Pattern Recognition Letters 32 (2011) 1511–1515

Contents lists available at ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec

Learning general Gaussian kernel hyperparameters of SVMs using optimization on symmetric positive-definite matrices manifold

Hicham Laanaya [a,b,*], Fahed Abdallah [a], Hichem Snoussi [c], Cédric Richard [c]

[a] *Centre de Recherche de Royallieu, Lab. Heudiasyc, UMR CNRS 6599, BP 20529, 60205 Compiègne, France*
[b] *Faculté des Sciences Rabat, Université Mohammed V-Agdal, 4 Avenue Ibn Battouta, B.P. 1014 RP, Rabat, Morocco*
[c] *Institut Charles Delaunay (FRE CNRS 2848), Université de Technologie de Troyes, 10010 Troyes, France*

ARTICLE INFO

Article history:
Received 21 July 2010
Available online 24 May 2011
Communicated by Y. Ma

Keywords:
Kernel optimization
Support vector machines
General Gaussian kernel
Symmetric positive-definite matrices manifold

ABSTRACT

We propose a new method for general Gaussian kernel hyperparameter optimization for support vector machines classification. The hyperparameters are constrained to lie on a differentiable manifold. The proposed optimization technique is based on a gradient-like descent algorithm adapted to the geometrical structure of the manifold of symmetric positive-definite matrices. We compare the performance of our approach with the classical support vector machine for classification and with other methods of the state of the art on toy data and on real world data sets.

1. Introduction

Support Vector Machine (SVM) is a promising pattern classification technique proposed by Vapnik (1995). Unlike traditional methods which minimize the empirical training error, SVM aims at minimizing an upper bound of the generalization error through maximizing the margin between the separating hyperplane and the data. This can be regarded as an approximate implementation of the Structure Risk Minimization principle. What makes SVM attractive is the property of condensing information in the training data and providing a sparse representation by using a very small number of data points called support vectors (SVs) (Girosi, 1998).

The key features of SVMs are the use of kernels, the absence of local minima, the sparseness of the solution and the capacity control obtained by optimizing the margin (Cristianini and Shawe-Taylor, 2000). Nevertheless an SVM based method is unable to give accurate results in high dimensional spaces when more than one dimension are noisy (Grandvalet and Canu, 2002; Weston et al., 2000). Another limitation of the support vector approach lies in the choice of the kernel and its eventual hyperparameter. Hyperparameter selection is in fact crucial to enhance the performance of an SVM classifier. Different works were introduced to deal with this problem for different aims; Gold and Sollich (2003), Grandvalet and Canu (2002), Lanckriet et al. (2004) and Weston et al. (2000) introduced methods for feature selection problem using a Gaussian kernel and Chen and Ye (2008), Lanckriet et al. (2004) and Luss and d'Aspremont (2008) learn directly the optimal kernel matrix, also called Gram matrix, from the training data using semidefinite programming or using an initial guess (similarity matrix) of the kernel. These methods use similar optimization problem and give the solution based on gradient descent approaches. Note that authors in (Lanckriet et al., 2004; Luss and d'Aspremont, 2008) estimate simultaneously the kernel matrix for training and test examples and the kernel function expression is not determined. Well, learning directly the kernel matrix is technically consuming as we have to learn and store $n \times (n+1)/2$ parameters, where n is the number of examples in the database. Furthermore, estimating the kernel matrix on the given data set will not be directly usable to classify unseen examples.

In a different manner, and for the same classification problem, methods proposed for feature selection learn the Gaussian kernel hyperparameter as a diagonal matrix Q of dimension $d \times d$ where d is the number of features, and do not take into account the eventual relationship between features (as in feature extraction problems).

We propose here a new method for hyperparameters learning for general Gaussian kernel of the form:

$$k_Q(\boldsymbol{x},\boldsymbol{y}) = \exp\left(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{y})^T Q(\boldsymbol{x}-\boldsymbol{y})\right), \tag{1}$$

where $\boldsymbol{x},\boldsymbol{y} \in \mathbb{R}^d$, and Q is a $d \times d$ symmetric positive-definite matrix to be adjusted in order to answer adequately a specified criterion,

[*] Corresponding author at: Faculté des Sciences Rabat, Université Mohammed V-Agdal, 4 Avenue Ibn Battouta, B.P. 1014 RP, Rabat, Morocco. Tel.: +212 6 64 73 18 00.
E-mail addresses: Hicham.Laanaya@hds.utc.fr, hicham.laanaya@gmail.com (H. Laanaya), Fahed.Abdallah@hds.utc.fr (F. Abdallah), Hichem.Snoussi@utt.fr (H. Snoussi), cedric.richard@unice.fr (C. Richard).

