Feature Space Interpretation of SVMs with Indefinite Kernels

Bernard Haasdonk

Abstract—Kernel methods are becoming increasingly popular for various kinds of machine learning tasks, the most famous being the support vector machine (SVM) for classification. The SVM is well understood when using conditionally positive definite (cpd) kernel functions. However, in practice, non-cpd kernels arise and demand application in SVMs. The procedure of "plugging" these indefinite kernels in SVMs often yields good empirical classification results. However, they are hard to interpret due to missing geometrical and theoretical understanding. In this paper, we provide a step toward the comprehension of SVM classifiers in these situations. We give a geometric interpretation of SVMs with indefinite kernel functions. We show that such SVMs are optimal hyperplane classifiers not by margin maximization, but by minimization of distances between convex hulls in pseudo-Euclidean spaces. By this, we obtain a sound framework and motivation for indefinite SVMs. This interpretation is the basis for further theoretical analysis, e.g., investigating uniqueness, and for the derivation of practical guidelines like characterizing the suitability of indefinite SVMs.

Index Terms—Support vector machine, indefinite kernel, pseudo-Euclidean space, separation of convex hulls, pattern recognition.

1 Introduction

In the last decade various so-called *kernel methods* for machine learning and data analysis have been developed and successfully applied. These methods do not necessarily require vectorial representations of the objects in contrast to many traditional methods. Instead, they are based on a problem specific choice of similarity measure between pairs of objects, the *kernel function*. By various possible choices of such functions, kernel methods are applicable to a wide range of structured or unstructured data types, e.g., general discrete structures [1], strings [2], weighted automata [3], etc. Every data analysis algorithm that only makes use of inner products between data vectors can be transformed into a kernel method by the *kernel trick*, which consists of replacing the inner product by an arbitrary kernel function. The most popular representatives of kernel methods are support vector machines (SVMs) for classification problems. In recent years, they have been established as methods of first choice on various learning problems in many fields of applications, cf. [4], [5]. There are several reasons for their success. The main arguments for practitioners are existing fast implementations and, the general ease of use, as in SVMs only few architectural decisions have to be taken: Only a positive definite kernel function and some parameters have to be provided. Even the choice of these few parameters can be automatized by model selection strategies, e.g., [6]. Therefore, the crucial component where the user can introduce some available problem specific a priori knowledge is the kernel function.

Very important arguments for theoreticians are the foundations in statistical learning theory and the clear intuitive geometric interpretation [7]. SVMs are hyperplane classifiers in implicitly defined Euclidean feature spaces. They perform optimal separation of patterns by margin maximization. This is the basis for general understanding, adequate practical application, improvements, and new algorithms. However, this geometric interpretation is only available in the case of conditionally positive definite (cpd) kernel functions (cf. Section 3 for definitions).

In practice, the requirement of a kernel function to be cpd turns out to be a very strict assumption. Many situations exist where standard cpd kernels are not applicable, as in the case of general nonvectorial data. Unless some specialized kernels exist, the user must construct a kernel function by hand. As a starting point, ad hoc or even sophisticated dissimilarity or similarity measures may be available, but they often produce non-cpd kernels. In other situations, standard kernels can be applied, but additional problem specific a priori knowledge needs to be incorporated in order to improve the method's performance. This also frequently leads to non-cpd kernels. We review various examples in Section 2. Therefore, non-cpd kernels often are available, but it is not clear what is the best way to use them in the SVM framework. A practical "heuristic" approach is to use the indefinite kernels in SVMs as usual. This has been realized in various publications [8], [9], [10], [11], [12]. The empirical classification results of such non-cpd kernels often are very good, but theoretical foundation is missing. Using these kernels has consequences for the numerical optimization problem, as convexity is lost. This gives rise to questions on the number and optimality of solutions.

The motivation for the present work now stems from these two facts: Good empirical results demand theoretical understanding and geometry is a fundamental step towards such understanding. We therefore concentrate on providing a geometric interpretation of training and classification of SVMs with non-cpd kernels. This interpretation enables further theoretical investigations, such as statements on

- *The author is with the Computer Science Department, Albert-Ludwigs-University Freiburg, 79110 Freiburg, Germany.*
E-mail: haasdonk@informatik.uni-freiburg.de.

Manuscript received 2 Oct. 2003; revised 14 Sept. 2004; accepted 16 Sept. 2004; published online 10 Feb. 2005.
Recommended for acceptance by L. Kuncheva.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number TPAMI-0302-1003.

optimality of solutions and derivation of practically relevant criteria for the application of indefinite SVMs.

The structure of the paper is as follows: In the next section, we review existing work dealing with indefinite kernels in SVMs. In Section 3, we introduce the necessary notations concerning kernels and the pseudo-Euclidean spaces, which is the framework where indefinite SVMs can be interpreted. The main part of the work starts in Section 4, where we illustrate linear classification in pseudo-Euclidean spaces by minimizing the distance of convex hulls. Section 5 then demonstrates that SVM classification exactly coincides with the pseudo-Euclidean convex hull classification and we present examples of the correspondences. Section 6 comments on theoretical statements on uniqueness of the solutions. We illustrate the practical aspects and implications of our work in Section 7 and conclude with final remarks in Section 8. Mathematical details, derivations and proofs are omitted in the main text, these can be found in the Appendix which appears at www.computer.org/publications/dlib.

2 Existing Work on Indefinite Kernels in SVMs

We give a brief literature review of existing work on indefinite kernels in SVMs. This will additionally accentuate the need for *theoretically* investigating these kernels, as the *experimental* results are increasingly dominating.

The first examples of such kernels are cpd kernels which are not positive definite. These kernels are known to produce convenient convex optimization problems for SVMs [13]. Some standard kernels are known to be cpd for suitable parameter ranges. For instance, the Sigmoid kernel, which is widely used in neural net design, has been investigated on cpd-ness [14]. This reference also presents relevant theoretical results. A main finding is that the widespread SVM implementation *libsvm* [15] does converge for indefinite kernels, namely, to a stationary point of the nonconvex optimization problem. In [16], the Sigmoid kernel is again addressed and an interpretation in a so-called hyperbolic space is given. For the interpretation of SVMs, we regard the embedding applied in the following section as more suitable, as it is a Euclidean embedding for cpd functions.

The major part of the literature consists of successful applications of non-cpd kernels which result from problem-specific kernel constructions. As one example, geometric transformation knowledge is known to be advantageous for image classification. If incorporated into kernel functions, this often leads to non-cpd kernels with good SVM classification results. Examples are the *jittering kernels* [9] approach, where the kernel evaluation is performed involving a set of locally transformed patterns, or *tangent distance kernels* [10], where tangents of the transformations are applied in the kernel evaluations. This example indicates a much wider field of indefinite kernels, namely, kernels constructed from distance measures. For instance, *Kullback-Leibler divergence kernels* [11] were constructed for objects represented as probability density functions. In [8], a problem specific dissimilarity measure for time sequences called *dynamic time warping* was incorporated into Gaussian kernels. More general *distance substitution kernels* were investigated in [17]. In order to avoid working with non-cpd kernels, regularization methods have been proposed, which aim at making the kernel matrix positive definite, e.g., [18], [19]. One obtains convex optimization problems by this, but other severe conceptional and computational problems arise, cf. Section 7.5. In general, few explanations for the success of non-cpd kernels are given. So, various publications gain sound mathematical foundation by the present work.

3 Notation

We will use the following general notations and terminology. x, b, f, etc., denote general variables or unstructured objects. $\mathbf{M}^T$ or $\mathbf{w}^T$ stand for the transpose of a matrix $\mathbf{M}$ or a vector $\mathbf{w}$. $\mathbf{I}_n$ is the $n \times n$ identity matrix. $\mathbf{1}_p, \mathbf{0}_p$, and $\mathbf{e}_i \in \mathbb{R}^p$ denote the vector of ones, zeros, and the ith unit-vector. The matrix $\mathrm{diag}(\mathbf{v}_1, \ldots, \mathbf{v}_m)$ is the diagonal matrix with entries given by the concatenation of the vectors $\mathbf{v}_i$. Sums will be abbreviated by $\sum_i := \sum_{i=1}^n$ and $\sum_{i,j} := \sum_{i,j=1}^n$.

Having a maximization problem $\max_{\boldsymbol{\alpha}} J(\boldsymbol{\alpha})$ under some constraints, we will use the notion *feasible point* $\boldsymbol{\alpha}$ for a point satisfying the constraints of the optimization problem. A *feasible direction* $\Delta\boldsymbol{\alpha}$ in $\boldsymbol{\alpha}$ will be a direction such that $\boldsymbol{\alpha} + \lambda\Delta\boldsymbol{\alpha}$ is a feasible point for some range $\lambda \in [0, \epsilon]$ with some $\epsilon > 0$. A *stationary point* $\boldsymbol{\alpha}$ of the maximization problem is a point where the derivatives of the optimization function in all feasible directions are nonpositive. A *local optimum* $\boldsymbol{\alpha}$ of a maximization problem is a stationary point where additionally the curvature of the optimization function in feasible directions with vanishing directional derivative is nonpositive. A stationary point includes possible saddle-points, which is the reason for considering the second order condition for local optima.

