Robust Support Vector Machine Training via Convex Outlier Ablation

Linli Xu[*]
University of Waterloo
l5xu@cs.uwaterloo.ca

Koby Crammer
University of Pennsylvania
crammer@cis.upenn.edu

Dale Schuurmans
University of Alberta
dale@cs.ualberta.ca

Abstract

One of the well known risks of large margin training methods, such as boosting and support vector machines (SVMs), is their sensitivity to outliers. These risks are normally mitigated by using a soft margin criterion, such as hinge loss, to reduce outlier sensitivity. In this paper, we present a more direct approach that explicitly incorporates outlier suppression in the training process. In particular, we show how outlier detection can be encoded in the large margin training principle of support vector machines. By expressing a convex relaxation of the joint training problem as a semidefinite program, one can use this approach to robustly train a support vector machine while suppressing outliers. We demonstrate that our approach can yield superior results to the standard soft margin approach in the presence of outliers.

Introduction

The fundamental principle of large margin training, though simple and intuitive, has proved to be one of the most effective estimation techniques devised for classification problems. The simplest version of the idea is to find a hyperplane that correctly separates binary labeled training data with the largest margin, intuitively yielding maximal robustness to perturbation and reducing the risks of future misclassifications. In fact, it has been well established in theory and practice that if a large margin is obtained, the separating hyperplane is likely to have a small misclassification rate on future test examples (Bartlett & Mendelson 2002; Bousquet & Elisseeff 2002; Schoelkopf & Smola 2002; Shawe-Taylor & Cristianini 2004).

Unfortunately, the naive maximum margin principle yields poor results on non-linearly separable data because the solution hyperplane becomes determined by the most misclassified points, causing a breakdown in theoretical and practical performance. In practice, some sort of mechanism is required to prevent training from fixating solely on anomalous data. For the most part, the field appears to have fixated on the soft margin SVM approach to this problem (Cortes & Vapnik 1995), where one minimizes a combination of the inverse squared margin and linear margin violation penalty (hinge loss). In fact, many variants of this approach have been proposed in the literature, including the ν-SVM reformulation (Schoelkopf & Smola 2002).

Unfortunately, the soft margin SVM has serious shortcomings. One drawback is the lack of a probabilistic interpretation of the margin loss, which creates an unintuitive parameter to tune and causes difficulty in modeling overlapping distributions. However, the central drawback we address in this paper is that outlier points are guaranteed to play a maximal role in determining the decision hyperplane, since they tend to have the largest margin loss. In this paper, we modify the standard soft margin SVM scheme with an explicit outlier suppression mechanism.

There have been a few previous attempts to improve the robustness of large margin training to outliers. The theoretical literature has investigated the concept of a robust margin loss that does not increase the penalty after a certain point (Bartlett & Mendelson 2002; Krause & Singer 2004; Mason *et al.* 2000). One problem with these approaches though is that they lose convexity in the training objective, which prevents global optimization. There have also been a few attempts to propose convex training objectives that can mitigate the effect of outliers. Song et al. (2002) formulate a robust SVM objective by scaling the margin loss by the distance from a class centroid, reducing the losses (hence the influence) of points that lie far from their class centroid. Weston and Herbrich (2000) formulate a new training objective based on minimizing a bound on the leave one out cross validation error of the soft margin SVM. We discuss these approaches in more detail below, but one property they share is that they do not attempt to identify outliers, but rather alter the margin loss to reduce the effect of misclassified points.

In this paper we propose a more direct approach to the problem of robust SVM training by formulating outlier detection and removal directly in the standard soft margin framework. We gain several advantages in doing so. First, the robustness of the standard soft margin SVM is improved by explicit outlier ablation. Second, our approach preserves the standard margin loss and thereby retains a direct connection to standard theoretical analyses of SVMs. Third, we obtain the first practical training algorithm for training on the robust hinge loss proposed in the theoretical literature. Finally, outlier detection itself can be a significant benefit.

Although we do not pursue outlier detection as a central

[*] Work performed at the Alberta Ingenuity Centre for Machine Learning, University of Alberta.

goal, it is an important problem in many areas of machine learning and data mining (Aggarwal & Yu 2001; Brodley & Friedl 1996; Fawcett & Provost 1997; Tax 2001; Manevitz & Yoursef 2001). Most work focuses on the unsupervised case where there is no designated class variable, but we focus on the supervised case here.

Background: Soft margin SVMs

We will focus on the standard soft margin SVM for binary classification. In the primal representation the classifier is given by a linear discriminant on input vectors, $h(\mathbf{x}) = \text{sign}(\mathbf{x}^\top \mathbf{w})$, parameterized by a weight vector $\mathbf{w}$. (Note that we drop the scalar offset b for ease of exposition.) Given a training set $(\mathbf{x}_1, y_1), ..., (\mathbf{x}_t, y_t)$ represented as an $n \times t$ matrix of (column) feature vectors, X, and a $t \times 1$ vector of training labels, $\mathbf{y} \in \{-1, +1\}^t$, the goal of soft margin SVM training is to minimize a regularized *hinge loss*, which for example $(\mathbf{x}_i, y_i)$ is given by:

$$\textit{hinge}(\mathbf{w}, \mathbf{x}_i, y_i) \quad = \quad [1 - y_i \mathbf{x}_i^\top \mathbf{w}]_+$$

Here we use the notation $[u]_+ = \max(0, u)$. Let the misclassification error be denoted by

$$\textit{err}(\mathbf{w}, \mathbf{x}_i, y_i) \quad = \quad 1_{(y_i \mathbf{x}_i^\top \mathbf{w} < 0)}$$

Then it is easy to see that the hinge loss gives an upper bound on the misclassification error; see Figure 1.

