4	2.6	4.5±9.1	4.5
$\mathcal{PS}_{\sigma,\eta}\, k$	$\eta_{.5r}$	3.5±8.6	5.3±8.1	0.7±2.1	3.9±7.6	4.9±11.1	0.9	9.8±15.3	1.9	0.8±2.5	2.2	6.7±14.3	6.7
$\mathcal{PS}_{\sigma,\eta}\, k$	$\eta_{.9r}$	3.4±8.7	6.1±8.5	0.8±2.1	4.8±8.9	4.8±11.3	1.0	11.5±16.8	1.7	0.9±2.6	2.0	8.4±16.6	8.4
$\mathcal{PS}_{\sigma,\eta}\, k$	$\eta^{\mathcal{F}}_{.1}$	3.0±7.8	0.5±0.8	0.6±2.1	0.6±1.3	4.2±9.9	0.8	0.7±1.3	2.1	0.7±2.5	2.5	1.0±2.3	1.0
$\mathcal{PS}_{\sigma,\eta}\, k$	$\eta^{\mathcal{F}}_{.5}$	3.2±8.9	0.9±1.5	0.7±2.3	2.1±4.9	4.4±11.5	1.0	1.5±2.6	2.7	0.7±2.8	1.6	3.4±7.7	3.4
$\mathcal{PS}_{\sigma,\eta}\, k$	$\eta^{\mathcal{F}}_{.9}$	3.0±9.0	1.7±2.5	0.6±2.4	3.3±7.1	4.2±11.5	1.0	2.9±4.7	2.9	0.7±2.8	1.6	5.7±12.6	5.7

Table 3: Experiment 1. Mean and standard deviation of the absolute differences, and mean, standard deviation and skewness of the relative differences between Loo accuracies of quasi-local kernels and of corresponding input kernels, for all the 20 datasets.

in only one case (*leukemia* dataset) is the ratio greater then 100.

Table 5 shows the Loo accuracy of the KNNSVM classifier and of the KNNSVM classifier with fixed a-priori K values for the *segment*, *sonar*, *vehicle* and *splice* datasets (the results for all the datasets are available in the Additional Material). Obviously for values of K in KNNSVM greater than the dataset cardinality the accuracy results are missing. Similarly to the classifier accuracy statistics reported in Table 3 and Table 4, the KNNSVM classifier and the K^*NNSVM values are also summarized in Table 6 using the relative Loo accuracy values.

Finally, the scatter plot in Figure 5 compares the performance of the $\mathcal{S}_{\sigma,\eta}\, k$ with $\eta_{.5}$ and $\sigma_{.1}$ with the KNNSVM classifier and with K^*NNSVM for every dataset.

5.3 Discussion

Observing the box-plots diagrams of Figure 4 regarding the relative variation in the Loo accuracy produced by the quasi-local kernels in the SVM classification, we can see that most of the quasi-local kernels are able to significantly improve the classification accuracy of the reference input kernels. The same conclusion can be deduced from Table 4 in which, apart from $\mathcal{E}_\sigma\, k$ and $\mathcal{P}_\sigma\, k$ with $k = k^{rbf}$, the quasi local kernels exhibit always a higher (or at lest equal) percentage of datasets showing accuracy gains than percentage of datasets

k \ η		k^{lin}				k^{pol}				k^{rbf}				k^{sig}			
		$\nu_{\delta>1}$	$\nu_{\delta<\text{-}1}$	$\nu_{\delta>5}$	$\nu_{\delta<\text{-}5}$	$\nu_{\delta>1}$	$\nu_{\delta<\text{-}1}$	$\nu_{\delta>5}$	$\nu_{\delta<\text{-}5}$	$\nu_{\delta>1}$	$\nu_{\delta<\text{-}1}$	$\nu_{\delta>5}$	$\nu_{\delta<\text{-}5}$	$\nu_{\delta>1}$	$\nu_{\delta<\text{-}1}$	$\nu_{\delta>5}$	$\nu_{\delta<\text{-}5}$
$\mathcal{E}_\sigma\, k$		50	35	40	5	45	15	35	10	10	55	0	45	40	25	30	25
$\mathcal{P}_\sigma\, k$		50	40	40	10	15	10	5	5	5	30	0	25	35	5	25	0
$\mathcal{S}_\sigma\, k$		50	10	35	0	50	0	45	0	15	10	0	0	50	0	30	0
$\mathcal{S}_{\sigma,\eta}\, k$	$\eta_{.1}$	45	20	35	5	60	0	45	0	30	10	5	0	45	25	20	5
$\mathcal{S}_{\sigma,\eta}\, k$	$\eta_{.5}$	50	25	40	5	65	10	55	0	40	10	10	0	55	15	30	0
$\mathcal{S}_{\sigma,\eta}\, k$	$\eta_{.9}$	50	30	40	5	65	10	55	0	30	30	10	0	45	30	35	10
$\mathcal{S}_{\sigma,\eta}\, k$	$\eta_{.1r}$	50	20	35	0	55	0	40	0	15	0	0	0	50	5	20	0
$\mathcal{S}_{\sigma,\eta}\, k$	$\eta_{.5r}$	50	25	40	0	60	0	50	0	20	0	0	0	45	5	25	0
$\mathcal{S}_{\sigma,\eta}\, k$	$\eta_{.9r}$	50	25	40	0	60	0	55	0	25	10	0	0	50	5	30	0
$\mathcal{S}_{\sigma,\eta}\, k$	$\eta^{\mathcal{F}}_{.1}$	45	20	35	5	10	0	0	0	15	10	0	0	20	5	15	0
$\mathcal{S}_{\sigma,\eta}\, k$	$\eta^{\mathcal{F}}_{.5}$	50	25	40	5	35	0	10	0	20	20	0	0	35	5	15	0
$\mathcal{S}_{\sigma,\eta}\, k$	$\eta^{\mathcal{F}}_{.9}$	50	30	40	5	45	0	25	0	25	20	0	0	40	0	20	0
$\mathcal{PS}_\sigma\, k$		55	25	40	5	60	0	35	0	30	10	5	0	45	0	30	0
$\mathcal{PS}_{\sigma,\eta}\, k$	$\eta_{.1}$	50	35	40	10	50	5	35	0	25	5	5	5	45	10	20	5
$\mathcal{PS}_{\sigma,\eta}\, k$	$\eta_{.5}$	50	35	40	10	55	10	45	0	40	20	10	5	50	15	25	5
$\mathcal{PS}_{\sigma,\eta}\, k$	$\eta_{.9}$	50	35	40	20	60	10	55	0	30	35	10	5	55	20	40	5
$\mathcal{PS}_{\sigma,\eta}\, k$	$\eta_{.1r}$	45	30	40	10	60	0	35	0	25	10	5	0	45	10	20	0
$\mathcal{PS}_{\sigma,\eta}\, k$	$\eta_{.5r}$	50	30	40	10	60	0	35	0	35	15	5	0	50	10	20	0
$\mathcal{PS}_{\sigma,\eta}\, k$	$\eta_{.9r}$	50	35	40	10	60	5	45	0	35	15	5	0	55	10	25	5
$\mathcal{PS}_{\sigma,\eta}\, k$	$\eta^{\mathcal{F}}_{.1}$	50	35	40	10	25	0	0	0	35	20	5	0	25	0	15	0
$\mathcal{PS}_{\sigma,\eta}\, k$	$\eta^{\mathcal{F}}_{.5}$	50	35	40	10	35	0	5	0	35	25	5	0	35	5	20	0
$\mathcal{PS}_{\sigma,\eta}\, k$	$\eta^{\mathcal{F}}_{.9}$	50	35	40	20	55	0	20	0	35	25	5	0	35	0	20	0

