Regularization Constants in LS-SVMs: a Fast Estimate via Convex Optimization

Kristiaan Pelckmans, Johan A.K. Suykens, Bart De Moor
K.U. Leuven, ESAT- SCD-SISTA
Kasteelpark Arenberg 10
B-3001 Leuven (Heverlee), Belgium
Tel: 32 16 32 19 07 - Fax: 32 16 32 19 70
E-mail: {kristiaan.pelckmans, johan.suykens}@esat.kuleuven.ac.be

***Abstract*— In this paper, the tuning of the regularization constant in applications of Least Squares Support Vector Machines (LS-SVMs) for regression and classification is considered. The formulation of the LS-SVM training and regularization constant tuning problem (w.r.t. the validation performance) is considered as a single constrained optimization problem. In the formulation with Tikhonov regularization the problem of estimation the weights, validation errors and the regularization constants is a non-convex problem. The main result of this paper is a conversion of the nonlinear constraints into a set of linear constraints which turns the problem into a convex one. This is done based upon a simple Nadaraya-Watson kernel estimator via approximating the LS-SVM smoother matrix by the Nadaraya-Watson smoother. The paper further illustrates how to use this initial estimate towards grid search or local search methods. Numerical examples show considerable speed-ups by the proposed method.**

I. Introduction

Regularization has a rich history which dates back to the theory of inverse ill-posed and ill-conditioned problems [1]. It has stimulated research in modeling by the introduction of regularized cost functions as e.g. in splines [2], multilayer-perceptrons [3], regularization networks [4], support vector machines (SVM) and related methods (see e.g. [5]). SVM [6] is a powerful methodology for solving problems in nonlinear classification, function estimation and density estimation which has also led to many other recent developments in kernel based learning methods in general [7], [8]. SVMs and related methods have been introduced within the context of statistical learning theory and structural risk minimization. In the methods one solves convex optimization problems, typically quadratic programs. Least Squares Support Vector Machines (LS-SVMs)[1] [9], [10] are reformulations to standard SVMs which lead to solving linear KKT systems for classification tasks as well as regression. In [10] LS-SVMs have been proposed as a class of kernel machines with primal-dual formulations in relation to kernel FDA, RR, PLS, CCA, PCA, recurrent networks and control. The dual problems for the static regression without bias term are closely related to Guassian processes, regularization networks and kriging, while LS-SVMs rather take an optimization approach with primal-dual formulations and application of the kernel trick.

[1]http://www.esat.kuleuven.ac.be/sista/lssvmlab

The relative importance between the smoothness of the solution and the norm of the residuals in the cost function involves a tuning parameter, usually called the regularization constant. The determination of regularization constants is important in order to achieve good generalization performance with the trained model and is an important problem in statistics and learning theory [5], [8], [11]. The practical relevance of this question is reflected in the amount of (computation-) time spent on this issue in applications of nonlinear models as e.g. LS-SVMs. A number of model selection criteria were introduced to tune the model (hyper-parameters) to the data.

In [12] a framework of fusion for LS-SVMs has been recently proposed with additive regularization which converts validated and cross-validated kernel based learning problems into convex programs or even linear systems. Conceptually, the levels of training and validation are treated at different levels but computationally the levels have been fused and the additive form of regularization has been exploited at this point. In this paper we consider LS-SVMs with Tikhonov regularization for classification and regression where we fuse the training and validation formulations. The resulting problem in the unknown weights, validation errors and the regularization constants is a non-convex problem. The main result of this paper is a conversion of the nonlinear constraints into a set of linear constraints which turns the problem into a convex one. This is done based upon a simple Nadaraya-Watson kernel estimator via approximating the LS-SVM smoother matrix by the Nadaraya-Watson smoother. The paper further illustrates how to use this initial estimate towards grid search or local search methods.

This paper is organized as follows. In Section II we review LS-SVMs for classification and regression. In Section III we discuss how to obtain the convex estimate for the fused problem based on the Nadaraya-Watson kernel smoother matrix estimate. In Section IV we explain how the further use this convex estimate for grid search or local search. In Section V we present illustrative examples.

II. Least Squares Support Vector Machines

Let $\{x_i, y_i\}_{i=1}^N \subset \mathbb{R}^d \times \mathbb{R}$ be such that $y_i = f(x_i) + e_i$ where $x_1, \ldots, x_N$ are deterministic points (fixed design), $f : \mathbb{R}^d \to \mathbb{R}$ is an unknown real-valued smooth function and $e_1, \ldots, e_N$

are uncorrelated random errors with $E[e_i] = 0$, $E[(e_i)^2] = \sigma_e^2 < \infty$. The n data points of the validation set are denoted as $\{x_j^v, y_j^v\}_{j=1}^n$. In the case of classification, $y, y^v \in \{-1, 1\}$.