Proposition 1 $\textit{hinge}(\mathbf{w}, \mathbf{x}, y) \geq \textit{err}(\mathbf{w}, \mathbf{x}, y)$

The hinge loss is a well motivated proxy for misclassification error, which itself is non-convex and NP-hard to optimize (Kearns, Schapire, & Sellie 1992; Hoeffgen, Van Horn, & Simon 1995). To derive the soft margin SVM, let $Y = \text{diag}(\mathbf{y})$ be the diagonal label matrix, and let $\mathbf{e}$ denote the vector of all 1s. One can then write (Hastie *et al.* 2004)

$$\min_{\mathbf{w}} \ \tfrac{\beta}{2}\|\mathbf{w}\|^2 + \textstyle\sum_i [1 - y_i \mathbf{x}_i^\top \mathbf{w}]_+ \qquad (1)$$

$$= \ \min_{\mathbf{w}} \tfrac{\beta}{2}\|\mathbf{w}\|^2 + \mathbf{e}^\top \boldsymbol{\xi} \quad \text{s.t.} \quad \boldsymbol{\xi} \geq \mathbf{e} - YX^\top \mathbf{w}, \ \boldsymbol{\xi} \geq 0 \ (2)$$

$$= \ \max_{\boldsymbol{\alpha}} \ \boldsymbol{\alpha}^\top \mathbf{e} - \tfrac{1}{2\beta} \boldsymbol{\alpha}^\top Y X^\top X Y \boldsymbol{\alpha} \quad \text{s.t.} \quad 0 \leq \boldsymbol{\alpha} \leq 1 \ \ (3)$$

The quadratic program (3) is a dual of (2) and establishes the relationship $\mathbf{w} = XY\boldsymbol{\alpha}/\beta$ between the solutions. The dual classifier can thus be expressed as $h(\mathbf{x}) = \text{sign}(\mathbf{x}^\top XY\boldsymbol{\alpha})$. In the dual, the feature vectors only occur as inner products and therefore can be replaced by a kernel operator $k(\mathbf{x}_i, \mathbf{x}_j)$.

It is instructive to consider how the soft margin solution is affected by the presence of outliers. In general, the soft margin SVM limits the influence of any single training example, since $0 \leq \alpha_i \leq 1$ by (3), and thus the influence of outlier points is bounded. However, the influence of outlier points is not zero. In fact, of all training points, outliers will still retain maximal influence on the solution, since they will normally have the largest hinge loss. This results in the soft margin SVM still being inappropriately drawn toward outlier points, as Figure 2 illustrates.

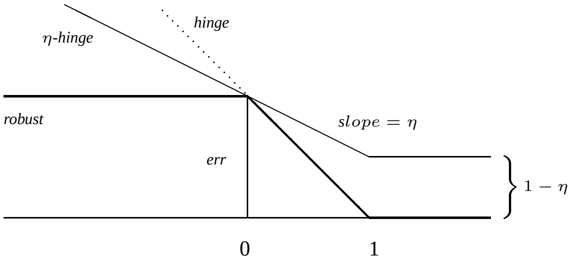

Figure 1: Margin losses as a function of $y\mathbf{x}^\top\mathbf{w}$: **dotted** *hinge*, **bold** *robust*, **thin** η-*hinge*, and **step** *err*. Note that η-*hinge* $\geq$ *robust* $\geq$ *err* for $0 \leq \eta \leq 1$. Also *hinge* $\geq$ *robust*. If $y\mathbf{x}^\top\mathbf{w} \leq 0$, then $\eta = 0$ minimizes η-*hinge*; else $\eta = 1$ minimizes η-*hinge*. Thus $\min_\eta$ η-*hinge* = *robust* for all $y\mathbf{x}^\top\mathbf{w}$.

Robust SVM training

Our main idea in this paper is to augment the soft margin SVM with indicator variables that can remove outliers entirely. The first application of our approach will be to show that outlier indicators can be used to directly minimize the robust hinge loss (Bartlett & Mendelson 2002; Shawe-Taylor & Cristianini 2004). Then we adapt the approach to focus more specifically on outlier identification.

Define a variable η_i for each training example $(\mathbf{x}_i, y_i)$ such that $0 \leq \eta_i \leq 1$, where $\eta_i = 0$ is intended to indicate that example i is an outlier. Assume initially that these outlier indicators are boolean, $\eta_i \in \{0, 1\}$, and known beforehand. Then one could trivially augment the soft SVM criterion (1) by

$$\min_{\mathbf{w}} \ \tfrac{\beta}{2}\|\mathbf{w}\|^2 + \textstyle\sum_i \eta_i [1 - y_i \mathbf{x}_i^\top \mathbf{w}]_+ \qquad (4)$$

In this formulation, no loss is charged for any points where $\eta_i = 0$, and these examples are removed from the solution. One problem with this initial formulation, however, is that $\eta_i [1 - y_i \mathbf{x}_i^\top \mathbf{w}]_+$ is no longer an upper bound on the misclassification error. Therefore, we add a constant term $1 - \eta_i$ to recover an upper bound. Specifically, we define a new loss

$$\eta\textit{-hinge}(\mathbf{w}, \mathbf{x}, y) \quad = \quad \eta \ [1 - y\mathbf{x}^\top \mathbf{w}]_+ + 1 - \eta$$

With this definition one can show for all $0 \leq \eta \leq 1$

Proposition 2 η-*hinge*$(\mathbf{w}, \mathbf{x}, y) \geq \textit{err}(\mathbf{w}, \mathbf{x}, y)$

In fact, this upper bound is very easy to establish; See Figure 1. Similar to (4), minimizing the objective

$$\min_{\mathbf{w}} \ \tfrac{\beta}{2}\|\mathbf{w}\|^2 + \textstyle\sum_i \eta_i\textit{-hinge}(\mathbf{w}, \mathbf{x}_i, y_i) \qquad (5)$$

ignores any points with $\eta_i = 0$ since their loss is a constant.

Now rather than fix $\boldsymbol{\eta}$ ahead of time, we would like to simultaneously optimize $\boldsymbol{\eta}$ and $\mathbf{w}$, which would achieve concurrent outlier detection and classifier training. To facilitate efficient computation, we relax the outlier indicator variables to be $0 \leq \boldsymbol{\eta} \leq 1$. Note that Proposition 2 still applies in this case, and we retain the upper bound on misclassification error for relaxed $\boldsymbol{\eta}$. Thus, we propose the joint objective

$$\min_{\mathbf{w}} \min_{0 \leq \boldsymbol{\eta} \leq 1} \ \tfrac{\beta}{2}\|\mathbf{w}\|^2 + \textstyle\sum_i \eta_i\textit{-hinge}(\mathbf{w}, \mathbf{x}_i, y_i) \qquad (6)$$

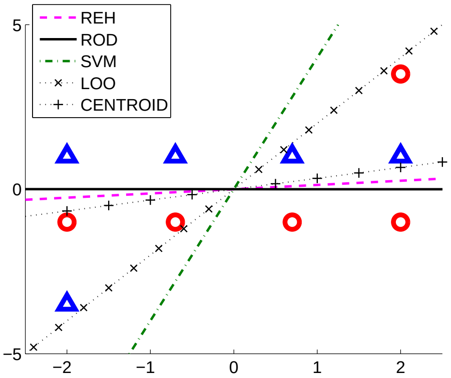

Figure 2: Illustrating behavior given outliers.

