Gaussian Margin Machines

Koby Crammer
Dep. of Comp. and Information Science
University of Pennsylvania
Philadelphia, PA 19104

Mehryar Mohri
Courant Institute and
Google Research
New York, NY 10012

Fernando Pereira
Google, Inc.
Mountain View
California

Abstract

We introduce Gaussian Margin Machines (GMMs), which maintain a Gaussian distribution over weight vectors for binary classification. The learning algorithm for these machines seeks the least informative distribution that will classify the training data correctly with high probability. One formulation can be expressed as a convex constrained optimization problem whose solution can be represented linearly in terms of training instances and their inner and outer products, supporting kernelization. The algorithm admits a natural PAC-Bayesian justification and is shown to minimize a quantity directly related to a PAC-Bayesian generalization bound. A preliminary evaluation on handwriting recognition data shows that our algorithm improves on SVMs for the same task, achieving lower test error and lower test error variance.

1 Introduction

Linear classifiers learned with support vector machine (SVM) methods (Cortes and Vapnik, 1995; Boser *et al.*, 1992) are widely used and commonly regarded as the state of the art for a variety of learning tasks. SVMs and most other linear classification learners output a *single* weight vector but they do not supply additional information about alternative weight vectors or a confidence information associated to the weight vector learned.

Bayesian methods, on the other hand, maintain a distribution over weight vectors and do not commit to a single choice. This posterior distribution follows by Bayes's rule from the prior distribution and the training observations, and in theory provides for randomized optimal decisions assuming that the prior distribution correctly models the constraints of the situation. Unfortunately, the posterior distribution is very complicated even for simple Bayesian logistic regression (Jaakkola and Jordan, 1997), requiring approximations that limit the applicability and effectiveness of the Bayesian approach.

We propose here a learning objective which draws from both SVMs and Bayesian ideas. As in Bayesian methods, we maintain a distribution over alternative weight vectors, rather than committing to a single specific one. However, these distributions are not derived by Bayes' rule. Instead, they represent our knowledge of the weights given constraints imposed by the training examples, expressed as a Gaussian distribution over weight vectors, learned from the training data. The learning algorithm seeks a distribution with small relative entropy with respect to a fixed isotropic distribution, such that each training example is correctly classified by a strict majority of the weight vectors. This condition can be viewed as a probabilistic version of the geometric large-margin principle underlying algorithms such as SVMs.

The learning problem for GMMs is a convex constrained optimization whose optimal solution is a linear combination of training instances and their inner and outer products, thereby supporting the use of arbitrary Mercer kernels. The form of the algorithm allows us to use directly the PAC-Bayesian family of generalization bounds. Alternatively, a slight variant of the algorithm can be seen as a robust variant of SVMs.

We compare the performance of GMMs to SVMs on a handwritten digit classification task, and show that over random samples of the problem, GMMs achieve improved average performance. We also show that GMMs are more robust in the sense that they achieve lower test error variance than SVMs.

Appearing in Proceedings of the 12^{th} International Confe-rence on Artificial Intelligence and Statistics (AISTATS) 2009, Clearwater Beach, Florida, USA. Volume 5 of JMLR: W&CP 5.

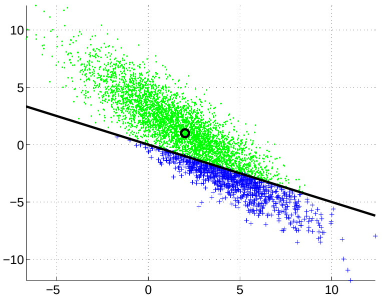

Figure 1: Gaussian distribution over two-dimensional weight vectors. Green vectors classify incorrectly the example $((0.5, 1), +1)$, blue vectors. The density around a weight vector is proportional to its relative importance. The black circle marks the mean of the Gaussian.

2 Gaussian Margin Algorithms

Standard linear classification learning algorithms return a single weight vector $\boldsymbol{w}$ used to predict the label of any test point. We study a generalization of these algorithms where hypotheses are probability distributions over weight vectors $\boldsymbol{w}$. Such a hypothesis can be seen as a randomized linear classifier. To classify an instance $\boldsymbol{x}$, a parameter vector $\boldsymbol{w}$ is drawn according to the hypothesis and predicts the label $\operatorname{sign}(\boldsymbol{w} \cdot \boldsymbol{x})$.

One benefit of this randomization is to produce a more *robust* solution, as argued by Herbrich *et al.* in a similar context (Herbrich *et al.*, 2000, 2001). PAC-Bayesian analysis and its generalization bounds give additional justification to this approach, as we shall detail in Section 4.

The probability distribution over weight vectors learned by our algorithm is selected among the family of full Gaussian distributions $\mathcal{N}(\boldsymbol{\mu}, \Sigma)$ with mean $\boldsymbol{\mu} \in \mathbb{R}^d$ and covariance matrix $\Sigma \in \mathbb{R}^{d \times d}$. The component μ_p of the mean vector and the diagonal term $\Sigma_{p,p}$ of the covariance matrix learned, $p = 1 \ldots d$, convey the partial knowledge gained about the weight assigned to feature p. The larger $\Sigma_{p,p}$ and the more diversity is allowed for the weight $\boldsymbol{w}_p$. Similarly, each covariance term $\Sigma_{p,q}$ captures the correlation between features p and q. Fig. 1 illustrates this in the case of a simple two-dimensional Gaussian distribution. The multivariate Gaussian distribution over weight vectors $\mathcal{N}(\boldsymbol{\mu}, \Sigma)$ induces a univariate Gaussian distribution over the signed margin M of the hyperplanes they define:

$$M \sim \mathcal{N}\left(y(\boldsymbol{\mu} \cdot \boldsymbol{x}), (\boldsymbol{x}^\top \Sigma \boldsymbol{x})\right) . \tag{1}$$

At prediction time, the true value of y is of course unknown and should thus be omitted from (1).

The design of our algorithm is guided by both a large-margin requirement, as with most successful deterministic linear discrimination algorithms, and the maximum entropy principle.

Given a labeled sample $S = ((\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_n, y_n))$, the maximum entropy principle invites us to seek the probability distribution over weight vectors that is the closest to an uninformative distribution, e.g., an isotropic Gaussian distribution $\mathcal{N}(\mathbf{0}, a\mathbf{I})$ for some constant scalar $a > 0$, where closeness is measured by the relative entropy, or the Kullback-Leibler divergence. The large-margin requirement imposes that in the separable case, with high probability, a weight vector drawn from $\mathcal{N}(\boldsymbol{\mu}, \Sigma)$ correctly label the training samples. A relaxed version of this condition is required in the non-separable case. The next two sections present in detail the optimization problems for both cases.

