Journal of Machine Learning Research 8 (2007) 813-839 Submitted 5/06; Revised 2/07; Published 4/07

Gini Support Vector Machine: Quadratic Entropy Based Robust Multi-Class Probability Regression

Shantanu Chakrabartty SHANTANU@MSU.EDU
Department of Electrical and Computer Engineering
Michigan State University
East Lansing, MI 48824, USA

Gert Cauwenberghs GERT@UCSD.EDU
Division of Biological Sciences
University of California San Diego
La Jolla, CA 92093-0357, USA

Editor: Alex Smola

Abstract

Many classification tasks require estimation of output class probabilities for use as confidence scores or for inference integrated with other models. Probability estimates derived from large margin classifiers such as support vector machines (SVMs) are often unreliable. We extend SVM large margin classification to *Gini*SVM maximum entropy multi-class probability regression. *Gini*SVM combines a quadratic (Gini-Simpson) entropy based agnostic model with a kernel based similarity model. A form of Huber loss in the *Gini*SVM primal formulation elucidates a connection to robust estimation, further corroborated by the impulsive noise filtering property of the reverse water-filling procedure to arrive at normalized classification margins. The *Gini*SVM normalized classification margins directly provide estimates of class conditional probabilities, approximating kernel logistic regression (KLR) but at reduced computational cost. As with other SVMs, *Gini*SVM produces a sparse kernel expansion and is trained by solving a quadratic program under linear constraints. *Gini*SVM training is efficiently implemented by sequential minimum optimization or by growth transformation on probability functions. Results on synthetic and benchmark data, including speaker verification and face detection data, show improved classification performance and increased tolerance to imprecision over soft-margin SVM and KLR.

Keywords: support vector machines, large margin classifiers, kernel regression, probabilistic models, quadratic entropy, Gini index, growth transformation

1. Introduction

Support vector machines (SVMs) have gained much popularity in the machine learning community as versatile tools for classification and regression from sparse data (Boser et al., 1992; Vapnik, 1995; Burges, 1998; Schölkopf et al., 1998). The foundations of SVMs are rooted in statistical learning theory (Vapnik, 1995) with also connections to regularization theory (Girosi et al., 1995; Pontil and Verri, 1998a). The principle of structural risk minimization provides bounds on generalization performance which make SVMs well suited for applications with sparse training data (Joachims, 1997; Oren et al., 1997; Pontil and Verri, 1998b).

Several classification problems in machine learning require estimation of multi-class output probabilities. Besides their use as confidence scores in classification, the class probability estimates can also be used in combination with other probabilistic models such as hidden Markov models for inference across graphs. For instance text-independent speaker verification systems require normalized classifier scores to be integrated over several speech frames in an utterance (Auckenthaler et al., 2000) to arrive at global acceptance/rejection scores. Even though SVMs have been successfully applied for the task of speaker verification (Schmidt and Gish, 1996; Gu and Thomas, 2001), the cumulative scores generated by SVMs are susceptible to corruption by impulse noise, which increases false acceptance rate.

Multi-class extensions to SVM classification have been formulated, based on 'one vs. all' (Weston and Watkins, 1998; Crammer and Singer, 2000) or 'one vs. one' (Schölkopf et al., 1998; Dietterich and Bakiri, 1995; Allewin et al., 2000; Hsu and Lin, 2002) methods. In its general setting multi-class SVMs generate unnormalized and biased estimates of class conditional probabilities (Platt, 1999a). Calibration and moderation methods have been proposed to arrive at class probability estimates from the trained SVM classifier (Kwok, 1999; Platt, 1999a). For instance Platt (1999a) applied sigmoidal regression to the output of an SVM and showed a performance comparable to regularized maximum likelihood kernel methods (Jaakkola and Haussler, 1999; Zhu and Hastie, 2002). Vapnik has proposed a probability regression technique based on mixture of cosine functions (Vapnik, 1995), where the coefficients of the cosine expansion minimize a regularized function. A drawback of these methods is their difficulty in embedding other inference models like graphical models where re-estimation of SVM parameters can be naturally performed. Kernel logistic regression (KLR) (Jaakkola and Haussler, 1999) provides such a framework to estimate probabilities and can be easily embedded into graphical models with its parameters estimated using an expectation-maximization (EM) like procedure (Jordan and Jacobs, 1994). However, one of the disadvantages of KLR is that the kernel expansion is non-sparse in the data making regression infeasible for large classification problems. A Bayesian learning framework using relevance determination on linear models more general than kernel regression (Tipping, 2001) produces a very sparse expansion but involves significant computation during training that does not scale well to very large data. Recently sparse Gaussian process based methods have been reported (Lawrence et al., 2003), that alleviate scalability problems of relevance determination through use of greedy optimization techniques.

