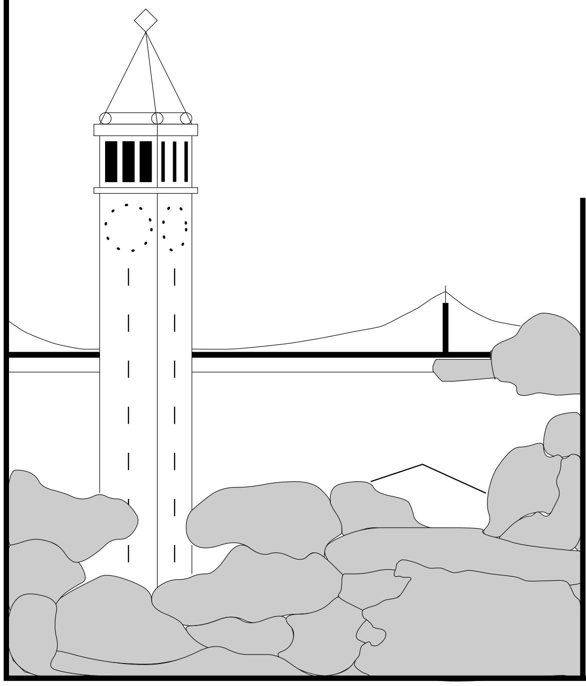
Convex Tuning of the Soft Margin Parameter

Tijl De Bie
tijl.debie@esat.kuleuven.ac.be
ESAT-SCD/SISTA
K.U. Leuven, Leuven, Belgium

Gert R.G. Lanckriet
gert@cs.berkeley.edu
Department of Electrical Engineering and Computer Science
University of California, Berkeley, CA 94720, USA

Nello Cristianini
nello@support-vector.net
Department of Statistics
University of California, Davis, CA 95616, USA



Report No. UCB/CSD-03-1289

November, 2003

Computer Science Division (EECS)
University of California
Berkeley, California 94720

Convex Tuning of the Soft Margin Parameter

Tijl De Bie
tijl.debie@esat.kuleuven.ac.be
ESAT-SCD/SISTA
K.U. Leuven, Leuven, Belgium

Gert R.G. Lanckriet
gert@cs.berkeley.edu
Department of Electrical Engineering and Computer Science
University of California, Berkeley, CA 94720, USA

Nello Cristianini
nello@support-vector.net
Department of Statistics
University of California, Davis, CA 95616, USA

November, 2003

Abstract

In order to deal with known limitations of the hard margin support vector machine (SVM) for binary classification — such as overfitting and the fact that some data sets are not linearly separable —, a soft margin approach has been proposed in literature [2, 4, 5]. The soft margin SVM allows training data to be misclassified to a certain extent, by introducing slack variables and penalizing the cost function with an error term, i.e., the 1-norm or 2-norm of the corresponding slack vector. A regularization parameter C trades off the importance of maximizing the margin versus minimizing the error. While the 2-norm soft margin algorithm itself is well understood, and a generalization bound is known [4, 5], no computationally tractable method for tuning the soft margin parameter C has been proposed so far. In this report we present a convex way to optimize C for the 2-norm soft margin SVM, by maximizing this generalization bound. The resulting problem is a quadratically constrained quadratic programming (QCQP) problem, which can be solved in polynomial time $O(l^3)$ with l the number of training samples.

1 Introduction

The first section briefly reviews the standard 2-norm soft margin SVM formulation for binary classification. In the subsequent section, we show how inspired by the approach taken in [6], the soft margin parameter C can be tuned in a convex way by optimizing the 2-norm margin cost with respect to $1/C$, subject to a trace constraint.