2.1 Optimization in the Separable Case

This section derives the optimization problem for learning GMMs in the case where the training sample is linearly separable. In this case, we can require the weight vectors to correctly classify all training points, with high probability, that is

$$\Pr\left[\operatorname{sign}(\boldsymbol{w} \cdot \boldsymbol{x}_i) = y_i\right] \geq \eta , \tag{2}$$

where $\eta \in (0.5, 1]$ is a fixed confidence parameter. In view of the maximum entropy principle already discussed, the optimization problem in this case can thus be written as

$$\begin{aligned} \min_{\boldsymbol{\mu}, \Sigma} \; & \mathrm{D}_{\mathrm{KL}}(\mathcal{N}(\boldsymbol{\mu}, \Sigma) \,\|\, \mathcal{N}(\mathbf{0}, a\mathbf{I})) \\ \text{s.t. } & \Pr[\operatorname{sign}(\boldsymbol{w} \cdot \boldsymbol{x}_i) = y_i] \geq \eta \quad i = 1, \ldots, n . \end{aligned} \tag{3}$$

We now give a more explicit expression for both the objective and the constraints of this optimization problem, starting with the constraints. The constraint on point $\boldsymbol{x}_i$, $i = 1, \ldots, n$, can be rewritten as

$$\Pr[y_i (\boldsymbol{w} \cdot \boldsymbol{x}_i) \geq 0] \geq \eta . \tag{4}$$

Since $\boldsymbol{w}$ is drawn from a Gaussian distribution $\mathcal{N}(\boldsymbol{\mu}, \Sigma)$, the signed-margin random variable $M_i = y_i (\boldsymbol{w} \cdot \boldsymbol{x}_i)$ for point $(\boldsymbol{x}_i, y_i)$ also follows a Gaussian distribution with the following mean and variance:

$$\mu_i = y_i (\boldsymbol{\mu} \cdot \boldsymbol{x}_i) \qquad \sigma_i^2 = \boldsymbol{x}_i^\top \Sigma \boldsymbol{x}_i . \tag{5}$$

Let Φ denote the standard normal cumulative distribution function:

$$\forall u \in \mathbb{R},\ \Phi(u) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{u} e^{-\frac{v^2}{2}} dv . \tag{6}$$

Since $(M_i - \mu_i)/\sigma_i$ is a standard normal distribution, it follows that $\Pr[y_i (\boldsymbol{w} \cdot \boldsymbol{x}_i) \geq 0]$ can be written as

$$\Pr\left[(M_i - \mu_i)/\sigma_i \geq -\frac{\mu_i}{\sigma_i}\right] = 1 - \Phi\left(-\frac{\mu_i}{\sigma_i}\right) . \tag{7}$$

Thus, the constraint (4) can be expressed in terms of Φ by

$$-\frac{\mu_i}{\sigma_i} \leq \Phi^{-1}(1 - \eta) = -\Phi^{-1}(\eta) . \tag{8}$$

Plugging back the expression of μ_i and σ_i in terms of $\boldsymbol{\mu}$ and Σ (5) leads to the following formulation of the constraint related to point $(\boldsymbol{x}_i, y_i)$:

$$y_i(\boldsymbol{\mu} \cdot \boldsymbol{x}_i) \geq \phi\sqrt{\boldsymbol{x}_i^\top \Sigma \boldsymbol{x}_i} \quad \text{where } \phi = \Phi^{-1}(\eta) \ . \tag{9}$$

This can be viewed as a large-margin constraint where the value of the margin required depends on the example $\boldsymbol{x}_i$ via a quadratic form. Interestingly, the large-margin constraint (9) arises here from a high-confidence probabilistic constraint (4), rather than from standard geometric considerations.

We now study the objective function of the optimization problem (3). The relative entropy of $\mathcal{N}(\boldsymbol{\mu}, \Sigma)$ and $\mathcal{N}(\mathbf{0}, a\mathbf{I})$ is given by

$$\begin{aligned} 2\,\mathrm{D}_{\mathrm{KL}}(\mathcal{N}(\boldsymbol{\mu}, \Sigma) \parallel \mathcal{N}(\mathbf{0}, a\mathbf{I})) &= \log\left(\frac{\det a\mathbf{I}}{\det \Sigma}\right) \\ &+ \mathrm{Tr}\left(\frac{1}{a}\Sigma\right) - d + (\boldsymbol{\mu} - \mathbf{0})^\top \frac{1}{a}(\boldsymbol{\mu} - \mathbf{0}) \ , \end{aligned} \tag{10}$$

where d is the dimension of the space. This can be written as a sum of two Bregman divergences (Censor and Zenios, 1997): the Itakura-Saito matrix divergence between the two covariance matrices (Tsuda *et al.*, 2005), and a Euclidean distance between the weight vectors.

In view of (9) and (10) and disregarding constant terms, we obtain the following explicit formulation of the optimization problem for GMMs in the separable case:

$$\begin{aligned} \min_{\boldsymbol{\mu}, \Sigma} \ & \frac{1}{2}\left(-\log\det\Sigma + \frac{1}{a}\mathrm{Tr}(\Sigma) + \frac{1}{a}\|\boldsymbol{\mu}\|^2\right) \\ \text{s.t. } & y_i(\boldsymbol{\mu} \cdot \boldsymbol{x}_i) \geq \phi\sqrt{\boldsymbol{x}_i^\top \Sigma \boldsymbol{x}_i}, \quad i = 1, \ldots, n \\ & \Sigma \succeq 0 \ . \end{aligned} \tag{11}$$

2.2 Optimization in the Non-Separable Case

To deal with the more general case of linearly non-separable samples, we can relax the inequality constraints by introducing a slack variable ξ_i for each point $\boldsymbol{x}_i$ and augmenting the objective function with a corresponding slack penalty term, as in the case of support vector machines (Cortes and Vapnik, 1995), or other similar optimization problems. Proceeding in this way, we obtain the following relaxed version of the previous optimization problem:

$$\begin{aligned} \min \ & \frac{1}{2}\left(-\log\det\Sigma + \frac{1}{a}\mathrm{Tr}(\Sigma) + \frac{1}{a}\|\boldsymbol{\mu}\|^2\right) + C\sum_{i=1}^n \xi_i \\ \text{s.t. } & y_i(\boldsymbol{\mu} \cdot \boldsymbol{x}_i) \geq \phi\sqrt{\boldsymbol{x}_i^\top \Sigma \boldsymbol{x}_i} - D_i\xi_i \\ & \Sigma \succeq 0, \text{ and } \xi_i \geq 0 \text{ for } i = 1, \ldots, n \ , \end{aligned} \tag{12}$$

where $C > 0$ is a tradeoff parameter and the D_i, $i = 1, \ldots, n$, are non-negative slack scale factors whose possible values will be discussed later.