0167-8655/$ - see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.patrec.2011.05.009

namely here margin maximization used by the well-known SVM method.

Note that SVM-based classification methods of the state of the art restrict Q to the identity matrix multiplied by a positive real $\sigma^{-2}(Q = \sigma^{-2}\mathbb{I}$; where $\mathbb{I} \in \mathbb{R}^{d \times d}$ is the identity matrix) or to a positive-definite diagonal matrix and use gradient-based approaches for optimization (Grandvalet and Canu, 2002; Lanckriet et al., 2004; Weston et al., 2000). The assumption of positive-definite diagonal matrix Q means that we are performing a feature selection scheme simultaneously with the optimization algorithm. The method proposed in this paper use a full symmetric positive definite matrix Q and constitutes a general alternative for the usual Gaussian kernel where we have only one parameter σ to estimate. It generalizes also the assumption of positive-definite diagonal matrix by being able to capture feature correlation by the non-diagonal elements of the matrix Q. The method presented in (Glasmachers and Igel, 2005), dealing with the same subject, has been applied on a bound of the generalization error, which is the radius margin quotient. The relevance of this method was proven on a simple 2d example where the best results were achieved by constraining the optimization to a constant trace subspace in order to control the size of the kernel. In contrast, our method is working on an exact margin criterion and an explicit expression of the gradient of this criterion is given in the paper. The kernel size variation is controlled using a regularization term based on the Frobenius norm. In (Friedrichs and Igel, 2005), the authors proposed an approach based on genetic algorithms optimization. As we know, this kind of methods is time consuming as we have to evaluate a fitness function for each member of the population of the genetic algorithm and then apply mutation, crossover and selection to get the best individuals.

The article is organized as follows. In Section 2.1, we give a brief introduction of optimization on the manifold of symmetric positive-definite matrices. In Section 2.2 we recall the principle of SVM. We then introduce, in Section 2.3, our new approach for general Gaussian kernel hyperparameters optimization. Finally, Section 3 describes results obtained on toy and real world data indicating the performance of our approach.

2. General Gaussian kernel optimization

The aim of this work is to optimize the general Gaussian kernel parameter Q (*cf.* Eq. (1)) under the maximum margin criterion using gradient-based approach in the manifold of symmetric positive-definite matrices. We first begin with a brief overview of the optimization on the manifold of positive-definite symmetric matrices.

2.1. Optimization on the manifold of symmetric positive-definite matrices

Let $\mathcal{S}_d^+$ be the set of all symmetric positive definite matrices of dimension d:

$$\mathcal{S}_d^+ = \left\{ Q \in \mathbb{R}^{d \times d}; Q^T = Q, \boldsymbol{x}^T Q \boldsymbol{x} > 0, \forall \boldsymbol{x} \in \mathbb{R}_*^d \right\}. \tag{2}$$

We consider the minimization of a function $f : \mathcal{S}_d^+ \mapsto \mathbb{R}$ over $\mathcal{S}_d^+$. Classical optimization approaches like gradient descent or Newton algorithm can be extended to deal with optimization on the Riemannian manifold $\mathcal{S}_d^+$ (Absil et al., 2008) by considering the generic update classically used in optimization methods:

$$Q_{p+1} = Q_p + \eta_p S_p, \tag{3}$$

where Q_p is a member of $\mathcal{S}_d^+$, η_p is the step size and S_p is the adaptation rule.