Table 4: Experiment 1. Table of the percentages of datasets $\nu_{\delta>t}$ (and $\nu_{\delta<-t}$) in which the quasi-local kernels achieve sensibly better (and worse) Loo accuracy values with respect to to the corresponding input kernels. Thresholds of 1% and 5% are considered.

showing accuracy losses. More precisely, analysing also the mean relative Loo variations in Table 3, the quasi-local kernels that are shown to be more accurate are $\mathcal{S}_\sigma$, $\mathcal{S}_{\sigma,\eta}$ with $\eta \in \{\eta_{.1},\, \eta_{.5},\, \eta_{.9},\, \eta_{.1r},\, \eta_{.5r},\, \eta_{.9r}\}$, $\mathcal{PS}_\sigma$ and $\mathcal{PS}_{\sigma,\eta}$ with $\eta \in \{\eta_{.1},\, \eta_{.5},\, \eta_{.9},\, \eta_{.1r},\, \eta_{.5r},\, \eta_{.9r}\}$ all with $\sigma = \sigma_{.1}$. All their Loo variations are in fact positive, and observing that also the skewness is always positive, the standard deviation is rather high because of positive outliers. On the other hand, the kernels resulting from $\mathcal{E}_\sigma$ shows important limitations, as reported in Table 4; for example only 45% of the datasets do not give accuracy values lower than 1% of the accuracy of the RBF kernel, meaning that in 55% of cases the results are lower than 1%. The only case in which $\mathcal{E}_\sigma\, k$ seem to perform quite well is with the linear kernel as input kernel, but this is not surprising since $\mathcal{E}_\sigma\, k^{lin} = k^{rbf}$. Kernels obtained from $\mathcal{P}_\sigma$ are even worse than the one produced by $\mathcal{E}_\sigma$. The low Loo accuracy results of $\mathcal{E}_\sigma$ and $\mathcal{P}_\sigma$ are also highlighted by the scatter plots of Figure 2, in which it is clear that there is not a predominance of points over the bisector line.

For $\mathcal{S}_{\sigma,\eta}$ the best results in terms of Loo accuracy improvements are achieved for $\eta_{.1}$, $\eta_{.5}$ and $\eta_{.9}$ and in general $\mathcal{S}_{\sigma,\eta}\, k$ gives Loo accuracy values higher than $\mathcal{PS}_{\sigma,\eta}\, k$ as we can see especially in Table 3. In contrast, the quasi-local kernels that seem to guarantee less risk of achieving worse results are those obtained from $\mathcal{S}_{\sigma,\eta}$ with $\eta_{.1r}$, $\eta_{.5r}$ and $\eta_{.9r}$ (see the percentages of relative Loo accuracy losses in Table 4). So we can reasonably conclude that the quasi-local kernel that seems more promising are produced by the operators $\mathcal{S}_{\sigma,\eta}$

	segment				*sonar*				*vehicle*				*splice*			
k	k^{lin}	k^{pol}	k^{rbf}	k^{sig}	k^{lin}	k^{pol}	k^{rbf}	k^{sig}	k^{lin}	k^{pol}	k^{rbf}	k^{sig}	k^{lin}	k^{pol}	k^{rbf}	k^{sig}
KNNSVM	.97	.97	.97	.97	.88	.80	.85	.84	.70	.69	.70	.71	.77	.63	.86	.85
1-NNSVM	**.97**	**.97**	**.97**	**.97**	.88	**.80**	.88	**.87**	.70	.69	.70	.70	.70	.61	.70	.71
3-NNSVM	.97	.96	.96	.97	.86	.79	.84	.85	.71	.71	.70	.71	.70	.63	.70	.70
5-NNSVM	.96	.96	.96	.96	.88	.78	.82	.81	.72	**.71**	.70	**.72**	.72	.57	.70	.70
7-NNSVM	.95	.95	.96	.95	.89	.77	.80	.80	.72	.70	.71	.72	.72	.48	.71	.71
9-NNSVM	.95	.95	.96	.95	.89	.72	.76	.77	.72	.71	.70	.71	.75	.48	.71	.71
11-NNSVM	.95	.95	.95	.95	**.91**	.68	.81	.72	.72	.69	.72	.70	.75	.48	.71	.70
15-NNSVM	.95	.95	.96	.95	.88	.67	.83	.73	.72	.69	.72	.69	.77	**.64**	.73	.72
23-NNSVM	.95	.94	.97	.94	.87	.67	.86	.72	.74	.66	.73	.68	.78	.57	.75	.73
39-NNSVM	.95	.92	.97	.93	.89	.67	.88	.75	.74	.64	.74	.67	.79	.48	.77	.78
71-NNSVM	.95	.89	.97	.90	.88	.64	**.89**	.75	.76	.59	.75	.63	**.80**	.49	.83	.83
135-NNSVM	.95	.86	.97	.88	.80	.59	.88	.79	.78	.55	.75	.61	.78	.48	.85	**.85**
263-NNSVM	.95	.86	.97	.89	-	-	-	-	**.80**	.49	**.75**	.67	.79	.48	.86	.82
519-NNSVM	.94	.88	.96	.89	-	-	-	-	.79	.43	.75	.69	.77	.48	- **.86**	.80
1031-NNSVM	.94	.90	.96	.90	-	-	-	-	-	-	-	-	-	-	-	-

Table 5: Experiment 1. Loo accuracy result of the KNNSVM classifier and of KNNSVM classifier with fixed a-priori values of K. KNNSVM results for K greater than the dataset cardinality are missing because the classifier is not applicable. The K^*NNSVM values are highlighted in bold.

k	k^{lin}				k^{pol}				k^{rbf}				k^{sig}			
	mean	sd	skew	med	mean	sd	skew	med	mean	sd	skew	med	mean	sd	skew	med
KNNSVM	4.7	12.9	1.4	-0.4	21.7	31.4	-0.9	0.0	-0.7	2.7	-0.1	-1.3	11.9	23.4	2.1	1.4
K^*NNSVM	7.7	12.3	1.6	0.9	23.8	31.9	2.4	10.5	1.4	2.4	2.2	0.5	14.2	23.8	2.0	2.9
	$\nu_{\delta>1}$	$\nu_{\delta<\text{-}1}$	$\nu_{\delta>5}$	$\nu_{\delta<\text{-}5}$	$\nu_{\delta>1}$	$\nu_{\delta<\text{-}1}$	$\nu_{\delta>5}$	$\nu_{\delta<\text{-}5}$	$\nu_{\delta>1}$	$\nu_{\delta<\text{-}1}$	$\nu_{\delta>5}$	$\nu_{\delta<\text{-}5}$	$\nu_{\delta>1}$	$\nu_{\delta<\text{-}1}$	$\nu_{\delta>5}$	$\nu_{\delta<\text{-}5}$
KNNSVM	40	40	30	15	65	10	65	5	25	55	0	5	50	25	40	5
K^*NNSVM	50	10	35	0	65	0	65	0	45	0	5	0	55	5	45	0

Table 6: Experiment 1. Mean, standard deviation, skewness and median of relative Loo accuracy differences between SVM using the input kernels and the KNNSVM classifier and with K^*NNSVM on all the 20 datasets.

with $\eta_{.5}$ and $\sigma_{.1}$ and $\mathcal{S}_{\sigma,\eta}$ with $\eta_{.1r}$ and $\sigma_{.1}$; in particular the first is the one that appear to potentially achieve better results, while the second is the one with less possibility to achieve worse results. In addition, the performances of the $\mathcal{PS}_{\sigma,\eta}$ derived class of kernels are very close to the $\mathcal{S}_{\sigma,\eta}$ ones, although they are a bit lower. The fact that $\eta_{.5}$ is statistically almost 10 times greater than $\eta_{.1r}$ confirms the observation that $\mathcal{S}_{\sigma,\eta}\, k$ and $\mathcal{PS}_{\sigma,\eta}\, k$ with $\eta = \eta_{.1r}$ are more conservative with respect to the k input kernel behaviour than the same quasi-local kernels with $\eta = \eta_{.5}$ and make reasonable the use of both parameter settings.

