Journal of Machine Learning Research 7 (2006) 1713–1741 Submitted 10/05; Revised 7/06; Published 8/06

Considering Cost Asymmetry in Learning Classifiers

Francis R. Bach FRANCIS.BACH@MINES.ORG
Centre de Morphologie Mathématique
Ecole des Mines de Paris
35, rue Saint Honoré
77300 Fontainebleau, France

David Heckerman HECKERMAN@MICROSOFT.COM
Eric Horvitz HORVITZ@MICROSOFT.COM
Microsoft Research
Redmond, WA 98052, USA

Editor: John Shawe-Taylor

Abstract

Receiver Operating Characteristic (ROC) curves are a standard way to display the performance of a set of binary classifiers for all feasible ratios of the costs associated with false positives and false negatives. For linear classifiers, the set of classifiers is typically obtained by training once, holding constant the estimated slope and then varying the intercept to obtain a parameterized set of classifiers whose performances can be plotted in the ROC plane. We consider the alternative of varying the asymmetry of the cost function used for training. We show that the ROC curve obtained by varying both the intercept and the asymmetry, and hence the slope, always outperforms the ROC curve obtained by varying only the intercept. In addition, we present a path-following algorithm for the support vector machine (SVM) that can compute efficiently the entire ROC curve, and that has the same computational complexity as training a single classifier. Finally, we provide a theoretical analysis of the relationship between the asymmetric cost model assumed when training a classifier and the cost model assumed in applying the classifier. In particular, we show that the mismatch between the step function used for testing and its convex upper bounds, usually used for training, leads to a provable and quantifiable difference around extreme asymmetries.

Keywords: support vector machines, receiver operating characteristic (ROC) analysis, linear classification

1. Introduction

Receiver Operating Characteristic (ROC) analysis has seen increasing attention in the recent statistics and machine-learning literature (Pepe, 2000; Provost and Fawcett, 2001; Flach, 2003). The ROC is a representation of choice for displaying the performance of a classifier when the costs assigned by end users to false positives and false negatives are not known at the time of training. For example, when training a classifier for identifying cases of undesirable unsolicited email, end users may have different preferences about the likelihood of a false negative and false positive. The ROC curve for such a classifier reveals the ratio of false negatives and positives at different probability thresholds for classifying an email message as unsolicited or normal email.

In this paper, we consider linear binary classification of points in an Euclidean space—noting that it can be extended in a straightforward manner to non-linear classification problems by using

Mercer kernels (Schölkopf and Smola, 2002). That is, given data $x \in \mathbb{R}^d$, $d \geqslant 1$, we consider classifiers of the form $f(x) = \text{sign}(w^\top x + b)$, where $w \in \mathbb{R}^d$ and $b \in \mathbb{R}$ are referred to as the *slope* and the *intercept*. To date, ROC curves have been usually constructed by training once, holding constant the estimated slope and varying the intercept to obtain the curve. In this paper, we show that, while that procedure appears to be the most practical thing to do, it may lead to classifiers with poor performance in some parts of the ROC curve.

The crux of our approach is that we allow the asymmetry of the cost function to vary, that is, we vary the ratio of the cost of a false positive and the cost of a false negative. For each value of the ratio, we obtain a different slope and intercept, each optimized for this ratio. In a naive implementation, varying the asymmetry would require a retraining of the classifier for each point of the ROC curve, which would be computationally expensive. In Section 3.1, we present an algorithm that can compute the solution of a support vector machine (SVM) (see, for example, Schölkopf and Smola, 2002; Shawe-Taylor and Cristianini, 2004) for all possible costs of false positives and false negatives, with the same computational complexity as obtaining the solution for only one cost function. The algorithm extends to asymmetric costs the algorithm of Hastie et al. (2005) and is based on path-following techniques that take advantage of the piecewise linearity of the path of optimal solutions. In Section 3.2, we show how the path-following algorithm can be used to obtain ROC curves. In particular, by allowing both the asymmetry and the intercept to vary, we can obtain better ROC curves than by methods that simply vary the intercept.

In Section 4, we provide a theoretical analysis of the relationship between the asymmetry of costs assumed in training a classifier and the asymmetry desired in its application. In particular, we show that, even in the population (*i.e.*, infinite sample) case, the use of a training loss function which is a convex upper bound on the true or testing loss function (a step function) creates classifiers with sub-optimal accuracy. We quantify this problem around extreme asymmetries for several classical convex-upper-bound loss functions, including the square loss and the *erf loss*, an approximation of the logistic loss based on normal cumulative distribution functions (also referred to as the "error function," and usually abbreviated as erf). The analysis is carried through for Gaussian and mixture of Gaussian class-conditional distributions (see Section 4 for more details). The main result of this analysis is that given an extreme user-defined testing asymmetry, the training asymmetry should almost always be chosen to be less extreme.

As we shall see, the consequences of the potential mismatch between the cost functions assumed in testing versus training underscore the value of using the algorithm that we introduce in Section 4.3. Even when costs are known (*i.e.*, when only one point on the ROC curve is needed), the classifier resulting from our approach, which builds the entire ROC curve, is never less accurate and can be more accurate than one trained with the known costs using a convex-upper-bound loss function. Indeed, we show in Section 4.3 that computing the entire ROC curve using our algorithm can lead to substantial gains over simply training once.

The paper is organized as follows: In Section 2, we give an introduction to the linear classification problem and review the ROC framework. Section 3 contains the algorithmic part of the paper, while Section 4 provide a theoretical analysis of the discrepancy between testing and training asymmetries, together with empirical results. This paper is an extended version of previously published work (Bach et al., 2005a).

2. Problem Overview

Given data $x \in \mathbb{R}^d$ and labels $y \in \{-1,1\}$, we consider linear classifiers of the form $f(x) = \text{sign}(w^\top x + b)$, where w is the *slope* of the classifier and b the *intercept*. A classifier is determined by the parameters $(w,b) \in \mathbb{R}^{d+1}$. In Section 2.1, we introduce notation and definitions; in Section 2.2, we lay out the necessary concepts of ROC analysis, while in Section 2.3, we describe how these classifiers and ROC curves are typically obtained from data.

2.1 Asymmetric Cost and Loss Functions

Positive (resp. negative) examples are those for which $y = 1$ (resp. $y = -1$). The two types of misclassification, false positives and false negatives, are assigned two different costs. We let C_+ denote the cost of a false negative and C_- the cost of a false positive. The total *expected* cost is equal to

$$R(C_+, C_-, w, b) = C_+ P\{w^\top x + b < 0,\ y = 1\} + C_- P\{w^\top x + b \geqslant 0,\ y = -1\}.$$

In the context of large margin classifiers (see, for example, Bartlett et al., 2004), the expected cost is usually expressed in terms of the *0–1 loss function*; indeed, if we let $\phi_{0-1}(u) = 1_{u<0}$ be the 0–1 loss, we can write the expected cost as

$$R(C_+, C_-, w, b) = C_+ E\{1_{y=1}\phi_{0-1}(w^\top x + b)\} + C_- E\{1_{y=-1}\phi_{0-1}(-w^\top x - b)\},$$

where E denotes the expectation with respect to the joint distribution of (x,y).

The expected cost defined using the 0–1 loss is the cost that end users are usually interested in during the use of the classifier, while the other cost functions that we define below are used solely for training purposes. The convexity of these cost functions makes learning algorithms convergent without local minima, and leads to attractive asymptotic properties (Bartlett et al., 2004).

A traditional set-up for learning linear classifiers from labeled data is to consider a convex upper bound ϕ on the 0–1 loss ϕ_{0-1}, and to use the expected ϕ-cost:

$$R_\phi(C_+, C_-, w, b) = C_+ E\{1_{y=1}\phi(w^\top x + b)\} + C_- E\{1_{y=-1}\phi(-w^\top x - b)\}.$$

We refer to the ratio $C_+/(C_- + C_+)$ as the *asymmetry*. We shall use *training asymmetry* to refer to the asymmetry used for training a classifier using a ϕ-cost, and the *testing asymmetry* to refer to the asymmetric cost characterizing the testing situation (reflecting end user preferences) with the actual cost based on the 0–1 loss. In Section 4, we will show that these may be different in the general case.

We shall consider several common loss functions. Some of the loss functions (square loss, hinge loss) lead to attractive computational properties, while others (square loss, erf loss) are more amenable to theoretical manipulations (see Figure 1 for the plot of the loss functions, as they are commonly used and defined below[1]):

- **square loss** : $\phi_{sq}(u) = \frac{1}{2}(u-1)^2$; the classifier is equivalent to linear regression on y,
- **hinge loss** : $\phi_{hi}(u) = \max\{1-u, 0\}$; the classifier is the support vector machine (Shawe-Taylor and Cristianini, 2004),

1. Note that by rescaling, each of these loss functions can be made to be an upper bound on the 0–1 loss which is tight at zero.

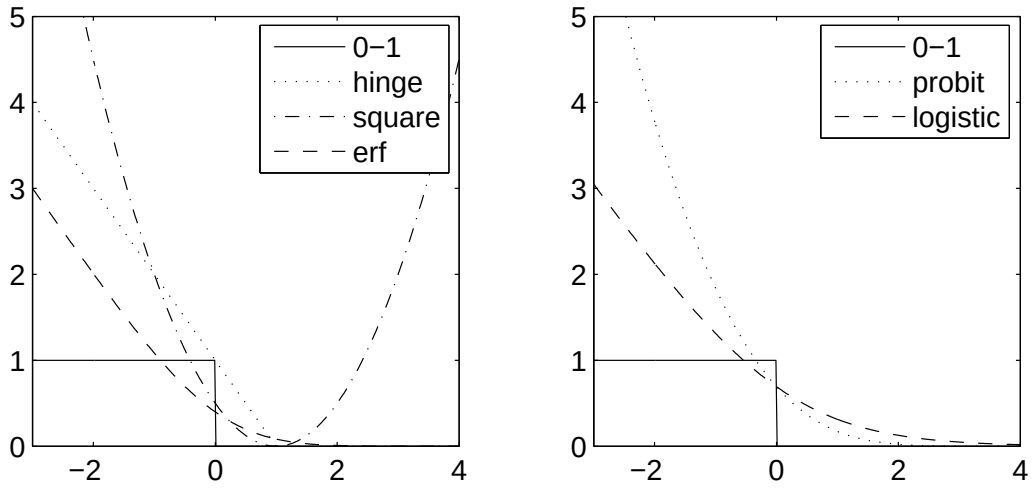

Figure 1: Loss functions. Left: plain: 0–1 loss; dotted: hinge loss, dashed: erf loss, dash-dotted: square loss. Right: plain: 0–1 loss, dotted: probit loss, dashed: logistic loss.

- **erf loss** : $\phi_{erf}(u) = [u\psi(u) - u + \psi'(u)]$, where ψ is the cumulative distribution of the standard normal distribution, that is : $\psi(v) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{v} e^{-t^2/2} dt$, and $\psi'(v) = \frac{1}{\sqrt{2\pi}} e^{-v^2/2}$. The erf loss can be used to provide a good approximation of the *logistic loss* $\phi_{log}(u) = \log(1+e^{-u})$ as well as its derivative, and is amenable to closed-form computations for Gaussians and mixture of Gaussians (see Section 4 for more details). Note that the erf loss is different from the *probit loss* $-\log\psi(u)$, which leads to probit regression (Hastie et al., 2001).

2.2 ROC Analysis

The aim of ROC analysis is to display in a single graph the performance of classifiers for all possible costs of misclassification. In this paper, we consider sets of classifiers $f_\gamma(x)$, parameterized by a variable $\gamma \in \mathbb{R}$ (γ can either be the intercept or the training asymmetry).

For a classifier $f(x)$, we can define a point (u,v) in the "ROC plane," where u is the proportion of false positives $u = P(f(x) = 1 | y = -1)$, and v is the proportion of true positives $v = P(f(x) = 1 | y = 1)$. When γ is varied, we obtain a curve in the ROC plane, the ROC curve (see Figure 2 for an example). Whether γ is the intercept or the training asymmetry, the ROC curve always passes through the point $(0,0)$ and $(1,1)$, which corresponds to classifying all points as negative (resp. positive).