In a geometric approach, S_p could be taken as the tangent vector to the space $\mathcal{S}_d^+$ (Amari and Nagaoka, 2000; Boothby, 1975), and the addition operation can be implemented *via* the exponential map (Absil et al., 2008). This results in a new generic iteration of the form

$$Q_{p+1} = \mathcal{E}_{Q_p}(\eta_p S_p), \tag{4}$$

where $\mathcal{E}_Q$ maps the tangent space $\mathcal{TS}_d^+$ (set of symmetric matrices) to the Riemannian manifold $\mathcal{S}_d^+$. It is given by $\mathcal{E}_Q(T) = Q^{1/2} \exp(Q^{-1/2} T Q^{-1/2}) Q^{1/2}$, where T is a symmetric matrix and,

$$\exp(T) = \sum_{k=0}^{\infty} \frac{T^k}{k!}. \tag{5}$$

For gradient-descent algorithm, S_p is given by the opposite of the gradient of $f(Q_p)$ and noted by $-\mathrm{grad}f(Q_p)$. Given the explicit analytic expression of the gradient $-\mathrm{grad}f(Q_p)$, the generating mechanism of the next step is:

$$Q_{p+1} = \mathcal{E}_{Q_p}(\eta_p S_p), \tag{6}$$

$$= Q_p^{1/2} \exp\left(-\eta_p Q_p^{-1/2}\ \mathrm{grad}\ f(Q_p) Q_p^{-1/2}\right) Q_p^{1/2}. \tag{7}$$

2.2. Support vector machines

The support vector machines approach, initiated by Vapnik (1998), is initially developed for binary classification problems. It classifies patterns from two classes (+1 and −1) by searching the optimal hyperplane which has a maximum margin between the nearest positive and negative examples. A significant advantage of SVMs is that the solution is global and unique.

Considering a training set $A_n = \{\boldsymbol{x}_i, i = 1, \ldots, n\}$ where $\boldsymbol{x}_i \in \mathbb{R}^d$ is an example with associated class $y_i \in \{-1, +1\}$.

The SVM optimization problem can be formulated using matrix notations

$$\max_{\boldsymbol{\alpha}} 2\boldsymbol{\alpha}^T \boldsymbol{e} - \mathrm{Tr}(K(Y\boldsymbol{\alpha})(Y\boldsymbol{\alpha})^T), \tag{8}$$

under the constraints

$$\sum_{i=1}^{n} \alpha_i y_i = 0, \tag{9}$$

$$0 \leqslant \alpha_i \leqslant C, \quad i = 1, \ldots, n. \tag{10}$$

where $\boldsymbol{\alpha} = (\alpha_i)_{i=1,\ldots,n}$, $Y = \mathrm{diag}(\boldsymbol{y})$, $\boldsymbol{y} = (y_i)_{i=1,\ldots,n}$,$\boldsymbol{e}$ is the n-vector of ones, $K_{ij} = k(\boldsymbol{x}_i, \boldsymbol{x}_j)$ is the Gram matrix, and C is a constant chosen by the user. Note that a high value of C corresponds to a great penalty to errors in the case of linearly non-separable data.

In this case, the classification of a new pattern $\boldsymbol{x}$ is given by the decision function:

$$f(\boldsymbol{x}) = \mathrm{sign}\left(\sum_{i \in SV} y_i \alpha_i k(\boldsymbol{x}, \boldsymbol{x}_i) + b \right). \tag{11}$$

The Gaussian kernel $k(\boldsymbol{x}, \boldsymbol{x}_i) = e^{-\|\boldsymbol{x} - \boldsymbol{x}_i\|^2/\sigma^2}$, $\sigma \in \mathbb{R}^+$, is one of the most popular and powerful kernel used in pattern recognition and classification methods, and in particular for SVM techniques. Unlike some conventional statistical approaches (for example neural network methods where each feature is multiplied by a synaptic weight), the classical SVM approach with Gaussian Kernel of parameter σ does not attempt to control model complexity by keeping the number of features small and all features are scaled, and thus weighted, according to the same parameter σ. This choice seems to be poor and not adequate for general classification problems where some features are only about noise, or when there are some features providing somewhat more pertinent information for the classification problem, or even when there are correlation between features.

Works of the state of the art answer partially these problems by considering the general Gaussian kernel of Eq. (1) with a diagonal matrix Q (Gold and Sollich, 2003; Grandvalet and Canu, 2002; Weston et al., 2000), and thus performing the feature selection scheme under the SVM criterion. Work introduced in (Glasmachers and Igel, 2005) consider a full matrix Q using optimization of radius-margin generalization performance measures for SVMs on the manifold of positive definite symmetric matrices by restricting the optimization to a constant trace subspace in order to control the size of the kernel. We consider in this work a more general solution by optimizing the maximum margin criterion augmented by an adapted regularized term. Experiments on toy and real-world data show that our strategy leads to a significant improvement of classification results as compared to existing classification methods.