A. Least Squares Support Vector Machine regression

The model of a LS-SVM regressor is given as $f(x) = w^T\varphi(x) + b$ in the primal space where $\varphi(\cdot) : \mathbb{R}^d \to \mathbb{R}^{d_f}$ denotes the potentially infinite ($d_f = \infty$) dimensional feature map. The regularized least squares cost function is given as [10]

$$\min_{w,b,e_i} \mathcal{J}_\gamma(w,e) = \frac{1}{2}w^Tw + \frac{\gamma}{2}\sum_{i=1}^N e_i^2 \quad \text{s.t.} \quad w^T\varphi(x_i) + b + e_i = y_i, \quad i = 1,\ldots,N. \quad (1)$$

Note that the regularization constant γ appears here as in classical Tikhonov regularization [1]. The solution corresponds with a form of ridge regression [13], regularization networks [4], Gaussian processes [14] and kriging [15], but usually considers a bias term b and formulates the problem in a primal-dual optimization context. The Lagrangian of the constrained optimization problem becomes

$$\mathcal{L}_\gamma(w,b,e_i;\alpha_i) = \frac{1}{2}w^Tw + \frac{\gamma}{2}\sum_{i=1}^N e_i^2 - \sum_{i=1}^N \alpha_i(w^T\varphi(x_i) + b + e_i - y_i) \quad (2)$$

By taking the conditions for optimality $\partial\mathcal{L}_\gamma/\partial\alpha_i = 0$, $\partial\mathcal{L}_\gamma/\partial b = 0$, $\partial\mathcal{L}_\gamma/\partial e_i = 0$ and $\partial\mathcal{L}_\gamma/\partial w = 0$ and application of the kernel trick $K(x_i,x_j) = \varphi(x_i)^T\varphi(x_j)$ with a positive definite (Mercer) kernel K, one gets following conditions for optimality

$$\begin{cases} y_i &= w^T\varphi(x_i) + b + e_i, \quad i = 1,\ldots,N \quad (a) \\ 0 &= \sum_{i=1}^N \alpha_i \quad (b) \\ e_i\gamma &= \alpha_i, \quad i = 1,\ldots,N \quad (c) \\ w &= \sum_{i=1}^N \alpha_i\varphi(x_i) \quad (d) \end{cases} \quad (3)$$

The dual problem is summarized as follows

$$\left[\begin{array}{c|c} 0 & 1_N^T \\ \hline 1_N & \Omega + I_N/\gamma \end{array}\right]\left[\begin{array}{c} b \\ \hline \alpha \end{array}\right] = \left[\begin{array}{c} 0 \\ \hline y \end{array}\right] \quad (4)$$

where $\Omega \in \mathbb{R}^{N\times N}$ with $\Omega_{ij} = K(x_i,x_j)$ and $y = (y_1,\ldots,y_N)^T$. The estimated function $\hat{f}$ can be evaluated at a new point x^* by $\hat{f}(x^*) = \sum_{i=1}^N \alpha_i K(x_i,x^*) + b$. The estimated outputs of the training output $\hat{y}_k$ is given as

$$\hat{y}_k = \sum_{i=1}^N \alpha_k K(x_i,x_k) + b \quad k = 1,\ldots,N. \quad (5)$$

Optimization of the optimal γ with respect to the validation performance in the regression case can be written as

$$\min_\gamma \sum_{j=1}^n (y_j^v - \hat{f}_\gamma(x_j^v))^2 = \left\| y^v - \left[\frac{1}{\Omega^v}\right]^T \left[\begin{array}{c|c} 0 & 1_N^T \\ \hline 1_N & \Omega + I_N/\gamma \end{array}\right]^{-1} \left[\frac{0}{y}\right] \right\|_2^2 \quad (6)$$

where $\Omega^v \in \mathbb{R}^{n\times N}$ with $\Omega_{ij}^v = K(x_i,x_j^v)$ and $y^v = (y_1^v,\ldots,y_n^v)^T$ The determination of γ becomes a non-convex optimization problem. For the choice of the kernel $K(\cdot,\cdot)$, see e.g. [16], [8], [7]. Typical examples are the use of a linear kernel $K(x_i,x_j) = x_i^Tx_j$ or the RBF kernel $K(x_i,x_j) = \exp(-\|x_i - x_j\|_2^2/\sigma^2)$ where σ denotes the bandwidth of the kernel.