The *upper convex envelope* of the curve is the set of optimal ROC points that can be achieved by the set of classifiers; indeed, if a point in the envelope is not one of the original points, it must lie in a segment between two points $(u(\gamma_0), v(\gamma_0))$ and $(u(\gamma_1), v(\gamma_1))$, and all points in a segment between two classifiers can always be attained by choosing randomly between the two classifiers. Note that this classifier itself is not a linear classifier; its performance, defined by given true positive and false positive rates, can only be achieved by a mixture of two linear classifiers. However, if the user is

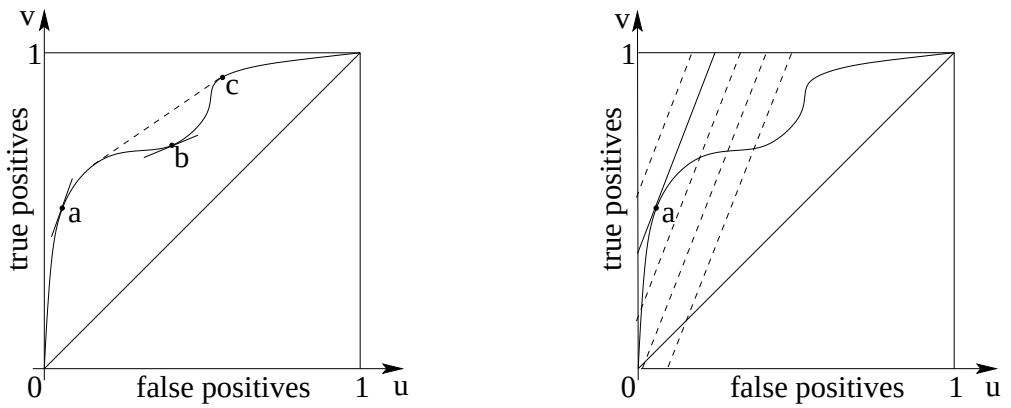

Figure 2: Left: ROC curve: (plain) regular ROC curve; (dashed) convex envelope. The points *a* and *c* are ROC-consistent and the point *b* is ROC-inconsistent. Right: ROC curve and dashed equi-cost lines: All lines have direction (p_+C+, p_-C_-), the plain line is optimal and the point "a" is the optimal classifier.

only interested in the testing cost, which is a weighted linear combination of the true positive and false positive rates, the lowest testing cost is always achieved by both of these two classifiers (see Section 4.3 for an example of such a situation).

Denoting $p_+ = P(y=1)$ and $p_- = P(y=-1)$, the expected (C_+, C_-)-cost for a classifier (u,v) in the ROC space, is simply $p_+C_+(1-v) + p_-C_-u$, and thus optimal classifiers for the (C_+, C_-)-cost can be found by looking at lines of slope that are normal to the direction $(p_-C_-, -p_+C_+)$, which intersects the ROC curve and are as close as the point $(0,1)$ as possible (see Figure 2).

A point $(u(\gamma), v(\gamma))$ is said to be *ROC-consistent* if it lies on the upper convex envelope; In this case, the tangent direction $(du/d\gamma, dv/d\gamma)$ defines a cost $(C_+(\gamma), C_-(\gamma))$ for which the classifier is optimal (for the testing cost, which is defined using the 0–1 loss). The condition introduced earlier, namely that $(du/d\gamma, dv/d\gamma)$ is normal to the direction $(p_-C_-, -p_+C_+)$, leads to:

$$p_-C_-\frac{du}{d\gamma}(\gamma) - p_+C_+\frac{dv}{d\gamma}(\gamma) = 0.$$

The *optimal testing asymmetry* $\beta(\gamma)$ defined as the ratio $\frac{C_+(\gamma)}{C_+(\gamma)+C_-(\gamma)}$, is thus equal to

$$\beta(\gamma) \triangleq \frac{C_+(\gamma)}{C_+(\gamma)+C_-(\gamma)} = \frac{1}{1+\frac{p_+}{p_-}\frac{dv}{d\gamma}(\gamma)/\frac{du}{d\gamma}(\gamma)}. \quad (1)$$

If a point $(u(\gamma), v(\gamma))$ is ROC-inconsistent, then the quantity $\beta(\gamma)$ has no meaning, and such a classifier is generally useless, because, for all settings of the misclassification cost, that classifier can be outperformed by the other classifiers or a combination of classifiers. See Figure 2 for examples of ROC-consistent and ROC-inconsistent points.

In Section 4, we relate the optimal asymmetry of cost in the testing or eventual use of a classifier in the real world, to the asymmetry of cost used to train that classifier; in particular, we show that

they differ and quantify this difference for extreme asymmetries (*i.e.*, close to the corner points $(0,0)$ and $(1,1)$). This analysis highlights the value of generating the entire ROC curve, even when only one point is needed, as we will present in Section 4.3.

Handling ROC surfaces In this paper, we will also consider varying both the asymmetry of the cost function and the intercept, leading to a set of points in the ROC plane parameterized by two real values. Although the concept of ROC-consistency could easily be extended to ROC surfaces, for simplicity we do not consider it here. In all our experiments, those ROC surfaces are reduced to curves by computing their convex upper envelopes.

2.3 Learning From Data

Given n labeled data points (x_i,y_i), $i=1,\ldots,n$, the *empirical cost* is equal to

$$\widehat{R}(C_+,C_-,w,b)=\frac{C_+}{n}\#\{y_i(w^\top x_i+b)<0,y_i=1\}+\frac{C_-}{n}\#\{y_i(w^\top x_i+b)<0,y_i=-1\},$$

where $\#A$ denotes the cardinality of the set A. The *empirical* ϕ*-cost* is equal to

$$\widehat{R}_\phi(C_+,C_-,w,b)=\frac{C_+}{n}\sum_{i\in I_+}\phi(y_i(w^\top x_i+b))+\frac{C_-}{n}\sum_{i\in I_-}\phi(y_i(w^\top x_i+b)),$$

where $I_+=\{i,y_i=1\}$ and $I_-=\{i,y_i=-1\}$. When learning a classifier from data, a classical setup is to minimize the sum of the *empirical* ϕ*-cost* and a regularization term $\frac{1}{2n}||w||^2$, that is, to minimize $\widehat{J}_\phi(C_+,C_-,w,b)=\widehat{R}_\phi(C_+,C_-,w,b)+\frac{1}{2n}||w||^2$.

Note that the objective function is no longer homogeneous in (C_+,C_-); the sum C_++C_- is referred to as the total amount of regularization. Thus, two end-user-defined parameters are needed to train a linear classifier: the *total amount of regularization* $C_++C_-\in\mathbb{R}^+$, and the *asymmetry* $\frac{C_+}{C_++C_-}\in[0,1]$. In Section 3.1, we show how the minimum of $\widehat{J}_\phi(C_+,C_-,w,b)$, with respect to w and b, can be computed efficiently for the hinge loss, for many values of (C_+,C_-), with a computational cost that is within a constant factor of the computational cost of obtaining a solution for one value of (C_+,C_-).

Building an ROC curve from data If a sufficiently large validation set is available, we can train on the training set and use the empirical distribution of the validation data to plot the ROC curve. If sufficient validation data is not available, then we can use several (typically 10 or 25) random splits of the data and average scores over those splits to obtain the ROC scores. We can also use this approach to obtain confidence intervals (Flach, 2003).

3. Building ROC Curves for the SVM

In this section, we present an algorithm to compute ROC curves for the SVM that explores the two-dimensional space of cost parameters (C_+,C_-) efficiently. We first show how to obtain optimal solutions of the SVM without solving the optimization problems many times for each value of (C_+,C_-). This method generalizes the results of Hastie et al. (2005) to the case of asymmetric cost functions. We then describe how the space (C_+,C_-) can be appropriately explored and how ROC curves can be constructed.

3.1 Building Paths of Classifiers

Given n data points x_i, $i = 1, \ldots, n$ which belong to $\mathbb{R}^d$, and n labels $y_i \in \{-1, 1\}$, minimizing the regularized empirical hinge loss is equivalent to solving the following convex optimization problem (Schölkopf and Smola, 2002):

$$\min_{w,b,\xi} \sum_i C_i \xi_i + \frac{1}{2}||w||^2 \quad \text{such that} \quad \forall i,\ \xi_i \geqslant 0,\ \xi_i \geqslant 1 - y_i(w^\top x_i + b),$$

where $C_i = C_+$ if $y_i = 1$ and $C_i = C_-$ if $y_i = -1$.

Optimality conditions and dual problems We know derive the usual Karush-Kuhn-Tucker (KKT) optimality conditions (Boyd and Vandenberghe, 2003). The Lagrangian of the problem is (with dual variables $\alpha, \beta \in \mathbb{R}^n_+$):

$$L(w, b, \xi, \alpha, \beta) = \sum_i C_i \xi_i + \frac{1}{2}||w||^2 + \sum_i \alpha_i (1 - y_i(w^\top x_i + b)) - \xi_i) - \sum_i \beta_i \xi_i.$$

The derivatives with respect to the primal variables are

$$\frac{\partial L}{\partial w} = w - \sum_i \alpha_i y_i x_i\ , \quad \frac{\partial L}{\partial b} = -\sum_i \alpha_i y_i\ , \quad \frac{\partial L}{\partial \xi_i} = C_i - \alpha_i - \beta_i\ ,$$

The first set of KKT conditions corresponds to the nullity of the Lagrangian derivatives with respect to the primal variables, that is:

$$w = \sum_i \alpha_i y_i x_i\ , \quad \sum_i \alpha_i y_i = \alpha^\top y = 0\ , \quad \forall i,\ C_i = \alpha_i + \beta_i. \tag{2}$$

The slackness conditions are

$$\forall i,\ \alpha_i(1 - \xi_i + y_i(w^\top x_i + b)) = 0 \ \text{ and } \ \beta_i \xi_i = 0. \tag{3}$$

Finally the dual problem can be obtained by computing the minimum of the Lagrangian with respect to the primal variables. If we let denote K the $n \times n$ Gram matrix of inner products, that is, defined by $K_{ij} = x_i^\top x_j$, and $\widetilde{K} = \mathrm{Diag}(y) K \mathrm{Diag}(y)$, the dual problem is:

$$\max_{\alpha \in \mathbb{R}^n} -\frac{1}{2}\alpha^\top \widetilde{K} \alpha + 1^\top \alpha \quad \text{such that} \quad \alpha^\top y = 0,\ \ \forall i,\ 0 \leqslant \alpha_i \leqslant C_i.$$

In the following, we only consider primal variables (w, b, ξ) and dual variables (α, β) that verify the first set of KKT conditions (2), which implies that w and β are directly determined by α.

Piecewise affine solutions Following Hastie et al. (2005), for an optimal set of primal-dual variables (w, b, α), we can separate data points in three disjoint sets, depending on the values of $y_i(w^\top x_i + b) = (\widetilde{K}\alpha)_i + b y_i$.

$$\begin{array}{lllll}
\text{Margin :} & \mathcal{M} & = \{i, & y_i(w^\top x_i + b) = 1 & \}, \\
\text{Left of margin :} & \mathcal{L} & = \{i, & y_i(w^\top x_i + b) < 1 & \}, \\
\text{Right of margin :} & \mathcal{R} & = \{i, & y_i(w^\top x_i + b) > 1 & \}.
\end{array}$$

Because of the slackness conditions (3), the sign of $y_i(w^\top x_i + b) - 1$ is linked with the location of α_i in the interval $[0, C_i]$. Indeed we have:

$$\begin{array}{lrcl} \text{Left of margin :} & i \in \mathcal{L} & \Rightarrow & \alpha_i = C_i, \\ \text{Right of margin :} & i \in \mathcal{R} & \Rightarrow & \alpha_i = 0. \end{array}$$

In the optimization literature, the sets $\mathcal{L}$, $\mathcal{R}$ and $\mathcal{M}$ are usually referred to as *active sets* (Boyd and Vandenberghe, 2003; Scheinberg, 2006). If the sets $\mathcal{M}$, $\mathcal{L}$ and $\mathcal{R}$ are known, then α, b are optimal if and only if:

$$\begin{array}{l} \forall i \in \mathcal{L}, \alpha_i = C_i \\ \forall i \in \mathcal{R}, \alpha_i = 0 \\ \forall i \in \mathcal{M}, (\widetilde{K}\alpha)_i + b y_i = 1 \\ \alpha^\top y = 0. \end{array}$$

This is a linear system with as many equations as unknowns (*i.e.*, $n+1$). The real unknowns (not clamped to C_i or 0) are $\alpha_{\mathcal{M}}$ and b, and the smaller system is (z_A denotes the vector z reduced to its components in the set A and $K_{A,B}$ denotes the submatrix of K with indices in A and B):

$$\begin{array}{rcl} \widetilde{K}_{\mathcal{M},\mathcal{M}} \alpha_{\mathcal{M}} + b y_{\mathcal{M}} & = & 1_{\mathcal{M}} - \widetilde{K}_{\mathcal{M},\mathcal{L}} C_{\mathcal{L}} \\ y_{\mathcal{M}}^\top \alpha_{\mathcal{M}} & = & -y_{\mathcal{L}}^\top C_{\mathcal{L}}, \end{array}$$

whose solution is affine in $C_{\mathcal{L}}$ and thus in C.