3.1 Kernels

A fundamental ingredient in SVMs is the notion of a *kernel* k, which is usually a symmetric function k taking two arguments of an arbitrary set $\mathcal{X}$ where the data stems from, i.e., $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$. For given data points $(x_i)_{i=1}^n \in \mathcal{X}^n$, which may be nonvectorial, the *kernel matrix* $\mathbf{K} := (k(x_i, x_j))_{i,j=1}^n$ can be defined. If for all n, all sets of data points and all vectors $\mathbf{v} \in \mathbb{R}^n$ the inequality $\mathbf{v}^T\mathbf{K}\mathbf{v} \geq 0$ holds, then k is called *positive definite*. If this is only satisfied for those $\mathbf{v}$ with $\mathbf{1}_n^T\mathbf{v} = 0$, then k is called *conditionally positive definite*. A kernel is *indefinite*, if for some $\mathbf{K}$ vectors $\mathbf{v}$ and $\mathbf{v}'$ exist with $\mathbf{v}^T\mathbf{K}\mathbf{v} > 0$ and $\mathbf{v}'^T\mathbf{K}\mathbf{v}' < 0$. The main contribution of the present work is the interpretation of SVMs for the case of non-cpd functions. Occasionally, we use the more expressive notion *indefinite*, which includes part of the cpd functions. For these functions, our result coincides with the existing Euclidean interpretations.

Starting with an arbitrary symmetric function k, a corresponding *squared distance* can be defined by

$$d^2(x, x') := k(x, x) - 2k(x, x') + k(x', x'). \qquad (1)$$

The synonym *dissimilarity* could be used, but for the sake of homogeneity we stick to the notion *distance*. In the case of a cpd kernel, this corresponds to the induced distance in a Euclidean feature space. For general symmetric kernels, this definition does not define the square of a metric, as d^2 might be negative or $\sqrt{d^2}$ violates the triangle inequality. But, at least, it yields a symmetric function d^2 with zero diagonal. This squared distance function will allow a representation of the data in certain vector spaces.

3.2 Pseudo-Euclidean Spaces

Our main argumentation is taking place in pseudo-Euclidean (pE) spaces. We briefly introduce our notation, details including illustrations are presented in Section 1.1 and in [20], [19].

With $\mathbb{R}^{(p,q)}$, we denote the pE space of signature (p,q), where $p,q \in \mathbb{N}_0$. This space can be seen as a product of a "real" and "imaginary" Euclidean vector space $\mathbb{R}^p \times i\mathbb{R}^q$. Its elements are denoted with $\mathbf{z}$, the vector of real coordinates. The bilinear but not necessarily positive definite *inner product* is defined by $\langle \mathbf{z}, \mathbf{z}' \rangle_{\mathrm{pE}} := \mathbf{z}^T \mathbf{M} \mathbf{z}'$, where $\mathbf{M} := \mathrm{diag}(\mathbf{1}_p, -\mathbf{1}_q)$. Usual geometric concepts can be straightforwardly defined: The *reduced convex hull* of a set of points is given by $\mathrm{conv}_\mu(\{\mathbf{z}_1, \dots, \mathbf{z}_n\}) := \{\sum_i \alpha_i \mathbf{z}_i | \sum_i \alpha_i = 1 \text{ and } 0 \leq \alpha_i \leq \mu\}$, where μ balances the reduction from $\mu = 1$ (ordinary non-reduced convex hull) until $\mu = 1/n$, where the set consists of the single point, the mean of the $\mathbf{z}_i$. The *squared norm* is defined as $||\mathbf{z}||^2_{\mathrm{pE}} := \langle \mathbf{z}, \mathbf{z} \rangle_{\mathrm{pE}}$. This notion immediately implies the *squared distance* of two points by $||\mathbf{z} - \mathbf{z}'||^2_{\mathrm{pE}}$. The mapping $\mathbf{Mz}$ defines the reflection of a vector in $\mathbb{R}^{(p,q)}$ with respect to the real space $\mathbb{R}^p$. *Orthogonality*, *hyperplanes*, and *normal vectors* can be reasonably defined and corresponding linear classification can be performed. Note that the squared norm and the squared distance can be negative in contrast to the Euclidean case. In particular, $||\mathbf{z} - \mathbf{z}'||^2_{\mathrm{pE}}$ may not define a metric, as it can violate the triangle inequality.

The relevance of these pE spaces is that they provide a unifying framework for both structural and vectorial data after appropriate embeddings [20]. By assuming a squared distance function on arbitrary structured data, an embedding in a pE space can be constructed, which allows to maintain this distance information. It was given by [20] and also used in [18], [19]. A concrete construction is given in the proof of Proposition 1 in the Appendix.

Proposition 1 (Isometric Embedding). *Let* $\{x_i\}_{i=1}^n \in \mathcal{X}^n$ *be data points and* $d^2 : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$ *be a symmetric function with zero diagonal. Then, there exists a pE space* $\mathbb{R}^{(p,q)}$ *with* $p + q < n$ *and an embedding* $\Phi : \{x_i\}_{i=1}^n \rightarrow \mathbb{R}^{(p,q)}$ *such that for all* i, j *holds*

$$d^2(x_i, x_j) = ||\Phi(x_i) - \Phi(x_j)||^2_{\mathrm{pE}}. \tag{2}$$

The signature (p,q) is given by the number p of positive and the number q of negative eigenvalues of a so-called *centered* kernel matrix constructed from the distance information. The most important point for our purpose is that the distance data can be induced by a kernel via (1). Then, the pE inner product and $-\frac{1}{2}d^2$ differ from k only by a suitable function h and h' of one argument, which is useful for various reformulation steps, cf. Lemma 6 in the Appendix for details:

$$\begin{aligned} k(x_i, x_j) &= -\frac{1}{2} d^2(x_i, x_j) + h'(x_i) + h'(x_j) \\ &= \Phi(x_i)^T \mathbf{M} \Phi(x_j) + h(x_i) + h(x_j). \end{aligned} \tag{3}$$

The embedding can be performed in such a way that it becomes centered (mean $\mathbf{0}$) and the coordinates are uncorrelated. In particular, for cpd kernels, this resulting space will be Euclidean. For many embeddings of real data, the variance in the imaginary directions is empirically much lower than the variance in the real directions [19].

In practice, kernels mostly produce nonnegative squared distances by (1), for instance, every kernel constructed from a Gaussian $k(x, x') = e^{-f(x,x')}$ with nonnegative f and $f(x,x) = 0$. Still, the study of negative squared distances is relevant in the case of such practical kernels. The reason is that constructions in the embedding pE space, like convex combinations of points, can result in negative squared distances even if the squared distances between the embedded points are all positive.

A slightly more abstract framework called Krein or Pontryagin-spaces [21] for embedding the whole original space $\mathcal{X}$ would also have been an appropriate framework for our interpretation, if we would confine ourselves to symmetric kernels which allow a so-called *Kolmogorov decomposition* [21]. But, as we demonstrate in the next section, only the space resulting from embedding the (finite) training data is required for understanding the SVM. So, we stick to the more easily accessible class of finite-dimensional Krein-spaces, which exactly are the pE spaces. In this finite-dimensional case, we do not have to make any further restrictions on the kernels other than being symmetric functions.

4 Optimal Separation of Convex Hulls in $\mathbb{R}^{(p,q)}$

Different methods of linear classification in pE spaces have been proposed, e.g., the Fisher linear discriminant or a generalized nearest mean classifier [20], [19]. Also, methods for using SVMs in these spaces have been proposed, which require the regularizations mentioned in Section 2 [18], [19]. In this section, we present another classification procedure which maintains and makes use of the pE geometry of the spaces. It turns out that it is an optimal hyperplane classification method and exactly the operation that is performed by a non-cpd SVM.

Note that the method described in the sequel only requires distance information. Therefore, we formulate it solely in terms of distances and completely avoid the kernel function, which induces the distance. So, the method may be particularly attractive for the active research field of distance based learning. If required, all expansions in terms of distances can be expressed by the kernel function by application of (1) and (3). The classification method is based on convex hulls; therefore, we denote it as *CH classification*.

It has been shown that maximization of the (soft) margin in Hilbert-spaces can equivalently be formulated as minimization of distances of (reduced) convex hulls [22], [23]. This separation of convex hulls can be applied in pE spaces, as we also have the notions of distance and convex hulls. As in Section 3, we assume to have training data $(x_i, y_i) \in \mathcal{X} \times \{\pm 1\}$ for $i = 1, \dots, n$ and an arbitrary squared distance measure $d^2 : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$, which can be given explicitly or induced by (1) based on some symmetric kernel function k. This data is assumed to be isometrically embedded in some $\mathbb{R}^{(p,q)}$ according to Proposition 1.