This objective yields a convex quadratic program in $\mathbf{w}$ given $\boldsymbol{\eta}$, and a linear program in $\boldsymbol{\eta}$ given $\mathbf{w}$. However, (6) is not jointly convex in $\mathbf{w}$ and $\boldsymbol{\eta}$, so alternating minimization is not guaranteed to yield a global solution. Instead, we will derive a semidefinite relaxation of the problem that removes all local minima below. However, before deriving a convex relaxation of (6) we first establish a very useful and somewhat surprising result: that minimizing (6) is equivalent to minimizing the regularized robust hinge loss from the theoretical literature.

Robust hinge loss

The robust hinge loss has often been noted as a superior alternative to the standard hinge loss (Krause & Singer 2004; Mason *et al.* 2000). This loss is given by

$$\mathit{robust}(\mathbf{w},\mathbf{x},y) \quad = \quad \min(1, \mathit{hinge}(\mathbf{w},\mathbf{x},y))$$

and is illustrated in bold in Figure 1.

The main advantage of robust over regular hinge loss is that the robust loss is bounded, meaning that outlier examples cannot have an effect on the solution beyond that of any other misclassified point. The robust hinge loss also retains an upper bound on the misclassification error, as shown in Figure 1. Given such a loss, one can pose the objective

$$\min_{\mathbf{w}} \ \tfrac{\beta}{2}\|\mathbf{w}\|^2 + \textstyle\sum_i \mathit{robust}(\mathbf{w},\mathbf{x}_i,y_i) \tag{7}$$

Unfortunately, even though robust hinge loss has played a significant role in generalization theory (Bartlett & Mendelson 2002; Shawe-Taylor & Cristianini 2004), the minimization objective (7) has not been often applied in practice because it is non-convex, and leads to significant difficulties in optimization (Krause & Singer 2004; Mason *et al.* 2000).

We can now offer an alternative characterization of robust hinge loss, by showing that it is equivalent to minimizing the η-*hinge* loss introduced earlier. This facilitates a new approach to the training problem that we introduce below. First, the η-*hinge* loss can be easily shown to be an upper bound on the robust hinge loss for all η.

Proposition 3 $\eta\text{-}\mathit{hinge}(\mathbf{w},\mathbf{x},y) \ \geq \ \mathit{robust}(\mathbf{w},\mathbf{x},y) \ \geq \ \mathit{err}(\mathbf{w},\mathbf{x},y)$

Second, minimizing the η-*hinge* loss with respect to η gives the same result as the robust hinge loss

Proposition 4 $\min_{\boldsymbol{\eta}} \eta\text{-}\mathit{hinge}(\mathbf{w},\mathbf{x},y) = \mathit{robust}(\mathbf{w},\mathbf{x},y)$

Both propositions are straightforward, but can be seen best by examining Figure 1. From these two propositions, one can immediately establish the following equivalence.

Theorem 1

$$\begin{aligned} &\min_{\mathbf{w}} \min_{0\leq\boldsymbol{\eta}\leq 1} \ \tfrac{\beta}{2}\|\mathbf{w}\|^2 + \textstyle\sum_i \eta_i\text{-}\mathit{hinge}(\mathbf{w},\mathbf{x}_i,y_i) \\ &\quad = \ \min_{\mathbf{w}} \ \tfrac{\beta}{2}\|\mathbf{w}\|^2 + \textstyle\sum_i \mathit{robust}(\mathbf{w},\mathbf{x}_i,y_i) \end{aligned}$$

Moreover, the minimizers are equivalent.

Proof Define $f_{\mathit{rob}}(\mathbf{w}) = \frac{\beta}{2}\|\mathbf{w}\|^2 + \sum_i \mathit{robust}(\mathbf{w},\mathbf{x}_i,y_i)$; $f_{\mathit{hng}}(\mathbf{w},\boldsymbol{\eta}) = \frac{\beta}{2}\|\mathbf{w}\|^2 + \sum_i \eta_i\text{-}\mathit{hinge}(\mathbf{w},\mathbf{x}_i,y_i)$; $\mathbf{w}_r = \arg\min_{\mathbf{w}} f_{\mathit{rob}}(\mathbf{w})$; $(\mathbf{w}_h,\boldsymbol{\eta}_h) = \arg\min_{\mathbf{w},0\leq\boldsymbol{\eta}\leq 1} f_{\mathit{hng}}(\mathbf{w},\boldsymbol{\eta})$; $\eta_r = \arg\min_{0\leq\boldsymbol{\eta}\leq 1} f_{\mathit{hng}}(\mathbf{w}_r,\boldsymbol{\eta})$. Then from Proposition 3

$$\begin{aligned} \min_{\mathbf{w}} \min_{0\leq\boldsymbol{\eta}\leq 1} f_{\mathit{hng}}(\mathbf{w},\boldsymbol{\eta}) \quad &= \quad \min_{0\leq\boldsymbol{\eta}\leq 1} f_{\mathit{hng}}(\mathbf{w}_h,\boldsymbol{\eta}) \\ &\geq \ f_{\mathit{rob}}(\mathbf{w}_h) \ \geq \ \min_{\mathbf{w}} f_{\mathit{rob}}(\mathbf{w}) \end{aligned}$$