Consequently, for known active sets, the solution is affine in C, which implies that the optimal variables (w, α, b) are piecewise affine continuous functions of the vector C. In our situation, C depends linearly on C_+ and C_-, and thus the path is piecewise affine in (C_+, C_-).

Following a path The active sets (and thus the linear system) remain the same as long as all constraints defining the active sets are satisfied, that is, (a) $y_i(w^\top x_i + b) - 1$ is positive for all $i \in \mathcal{R}$ and negative for all $i \in \mathcal{L}$, and (b) for each $i \in \mathcal{M}$, α_i remains between 0 and C_i. This defines a set of linear inequalities in (C_+, C_-). The facets of the polytope defined by these inequalities can always be found in linear time in n, if efficient convex hull algorithms are used (Avis et al., 1997). However, when we only follow a straight line in the (C_+, C_-)-space, the polytope is then a segment and its extremities are trivial to find (also in linear time $O(n)$).

Following Hastie et al. (2005), if a solution is known for one value of (C_+, C_-), we can follow the path along a line, by monitoring which constraints are violated at the boundary of the polytope that defines the allowed domain of (C_+, C_-) for the given active sets.

Numerical issues Several numerical issues have to be solved before the previous approach can be made efficient and stable. Some of the issues directly follow the known issues of the simplex method for linear programming (which is itself an active set method) (Maros, 2002).

- *Path initialization*: In order to easily find a point of entry into the path, we look at situations when all points are in $\mathcal{L}$, that is, $\forall i$, $\alpha_i = C_i$. In order to verify $\alpha^\top y = 0$, this implies that $\sum_{i \in I_+} C_i = \sum_{i \in I_-} C_i$, that is, this means $C_+ n_+ = C_- n_-$ (where $n_+ = |I_+|$ and $n_- = |I_-|$ are the

number of positive and negative training examples), which we now assume. The active sets remain unchanged as long as $\forall i, y_i(w^\top x_i + b) \leqslant 1$, that is:

$$\forall i \in I_+ \quad , \quad b \leqslant 1 - C_+ \left((\widetilde{K}\delta_+)_i + \frac{n_+}{n_-}(\widetilde{K}\delta_-)_i \right)$$
$$\forall i \in I_- \quad , \quad b \geqslant -1 + C_+ \left((\widetilde{K}\delta_+)_i + \frac{n_+}{n_-}(\widetilde{K}\delta_-)_i \right),$$

where δ_+ (resp. δ_-) is the indicator vector of the positive (resp. negative) examples.

Let us define the following two maxima: $m_+ = \max_{i \in I_+} \left((\widetilde{K}\delta_+)_i + \frac{n_+}{n_-}(\widetilde{K}\delta_-)_i \right)$ (attained at i_+) and $m_- = \max_{i \in I_-} \left((\widetilde{K}\delta_+)_i + \frac{n_+}{n_-}(\widetilde{K}\delta_-)_i \right)$ (attained at i_-). The previous conditions are equivalent to

$$-1 + C_+ m_- \leqslant b \leqslant 1 - C_+ m_+.$$

Thus, all points are in $\mathcal{L}$ as long as $C_+ \leqslant 2/(m_- + m_+)$, and when this is verified, b is undetermined, between $-1 + C_+ m_-$ and $1 - C_+ m_+$. At the boundary point $C_+ = 2/(m_- + m_+)$; then both i_+ and i_- are going from $\mathcal{L}$ to $\mathcal{M}$.

Since we vary both C_+ and C_- we can start by following the line $C_+ n_+ = C_- n_-$ and we are thus able to avoid to solve a quadratic program to enter the path, as is done by Hastie et al. (2005) when the data sets are not perfectly balanced.

- *Switching between active sets*: Indices can go from $\mathcal{L}$ to $\mathcal{M}$, $\mathcal{R}$ to $\mathcal{M}$, or $\mathcal{M}$ to $\mathcal{R}$ or $\mathcal{L}$. Empirically, when we follow a line in the plane (C_+, C_-), most points go from $\mathcal{L}$ to $\mathcal{R}$ through $\mathcal{M}$ (or from $\mathcal{R}$ to $\mathcal{L}$ through $\mathcal{M}$), with a few points going back and forth; this implies that empirically the number of kinks when following a line in the (C_+, C_-) plane is $O(n)$.

- *Efficient implementation of linear system*: The use of Cholesky updating and downdating is necessary for stability and speed (Golub and Van Loan, 1996).

- *Computational complexity*: Following the analysis of Hastie et al. (2005), if m is the maximum number of points in $\mathcal{M}$ along the path and p is the number of path following steps, then the algorithm has complexity $O(m^2 p + pmn)$. Empirically, the number of steps is $O(n)$ for one following one line in the (C_+, C_-) plane, so the empirical complexity is $O(m^2 n + mn^2)$.

  It is worth noting that the complexity of obtaining one path of classifiers across one line is roughly the same as obtaining the solution for only one SVM using classical techniques such as sequential minimal optimization (Platt, 1998).

Classification with kernels The path following algorithm developed in this section immediately applies to non-linear classification, by replacing the Gram matrix defined as $K_{ij} = x_i^\top x_j$, by any *kernel matrix* K defined as $K_{ij} = k(x_i, x_j)$, where k is a positive semi-definite kernel function (Shawe-Taylor and Cristianini, 2004).

3.2 Constructing the ROC Curve

Given the tools of Section 3.1, we can learn paths of linear classifiers from data. In this section, we present an algorithm to build ROC curves from the paths. We do this by exploring relevant parts of the (C_+, C_-) space, selecting the best classifiers among the ones that are visited.

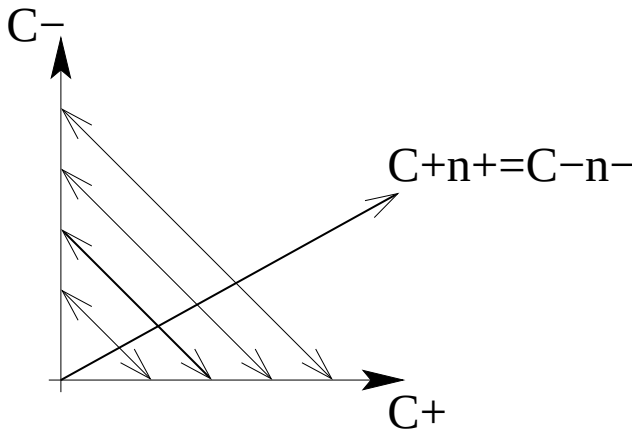

Figure 3: Lines in the (C_+, C_-)-space. The line $C_+ n_+ = C_- n_-$ is always followed first; then several lines with constant $C_+ + C_-$ are followed in parallel, around the optimal line for the validation data (bold curve).

We assume that we have two separate data sets, one for training and one for testing. This approach generalizes to cross validation settings in a straightforward manner.

Exploration phase In order to start the path-following algorithm, we need to start at $C_+ = C_- = 0$ and follow the line $C_+ n_+ = C_- n_-$. We follow this line up to a large upper bound on $C_+ + C_-$. For all classifiers along that line, we compute a misclassification cost on the testing set, with given asymmetry (C_+^0, C_-^0) (as given by the user, and usually, but not necessarily, close to a point of interest in the ROC space). We then compute the best performing pair (C_+^1, C_-^1) and we select pairs of the form (rC_+^1, rC_-^1), where r belongs to a set R of the type $R = \{1, 10, 1/10, 100, 1/100, \dots\}$. The set R provides further explorations for the total amount of regularization $C_+ + C_-$.

Then, for each r, we follow the paths of direction $(1,-1)$ and $(-1,1)$ starting from the point (rC_+^1, rC_-^1). Those paths have a fixed total amount of regularization but vary in asymmetry. In Figure 3, we show all of lines that are followed in the (C_+, C_-) space.

Selection phase After the exploration phase, we have $|R|+1$ different lines in the (C_+, C_-) space: the line $C_- n_- = C_+ n_+$, and the $|R|$ lines $C_+ + C_- = r(C_+^1 + C_-^1), r \in R$. For each of these lines, we know the optimal solution (w, b) for any cost settings on that line. The line $C_- n_- = C_+ n_+$ is used for computational purposes (*i.e.*, to enter the path), so we do not use it for testing purposes.

For each of the R lines in the (C_+, C_-)-plane, we can build the three following ROC curves, as shown in the top of Figure 4 for a simple classification problem involving mixtures of Gaussians:

- *Varying intercept*: We extract the slope w corresponding to the best setting $(C_+^1 + C_-^1)$, and vary the intercept b from $-\infty$ to ∞. This is the traditional method for building an ROC curve for an SVM.

- *Varying asymmetry*: We only consider the line $C_+ + C_- = C_+^1 + C_-^1$ in the (C_+, C_-)-plane; the classifiers that are used are the optimal solutions of the convex optimization problem. Note that for each value of the asymmetry, we obtain a different value of the slope and the intercept.

- *Varying intercept and asymmetry*: For each of the points on the R lines in the (C_+, C_-)-plane, we discard the intercept b and keep the slope w obtained from the optimization problem; we then use all possible intercept values; this leads to R two-dimensional surfaces in the ROC plane. We then compute the convex envelope of these, to obtain a single curve.

Since all classifiers obtained by varying only the intercept (resp. the asymmetry) are included in the set used for varying both the intercept and the asymmetry, the third ROC curve always outperforms the first two curves (*i.e.*, it is always closer to the top left corner). This is illustrated in Figure 4. Once ROC curves are obtained for each of the R lines, we can combine them by taking their upper convex envelope to obtain ROC curves obtained from even larger sets of classifiers, which span several total amounts of regularization. Note that the ROC scores that we consider are obtained by using held out testing data or cross-validation scores; thus by considering larger sets of classifiers, taking upper convex envelopes will always lead to better performance for these scores, which are only approximations of the expected scores on unseen data, and there is a potential risk of overfitting the cross-validation scores (Ng, 1997). In Section 4.3, we present empirical experiments showing that by carefully selecting classifiers based on held out data or cross-validation scores, we are not overfitting.

Intuitively, the ROC curve obtained by varying the asymmetry should be better than the ROC generated by varying the intercept because, for each point, the slope of the classifier is optimized. Empirically, this is generally true, but is not always the case, as displayed in the top of Figure 4. In the bottom of Figure 4, we show the same ROC curve, but in the infinite sample case, where the solution of the SVM was obtained by working directly with densities, separately for each training asymmetry. The ROC curve in the infinite sample case exhibit the same behavior than in the finite sample case, hinting that this behavior is not a small sample effect.

Another troubling fact is that the ROC curve obtained by varying asymmetry, exhibits strong concavities, that is, there are many ROC-inconsistent points: for those points, the solution of the SVM with the corresponding asymmetry is far from being the best linear classifier when performance is measured with the same asymmetry but with the exact 0–1 loss. In addition, even for ROC-consistent points, the training asymmetry and the testing asymmetry differ. In the next section, we analyze why they may differ and characterize their relationships in some situations.

4. Training Versus Testing Asymmetry

We observed in Section 3.2 that the training cost asymmetry can differ from the testing asymmetry. In this section, we analyze their relationships more closely for the population (*i.e.*, infinite sample) case. Although a small sample effect might alter some of the results presented in this section, we argue that most of the discrepancies come from using a convex surrogate to the $0-1$ loss.