The optimization problem just presented can be further simplified via a change of variables to eliminate the variance parameter a. Specifically, let $\tilde{\Sigma}$ and $\tilde{\boldsymbol{\mu}}$ be defined by $\tilde{\Sigma} = (\phi^2/a)\Sigma$ and $\tilde{\boldsymbol{\mu}} = (1/\sqrt{a})\boldsymbol{\mu}$. Then, the objective function can be rewritten as

$$\begin{aligned} & -\log\det\left(\frac{a}{\phi^2}\tilde{\Sigma}\right) + \frac{1}{a}\mathrm{Tr}\left(\frac{a}{\phi^2}\tilde{\Sigma}\right) + \frac{1}{a}\left\|\sqrt{a}\tilde{\boldsymbol{\mu}}\right\|^2 \\ = & -\log\det\left(\tilde{\Sigma}\right) - d\log\left(\frac{a}{\phi^2}\right) + \frac{1}{\phi^2}\mathrm{Tr}\left(\tilde{\Sigma}\right) + \|\tilde{\boldsymbol{\mu}}\|^2 \ , \end{aligned}$$

and the constraints reformulated as

$$\begin{aligned} & y_i\left(\sqrt{a}\tilde{\boldsymbol{\mu}} \cdot \boldsymbol{x}_i\right) \geq \phi\sqrt{\boldsymbol{x}_i^\top\left(\frac{a}{\phi^2}\tilde{\Sigma}\right)\boldsymbol{x}_i} - D_i\xi_i \\ \Leftrightarrow \quad & y_i\left(\tilde{\boldsymbol{\mu}} \cdot \boldsymbol{x}_i\right) \geq \sqrt{\boldsymbol{x}_i^\top \tilde{\Sigma}\boldsymbol{x}_i} - D_i\xi_i \ . \end{aligned}$$

where we absorbed the factor $1/\sqrt{a}$ into the scale factors D_i. Omitting additive constants and setting $\psi = 1/\phi^2$ leads to the following simplified form of the GMMs optimization problem for the non-separable case:

$$\begin{aligned} \min_{\boldsymbol{\mu}, \Sigma} \ & \frac{1}{2}\left(-\log\det\Sigma + \psi\mathrm{Tr}(\Sigma) + \|\boldsymbol{\mu}\|^2\right) + C\sum_{i=1}^n \xi_i \\ \text{s.t. } & y_i(\boldsymbol{\mu} \cdot \boldsymbol{x}_i) \geq \sqrt{\boldsymbol{x}_i^\top \Sigma \boldsymbol{x}_i} - D_i\xi_i \quad i = 1, \ldots, n \\ & \Sigma \succeq 0 \ , \ \xi_i \geq 0 \quad i = 1, \ldots, n \ . \end{aligned} \tag{13}$$

3 Dual Problem and Representer Theorem

This section derives the dual optimization problem for (13) and shows that any positive-definite symmetric kernel can be used for GMMs, instead of the dot product in the input space.

The objective function of (13) is convex both in $\boldsymbol{\mu}$ and Σ. The constraints are also linear in $\boldsymbol{\mu}$ and thus convex but they are *concave* in Σ. However, the change of variable $\Sigma = \Upsilon^2$, where Υ is a PSD matrix whose eigenvalues are the square roots of those for Σ, yields a convex optimization problem. The resulting optimization problem is then

$$\begin{aligned} \min_{\boldsymbol{\mu}, \Upsilon} \ & -\log\det\Upsilon + \frac{\psi}{2}\mathrm{Tr}\left(\Upsilon^2\right) + \frac{1}{2}\|\boldsymbol{\mu}\|^2 + C\sum_{i=1}^n \xi_i \\ \text{s.t. } & y_i(\boldsymbol{\mu} \cdot \boldsymbol{x}_i) \geq \|\Upsilon\boldsymbol{x}_i\| - D_i\xi_i \\ & \Upsilon \succeq 0 \ , \ \Upsilon = \Upsilon^\top \ , \ \xi_i \geq 0 \quad i = 1, \ldots, n \ . \end{aligned} \tag{14}$$

As we shall see later, the condition on Υ being PSD and symmetric can be omitted since it is always satisfied by the solution. The Lagrangian of the problem is therefore

$$\begin{aligned}\mathcal{L}(\boldsymbol{\mu}, \Upsilon; \boldsymbol{\alpha}) = & -\log\det\Upsilon + \frac{\psi}{2}\mathrm{Tr}\left(\Upsilon^2\right) + \frac{1}{2}\|\boldsymbol{\mu}\|^2 \\ & + \sum_{i=1}^{n}\alpha_i\left(\sqrt{\boldsymbol{x}_i^\top\Upsilon\Upsilon\boldsymbol{x}_i} - D_i\xi_i - y_i(\boldsymbol{\mu}\cdot\boldsymbol{x}_i)\right) \\ & + C\sum_{i=1}^{n}\xi_i - \sum_{i=1}^{n}\gamma_i\xi_i \ . \end{aligned} \tag{15}$$

where we omit the dependency of $\mathcal{L}$ on the γ_i because those multipliers can be eliminated as shown in (21) below. At the optimum, the gradient with respect to $\boldsymbol{\mu}$ and Υ is zero:

$$\nabla_{\boldsymbol{\mu}}\mathcal{L} = \boldsymbol{\mu} - \sum_{i=1}^{n}\alpha_i y_i\boldsymbol{x}_i = 0 \Rightarrow \boldsymbol{\mu} = \sum_{i=1}^{n}\alpha_i y_i\boldsymbol{x}_i \ . \tag{16}$$

$$\nabla_{\Upsilon}\mathcal{L} = -\Upsilon^{-1} + \psi\Upsilon + \sum_{i=1}^{n}\alpha_i\frac{\boldsymbol{x}_i\boldsymbol{x}_i^\top\Upsilon}{2\sqrt{\boldsymbol{x}_i^\top\Upsilon^2\boldsymbol{x}_i}} \tag{17}$$

$$+ \sum_{i=1}^{n}\alpha_i\frac{\Upsilon\boldsymbol{x}_i\boldsymbol{x}_i^\top}{2\sqrt{\boldsymbol{x}_i^\top\Upsilon^2\boldsymbol{x}_i}} = 0 \ . \tag{18}$$

Let U be defined by

$$U = \psi\mathbf{I} + \sum_{i=1}^{n}\alpha_i\frac{\boldsymbol{x}_i\boldsymbol{x}_i^\top}{\sqrt{\boldsymbol{x}_i^\top\Upsilon^2\boldsymbol{x}_i}} \ . \tag{19}$$