The purpose of this paper is to describe a unifying framework for SVM based classification that directly produces probability scores. Previous work in this area used Shannon entropy in a large margin framework (Jebara, 2001) which led directly to KLR and hence inherited its potential disadvantages of non-sparsity. One of the important contributions of the paper is exploration of links between maximum entropy based learning techniques and large margin classifiers with extensions to quadratic based impurity functions. Within this framework the paper introduces the *Gini* Support Vector Machine (*Gini*SVM) (Chakrabartty and Cauwenberghs, 2002), a large margin classifier based on a quadratic entropy formulation combined with kernel based quadratic distance. At the core of *Gini*SVM is a margin normalization procedure that moderates the output of the classifier. Training *Gini*SVM entails solving a quadratic programming problem analogous to soft-margin SVM. We also present algorithms for training *Gini*SVM classifiers with multiplicative updates, and by growth transformation on polynomial objective functions.

The paper is organized as follows: Section 2 introduces a supervised discriminative framework for obtaining classifiers that produce conditional probability scores. Section 4 introduces *Gini*SVM and derives its normalization properties based on a reverse water-filling algorithm. Section 5 presents algorithms for *Gini*SVM training based on conventional sequential minimum optimization (SMO) and a novel multiplicative update algorithm. Section 6 compares the performance of *Gini*SVM for benchmark UCI databases, a face detection and a text-independent speaker verification task. Section 7 provides concluding remarks with future directions.

2. Generalized Maximum Entropy Based Supervised Learning

In the framework of supervised learning, the learner is provided with a training set of feature vectors $\mathcal{T} \subset \mathcal{X} : \mathcal{T} = \{\mathbf{x}_i\}, i = 1,..,N$ drawn independently from a fixed distribution $P(\mathbf{x}), \mathbf{x} \in \mathcal{X}$. The formulation presented here assumes a countable set $\mathcal{X}$ even though it generalizes to uncountable sets. Also provided to the learner is a set of soft (or possibly hard) labels that represent conditional probability measures $y_{ik} = P(C_k|\mathbf{x}_i)$ defined over a discrete set of classes $C_k, k = 1,..,M$. The labels therefore are normalized and satisfy $\sum_{k=1}^{M} y_{ik} = 1$. The aim of the learner is to choose a finite set of regression functions $P = \{P_k(\mathbf{x})\}, k = 1,..,M$ that accurately predict the true conditional probabilities $P(C_k|\mathbf{x})$. For this purpose the learner uses a distance metric $D_Q : R^M \times R^M \rightarrow R$ that embeds prior knowledge about the topology of the feature space. Since the prior labels y_{ik} are available only for the training set, the learner also defines an agnostic (non-informative) distance metric $D_I : R^M \times R^M \rightarrow R$ which does not assume any knowledge of the training set. The embedded agnostic prior is consistent with maximum entropy principles (Jaynes, 1957; Pietra and Pietra, 1993) and enforces smoothness constraints on the the function $P_k(\mathbf{x})$ by avoiding solutions that over-fit to the training set. Estimating the probability functions $P = \{P_k(\mathbf{x})\}$ entails a training procedure involving minimization of a joint distance metric and is given by

$$\min_P G(P) = \min_P [D_Q(Y,P) + \gamma D_I(P,U)]. \tag{1}$$

Here $Y : R^{|\mathcal{T}|} \times R^M$ is a matrix of prior labels $y_{ik},\ i = 1,..,N,\ k = 1,..,M$, and U denotes a uniform distribution given by $U_k(\mathbf{x}) = 1/M,\ \forall k = 1,..,M$. $\gamma > 0$ is a hyper-parameter that determines a trade-off between the prior and agnostic distance metrics. Minimizing the cost function (1) leads to a solution P that is not only close to a prior distribution with respect to the distance metric $D_Q(.,.)$ but is also close to the non-informative (agnostic) uniform distribution U. In addition, the maximum entropy framework (Pietra and Pietra, 1993) allows to impose linear constraints on the optimization problem (1) based on cumulative statistics defined on the training set. One linear constraint equates the frequency of occurrence of a class $k = 1,..,M$ under the distribution P to an equivalent measure under the prior distribution y_{ik}. This first constraint can be written to express equivalence between average estimated probabilities and empirical frequencies for each class over the training set

$$\sum_{i=1}^{N} P_k(\mathbf{x}_i) = \sum_{i=1}^{N} y_{ik}, \quad k = 1,\ldots M \tag{2}$$

under the assumption that all features $\mathbf{x} \in \mathcal{X}$ are equally likely. A second set of linear constraints expresses boundary and normalization conditions for valid probability distributions

$$P_k(\mathbf{x}) \geq 0, \quad k = 1,\ldots M, \tag{3}$$

$$\sum_{k=1}^{M} P_k(\mathbf{x}_i) = 1 \tag{4}$$

where the additional inequality constraint $P_k(\mathbf{x}) \leq 1,\ k = 1,\ldots M$ is subsumed by the normalizing equality constraint.