B. Least Squares Support Vector Machine classifiers

A derivation of LS-SVMs was given originally for the classification task [9]. The LS-SVM classifier $f(x) = \text{sign}(\varphi(x)^Tw + b)$ is optimized with respect to

$$\min_{w,b,e_i} \mathcal{J}_\gamma(w,e) = \frac{1}{2}w^Tw + \frac{\gamma}{2}\sum_{i=1}^N e_i^2 \quad \text{s.t.} \quad y_i(w^T\varphi(x_i) + b) = 1 - e_i, \quad i = 1,\ldots,N. \quad (7)$$

Using a primal dual optimization interpretation, the unknowns α, b of the estimated classifier $\hat{f}(x) = \text{sign}(\sum_{i=1}^N \alpha_i y_i K(x_i,x) + b)$ are found by solving the dual set of linear equations

$$\left[\begin{array}{c|c} 0 & y^T \\ \hline y & \Omega^y + I_N/\gamma \end{array}\right]\left[\begin{array}{c} b \\ \hline \alpha \end{array}\right] = \left[\begin{array}{c} 0 \\ \hline 1_N \end{array}\right] \quad (8)$$

where $\Omega^y \in \mathbb{R}^{N\times N}$ with $\Omega_{ij}^y = y_iy_jK(x_i,x_j)$. This formulation corresponds with a form of kernel Fisher discriminant analysis. The remainder focusses on the regression case, although it is applicable just as well to the classification problem as illustrated in the experiments.

III. Convex approximation of the Regularization Tuning Problem

A. Fusion of training and validation

At first, the problem of the tuning of the regularization constant γ in (6) with respect to the performance on a validation set is considered as a constrained optimization problem. Equation (6) is equivalent to

$$\min_{\gamma,e^v,\alpha,b} \mathcal{J}_\gamma^v(e^v) = \|e^v\|_2^2 \text{ s.t. } \begin{cases} 1_N^T\alpha = & 0 \\ \Omega\alpha + \gamma^{-1}\alpha + b = & y \\ \Omega^v\alpha + b + e^v = & y^v \\ \gamma^{-1} > & 0. \end{cases} \quad (9)$$

Because of the second constraint, the optimization problem is non-convex in the unknowns α, b, e^v and γ. In the following, the b-term and the corresponding condition $1_N^T\alpha = 0$ (see

condition (3.b)) are omitted. The reduced fused problem becomes

$$\min_{\gamma,e^v,\alpha} \mathcal{J}_\gamma^{v,r}(e^v) = \|e^v\|_2^2 \text{ s.t. } \left\{ \begin{array}{rcl} \mathcal{A}(\gamma^{-1}) = & \alpha \\ \Omega^v\alpha + e^v = & y^v \\ \gamma^{-1} > & 0. \end{array} \right. \quad (10)$$

where the function $\mathcal{A}: \mathbb{R}_0^+ \to \mathbb{R}^N$ is defined as

$$\mathcal{A}(\gamma^{-1}) = \alpha = (\Omega + I_N\gamma^{-1})^{-1}y. \quad (11)$$

B. Convex approximation to fusion

This section shows how to obtain initial estimates of (10) based on a simpler kernel estimator. The key step of this approximation scheme is to use a full-rank matrix, say A_* such that

$$A_* = \arg\min_A J(A) = \|A\Omega\alpha\|_2^2 + \|A\alpha - \alpha\|_2^2 \quad \alpha \in \mathcal{C}^N \quad (12)$$

where $\mathcal{C}^N$ is an appropriate compact subset of $\mathbb{R}^N$ and A_*y as well as $A_*\alpha$ are assumed to be nonzero. Let $\epsilon = A\Omega\alpha$ for $\epsilon \in \mathbb{R}^N$. Using (11)

$$\begin{array}{rcl} \gamma\Omega\alpha + \alpha & = & \gamma y \\ \Leftrightarrow \gamma A_*\Omega\alpha + A_*\alpha & = & \gamma A_* y \\ \Leftrightarrow \gamma\epsilon + \alpha & = & \gamma A_* y. \end{array} \quad (13)$$

Under the assumption that $J(A_*)$ is sufficiently small, one obtains

$$\begin{array}{rcl} \alpha & \approx & \gamma A_* y \\ A_*\alpha & \approx & \alpha \end{array} \quad (14)$$