Recent results (Bartlett et al., 2004; Zhang, 2004) have shown that using a convex surrogate may lead, under certain conditions, to the *Bayes optimal*, that is, the (usually non-linear) classifier with minimal expected cost. However, those conditions are usually not met in practice, since they essentially implies that the class of functions over which the expected surrogate cost is minimized contains the Bayes optimal classifier. In many cases, the Bayes optimal classifier does not belong

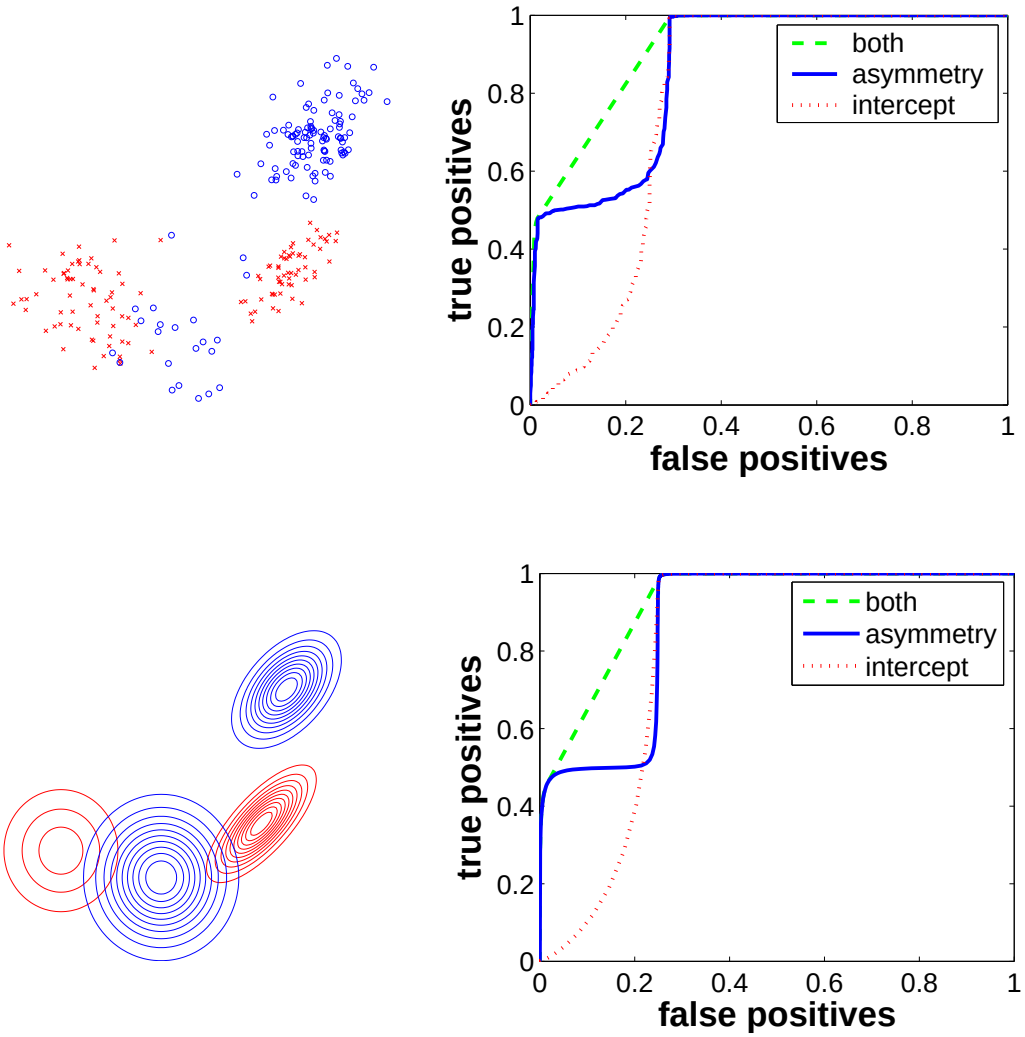

Figure 4: Two examples of ROC curves for bimodal class conditional densities, varying intercept (dotted), varying asymmetry (plain) and varying both (dashed). Top: ROC curves obtained from 10 random splits, using the data shown on the left side (one class is plotted as circles, the other one as crosses); Bottom: ROC curves obtained from corresponding population densities.

to the class of functions, and the best that can be hoped for is to obtain the classifier with minimal expected cost within the class of functions which is considered. Using a surrogate does not always lead to this classifier and the results of this section illustrate and quantify this fact in the context of varying cost asymmetries.

Since we are using population densities, we can get rid of the regularization term and thus only the asymmetry will have an influence on the results, that is, we can restrict ourselves to $C_+ + C_- = 1$. We let $\gamma = C_+/(C_+ + C_-) = C_+$ denote the training asymmetry. For a given training asymmetry γ and a loss function ϕ, in Section 2.2, we defined the *optimal testing asymmetry* $\beta(\gamma)$ for the training asymmetry γ as the testing asymmetry for which the classifier obtained by training with asymmetry γ is optimal. In this section, we will refer to the $\beta(\gamma)$ simply as the *testing asymmetry*.

Although a difference might be seen empirically for all possible asymmetries, we analyze the relationship between the testing cost asymmetry and training asymmetry in cases of extreme asymmetries, that is, in the ROC framework, close to the corner points $(0,0)$ and $(1,1)$. We prove that, depending on the class conditional densities, there are three possible different regimes for extreme asymmetries and that under two of these regimes, the training asymmetry should be chosen less extreme. We also provide, under certain conditions, a simple test that can determine the regime given class conditional densities.

In this section, we choose class conditional densities that are either Gaussian or a mixture of Gaussians, because (a) any density can be approximated as well as desired by mixtures of Gaussians (Hastie et al., 2001), and (b) for the square loss and the erf loss, they enable closed-form calculations that lead to Taylor expansions.

4.1 Optimal Solutions for Extreme Cost Asymmetries

We assume that the class conditional densities are mixtures of Gaussians, that is, the density of positive (resp. negative) examples is a mixture of k_+ Gaussians, with means μ_+^i and covariance matrix Σ_+^i, and mixing weights π_+^i, $i \in \{1,\ldots,k_+\}$ (resp. k_- Gaussians, with means μ_-^i and covariance matrix Σ_-^i, and mixing weights π_-^i, $i \in \{1,\ldots,k_-\}$). We denote M_+ (resp. M_-) the $d \times k_+$ (resp. $d \times k_-$) the matrix of means.

We denote p_+ and p_- as the marginal class densities, $p_+ = P(y=1)$, $p_- = P(y=-1)$. We assume that all mixing weights $\pi_\pm^i$ are strictly positive, that all covariance matrices $\Sigma_\pm^i$ have full rank, and that $p_+, p_- \in (0,1)$.

In the following sections, we provide Taylor expansions of various quantities around the null training asymmetry $\gamma = 0$. The quantities trivially extend around the reverse asymmetry $\gamma = 1$. We focus on two losses, the square loss and the erf loss. The square loss, which leads to a classifier obtained by usual least-square linear regression on the class labels, leads to closed form computations and simple analysis. However, losses $\phi(u)$ which remain bounded when u tends to $+\infty$ are viewed as preferable and usually lead to better performance (Hastie et al., 2001). Losses as the hinge loss or the logistic loss, which tend to zero as u tends to $+\infty$, lead to similar performances. In order to study if using such losses might alter the qualitative behavior observed and analyzed for the square loss around extreme testing asymmetries, we use another loss with similar behavior as u tends to $+\infty$, the erf loss. The erf loss is defined as $\phi_{erf}(u) = [u\psi(u) - u + \psi'(u)]$ and leads to simpler computations than the hinge loss or the logistic loss[2].

2. Note that the erf loss ϕ_{erf} is a tight approximation of a rescaled logistic loss $\frac{1}{2}\log(1+e^{-2u})$, with similar derivatives.

We start with an expansion of the unique global minimum (w,b) of the ϕ-cost with asymmetry γ. For the square loss, (w,b) can be obtained in closed form for any class conditional densities so the expansion is easy to obtain, while for the erf loss, an asymptotic analysis of the optimality conditions has to be carried through, and is only valid for mixtures of Gaussians (see Appendix A for a proof).

We use the following usual asymptotic notations, that is, if f and g are two functions defined around $x = 0$, with g everywhere nonnegative then, $f = O(g)$ if and only if there exists an A positive such that $|f(x)| \leqslant Ag(x)$ for all x sufficiently small, $f = o(g)$ if and only if $f(x)/g(x)$ tends to zero when x tends to zero, $f \sim g$ if and only if $f(x)/g(x)$ tends to one when x tends to zero.

Proposition 1 (square loss) *Under previous assumptions, we have the following expansions:*

$$\begin{aligned} w(\gamma) &= 2\frac{p_+}{p_-}\gamma\Sigma_-^{-1}(\mu_+ - \mu_-) + O(\gamma^2), \\ b(\gamma) &= -1 + \frac{p_+}{p_-}\gamma[2 - 2\mu_-^\top(\mu_+ - \mu_-)] + O(\gamma^2), \end{aligned}$$

where $m = \mu_+ - \mu_-$, and $\Sigma_\pm$ and $\mu_\pm$ are the class conditional means and covariance matrices. For mixtures of Gaussians, we have $\Sigma_\pm = \sum_i \pi_\pm^i \Sigma_\pm^i + M_\pm(\mathrm{diag}(\pi_\pm) - \pi_\pm\pi_\pm^\top)M_\pm^\top$ and $\mu_\pm = \sum_i \pi_\pm^i \mu_\pm^i$.

Proposition 2 (erf loss) *Under previous assumptions, we have the following expansions:*

$$\begin{aligned} w(\gamma) &= (2\log(1/\gamma))^{-1/2}\widetilde{\Sigma}_-^{-1}(\tilde{\mu}_+ - \tilde{\mu}_-) + o\left(\log(1/\gamma)^{-1/2}\right), \\ b(\gamma) &= -(2\log(1/\gamma))^{1/2} + o\left(\log(1/\gamma)^{1/2}\right), \end{aligned}$$

with $\tilde{m} = \mu_+ - \tilde{\mu}_-$, $\tilde{\mu}_- = \sum_i \xi_i \mu_-^i$ and $\widetilde{\Sigma}_- = \sum_i \xi_i \Sigma_-^i$, where $\xi \in \mathbb{R}_+^n$ verifies $\sum_i \xi_i = 1$ and ξ is the unique solution of a convex optimization problem defined in Appendix A.

Note that when there is only one mixture component (Gaussian densities), then $\xi_1 = 1$, and the expansion obtained in Proposition 2 has a simpler expression similar to the one from Proposition 1.

The previous propositions provide a closed-form expansion of the solutions of the convex optimization problems defined by the square loss and the erf loss. In the next section, we compute the testing costs using those classifiers.

4.2 Expansion of Testing Asymmetries

Using the expansions of Proposition 1 and 2, we can readily derive an expansion of the ROC coordinates for small γ, as well as the testing asymmetry $\beta(\gamma)$. We have (see Appendix B for a proof):

Proposition 3 (square loss) *Under previous assumptions, we have the following asymptotic expansion:*

$$\log\left(\frac{p_-}{p+}(\beta(\gamma)^{-1} - 1)\right) = A\frac{p_-^2}{8p_+^2}\frac{1}{\gamma^2} + o(1/\gamma^2),$$

where

$$A = \left(\max_{i_-} \frac{1}{m^\top\Sigma_-^{-1}\Sigma_-^{i_-}\Sigma_-^{-1}m} - \max_{i_+} \frac{1}{m^\top\Sigma_-^{-1}\Sigma_+^{i_+}\Sigma_-^{-1}m}\right).$$

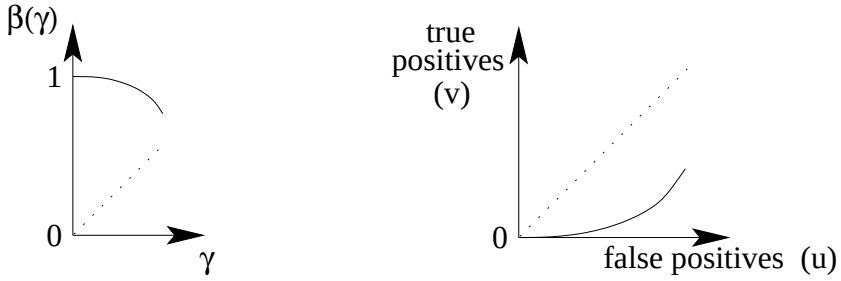

Figure 5: Case $A < 0$: (left) plot of the optimal testing asymmetry $\beta(\gamma)$ as a function of the training asymmetry γ, (right) ROC curve around (0,0) as γ varies close to zero. Classifiers corresponding to γ close to zero are ROC-inconsistent.