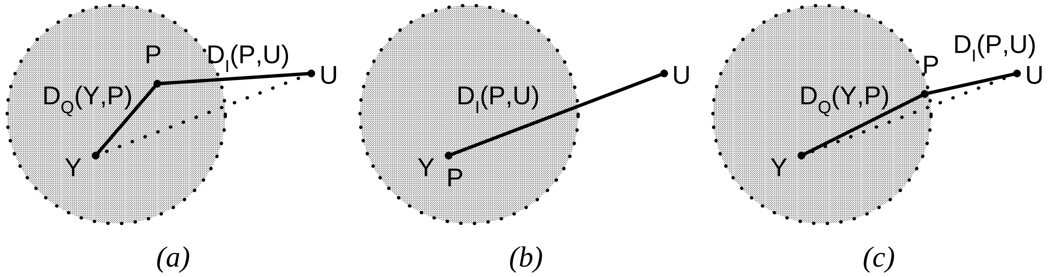


Figure 1: Generalized framework for maximum entropy probability regression. *(a)*: Solution P lies in the constraint space shown as a sphere such that the total distance to the distribution Y and U is minimized. *(b)*: Solution for $\gamma = 0$, where P coincides with Y. *(c)*: Solution for $\gamma \rightarrow \infty$, projecting U onto the constraint space.

Pictorially the solution to the optimization problem (1) is shown in Figure 1. For illustration purposes the linear constraints (2), (3) and (4) are represented by the shaded circle. The distance $D_Q(Y,P)$ determines the proximity of distribution P to a prior empirical distribution Y. $D_I(P,U)$ is a distance that defines an agnostic model when any prior knowledge about prior distribution is absent. This framework is similar to the maximum entropy approach (Jaynes, 1957; Pietra and Pietra, 1993; Jebara, 2001). The possible solutions to minimizing the cost function (1) with constraints (2)-(4) are shown in Figure 1 where the solution P lies within or at the boundary of the constraint space. Note that the constraint space also includes the prior distribution Y. Under non-degenerate conditions the agnostic U distribution will lie outside the constraint space. The value of the hyper-parameter $\gamma > 0$ influences the location of the solution P with respect to the prior Y and agnostic U distributions. As we will see further below, the parameters also determine the sparsity and generalization performance of classifiers defined by parameters P. As shown in Figure 1, for $\gamma = 0$, the solution is the prior distribution Y and thus over-fits to the training set. For the case when $\gamma \rightarrow \infty$, the solution is equivalent to maximum entropy, which is the projection of U on the constraint space.

The solution to the optimization (1) is obtained by first order Karush-Kuhn-Tucker (KKT) conditions (Bertsekas, 1995) with respect to the probability functions $P = \{P_k(\mathbf{x})\}$ and is given by

$$\gamma \frac{\partial D_I(P,U)}{\partial P_k(\mathbf{x})} = -\frac{\partial D_Q(Y,P)}{\partial P_k(\mathbf{x})} + b_k - z(\mathbf{x}) + \beta_k(\mathbf{x}). \tag{5}$$

Here b_k represent Lagrange multipliers corresponding to frequency constraints (2), $\beta_k(\mathbf{x}) \geq 0$ are Lagrange multipliers for the inequality constraints (3), and Lagrange multipliers $z(\mathbf{x})$ correspond to the normalization constraint (4). For the sake of simplicity we will assume a form of $D_I(P,U)$ that can be decomposed into independent, identically distributed (i.i.d.) components as

$$D_I(P,U) = \sum_{k=1}^{M} \sum_{\mathbf{x} \in \mathcal{T}} \Psi(P_k(\mathbf{x}), U_k(\mathbf{x})), \tag{6}$$

where $\Psi : R \times R \rightarrow R$ is a concave function. The first order condition (5) can be written as a Legendre transform (Rockefeller, 1970) with respect to $\Psi(.)$ as

$$P_k(\mathbf{x}) = \nabla\Psi^{-1}\left(\frac{1}{\gamma}\left[-\frac{\partial D_Q(Y,P)}{\partial P_k(\mathbf{x})} + b_k - z(\mathbf{x}) + \beta_k(\mathbf{x})\right]\right). \tag{7}$$

where $\nabla\Psi^{-1}(.)$ denotes the Legendre transform with respect to $P_k(\mathbf{x})$ for the bivariate function $\Psi(.)$. The Legendre transformation is commonly used in the dual formulation of support vector machines and other kernel machines (Vapnik, 1995; Schölkopf and Smola, 2001), and we refer to $\nabla\Psi^{-1}(.)$ as the dual potential function. Note that $\nabla\Psi^{-1}(.)$ is monotonic due to the concavity of $\Psi(.)$.