2 The 2-norm Soft Margin Support Vector Machine

The primal version of the standard 2-norm soft margin SVM formulation [2, 3, 4, 5] is

$$\begin{aligned} \min_{\mathbf{w},\xi_i} \quad & \frac{1}{2}\langle \mathbf{w} \cdot \mathbf{w} \rangle + \frac{C}{2}\sum_{i=1}^{l} \xi_i^2 \\ \text{s.t.} \quad & y_i(\langle \mathbf{w} \cdot \Phi(\mathbf{x}_i)\rangle + b) \geq 1 - \xi_i \quad \text{for } i = 1, \ldots l. \end{aligned} \tag{1}$$

As shown in [4] and [5], this boils down to maximizing the margin in an augmented feature space.

The corresponding dual optimization problem is given by

$$\begin{aligned} \max_{\alpha_i} \quad & \sum_{i=1}^{l} \alpha_i - \frac{1}{2}\sum_{i,j=1}^{l} y_i y_j \alpha_i \alpha_j \left(\langle \Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}_j) \rangle + \frac{1}{C}\delta_{ij} \right) \\ \text{s.t.} \quad & \alpha_i \geq 0 \quad \text{for } i = 1, \ldots l, \\ & \sum_{i=1}^{l} \alpha_i y_i = 0. \end{aligned}$$

Using the notation $[\mathbf{G}(\mathbf{K})]_{ij} = [\mathbf{K}]_{ij} y_i y_j = \langle \Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}_j) \rangle y_i y_j$, $\mathbf{e} = (1 \cdots 1)^T$, $\mathbf{y} = (y_1 \cdots y_l)^T$, $\boldsymbol{\alpha} = (\alpha_1 \cdots \alpha_l)^T$ and $\gamma = \frac{1}{C}$, this becomes

$$\begin{aligned} \max_{\boldsymbol{\alpha}} \quad & \boldsymbol{\alpha}^T \mathbf{e} - \frac{1}{2}\, \boldsymbol{\alpha}^T \mathbf{G}(\mathbf{K} + \gamma \mathbf{I}) \boldsymbol{\alpha} \\ \text{s.t.} \quad & \boldsymbol{\alpha} \geq 0, \\ & \boldsymbol{\alpha}^T \mathbf{y} = 0, \end{aligned} \tag{2}$$

where $\boldsymbol{\alpha} \geq 0$ is a componentwise inequality. So, using a 2-norm soft margin criterion boils down to modifying the kernel by adding $1/C$ to the diagonal. The optimal value of primal (1) and dual (2) coincide and are equal to the inverse margin of the hard margin SVM in the augmented feature space.

The problem of tuning the 2-norm soft margin parameter can be related to the more general kernel learning methodology presented in [6], which shows how to learn the best linear combination $\sum_i \mu_i \mathbf{K}_i$ of a given set of kernel matrices $\{K_1, K_2, \ldots, K_m\}$. The key observation connecting both problems is that in the tuning the soft margin parameter, we also want to learn an optimal linear combination of kernel matrices, namely $\mathbf{K}$ and $\mathbf{I}$. I.e., we want to learn the optimal value of the 'combination' parameter $\gamma = \frac{1}{C}$. Thus, inspired by the generalization bound provided in [4] and [5], we will apply a methodology similar to the approach adopted in [6], leading to an optimization problem that is convex in $\gamma = 1/C$.

More concretely, this generalization bound depends on the trace of the augmented kernel matrix $\mathbf{K} + \gamma\mathbf{I}$ — a larger trace leading to a looser bound — and on the margin achieved in the augmented feature space — a larger margin leading to a tighter bound. Instead of optimizing both quantities however, we fix the trace by normalizing the augmented kernel matrix $\mathbf{K} + \gamma\mathbf{I}$ by dividing it by its trace. Then we only have to maximize the margin in the augmented feature space corresponding to this normalized kernel, which can be accomplished by minimizing the optimal value of the objective of (1) or (2) where $\mathbf{K} + \gamma\mathbf{I}$ is replaced by $\frac{\mathbf{K}+\gamma\mathbf{I}}{\text{trace}(\mathbf{K})+\gamma l}$. (Note that this normalization is allowed without altering the optimal value for $\boldsymbol{\alpha}$, since (2) is homogeneous in trace$(\mathbf{K} + \gamma\mathbf{I})$.)