The resulting statistics (see for example the box-plots in Figure 4) show that all the input kernels benefit from the quasi-local transformation by the $\mathcal{S}_\sigma$, $\mathcal{S}_{\sigma,\eta}$, $\mathcal{PS}_\sigma$ and $\mathcal{PS}_{\sigma,\eta}$ operators. However, the quasi-local kernels with lower accuracy improvements are those applied on the RBF input kernel (they rarely improve the k^{rbf} accuracy by more than 5% as reported in Table 4). This is reasonable since the RBF kernel is already a local kernel and thus should not take advantage of the operator transformation, so even the small improvements observed are very positive. The explanation for the accuracy improvements of $\mathcal{O}\,k^{rbf}$ can regard the γ^{rbf} parameter for k^{rbf}; its estimation with the 0.1 quantile of the distribution of all pairwise distances probably gives values that permits the influence of points not very close to each other. It is reasonable to argue that, in these conditions, the quasi-local operators on the RBF kernel maintain the influence of non very close points due to rather large value of γ^{rbf}, enhancing locality only in some regions.

The scatter plot in Figure 5(a), showing the Loo accuracy of KNNSVM classifier and SVM with $\mathcal{S}_{\sigma,\eta}\,k$ kernel with $\eta_{.5}$ and $\sigma_{.1}$, highlights that the quasi-local kernels perform better in the majority of cases (61% against 35% of cases in which the KNNSVM performs better). The comparison with K^{*}NNSVM in Figure 5(b), on the other hand, gives a percentage of 34% of cases in which the quasi-local kernels perform better. The two comparisons lead us to conclude that, for the datasets of Experiment 1, the quasi-local kernels (and in particular $\mathcal{S}_{\sigma,\eta}\,k$ with $\eta_{.5}$ and $\sigma_{.1}$) perform significantly better than the KNNSVM classifier even though they do not perform as well as K^{*}NNSVM which is however not a proper classifier. The Loo accuracy differences between the family of SVM classifiers with quasi-local kernels and the family of KNNSVM classifiers can be due to the two theoretical differences between them: the partial preservation of the global kernel behaviour of the first and the hard-boundary between local and non local points of the second (see section 3).

6 Experiment 2

The second experiment applies the SVM with the quasi-local kernels that, in the exploratory Experiment 1, seem to achieve better accuracy values. We recall that we found that the most promising quasi-local kernels are those obtained by $\mathcal{S}_{\sigma,\eta}$ with $\eta = \eta_{.5}$ or $\eta = \eta_{.1r}$, and by $\mathcal{PS}_{\sigma,\eta}$ with $\eta = \eta_{.5}$ or $\eta = \eta_{.1r}$ all with $\sigma = \sigma_{.1}$. The aim of this experiment is to verify if these kernels are able to improve the input kernel classification accuracy in a considerable number of quite large datasets.

6.1 Experimental procedure

The 13 datasets used in the second experiment are summarized in Table 7 and are available at the homepage of LibSVM [7]. The datasets are quite large and for this reason kernels resulting from the four chosen operators with the four input kernels are simply trained on the training set and tested on the testing set as provided online on the LibSVM website. The evaluation of the classifiers is performed with the same statistical tools described for Experiment 1 using the testing set accuracy A^{T} instead of the Loo accuracy (A^{Loo}). The corresponding absolute and relative testing differences are $\Delta_{A^T}(\mathcal{O}\,k, k, D)$ and $\delta_{A^T}(\mathcal{O}\,k, k, D)$.

We do not test the KNNSVM classifier on these datasets because of the computational weight of the method. For example, on the *a9a* dataset, the SVM methods require several minutes of computation on a Pentium D 3.40 GHz desktop system, while K^{*}NNSVM with $K \in \mathcal{K}$ requires about 4 hours. Moreover the $\mathcal{K}$ set must be enlarged for this experiment since the maximum value in $\mathcal{K}$ is too small in comparison with the training set cardinalities thus further increasing the computational effort. For example, KNNSVM with a fixed value of K equal to 1/10 of the training set cardinality requires about 10 hours on the *a9a* dataset. Remembering that KNNSVM with automatic choice of K is done with the 10-fold cross validation, it requires the training of $|\mathcal{K}| \times (|train.\ set| - 1)$ SVM for every test point. This means that we can estimate the computational time for testing KNNSVM on the *a9a* dataset to be in the order of months.

Dataset name	source	classes	training size	testing size	features
dna	Statlog [18]	3	2000	1186	180
a9a	UCI [1]	2	32561	16281	123
shuttle	Statlog [18]	7	43500	14500	9
w8a	JP98a [23]	2	49749	14951	300
letter	Statlog [18]	26	15000	5000	16
satimage	Statlog [18]	6	4435	2000	36
news20	KL95a [19]	20	15935	3993	62061
ijcnn1	DP01a [24]	2	49990	91701	22
usps	JJH94a [17]	10	7291	2007	256
mnist1	YL98a [20]	10	21000	49000	780
rcv1.binary	DL04b [21]	2	20242	677399	47236
acoustic	Sensit [11]	3	78823	19705	50
seismic	Sensit [11]	3	78823	19705	50

Table 7: Datasets for Experiment 2.

6.2 Results

Table 8 shows the accuracy results of the input kernels k and of the four quasi-local kernels considered in this experiment on all the 13 datasets listed in Table 7. Formally we report $A^T(\mathcal{O}\, k, D)$ with $\mathcal{O} \in \{\mathcal{I},\ \mathcal{S}_{\sigma,\eta},\ \mathcal{S}_{\sigma,\eta},\ \mathcal{S}_{\sigma,\eta},\ \mathcal{S}_{\sigma,\eta}\}$, $\sigma = \sigma_{.1}$, $\eta \in \{\eta_{.1r}, \eta_{.5}\}$, $k \in \{k^{lin},\ k^{pol},\ k^{rbf},\ k^{sig}\}$, $D \in \{$*dna*, *a9a*, *shuttle*, *w8a*, *letter*, *satimage*, *news20*, *ijcnn1*, *usps*, *mnist1*, *rcv1.binary*, *acoustic*, *seismic*$\}$. The 12 cases in which a quasi-local kernel exhibits worse accuracy values with respect to the corresponding input kernels ($\nu_{\delta<0}$) are underlined. They corresponds to the 5.77% of the total comparisons and 8 of them regard the *a9a* dataset. On the other hand, the percentage of times in which the quasi-local kernels achieve, in total, better results with respect to the input kernels (i.e. $\nu_{\delta>0}$) is 79.81%. In 14.42% of cases the accuracy remains unchanged.