The formalization of minimizing the distance in $\mathbb{R}^{(p,q)}$ between the reduced convex hulls of the positive and of the negative training examples is

$$\min_{\mathbf{z}^-, \mathbf{z}^+} \quad ||\mathbf{z}^- - \mathbf{z}^+||^2_{\mathrm{pE}} \tag{4}$$

$$\text{s.t.} \quad \mathbf{z}^\pm \in \mathrm{conv}_\mu\{\Phi(x_i) | i : y_i = \pm 1\}. \tag{5}$$

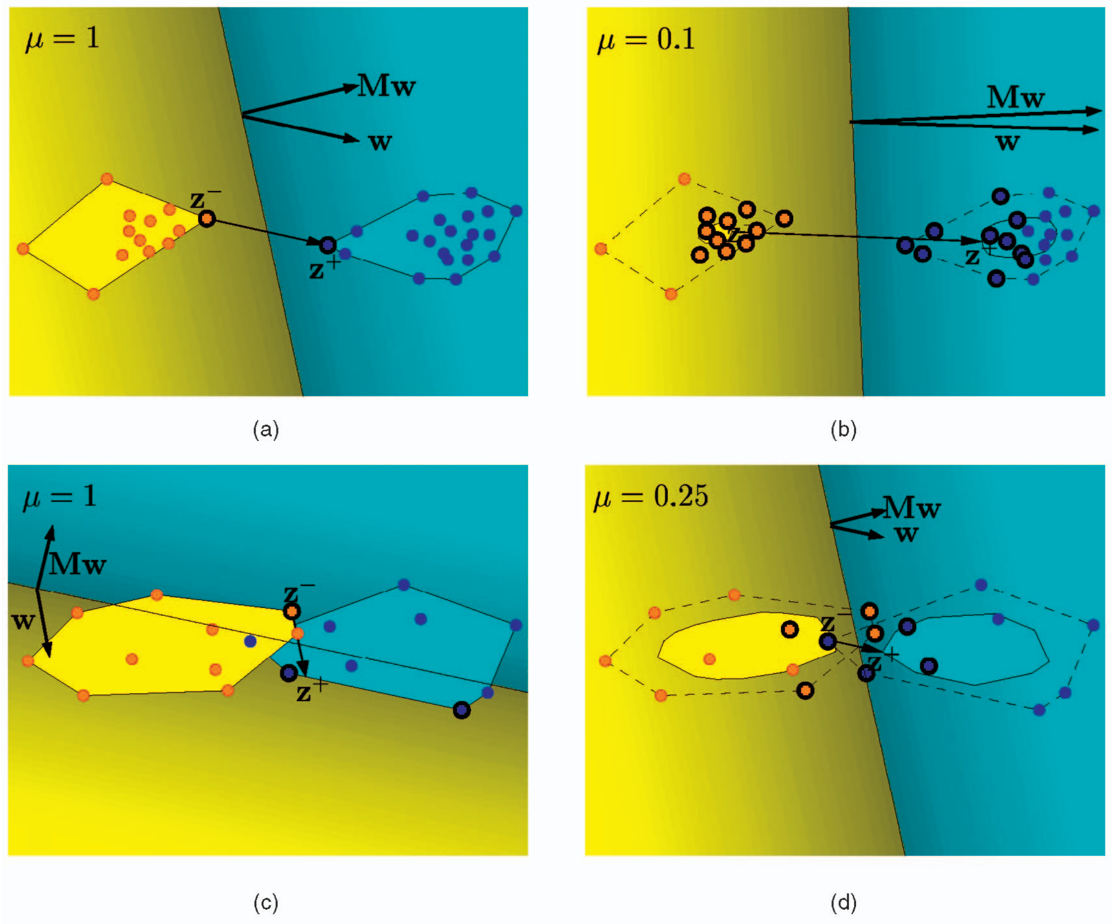


Fig. 1. Illustration of pseudo-Euclidean CH classification. (a) Separable data with convex hulls, (b) separable data with reduced convex hulls, (c) nonseparable data with convex hulls, and (d) nonseparable data with reduced convex hulls.

This can be expressed by distances between training points as described in the Appendix. This reformulation mainly makes use of rewriting $\mathbf{z}^+$ and $\mathbf{z}^-$ as convex combinations $\mathbf{z}^\pm = \sum_{i:y_i=\pm 1} \bar{\alpha}_i \Phi(x_i)$ with $\sum_{i:y_i=+1} \bar{\alpha}_i = 1$, $\sum_{i:y_i=-1} \bar{\alpha}_i = 1$, and $0 \leq \bar{\alpha}_i$. We obtain the dual optimization problem minimizing the distance between the convex hulls which we will refer to as (CH-DU)

$$\max_{\bar{\alpha}_1,\ldots,\bar{\alpha}_n} \quad \frac{1}{2}\sum_{i,j} \bar{\alpha}_i \bar{\alpha}_j y_i y_j d^2(x_i, x_j)$$

$$\text{s.t. } 0 \leq \bar{\alpha}_i \leq \mu, \quad \sum_i \bar{\alpha}_i y_i = 0 \quad \text{and} \quad \sum_i \bar{\alpha}_i = 2.$$

We choose the notation $\bar{\alpha}_i$ in order to discriminate from SVM-related variables α_i later on. Note that this optimization problem is quadratic, but not necessarily convex, as the quadratic form can be indefinite. This can cause phenomena like multiple local optima as we illustrate in Section 6. The existence of an optimum is trivial as the feasible domain is bounded. Note also that the optimization function in (CH-DU) can be replaced by $-\sum_{i,j} \bar{\alpha}_i \bar{\alpha}_j y_i y_j k(x_i, x_j)$ due to Lemma 6 in the Appendix.

The natural classifier in $\mathbb{R}^{(p,q)}$ based on a feasible point $\bar{\boldsymbol{\alpha}}$ from (CH-DU) is the minimum distance classifier with respect to the two points $\mathbf{z}^+$ and $\mathbf{z}^-$, i.e., the sign of

$$g(\mathbf{z}) = ||\mathbf{z} - \mathbf{z}^-||^2_{\text{pE}} - ||\mathbf{z} - \mathbf{z}^+||^2_{\text{pE}}. \tag{6}$$

Similar to the Euclidean case, this is a hyperplane classifier. For the image of a training point $\mathbf{z} = \Phi(x)$, the classification rule can be expressed in the original space $\mathcal{X}$ without explicit embedding, as is derived in the Appendix:

$$f(x) = -\sum_i \bar{\alpha}_i y_i d^2(x_i, x) + b \quad \text{with}$$
$$b = \frac{1}{2}\sum_{i,j} \bar{\alpha}_i \bar{\alpha}_j y_i d^2(x_i, x_j). \tag{7}$$

We now argue why this classification rule can be applied to the whole (possibly infinite) space $\mathcal{X}$. If an arbitrary $x \in \mathcal{X}$ has to be classified, we imagine (a possibly different) isometric embedding Φ' in a pE space $\mathbb{R}^{(p',q')}$, where x is simultaneously embedded with the training data. As the training procedure (CH-DU) and the classification rule (7) are independent of the specific embedding, training, and classification of $\Phi'(x)$ will exactly result in the decision rule (7). At this point, we see that it is no limitation that the embedding is data-dependent, as we do not explicitly make use of it.

So, we obtain a classification method for arbitrary symmetric distance data, which is an optimal hyperplane classifier in the sense that it is the minimum distance classifier with respect to closest points of convex hulls, where closeness is measured with the pE norm. Fig. 1 gives an illustration of the classification behavior of the classifier on simple data embeddings $\{\Phi(x_i)\}_{i=1}^n$ in the low-dimensional pE space $\mathbb{R}^{(1,1)}$. Here, and in all subsequent illustrations of $\mathbb{R}^{(1,1)}$, the real space is plotted horizontally, the imaginary part vertically, and both axes are identically scaled. The absolute scale and position of the axes are irrelevant in all figures; therefore, we omit further annotation with units or axes.

The CH classifier was trained by numerically solving (CH-DU) for different values of μ and the classification was performed by evaluations of (7) for a suitable rectangle in $\mathbb{R}^{(1,1)}$. The resulting decision values of the classifier are color-coded. The resulting vectors with $\bar{\alpha}_i > 0$, which we denote as *support vectors* (SVs) as usual in SVMs, are marked with a bold circle. Additionally, the normal $\mathbf{w} := \mathbf{z}^+ - \mathbf{z}^- = \sum_i y_i \bar{\alpha}_i \Phi(x_i)$ and its reflected version $\mathbf{Mw}$ are indicated. The classification boundary is the line passing through the midpoint of $\mathbf{z}^+\mathbf{z}^-$, which is orthogonal (in pE sense) to $\mathbf{w}$. Note that pE-orthogonality intuitively corresponds to Euclidean orthogonality after reflecting one vector with respect to the real space, here the horizontal axis. For separable data, the parameter μ is set to $\mu = 1$ in Fig. 1a and lowered to $\mu = 0.1$ in Fig. 1b. Both solutions separate the data very nicely, the lower μ results in more SVs. Fig. 1c illustrates that for nonseparable data with overlapping convex hulls, the CH classification for $\mu = 1$ does not seem to be reasonable. These cases, can however, often be solved by reducing μ until separation is possible, e.g. $\mu = 0.25$ in Fig. 1d. So, both separable and nonseparable data can be successfully discriminated by the CH classifier.