Then, $\nabla_{\Upsilon}\mathcal{L} = 0$ can be rewritten as $\nabla_{\Upsilon}\mathcal{L} = -\Upsilon^{-1} + \frac{1}{2}\Upsilon U + \frac{1}{2}U\Upsilon = 0$ at the optimum. From this, it follows that $\Upsilon = U^{-\frac{1}{2}}$ at the optimum, that is

$$\Upsilon^{-2} = \psi\mathbf{I} + \sum_{i=1}^{n}\alpha_i\frac{\boldsymbol{x}_i\boldsymbol{x}_i^\top}{\sqrt{\boldsymbol{x}_i^\top\Upsilon^2\boldsymbol{x}_i}} \ . \tag{20}$$

Note that this implies that Υ^{-2} and thus Υ is a PSD matrix. Finally, setting the gradient with respect to ξ_i to zero yields:

$$\nabla_{\xi_i}\mathcal{L} = -\alpha_i D_i + C - \gamma_i = 0 \implies \alpha_i \le C/D_i \ . \tag{21}$$

Let $\mathbf{X} = [\boldsymbol{x}_1 \ldots \boldsymbol{x}_n]$ be the matrix whose column vectors are the training examples $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$, and let $\mathbf{B}$ be the diagonal matrix defined by $\mathbf{B} = \mathrm{diag}(\beta_1, \ldots, \beta_d)$ where

$$\beta_i \overset{\text{def}}{=} \frac{\alpha_i}{\sqrt{v_i}} \quad \text{and} \quad v_i = \boldsymbol{x}_i^\top\Upsilon^2\boldsymbol{x}_i \ . \tag{22}$$

Denote by $\mathbf{K}$ the kernel matrix of the training data, $\mathbf{K} = \mathbf{X}^\top\mathbf{X}$, where $\mathbf{K}_{i,j} = \boldsymbol{x}_j\cdot\boldsymbol{x}_i$, and let $\boldsymbol{k}_i$ be the ith column of $\mathbf{K}$. Rewriting Equation (20) in matrix form in terms of $\mathbf{X}$ and $\mathbf{B}$ and using the matrix inversion identity (or Sherman-Morrison-Woodbury formula) to compute Υ^2 helps us derive an equivalent expression in terms of the kernel matrix $\mathbf{K}$ and the new parameters $\{\beta_i\}$ (see the Appendix for the details of this derivation). This leads to the following dual optimization problem equivalent to (14):

$$\begin{aligned} \max_{\beta_i} \ & \log\det\left(\psi\mathbf{I} + \sqrt{\mathbf{B}}\mathbf{K}\sqrt{\mathbf{B}}\right) \\ & - \frac{1}{2}\mathrm{Tr}\Big[\Big((\psi\mathbf{B})^{-1} + \mathbf{K}^{\;-1}\mathbf{K}\Big)\Big] \\ & + \sum_{i=1}^{n}\beta_i v_i - \frac{1}{2}\sum_{i,j}\beta_i\beta_j\sqrt{v_i}\sqrt{v_j}y_i y_j\mathbf{K}_{i,j} \qquad (23) \\ \text{s.t.} \ & 0 \le \beta_i \le C/(D_i\sqrt{v_i}) \quad i = 1, \ldots, n \\ & v_i = \frac{1}{\psi}\left(\mathbf{K}_{i,i} - \boldsymbol{k}_i^\top\left((\psi\mathbf{B})^{-1} + \mathbf{K}\right)^{-1}\boldsymbol{k}_i\right) \ . \end{aligned}$$

Since the dual problem is expressed in terms of the kernel matrix $\mathbf{K}$, the following result can be shown as for SVMs.

Theorem 1 *The optimal mean $\boldsymbol{\mu}$ and covariance Υ^2 parameters of* (13) *can be written as a linear combination of the input vectors where the coefficients are dependent only on inner product of the input vectors.*

The dual optimization problem helps us further understand the role of the two parameters C and ϕ (or ψ). As with SVMs, the parameter C determines the trade-off between two terms of the primal's objective (13): better accuracy on the training data (larger values) versus "simplicity" (smaller values). This trade-off translates into an upper bound on the dual parameters (23): with larger values of C, some examples may significantly affect the optimal solution. The parameter ϕ appears only in the constraints of (11). For $\phi = 0$, the constraints are invariant to Σ and lead to the optimal solution $\Sigma = \mathbf{I}$. As ϕ increases, the standard deviation of the margin $\sqrt{\boldsymbol{x}_i^\top\Sigma\boldsymbol{x}_i}$ plays an increasingly important role, producing solutions with smaller (and more skewed) eigenvalues. This can also be observed from (20): for large values of ψ (small ϕ) the solution is more similar to the identity matrix, while for smaller values of ϕ, its shape depends on the training examples.

4 Analysis

This section presents generalization bounds for GMMs both in the separable and non-separable case, based on a PAC-Bayesian analysis. PAC-Bayesian bounds were first introduced by McAllester (1999), and further refined by McAllester (2003a), and Langford and Seeger (Langford and Seeger, 2002; Seeger, 2002). They have been shown to be often quite tight. Langford and Shawe-Taylor also used PAC-Bayesian methods to analyze large-margin algorithms (Langford and Shawe-Taylor, 2002).

We first introduce some notation needed for the discussion of these bounds. Let $\ell(\boldsymbol{w}, (\boldsymbol{x}, y))$ denote the zero-one loss, that is $\ell(\boldsymbol{w}, (\boldsymbol{x}, y)) = 1$ if $\mathrm{sign}(\boldsymbol{w}\cdot\boldsymbol{x}) \neq y$ and $\ell(\boldsymbol{w}, (\boldsymbol{x}, y)) = 0$ otherwise.

Let $\mathcal{D}$ be a distribution over the labeled examples $(\boldsymbol{x}, y)$ and denote by $\ell(\boldsymbol{w}, \mathcal{D})$ the expected zero-one loss of a linear classifier characterized by its weight vector $\boldsymbol{w}$:

$$\begin{aligned}\ell(\boldsymbol{w}, \mathcal{D}) &= \Pr_{(\boldsymbol{x},y)\sim\mathcal{D}}[\operatorname{sign}(\boldsymbol{w}\cdot\boldsymbol{x}) \neq y] \\ &= \mathop{\mathrm{E}}_{(\boldsymbol{x},y)\sim\mathcal{D}}[\ell(\boldsymbol{w}, (\boldsymbol{x}, y))] \ .\end{aligned}$$

We denote abusively by $\ell(\boldsymbol{w}, S)$ the expected loss $\ell(\boldsymbol{w}, \mathcal{D}_S)$ for the empirical distribution $\mathcal{D}_S$ of a sample S. We also denote by $\ell(\mathcal{N}(\boldsymbol{\mu}, \Sigma), \mathcal{D})$ the expectation of $\ell(\boldsymbol{w}, \mathcal{D})$ over weight vectors $\boldsymbol{w}$ drawn from a Gaussian distribution $\mathcal{N}(\boldsymbol{\mu}, \Sigma)$:

$$\ell(\mathcal{N}(\boldsymbol{\mu}, \Sigma), \mathcal{D}) = \mathop{\mathrm{E}}_{\substack{(\boldsymbol{x},y)\sim\mathcal{D} \\ \boldsymbol{w}\sim\mathcal{N}(\boldsymbol{\mu},\Sigma)}}[\ell(\boldsymbol{w}, (\boldsymbol{x}, y))] \ . \qquad (24)$$

We use the following two-sided PAC-Bayesian theorem, which is a Gaussian version of a theorem of McAllester (2003b, Sec. 2).