Let $\tilde{\alpha}$ be an approximation to α, such that

$$\tilde{\alpha} = \gamma A_* y. \quad (15)$$

From (10) we obtain the following convex optimization problem which is linear in the unknowns

$$\min_{\gamma,e^v,\tilde{\alpha}} \mathcal{J}_\gamma^{a,v}(e^v) = \|e^v\|_2^2 \text{ s.t. } \left\{ \begin{array}{rcl} \tilde{\alpha} = & \gamma A_* y \\ \Omega^v\tilde{\alpha} + e^v = & y^v \\ \gamma > & 0. \end{array} \right. \quad (16)$$

We omit the constraint $\gamma > 0$ in the next derivation (as this condition is usually satisfied and can also be checked afterwards). One constructs the Lagrangian $\mathcal{L} = \frac{1}{2}\|e^v\|_2^2 - \beta^T(\tilde{\alpha} - \gamma A_* y) - \xi^T(\Omega^v\tilde{\alpha} + e^v - y^v)$ with Lagrange multiplier vectors β and ξ. By taking the conditions for optimality $\partial\mathcal{L}/\partial\gamma = 0$, $\partial\mathcal{L}/\partial e_v = 0$, $\partial\mathcal{L}/\partial\tilde{\alpha} = 0$, $\partial\mathcal{L}/\partial\beta = 0$, $\partial\mathcal{L}/\partial\xi = 0$, one obtains

$$\left\{ \begin{array}{rl} e^v & = \xi \\ \beta^T A_* y & = 0 \\ \beta^T & = -\xi^T\Omega^v \\ \tilde{\alpha} & = \beta^T A_* y \\ y^v & = \Omega^v\tilde{\alpha} + e^v \end{array} \right. \quad (17)$$

which results into the following set of linear equations after elimination of the Lagrange multipliers β and ξ:

$$\left[\begin{array}{c|c|c} I_N & 0 & -A_* y \\ \hline \Omega^v & I_n & 0 \\ \hline 0 & (\Omega^v A_* y)^T & 0 \end{array} \right] \left[\begin{array}{c} \tilde{\alpha} \\ \hline e^v \\ \hline \gamma \end{array} \right] = \left[\begin{array}{c} 0_N \\ \hline y^v \\ \hline 0 \end{array} \right] \quad (18)$$

One sees that both the weights of the network and the regularization constant jointly follow from this linear system.

Alternatively, using elimination of the parameters α and e^v, (16) results into

$$\hat{\gamma} = \arg\min_\gamma \mathcal{J}_\gamma^{a,v}(\gamma) = \|\gamma\,\Omega^v A_* y - y_v\|_2^2 \text{ s.t. } \gamma > 0 \quad (19)$$

which can be solved as a quadratic optimization problem in γ.

C. Choosing A_*

Let us focus now on the choice of an appropriate matrix A_*. Assume that there exists a vector $\tilde{e}$ such that $\tilde{\alpha} = \gamma\tilde{e} = \gamma A_* y$. For the LS-SVM estimator (depending from the regularization constant γ), one can consider the smoother matrix S^{LS} for the LS-SVM which satisfies

$$\hat{y} = S_\gamma^{\text{LS}} y = \Omega(\Omega + I_N\gamma^{-1})^{-1}y. \quad (20)$$

One has

$$\tilde{e} = A_{*,\gamma}y = y - \hat{y} = y - S_\gamma^{\text{LS}}y = (I_N - S^{\text{LS}_\gamma})y \quad (21)$$

Hence

$$A_{*,\gamma} = (I_N - S_\gamma^{\text{LS}}) \quad (22)$$

Also

$$\alpha - A_{*,\gamma}\alpha = \alpha - \left(I_N - S_\gamma^{\text{LS}}\right)\alpha = S_\gamma^{\text{LS}}\alpha \quad (23)$$

Note that by substituting $A_{*,\gamma}$ in (16), one recovers the original (nonconvex) fusion problem (10).

To break the circle of the fact that the matrix $A_{*,\gamma}$ depends on the regularization constant, the Nadaraya-Watson kernel estimator (see e.g. [5]) is used towards the smoother matrix

$$\hat{f}(x) = \frac{\sum_{i=1}^N K(\frac{x-x_i}{h})y_i}{\sum_{i=1}^N K(\frac{x-x_i}{h})} \quad (24)$$

where the strict positive kernel K satisfies $\int K(x,z)dz = 1$ for all x. As such, A_* can be written as

$$A = I_N - S^{\text{NW}} \quad (25)$$

where the full rank matrix S^{NW} is given as $S_{ij}^{\text{NW}} = (K(\frac{x_i-x_j}{h})y_i)/(\sum_{k=1}^N K(\frac{x_i-x_k}{h}))$. The plugin method is commonly used to set the additional smoothing parameter h.