The optimization problem (CH-DU) is similar to the ν-SVM dual [24] and we similarly can set up the corresponding convex hull primal optimization problem, see Proposition 7 in the Appendix for details. The main insight from this is, that for cases where $\mathbf{w}$ and $\mathbf{Mw}$ point to the same side of the decision line or, equivalently, $\mathbf{w}^T\mathbf{Mw} > 0$, the resulting solution will possess the nice property of correctly classifying all non-SVs, which is illustrated in Figs. 1a, 1b, and 1d. Fig. 1c indicates that this cannot be guaranteed for cases where $\mathbf{w}^T\mathbf{Mw} \leq 0$. Also, intuitively this case is problematic. The points of minimum distance ($\mathbf{z}^+$ and $\mathbf{z}^-$) are no longer located on the Euclidean closest boundaries. A crucial assumption of distance based classification is violated which requires "lower (squared) distance" meaning "higher similarity": In the example, points have negative squared distance to points from the convex hull of the other class, but have 0 distance to themselves. So, the requirement of $\mathbf{w}^T\mathbf{Mw} > 0$ for a solution is relevant for theoretical desirable properties and a geometric suitable interpretation.

5 Interpretation of Indefinite SVMs in $\mathbb{R}^{(p,q)}$

In this section, we establish the geometric interpretation of non-cpd SVMs in pseudo-Euclidean spaces. Assuming training data $(x_i, y_i) \in \mathcal{X} \times \{\pm 1\}$ for $i = 1, \ldots, n$ and a symmetric kernel function k, the usual SVM classification approach solves the dual optimization problem which we will refer to as (SVM-DU)

$$\begin{aligned} \max_{\alpha_1,\ldots,\alpha_n} \quad & \sum_i \alpha_i - \frac{1}{2}\sum_{i,j} \alpha_i \alpha_j y_i y_j k(x_i, x_j) \\ \text{s.t.} \quad & 0 \leq \alpha_i \leq C \quad \text{and} \quad \sum_i \alpha_i y_i = 0. \end{aligned}$$

Here, $C > 0$ is a factor penalizing data fitting errors. The classification of new patterns x is then based on the sign of

$$f(x) = \sum_i \alpha_i y_i k(x_i, x) + b, \tag{8}$$

where b is determined such that f has identical absolute values on unbounded SVs as is identically done in the following proposition. In the case of a cpd kernel k, this procedure is well established and can be understood as an optimal hyperplane classifier in a Euclidean space, e.g., after the kernel PCA-map [5].

In the case of arbitrary indefinite k, we can state the primal optimization problem corresponding to (SVM-DU). We emphasize, however, that there is no strict "duality" between primal and dual solutions in nonconvex optimization problems. But, we keep the notions to emphasize the relation to the cpd case. The statement is that these SVMs in both separable and nonseparable cases have a similar primal target as ordinary SVMs. The proof is skipped in this presentation since it is similar to Proposition 7 presented in the Appendix.

Proposition 2 (SVM Primal in $\mathbb{R}^{(p,q)}$). *Let $\boldsymbol{\alpha}$ be a stationary point of (SVM-DU) for arbitrary symmetric k, such that there exist two nonbounded coefficients of different classes, i.e., $0 < \alpha_k, \alpha_l < C$ with $y_k = +1$ and $y_l = -1$. Let $\Phi : \{x_i\}_{i=1}^n \to \mathbb{R}^{(p,q)}$ be an isometric embedding according to Proposition 1 corresponding to the squared distance measure (1) induced by k. Then, we obtain a stationary point $\mathbf{w} \in \mathbb{R}^{(p,q)}$, $b \in \mathbb{R}$, and $\boldsymbol{\xi} \in \mathbb{R}_+^n$ of the primal optimization problem*

$$\begin{aligned} \min_{\mathbf{w},b,\xi} \quad & \frac{1}{2}\mathbf{w}^T\mathbf{Mw} + C\sum_i \xi_i \\ \text{s.t.} \quad & y_i(\mathbf{w}^T\mathbf{M}\Phi(x_i) + b) \geq 1 - \xi_i \quad \text{and} \quad 0 \leq \xi_i \end{aligned} \tag{9}$$

by setting $\mathbf{w} = \sum_i \alpha_i y_i \Phi(x_i)$, $b := -\frac{1}{2}\mathbf{w}^T\mathbf{M}(\Phi(x_k) + \Phi(x_l))$ and

$$\xi_i := \begin{cases} 1 - y_i(\mathbf{w}^T\mathbf{M}\Phi(x_i) + b) & \textit{if} \quad \alpha_i = C \\ 0 & \textit{otherwise.} \end{cases}$$

Note that the solution is independent of the choice of k and l as long as they satisfy the stated requirements. The relation between this primal and the common SVM primal is that the common squared norm and inner products are replaced by the corresponding pE notions. If $\mathbf{M} = \mathbf{I}_n$, we perfectly recover the common SVM primal.

The relevance of this result is twofold. First, we recover a finding from [14], which states that a stationary point of (SVM-DU) satisfies certain separability constraints, which turn out to be equivalent to our constraints of the SVM primal. They imply that a non-cpd SVM is a reasonable classifier in the sense that similar to an ordinary SVM it correctly classifies the training data with $\alpha_i < C$ and $\alpha_i = C$, $\xi_i < 1$. Examples with $\alpha_i = C$, $\xi_i > 1$ are wrongly classified. This also holds in the case of $\mathbf{w}^T\mathbf{Mw} \leq 0$, in contrast to the CH classifier. The second important conclusion from this proposition is that we found the regularizer $\mathbf{w}^T\mathbf{Mw}$. A geometrical margin can be defined in analogy to the cpd case as $2/\sqrt{\mathbf{w}^T\mathbf{Mw}}$, if the regularizer is positive. This might be a starting point for learning theoretic investigations, which could give further insights in usability and provide a further theoretic underpinning of non-cpd SVMs.

An important comment is appropriate at this point. We have a notion of *margin* for a non-cpd SVM and Proposition 2 indicates that training is *related* to maximizing this quantity in

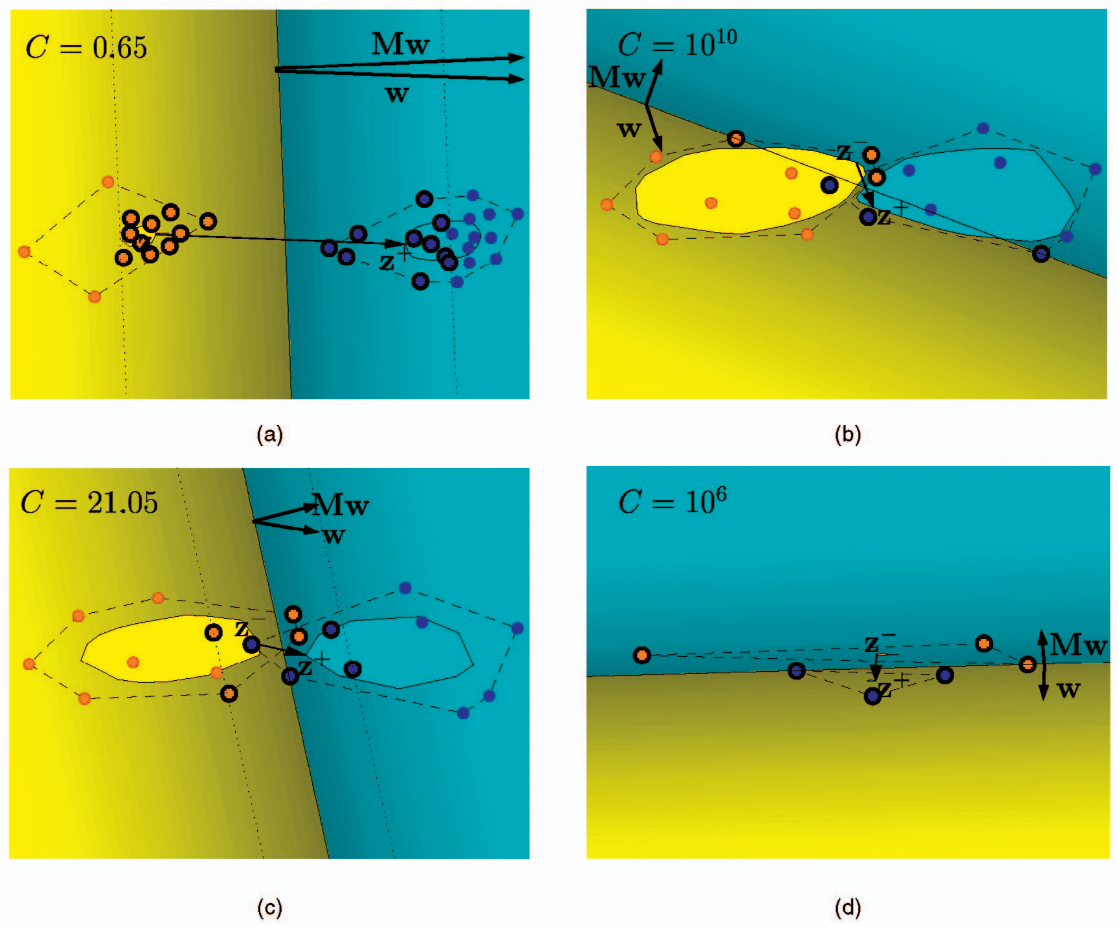


Fig. 2. Linear non-cpd SVM classification examples in $\mathbb{R}^{(1,1)}$ and the corresponding reduced convex hulls of the CH classifier. (a) Separable case, (b) nonseparable case with large C, (c) nonseparable case with small C, and (d) unsuitable non-cpd SVM classification.

the sense that we obtain a stationary point. However, in general, *training of a non-cpd SVM is not identical to margin maximization*. The reason is that margin maximization by (9) is, in general, not well defined. We give a separable two point example: Take $y_1 = +1, y_2 = -1$, and $\mathbf{x}_i = (y_i, 0)^T \in \mathbb{R}^{(1,1)}$. One can easily check that $\mathbf{w} := (1, \lambda)^T, b = 0$, and $\xi = (0,0)^T$ satisfy the constraints with equality, however, the optimization value $\mathbf{w}^T\mathbf{M}\mathbf{w} = 1 - \lambda^2$ diverges to $-\infty$ as λ increases. This is a fundamental difference to the cpd case.