Proposition 4 (erf loss) *Under previous assumptions, we have the following asymptotic expansion:*

$$\log\left(\frac{p_-}{p+}(\beta(\gamma)^{-1}-1)\right) = 2\widetilde{A}\log(1/\gamma) + o(\log(1/\gamma)), \tag{4}$$

where

$$\widetilde{A} = \left(\max_{i_-}\frac{1}{\tilde{m}^\top\widetilde{\Sigma}_-^{-1}\Sigma_-^{i_-}\widetilde{\Sigma}_-^{-1}\tilde{m}} - \max_{i_+}\frac{1}{\tilde{m}^\top\widetilde{\Sigma}_-^{-1}\Sigma_+^{i_+}\widetilde{\Sigma}_-^{-1}\tilde{m}}\right).$$

The rest of the analysis is identical for both losses and thus, for simplicity, we focus primarily on the square loss. For the square loss, we have two different regimes, depending on the sign of $A = \left(\max_{i_-}\frac{1}{m^\top\Sigma_-^{-1}\Sigma_-^{i_-}\Sigma_-^{-1}m} - \max_{i_+}\frac{1}{m^\top\Sigma_-^{-1}\Sigma_+^{i_+}\Sigma_-^{-1}m}\right)$:

- $A < 0$: from the expansion in Eq. (3), we see that $\log\left(\frac{p_-}{p+}(\beta(\gamma)^{-1}-1)\right)$ is asymptotically equivalent to a negative constant times $1/\gamma^2$, implying that the testing asymmetry $\beta(\gamma)$ tends to one exponentially fast. In addition, from Eq. (1), $\frac{p_-}{p+}(\beta(\gamma)^{-1}-1)$ is equal to $\frac{dv/d\gamma}{du/d\gamma}$, which is the slope of the ROC curve. Because this is an expansion around the null training asymmetry, the ROC curve must be starting from the point (0,0); since the slope at that point is zero, that is, proportional to the horizontal axis, the ROC curve is on the bottom right part of the main diagonal and the points corresponding to training asymmetries close to $\gamma = 0$ are not ROC-consistent, that is, the classifiers with training asymmetry too close to zero are useless as they are too extreme. See Figure 5 for a plot of the ROC curve around (0,0) and of $\beta(\gamma)$ around $\gamma = 0$. In this situation, better performance for a given testing asymmetry can be obtained by using a less extreme training asymmetry, simply because too extreme training asymmetries lead to classifier who perform worse than trivial classifiers.

- $A > 0$: from the expansion in Eq. (3), we see that the testing asymmetry tends to 0 exponentially fast, in particular, the derivative $d\beta/d\gamma$ is null at $\gamma = 0$, meaning, that the testing asymmetry is significantly smaller than the training asymmetry, that is, more extreme. The slope of the ROC curve around (0,0) is then vertical. See Figure 6 for a plot of the ROC curve

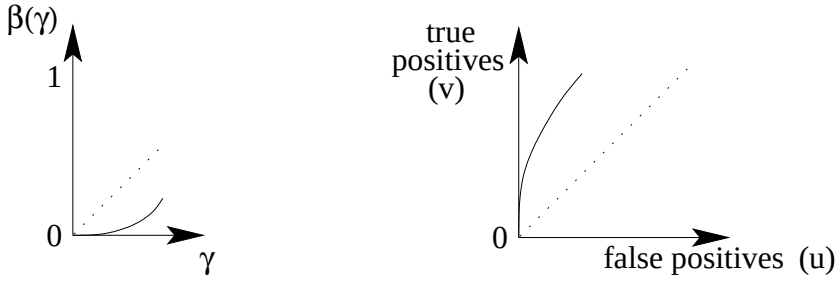

Figure 6: Case $A > 0$. Left: Plot of the optimal testing asymmetry $\beta(\gamma)$ as a function of the training asymmetry γ; Right: ROC curve around (0,0) as γ varies close to zero. Classifiers corresponding to γ close to zero are ROC-consistent, but since $\beta(\gamma) < \gamma$, best performance for a given testing asymmetry is obtained for less extreme training asymmetries.

around (0,0) and of $\beta(\gamma)$ around zero. In this situation, better performance for a given testing asymmetry can be obtained by using a less extreme training asymmetry.

- $A = 0$: the asymptotic expansion does not provide any information relating to the behavior of the testing asymmetry. We are currently investigating higher-order expansions in order to study the behavior of this limiting case. Given two random class-conditional distributions, this regime is unlikely. Precise measure theoretic statements regarding conditions under which the measure of the set of pairs of class-conditional distributions that belong to this regime is zero, are beyond the scope of this paper.

  Note that when the two class conditional densities are Gaussians with identical covariance (a case where the Bayes optimal classifier is indeed linear for all asymmetries), we are in the present case.

Overall, in the two more likely regimes where $A \neq 0$, we have shown that, given an extreme testing asymmetry, the training asymmetry should be chosen less extreme. Because we have considered mixtures of Gaussians, this result applies to a wide variety of distributions. Moreover, this result is confirmed empirically in Section 4.3, where we show in Table 2 examples of the mismatch between testing asymmetry and training asymmetry. Moreover, the strength of the effects we have described above depends on the norm of $m = \mu_+ - \mu_-$: if m is large, that is, the classification problem is simple, then those effects are less strong, while when m is small, they are stronger.

For the erf loss, there are also three regimes, but with weaker effects since the testing asymmetry $\beta(\gamma)$ tends to zero or one polynomially fast, instead of exponentially fast. However, the qualitative result remains the same: there is a mismatch between testing and training asymmetries.

In Figure 7 and Figure 8, we provide several examples for the square loss and the erf loss, with the two regimes $A > 0$ and $A < 0$ and different strengths. Those figures (as well as the bottom of Figure 4) are obtained by solving the respective convex optimization problems with population densities separately for each asymmetry, using Newton's method (Boyd and Vandenberghe, 2003). It is worth noting, that, although the theoretical results obtained in this section are asymptotic expansions around the corners (*i.e.*, extreme asymmetries), the effects also often remain valid far from

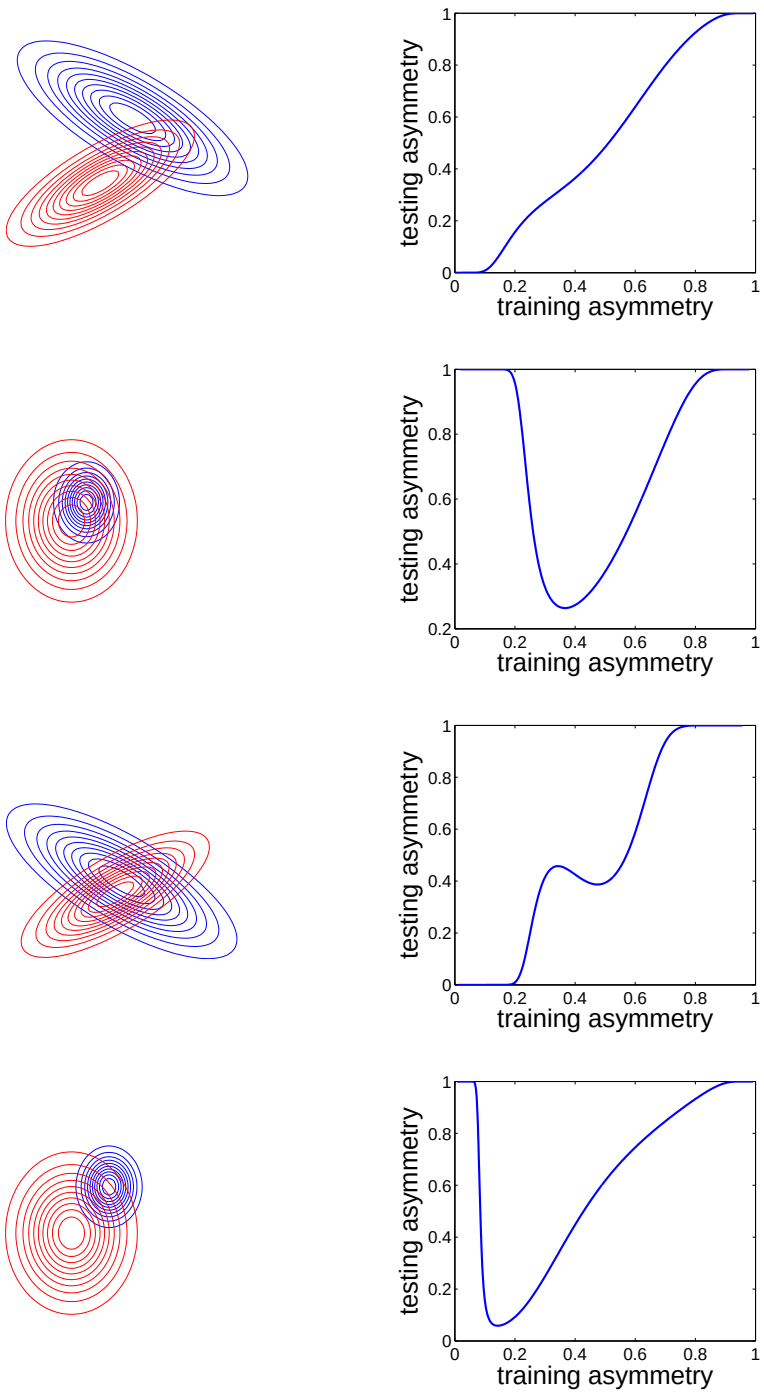

Figure 7: Training asymmetry vs. testing asymmetry, **square loss**: (Left) Gaussian class conditional densities, (right) testing asymmetry vs. training asymmetry; from top to bottom, the values of A are 0.12, -6, 3, -0.96.

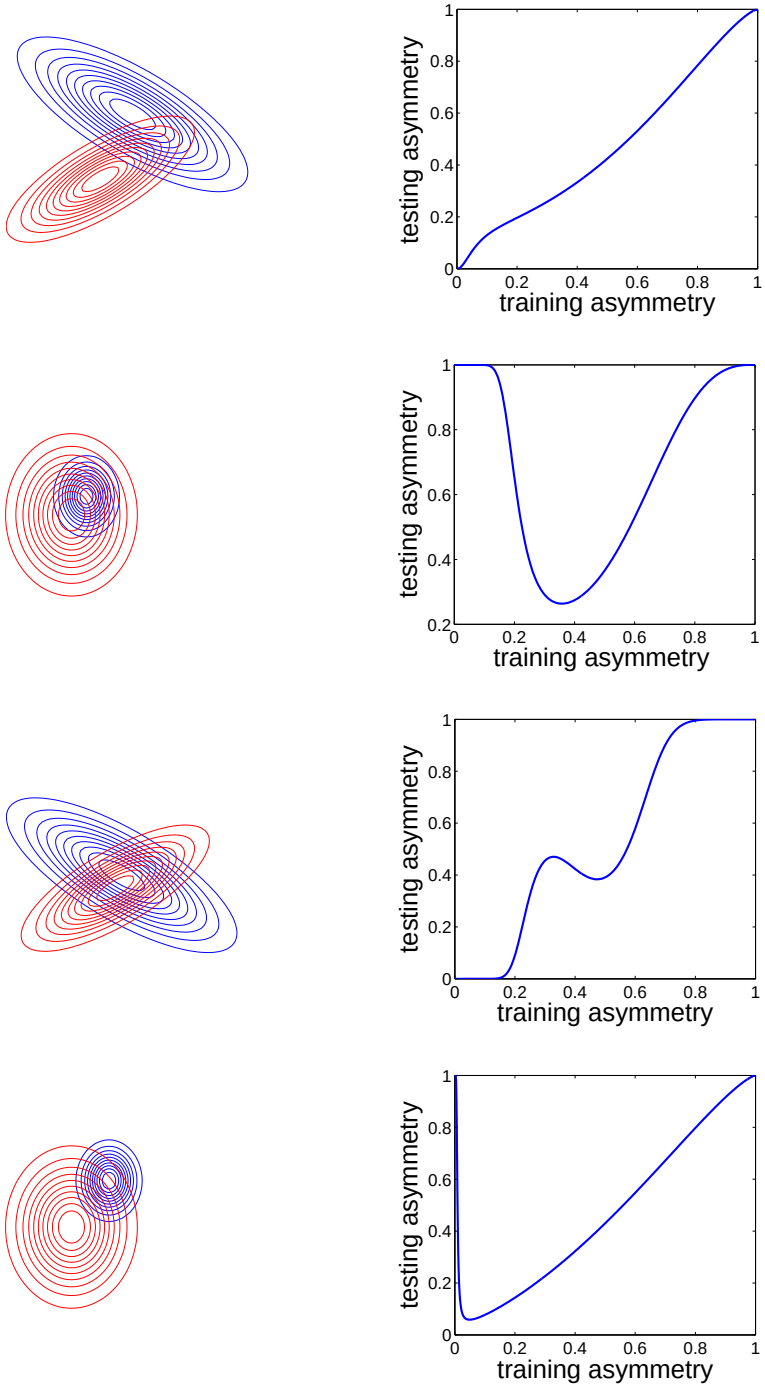

Figure 8: Training asymmetry vs. testing asymmetry, **erf loss**. Left: Gaussian class conditional densities; Right: Testing asymmetry vs. training asymmetry; from top to bottom, the values of $\widetilde{A}$ are 0.12, -6, 3, -0.96.

the corners. However, in general, for non-extreme asymmetries, the mismatch might be inverted, that is, a more extreme training asymmetry might lead to better performance.