IV. Further use of the convex approximation

A. Accelerating grid search

A highly generic but often less efficient tuning procedure is the so-called *grid search* (see Figure 2.a). The convex estimate can however speed-up the experiments towards defining the grid and its search region.

B. Improving the initial estimate

Once an initial estimate is given of the solution to fusion (9), the solution can be refined. In order to do so, a variation on the theme of the Rayleigh coefficient iteration [17] is proposed, using iteratively linearizations of the quadratic constraint. This iterative procedure has also strong resemblance to the Newton-Raphson procedure and the sequential quadratic optimization algorithm [18].

Say, an initial estimate γ_0 (and its corresponding $\tilde{\alpha}$) is found. The first order Taylor approximation of the function $\mathcal{A}$ around center γ_0^{-1} is given as

$$\begin{aligned}\tilde{\mathcal{A}_{\gamma_0}}(\gamma^{-1}) &= \mathcal{A}(\gamma_0^{-1}) + (\gamma^{-1} - \gamma_0^{-1})\,\frac{d\mathcal{A}}{d\gamma^{-1}} \\ &= \mathcal{A}(\gamma_0^{-1}) + (\gamma^{-1} - \gamma_0^{-1})\,U\tilde{\Lambda}_{\gamma_0}^{-2}U^T y, \quad (26)\end{aligned}$$

where the Singular Value Decomposition (SVD, [17]) of the positive definite symmetric kernel matrix is given as $\Omega = U\Lambda U^T$ where $U \in \mathbb{R}^{N\times N}$ is a symmetric unitary matrix and $\Lambda = \mathrm{diag}(\lambda_1, \ldots, \lambda_N) \in \mathbb{R}^{N\times N}$. Here, $\lambda_i \geq 0$ for all $i = 1, \ldots, N$ denote the singular values. Let $\Lambda_\gamma^{-1} = \mathrm{diag}(\lambda_{\gamma,1}^{-1}, \ldots, \lambda_{\gamma,N}^{-1})$ and $\lambda_{\gamma,i} = \lambda_i + \gamma^{-1}$ for all $i = 1, \ldots, N$. The linearization of fusion (9) becomes

$$\begin{aligned}&\min_{\gamma, e^v, \tilde{\alpha}} \mathcal{J}_\gamma^{a,v}(e^v) = \|e^v\|_2^2 \text{ s.t.} \\ &\quad \begin{cases} \tilde{\alpha} &= \mathcal{A}(\gamma_0^{-1}) + \left(\gamma^{-1} - \gamma_0^{-1}\right)\, U\tilde{\Lambda}_{\gamma_0}^{-2}U^T y \\ y^v &= \Omega^v\tilde{\alpha} + e^v \\ \gamma^{-1} &> 0. \end{cases} \quad (27)\end{aligned}$$

which can be solved as a linear constrained optimization as in Subsection III-B. Computational more attractive, the variables $\tilde{\alpha}$ and e^v are eliminated resulting in following quadratic optimization problem (as in (19))

$$\begin{aligned}&\hat{\gamma} = \arg\min_\gamma \mathcal{J}_\gamma^{t,v}(\gamma) \\ &= \left\| \gamma^{-1}\,\Omega^v \frac{d\mathcal{A}}{d\gamma^{-1}} + \Omega^v \left(\mathcal{A}(\gamma_0^{-1}) - \gamma_0^{-1}\,\frac{d\mathcal{A}}{d\gamma^{-1}} \right) - y^v \right\|_2^2 \\ &\qquad \text{such that} \quad \gamma > 0. \quad (28)\end{aligned}$$

We end this Subsection by recapitulating the key steps for a practical algorithm for regularization constant tuning.

Algorithm 1: **Regularization constant tuning**

1) Find an initial estimate γ_0 using (19);
2) Compute the SVD of the kernel matrix;
3) Given γ_0, construct a linearization of $\mathcal{A}$;
4) Given the linearization, find a new estimate γ_1 by solving the fusion problem (28);
5) Iterate from step 3. with $\gamma_0 \leftarrow \gamma_1$.