Figure 7 shows by means of scatter plots the comparison of the accuracy of the four quasi-local kernels considered in this experiment with the accuracy of the corresponding input kernels. The points representing accuracy values lower than 0.7 are not shown in the scatter plots, but are reported in the complete results table (Table 8) and are all cases in which the quasi-local kernels perform better or at least equally to the input kernels. Similarly to Experiment 1, in Table 9 the description of the distribution of the difference in test classification accuracy between the quasi-local kernels and the corresponding input kernels is performed with the mean and the standard deviation of the distribution of the absolute differences and the mean, the standard deviation and the skewness of the distribution of the relative differences. The description of the relative differences distribution is also performed with the box-plots in Figure 6 representing graphically the median, the first and third quartile, the whispers and the extreme values. The mean and the median percentage

	η k	*dna* k^{lin}	k^{pol}	k^{rbf}	k^{sig}	*a9a* k^{lin}	k^{pol}	k^{rbf}	k^{sig}	*shuttle* k^{lin}	k^{pol}	k^{rbf}	k^{sig}	*w8a* k^{lin}	k^{pol}	k^{rbf}	k^{sig}
k		.931	.508	.951	.938	.850	.764	.850	.849	.972	.823	.998	.960	.987	.970	.991	.972
$\mathcal{S}_{\sigma,\eta}\, k$	$\eta_{.5}$	.954	.508	.954	.947	$\underline{.832}$	.821	$\underline{.846}$	.850	.997	.922	.998	.989	.995	.970	.993	.982
$\mathcal{S}_{\sigma,\eta}\, k$	$\eta_{.1r}$	.954	.508	.955	.949	$\underline{.840}$	.814	$\underline{.849}$	.850	.986	.894	.998	.981	.994	.970	.993	.977
$\mathcal{PS}_{\sigma,\eta}\, k$	$\eta_{.5}$	.954	.931	.955	.938	$\underline{.807}$	.834	$\underline{.826}$	.850	.993	.833	.998	.981	.995	.970	.995	.985
$\mathcal{PS}_{\sigma,\eta}\, k$	$\eta_{.1r}$	.954	.920	.955	.941	$\underline{.807}$	.828	$\underline{.833}$	.850	.984	.825	.998	.977	.995	.970	.994	.982

	η k	*letter* k^{lin}	k^{pol}	k^{rbf}	k^{sig}	*satimage* k^{lin}	k^{pol}	k^{rbf}	k^{sig}	*news20* k^{lin}	k^{pol}	k^{rbf}	k^{sig}	*ijcnn1* k^{lin}	k^{pol}	k^{rbf}	k^{sig}
k		.843	.376	.955	.772	.858	.666	.910	.833	.840	.050	.836	.050	.916	.905	.976	.906
$\mathcal{S}_{\sigma,\eta}\, k$	$\eta_{.5}$	.967	.656	.977	.842	.913	.801	.917	.870	.841	.050	.842	.050	.971	.905	.980	.949
$\mathcal{S}_{\sigma,\eta}\, k$	$\eta_{.1r}$	.957	.554	.973	.828	.912	.721	.918	.852	.841	.050	.842	.050	.963	.905	.981	.926
$\mathcal{PS}_{\sigma,\eta}\, k$	$\eta_{.5}$	.975	.599	.979	.846	.909	.793	.922	.871	.844	.050	.840	.050	.969	.905	.979	.930
$\mathcal{PS}_{\sigma,\eta}\, k$	$\eta_{.1r}$	.970	.504	.975	.824	.910	.712	.920	.851	.843	.050	.840	.050	.966	.905	.981	.922

	η k	*usps* k^{lin}	k^{pol}	k^{rbf}	k^{sig}	*mnist1* k^{lin}	k^{pol}	k^{rbf}	k^{sig}	*rcv1.binary* k^{lin}	k^{pol}	k^{rbf}	k^{sig}	*acoustic* k^{lin}	k^{pol}	k^{rbf}	k^{sig}
k		.930	.938	.953	.917	.922	.264	.971	.915	.963	.525	.965	.525	.702	.704	.790	.681
$\mathcal{S}_{\sigma,\eta}\, k$	$\eta_{.5}$	.953	.949	.957	$\underline{.915}$	.972	.653	.975	.928	.964	.525	.966	.525	.795	.756	.799	.690
$\mathcal{S}_{\sigma,\eta}\, k$	$\eta_{.1r}$	.952	.947	.957	.922	.970	.579	.975	.926	.964	.525	.966	.525	.789	.743	.798	.688
$\mathcal{PS}_{\sigma,\eta}\, k$	$\eta_{.5}$	.953	.954	.957	$\underline{.878}$	.976	.930	.976	.934	.965	.525	.966	.525	.785	.759	.798	.693
$\mathcal{PS}_{\sigma,\eta}\, k$	$\eta_{.1r}$	.953	.954	.957	$\underline{.882}$	.976	.918	.976	.935	.965	.525	.966	.525	.795	.746	.799	.690

	η k	*seismic* k^{lin}	k^{pol}	k^{rbf}	k^{sig}
k		.727	.725	.760	.691
$\mathcal{S}_{\sigma,\eta}\, k$	$\eta_{.5}$	.766	.745	.761	.693
$\mathcal{S}_{\sigma,\eta}\, k$	$\eta_{.1r}$	.758	.735	.764	.704
$\mathcal{PS}_{\sigma,\eta}\, k$	$\eta_{.5}$	.748	.746	$\underline{.758}$	.700
$\mathcal{PS}_{\sigma,\eta}\, k$	$\eta_{.1r}$	.766	.735	.764	.706

Table 8: Experiment 2. The test accuracy results of the quasi-local kernels. The cases in which the quasi-local kernels achieve less accurate results with respect to the input kernels are underlined.

variation in accuracy are always positive (Table 9 and Figure 6) and, in total, are 7.76% and 1.07% respectively. Finally, the percentages of datasets that achieve significantly better or worse accuracy results with the quasi-local kernels with respect to the input kernels are shown in Table 10, using 1% and 5% as thresholds.

6.3 Discussion

The two quasi-local kernels with two different parameter choices considered in the experiment are shown to improve the corresponding reference input kernel results in terms of accuracy gain in the great majority of cases (80% improvements and less than 6% accuracy losses). Looking at Table 10 the accuracy improvements of at least 1% are present in the majority of cases, while the number of improvements of at least 5% remain considerable only for the linear and polynomial kernel. The losses of accuracy greater than 1% are very rare. The main statistical descriptors of the distribution of relative differences are always positive (the mean, the median) and the first quartile is also always non-negative. So Experiment 2 confirms the results of Experiment 1, concluding that the $\mathcal{S}_{\sigma,\eta}$ and $\mathcal{PS}_{\sigma,\eta}$ operators with parameter estimation based on the dataset statistics ($\sigma = \sigma_{.1}$ and $\eta = \eta_{.1r}$