Although the results presented in this section show that there are two regimes, given data, they do not readily provide a test to determine which regime is applicable and compute the corresponding optimal training asymmetry for a given testing asymmetry; an approach similar to the one presented by Tong and Koller (2000) could be followed, that is, we could estimate class conditional Gaussian mixture densities and derive the optimal hyperplane from those densities using the tests presented in this paper (which can be performed in closed form for the square loss, while for the erf loss, they require to solve a convex optimization problem). However, this approach would only applicable to extreme asymmetries (where the expansions are valid). Instead, since we have developed an efficient algorithm to obtain linear classifiers for all possible training asymmetries, it is simpler to choose among all asymmetries the one that works best for the problem at hand, which we explore in the next section.

4.3 Building the Entire ROC Curve for a Single Point

As shown empirically in Section 3.2, and demonstrated theoretically in the previous section, training and testing asymmetries may differ; this difference suggests that even when the user is interested in only one cost asymmetry, the training procedure should explore more cost asymmetries, that is, build the ROC curve as presented in Section 3.2 and chose the best classifier as follows: a given testing asymmetry leads to a unique slope in the ROC space, and the optimal point for this asymmetry is the point on the ROC curve whose tangent has the corresponding slope and which is closest to the upper-left corner. In our simulations, we compare two types of ROC curves, one which is obtained by varying the intercept, and one which is obtained by varying both the training asymmetry and the intercept. In our context, using the usual ROC curve obtained by varying only the intercept is essentially equivalent to the common practice of keeping the slope optimized with the convex risk and optimizing the intercept directly using the 0-1 loss instead of the convex surrogate (which is possible by grid search since only one real variable is considered for optimization).

We thus compare in Table 1 and 2, for various data sets and linear classifiers, and for all testing cost asymmetries[3] γ, the three different classifiers: (a) the classifier obtained by optimizing the convex risk with training cost asymmetry γ (a classifier referred to as "one"), (b) the classifier selected from the ROC curve obtained from varying the intercept of the previous classifier (a classifier referred to as "int"), and (c) the classifier selected from the ROC curve obtained by varying both the training asymmetry and the intercept (a classifier referred to as "all").

The ROC curves are obtained by 10-fold cross validation with a wrapper approach to handle the selection of the training asymmetries. The goals of these simulations are to show (1) that using only an approximation to the performance on unseen data is appropriate to select classifiers, and (2) that the performance can be significantly enhanced by simply looking at more training asymmetries. We use several data sets from the UCI data set repository (Blake and Merz, 1998), as well as the mixture of Gaussians shown in Figure 4 and the first Gaussian density in Figure 7.

We used the following wrapper approach to build the ROC curves and compare the performance of the three methods by 10-fold cross validation (Kohavi and John, 1997): for each of the

3. More precisely, 1000 values of testing and training asymmetries were tried, uniformly spread in $[0,1]$. Note that our algorithmic framework allows us to obtain classifiers for all asymmetries in $[0,1]$; therefore considering 1000 values is not computationally expensive.

10 outer folds, we select the best parameters (*i.e.*, training asymmetry and/or intercept) by 10-fold cross validation on the training data. The best classifier for each of the two ROC curves is kept if the difference in performance with the usual classifier "one" on the ten inner folds is statistically significant, as measured by one-tailed Wilcoxon signed rank tests with 5% confidence level (Hollander and Wolfe, 1999). In cases where statistical significance is not found, the original classifier obtained with the training asymmetry equal to the testing asymmetry is used (in this case, the classifier is identical to the classifier "one"). The selected classifier is then trained on the entire training data of the outer fold and tested on the corresponding testing data. We thus obtain for each outer fold, the performance of the three classifiers ("one", "int","all").

In Table 1, we report the results of statistical tests performed to compare the three classifiers. For each testing asymmetry, we performed one-tailed Wilcoxon signed rank tests on the testing costs of the outer folds, with 5% confidence levels (Hollander and Wolfe, 1999). The proportions of acceptance (*i.e.*, the number of acceptances divided by the number of testing asymmetries that are considered) for all one-sided tests are reported in Table 1. The empirical results from columns "all>one" and "int>one" show that comparing classifiers with only an approximation of the testing cost on unseen data, namely cross-validation scores, leads to classifiers that most of the time perform no worse than the usual classifier (based only on the testing asymmetry and the convex risk), which strongly suggests that we are not overfitting the cross-validation data. Moreover, the column "one>int" shows that there is a significant gain in simply optimizing the intercept using the non-convex non-differentiable risk based on the 0-1 loss, and columns "one>all" and "int>all" show that there is even more gain in considering all training asymmetries[4]. As mentioned earlier, when the increase of performance of the classifiers "all" and "int" is not deemed statistically significant on the inner cross-validation folds, the classifier "one" is used instead. It is interesting to note that, when we do not allow the possibility to keep "one", we obtain significantly worse performance on the outer-fold tests

To highlight the potential difference between testing and training asymmetries, Table 2 gives some examples of mismatches between the testing asymmetry and the training asymmetry selected by the cross validation procedure. Since the training asymmetries corresponding to a given testing asymmetry may differ for each outer fold, we use the training asymmetry that was selected by the first (*i.e.*, a random) outer fold of cross validation, and compute the cross-validation scores corresponding to this single training asymmetry for all nine other outer folds. For each data set, testing asymmetries were chosen so that the difference in testing performance between the classifier "one" and "all" on the first fold is greatest. Note that the performance of the classifier "int" is different from the performance of "all" only if the difference was deemed significant on the outer fold used to select the training asymmetry. Results in Table 2 show that in some cases, the difference is large, and that a simple change in the training procedure may lead to substantial gains. Moreover, when the testing asymmetry is extreme, such as for the GAUSSIAN and PIMA data sets, the training asymmetry is less extreme and leads to better performance, illustrating the theoretical results of Section 4. Note also, that for other data sets, such as TWONORM or IONOSPHERE, the optimal training asymmetry is very different from the testing asymmetry: we find that using all asymmetries

4. In some cases, considering all training asymmetries performs worse than using a single asymmetry with optimized intercept (column "all>int" in Table 1); in those cases, the difference in performance, although statistically significant, is small and we conjecture it is due to using an approximation of expected testing performance to select the training asymmetry.

Data set	d	n	one>all	all>one	int>all	all>int	one>int	int>one
BREAST	9	683	8.7	0.0	4.1	0.0	5.2	0.0
DERMATOLOGY	34	358	0.0	0.0	0.0	0.0	0.0	0.0
GAUSSIANS	2	2000	17.6	0.0	14.2	0.0	9.5	0.0
MIXTURES	2	2000	49.7	0.0	25.8	0.0	41.2	0.0
LIVER	6	345	0.0	0.0	0.0	0.0	0.0	0.0
VEHICLE	18	416	13.1	0.0	13.1	0.0	0.0	0.0
PIMA	8	768	9.2	0.0	0.0	0.0	5.6	0.0
RINGNORM	2	2000	48.6	0.0	5.9	0.0	40.4	0.0
TWONORM	2	2000	97.4	0.0	15.8	5.2	91.3	0.0
ADULT	13	2000	12.0	0.0	8.8	0.0	3.2	0.0
IONOSPHERE	33	351	42.0	0.0	13.2	0.0	22.5	0.0

Table 1: Comparison of performances of classifiers: for each data set, the number of features is d, the total number of data points is n, and the proportions of acceptance (*i.e.*, the number of acceptances divided by the number of testing asymmetries that are considered) of the six one-sided tests between the three classifiers are given in the last six columns (Wilcoxon signed rank tests with 5% confidence level comparing cross-validation scores). Note that we compare testing by using costs, and hence better performance corresponds to smaller costs. See text for details.

Data set	γ (test)	"one"	"int"	"all"	γ (train)
BREAST	0.008	0.51 ± 0.07	0.55 ± 0.96	0.51 ± 0.07	0.995
DERMATOLOGY	0.991	0.37 ± 0.05	0.37 ± 0.05	0.37 ± 0.05	0.991
GAUSSIANS	0.067	6.66 ± 0.00	5.10 ± 0.59	6.66 ± 0.00	0.128
MIXTURES	0.327	29.80 ± 2.61	16.73 ± 2.20	19.70 ± 1.82	0.990
LIVER	0.192	16.10 ± 0.96	15.96 ± 4.44	16.10 ± 0.96	0.560
PIMA	0.917	10.83 ± 0.35	9.61 ± 2.32	10.83 ± 0.35	0.500
RINGNORM	0.327	33.18 ± 1.07	25.53 ± 3.95	25.15 ± 4.86	0.500
VEHICLE	0.010	0.21 ± 0.15	0.21 ± 0.15	0.21 ± 0.15	0.010
TWONORM	0.382	36.98 ± 0.34	1.46 ± 1.00	1.55 ± 1.35	0.872
ADULT	0.947	2.67 ± 0.09	2.50 ± 0.08	2.67 ± 0.09	0.500
IONOSPHERE	0.933	4.77 ± 0.21	2.30 ± 2.22	4.77 ± 0.21	0.067

Table 2: Training with the testing asymmetry γ versus training with all cost asymmetries: we report cross-validation testing costs (multipled by 100). Only the asymmetry with the largest difference in testing performance between the classifier "one" and "all" is reported. We also report the training asymmetry which led to the best performance. Given an asymmetry γ we use the cost settings $C_+ = 2\gamma$, $C_- = 2(1-\gamma)$ (which leads to the misclassification error if $\gamma = 1/2$). See text for details.

leads to better performance. However, we have not identified general rules for selecting the best training asymmetry.

5. Conclusion

We have presented an efficient algorithm to build ROC curves by varying the training cost asymmetries for SVMs. The algorithm is based on the piecewise linearity of the path of solutions when the cost of false positives and false negatives vary. We have also provided a theoretical analysis of the relationship between the potentially different cost asymmetries assumed in training and testing classifiers, showing that they differ under certain circumstances. In particular, in case of extreme asymmetries, our theoretical analysis suggests that training asymmetries should be chosen less extreme than the testing asymmetry.

We have characterized key relationships, and have worked to highlight the potential practical value of building the entire ROC curve even when a single point may be needed. All learning algorithms considered in this paper involve using a convex surrogate to the correct non differentiable non convex loss function. Our theoretical analysis implies that because we use a convex surrogate, using the testing asymmetry for training leads to non-optimal classifiers. We thus propose to generate all possible classifiers corresponding to all training asymmetries, and select the one that optimizes a good approximation to the true loss function on unseen data (*i.e.*, using held out data or cross validation). As shown in Section 3, it turns out that this can be done efficiently for the support vector machine. Such an approach can lead to a significant improvement of performance with little added computational cost.

Finally, we note that, although we have focused in this paper on the single kernel learning problem, our approach can be readily extended to the multiple kernel learning setting (Bach et al., 2005b) with appropriate numerical path following techniques.

Acknowledgments

We thank John Platt for sharing his experiences and insights with considerations of cost functions in training and testing support vector machines.

Appendix A. Proof of Expansion of Optimal Solutions

In this appendix, we give the proof of the expansions of optimal solutions for extreme asymmetries, for the square and erf loss. We perform the expansions using the variable $\rho(\gamma) = \dfrac{C_+ p_+}{C_- p_-} = \dfrac{\gamma p_+}{(1-\gamma) p_-} = \dfrac{\gamma p_+}{p_-} + O(\gamma^2)$ around zero.

A.1 Square Loss. Proof of Proposition 1.

In this case, the classifier is simply a linear regression on y and (w,b) can be obtained in closed form as the solution of the following linear system (obtained from the normal equations):

$$\begin{aligned} b &= \frac{\rho-1}{\rho+1} - w^\top \frac{\rho\mu_+ + \mu_-}{\rho+1}, \\ \left(\rho\Sigma_+ + \Sigma_- + \frac{\rho}{\rho+1}(\mu_+-\mu_-)(\mu_+-\mu_-)^\top\right) w &= \frac{2\rho}{\rho+1}(\mu_+-\mu_-), \end{aligned}$$

where Σ_+ and Σ_- are the class conditional means and covariance matrices.

The first two terms of the Taylor expansions around $\rho = 0$ (*i.e.*, around $\gamma = 0$) are straightforward to obtain:

$$\begin{aligned} w &= 2\rho\Sigma_-^{-1}m - 2\rho^2\left[\Sigma_-^{-1}m + \Sigma_-^{-1}(\Sigma_+ + mm^\top)\Sigma_-^{-1}m\right] + O(\rho^3), \\ b &= -1 + \rho\left[2 - 2\mu_-^\top\Sigma_-^{-1}m\right] + O(\rho^2). \end{aligned}$$

A.2 Erf Loss. Proof of Proposition 2.