So, margin maximization is not the right interpretation of non-cpd SVMs, instead optimal separation of convex hulls is adequate, which will be formulated and proven in the following. In addition to a justification for using non-cpd SVMs, we obtain a constructive interpretation. This allows easy understanding of SVM classification as the operation of separating convex hulls is geometrically easily accessible.

We now present the main result, which settles the relation between solutions of (CH-DU) and (SVM-DU) and corresponding decision boundaries. The proposition states that a local optimum of (SVM-DU) implies a local optimum of (CH-DU) for certain μ. The implication in the other direction however is not valid in general.

For the cpd case the statements follow from [23], [24]. However, their proofs explicitly use the Euclidean primal and positive definiteness, e.g., for optimality only first order derivative conditions have to be checked. We present a detailed proof for the case of arbitrary symmetric k. It emphasizes that the correspondences only rely on properties of the quadratic problems and do not require primal solutions. We use $\mathbf{Q}$ for the matrix with entries $Q_{ij} = y_i y_j k(x_i, x_j)$.

Proposition 7 (Equivalence of CH and SVM). *Let k be an arbitrary symmetric function and d^2 be the induced squared distance as given in (1).*

1. *A nonzero stationary point $\boldsymbol{\alpha}$ of (SVM-DU) induces a stationary point $\bar{\boldsymbol{\alpha}} := 2\boldsymbol{\alpha}/\sum_i \alpha_i$ of (CH-DU) with $\mu = 2C/\sum_i \alpha_i$.*

   If additionally $\boldsymbol{\alpha}$ is a local optimum, then $\bar{\boldsymbol{\alpha}}$ is a local optimum.
2. *A stationary point $\bar{\boldsymbol{\alpha}}$ of (CH-DU) induces a stationary point $\boldsymbol{\alpha} := \bar{\boldsymbol{\alpha}}/\rho$ of (SVM-DU) with $C = \mu/\rho$, if there are two unbounded coefficients of opposite classes, i.e., $0 < \bar{\alpha}_k, \bar{\alpha}_l < \mu, y_k = +1$ and $y_l = -1$, such that $\rho := \frac{1}{2}\bar{\boldsymbol{\alpha}}^T\mathbf{Q}(\mathbf{e}_k + \mathbf{e}_l)$ is positive.*

   If additionally $\bar{\boldsymbol{\alpha}}$ is a local optimum, then $\boldsymbol{\alpha}$ is a local optimum in the case of $\mathbf{Q}$ being positive semidefinite in all feasible directions $\boldsymbol{\Delta\alpha}$ of (SVM-DU) with $\langle \mathbf{1}_n - \mathbf{Q}\boldsymbol{\alpha}, \boldsymbol{\Delta\alpha}\rangle = 0$.
3. *In both Cases 1 and 2, the corresponding decision planes defined by (7) and (8) are parallel and even identical if $\boldsymbol{\alpha}$ or $\bar{\boldsymbol{\alpha}}$ are not upper bounded.*

We present some examples in Fig. 2 which depicts the classification behavior of non-cpd SVMs and the relation to CH classification in the illustrative low-dimensional feature

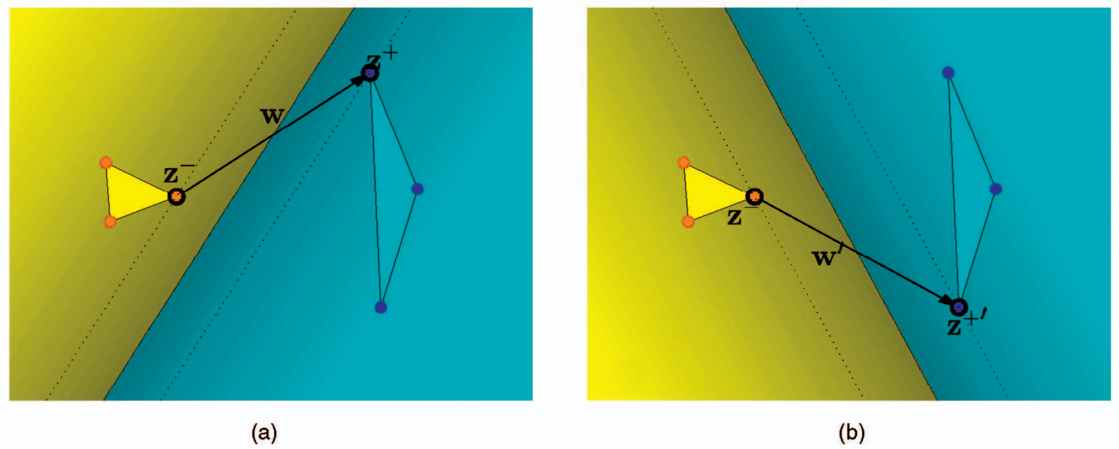


Fig. 3. Illustration of multiple local optima of non-cpd SVM classification. (a) Initial solution and (b) solution after training set permutation.

space $\mathbb{R}^{(1,1)}$. An easy way of obtaining such a feature space is by directly assuming $\mathcal{X} = \mathbb{R}^2$ with the non-cpd kernel function $k(\mathbf{x}, \mathbf{x}') := x_1 x_1' - x_2 x_2'$. An isometric embedding then is interpreting $\mathbf{x}$ as element of $\mathbb{R}^{(1,1)}$. Since every SVM solution has a corresponding CH classifier, we additionally visualize the corresponding (reduced) convex hulls, the normal $\mathbf{w}$ of the CH classifier, which is a scaled version of the normal defined by the SVM, and the reflected normal $\mathbf{Mw}$.

The situation presented in Fig. 1a already illustrates the "hard margin" case: The figure not only demonstrates CH classification, but this coincides with the corresponding SVM for sufficiently large C according to Proposition 3, Cases 1 and 2. Also, in the "soft margin" formulation, where some α_i are bounded by C the result of a non-cpd SVM is interpretable, cf. Fig. 2a. We choose C as indicated by Proposition 3, Case 1, such that it corresponds to μ from Fig. 1b. So, the decision line in Fig. 2a is due to Proposition 3, Case 3 not an exact CH classification, but is slightly shifted compared to Fig. 1b.

In the previous section, we have explained that CH classification is counterintuitive in situations where $\mathbf{w}^T\mathbf{Mw} \leq 0$. On the other hand, every SVM decision plane corresponds to some CH classification plane by simple positive scaling. Therefore, the requirement of $\mathbf{w}^T\mathbf{Mw} > 0$ transfers identically to SVM classifiers. This can be interpreted as follows: For ordinary SVMs, the requirement for successful application is that the data is reasonably separable by a hyperplane in some implicit Hilbert-space. For non-cpd SVMs linear separability is not enough, separability with a hyperplane that has *positive norm* is required.

As an example where the SVM solution violates this condition, we recall Fig. 1c, which demonstrates unreasonable CH classification. This CH solution does *not* have a correspondence to any SVM solution, so this is a confirmation that the conditions in Proposition 3, Case 2 are not superfluous as for the other direction Case 1. So, the question arises, what is the result of an SVM on the data set in Fig. 1c. Fixing some large value, e.g., $C = 10^{10}$ yields the result illustrated in Fig. 2b. The resulting $\mathbf{w}$ has negative squared norm, but it still guarantees, in contrast to the CH classifiers, that non-SVs are correctly classified and only bounded SVs are candidates to be misclassified. In the case of this data set, a lowering of C results in a decision boundary with positive squared norm of $\mathbf{w}$, cf. Fig. 2c, which seems to be reasonable, and is parallel to the decision line of Fig. 2d. However, in other cases, e.g., Fig. 1d, no choice of C can produce a decision normal with positive squared norm. The reason is that the class means have negative squared distance. Note that this data is even well-behaved in the sense that only positive squared distances between training objects are at hand, which is a realistic assumption for kernels in practice. Obviously, constructions like convex combinations can produce negative squared distances, even in such realistic cases. In the Figs. 2b and 2d examples, the objective value of the optimization problem diverges to $-\infty$ if $C = \infty$. This has been observed and discussed from the dual point of view in [14], as it can happen for the sigmoid kernel if the constraint $\sum_i y_i \alpha_i = 0$ is removed.