2.3. Kernel hyperparameter optimization under the SVM framework

We formulate the hyperparameter kernel learning problem as in (Lanckriet et al., 2004; Luss and d'Aspremont, 2008), where the authors minimize a modified form of Eq. (8),

$$w_{C,\rho}(Q) = \max_{\{0\leqslant\boldsymbol{\alpha}\leqslant C,\boldsymbol{\alpha}^T\boldsymbol{y}=0\}} 2\boldsymbol{\alpha}^T\boldsymbol{e} - \mathrm{Tr}(K^Q(Y\boldsymbol{\alpha})(Y\boldsymbol{\alpha})^T) + \rho\|K^Q - K'\|_F^2 \tag{12}$$

with $Y = \mathrm{diag}(\boldsymbol{y})$, $K_{ij}^Q = \exp(-(\boldsymbol{x}_i - \boldsymbol{x}_j)^T Q(\boldsymbol{x}_i - \boldsymbol{x}_j)/2)$, where $Q \in \mathcal{S}_d^+$ and $\|K\|_F = \sqrt{\mathrm{trace}(KK^T)}$ is the Frobenius norm. The term $\rho\|K^Q - K'\|_F^2$ is a regularization term used to constrain the solution using an eventually indefinite kernel matrix K' (Chen and Ye, 2008; Luss and d'Aspremont, 2008), *e.g.*, a similarity matrix or a guess of the best kernel calculated over training database. Setting $\rho = 0$ leads to the optimization problem of classical SVM classification.

2.3.1. Gradient calculation

If $\boldsymbol{\alpha} = (\alpha_i)_{i=1,\ldots,n}$ is solution of the following maximization problem,

$$\max_{\{0\leqslant\boldsymbol{\alpha}\leqslant C,\boldsymbol{\alpha}^T y=0\}} 2\boldsymbol{\alpha}^T\boldsymbol{e} - \mathrm{Tr}(K^Q(Y\boldsymbol{\alpha})(Y\boldsymbol{\alpha})^T) + \rho\|K^Q - K'\|_F^2, \tag{13}$$

we search for Q that minimizes $w_{C,\rho}(Q)$. Eq. (13) is convex with respect to each hyperparameter $Q_{kl}(k,l \in \{1,\ldots,d\})$ of the general Gaussian kernel hyperparameter Q. In fact, $\|K^Q - K'\|_F^2$ is convex (composition of the convex function $\|\cdot\|_F$ and exponential functions), and $-\mathrm{Tr}(K^Q(Y\boldsymbol{\alpha})(Y\boldsymbol{\alpha})^T)$ is a linear combination of composed convex functions (affine and exponential). Thus, the minimum of Eq. (13) exists and is unique. We calculate the gradient of $w_{C,\rho}(Q)$ that we will use for the update step of the gradient descent method applied in the manifold $\mathcal{S}_d^+$. The gradient of $w_{C,\rho}(Q)$ is given by

$$\mathrm{grad}w_{C,\rho} = \left(\frac{\partial w_{C,\rho}(Q)}{\partial Q_{kl}}\right)_{k,l=1,\ldots d}. \tag{14}$$

Let $R^Q = K^Q(Y\boldsymbol{\alpha})(Y\boldsymbol{\alpha})^T$ and $r^Q = \mathrm{Tr}(R^Q)$. We have then

$$\begin{aligned} R_{ii}^Q &= -\sum_{j=1}^n y_i y_j \alpha_i \alpha_j \exp(-(\boldsymbol{x}_i - \boldsymbol{x}_j)^T Q(\boldsymbol{x}_i - \boldsymbol{x}_j)/2), \\ &= -\sum_{j=1}^n y_i y_j \alpha_i \alpha_j \exp\left(-\frac{1}{2}\sum_{k=1}^d\sum_{l=1}^d (x_{ik} - x_{jk})(x_{il} - x_{jl})Q_{kl}\right) \end{aligned} \tag{15}$$

and

$$\begin{aligned} r^Q &= \sum_{i=1}^n R_{ii}^Q, \\ &= -\sum_{i=1}^n\sum_{j=1}^n y_i y_j \alpha_i \alpha_j \exp\left(-\frac{1}{2}\sum_{k=1}^d\sum_{l=1}^d (x_{ik} - x_{jk})(x_{il} - x_{jl})Q_{kl}\right), \\ &= -\sum_{i=1}^n\sum_{j=1}^n y_i y_j \alpha_i \alpha_j \prod_{k=1}^d\prod_{l=1}^d \exp\left(-\frac{1}{2}(x_{ik} - x_{jk})(x_{il} - x_{jl})Q_{kl}\right). \end{aligned} \tag{16}$$