We begin by proving Proposition 2 in the Gaussian case, where the proof is straightforward, and we then extend to the Gaussian mixture case. In order to derive optimality conditions for the erf loss, we first need to compute expectations of the erf loss and its derivatives for Gaussian densities.

A.2.1 EXPECTATION OF THE ERF LOSS AND ITS DERIVATIVES FOR GAUSSIAN DENSITIES

A short calculation shows that, when expectations are taken with respect to a normal distribution with mean μ and covariance matrix Σ, we have:

$$\begin{aligned} E\phi_{erf}(w^\top x+b) &= (-w^\top\mu - b)\psi\left(\frac{-w^\top\mu-b}{(1+w^\top\Sigma w)^{1/2}}\right) \\ &\qquad + (1+w^\top\Sigma w)^{1/2}\psi'\left(\frac{-w^\top\mu-b}{(1+w^\top\Sigma w)^{1/2}}\right), \\ E\frac{\partial\phi_{erf}(w^\top x+b)}{\partial w} &= -\mu\psi\left(\frac{-w^\top\mu-b}{(1+w^\top\Sigma w)^{1/2}}\right) + \frac{\Sigma w}{(1+w^\top\Sigma w)^{1/2}}\psi'\left(\frac{-w^\top\mu-b}{(1+w^\top\Sigma w)^{1/2}}\right), \\ E\frac{\partial\phi_{erf}(w^\top x+b)}{\partial b} &= -\psi\left(\frac{-w^\top\mu-b}{(1+w^\top\Sigma w)^{1/2}}\right). \end{aligned}$$

A.2.2 GAUSSIAN CASE

In order to derive the asymptotic expansion, we derive the optimality conditions for (w,b) and study the behavior as ρ tends to zero. Let's define $t_- = \dfrac{w^\top\mu_- + b}{(1+w^\top\Sigma_- w)^{1/2}}$ and $t_+ = \dfrac{-w^\top\mu_+ - b}{(1+w^\top\Sigma_+ w)^{1/2}}$.

The optimality conditions for (w,b) are the following (obtained by zeroing derivatives with respect to b and w):

$$p_+C_+\left(-\mu_+\psi(t_+) + \frac{\Sigma_+ w}{(1+w^\top\Sigma_+ w)^{1/2}}\psi'(t_+)\right)$$

$$+p_- C_- \left(\mu_- \psi(t_-) + \frac{\Sigma_- w}{(1+w^\top \Sigma_- w)^{1/2}} \psi'(t_-) \right) = 0,$$

$$-p_+ C_+ \psi(t_+) + p_- C_- \psi(t_-) = 0.$$

They can be rewritten as follows (with $\rho = \frac{C_+ p_+}{C_- p_-}$):

$$\rho \left(-\mu_+ \psi(t_+) + \frac{\Sigma_+ w}{(1+w^\top \Sigma_+ w)^{1/2}} \psi'(t_+) \right) + \left(\mu_- \psi(t_-) + \frac{\Sigma_- w}{(1+w^\top \Sigma_- w)^{1/2}} \psi'(t_-) \right) = 0, \quad (5)$$

$$\rho \psi(t_+) = \psi(t_-). \quad (6)$$

Since the cumulative function ψ is always between zero and one, Eq. (6) implies that as ρ tends to zero, $\psi(t_-)$ tends to zero, and thus t_- tends to $-\infty$. This in turn implies that $b/(1+||w||)$ tends to $-\infty$, which in turn implies that t_+ tends to infinity and $\psi(t_+)$ tends to 1.

It is well known that as z tends to $-\infty$, we have $\psi(z) \approx \frac{\psi'(z)}{-z}$ (see, for example, Bleistein and Handelsman (1986)). Thus, if we divide Eq. (5) by $\psi(t_-)$, we get:

$$-\rho \frac{\psi(t_+)}{\psi(t_-)} \mu_+ + \rho \frac{\Sigma_+ w}{(1+w^\top \Sigma_+ w)^{1/2}} \frac{\psi'(t_+)}{\psi(t_+)} \frac{\psi(t_+)}{\psi(t_-)} + \mu_- + \frac{\Sigma_- w}{(1+w^\top \Sigma_- w)^{1/2}} \frac{\psi'(t_-)}{\psi(t_-)} = 0, \text{ that is,}$$

$$\frac{\Sigma_+ w}{(1+w^\top \Sigma_+ w)^{1/2}} \frac{\psi'(t_+)}{\psi(t_+)} + \frac{\Sigma_- w}{(1+w^\top \Sigma_- w)^{1/2}} (-t_-) \sim \mu_+ - \mu_-. \quad (7)$$

The first term in the left hand side of Eq. (7) is the product of a bounded term and a term that goes to zero, it is thus tending to zero as ρ tends to zero. Since the right hand side is constant, this implies that the second term in the left hand side must be bounded, and thus, since $|t_-|$ tends to infinity, that w tends to zero. By removing all negligible terms in Eq. (7), we have the following expansion for w:

$$w \approx \frac{1}{-t_-} \Sigma_-^{-1} (\mu_+ - \mu_-).$$

In order to obtain the expansion as a function of ρ, we need to expand t_-. From Eq. (6) and the fact that $\psi(t_+)$ tends to one, we obtain $\psi(t_-) \sim \rho$, A classical asymptotic result on the inverse erf function shows that $t_- \sim -\sqrt{2\log(1/\rho)}$. This finally leads to:

$$\begin{aligned} b(\rho) &\sim -(2\log(1/\rho))^{1/2}, \\ w(\rho) &\sim (2\log(1/\rho))^{-1/2} \Sigma_-^{-1} (\mu_+ - \mu_-). \end{aligned}$$

A.2.3 GENERAL CASE

We assume that each $\pi_\pm^i$ is strictly positive and that each covariance matrix $\Sigma_\pm^i$ is positive definite. We define $t_-^i = \frac{w^\top \mu_-^i + b}{(1+w^\top \Sigma_-^i w)^{1/2}}$ and $t_+^i = \frac{-w^\top \mu_+^i - b}{(1+w^\top \Sigma_+^i w)^{1/2}}$.

Without loss of generality, we can assume that the positive class is centered, that is, $\mu_+ = \sum_i \pi_+^i \mu_+^i = 0$. The optimality conditions for (w,b) are the following (obtained by zeroing derivatives with respect to b and w):

$$\rho\left(\sum_i \pi_+^i \left\{-\mu_+^i \psi(t_+^i) + \frac{\Sigma_+^i w}{(1+w^\top \Sigma_+^i w)^{1/2}}\psi'(t_+^i)\right\}\right) + \left(\sum_i \pi_-^i \left\{\mu_-^i \psi(t_-^i) + \frac{\Sigma_-^i w}{(1+w^\top \Sigma_- w)^{1/2}}\psi'(t_-^i)\right\}\right) = 0, \tag{8}$$

$$-\rho \sum_i \pi_+^i \psi(t_+^i) + \sum_i \pi_-^i \psi(t_-^i) = 0. \tag{9}$$

From Eq. (9), we obtain that $\psi(t_-^i)$ tends to zero for all i, and this implies that $b/(1+||w||)$ tends to $-\infty$, and in turn that $\psi(t_+^i)$ tends to 1 for all i. We can now divide Eq. (8) by $\sum_i \pi_-^i \psi(t_-^i) = \rho \sum_i \pi_+^i \psi(t_+^i)$, to obtain:

$$\frac{\sum_i \pi_-^i \psi(t_-^i)\mu_-^i}{\sum_i \pi_-^i \psi(t_-^i)} - \frac{\sum_i \pi_+^i \psi(t_+^i)\mu_+^i}{\sum_i \pi_+^i \psi(t_+^i)} = -\frac{1}{\sum_i \pi_+^i \psi(t_+^i)} \sum_i \pi_+^i \psi(t_+^i) \left\{\frac{\Sigma_+^i w}{(1+w^\top \Sigma_+^i w)^{1/2}} \frac{\psi'(t_+^i)}{\psi(t_+^i)}\right\} - \frac{1}{\sum_i \pi_-^i \psi(t_-^i)} \sum_i \pi_-^i \psi(t_-^i) \left\{\frac{\Sigma_-^i w}{(1+w^\top \Sigma_-^i w)^{1/2}} \frac{\psi'(t_-^i)}{\psi(t_-^i)}\right\}.$$

Let $\tilde{\pi}_i^+ = \frac{\pi_+^i \psi(t_+^i)}{\sum_j \pi_+^j \psi(t_+^j)}$ and $\tilde{\pi}_i^- = \frac{\pi_-^i \psi(t_-^i)}{\sum_j \pi_-^j \psi(t_-^j)}$. We can rewrite the preceding equation as

$$\sum_i \tilde{\pi}_-^i \mu_-^i - \sum_i \tilde{\pi}_+^i \mu_+^i = -\sum_i \tilde{\pi}_+^i \left\{\frac{\Sigma_+^i w}{(1+w^\top \Sigma_+^i w)^{1/2}} \frac{\psi'(t_+^i)}{\psi(t_+^i)}\right\} - \sum_i \tilde{\pi}_-^i \left\{\frac{\Sigma_-^i w}{(1+w^\top \Sigma_-^i w)^{1/2}} \frac{\psi'(t_-^i)}{\psi(t_-^i)}\right\}.$$

As in the Gaussian case, the first term of the right hand is negligible compared to the second term when ρ goes to zero. Moreover, for all i, we have $\psi'(t_-^i)/\psi(t_-^i) \approx -t_-^i$ and we thus get:

$$\sum_i \tilde{\pi}_-^i \mu_-^i - \sum_i \tilde{\pi}_+^i \mu_+^i = -\sum_i \tilde{\pi}_-^i \left\{\frac{-t_-^i}{(1+w^\top \Sigma_-^i w)^{1/2}}\right\} \Sigma_-^i w. \tag{10}$$

The left hand side of Eq. (10) is bounded; Because $\tilde{\pi}^-$ sums to one, the lowest eigenvalue of the matrix $\sum_i \tilde{\pi}_-^i \Sigma_-^i$ is lower-bounded by the smallest of the smallest eigenvalues of Σ_-^i, $i = 1,\ldots,k_-$. Thus Eq. (10) implies that $t_-^i w$ is bounded as ρ tends to zero. This in turn implies that w tends to zero, that $b \sim t_-^i$ for all i, and that bw is bounded.

The quantities bw, $\tilde{\pi}_+$ and $\tilde{\pi}_-$ are functions of ρ. As ρ tends to zero, they all remain bounded. We now proceed to prove that all points of accumulation of those quantities as ρ tends to zero satisfy a set of equations with an unique solution. This will imply that those quantities converge as ρ tends to zero.

Equation for $\tilde{\pi}_+$ Since t_+^i tends to $+\infty$, $\tilde{\pi}_+^i$ tends to π_+^i, and $\sum_i \tilde{\pi}_+^i \mu_+^i$ tends to zero, since we have assumed that $\sum_i \pi_+^i \mu_+^i = 0$.

Equation for $\tilde{\pi}_-$ and bw Let θ and ξ be points of accumulation of bw and $\tilde{\pi}_-$ as ρ tends to zero (*i.e.*, θ and ξ are limits of sequences $b(\rho_k)w(\rho_k)$ and $\tilde{\pi}_-(\rho_k)$ as k tends to ∞, with $\rho_k \to 0$). From Eq. (10), we get:

$$\sum_i \xi_i \mu_-^i - 0 = \left(\sum_i \xi_i \Sigma_-^i \right) \theta. \tag{11}$$

We can now expand $(t_-^i)^2 - (t_-^j)^2$ for all i, j by keeping the leading terms, noting that $w \to 0$, $|b| \to +\infty$, and bw is bounded:

$$\begin{aligned}
(t_-^i)^2 - (t_-^j)^2 &= \frac{(w^\top \mu_-^i + b)^2}{1 + w^\top \Sigma_-^i w} - \frac{(w^\top \mu_-^j + b)^2}{1 + w^\top \Sigma_-^j w} \\
&\sim 2bw^\top(\mu_-^i - \mu_-^j) - b^2 w^\top(\Sigma_-^i - \Sigma_-^j)w \\
&\to (2\theta^\top \mu_-^i - \theta^\top \Sigma_-^i \theta) - (2\theta^\top \mu_-^j - \theta^\top \Sigma_-^j \theta) \text{ as } \rho_k \to 0.
\end{aligned}$$

We thus have

$$\begin{aligned}
\frac{\tilde{\pi}_-^i}{\tilde{\pi}_-^j} &= \frac{\pi_-^i \psi(t_-^i)}{\pi_-^j \psi(t_-^j)} = \frac{\pi_-^i \psi'(t_-^i)}{\pi_-^j \psi'(t_-^j)} \frac{\psi'(t_-^j)}{\psi(t_-^j)} \\
&\sim \frac{\pi_-^i}{\pi_-^j} \frac{\psi'(t_-^i)}{\psi'(t_-^j)} \frac{t_-^j}{t_-^i} \sim \frac{\pi_-^i}{\pi_-^j} \exp(-\frac{1}{2}[(t_-^i)^2 - (t_-^j)^2]) \text{ since } t_-^i \sim t_-^j \\
&\to \frac{\pi_-^i \exp(-\theta^\top \mu_-^i + \frac{1}{2}\theta^\top \Sigma_-^i \theta)}{\pi_-^j \exp(-\theta^\top \mu_-^j + \frac{1}{2}\theta^\top \Sigma_-^j \theta)} \text{ as } \rho_k \to 0,
\end{aligned}$$

which implies that the vector ξ is proportional to the vector with components $\pi_-^i \exp(-\theta^\top \mu_-^i + \frac{1}{2}\theta^\top \Sigma_-^i \theta)$, that is:

$$\forall i,\ \xi_i = \frac{\pi_i \exp(-\theta^\top \mu_-^i + \frac{1}{2}\theta^\top \Sigma_-^i \theta)}{\sum_j \pi_j \exp(-\theta^\top \mu_-^j + \frac{1}{2}\theta^\top \Sigma_-^j \theta)}. \tag{12}$$

We have shown that points of accumulation (θ, ξ) must verify two equations, Eq. (11) and Eq. (12).