6 UNIQUENESS OF SOLUTIONS

In this section, we want to exemplify how the presented interpretation can serve as a basis for further theoretical analysis. We address the question of uniqueness of non-cpd SVMs and derive some simple statements.

An easy example in Fig. 3 illustrates that, in general, uniqueness cannot be expected. The illustrated data set in $\mathbb{R}^{(1,1)}$ obviously has two local optimal SVM/CH decision lines. These indeed are found if using different optimization algorithms, e.g., libsvm [15] or the QP-solver from MATLAB, or by permuting the training set. One can easily see that by shifting $\mathbf{z}^{+\prime}$ down one can obtain arbitrary norms of $\mathbf{w}'$, so different local solutions can have arbitrarily differing values of the optimization function.

In this example, the points $\mathbf{z}^+$ and $\mathbf{z}^{+\prime}$ lie on the vertices of a line. By adding further imaginary dimensions, i.e., increasing q, one can easily obtain examples where the multiple solutions $\mathbf{z}^+$ are located on the vertices of a square for $q = 2$, of a cube for $q = 3$, etc. This is the basic idea for the (omitted) proof of the following lemma.

Lemma 4 (Exponential Number of Local Optima). *For every $q \geq 1$ there exist kernels and points with suitable labeling, such that the data can be embedded to $\mathbb{R}^{(1,q)}$ according to Proposition 1 and the corresponding optimization problem (SVM-DU) has 2^q local optima, which all perform correct separation.*

The other extreme situation is uniqueness of solutions. As a simple result, we refer to Fig. 1a, which demonstrates the uniqueness of a global optimum, local optimum and stationary point in the case of a non-cpd SVM. This uniqueness does not only hold in the case of this low-dimensional example, but can be extended to $\mathbb{R}^{(1,q)}$ under similar conditions. The proof can be found in the Appendix.

Lemma 5 (Uniqueness of Stationary Points). *If the given training data with a non-cpd kernel induces an isometric embedding to $\mathbb{R}^{(1,q)}$ and (SVM-DU) has a stationary point $\boldsymbol{\alpha}$ such that the corresponding $\mathbf{z}^+$ and $\mathbf{z}^-$ have positive squared distance, and they have positive squared distance to all points of their corresponding convex hulls, then $\mathbf{z}^+$ and $\mathbf{z}^-$ are unique.*

This particularly implies that the stationary point is the global optimum. This is an improbable situation, but a quite remarkable result as it states uniqueness for kernel functions which are extremely non-cpd in the sense that their (centered) kernel matrices have only one positive eigenvalue. This indicates that the number of local optima may be very well behaved under certain assumptions. More detailed investigations concerning number, quality, and relation of local optima or conditions for uniqueness might be promising.

7 Practical Implications

The obtained interpretation not only allows further theoretical investigation, but also has useful implications for practice. We comment on implications for distance-based classification and mainly on indicators for application of non-cpd SVMs. The interpretation provides us with different nontrivial criteria for checking the suitability or unsuitability of a given kernel or a trained SVM. The feature space interpretation additionally gives hints on possible parameter modifications.

7.1 CH Classification

For distance-based learning, we have formulated the CH classification procedure, which seems to be new for general symmetric distance data d. Due to the equivalence to the SVM, this classification procedure can be realized by using $k(x,x') := -\frac{1}{2}d(x,x')^2$ in an SVM. Whether (CH-DU) or (SVM-DU) is chosen, does not make much difference in computation time. Both formulations can be solved efficiently by available C-SVM or ν-SVM implementations, e.g., described in [25]. A minor advantage of (SVM-DU) is that in contrast to (CH-DU) any solution guarantees geometric interpretation of the obtained coefficients: Only vectors with bounded α_i can be wrongly classified, all points with $\alpha_i = 0$ are correctly classified.

7.2 SVM Training Preliminaries

Some comments on the practical solution of the nonconvex optimization problems are in order. Algorithmically, nonconvex optimization problems are known to be NP-hard [26]. Approaches for global optimization are much more complex than local optimizers since the global optimizers have to keep track of the multiple candidates for global optima. Many such methods are branch and bound approaches, employing *branching* by subdivision of the feasible domain and *bounding* the optimization function on the components of the subdivisions. This conceptionally remarkably increased computational complexity might be the reason for missing large scale implementations. Therefore, practical approaches for solving such problems must be suboptimal. We have seen in Lemma 5 that even in extreme nonconvex optimization problems, unique solutions are possible, which can be found by local solvers. So, in practice, such optimizers that find local optima or stationary points seem to be a good choice for real-world problems, as is indicated by the existing experimental results, cf. Section 2. Random initializations of the optimization algorithm as performed in [14] could be applied for improving the objective value of local solutions.

Care has to be taken, as not all quadratic programming algorithms can be used for the nonconvex problems. A requirement for successful application is to use implementations, which have explicitly been designed for dealing with nonconvex optimization problems. In particular, they should terminate and allow some optimality statement of the solution, as it exists for libsvm [14] and the derived package LIBSVMTL [27]. To prevent occasional divergence with such suitable implementations, the penalty parameter C should be set to some finite value. Note also that many (nicely separable) problems can also be solved with $C = \infty$.

7.3 Suitability Criteria Before SVM Training

In some situations, it is possible to predict the unsuitability of a non-cpd SVM or kernel before training or even data-independently.

Negated cpd kernels. A kernel matrix which is the negative of a cpd matrix produces $\mathbf{w}^T\mathbf{M}\mathbf{w} \leq 0$ independently of the y_i or α_i, as $\mathbf{M}$ will only have negative entries. This is particularly satisfied, if k is the negative of a cpd function. Thus, kernels like $k(\mathbf{x},\mathbf{x}') = -\langle \mathbf{x},\mathbf{x}'\rangle$ or $k(\mathbf{x},\mathbf{x}') = -e^{-\gamma||\mathbf{x}-\mathbf{x}'||^2}$ are inadequate choices for SVMs, independent of the given training data.

Number of negative eigenvalues. In general, an increasing number of negative eigenvalues of the kernel matrix makes $\mathbf{w}^T\mathbf{M}\mathbf{w} \leq 0$ more likely, so useful separation of the training data is getting more difficult. Therefore, the number of negative eigenvalues of the kernel matrix is a rough criterion of how difficult it is to obtain a suitable solution by training the corresponding SVM.

Squared distance of class means. A more precise indicator for possible separability with positive $\mathbf{w}^T\mathbf{M}\mathbf{w} \leq 0$ can be obtained from the class means $\mathbf{m}^{\pm} := \frac{1}{n_\pm}\sum_{y_i=\pm 1}\Phi(x_i)$ in the pE feature space, where $n_\pm$ denotes the number of positive/negative examples. These points lie within every reduced convex hull of their respective point sets. So, their squared distance is an upper bound for $\mathbf{w}^T\mathbf{M}\mathbf{w}$. If this squared distance is negative, every CH/SVM classifier will result in $\mathbf{w}^T\mathbf{M}\mathbf{w} < 0$. In the case of equal numbers $n_+ = n_-$, positivity of $||\mathbf{m}^- - \mathbf{m}^+||^2_{\mathrm{pE}}$ implies that there exists a CH/SVM solution with positive $\mathbf{w}^T\mathbf{M}\mathbf{w}$ if the parameters μ or C are sufficiently lowered. The computation of the squared distance of class means does not require the explicit feature space embedding, but can be obtained by $||\mathbf{m}^- - \mathbf{m}^+||^2_{\mathrm{pE}} = \mathbf{c}^T\mathbf{K}\mathbf{c}$, where $\mathbf{c}$ is chosen as $c_i := 1/n_+$ if $y_i = +1$ and $c_i = -1/n_-$ otherwise.

7.4 Suitability Criteria after SVM Training

After successful training, further criteria are available for getting insights in the quality of the resulting classifier. These can be obtained without performing an explicit feature space mapping.