The derivative of r^Q is thus given by

$$\begin{aligned} \frac{\partial r^Q}{\partial Q_{k'l'}} &= \sum_{i=1}^n\sum_{j=1}^n \frac{1}{2} y_i y_j \alpha_i \alpha_j (x_{ik'} - x_{jk'})(x_{il'} - x_{jl'}) \\ &\quad\times \prod_{k=1}^d\prod_{l=1}^d \exp\left(-\frac{1}{2}(x_{ik} - x_{jk})(x_{il} - x_{jl})Q_{kl}\right), \\ &= \sum_{i=1}^n\sum_{j=1}^n \frac{1}{2} y_i y_j \alpha_i \alpha_j (x_{ik'} - x_{jk'})(x_{il'} - x_{jl'}) K_{ij}^Q. \end{aligned} \tag{17}$$

Let $S^Q = (K^Q - K')^2$ and $s^Q = \rho\|K^Q - K'\| = \rho\mathrm{Tr}(S^Q)$. We have

$$S_{ii}^Q = \sum_{j=1}^n (K_{ij}^Q - K'_{ij})^2 \tag{18}$$

and

$$\begin{aligned} s^Q &= \rho\sum_{i=1}^n\sum_{j=1}^n (K_{ij}^Q - K'_{ij})^2, \\ &= \rho\sum_{i=1}^n\sum_{j=1}^n \left(\exp\left(-\frac{1}{2}\sum_{k=1}^d\sum_{l=1}^d (x_{ik} - x_{jk})(x_{il} - x_{jl})Q_{kl}\right) - K'_{ij}\right)^2, \\ &= \rho\sum_{i=1}^n\sum_{j=1}^n \left(\prod_{k=1}^d\prod_{l=1}^d \exp\left(-\frac{1}{2}(x_{ik} - x_{jk})(x_{il} - x_{jl})Q_{kl}\right) - K'_{ij}\right)^2. \end{aligned} \tag{19}$$

The derivative of s_Q is given by

$$\begin{aligned} \frac{\partial s_Q}{\partial Q_{k'l'}} &= -\rho\sum_{i=1}^n\sum_{j=1}^n (x_{ik'} - x_{jk'})(x_{il'} - x_{jl'}) \\ &\quad\times \prod_{k=1}^d\prod_{l=1}^d \exp\left(-\frac{1}{2}(x_{ik} - x_{jk})(x_{il} - x_{jl})Q_{kl}\right) \\ &\quad\left(\prod_{k=1}^d\prod_{l=1}^d \exp\left(-\frac{1}{2}(x_{ik} - x_{jk})(x_{il} - x_{jl})Q_{kl}\right) - K'_{ij}\right), \\ &= -\rho\sum_{i=1}^n\sum_{j=1}^n (x_{ik'} - x_{jk'})(x_{il'} - x_{jl'}) K_{ij}^Q \left(K_{ij}^Q - K'_{ij}\right). \end{aligned} \tag{20}$$

Thus, the derivative of $w_{C,\rho}(Q) := r_Q + s_Q$ will be

$$\begin{aligned} \frac{\partial w_{C,\rho}(Q)}{\partial Q_{k'l'}} &= \sum_{i=1}^n\sum_{j=1}^n \frac{1}{2} y_i y_j \alpha_i \alpha_j (x_{ik'} - x_{jk'})(x_{il'} - x_{jl'}) K_{ij}^Q \cdots \\ &\quad\cdots - \rho\sum_{i=1}^n\sum_{j=1}^n (x_{ik'} - x_{jk'})(x_{il'} - x_{jl'}) K_{ij}^Q \left(K_{ij}^Q - K'_{ij}\right), \\ &= \sum_{i=1}^n\sum_{j=1}^n \frac{1}{2} y_i y_j \alpha_i \alpha_j (x_{ik'} - x_{jk'})(x_{il'} - x_{jl'}) K_{ij}^Q \cdots \\ &\quad\cdots - \rho(x_{ik'} - x_{jk'})(x_{il'} - x_{jl'}) K_{ij}^Q \left(K_{ij}^Q - K'_{ij}\right), \\ &= \sum_{i=1}^n\sum_{j=1}^n (x_{ik'} - x_{jk'})(x_{il'} - x_{jl'}) K_{ij}^Q \left(\frac{1}{2} y_i y_j \alpha_i \alpha_j - \rho(K_{ij}^Q - K'_{ij})\right). \end{aligned} \tag{21}$$