Conversely, by Proposition 4 we have

$$\begin{aligned} \min_{\mathbf{w}} f_{\mathit{rob}}(\mathbf{w}) \quad &= \quad f_{\mathit{rob}}(\mathbf{w}_r) \\ = \ \min_{0\leq\boldsymbol{\eta}\leq 1} f_{\mathit{hng}}(\mathbf{w}_r,\boldsymbol{\eta}) \ &\geq \ \min_{\mathbf{w}} \min_{0\leq\boldsymbol{\eta}\leq 1} f_{\mathit{hng}}(\mathbf{w},\boldsymbol{\eta}) \end{aligned}$$

Thus, the two objectives achieve equal values. Finally, the minimizers $\mathbf{w}_r$ and $\mathbf{w}_h$ must be interchangeable, since $f_{\mathit{rob}}(\mathbf{w}_r) = f_{\mathit{hng}}(\mathbf{w}_r,\boldsymbol{\eta}_r) \geq f_{\mathit{hng}}(\mathbf{w}_h,\boldsymbol{\eta}_h) = f_{\mathit{rob}}(\mathbf{w}_h) \geq f_{\mathit{rob}}(\mathbf{w}_r)$, showing all values are equal. ■

Therefore minimizing regularized robust loss is equivalent to minimizing the regularized η-*hinge* loss we introduced. Previously, we observed that the regularized η-*hinge* objective can be minimized by alternating minimization on $\mathbf{w}$ and $\boldsymbol{\eta}$. Unfortunately, as Figure 1 illustrates, the minimization of $\boldsymbol{\eta}$ given $\mathbf{w}$ always results in boolean solutions that set $\eta_i = 0$ for all misclassified examples and $\eta_i = 1$ for correct examples. Such an approach immediately gets trapped in local minima. Therefore, a better computational approach is required. To develop an efficient training technique for robust loss, we now derive a semidefinite relaxation of the problem.

Convex relaxation

To derive a convex relaxation of (6) we need to work in the dual of (5). Let $N = \mathrm{diag}(\boldsymbol{\eta})$ be the diagonal matrix of η values, and let $\circ$ denote componentwise multiplication. We then obtain

Proposition 5 *For fixed* $\boldsymbol{\eta}$

$$\begin{aligned} &\min_{\mathbf{w}} \ \tfrac{\beta}{2}\|\mathbf{w}\|^2 + \textstyle\sum_i \eta_i\text{-}\mathit{hinge}(\mathbf{w},\mathbf{x}_i,y_i) \\ &\quad = \ \min_{\mathbf{w},\boldsymbol{\xi}} \ \tfrac{\beta}{2}\|\mathbf{w}\|^2 + \mathbf{e}^\top\boldsymbol{\xi} + \mathbf{e}^\top(\mathbf{e}-\boldsymbol{\eta}) \quad \textit{subject to} \qquad (8) \\ &\qquad\qquad\qquad \boldsymbol{\xi} \geq 0, \ \ \boldsymbol{\xi} \geq N(\mathbf{e} - YX^\top\mathbf{w}) \\ &\quad = \ \max_{\boldsymbol{\alpha}} \ \boldsymbol{\eta}^\top(\boldsymbol{\alpha}-\mathbf{e}) - \tfrac{1}{2\beta}\boldsymbol{\alpha}^\top(X^\top X \circ \mathbf{y}\mathbf{y}^\top \circ \boldsymbol{\eta}\boldsymbol{\eta}^\top)\boldsymbol{\alpha} + t \\ &\qquad\qquad\qquad \textit{subject to} \ \ 0 \leq \boldsymbol{\alpha} \leq 1 \end{aligned}$$

Proof The Lagrangian of (8) is $L_1 = \frac{\beta}{2}\mathbf{w}^\top\mathbf{w} + \mathbf{e}^\top\boldsymbol{\xi} + \boldsymbol{\alpha}^\top(\boldsymbol{\eta} - NYX^\top\mathbf{w} - \boldsymbol{\xi}) - \boldsymbol{\nu}^\top\boldsymbol{\xi} + \mathbf{e}^\top(\mathbf{e} - \boldsymbol{\eta})$ such that $\boldsymbol{\alpha} \geq 0$, $\boldsymbol{\nu} \geq 0$. Computing the gradient with respect to $\boldsymbol{\xi}$ yields $dL_1/d\boldsymbol{\xi} = \mathbf{e} - \boldsymbol{\alpha} - \boldsymbol{\nu} = 0$, which implies $\boldsymbol{\alpha} \leq \mathbf{e}$. The Lagrangian can therefore be equivalently expressed by $L_2 = \frac{\beta}{2}\mathbf{w}^\top\mathbf{w} + \boldsymbol{\alpha}^\top(\boldsymbol{\eta} - NYX^\top\mathbf{w}) + \mathbf{e}^\top(\mathbf{e} - \boldsymbol{\eta})$ subject to $0 \leq \boldsymbol{\alpha} \leq 1$. Finally, taking the gradient with respect to $\mathbf{w}$ yields $dL_2/d\mathbf{w} = \beta\mathbf{w} - XYN\boldsymbol{\alpha} = 0$, which implies $\mathbf{w} = XYN\boldsymbol{\alpha}/\beta$. Substituting back into L_2 yields the result. ∎

We can subsequently reformulate the joint objective as

Corollary 1

$$\min_{0\leq\boldsymbol{\eta}\leq 1} \min_{\mathbf{w}} \frac{\beta}{2}\|\mathbf{w}\|^2 + \textstyle\sum_i \eta_i\text{-}\textit{hinge}(\mathbf{w}, \mathbf{x}_i, y_i) \tag{9}$$

$$= \min_{0\leq\boldsymbol{\eta}\leq 1} \max_{0\leq\boldsymbol{\alpha}\leq 1} \boldsymbol{\eta}^\top(\boldsymbol{\alpha} - \mathbf{e}) - \frac{1}{2\beta}\boldsymbol{\alpha}^\top(X^\top X \circ \mathbf{y}\mathbf{y}^\top \circ \boldsymbol{\eta}\boldsymbol{\eta}^\top)\boldsymbol{\alpha} + t$$