Furthermore, we slightly change the parameterization by redefining γ as $\frac{\gamma l}{\gamma l+\text{trace}(\mathbf{K})}$, giving rise to the soft margin problem formulation that we will use throughout this report:

$$\begin{aligned} \max_{\boldsymbol{\alpha}} \quad & \boldsymbol{\alpha}^T\mathbf{e} - \frac{1}{2}\boldsymbol{\alpha}^T\mathbf{G}\left((1-\gamma)\frac{\mathbf{K}}{\text{trace}(\mathbf{K})} + \gamma\frac{\mathbf{I}}{l}\right)\boldsymbol{\alpha} \\ \text{s.t.} \quad & \boldsymbol{\alpha} \geq 0, \\ & \boldsymbol{\alpha}^T\mathbf{y} = 0. \end{aligned} \tag{3}$$

Notice how by redefining γ in this way the trace of the augmented kernel matrix is kept constant explicitly — equal to one, without loss of generality — while the object function is again linear in γ.

In the next section, we will show how this 2-norm soft margin cost function can be optimized with respect to γ, in a fast and convex way. In a first subsection, we optimize over all possible values of γ, even negative ones, constraining the resulting kernel matrix to be positive semidefinite. Note that a value of $\gamma > 1$ corresponds to a negative weight for $\mathbf{K}$, which is actually not allowed by the original parameterization in (2), and which is often undesirable. To prevent this γ can be upper bounded to one. On the other hand, a negative γ in fact corresponds to using a reduced feature space instead of an augmented one, which can be interesting in particular for diagonal dominant kernels. After this general problem setting, a subsequent subsection will deal with the standard problem where the soft margin parameter is allowed to be positive only.

3 Learning the Soft Margin Parameter using QCQP

3.1 The General Problem

In the general case, we allow γ to attain positive as well as negative values. As mentioned earlier, we maximize the margin in the augmented/reduced feature space, while keeping the trace of the resulting kernel matrix constant, by minimizing the optimal value of the objective of (3). Since γ is allowed to be negative and larger than one, we need an additional constraint to ensure the positive semidefiniteness of the augmented/reduced kernel matrix. This yields

$$\begin{aligned} \min_{\gamma} \quad \max_{\boldsymbol{\alpha}} \quad & \boldsymbol{\alpha}^T\mathbf{e} - \frac{1}{2}\boldsymbol{\alpha}^T\mathbf{G}\left((1-\gamma)\frac{\mathbf{K}}{\text{trace}(\mathbf{K})} + \gamma\frac{\mathbf{I}}{l}\right)\boldsymbol{\alpha} \\ \text{s.t.} \quad & \boldsymbol{\alpha} \geq 0, \\ & \boldsymbol{\alpha}^T\mathbf{y} = 0, \\ & (1-\gamma)\frac{\mathbf{K}}{\text{trace}(\mathbf{K})} + \gamma\frac{\mathbf{I}}{l} \succeq 0. \end{aligned} \tag{4}$$

The last constraint can be reformulated as two linear constraints in terms of the smallest and the largest eigenvalue $\lambda_{\min}$ and $\lambda_{\max}$ of $\mathbf{K}$. Indeed, we only need to assure that the smallest eigenvalue of the reduced/augmented kernel matrix

$$\kappa = \min\left\{\lambda_{\min}\frac{1-\gamma}{r} + \frac{\gamma}{l}, \lambda_{\max}\frac{1-\gamma}{r} + \frac{\gamma}{l}\right\},$$

is non-negative:

$$
\begin{aligned}
&\text{for } \gamma \geq 1 \ : \ \kappa = \lambda_{\min}\frac{1-\gamma}{r} + \frac{\gamma}{l} \geq 0 \quad \Leftrightarrow \quad \gamma \geq \frac{-1}{\frac{r}{l\lambda_{\min}} - 1} = \gamma_{\min}, \\
&\text{for } \gamma < 1 \ : \ \kappa = \lambda_{\max}\frac{1-\gamma}{r} + \frac{\gamma}{l} \geq 0 \quad \Leftrightarrow \quad \gamma \leq \frac{1}{1 - \frac{r}{l\lambda_{\max}}} = \gamma_{\max},
\end{aligned}
$$