2.3.2. Algorithm steps

After calculating the gradient, we can now introduce the steps used for general Gaussian hyperparameters optimization:

Set $q := 0$. Given Q_0 a symmetric positive-definite matrix and K' a kernel matrix (eventually an indefinite kernel matrix). Firstly, we calculate the Lagrange multipliers $\boldsymbol{\alpha}^q$ solution of the SVM optimization problem (13) associated to the kernel matrix K^{Q_q}. Secondly, we look for Q_{q+1}, using gradient descent method on the manifold $\mathcal{S}_d^+$: Using the gradient expression of $w_{C,\rho}$ defined in Eq. (21) and the exponential mapping introduced in Section 2.1, the update of Q_{q+1} is the result of the gradient-descent optimization method applied to $w_{C,\rho}$, defined in Eq. (12) and starting from the symmetric positive-definite matrix Q_q. In other words, Q_{q+1} is the convergence result of the sequence $(Q_{p,q})_p$ defined using the gradient-descent adaptation rule

$$Q_{p+1,q} := Q_{p,q}^{1/2} \exp\left(-\eta_p Q_{p,q}^{-1/2} \text{grad } w_{C,\rho}(Q_{p,q}) Q_{p,q}^{-1/2}\right) Q_{p,q}^{1/2} \tag{22}$$

where $Q_{0,q} := Q_q$ and η_p is the step-size at the iteration p. These steps are repeated until $Q_{q+1} = Q_q$.

As any gradient-based optimization method, the speed of convergence of our approach depends on the choice of the value of the step-size η_p, however, it is somewhat unreliable: if we choose a step-size too large, than the objective function might actually get worse on some steps, if the step-size is too small, then the algorithm will take a very long time to make progress. The value of η_p can be optimized at each step p by searching the minimum of the function

$$\eta_p = \min_{\eta>0} \frac{1}{2} \text{Tr}(K^{Q_q(\eta)}(Y\boldsymbol{\alpha}^q)(Y\boldsymbol{\alpha}^q)^T) - \rho \|K^{Q_q(\eta)} - K'\|_F^2, \tag{23}$$

where

$$Q_q(\eta) = Q_{p,q}^{1/2} \exp\left(-\eta Q_{p,q}^{-1/2} \text{grad } w_{C,\rho}(Q_{p,q}) Q_{p,q}^{-1/2}\right) Q_{p,q}^{1/2}. \tag{24}$$

Algorithm 1 gives the steps of optimization of the general Gaussian kernel hyperparameters for SVM classification.

Algorithm 1: Optimization algorithm

```
1.Inputs:$Q_0$ initial value of $Q$ (cf. Eq. (1)) and $K'$ (cf. Eq. (12))
  computed over training data $\{\boldsymbol{x}_i; i = 1,\ldots,n\}$
2. $q := 0$
repeat
  1. Compute Lagrange multipliers $\boldsymbol{\alpha}^q$ using the kernel matrix
  $K^{Q_q}$ (cf. Eq. (12))
  2. $p := 0$
  3. $Q_{0,q} := Q_q$
  repeat
    1. Compute grad$w_{C,\rho}(Q_{p,q})$ using Eqs. (14) and (21)
    2. Compute $\eta_p > 0$ using Eq. (23)
    3. $Q_{p+1,q} := Q_{p,q}^{1/2} \exp\left(-\eta_p Q_{p,q}^{-1/2} \text{grad } w_{C,\rho}(Q_{p,q}) Q_{p,q}^{-1/2}\right) Q_{p,q}^{1/2}$
    4. $p := p + 1$
  until $w_{C,\rho}(Q_{p+1,q}) < w_{C,\rho}(Q_{p,q})$
  3. $Q_{q+1} := Q_{p,q}$
  4. $q := q + 1$
until $Q_q = Q_{q-1}$
```

3. Experiments

For simulated and real experiments, we used 30 partitions of each dataset separated into disjoint training and test sets. Each partition contains 200 samples: 100 for training and 100 samples for test. For each SVMs hyperparameters (C and σ) combination, five classical SVMs are built using the training sets of the first five data partitions. The hyperparameters with the best classification rate is selected and their performance is measured by using all 30 partitions. The initial value of Q is taken as a identity matrix multiplied by the best hyperparameter σ selected above.