	Statistics on Δ_{AT}				Statistics on δ_{AT}							
k	k^{lin}	k^{pol}	k^{rbf}	k^{sig}	k^{lin}		k^{pol}		k^{rbf}		k^{sig}	
η	mean±sd	mean±sd	mean±sd	mean±sd	mean±sd	skew	mean±sd	skew	mean±sd	skew	mean±sd	skew
$\mathcal{S}_{\sigma,\eta}\, k \quad \eta_{.5}$	3.7±3.9	8.0±12.3	0.5±0.6	1.7±2.2	4.4±5.0	1.00	21.0±42.9	2.06	0.5±0.66	1.06	2.0±2.7	1.07
$\mathcal{S}_{\sigma,\eta}\, k \quad \eta_{.1r}$	3.4±3.6	5.6±9.2	0.5±0.5	1.3±1.5	4.1±4.6	1.02	15.2±33.7	2.09	0.5±0.52	1.03	1.6±1.9	2.04
$\mathcal{PS}_{\sigma.\eta}\, k \; \eta_{.5}$	3.3±4.3	12.4±20.3	0.3±1.0	1.3±2.6	4.0±5.1	0.08	33.5±70.5	2.09	0.3±1.17	-1.2	1.6±3.1	0.09
$\mathcal{PS}_{\sigma,\eta}\, k \; \eta_{.1r}$	3.4±4.3	10.6±19.9	0.4±0.8	1.0±1.9	4.1±5.4	0.07	30.0±69.15	3.00	0.4±0.91	-1.1	1.2±2.3	0.03

Table 9: Experiment 2. Mean, standard deviation of absolute differences and mean, standard deviation and skewness of relative differences between the test accuracies of the quasi-local kernels and the corresponding input kernels.

k	k^{lin}				k^{pol}				k^{rbf}				k^{sig}			
η	$\nu_{\delta>1}$	$\nu_{\delta<\text{-}1}$	$\nu_{\delta>5}$	$\nu_{\delta<\text{-}5}$	$\nu_{\delta>1}$	$\nu_{\delta<\text{-}1}$	$\nu_{\delta>5}$	$\nu_{\delta<\text{-}5}$	$\nu_{\delta>1}$	$\nu_{\delta<\text{-}1}$	$\nu_{\delta>5}$	$\nu_{\delta<\text{-}5}$	$\nu_{\delta>1}$	$\nu_{\delta<\text{-}1}$	$\nu_{\delta>5}$	$\nu_{\delta<\text{-}5}$
$\mathcal{S}_{\sigma,\eta}\, k \quad \eta_{.5}$	69.2	7.7	46.2	0.0	61.5	0.0	46.2	0.0	15.4	0.0	0.0	0.0	53.8	0.0	7.7	0.0
$\mathcal{S}_{\sigma,\eta}\, k \quad \eta_{.1r}$	69.2	7.7	38.5	0.0	61.5	0.0	46.2	0.0	15.4	0.0	0.0	0.0	53.8	0.0	7.7	0.0
$\mathcal{PS}_{\sigma,\eta}\, k \quad \eta_{.5}$	69.2	7.7	38.5	7.7	69.2	0.0	46.2	0.0	15.4	7.7	0.0	0.0	61.5	7.7	7.7	0.0
$\mathcal{PS}_{\sigma,\eta}\, k \quad \eta_{.1r}$	69.2	7.7	46.2	7.7	61.5	0.0	46.2	0.0	23.1	7.7	0.0	0.0	61.5	7.7	7.7	0.0

Table 10: Experiment 2. Percentages of datasets in which the quasi-local kernels achieve better (and worse) results with respect to the input kernels with thresholds of 1% and 5%.

or $\eta = \eta_{.5}$) are able to improve the classification accuracies of the input kernels in the majority of the cases, while only in very few cases do the accuracies deteriorate.

As for Experiment 1, the reference input kernels whose accuracy is more improved are the polynomial and the sigmoidal ones (this is easy to see from box-plots of Figure 6). The explanation for this regards the parameter choices that are not very well tuned for k^{pol} and k^{sig}, since we do not perform model selection on the training set. The estimation of γ^{rbf} for k^{rbf} with the inverse of the 0.1 quantile of the distribution of the Euclidean distances between every pair of points is instead a good choice, so k^{rbf} is reasonably well tuned even without model selection and thus the possible gain is less marked. In fact, from Table 10, the quasi-local operators never improve the classification accuracy of k^{rbf} by more than 5%. In any case we can observe the ability of the quasi-local kernels to improve the input kernels as much as the reference kernels are badly tuned (see for example in Table 8 the *mnist1* dataset in which the accuracy of $\mathcal{PS}_{\sigma,\eta}\, k^{pol}$ with $\eta = \eta_{.5}$ is .930 against the accuracy of k^{pol} which is .264). This allows us to argue that the quasi-local kernels make the input kernel parameter selection less critical. Further investigations are however necessary to confirm this observation.

From the box-plots and the scatter plots of Experiment 2 (Figures 6 and 7), we can observe that the accuracy results of the quasi-local kernels with the two different parameter settings ($\eta = \eta_{.5}$ and $\eta = \eta_{.1r}$) are similar. So, what was hypothesized in Experiment 1 regarding the fact that $\eta = \eta_{.1r}$ is more conservative with respect to the input kernels than $\eta = \eta_{.5}$, appears here less accentuated. The dataset-dependent estimation of η is thus not so crucial for good accuracy results of the quasi-local kernels. On the other hand, the $\mathcal{S}_{\sigma,\eta}$ operator seems to perform better than $\mathcal{PS}_{\sigma,\eta}$, especially in terms of accuracy losses. In fact the 4 cases in which the quasi-local kernels achieve accuracy results lower than 3% of the corresponding input kernels are all due to $\mathcal{PS}_{\sigma,\eta}$. The same aspect is highlighted by the negative values of the skewness of the relative accuracy differences between $\mathcal{PS}\, k^{rbf}$ and k^{rbf} shown in Table 9. In conclusion the operator that has demonstrated to be more accurate is $\mathcal{S}_{\sigma,\eta}$ with $\sigma = \sigma_{.1}$ and $\eta = \eta_{.5}$.

The *news20* and *rcv1.binary* datasets are the only datasets in which very low accuracy results of k^{pol} and k^{sig} are not improved by the quasi-local kernels and in general the improvements for k^{lin} and k^{rbf} are minimal. Observing that these two datasets are the only ones that have a large number of features (62061 for *news20*, 47236 for *rcv1.binary*) this can be due to the curse of dimensionality problem (discussed in general for local learning algorithms in [3]) that affects the locality information. This means that, for a very high number of features, the kernels resulting from the $\mathcal{E}_\sigma$ operator have low classification capability. However, in the quasi-local kernels the input kernel and the local kernel term are both considered and thus, as the locality information is lost for the curse of dimensionality problem, the kernel resulting from the $\mathcal{E}_\sigma$ term becomes negligible and the maximal separating hyperplane is determined only by the k term. So we can reasonably conclude that the quasi-local kernels are affected by the curse of dimensionality problem in the sense that it is difficult to improve the input kernels for a very high number of features, but at the same time they remain robust since the input kernel k becomes predominant and thus the resulting kernel accuracy does not decrease.

This experiment also confirms the scalability of the SVM approach with the quasi-local kernels because it requires a computational effort very similar to the SVM with the corresponding input kernels even for medium and large datasets. Since we showed in the Experiment 1 that the quasi-local kernels achieve accuracy results that are at least statistically equal to the KNNSVM classifier, and since the KNNSVM has computational limitations as the training and the testing sets become larger, we can conclude that, as far as we know, SVM with quasi-local kernels is the only SVM-based method able to capture the locality information in the feature space for any input kernel.

7 Conclusions

In this paper, we have presented a novel family of operators on kernels that add locality information to the input kernel. The resulting kernels are called quasi-local kernels since they balance the global information of the original kernel (if it is a non-local kernel) with the local kernel with respect to the distance in the feature space. The intuition is that the resulting kernels are able to maintain the original kernel behaviour for regions in which the information is not local, adapting instead the separating hyperplane following the local distribution of the data.