The significance of this reformulation is that it allows us to express the inner optimization as a *maximum*, which allows a natural convex relaxation for the outer minimization. The key observation is that $\boldsymbol{\eta}$ appears in the inner maximization only as $\boldsymbol{\eta}$ and the symmetric matrix $\boldsymbol{\eta}\boldsymbol{\eta}^\top$. If we create a matrix variable $M = \boldsymbol{\eta}\boldsymbol{\eta}^\top$, we can re-express the problem as a maximum of *linear* functions of $\boldsymbol{\eta}$ and M, yielding a convex objective in $\boldsymbol{\eta}$ and M (Boyd & Vandenberghe 2004)

$$\min_{0\leq\boldsymbol{\eta}\leq 1,\, M=\boldsymbol{\eta}\boldsymbol{\eta}^\top} \max_{0\leq\boldsymbol{\alpha}\leq 1} \boldsymbol{\eta}^\top(\boldsymbol{\alpha} - \mathbf{e}) - \frac{1}{2\beta}\boldsymbol{\alpha}^\top(G \circ M)\boldsymbol{\alpha}$$

Here $G = X^\top X \circ \mathbf{y}\mathbf{y}^\top$. The only problem that remains is that $M = \boldsymbol{\eta}\boldsymbol{\eta}^\top$ is a non-convex quadratic constraint. This constraint forces us to make our only approximation: we relax the equality to $M \succeq \boldsymbol{\eta}\boldsymbol{\eta}^\top$, yielding a convex problem

$$\min_{0\leq\boldsymbol{\eta}\leq 1} \min_{M\succeq\boldsymbol{\eta}\boldsymbol{\eta}^\top} \max_{0\leq\boldsymbol{\alpha}\leq 1} \boldsymbol{\eta}^\top(\boldsymbol{\alpha} - \mathbf{e}) - \frac{1}{2\beta}\boldsymbol{\alpha}^\top(G \circ M)\boldsymbol{\alpha} \tag{10}$$

This problem can be equivalently expressed as a semidefinite program.

Theorem 2 *Solving (10) is equivalent to solving*

$$\min_{\boldsymbol{\eta}, M\boldsymbol{\nu}, \boldsymbol{\omega}, \delta} \delta \quad \textit{s.t.} \;\; \boldsymbol{\nu} \geq 0,\; \boldsymbol{\omega} \geq 0,\; 0 \leq \boldsymbol{\eta} \leq 1,\; M \succeq \boldsymbol{\eta}\boldsymbol{\eta}^\top,$$
$$\begin{bmatrix} G \circ M & \boldsymbol{\eta} + \boldsymbol{\nu} - \boldsymbol{\omega} \\ (\boldsymbol{\eta} + \boldsymbol{\nu} - \boldsymbol{\omega})^\top & \frac{2}{\beta}(\delta - \boldsymbol{\omega}^\top\mathbf{e} + \boldsymbol{\eta}^\top\mathbf{e}) \end{bmatrix} \succeq 0$$

Proof Objective (10) is equivalent to minimizing a gap variable δ with respect to $\boldsymbol{\eta}$ and M subject to $\delta \geq \boldsymbol{\eta}^\top(\boldsymbol{\alpha} - \mathbf{e}) - \boldsymbol{\alpha}^\top(G \circ M/2\beta)\boldsymbol{\alpha}$ for all $0 \leq \boldsymbol{\alpha} \leq 1$. Consider the right hand maximization in $\boldsymbol{\alpha}$. By introducing Lagrange multipliers for the constraints on $\boldsymbol{\alpha}$ we obtain $L_1 = \boldsymbol{\eta}^\top(\boldsymbol{\alpha} - \mathbf{e}) - \boldsymbol{\alpha}^\top(G \circ M/2\beta)\boldsymbol{\alpha} + \boldsymbol{\nu}^\top\boldsymbol{\alpha} + \boldsymbol{\omega}^\top(\mathbf{e} - \boldsymbol{\alpha})$, to be maximized in $\boldsymbol{\alpha}$ and minimized in $\boldsymbol{\nu}, \boldsymbol{\omega}$ subject to $\boldsymbol{\nu} \geq 0$, $\boldsymbol{\omega} \geq 0$. The gradient with respect to $\boldsymbol{\alpha}$ is given by $dL_1/d\boldsymbol{\alpha} = \boldsymbol{\eta} - (G \circ M/\beta)\boldsymbol{\alpha} + \boldsymbol{\nu} - \boldsymbol{\omega} = 0$, yielding $\boldsymbol{\alpha} = \beta(G \circ M)^{-1}(\boldsymbol{\eta} + \boldsymbol{\nu} - \boldsymbol{\omega})$. Substituting this back into L_1 yields $L_2 = \boldsymbol{\omega}^\top\mathbf{e} - \boldsymbol{\eta}^\top\mathbf{e} + \beta/2(\boldsymbol{\eta} + \boldsymbol{\nu} - \boldsymbol{\omega})^\top(G \circ M)^{-1}(\boldsymbol{\eta} + \boldsymbol{\nu} - \boldsymbol{\omega})$. Finally, we obtain the result by applying the Schur complement to $\delta - L_2 \geq 0$. ∎

This formulation as a semidefinite program admits a polynomial time training algorithm (Nesterov & Nimirovskii 1994; Boyd & Vandenberghe 2004). We refer to this algorithm as the robust η-hinge (REH) SVM. One minor improvement is that the relaxation (10) can be tightened slightly by using the stronger constraint $M \succeq \boldsymbol{\eta}\boldsymbol{\eta}^\top$, $\text{diag}(M) = \boldsymbol{\eta}$ on M, which would still be valid in the discrete case.

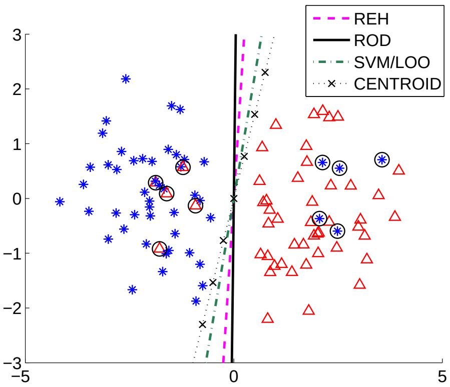

Figure 3: Gaussian blobs, with outliers.