3.1. Toy data

We compared our method with the standard SVMs, with feature selection approach proposed in (Grandvalet and Canu, 2002) and with the method of Glasmachers and Igel (2005). We used the non-linear toy data presented in (Weston et al., 2000). The number of features of the database is 52 where only the two first features are relevant. See Weston et al. (2000) for more details about the data. We search here for a diagonal matrix Q that gives the best classification rate. We used $Q_0 = \sigma\mathbb{I}$ for initialization and $K' = K^{Q_0}$. Table 1 shows classification rates for the classical SVM, adaptive scaling, Glasmachers and Igel and our approach. The new approach gives the best result with 93.36% of data which are well classified compared to 51.28% for SVM, 90.63% for adaptive scaling and 65.85% for Glasmachers and Igel approach. In our opinion, the aberrant results obtained for Glasmachers and Igel approach is due mostly to the use of noisy data (in (Glasmachers and Igel, 2005), the approach was tested on a noise-free data based on uniform 2d distribution). Also controlling the kernel size in large space-dimension with only a single parameter seems to be difficult to accomplish and the radius margin quotient can lead to undesirable solutions under these conditions.

To illustrate the stability of our approach using different values of σ for the initialization of $Q(Q_0 = \sigma\mathbb{I})$, we show in Fig. 1 the variation of the probability error of different approaches. The figure shows that the initialization of Q did not affect significantly the result of our method and that the new approach gives always the best classification rates compared to adaptive scaling or to the classical SVM.

We note that we do not fixe the number of features to be selected as done in (Weston et al., 2000). In the next section, we provide a more general application of our approach to handle correlation between features using a full matrix.

3.2. Real-world data

For the evaluation of our hyperparameter optimization method on real-world data, we used the common medical benchmark datasets *Breast-Cancer* and *Heart* with input dimension d equal to 10 and 13, respectively. Each component of the input data is normalized to zero mean and unit standard deviation.

Table 2 gives the results obtained using the ordinary SVM, adaptive scaling approach, the method of Glasmachers and Igel (2005), and our approach. We achieved significantly better results by our approach; on the first database, we get 94.75% of classification rate compared to 92.24% for the ordinary SVM and for the adaptive scaling approaches and 92.56% for the approach of Glasmachers and Igel (2005). We get also a better classification rate on the second database with 92.93% for our approach and 85.97% with the classical SVM and the adaptive scaling approaches, and 87.03% for the approach of Glasmachers and Igel (2005). Note that the results of the adaptive scaling method is similar to that of SVM be-

Table 1
Results averaged over 30 trials using a diagonal matrix Q.

Approach	SVM (%)	Adaptive scaling (%)	Glasmachers and Igel (%)	Our approach(%)
Classification rate	51.28	90.63	65.85	93.36

Table 2
Results averaged over 30 trials using a full matrix Q.

Approach	SVM (%)	Adaptive scaling (%)	Glasmachers and Igel (%)	Our approach(%)
Breast-cancer	92.24	92.24	92.56	94.75
Heart	85.97	85.97	87.03	92.93

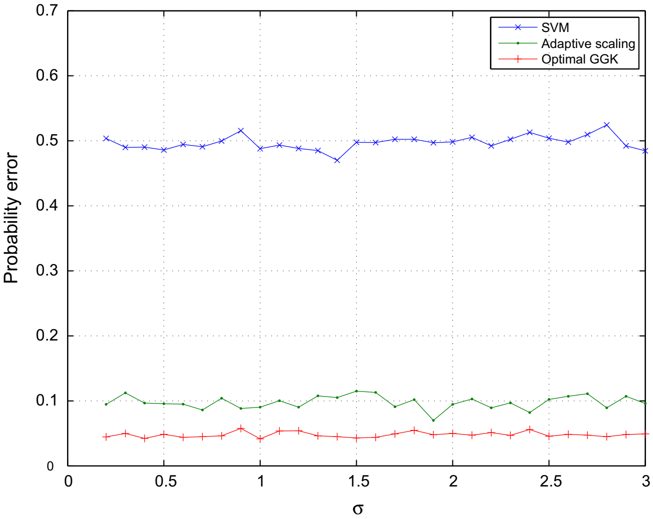

Fig. 1. Probability error for different value of the initial σ.