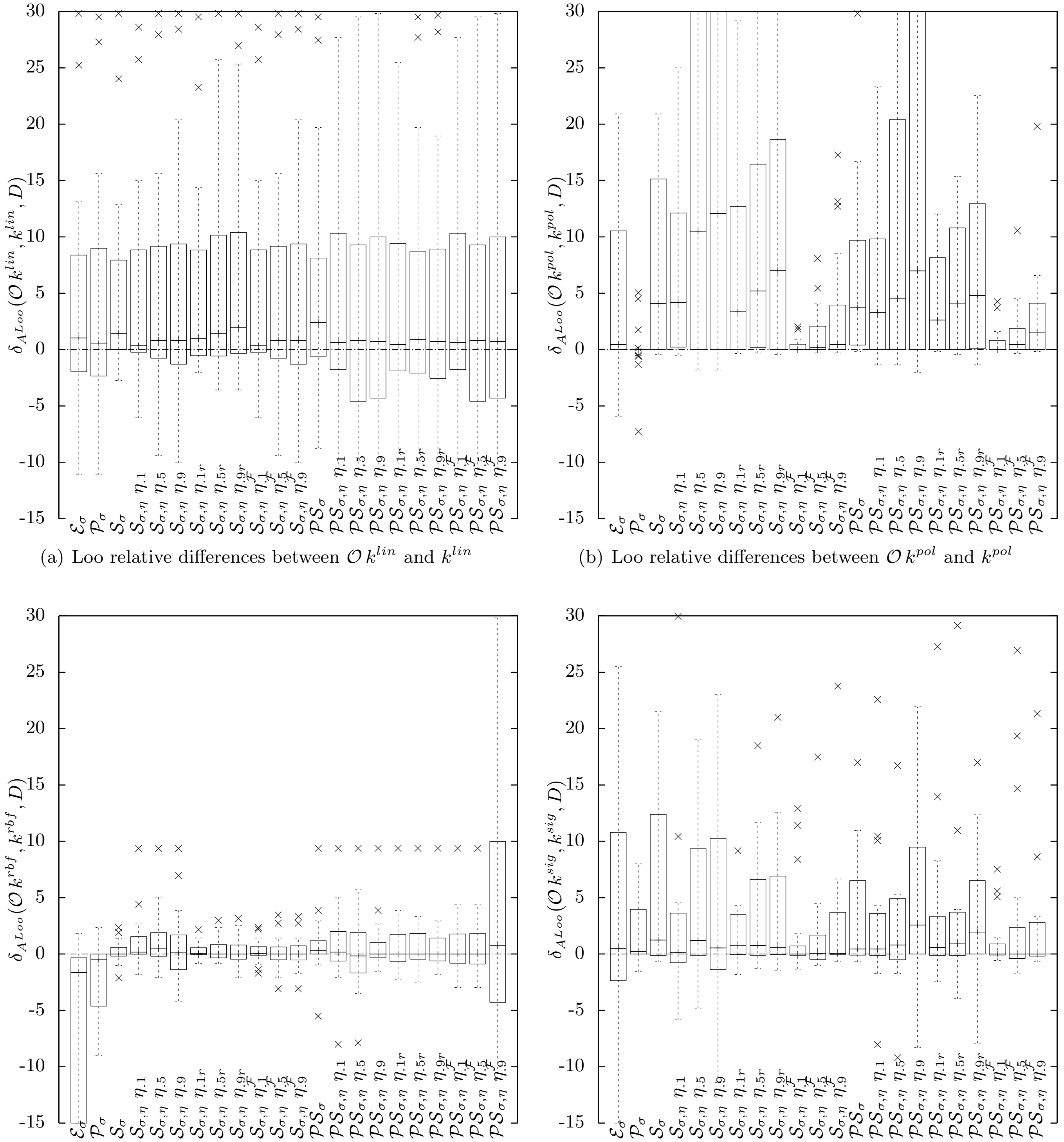
In the first experiment we tested the classification capability of all the quasi-local operators, setting the operator parameters through statistical analysis of the dataset without an expensive model selection phase, achieving very encouraging results in comparison with the classification capability of the input kernels. The second experiment confirmed that the two more promising operators ($\mathcal{S}_{\sigma,\eta}$ and $\mathcal{PS}_{\sigma,\eta}$) with two different parameter settings

($\sigma = \sigma_{.1}$ and $\eta = \eta_{.1r}$ or $\eta = \eta_{.5}$), achieved in the majority of the datasets an accuracy gain with respect to to the input kernels, with very few cases in which the accuracies are lower. In particular, although the quasi-local kernels were shown to be rather robust to the dataset-dependent estimation of their parameters, the operator that demonstrated better performances is $(\mathcal{S}_{\sigma,\eta}\, k)(x, x') = k(x, x') + \eta \cdot \exp\left(\frac{-k(x,x)-k(x',x')+2k(x,x')}{\sigma}\right)$ with $\sigma = \sigma_{.1}$ and $\eta = \eta_{.5}$. We tested the $\mathcal{S}_{\sigma,\eta}$ and $\mathcal{PS}_{\sigma,\eta}$ quasi-local kernels on a total of 33 datasets, and the results let us argue that their application on a specific dataset with a particular input kernel (not necessarily the four input kernels considered here), possibly with model selection, is extremely promising.

We compared the classification capability of the quasi-local kernel with the local SVM approach, finding that the quasi-local kernels are a little more precise than KNNSVM with the automatic tuning of the K parameter, even though they cannot reach the theoretical results of K^*NNSVM, i.e. with the a-posteriori best choice for K. Quasi-local kernels and local SVM are both based on the notion of locality in the feature space, but differ since the first always balances the local and non-local components of the kernel and the locality decreases exponentially, while the second is a compromise between locality and non-locality (depending on K) and defines a hard boundary between local and non-local points. Considering the computational performances we can conclude that, at least for large datasets, quasi-local kernels are preferable to local SVM since the theoretical approach and the classification accuracy are very similar but the computational weight of local SVM is much greater, especially if we need to classify a considerable number of new samples.