Theorem 2 *Fix a prior distribution over weight vectors $\mathcal{N}(\boldsymbol{\mu}_0, \Sigma_0)$. For any $\delta \in [0, 1]$, with probability at least $1 - \delta$ over samples $S = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^n$ of size n, for all posterior distributions $\mathcal{N}(\boldsymbol{\mu}, \Sigma)$ the following holds:*

$$\begin{aligned}&\mathrm{D}_{\mathrm{KL}}\left(\ell(\mathcal{N}(\boldsymbol{\mu}, \Sigma), S) \,\|\, \ell(\mathcal{N}(\boldsymbol{\mu}, \Sigma), \mathcal{D})\right) \\ &\qquad \leq \frac{\mathrm{D}_{\mathrm{KL}}\left(\mathcal{N}(\boldsymbol{\mu}, \Sigma) \,\|\, \mathcal{N}(\boldsymbol{\mu}_0, \Sigma_0)\right) + \log\frac{2n}{\delta}}{n-1}. \qquad (25)\end{aligned}$$

The theorem states that the average generalization error diverges from the average training error by no more than a quantity depending on the divergence between the posterior and prior distributions over weight vectors, where divergence is measured by the relative entropy. Thus, to guarantee a low generalization error, two quantities should be minimized: the training error $\ell(\mathcal{N}(\boldsymbol{\mu}, \Sigma), S)$ and the relative entropy between the posterior and prior distributions over weight vectors $\mathrm{D}_{\mathrm{KL}}(\mathcal{N}(\boldsymbol{\mu}, \Sigma) \,\|\, \mathcal{N}(\boldsymbol{\mu}_0, \Sigma_0))$.

Following McAllester (2003b), we can state the following somewhat "weaker but perhaps clearer statement".

Theorem 3 *Fix a prior distribution over weight vectors $\mathcal{N}(\boldsymbol{\mu}_0, \Sigma_0)$. Then, for any $\delta \in [0, 1]$, with probability at least $1 - \delta$ over samples $S = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^n$ of size n, for all posterior distributions $\mathcal{N}(\boldsymbol{\mu}, \Sigma)$ the following holds:*

$$\begin{aligned}\ell(\mathcal{N}(\boldsymbol{\mu}, \Sigma), \mathcal{D}) &\leq C_1 \frac{1}{n}\sum_{i=1}^n \Phi\left(-\frac{\mu_i}{\sigma_i}\right) \\ &+ C_2 \frac{\mathrm{D}_{\mathrm{KL}}\left(\mathcal{N}(\boldsymbol{\mu}, \Sigma) \,\|\, \mathcal{N}(\boldsymbol{\mu}_0, \Sigma_0)\right) + \log\frac{2n}{\delta}}{n-1} \ , \qquad (26)\end{aligned}$$

where $\mu_i = y_i(\boldsymbol{\mu}\cdot\boldsymbol{x}_i)$, $\sigma_i = \sqrt{\boldsymbol{x}_i^\top \Sigma \boldsymbol{x}_i}$, $C_1 = 1 + \sqrt{2}/2 \approx 1.7$, and $C_2 = 2 + \sqrt{2}/2 \approx 2.7$.

Proof: We give a more explicit expression of the bound of Theorem 2. Following McAllester (2003b), we note that for $q > p$, $\mathrm{D}_{\mathrm{KL}}(p\|q) \leq x$ implies $q < p + \sqrt{2px} + 2x$. Using the inequality $\sqrt{px} \leq \frac{1}{2}(p + x)$, we obtain: $q \leq (1 + \sqrt{2}/2)p + (2 + \sqrt{2}/2)x = C_1\, p + C_2\, x$.

To conclude the proof we observe that by the definition of $\ell(\mathcal{N}(\boldsymbol{\mu}, \Sigma), S)$, the following holds:

$$\begin{aligned}\ell(\mathcal{N}(\boldsymbol{\mu}, \Sigma), S) &= \frac{1}{n}\sum_{i=1}^n \ell(\mathcal{N}(\boldsymbol{\mu}, \Sigma), (\boldsymbol{x}_i, y_i)) \\ = \ \frac{1}{n}\sum_{i=1}^n \Pr\left[y_i(\boldsymbol{w}\cdot\boldsymbol{x}_i) \leq 0\right] &= \frac{1}{n}\sum_{i=1}^n \Phi\left(-\frac{\mu_i}{\sigma_i}\right) \ .\end{aligned}$$

This last equality was established earlier in Section 2.1 to formulate the GMMs optimization problem in the separable case. ■

The next result states our first generalization bound for the performance of the GMMs classifier in the separable case.

Corollary 4 *Fix a distribution over weight vectors $\mathcal{N}(\mathbf{0}, \mathbf{I})$. Then, for any $\delta \in [0, 1]$, with probability at least $1 - \delta$ over the choice of a sample S of size n, the following bound holds simultaneously for all distributions $\mathcal{N}(\boldsymbol{\mu}, \Sigma)$ that satisfy $\Pr_{\boldsymbol{w}\sim\mathcal{N}(\boldsymbol{\mu},\Sigma)}[y_i(\boldsymbol{w}\cdot\boldsymbol{x}_i) \geq 0] \geq \eta$ for some $\eta \in (0.5, 1]$:*

$$\begin{aligned}&\ell(\mathcal{N}(\boldsymbol{\mu}, \Sigma), \mathcal{D}) \leq C_1(1 - \eta) \\ &+ C_2 \frac{\frac{1}{2}\left(-\log(\det\Sigma) + \mathrm{Tr}(\Sigma) + \|\boldsymbol{\mu}\|^2 - d\right) + \log\frac{2n}{\delta}}{n-1} \ .\end{aligned}$$

Proof: The result follows from Theorem 3. By assumption,

$$\Pr\left[y_i(\boldsymbol{w}\cdot\boldsymbol{x}_i) \leq 0\right] = \Phi\left(-\frac{\mu_i}{\sigma_i}\right) \leq (1 - \eta) \ .$$

Using this inequality to bound the first term of the right-hand side of the bound of Theorem 3 and identity (10) to give an explicit expression of the relative entropy between the posterior $\mathcal{N}(\boldsymbol{\mu}, \Sigma)$ and the prior $\mathcal{N}(\mathbf{0}; \mathbf{I})$ in the second term yields directly the statement of the corollary. ■

In the separable case, the GMMs optimization problem (11) precisely consists of minimizing the bound on the generalization error given by Corollary 4. Thus, the corollary gives a strong justification for our algorithm in that case. A similar analysis holds in the general case of non-separable training samples.