Unique solution of Eq. (11) and Eq. (12) We now prove that Eq. (11) and Eq. (12) together have an unique solution, obtained as the optimum solution of a strictly convex problem. From Eq. (11), we can write θ as a function of ξ as:

$$\theta(\xi) = \left(\sum_i \xi_i \Sigma_-^i \right)^{-1} \sum_i \xi_i \mu_-^i. \tag{13}$$

Let us define the following function defined on the positive orthant $\{\xi, \xi_i > 0, \forall i\}$:

$$H(\xi) = \sum_i \xi_i \log \xi_i - \sum_i \xi_i \left\{ \log \pi_-^i - \theta(\xi)^\top \mu_-^i + \frac{1}{2}\theta(\xi)^\top \Sigma_-^i \theta(\xi) \right\}.$$

Short calculations show that:

$$\begin{aligned}
\frac{\partial \theta}{\partial \xi_i} &= \left(\sum_k \xi_k \Sigma_-^k\right)^{-1} \left(\mu_-^i - \Sigma_-^i \theta(\xi)\right), \\
\frac{\partial \left\{-\theta(\xi)^\top \mu_-^i + \frac{1}{2}\theta(\xi)^\top \Sigma_-^i \theta(\xi)\right\}}{\partial \xi_j} &= -\left(\mu_-^i - \Sigma_-^i \theta(\xi)\right) \left(\sum_k \xi_k \Sigma_-^k\right)^{-1} \left(\mu_-^j - \Sigma_-^j \theta(\xi)\right), \\
\frac{\partial H}{\partial \xi_i} &= \log \xi_i + 1 - \left(\log \pi_-^i - \theta(\xi)^\top \mu_-^i + \frac{1}{2}\theta(\xi)^\top \Sigma_-^i \theta(\xi)\right), \\
\frac{\partial^2 H}{\partial \xi_i \partial \xi_j} &= \delta_{ij} \frac{1}{\xi_i} + \left(\mu_-^i - \Sigma_-^i \theta(\xi)\right) \left(\sum_k \xi_k \Sigma_-^k\right)^{-1} \left(\mu_-^j - \Sigma_-^j \theta(\xi)\right).
\end{aligned}$$

The last equation shows that the function H is strictly convex in the positive orthant. Thus, minimizing $H(\xi)$ subject to $\sum_i \xi_i = 1$ has an unique solution. Optimality conditions are derived by writing down the Lagrangian:

$$\mathcal{L}(\xi, \alpha) = H(\xi) + \alpha\left(\sum_i \xi_i - 1\right),$$

which leads to the following optimality conditions:

$$\forall i,\ \frac{\partial H}{\partial \xi_i} + \alpha = 0, \tag{14}$$

$$\sum_i \xi_i = 1. \tag{15}$$

The last two equations are exactly equivalent to Eq. (12). We have thus proved that the system defining θ and ξ (Eq. (11) and Eq. (12)) has an unique solution obtained from the solution of the convex optimization problem:

$$\begin{array}{rl}
\text{minimize} & \displaystyle\sum_i \xi_i \log \xi_i - \sum_i \xi_i \left\{\log \pi_-^i - \theta(\xi)^\top \mu_-^i + \frac{1}{2}\theta(\xi)^\top \Sigma_-^i \theta(\xi)\right\} \\
\text{with respect to} & \xi \\
\text{such that} & \xi_i \geqslant 0, \forall i \\
& \displaystyle\sum_i \xi_i = 1
\end{array}$$

with

$$\theta(\xi) = \left(\sum_i \xi_i \Sigma_-^i\right)^{-1} \sum_i \xi_i \mu_-^i.$$

Asymptotic equivalent From the value of θ and ξ obtained above, we can derive the asymptotic expansions of w and b. From Eq. (9) and the fact that t_+^i tends to $+\infty$, we get $\sum_i \pi_-^i \psi(t_-^i) \sim \rho \sum_i \pi_+^i = \rho$. In addition, we can show by expanding $(t_-^i)^2$ that $(t_-^i)^2 - b^2$ has a limit when ρ tends to 0, which in turn implies that $\psi(b)/\rho$ has a finite limit. This implies that $b \approx -(2\log(1/\rho))^{1/2}$. From the fact that bw tends to a limit θ, we immediately obtain that $w \approx \theta/b$, which completes the proof of Proposition 2.

Appendix B. Proof of Expansion of Testing Asymmetries

For the two losses we considered (square and erf), the expansions of w and b around $\gamma = 0$ lead to

$$\frac{w(\gamma)}{b(\gamma)} \approx -c(\gamma)a,$$

where $c(\gamma) = 2\frac{p_+}{p_-}\gamma$, $a = \Sigma_-^{-1}(\mu_+ - \mu_-)$ for the square loss and $c(\gamma) = (2\log(1/\gamma))^{-1}$, $a = \widetilde{\Sigma}_-^{-1}(\tilde{\mu}_+ - \tilde{\mu}_-)$ for the erf loss.

The proportion of false positives $u(\gamma)$ and true positives $v(\gamma)$ can be obtained as:

$$\begin{aligned} u(\gamma) &= P(w^\top x + b \geqslant 0 | y = -1) = \sum_i \pi_-^i \psi\left(\frac{w^\top \mu_-^i + b}{(w^\top \Sigma_-^i w)^{1/2}}\right) = \psi(t_u^i(\gamma)), \\ v(\gamma) &= P(w^\top x + b \geqslant 0 | y = 1) = \sum_i \pi_+^i \psi\left(\frac{w^\top \mu_+^i + b}{(w^\top \Sigma_+^i w)^{1/2}}\right) = \psi(t_v^i(\gamma)), \end{aligned}$$

and we have the expansions

$$\begin{aligned} t_u^i(\gamma) \triangleq \frac{w^\top \mu_-^i + b}{(w^\top \Sigma_-^i w)^{1/2}} &\approx \frac{-1}{c(\gamma)(a^\top \Sigma_-^i a)^{1/2}}, \\ t_v^i(\gamma) \triangleq \frac{w^\top \mu_+^i + b}{(w^\top \Sigma_+^i w)^{1/2}} &\approx \frac{-1}{c(\gamma)(a^\top \Sigma_+^i a)^{1/2}}, \\ \frac{du}{d\gamma} &= \sum_i \pi_-^i \frac{dt_u^i}{dc}\frac{dc}{d\gamma}\psi'(t_u^i(\gamma)) \sim \frac{dc}{d\gamma}\sum_i \pi_-^i \frac{1}{\sqrt{2\pi}}\frac{\exp(-(a^\top \Sigma_-^i a)^{-1}/2c(\gamma)^2)}{c(\gamma)^2 (a^\top \Sigma_-^i a)^{-1/2}}, \\ \frac{dv}{d\gamma} &= \sum_i \pi_+^i \frac{dt_v^i}{dc}\frac{dc}{d\gamma}\psi'(t_v^i(\gamma)) \sim \frac{dc}{d\gamma}\sum_i \pi_+^i \frac{1}{\sqrt{2\pi}}\frac{\exp(-(a^\top \Sigma_+^i a)^{-1}/2c(\gamma)^2)}{c(\gamma)^2 (a^\top \Sigma_+^i a)^{-1/2}}. \end{aligned}$$

The expansions of $\frac{du}{d\gamma}$ and $\frac{dv}{d\gamma}$ are each dominated by a single term, corresponding to indices i_- and i_+ that respectively maximized $(a^\top \Sigma_-^i a)^{-1}$ and $(a^\top \Sigma_+^i a)^{-1}$ (we assume for simplicity that all values of $a^\top \Sigma_\pm^i a$ are distinct). We then obtain

$$\log\left(\frac{dv}{d\gamma}\Big/\frac{du}{d\gamma}\right) \sim \frac{1}{2c(\gamma)^2}\left(\frac{1}{a^\top \Sigma_-^{i_-} a} - \frac{1}{a^\top \Sigma_-^{i_+} a}\right).$$

Proposition 3 and 4 follows from $\frac{dv}{d\gamma}\Big/\frac{du}{d\gamma} = \frac{p_-}{p_+}(\beta(\gamma)^{-1} - 1)$, which is a consequence of Eq. (1).

References

D. Avis, D. Bremner, and R. Seidel. How good are convex hull algorithms ? In *Computational Geometry: Theory and Applications*, volume 7, 1997.

F. R. Bach, D. Heckerman, and E. Horvitz. On the path to an ideal ROC curve: considering cost asymmetry in learning classifiers. In *Workshop on Artificial Intelligence and Statistics (AISTATS)*, 2005a.

F. R. Bach, R. Thibaux, and M. I. Jordan. Computing regularization paths for learning multiple kernels. In *Advances in Neural Information Processing Systems, 17*. MIT Press, 2005b.

P. L. Bartlett, M. I. Jordan, and J. D. McAuliffe. Large margin classifiers: convex loss, low noise, and convergence rates. In *Advances in Neural Information Processing Systems, 16*. MIT Press, 2004.

C.L. Blake and C.J. Merz. UCI repository of machine learning databases, 1998.

N. Bleistein and R. A. Handelsman. *Asymptotic Expansions of Integrals*. Dover, 1986.

S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2003.

P. A. Flach. The geometry of ROC space: understanding machine learning metrics through ROC isometrics. In *International Conference on Machine Learning (ICML)*, 2003.

G. H. Golub and C. F. Van Loan. *Matrix Computations*. Johns Hopkins University Press, 1996.

T. Hastie, S. Rosset, R. Tibshirani, and J. Zhu. The entire regularization path for the support vector machine. *Journal of Machine Learning Research*, 5:1391–1415, 2005.

T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning*. Springer-Verlag, 2001.

M. Hollander and D. A. Wolfe. *Nonparametric statistical inference*. John Wiley & Sons, 1999.

R. Kohavi and G. John. Wrappers for feature subset selection. *Artificial Intelligence*, 97(1-2): 273–324, 1997.

I. Maros. *Computational Techniques of the Simplex Method*. Klüwer Academic Publishers, 2002.

A. Y. Ng. Preventing overfitting of cross-validation data. In *International Conference on Machine Learning (ICML)*, 1997.

M. S. Pepe. Receiver operating characteristic methodology. *Journal of the American Statistical Association*, 95(449):308–311, 2000.

J. Platt. Fast training of support vector machines using sequential minimal optimization. In *Advances in Kernel Methods: Support Vector Learning*. MIT Press, 1998.

F. Provost and T. Fawcett. Robust classification for imprecise environments. *Machine Learning Journal*, 42(3):203–231, 2001.

K. Scheinberg. An efficient implementation of an active set method for SVM. *Journal of Machine Learning Research*, to appear, 2006.

B. Schölkopf and A. J. Smola. *Learning with Kernels*. MIT Press, 2002.

J. Shawe-Taylor and N. Cristianini. *Kernel Methods for Pattern Analysis*. Cambridge University Press, 2004.

S. Tong and D. Koller. Restricted Bayes optimal classifiers. In *American Conference on Artificial Intelligence (AAAI-00)*, 2000.

T. Zhang. Statistical behavior and consistency of classification methods based on convex risk minimization. *Annals of Statistics*, 32:56–85, 2004.