Note that numerical implementations based on empirical estimates of the derivative can be devised making the computational burden of step 2 redundant.

C. Using the L-fold Cross-Validation Score Function

Extension of the elaborated algorithm towards an L-fold Cross-Validation (CV) criterion is mainly a matter of bookkeeping of indices. The L-fold cross-validation score function is based on a division of the dataset in L disjunct subsets of similar size covering the training data exactly. In the lth fold, the data contained in the lth subset is used for validation based on a model trained on all remaining data. Following notation is used for training data ($x^{(l)}, y^{(l)}$ and kernel matrix $\Omega^{(l)}$), validation data ($x^{(l)v}, y^{(l)v}$ and kernel matrix $\Omega^{(l)v}$) and LS-SVM (with associated function $\mathcal{A}^{(l)}$ and solution $\alpha^{(l)}$ as in (11)) in the lth fold for all $l = 1, \ldots, L$. In order to make the estimation unbiased towards a specific data arrangement, every individual estimate should be based on a fresh data division. The following local search omits this step as it would lead to a non-smooth costfunction. So, given a fixed data division of L disjunctive sets, the CV tuning problem can be written as

$$\hat{\gamma}^{cv} = \arg\min_\gamma \sum_{l=1}^{L} \left\| \Omega^{(l)v}\mathcal{A}(\gamma^{-1}) - y^{(l)v} \right\|_2^2 \quad (29)$$

As the first order Taylor approximation equals the sum of the individual Taylor approximations, the derivations of previous subsection can be used almost straightforwardly. The key issue is that the step size (determined from (28)) should be replaced by

$$\begin{aligned}&\hat{\gamma} = \arg\min_\gamma \mathcal{J}_\gamma^{t,cv}(\gamma) \\ &= \sum_{l=1}^{L} \left\| \gamma^{-1}\,\Omega^{(l)v}\frac{d\mathcal{A}^{(l)}}{d\gamma^{-1}} + \Omega^{(l)v}\left(\mathcal{A}^{(l)}(\gamma_0^{-1}) - \gamma_0^{-1}\,\frac{d\mathcal{A}^{(l)}}{d\gamma^{-1}} \right) - y^{(l)v} \right\|_2^2 \\ &\qquad \text{such that} \quad \gamma > 0. \quad (30)\end{aligned}$$

Again, numerical estimates of the derivative using the difference of two succesive estimates can make the SVD step obsolete.

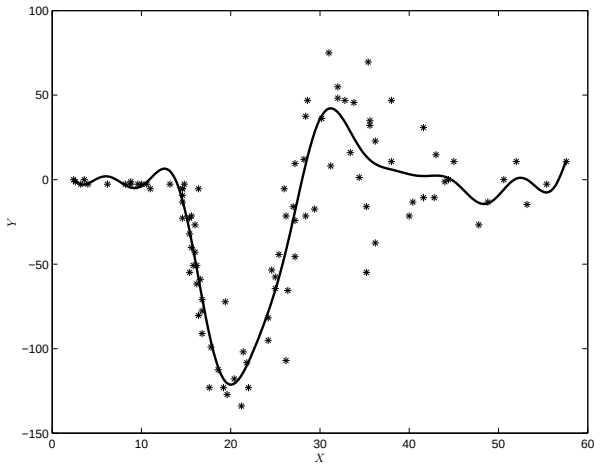
V. Illustrative examples

A. Example 1: motorcycle regression data

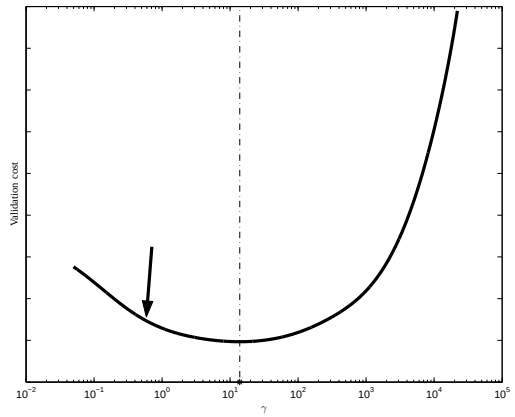
The motorcycle dataset is used to explore the capacities of proposed method on a real world regression task. The data is normalized (zero mean and unit variance). The data is given in Figure 1.a. Two-third of the observations (100 datapoints) were used for training, the remaining 33 were reserved for validation. The validation performance with respect to the regularization trade-off is given in Figure 1.b. The black arrow gives the initial value of γ found by proposed approximation. The dashed-dotted line indicates the resulting regularization parameter after 10 additional iterations. Tuning of the left hyper-parameter (kernel parameter σ^2) of the LS-SVM is reduced to a one-dimensional line search Figure 1.c.