Corollary 5 *Fix a distribution over weight vectors $\mathcal{N}(\mathbf{0}, \mathbf{I})$ and let ϕ denote $\phi = \Phi^{-1}(\eta)$. Then, for any $\delta \in [0, 1]$, with probability at least $1 - \delta$ over the choice of a sample $S = ((\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_n, y_n))$ of size n, the following bound holds simultaneously for all distributions*

$\mathcal{N}(\boldsymbol{\mu},\Sigma)$ and all values of $\eta \in (0.5, 1]$:

$$\ell\left(\mathcal{N}(\boldsymbol{\mu},\Sigma),\mathcal{D}\right) \leq \frac{C_1}{n}\sum_{i=1}^{n}\Phi\Big(-\phi+\max\Big\{\phi-\frac{\mu_i}{\sigma_i},0\Big\}\Big)$$
$$+C_2\,\frac{\frac{1}{2}\Big(-\log\left(\det\Sigma\right)+\mathrm{Tr}\left(\Sigma\right)+\left\|\boldsymbol{\mu}\right\|^2-d\Big)+\log\frac{2n}{\delta}}{n-1}\ .$$

Proof: The corollary follows from Theorem 3. The relative entropy appearing in the right-hand side of the bound of Theorem 3 can be replaced by a more explicit expression as in Corollary 4. The first term of the right-hand side of the bound of Theorem 3 can be bounded using $-x \leq -y + \max\{y-x, 0\}$ and the fact that Φ is monotonically increasing. This yields the statement of the corollary. ■

As in the separable case, Corollary 5 provides a theoretical justification for the GMMs algorithm in the non-separable case. Indeed, by definition of the ξ_is in the optimization problem (13) for GMMs, $\xi_i = \max\left\{\left(\phi\sigma_i - \mu_i\right)/D_i, 0\right\}$. Thus, if we set $D_i = \sigma_i, i = 1, \ldots, n$, the algorithm can be viewed as minimizing a monotonic function of the bound since for our choice of D_i, $\Phi\Big(-\phi+\max\Big\{\phi-\frac{\mu_i}{\sigma_i},0\Big\}\Big) = \Phi\left(-\phi+\xi_i\right)$.

Note however that σ_i is a function of the optimal solution $\boldsymbol{\mu}$ and Σ and thus can not be set in advance. Also, replacing the scale parameters D_i with $\sigma_i = \sqrt{\boldsymbol{x}_i^\top\Sigma\boldsymbol{x}_i}$ in (13) leads to a non-convex optimization problem. In the next section, we present results of experiments in which we simply set $D_i = 1$. This choice may not be optimal, yet it allows us to avoid algorithmic complexities arising from non-convexity.

5 Alternative View

The GMMs learning algorithms of Sec. 2 were motivated by a generalized maximum entropy principle. However, a similar optimization problem can be derived starting from the standard optimization problem of SVMs. In the separable case, the QP problem for SVMs is the following (Boser *et al.*, 1992):

$$\min_{\boldsymbol{w}}\frac{1}{2}\left\|\boldsymbol{w}\right\|^2 \qquad \text{s.t.} \qquad y_i(\boldsymbol{w}\cdot\boldsymbol{x}_i) \geq 1 \text{ for } i=1,\ldots,n\ . \tag{27}$$

To obtain a robust formulation we can replace the single weight vector $\boldsymbol{w}$ with a Gaussian distribution over weight vectors $\boldsymbol{w} \sim \mathcal{N}(\boldsymbol{\mu},\Sigma)$, and the objective function and constraints with their probabilistic counterparts.

The inequality constraints of the SVM optimization problem (27) are thus replaced with the requirement that the inequality hold with probability at least η, that is $\Pr\left[y_i(\boldsymbol{w}\cdot\boldsymbol{x}_i) \geq 1\right] \geq \eta$. This inequality can be equivalently rewritten as follows, as in Section 2.1:

$$y_i(\boldsymbol{\mu}\cdot\boldsymbol{x}_i) \geq 1 + \phi\sqrt{\boldsymbol{x}_i^\top\Sigma\boldsymbol{x}_i}\ , \tag{28}$$

where, as in (9), $\eta = \Phi(\phi)$.

The objective function can be replaced with its expectation $\mathrm{E}\left[\left\|\boldsymbol{w}\right\|^2\right]$. This however is not sufficient since the solution could then be trivially $\Sigma = 0$. Instead, we can subtract from the objective a term proportional to the entropy, to ensure that the entropy of the optimal solution is non-zero. The new objective function is thus:

$$-\mathcal{H}\left[\mathcal{N}(\boldsymbol{\mu},\Sigma)\right] + \frac{A}{2}\mathrm{E}\left[\left\|\boldsymbol{w}\right\|^2\right] =$$
$$-\frac{1}{2}d\log(2\pi) - \frac{1}{2}\log\det\Sigma + \frac{A}{2}\Big(\mathrm{Tr}\left(\Sigma\right)+\left\|\boldsymbol{\mu}\right\|^2\Big)\ .$$

Omitting additive constants and relaxing the constraints ((Cortes and Vapnik, 1995)) we obtain the following robust version of SVMs:

$$\begin{aligned}\min_{\boldsymbol{\mu},\Sigma}\ & -\frac{1}{2}\log\det\Sigma + \frac{A}{2}\Big(\mathrm{Tr}\left(\Sigma\right)+\left\|\boldsymbol{\mu}\right\|^2\Big) + C\sum_i\xi_i \\ \text{s.t.}\ & y_i(\boldsymbol{\mu}\cdot\boldsymbol{x}_i) \geq 1-\xi_i+\phi\sqrt{\boldsymbol{x}_i^\top\Sigma\boldsymbol{x}_i} \quad i=1\ldots n \\ & \Sigma \succeq 0\ .\end{aligned} \tag{29}$$

The comparison of this optimization problem (29) and the GMMs optimization problem (11) shows that that the objectives of the two optimization problems coincide for $A = 1/a$. However, the constraints of the problem (11) are homogeneous while those of (29) are not because of the additional term 1. As a result, the three hyperparameters A, C and ϕ cannot be reduced to two, unlike what was done in deriving (13) from (12).

6 Experiments

We implemented in matlab a Hildreth-like algorithm (Censor and Zenios, 1997) to solve (13) in the case where $D_i = 1$ for all i, which is then a well defined convex optimization problem both in the separable and the non-separable cases. Our algorithm iterates over the training points and for each point updates the parameters to classify that point optimally. Each iteration requires $O(d^2)$ time to access the covariance matrix.