cause data are constructed using only relevant features. The obtained results show clearly that our method is able to capture the correlation between different features leading for this better classification results. The superiority of our method compared to the method of Glasmachers and Igel (2005) is due to the fact that our method is using an exact SVM criterion for kernel optimization on the manifold of positive-definite matrices. The regularization term $\rho\|K^Q - K'\|_F^2$ used to constrain the solution seems to be more adapted than that of the kernel size when data space dimension is high.

4. Conclusion

We proposed here a new method for SVM hyperparameters optimization in the case of general Gaussian kernel. An approach that handle diagonal and full matrix for the hyperparameter optimization under the SVM framework. This new method adapts (in the case of full matrix) the orientation of Gaussian kernels, i.e., that can detect correlations in the input data features relevant for the kernel machine method. Results on real and simulated data show the effectiveness of the proposed method.

Also the new approach can be adapted to address the problem of support vector regression using the general Gaussian kernel. Future work will address this problem. Other optimization criteria may also be used as in: kernel fisher discriminant, kernel principal component analysis and others.

References

Absil, P.-A., Mahony, R., Sepulchre, R., 2008. Optimization Algorithms on Matrix Manifolds. Princeton University Press, Princeton, NJ.

Amari, S., Nagaoka, H., 2000. Methods of Information Geometry. American Mathematical Society.

Boothby, W.M., 1975. An Introduction to Differentiable Manifolds and Riemannian Geometry/William M. Boothby. Academic Press, New York.

Chen, J., Ye, J., 2008. Training SVM with indefinite kernels. In: Cohen, W.W., McCallum, A., Roweis, S.T. (Eds.), Machine Learning, Proc. Twenty-Fifth Internat. Conf. (ICML 2008), Helsinki, Finland, June 5-9, 2008, ACM International Conference Proceeding Series, vol. 307, ACM, pp. 136–143. <http://doi.acm.org/10.1145/1390156.1390174>.

Cristianini, N., Shawe-Taylor, J., 2000. An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods. Cambridge University Press, Cambridge, UK.

Friedrichs, F., Igel, C., 2005. Evolutionary tuning of multiple SVM parameters. Neurocomputing 64, 107–117 <http://dx.doi.org/10.1016/j.neucom.2004.11.022>.

Girosi, F., 1998. An equivalence between sparse approximation and support vector machines. Neural Computat. 10 (6), 1455–1480.

Glasmachers, T., Igel, C., 2005. Gradient-based adaptation of general Gaussian kernels. Neural Computat. 17 (10), 2099–2105 <http://neco.mitpress.org/cgi/content/abstract/17/10/2099>.

Gold, C., Sollich, P., 2003. Model selection for support vector machine classification. Neurocomputing 55 (1–2), 221–249 <http://dx.doi.org/10.1016/S0925-2312(03)00375-8>.

Grandvalet, Y., Canu, S., 2002. Adaptive scaling for feature selection in SVMs. In: Becker, S., Thrun, S., Obermayer, K. (Eds.), NIPS. MIT Press, pp. 553–560 <http://books.nips.cc/papers/files/nips15/AA09.pdf>.

Lanckriet, G.R.G., Cristianini, N., Bartlett, P., Ghaoui, L.E., Jordan, M.I., 2004. Learning the kernel matrix with semidefinite programming. J. Machine Learn. Res. 5, 27–72.

Luss, R., d'Aspremont, A., 2008. Support vector machine classification with indefinite kernels. CoRR abs/0804.0188, Informal publication. <http://arxiv.org/abs/0804.0188>.

Vapnik, V., 1995. The Nature of Statistical Learning Theory. Springer Verlag, New York.

Vapnik, V.N., 1998. Statistical Learning Theory. John Wesley and Sons.

Weston, J., Mukherjee, S., Chapelle, O., Pontil, M., Poggio, T., Vapnik, V., 2000. Feature selection for SVMs. In: Leen, T.K., Dietterich, T.G., Tresp, V. (Eds.), NIPS. MIT Press, pp. 668–674.