B. Example 2: Pima classification problem

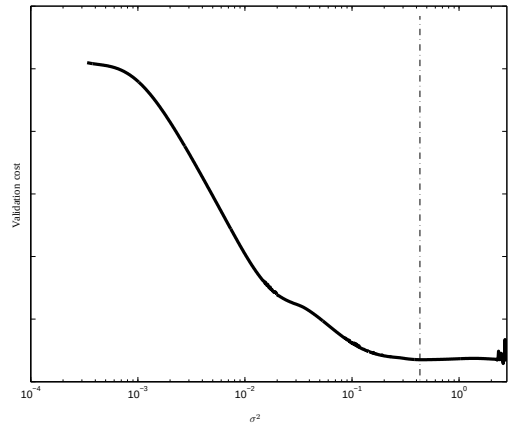
The Pima Indians Diabetes Database dataset of the UCI Machine Learning Repository was used to illustrate the tuning on a classification task. 512 observations were considered for training and tuning purposes. (The remaining 256 observations could be used for testing purposes in a comparative study with other models). The observations consists of 8 input variables and a binary label as output. An indication of the computational speedups is given The computation of the grid shown in Figure 2.a using 10-fold crossvalidation took ± 140



(a)

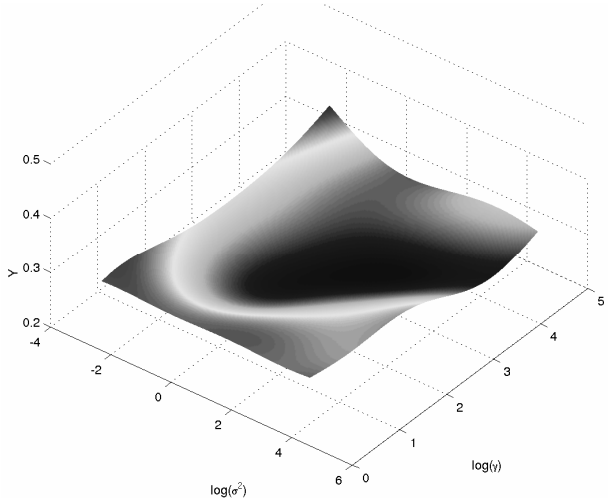


(b)

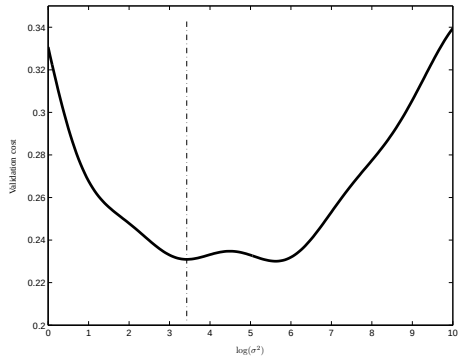


(c)

Fig. 1. *(a) Experiments on the motorcycle dataset; (b) The cost function of the validation performance w.r.t. regularization problem. The black arrow gives the initial estimate; (c) Only additional tuning of the kernel parameter σ is needed. The dashed-dotted line indicates the minimum.*



(a)



(b)

Fig. 2. *Experiments on the Pima Indians Diabetes Database dataset of the UCI Machine Learning Repository; (a) The generic implementation* grid search *makes an exhaustive exploration of the hyper-parameter space (here γ and σ^2 displayed in log-log scale) and associated model selection cost surface (*10*-fold cross validation); (b) Using proposed estimate of γ and a few extra iterations of the cross-validation score function minimizer, the tuning problem is restricted to a one-dimensional line search over the kernel parameter (σ^2). The dashed-dotted line indicates the minimum.*

minutes on a PIV Intel Linux machine (The surface is based on 900 evaluations of the costfunction for different combinations of values of γ and σ). The cost function of Figure 2.b shows for each value of σ (for 30 different values), the cross-validation performance measure for a model using an optimal γ found by the initial estimate and 10 extra iterations based on cross-validation. The computation took 8 minutes on the same machine.