Bound on the training error. As we have seen in the interpretation of the SVM primal problem following

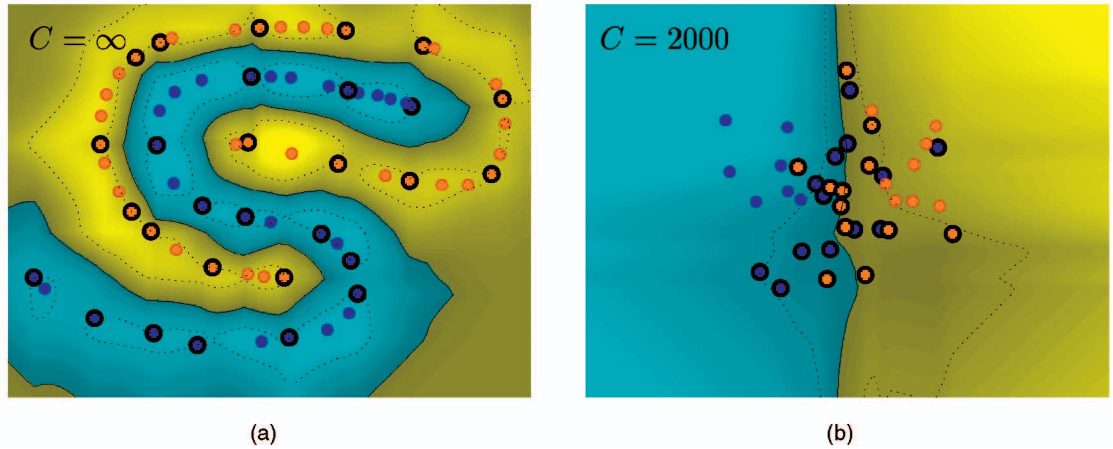


Fig. 4. Nonlinear non-cpd SVM classification examples in $\mathbb{R}^2$ with kernel $k(\mathbf{x},\mathbf{x}') = e^{-\gamma\|\mathbf{x}-\mathbf{x}'\|_1^2}$. (a) Separable problem and (b) nonseparable problem.

Proposition 2, the common interpretation of the coefficients and slack variables also holds in the non-cpd case. In particular, the ratio of upper bounded coefficients is an upper bound on the training error. So, low value of this quantity is an indicator of a suitable (but possibly overfitted) SVM solution.

Sign of $\mathbf{w}^T\mathbf{M}\mathbf{w}$. We have seen that a crucial property of a suitable solution is that $\mathbf{M}\mathbf{w}$ and $\mathbf{w}$ of the CH classifier point to identical sides of the decision line, which requires $\mathbf{w}^T\mathbf{M}\mathbf{w} > 0$. This quantity can also be calculated without explicit mapping to $\mathbb{R}^{(p,q)}$ by using the representation of $\mathbf{w}$ as linear combination of embedded training points, Lemma 6 in Section 1.2 and Proposition 3, Case 1:

$$\mathbf{w}^T\mathbf{M}\mathbf{w} = \sum_{i,j} \bar{\alpha}_i\bar{\alpha}_j y_i y_j \Phi(x_i)^T \mathbf{M}\Phi(x_j) = \sum_{i,j} \bar{\alpha}_i\bar{\alpha}_j y_i y_j k(x_i,x_j)$$
$$= \lambda \sum_{i,j} \alpha_i\alpha_j y_i y_j k(x_i,x_j)$$

with the positive scaling factor $\lambda = \left(2/\sum_i \alpha_i\right)^2$. If $\mathbf{w}^T\mathbf{M}\mathbf{w}$ is nonpositive, the SVM performs as a counterintuitive CH classifier. Reduction of C can then result in a solution with positive squared norm, if the squared distance of class means is sufficiently large.

In the case of having found an unsuitable kernel or SVM, it is still possible to apply it. Although the interpretation as a CH classifier is clearly counterintuitive, the obtained solution still correctly classifies all unbounded training examples, and might be a sparse solution, see Fig. 2b. If a non-cpd kernel function does not produce usable solutions for any choice of kernel parameters, modification of the kernel, like regularization or other approaches have to be applied.

7.5 Experiments

We now present some toy and real-world experiments to demonstrate the usability of the gained insights and criteria. We start with nonlinear classification examples of non-cpd SVMs in Fig. 4. In order to keep it illustrative, we decide for data within the unit square of $\mathcal{X} = \mathbb{R}^2$. We want to model the situation, where some simple problem specific knowledge is incorporated into a kernel. Let this toy a priori knowledge be that the L^1-distance is more suitable than the L^2-distance, which means that the diamond-shaped L^1-spheres are assumed to be more adequate than the circular L^2-spheres. Then, the kernel $k(\mathbf{x},\mathbf{x}') = e^{-\gamma\|\mathbf{x}-\mathbf{x}'\|_1^2}$ seems to be an appropriate choice. Note that an equally well separation can be obtained by the standard Gaussian in these cases. Fig. 4a demonstrates a separable problem of 68 points, where $\gamma = 50$ produces a centered kernel matrix with 52 positive, 1 zero, and 15 negative eigenvalues. So, the kernel matrix is clearly non-cpd. The squared distance of class means in the corresponding pE space of dimension 67 is positive, which guarantees the existence of an SVM solution with positive squared norm. Indeed, the SVM with $C = \infty$ yields such a solution with good separation results. A sparse solution of 29 SVs is produced. None of them are bounded, so no misclassifications can occur. Fig. 4b demonstrates a nonseparable problem consisting of 40 points drawn from two normal distributions with identical isotropic covariance matrices and means slightly differing along the horizontal. So, the Bayes-optimal decision would be the vertical line passing through the mean of the point distribution.With $\gamma = 0.005$, the centered kernel matrix has 18 positive, 1 zero, and 21 negative eigenvalues. The squared distance of class means is positive in the embedding pE space of dimension 39, which again guarantees the existence of a positive squared norm solution. For $C = \infty$ this problem diverges, but for $C = 2,000$ the presented solution with positive squared norm is obtained. Twenty-five vectors end up as SVs, many of them are bounded, as they must be wrongly classified. All non-SVs are classified correctly as expected.

Finally, we present some results demonstrating the pE characteristics of real-world problems and the applicability of non-cpd SVMs. We refined results from [17], where further experiments can be found. The data set *proteins* (226 samples, 4 classes) is frequently used in the literature, cf. [18], [28], and consists of pairwise evolutionary distances between amino acid sequences of proteins. The data set *cat-cortex* (65 samples, four classes) is based on a matrix $\mathbf{S}$ of connectivity strengths between four cortical areas of a cat. Other experiments with this data have also been presented in [18], [28]. Here, we symmetrized the similarity matrix and produced a zero-diagonal distance matrix by $\mathbf{D} := 4 \cdot \mathbf{1}_{65}\mathbf{1}_{65}^T - \frac{1}{2}(\mathbf{S}+\mathbf{S}^T)$. For both problems, a kernel matrix $\mathbf{K}$ was constructed from these distances by $K_{ij} := e^{-\gamma D_{ij}^2}$. For both data sets, we used four different binary labelings corresponding to one-versus-rest problems. In addition to non-cpd SVMs, we performed experiments with two regularization methods which make the kernel matrix positive definite as in [18], [19]. These

TABLE 1
Comparison of Classification Results on Real-World Data Sets and SVM Model Details

	LOO-errors [%]				SVM model details				
dataset	SVM	CNE	RNE	k-nn (k)	(p,q)	DCM	$\mathbf{w}^T\mathbf{M}\mathbf{w}$	#SVs	#bSVs
proteins-H-α	0.89	0.89	0.89	1.33 (1)	(218,4)	0.0074	0.0021	49	0
proteins-H-β	2.65	2.65	2.65	3.54 (1)	(219,4)	0.2178	0.0113	70	0
proteins-M	0.00	0.00	0.00	0.00 (1)	(218,4)	0.0149	0.0076	44	0
proteins-GH	0.00	0.00	0.00	1.77 (1)	(218,4)	0.0080	0.0040	44	0
cat-cortex-V	0.00	1.54	3.08	3.08 (1)	(51,13)	0.2974	0.1112	30	4
cat-cortex-A	4.62	4.62	6.15	6.15 (1)	(41,23)	0.0089	0.0003	17	7
cat-cortex-S	3.08	3.08	3.08	3.08 (3)	(41,23)	0.0401	0.0181	31	23
cat-cortex-F	4.62	4.62	4.62	3.08 (2)	(51,13)	0.2951	0.2000	44	17

methods have in common that the non-Euclidean geometry of the embedding space is replaced with a Euclidean one. Both methods center the kernel matrix and perform an eigenvalue decomposition. One approach **C**uts off contributions corresponding to **N**egative **E**igenvalues (CNE) and the second **R**eflects these **N**egative **E**igenvalue contributions (RNE). Additionally, the best k-nearest-neighbor (k-nn) classifier was chosen for comparisons. We computed the leave-one-out (LOO) error of the classifiers, while logarithmically varying the parameters C, γ in a suitable grid. We report the best LOO-error and optimal k for all data sets in the left part of Table 1. Note that classification errors of 0.44 percent and 1.54 percent correspond to one misclassified sample. So, the absolute number of errors varies only by up to four samples per data set, therefore only limited conclusions should be drawn. Still, the results indicate that the non-cpd SVM may compete with or outperform the k-nn classifier. Experiments in [17] with larger data sets support this conclusion consistently. For the proteins data, the pE embedding space turns out to be only slightly indefinite. This explains the identical error rates of the non-cpd and regularized SVMs. But, even in stronger indefinite spaces, as for the cat-cortex data, regularization can not guarantee improvements. Additionally, there are some conceptional drawbacks of these matrix-modification methods. First, the solution depends on the specific feature space embedding. For a given non-cpd kernel arbitrary many embeddings are possible, which yield different solutions. Moreover, explicit operating in the embedding feature space is required, which impairs the complexity advantages of applying the kernel trick. When new objects are to be classified, the same regularization operations have to be performed for these testing examples [19]. This means that the embedding of all training data has to be retained and involved during classification, which nullifies the computational advantages of the "sparsity" of the resulting SVM. In contrast, the solution of the non-cpd SVM does not depend on the specific embedding as it avoids explicit operations in the feature space and maintains the sparsity of the solutions.