Explicit outlier detection

Note that the technique developed above does not actually identify outliers, but rather just improves robustness against the presence of outliers. That is, a small value of η_i in the computed solution does not necessarily imply that example i is an outlier. To explicitly identify outliers, one needs to be able to distinguish between true outliers and points that are just misclassified because they are in a class overlap region. To adapt our technique to explicitly identify outliers, we reconsider a joint optimization of the original objective (4), but now add a constraint that at least a certain proportion ρ of the training examples must not be considered as outliers

$$\min_{\mathbf{w}} \min_{0\leq\boldsymbol{\eta}\leq 1} \frac{\beta}{2}\|\mathbf{w}\|^2 + \textstyle\sum_i \eta_i[1 - y_i\mathbf{x}_i^\top\mathbf{w}]_+ \;\; \text{s.t.} \;\; \mathbf{e}^\top\boldsymbol{\eta} \geq \rho t$$

The difference is that we drop the extra $1 - \eta_i$ term in the η-*hinge* loss and add the proportion ρ constraint. The consequence is that we lose the upper bound on the misclassification error, but the optimization is now free to drop a proportion $1 - \rho$ of the points without penalty, to minimize hinge loss. The points that are dropped should correspond to ones that would have obtained the largest hinge loss; i.e. the outliers. Following the same steps as above, one can derive a semidefinite relaxation of this objective that allows a reasonable training algorithm. We refer to this method as the robust outlier detection (ROD) algorithm. Figure 3 shows anecdotally that this outlier detection works well in a simple synthetic setting, discovering a much better classifier than the soft margin SVM (hinge loss), while also identifying the outlier points. The robust SVM algorithm developed above also produces good results in this case, but does not identify outliers.

Comparison to existing techniques

Before discussing experimental results, we briefly review related approaches to robust SVM training. Interestingly, the original proposal for soft margin SVMs (Cortes & Vapnik 1995) considered alternative losses based on the transformation $\textit{loss}(\mathbf{w}, \mathbf{x}, y) = \textit{hinge}(\mathbf{w}, \mathbf{x}, y)^p$. Unfortunately, choosing $p > 1$ exaggerates the largest losses and makes the technique more sensitive to outliers. Choosing $p < 1$ improves robustness, but creates a non-convex training problem.

There have been a few more recent attempts to improve the robustness of soft margin SVMs to outliers. Song et al. (2002) modify the margin penalty by shifting the loss according to the distance from the class centroid

$$\min_{\mathbf{w}} \ \tfrac{1}{2}\|\mathbf{w}\|^2 + \textstyle\sum_i [1 - y_i \mathbf{x}_i^\top \mathbf{w} - \lambda \|\mathbf{x}_i - \boldsymbol{\mu}_{y_i}\|^2]_+$$

where $\boldsymbol{\mu}_{y_i}$ is the centroid for class $y_i \in \{-1, +1\}$. Intuitively, examples that are far away from their class centroid will have their margin losses automatically reduced, which diminishes their influence on the solution. If the outliers are indeed far from the class centroid the technique is reasonable; see Figure 2. Unfortunately, the motivation is heuristic and loses the upper bound on misclassification error, which blocks any simple theoretical justification.

Another interesting proposal for robust SVM training is the leave-one-out (LOO) SVM and its extension to the adaptive margin SVM (Weston & Herbrich 2000). The LOO SVM minimizes the leave-one-out error bound on dual soft margin SVMs, derived by Jaakkola and Haussler (1999). The bound shows that the misclassification error achieved on a single example i by training a soft margin SVM on the remaining $t-1$ data points is at most $\textit{loo\_err}(\mathbf{x}_i, y_i) \leq y_i \sum_{j \neq i} \alpha_j y_j \mathbf{x}_i^\top \mathbf{x}_j$, where $\boldsymbol{\alpha}$ is the dual solution trained on the entire data set. Weston and Herbrich (2000) propose to directly minimize the upper bound on the *loo_err*, leading to

$$\min_{\boldsymbol{\alpha} \geq 0} \ \textstyle\sum_i [1 - y_i \mathbf{x}_i^\top X Y \boldsymbol{\alpha} + \alpha_i \|\mathbf{x}_i\|^2]_+ \qquad (11)$$

Although this objective is hard to interpret as a regularized margin loss, it is closely related to a standard form of soft margin SVM using a modified regularizer

$$\min_{\boldsymbol{\alpha} \geq 0} \ \textstyle\sum_i \alpha_i \|\mathbf{x}_i\|^2 + \sum_i [1 - y_i \mathbf{x}_i^\top X Y \boldsymbol{\alpha}]_+$$

The objective (11) implicitly reduces the influence of outliers, since training examples contribute to the solution only in terms of how well they help predict the labels of *other* training examples. This approach is simple and elegant. Nevertheless, its motivation remains a bit heuristic: (11) does not give a bound on the leave-one-out error of the LOO SVM technique itself, but rather minimizes a bound on the leave-one-out error of another algorithm (soft margin SVM) that was not run on the data. Consequently, the technique is hard to interpret and requires novel theoretical analysis. It can also give anomalous results, as Figure 2 indicates.

Experimental results

We conducted a series of experiments on synthetic and real data sets to compare the robustness of the various SVM training methods, and also to investigate the outlier detection capability of our approach. We implemented our training methods using SDPT3 (Toh, Todd, & Tutuncu 1999) to solve the semidefinite programs.

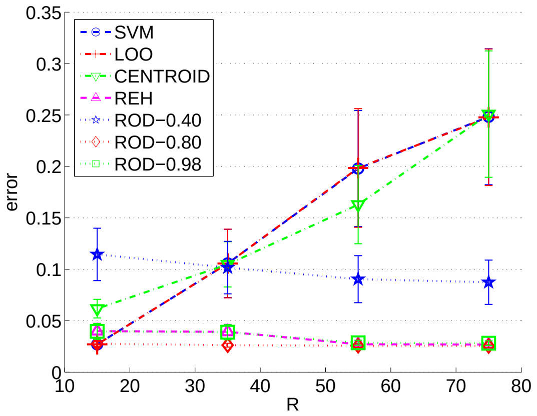

Figure 4: Synthetic results: test error as a function of noise level.