VI. Conclusions

This paper approaches an acute question for practitioners of regularized models as for example LS-SVMs, namely the tuning of the regularization trade-off. Three imporant issues are dealt with in this paper, namely (1) the constrained optimization problem underlying this tuning is made explicit; (2) the non-linear constraint causing the problem behave in a non-convex way is identified and substituted by a linear one based on a simple kernel estimator; (3) a practical setup was completed by the elaboration of a refinement procedure of this initial estimate. This iterative refinement is described for the validation as well as the L-fold crossvalidation cost function.

Acknowledgements. This research work was carried out at the ESAT laboratory of the Katholieke Universiteit Leuven. It is supported by grants from several funding agencies and sources: Research Council KU Leuven: Concerted Research Action GOA-Mefisto 666 (Mathematical Engineering), IDO (IOTA Oncology, Genetic networks), several PhD/postdoc & fellow grants; Flemish Government: Fund for Scientific Research Flanders (several PhD/postdoc grants, projects G.0407.02 (support vector machines), G.0256.97 (subspace), G.0115.01 (bio-i and microarrays), G.0240.99 (multilinear algebra), G.0197.02 (power islands), research communities ICCoS, ANMMM), AWI (Bil. Int. Collaboration Hungary/ Poland), IWT (Soft4s (softsensors), STWW-Genprom (gene promotor prediction), GBOU-McKnow (Knowledge management algorithms), Eureka-Impact (MPC-control), Eureka-FLiTE (flutter modeling), several PhD grants); Belgian Federal Government: DWTC (IUAP IV-02 (1996-2001) and IUAP V-10-29 (2002-2006) (2002-2006): Dynamical Systems and Control: Computation, Identification & Modelling), Program Sustainable Development PODO-II (CP/40: Sustainibility effects of Traffic Management Systems); Direct contract research: Verhaert, Electrabel, Elia, Data4s, IPCOS. JS is an associate professor and BDM is a full professors at K.U.Leuven Belgium, respectively.

References

[1] A. N. Tikhonov and V. Arsenin, *Solution of Ill-Posed Problems*. Washington DC: Winston, 1977.
[2] G. Wahba, *Spline models for observational data*. SIAM, 1990.
[3] C. Bishop, *Neural Networks for Pattern Recognition*. Oxford University Press, 1995.
[4] T. Poggio and F. Girosi, "Networks for approximation and learning," in *Proceedings of the IEEE*, vol. 78, no. 9. Proceedings of the IEEE, septenber 1990, pp. 1481–1497.
[5] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*. Heidelberg: Springer-Verlag, 2001.
[6] V. Vapnik, *Statistical Learning Theory*. Wiley and Sons, 1998.
[7] N. Cristianini and J. Shawe-Taylor, *An Introduction to Support Vector Machines*. Cambridge University Press, 2000.
[8] B. Schoelkopf and A. Smola, *Learning with Kernels*. MIT Press, 2002.
[9] J. A. K. Suykens and J. Vandewalle, "Least squares support vector machine classifiers," *Neural Processing Letters*, vol. 9, no. 3, pp. 293–300, 1999.
[10] J. A. K. Suykens, T. Van Gestel, D. B. J., B. De Moor, and J. Vandewalle, *Least Squares Support Vector Machines*. World Scientific, Singapore, 2002.
[11] J. A. K. Suykens, G. Horvath, S. Basu, C. Micchelli, and J. Vandewalle, Eds., *Advances in Learning Theory: Methods, Models and Applications*, ser. NATO Science Series III: Computer & Systems Sciences. IOS Press Amsterdam, 2003, vol. 190.
[12] K. Pelckmans, J. A. K. Suykens, and B. De Moor, "Additive regularization: Fusion of training and validation levels in kernel methods," *(Submitted for Publication) Internal Report 03-184, ESAT-SISTA, K.U.Leuven (Leuven, Belgium)*, 2003.
[13] C. Saunders, A. Gammerman, and V. Vovk, "Rich regression learning algorithm in dual variables," in *Proc. of the 15th Int. Conf. on Machine learning(ICML'98)*. Morgan Kaufmann, 1998, pp. 515–521.
[14] D. J. C. MacKay, "The evidence framework applied to classification networks," *Neural Computation*, vol. 4, pp. 698–714, 1992.
[15] N. Cressie, *Statistics for spatial data*. Wiley, 1993.
[16] O. Chapelle, V. Vapnik, O. Bousquet, and S. Mukherjee, "Choosing multiple parameters for support vector machines," *Machine Learning*, vol. 46, no. 1-3, pp. 131–159, 2002.
[17] G. Golub and C. Van Loan, *Matrix Computations*. John Hopkins University Press, 1989.
[18] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.