Acknowledgment

This work was supported by the FIRB - *Computational tools for Systems Biology.*



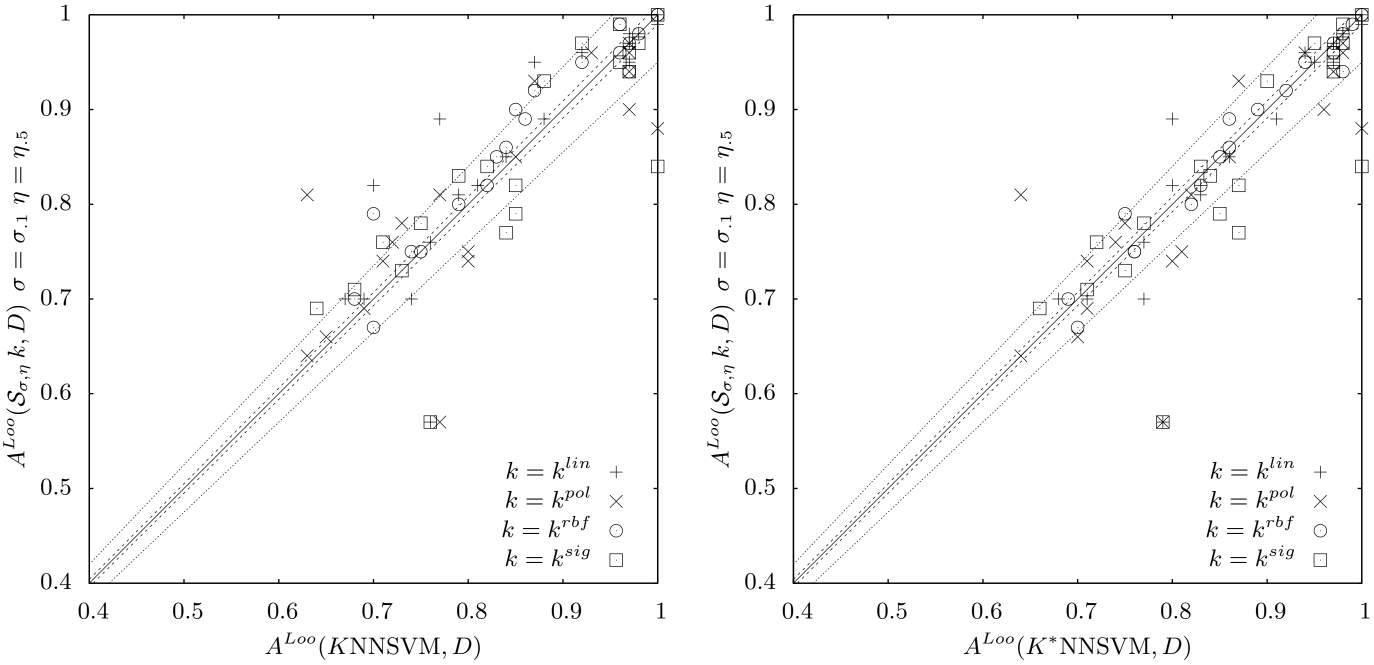
(a) Loo relative differences between $\mathcal{O}\,k^{lin}$ and k^{lin}

(b) Loo relative differences between $\mathcal{O}\,k^{pol}$ and k^{pol}

(c) Loo relative differences between $\mathcal{O}\,k^{rbf}$ and k^{rbf}

(d) Loo relative differences between $\mathcal{O}\,k^{sig}$ and k^{sig}

Figure 4: Experiment 1. Box-plots representing the distribution of the Loo relative difference $\delta_{A^{Loo}}(\mathcal{O}\,k, k, D)$ between quasi-local kernels and the corresponding input kernels on all the 20 datasets D .



(a) $\mathcal{S}_{\sigma,\eta}\,k$ with $\sigma = \sigma_{.1}$ and $\eta = \eta_{.5}$ vs. KNNSVM

(b) $\mathcal{S}_{\sigma,\eta}\,k$ with $\sigma = \sigma_{.1}$ and $\eta = \eta_{.5}$ vs. K^{*}NNSVM

Figure 5: Experiment 1. Scatter plots for Loo accuracy comparison between input kernels $k \in \{k^{lin}, k^{pol}, k^{rbf}, k^{sig}\}$ and the KNNSVM classifier and between input kernels and the K^{*}NNSVM values for all the 20 datasets.

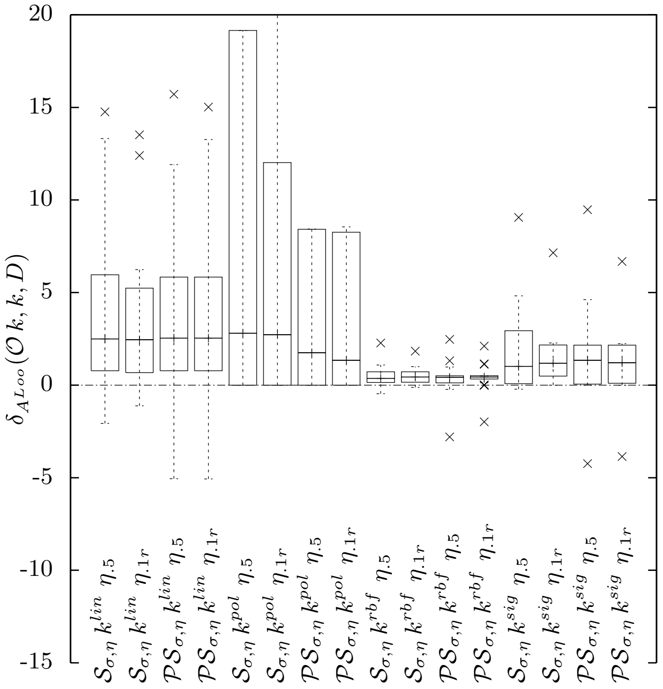
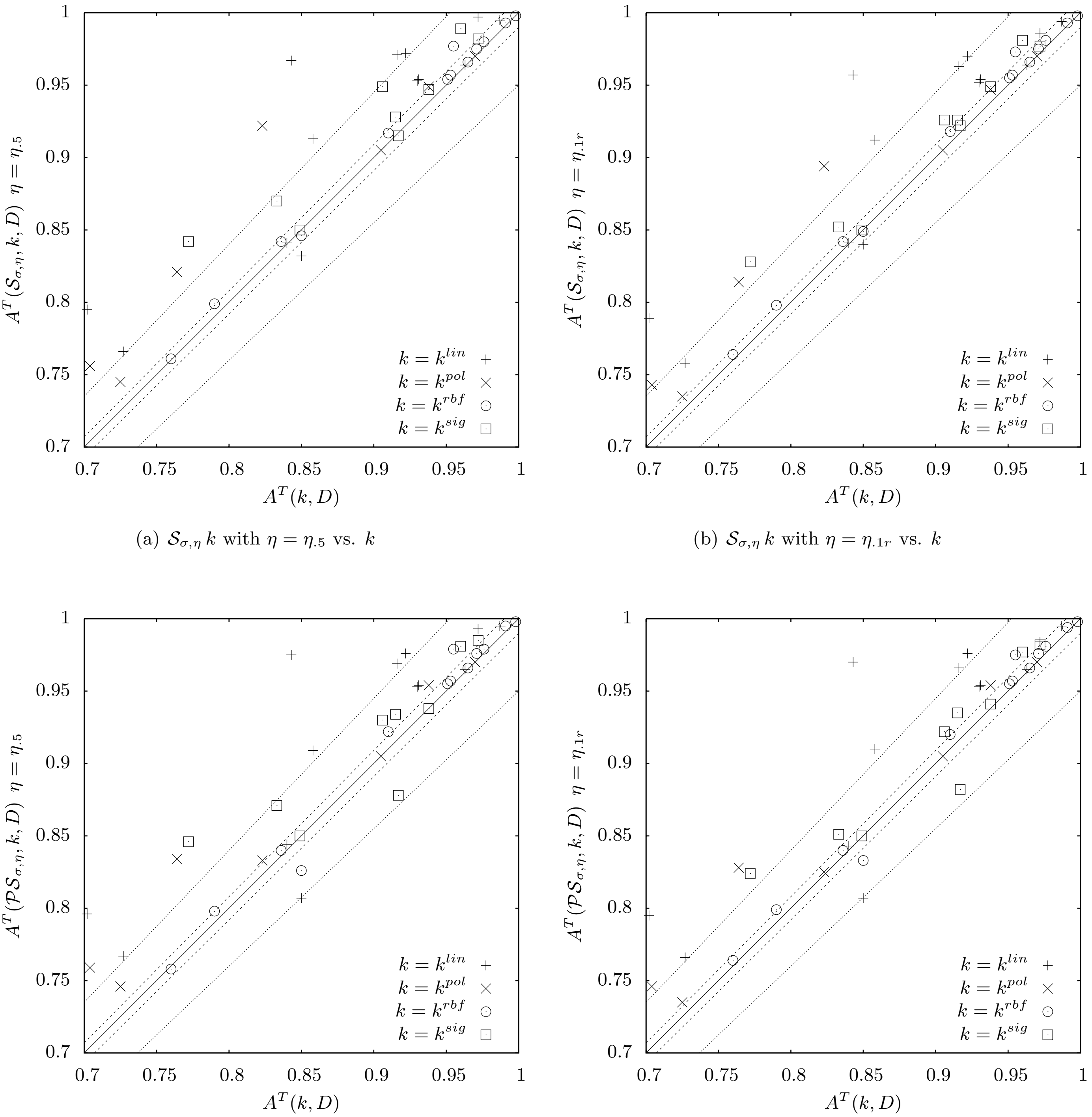