where $r = \mathrm{trace}(\mathbf{K})$. The first equivalence follows from the fact that $r = \sum_i \lambda_i > l\lambda_{\min}$, while the second uses $r = \sum_i \lambda_i < l\lambda_{\max}$, where we assume $\mathbf{K}$ is not a scaled version of the unity matrix $\mathbf{I}$. Note that $\gamma_{\min}$ is always negative, while $\gamma_{\max}$ is always larger than 1. They can be computed in $O(l^3)$.

Optimization problem (4) then becomes

$$
\begin{aligned}
\min_\gamma \max_{\boldsymbol{\alpha}} \quad & \boldsymbol{\alpha}^T \mathbf{e} - \frac{1}{2}\boldsymbol{\alpha}^T \mathbf{G}\left((1-\gamma)\frac{\mathbf{K}}{r} + \gamma\frac{\mathbf{I}}{l}\right)\boldsymbol{\alpha} \\
\text{s.t.} \quad & \boldsymbol{\alpha} \geq 0, \\
& \boldsymbol{\alpha}^T \mathbf{y} = 0, \\
& \gamma \geq \gamma_{\min}, \\
& \gamma \leq \gamma_{\max}.
\end{aligned}
$$

Note that all constraints are linear in both $\boldsymbol{\alpha}$ and γ. Since the objective is convex in γ (it is linear in γ) and concave in $\boldsymbol{\alpha}$, and because the minimization problem is strictly feasible in γ, and the maximization problem strictly feasible in $\boldsymbol{\alpha}$ — we can skip the case for all elements of $\mathbf{y}$ having the same sign, because we cannot even define a margin in such a case —, standard results from convex optimization (see, e.g., [1]) allow us to interchange the order of the minimization and the maximization. This yields

$$
\begin{aligned}
\min_{\gamma:\gamma_{\max}\geq\gamma\geq\gamma_{\min}} \quad & \max_{\boldsymbol{\alpha}} \quad \boldsymbol{\alpha}^T \mathbf{e} - \frac{1}{2}\boldsymbol{\alpha}^T \mathbf{G}\left((1-\gamma)\frac{\mathbf{K}}{r} + \gamma\frac{\mathbf{I}}{l}\right)\boldsymbol{\alpha} \\
= \quad & \max_{\boldsymbol{\alpha}} \quad \left(\boldsymbol{\alpha}^T \mathbf{e} - \frac{1}{2}\max_{\gamma:\gamma_{\max}\geq\gamma\geq\gamma_{\min}}\left[\boldsymbol{\alpha}^T \mathbf{G}\left((1-\gamma)\frac{\mathbf{K}}{r} + \gamma\frac{\mathbf{I}}{l}\right)\boldsymbol{\alpha}\right]\right) \\
= \quad & \max_{\boldsymbol{\alpha}} \quad \left(\boldsymbol{\alpha}^T \mathbf{e} - \frac{1}{2}\max\left\{\boldsymbol{\alpha}^T \mathbf{G}\left((1-\gamma_{\min})\frac{\mathbf{K}}{r} + \gamma_{\min}\frac{\mathbf{I}}{l}\right)\boldsymbol{\alpha},\right.\right. \\
& \qquad \left.\left.\boldsymbol{\alpha}^T\left((1-\gamma_{\max})\frac{\mathbf{G}(\mathbf{K})}{r} + \gamma_{\max}\frac{\mathbf{I}}{l}\right)\boldsymbol{\alpha}\right\}\right) \\
\text{s.t.} \quad & \boldsymbol{\alpha} \geq 0, \\
& \boldsymbol{\alpha}^T \mathbf{y} = 0,
\end{aligned}
$$