We evaluated our algorithm using the USPS handwritten digits dataset. The training set contained 7,291 training examples and the test set 2,007 examples. Originally, each instance represented an image of size 16×16 pixels of a digit, with ten possible digits. Due to our preliminary implementation's limitations, we reduced the dimensionality of the data by replacing each four adjacent pixels with their mean, which resulted in image size of 8×8, thereby reducing the dimensionality from 256 to 64. We repeated the following process over all 45 pairs of digits 10 times: for each pair, we randomly selected 100 examples which were associated with one of the two digits of the current pair, the remaining training examples associated with the pair were

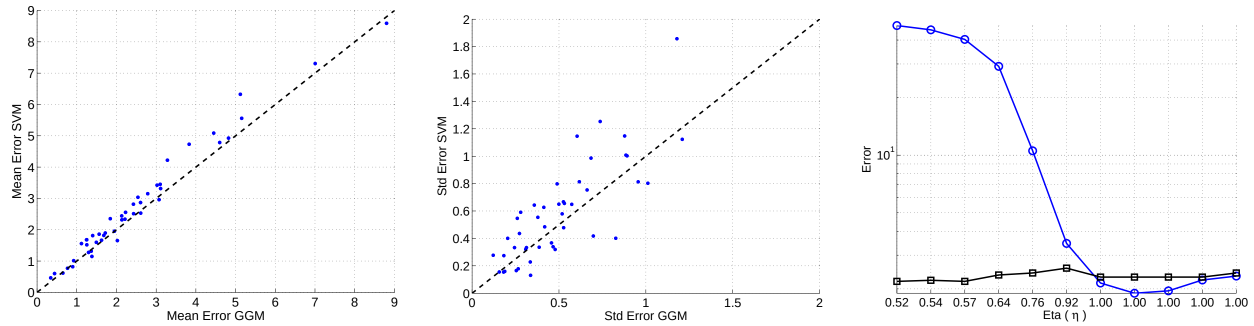

Figure 2: Average (left) and standard deviation (middle) test error (×100) of GMMs (x-axis) vs SVMs (y-axis) for 45 label-pairs of the USPS dataset. A point above the line $y = x$ indicates better performance for the GMM algorithm. Right: Average test error (×100) of GMMs (y-axis) using the mean predictor $\text{sign}\,(\boldsymbol{\mu} \cdot \boldsymbol{x})$ (black squares) and the Gibbs predictor $\Pr\left[y \neq \text{sign}\,(\boldsymbol{w} \cdot \boldsymbol{x})\right]$ (blue circles) as functions of η for 3 *vs.* 8 discrimination.

used as a validation set. The test set was the standard USPS test set restricted to the relevant two digits.

We trained two algorithms: support vector machines (SVMs) and Gaussian Margin Machines (GMMs). For SVMs, we experimented with 9 different values of the regularization parameter C, and for the GMMs with 11 values for ϕ and 12 for the regularization parameter C. We trained each of the algorithms using all these parameter values and selected the model with the minimal error over the validation set. We then used that model to compute the error over the test set and averaged the results over the 10 repeats.

The left panel of Fig. 2 shows the results for both SVMs and the GMMs. Each point corresponds to one of the 45 binary classification problems. A point above the line $y = x$ corresponds to a pair where GMMs performs better than SVMs, and vise-versa. GMMs outperforms SVMs since 36 of the points are above the line $y = x$ and 9 points below. We also evaluated the robustness of each method by computing the standard deviation of the test error over the 10 repeats. The results are summarized in the middle panel of Fig. 2. GMMs seem to be more robust as 28 points are above the line $y = x$ while 17 are below.

The right panel of Fig. 2 shows the results of our empirical study of the effect of the parameter η on performance when using for prediction the mean $\text{sign}\,(\boldsymbol{\mu} \cdot \boldsymbol{x})$ (bottom black line with squares) or the averaged Gibbs prediction $\Pr\left[y \neq \text{sign}\,(\boldsymbol{w} \cdot \boldsymbol{x})\right]$ (top blue line with circles). Interestingly, the minimal error of the the Gibbs predictor is reached for η close to 1 and its error is close to .5 when η is close to .5. For the mean predictor $\text{sign}\,(\boldsymbol{\mu} \cdot \boldsymbol{x})$, the error values are within a smaller range, with the smallest error attained for a small value of η. These observations apply also to other digit pairs, with the optimal setting for all tests being $\eta = 0.54$.

7 Related Work

The work presented here bears some similarity with that of Jaakkola *et al.* on maximum entropy discrimination (Jaakkola *et al.*, 1999) in which they propose a training approach that maximizes the relative entropy between a prior distribution over the parameters and some given distribution. However, in that work, both the prior distribution and the learned distribution over weight vectors are Gaussian distributions with fixed covariance matrices, while within our formulation the covariance of the distributions is also learned. Jaakkola *et al.* further propose to make a prediction by taking the sign of the average "margin", $\text{sign}\,\mathrm{E}\left[\boldsymbol{w} \cdot \boldsymbol{x}\right]$, while we propose to use the probability of error, effectively replacing the sign operator with the expectation. Jaakkola *et al.* define a distribution over "margin" variables as well. Our method does not provide an explicit notion of margin, instead that stands out as a byproduct of our derivations. Finally, the dual form of their algorithm (Jaakkola *et al.*, 1999, Theorem 2) is very similar to the SVMs dual, with the addition of an extra term to the objective. The dual form of our algorithm is more involved, and finding a simple useful equivalent is still an open problem.

Other previous work related to this topic typically assumes a Gaussian or uniform distribution over the input data rather than over the classifiers. Lanckriet *et al.* (2002) assume that the points associated with each of the classes are distributed according to a class-dependent Gaussian distribution. Nath *et al.* (2006) use a clustering technique to group data points, and then optimize an SVM-like criterion such that a large fraction of the points of each cluster be classified correctly. (Bi and Zhang, 2004) assume a uniform-isotropic noise over input vectors, and modify SVMs to classify well the worst noise-instance per input vector. Shivaswamy and Jebara (2007) use a geometric motivation to modify SVMs. That effort and other related ones prepare first the additional knowledge about the problem (specific covariance matrix of the input data (Shivaswamy and Jebara, 2007), per class covariance matrix (Lanckriet *et al.*, 2002; Nath *et al.*, 2006), or per-point noise level (Bi and Zhang, 2004)), and keep it fixed during learning. In contrast, our method learns together the classifier and the additional information.

8 Conclusion

We proposed a new form of linear classifier that extends the commonly used large-margin linear classifiers to probability distributions over weight vectors. Our learning algorithm is based on a probabilistic large-margin requirement and the maximum entropy principle and benefits from strong theoretical guarantees based on tight PAC-Bayesian generalization bounds.