In the right part of the table, we list the pE characteristics of the non-cpd SVM model trained with the optimal parameters. The signature of the implicitly defined pE space $\mathbb{R}^{(p,q)}$ was computed and the squared **D**istance of the **C**lass **M**eans (DCM) in this space was determined. In addition to these pretraining indicators, we list the post-training quantities $\mathbf{w}^T\mathbf{M}\mathbf{w}$, the number of support vectors (#SVs) and the number of bounded support vectors (#bSVs), where $\mathbf{w}$ is the CH solution obtained from the SVM by scaling. We see that the SVMs are non-cpd, as the signature of the embedding pE spaces have several nonzero imaginary dimensions. For the proteins, however, the negative eigenvalues of the kernel matrices are two orders of magnitude smaller than the positive ones, so the embeddings are only slightly indefinite. In contrast, the cat-cortex data is strongly indefinite, the negative eigenvalues are at the same order of magnitude as the positive ones.

For all data sets, the squared distance of the class means DCM in the pE space is positive which raises hope of separation with positive squared norm. Indeed, the resulting norms are positive in all cases and bounded from above by the DCM. The sparsity of the solutions can be seen in the number of SVs. For the proteins, the SVMs have zero training error, as can be concluded from the number of bounded SVs.

8 Conclusion

We have shown that using SVMs with arbitrary symmetric kernels, in particular, non-cpd kernels, is not only a heuristic procedure, but has a reasonable interpretation as optimal hyperplane classifiers in pE spaces. They are minimum distance classifiers with respect to certain points from the convex hulls of embedded training points. This interpretation already existed in the Euclidean case, we have extended it to the pE case. We have explained that non-cpd SVMs, in general, *cannot* be seen as margin maximizers, although a notion of margin can be defined.

The interpretation results in a constructive method to illustrate the classification behavior of an SVM in the corresponding pE space. We have demonstrated how the geometric understanding can serve as a basis for further theoretical analysis. To exemplify this, we have commented on the uniqueness of the local solutions in certain situations. We further have given practically relevant implications, mainly criteria for checking whether a given non-cpd SVM classifier is promising. An important requirement is a positive squared distance of the class means and a positive squared norm of the resulting normal vector $\mathbf{w}$. For some kernels, their unsuitability can be decided without SVM training or even data-independently. We have demonstrated the applicability of these indicators on toy and real-world data. Although being limited, the results on these small sets are encouraging and further experimental investigation should be performed.

With this work, we establish a sound basis for practical application of non-cpd kernels in SVMs. In particular, for cases where no standard kernels can be applied, indefinite SVMs seem attractive, as in the case of distance based learning.

The pE spaces provide a representation of data which only depends on a given arbitrary symmetric kernel function. It is independent of the specific algorithm to be applied to this data. We therefore expect that these spaces provide the suitable geometry for investigating other kernel-methods which involve non-cpd functions.

Acknowledgments

The author would like to thank H. Burkhardt for providing the opportunity for this work, B. Schölkopf for the kind hospitality during the visit at the MPI Tübingen and U. von Luxburg, M. Hein, O. Bousquet, N. Canterakis, O. Ronneberger, and the anonymous reviewers for valuable comments.

References

[1] D. Haussler, “Convolution Kernels on Discrete Structures,” Technical Report UCS-CRL-99-10, Univ. of California, Santa Cruz, 1999.

[2] H. Lodhi et al., “Text Classification Using String Kernels,” *J. Machine Learning Research,* vol. 2, pp. 419-444, 2002.

[3] C. Cortes, P. Haffner, and M. Mohri, “Rational Kernels,” *Proc. Advances in Neural Information Processing Systems,* vol. 15, 2003.

[4] N. Cristianini and J. Shawe-Taylor, *An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods.* Cambridge, U.K.: Cambridge Univ. Press, 2000.

[5] B. Schölkopf and A.J. Smola, *Learning with Kernels.* Cambridge, Mass.: MIT Press, 2002.

[6] O. Chapelle and V. Vapnik, “Model Selection for Support Vector Machines,” *Proc. Advances in Neural Information Processing Systems,* pp. 230-236, 2000.

[7] V. Vapnik, *The Nature of Statistical Learning Theory.* New York: Springer, 1995.

[8] C. Bahlmann, B. Haasdonk, and H. Burkhardt, “On-Line Handwriting Recognition with Support Vector Machines—A Kernel Approach,” *Proc. Eighth Int'l Workshop Frontiers in Handwriting Recognition,* pp. 49-54, 2002.

[9] D. DeCoste and B. Schölkopf, “Training Invariant Support Vector Machines,” *Machine Learning,* vol. 46, no. 1, pp. 161-190, 2002.

[10] B. Haasdonk and D. Keysers, “Tangent Distance Kernels for Support Vector Machines,” *Proc. 16th Int'l Conf. Pattern Recognition,* pp. 864-868, 2002.

[11] P.J. Moreno, P. Ho, and N. Vasconcelos, “A Kullback-Leibler Divergence Based Kernel for SVM Classification in Multimedia Applications,” *Proc. Advances in Neural Information Processing Systems,* vol. 16, pp. 1385-1392, 2004.

[12] H. Shimodaira et al., “Dynamic Time-Alignment Kernel in Support Vector Machine,” *Proc. Advances in Neural Information Processing Systems,* vol. 14, pp. 921-928, 2002.

[13] B. Schölkopf, “The Kernel Trick for Distances,” Technical Report MSR 2000-51, Microsoft Research, Redmond, Wash., 2000.

[14] 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, Nat'l Taiwan Univ., Mar. 2003.

[15] C.-C. Chang and C.-J. Lin, “LIBSVM: A Library for Support Vector Machines,” http://www.csie.ntu.edu.tw/~cjlin/libsvm, 2001.

[16] M. Sellathurai and S. Haykin, “The Separability Theory of Hyperbolic Tangent Kernels and Support Vector Machines for Pattern Classification,” *Proc. IEEE Int'l Conf. Acoustics, Speech, and Signal Processing,* pp. 1021-1024, 1999.

[17] B. Haasdonk and C. Bahlmann, “Learning with Distance Substitution Kernels,” *Proc. 26th DAGM Symp.,* pp. 220-227, 2004.

[18] T. Graepel et al., “Classification on Pairwise Proximity Data,” *Proc. Advances in Neural Information Processing Systems,* vol. 11, pp. 438-444, 1999.

[19] E. Pekalska, P. Paclik, and R. Duin, “A Generalized Kernel Approach to Dissimilarity Based Classification,” *J. Machine Learning Research,* vol. 2, pp. 175-211, 2001.

[20] L. Goldfarb, “A New Approach to Pattern Recognition,” *Progress in Pattern Recognition 2,* pp. 241-402, 1985.

[21] X. Mary, “Hilbertian Subspaces, Subdualities and Applications,” PhD dissertation, INSA Rouen, 2003.

[22] K.P. Bennett and E.J. Bredensteiner, “Duality and Geometry in SVM Classifiers,” *Proc. 17th Int'l Conf. Machine Learning,* pp. 57-64, 2000.

[23] D.J. Crisp and C.J.C. Burges, “A Geometric Interpretation of nu-SVM Classifiers,” *Proc. Advances in Neural Information Processing Systems,* vol. 12, pp. 223-229, 2000.

[24] B. Schölkopf et al., “New Support Vector Algorithms,” *Neural Computation,* vol. 12 pp. 1083-1121, 2000.

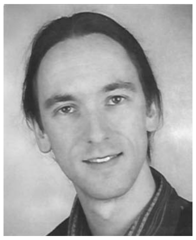
[25] C.-C. Chang and C.-J. Lin, “Training ν-Support Vector Classifiers: Theory and Algorithms,” *Neural Computation,* vol. 13, no. 9, pp. 2119-2147, 2001.

[26] P.M. Pardalos and J.B. Rosen, *Constrained Global Optimization: Algorithms and Applications.* Berlin: Springer, 1987.

[27] O. Ronneberger and F. Pigorsch, “LIBSVMTL: A Support Vector Machine Template Library,” http://lmb.informatik.uni-freiburg.de/lmbsoft/libsvmtl/, 2004.

[28] T. Graepel et al., “Classification on Proximity Data with LP-Machines,” *Proc. Ninth Int'l Conf. Artificial Neural Networks,* pp. 304-309, 1999.

[29] M. Hein and O. Bousquet, “Maximal Margin Classification for Metric Spaces,” *Proc. 16th Ann. Conf. Computational Learning Theory,* pp. 72-86, 2003.



Bernard Haasdonk received the BS and MS degrees in mathematics with distinction from the University of Freiburg, Germany, in 1997 and 2000, respectively. In 2000, he joined the Computer Science Department of the University of Freiburg as a research associate, where he is currently pursuing his PhD studies. His interests include pattern recognition and machine learning, in particular, support vector machines and general kernel methods.