or equivalently,

$$\begin{aligned}
\max_{\boldsymbol{\alpha}} \quad & \boldsymbol{\alpha}^T \mathbf{e} - \frac{1}{2} t \\
\text{s.t.} \quad & \boldsymbol{\alpha} \geq 0, \\
& \boldsymbol{\alpha}^T \mathbf{y} = 0, \\
& t \geq \boldsymbol{\alpha}^T \mathbf{G} \left((1-\gamma_{\max}) \frac{\mathbf{K}}{r} + \gamma_{\max} \frac{\mathbf{I}}{l} \right) \boldsymbol{\alpha}, \\
& t \geq \boldsymbol{\alpha}^T \mathbf{G} \left((1-\gamma_{\min}) \frac{\mathbf{K}}{r} + \gamma_{\min} \frac{\mathbf{I}}{l} \right) \boldsymbol{\alpha}.
\end{aligned} \tag{5}$$

Often we do not want the weight of $\mathbf{K}$ to be negative, i.e., we want $1 - \gamma \geq 0$ or $\gamma \leq 1$. Since $\gamma_{\max} \geq 1$, as mentioned earlier, this can be accomplished by replacing $\gamma_{\max}$ by 1 in (5), yielding

$$t \geq \frac{\boldsymbol{\alpha}^T \boldsymbol{\alpha}}{l},$$

for the third constraint.

The resulting problem is a quadratically constrained quadratic programming (QCQP) problem [1], which can be solved in $O(l^3)$. Since the complexity for computing $\gamma_{\min}$ and $\gamma_{\max}$ is similar, the complexity of the entire algorithm is $O(l^3)$ as well.

3.2 The Standard Problem, $\gamma \geq 0$

Since the standard 2-norm soft margin SVM formulation assumes $C > 0$, both $\mathbf{K}$ and $\mathbf{I}$ should have positive weights when combined, meaning that $1 \geq \gamma \geq 0$. Since $\gamma_{\min} \leq 0$, as mentioned earlier, we replace $\gamma_{\min}$ by 0 in (5) and finally obtain

$$\begin{aligned}
\max_{\boldsymbol{\alpha}} \quad & \boldsymbol{\alpha}^T \mathbf{e} - \frac{1}{2} t \\
\text{s.t.} \quad & \boldsymbol{\alpha} \geq 0, \\
& \boldsymbol{\alpha}^T \mathbf{y} = 0, \\
& t \geq \frac{\boldsymbol{\alpha}^T \boldsymbol{\alpha}}{l}, \\
& t \geq \boldsymbol{\alpha}^T \frac{\mathbf{G}(\mathbf{K})}{r} \boldsymbol{\alpha},
\end{aligned}$$

again a QCQP problem, which can be solved efficiently in $O(l)^3$.

References

[1] S. Boyd, L. Vandenberghe, "Convex Optimization", Course notes for EE364, Stanford University, 2003. Available at `http://www.stanford.edu/class/ee364`.

[2] C. Cortes, V. Vapnik, "Support Vector Networks", Machine Learning, 20:273–297, 1995.

[3] Nello Cristianini, John Shawe-Taylor, "Support Vector Machines and other kernel-based learning methods", Cambridge University Press, 2000.

[4] John Shawe-Taylor, Nello Cristianini, "Margin Distribution and Soft Margin", in Advances in Large Margin Classifiers; MIT Press; ed. by A. Smola; B. Schoelkopf; P. Bartlett; D. Schuurmans, 1999.

[5] John Shawe-Taylor, Nello Cristianini, "On the Generalization of Soft Margin Algorithms", IEEE Transactions on Information Theory, 1999.

[6] Gert Lanckriet, Nello Cristianini, Peter Bartlett, Laurent El Ghaoui, Michael Jordan, "Learning the Kernel Matrix with Semi-Definite Programming", in C. Sammut and A. Hoffmann (Eds.), Proceedings of the 19th International Conference on Machine Learning, Sydney, Australia: Morgan Kaufmann, 2002.