The preliminary empirical evaluation presented shows that our method not only performs favorably with respect to SVMs, but also that it succeeds indeed in constructing a robust classifier with reduced variance. Future larger-scale implementations of our algorithms will help us explore its properties when applied to a variety of tasks and data sets.

References

J. Bi and T. Zhang. Support vector classification with input data uncertainty. In *NIPS*, 2004.

B. E. Boser, I. M. Guyon, and V. N. Vapnik. A training algorithm for optimal margin classifiers. In *COLT*, 1992.

Y. Censor and S.A. Zenios. *Parallel Optimization: Theory, Algorithms, and Applications*. Oxford University Press, New York, NY, USA, 1997.

C. Cortes and V. Vapnik. Support-vector networks. *Machine Learning*, 20(3):273–297, September 1995.

R. Herbrich, T. Graepel, and C. Campbell. Robust Bayes point machines. In *ESANN 2000*, pages 49–54, 2000.

R. Herbrich, T. Graepel, and C. Campbell. Bayes point machines. *JMLR*, 1:245–279, 2001.

T. Jaakkola and M. Jordan. A variational approach to Bayesian logistic regression models and their extensions. In *Workshop on Artificial Intelligence and Statistics*, 1997.

T. Jaakkola, M. Meila, and T. Jebara. Maximum entropy discrimination. In *NIPS 12*, 1999.

G. Lanckriet, L. Ghaoui, C. Bhattacharyya, and M. Jordan. A robust minimax approach to classification. *JMLR*, 3:555–582, 2002.

J. Langford and M. Seeger. Bounds for averaging classifiers, 2002.

J. Langford and J. Shawe-Taylor. PAC-bayes and margins. In *Neural Information Processing Systems (NIPS)*, 2002.

D. McAllester. PAC-Bayesian model averaging. In *Proceedings of COLT*, 1999.

D. McAllester. PAC-Bayesian stochastics model selection. *Machine Learning Journal*, 5:5–21, 2003.

D. McAllester. Simplified PAC-Bayesian margin bounds. In *Proceedings of COLT*, 2003.

J. Nath, C. Bhattacharyya, and M. Murty. Clustering based large margin classification: A scalable approach using SOCP formulation. In *KDD*, 2006.

M. Seeger. PAC-Bayesian generalization bounds for gaussian processes. *JMLR*, 3:233–269, 2002.

P. Shivaswamy and T. Jebara. Ellipsoidal kernel machines. In *Artificial Intelligence and Statistics (AISTATS)*, 2007.

K. Tsuda, G. Rätsch, and M.K. Warmuth. Matrix exponentiated gradient updates for on-line learning and Bregman projection. *JMLR*, 6:995–1018, 2005.

Appendix: Derivation of the Dual Problem

We start from (22) and supporting definitions. Equation (20) can be rewritten in matrix notation as follows: $\Upsilon^{-2} = \psi\mathbf{I} + \mathbf{X}\mathbf{B}\mathbf{X}^\top$. Thus, by the matrix inversion identity, Υ^2 can be written as

$$\Upsilon^2 = \frac{1}{\psi}\big[\mathbf{I} - \mathbf{X}\big((\psi\mathbf{B})^{-1} + \mathbf{X}^\top\mathbf{X}\big)^{-1}\mathbf{X}^\top\big]. \tag{30}$$

In view of (16), (20), (22), and this equation, the optimization problem (15) can be rewritten as

$$\begin{aligned}\mathcal{L} = &\log\det\left(\psi\mathbf{I} + \mathbf{X}\mathbf{B}\mathbf{X}^\top\right)\\ &+ \frac{\psi}{2}\mathrm{Tr}\Big[\frac{1}{\psi}\big(\mathbf{I} - \mathbf{X}\left((\psi\mathbf{B})^{-1} + \mathbf{X}^\top\mathbf{X}\right)^{-1}\mathbf{X}^\top\big)\Big]\\ &+ \sum_{i=1}^n \alpha_i\sqrt{v_i} + \frac{1}{2}\sum_{i,j}\alpha_i\alpha_j y_i y_j(\boldsymbol{x}_i\cdot\boldsymbol{x}_j)\\ &- \sum_{i,j}\alpha_i\alpha_j y_i y_j(\boldsymbol{x}_i\cdot\boldsymbol{x}_j)\ .\end{aligned}$$

Plugging in β_i and the v_i from (22) and removing additive constants, the Lagrangian is given by:

$$\begin{aligned}&\log\det\left(\psi\mathbf{I} + \mathbf{X}\mathbf{B}\mathbf{X}^\top\right) - \frac{1}{2}\mathrm{Tr}\big[\left((\psi\mathbf{B})\right)^{-1} + \mathbf{K}^{-1}\mathbf{K}\big]\\ &+ \sum_{i=1}^n \beta_i v_i - \frac{1}{2}\sum_{i,j}\beta_i\beta_j\sqrt{v_i}\sqrt{v_j}y_i y_j\mathbf{K}_{i,j}\ .\end{aligned} \tag{31}$$

For each example i, the variance v_i can be rewritten as follows:

$$\begin{aligned}v_i &= \boldsymbol{x}_i^\top\Upsilon^2\boldsymbol{x}_i\\ &= \frac{1}{\psi}\boldsymbol{x}_i^\top\Big(\mathbf{I} - \mathbf{X}\big(\,(\psi\mathbf{B})^{-1} + \mathbf{X}^\top\mathbf{X}\big)^{-1}\mathbf{X}^\top\Big)\boldsymbol{x}_i^\top\\ &= \frac{1}{\psi}\Big(\mathbf{K}_{i,i} - \boldsymbol{k}_i^\top\big(\,(\psi\mathbf{B})^{-1} + \mathbf{K}\big)^{-1}\boldsymbol{k}_i\Big)\ ,\end{aligned}$$

that is $v_i = v_i\left(\beta_1,\ldots,\beta_n\right) = v_i\left(\mathbf{B}\right)$. Since $\det(\mathbf{I} + \mathbf{A}^\top\mathbf{A}) = \det(\mathbf{I} + \mathbf{A}\mathbf{A}^\top)$, we get:

$$\begin{aligned}&\log\det\left(\psi\mathbf{I} + \mathbf{X}\mathbf{B}\mathbf{X}^\top\right)\\ &\qquad = \log\det\left(\psi\mathbf{I} + (\sqrt{\mathbf{B}}\mathbf{X}^\top)(\mathbf{X}\sqrt{\mathbf{B}})\right)\\ &\qquad = \log\det\left(\psi\mathbf{I} + \sqrt{\mathbf{B}}\mathbf{K}\sqrt{\mathbf{B}}\right)\ .\end{aligned} \tag{32}$$

Using this identity and substituting the expression for v_i in (31) yields (23).