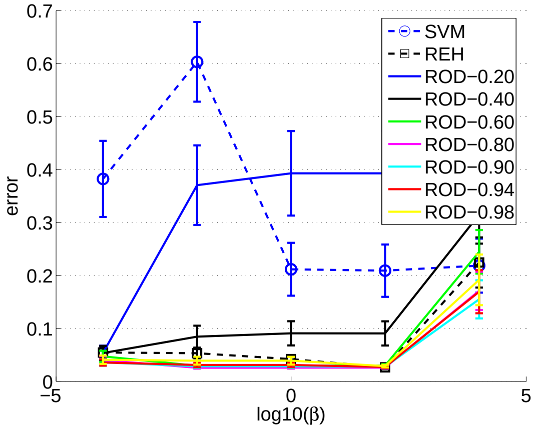

Figure 5: Synthetic results: test error as a function of the regularization parameter β.

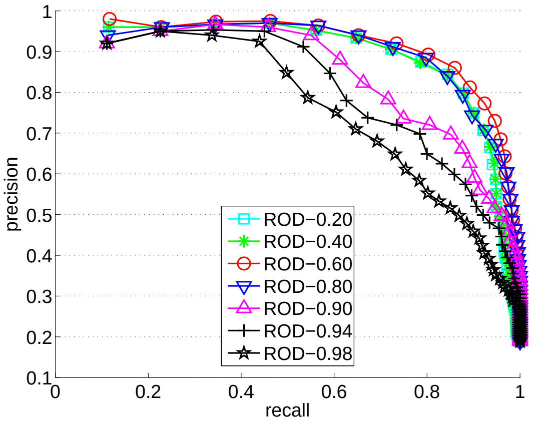

Figure 6: Synthetic results: recall-precision curves for the robust outlier detection algorithm (ROD).

The first experiments were conducted on synthetic data and focused on measuring generalization given outliers, as well as the robustness of the algorithm to their parameters. We assigned one Gaussian per class, with the first given by $\mu = (3, -3)$ and $\Sigma = \begin{pmatrix} 20 & 16 \\ 16 & 20 \end{pmatrix}$ and the second by $-\mu$ and Σ. Since the two Gaussians overlap, the Bayes error is 2.2%. We added outliers to the training set by drawing examples uniformly from a ring with inner-radius of R and outer-radius of $R + 1$, where R was set to one of the values $15, 35, 55, 75$. These examples were labeled randomly with even probability. In all experiments, the training set contained 50 examples: 20 from each Gaussian and 10 from the ring. The test set contained $1,000$ examples from each class. Here the examples from the ring caused about 10% outliers.

We repeated all the experiments 50 times, drawing a training set and a test set every repetition. All the results reported are averaged over the 50 runs, with a 95% confidence interval. We compared the performance of standard soft margin SVM, robust η-hinge SVM (REH) and the robust outlier detector SVM (ROD). All algorithms were run with the generalization tradeoff parameter set to one of five possible values: $\beta = 10^{-4}, 10^{-2}, 10^{0}, 10^{2}, 10^{4}$. The robust outlier detector SVM was run with outlier parameter set to one of seven possible values, $\rho = 0.2, 0.4, 0.6, 0.8, 0.9, 0.94, 0.98$.

Figure 4 shows the results for the two versions of our robust hinge loss training versus soft margin SVM, LOO SVM, and centroid SVM. For centroid SVM we used 8 values for the λ parameter and chose the *best* over the test-set. For all other methods we used the best value of β for standard SVM over the test set. The x-axis is the noise level indicated by the radius R and the y-axis is test error.

These results confirm that the standard soft margin SVM is sensitive to the presence of outliers, as its error rate increases significantly when the radius of the ring increases. By contrast, the robust hinge SVM is not as affected by label noise on distant examples. This is illustrated both in the value of the mean test error and the standard deviation. Here the outlier detection algorithm with $\rho = 0.80$ achieved the best test error, and robust η-hinge SVM second best.

Figure 5 shows the test error as a function of β for all methods for high noise level $R = 55$. From the plot we can draw a few more conclusions: First, when β is close to zero the value of ρ does not affect performance very much. But otherwise, if the value of ρ is too small, then the performance degrades. Third, the robust methods are generally less sensitive to the value of the regularization parameter β. Fourth, if β is very high then it seems that the robust methods converge to the standard SVM.

We also evaluated the outlier detection algorithm as follows. Since the identity of the best linear classifier is known, we identified all misclassified examples and ordered the examples using the η values assigned by the ROD training algorithm. We compute the recall and precision using this ordering and averaged over all 50 runs. Figure 6 shows the precision versus recall for the outlier detection algorithm (ROD) for various values of ρ and for the minimal value of β. As we can see from the plot, if ρ is too large (i.e. we guess that the number of outliers is smaller than their actual number), the detection level is low and the F-measure is about 0.75. For all other values of ρ, we get an F-measure of about 0.85.

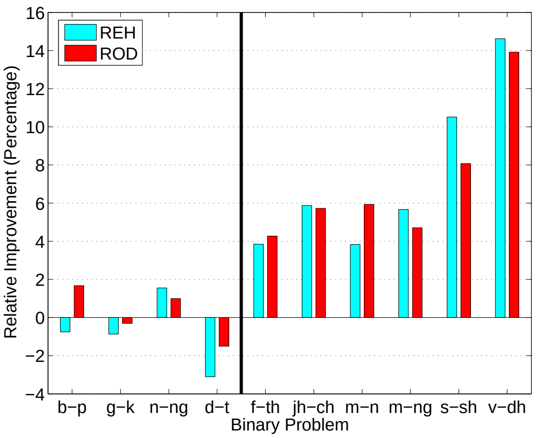

Figure 7: Relative improvement of the robust algorithms over standard soft SVM for the speech data.

A few further comments, which unfortunately rely on plots that are not included in the paper due to lack of space: First, when β is large we can achieve better F-measures by tuning ρ. Second, we found two ways to set ρ. The simple method is to perform cross-validation using the training data and to set ρ to the value that minimized the averaged error. However, we found an alternative method that worked well in practice. If ρ is large, then the graph of sorted η values attains many values near one (corresponding to non-outlier examples) before decreasing to zero for outliers. However, if ρ is small, then all values of η fall below one. Namely, there is a second order phase transition in the maximal value of η, and this phase transition occurs at the value of ρ which corresponds to the true number of outliers. We are investigating a theoretical characterization of this phenomenon.