Figure 6: Experiment 2. Box-plots of the distribution of relative difference in test accuracy between the quasi-local kernels and the corresponding four input kernels (k^{lin}, k^{pol}, k^{rbf} and k^{sig}) for all the datasets.



(a) $\mathcal{S}_{\sigma,\eta}\,k$ with $\eta=\eta_{.5}$ vs. k

(b) $\mathcal{S}_{\sigma,\eta}\,k$ with $\eta=\eta_{.1r}$ vs. k

(c) $\mathcal{PS}_{\sigma,\eta}\,k$ with $\eta=\eta_{.5}$ vs. k

(d) $\mathcal{PS}_{\sigma,\eta}\,k$ with $\eta=\eta_{.1r}$ vs. k

Figure 7: Experiment 2. Scatter plots for comparison between test accuracy of $\mathcal{S}_{\sigma,\eta}\,k$ and $\mathcal{PS}_{\sigma,\eta}\,k$ quasi-local kernels and the corresponding input kernels k.

References

[1] D.J. Newman A. Asuncion. UCI machine learning repository, 2007.

[2] Y. Bengio, O. Delalleau, and N. Le Roux. The curse of dimensionality for local kernel machines. Technical Report 1258, Departement dinformatique et recherche operationnelle, Universite de Montreal, 2005.

[3] Y. Bengio, O. Delalleau, and N. Le Roux. The curse of highly variable functions for local kernel machines. *Adv Neural Inform Process Syst*, 18:107–114, 2006.

[4] E. Blanzieri and F. Melgani. An adaptive SVM nearest neighbor classifier for remotely sensed imagery. *IEEE International Conference on Geoscience and Remote Sensing Symposium (IGARSS-2006)*, pages 3931–3934, 2006.

[5] E. Blanzieri and F. Melgani. Nearest neighbor classification of remote sensing images with maximal margin principle. *IEEE Trans Geosci Rem Sens*, to appear.

[6] L. Bottou and V. Vapnik. Local learning algorithms. *Neural Comput*, 4(6):888–900, 1992.

[7] Chih-Chung Chang and Chih-Jen Lin. *LIBSVM: a library for support vector machines*, 2001. Available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

[8] C. Cortes and V. Vapnik. Support-vector networks. *Mach Learn*, 20(3):273–297, 1995.

[9] N. Cristianini and J. Shawe-Taylor. *An introduction to support Vector Machines: and other kernel-based learning methods.* Cambridge University Press New York, NY, USA, 1999.

[10] B. V. Dasarathy. *Nearest neighbor (NN) norms: NN pattern classification techniques.* Los Alamitos: IEEE Computer Society Press, 1990, 1990.

[11] M.F. Duarte and Y. Hen Hu. Vehicle classification in distributed sensor networks. *J Parallel Distr Comput*, 64(7):826–838, 2004.

[12] Y. Fu, Q. Yang, R. Sun, D. Li, R. Zeng, C.X. Ling, and W. Gao. Exploiting the kernel trick to correlate fragment ions for peptide identification via tandem mass spectrometry. *Bioinformatics*, 20(12):1948–1954, 2004.

[13] M.G. Genton. Classes of kernels for machine learning: A statistics perspective. *J Mach Learn Res*, 2(2):299–312, 2001.

[14] TR Golub, DK Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, JP Mesirov, H. Coller, ML Loh, JR Downing, MA Caligiuri, et al. Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring. *Science*, 286(5439):531, 1999.

[15] T.K. Ho and E.M. Kleinberg. Building projectable classifiers of arbitrary complexity. *Proc. of the 13th International Conference on Pattern Recognition (ICPR-96)*, 2:880, 1996.

[16] C.W. Hsu, C.C. Chang, C.J. Lin, et al. A practical guide to support vector classification. Technical report, Department of Computer Science, National Taiwan University, 2003.

[17] JJ Hull. A database for handwritten text recognition research. *IEEE Trans Pattern Anal Mach Intell*, 16(5):550–554, 1994.

[18] RD King, C. Feng, and A. Sutherland. Statlog: comparison of classification algorithms on large real-world problems. *Appl Artif Intell*, 9(3):289–333, 1995.

[19] K. Lang. Newsweeder: Learning to filter netnews. *Proc. of the Twelfth International Conference on Machine Learning*, 331339, 1995.

[20] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[21] D.D. Lewis, Y. Yang, T.G. Rose, and F. Li. Rcv1: A new benchmark collection for text categorization research. *J Mach Learn Res*, 5:361–397, 2004.

[22] H.T. Lin and C.J. Lin. A study on sigmoid kernels for svm and the training of non-psd kernels by smo-type methods. Technical report, National Taiwan University, 2003.

[23] J.C. Platt. Fast training of support vector machines using sequential minimal optimization. *Advances in kernel methods: support vector learning table of contents*, pages 185–208, 1999.

[24] D. Prokhorov. Ijcnn 2001 neural network competition. slide presentation in ijcnn01, ford research laboratory, 2001.

[25] B. Schölkopf. *Support Vector Learning.* R. Oldenbourg Verlag, 1997.

[26] B. Schölkopf. The kernel trick for distances. *Adv Neural Inform Process Syst*, 13:301–307, 2001.

[27] B. Scholkopf, P. Simard, A. Smola, and V. Vapnik. Prior knowledge in support vector kernels. *Adv Neural Inform Process Syst*, 10:640–646, 1998.

[28] B. Schölkopf and A.J. Smola. *Learning with kernels: support vector machines, regularization, optimization, and beyond.* MIT Press, 2002.

[29] GF Smits and EM Jordaan. Improved SVM regression using mixtures of kernels. *Proc. of the 2002 International Joint Conference on Neural Networks (IJCNN'02)*, 3, 2002.

[30] Alexander J. Smola. Personal communication.

[31] M. Stone. Cross-validatory choice and assessment of statistical predictions. *J Roy Stat Soc B Stat Meth*, 36(2):111–147, 1974.

[32] V. Vapnik and O. Chapelle. Bounds on error expectation for support vector machines, 2000.

[33] V.N. Vapnik. *The Nature of Statistical Learning Theory.* Springer, 2000.

[34] H. Zhang, AC Berg, M. Maire, and J. Malik. SVM-KNN: Discriminative nearest neighbor classification for visual category recognition. *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2006)*, 2, 2006.

[35] A. Zien, G. Rätsch, S. Mika, B. Schölkopf, T. Lengauer, and K.R. Müller. Engineering support vector machine kernels that recognize translation initiation sites. *Bioinformatics*, 16(9):799–807, 2000.