Given these conclusions, we proceed with experiments on real data. We conducted experiments on the TIMIT phone classification task. Here we used experimental setup similar to (Gunawardana *et al.* 2005) and mapped the 61 phonetic labels into 48 classes. We then picked 10 pairs of classes to construct binary classification tasks. We focused mainly on unvoiced phonemes, whose instantiations have many outliers since there is no harmonic underlying source. The ten binary classification problems are identified by a pair of phoneme symbols (one or two Roman letters). For each of the ten pairs we picked 50 random examples from each class, yielding a training set of size 100. Similarly, for test, we picked $2,500$ random examples from each class and generated a test set of size $5,000$. Our preprocessor computed mel-frequency cepstral coefficients (MFCCs) with 25ms windows at a 10ms frame rate. We retained the first 13 MFCC coefficients of each frame, along with their first and second time derivatives, and the energy and its first derivative. These coefficient vectors (of dimension 41) were whitened using PCA. A standard representation of speech phonemes is a multivariate Gaussian, which uses the first order and second order interaction between the vector components. We thus represented each phoneme using a feature vector of dimension 902 using all the first order coefficients (41) and the second order coefficients (861).

For each problem we first ran the soft margin SVM and set β using five-fold-cross validation. We then used this β for all runs of the robust methods. The results, summarized in Figure 7, show the relative test error between SVM and the two robust SVM algorithms. Formally, each bar is proportional to $(\epsilon_s - \epsilon_r)/\epsilon_s$, where $\epsilon_s(\epsilon_r)$ is the test error. The results are ordered by their statistical significance for robust hinge SVM (REH) according to McNemar test—from the least significant results (left) to the most significant (right). All the results right to the black vertical line are significant with 95% confidence for *both* algorithms. Here we see that the robust SVM methods achieve significantly better results than the standard SVM in six cases, while the differences are insignificant in four cases. (The difference in performance between the two robust algorithms is not significant.) We also ran the other two methods (LOO SVM and centroid SVM) on this data, and found that they performed worse than the standard SVM in 9 out of 10 cases, and always worse than the two robust SVMs.

Conclusion

In this paper we proposed a new form of robust SVM training that is based on identifying and eliminating outlier training examples. Interestingly, we found that our principle provided a new but equivalent formulation to the robust hinge loss often considered in the theoretical literature. Our alternative characterization allowed us to derive the first practical training procedure for this objective, based on a semidefinite relaxation. The resulting training procedure demonstrates superior robustness to outliers than standard soft margin SVM training, and yields generalization improvements in synthetic and real data sets. A useful side benefit of the approach, with some modification, is the ability to explicitly identify outliers as a byproduct of training.

The main drawback of the technique currently is computational cost. Although algorithms for semidefinite programming are still far behind quadratic and linear programming techniques in efficiency, semidefinite programming is still theoretically polynomial time. Current solvers are efficient enough to allow us to train on moderate data sets of a few hundred points. An important direction for future work is to investigate alternative approximations that can preserve the quality of the semidefinite solutions, but reduce run time.

There are many extensions of this work we are pursuing. The robust loss based on $\boldsymbol{\eta}$ indicators is generally applicable to any SVM training algorithm, and we are investigating the application of our technique to multi-class SVMs, one-class SVMs, regression, and ultimately to structured predictors.

Acknowledgments

Research supported by the Alberta Ingenuity Centre for Machine Learning, NSERC, MITACS, CFI, and the Canada Research Chairs program.

References

Aggarwal, C., and Yu, P. 2001. Outlier detection for high dimensional data. In *Proceedings SIGMOD*.

Bartlett, P., and Mendelson, S. 2002. Rademacher and Gaussian complexities: Risk bounds and structural results. *Journal of Machine Learning Research* 3.

Bousquet, O., and Elisseeff, A. 2002. Stability and generalization. *Journal of Machine Learning Research* 2.

Boyd, S., and Vandenberghe, L. 2004. *Convex Optimization*. Cambridge U. Press.

Brodley, C., and Friedl, M. 1996. Identifying and eliminating mislabeled training instances. In *Proceedings AAAI*.

Cortes, C., and Vapnik, V. 1995. Support vector networks. *Machine Learning* 20.

Fawcett, T., and Provost, F. 1997. Adaptive fraud detection. *Data Mining and Knowledge Discovery* 1.

Gunawardana, A.; Mahajan, M.; Acero, A.; and C., P. J. 2005. Hidden conditional random fields for phone classification. In *Proceedings of ICSCT*.

Hastie, T.; Rosset, S.; Tibshirani, R.; and Zhu, J. 2004. The entire regularization path for the support vector machine. *Journal of Machine Learning Research* 5.

Hoeffgen, K.; Van Horn, K.; and Simon, U. 1995. Robust trainability of single neurons. *JCSS* 50(1).

Jaakkola, T., and Haussler, D. 1999. Probabilistic kernel regression methods. In *Proceedings AISTATS*.

Kearns, M.; Schapire, R.; and Sellie, L. 1992. Toward efficient agnostic leaning. In *Proceedings COLT*.

Krause, N., and Singer, Y. 2004. Leveraging the margin more carefully. In *Proceedings ICML*.

Manevitz, L., and Yoursef, M. 2001. One-class svms for document classification. *Journal of Machine Learning Research* 2.

Mason, L.; Baxter, J.; Bartlett, P.; and Frean, M. 2000. Functional gradient techniques for combining hypotheses. In *Advances in Large Margin Classifiers*. MIT Press.

Nesterov, Y., and Nimirovskii, A. 1994. *Interior-Point Polynomial Algorithms in Convex Programming*. SIAM.

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

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

Song, Q.; Hu, W.; and Xie, W. 2002. Robust support vector machine with bullet hole image classification. *IEEE Trans. Systems, Man and Cybernetics C* 32(4).

Tax, D. 2001. *One-class classification; Concept-learning in the absence of counter-examples*. Ph.D. Dissertation, Delft University of Technology.

Toh, K.; Todd, M.; and Tutuncu, R. 1999. SDPT3–a Matlab software package for semidefinite programming. *Optimization Methods and Software* 11.

Weston, J., and Herbrich, R. 2000. Adaptive margin support vector machines. In *Advances in Large Margin Classifiers*. MIT Press.