Several choices exist for the prior distance metric $D_Q(.,.)$. A popular metric is a quadratic distance extensively used in kernel methods (Schölkopf et al., 1998) and as covariance functions in Bayesian methods (Jordan and Jacobs, 1994). In its general form the quadratic distance between two conditional distributions $\hat{P} = \{\hat{P}_k(\mathbf{x})\}$ and $P = \{P_k(\mathbf{x})\}$ is given by

$$D_Q(\hat{P},P) = \frac{C}{2}\sum_{k=1}^{M}\sum_{\mathbf{x},\mathbf{v}\in\mathcal{T}} K(\mathbf{x},\mathbf{v})\left[\hat{P}_k(\mathbf{x}) - P_k(\mathbf{x})\right]\left[\hat{P}_k(\mathbf{v}) - P_k(\mathbf{v})\right]. \tag{8}$$

Here $K : R^M \times R^M \to R$ represents a symmetric, positive definite kernel satisfying the *Mercer's criterion*,[1] such as a Gaussian radial basis function or a polynomial spline (Schölkopf et al., 1998; Wahba, 1998). The distance $D_Q(.,.)$ embeds prior knowledge induced by the kernel $K(\mathbf{x},\mathbf{v})$ and therefore quantifies a topology of a metric space for points $\mathbf{x},\mathbf{v} \in \mathcal{X}$.

For the quadratic form $D_Q(.,.)$ given by Equation (8) the first order conditions (7) can be written as

$$P_k(\mathbf{x}) = \nabla\Psi^{-1}\left(\frac{1}{\gamma}[f_k(\mathbf{x}) - z(\mathbf{x}) + \beta_k(\mathbf{x})]\right) \tag{9}$$

where

$$f_k(\mathbf{x}) = \sum_{i=1}^{N}\lambda_k^i K(\mathbf{x}_i,\mathbf{x}) + b_k$$

with inference parameters

$$\lambda_k^i = C[y_{ik} - P_k(\mathbf{x}_i)].$$

The Lagrange parameter function $\beta_k(\mathbf{x})$ in Equation (9) needs to ensure that the probability scores $P_k(\mathbf{x}) \geq 0 \quad \forall\mathbf{x} \in \mathcal{X}$ according to (3), and the Lagrange parameter function $z(\mathbf{x})$ needs to ensure normalized probabilities $\sum_{k=1}^{M} P_k(\mathbf{x}) = 1$ according to (4).

The set of inference parameters $\Lambda = \{\lambda_k^i\}, i = 1,..,N, \quad k = 1,..,M$ is obtained by solving (1) over the training set $\mathcal{T}$. Expressing the general form (6) for the agnostic distance $D_I(P,U)$ and the quadratic distance (8) for the prior distance $D_Q(Y,P)$ in terms of the inference parameters λ_k^i in the cost function (1) leads to a dual formulation H_d

$$H_d = \sum_{k=1}^{M}\left[\frac{1}{2C}\sum_{i=1}^{N}\sum_{j=1}^{N}\lambda_k^i Q_{ij}\lambda_k^j + \gamma\sum_{i=1}^{N}\Psi(y_{ik} - \lambda_k^i/C)\right] \tag{10}$$

where $Q_{ij} = K(\mathbf{x}_i,\mathbf{x}_j)$ denote elements of the *kernel matrix* $\mathbf{Q}$. Like the primal (1), minimization of the dual H_d is subject to linear constraints (2)-(4) rewritten in terms of the inference parameters as

$$\begin{aligned} \sum_{k=1}^{M}\lambda_k^i &= 0, \quad i = 1,\ldots N, \\ \sum_{i=1}^{N}\lambda_k^i &= 0, \quad k = 1,\ldots M, \\ \lambda_k^i &\leq Cy_{ik}. \end{aligned} \tag{11}$$

[1] $K(\mathbf{x},\mathbf{v}) = \Phi(\mathbf{x})\cdot\Phi(\mathbf{v})$. The map $\Phi(\cdot)$ need not be computed explicitly, as it only appears in inner-product form.

3. Kernel Logistic Regression

For a frame of reference in the comparison between different formulations of cost functions, the optimization framework given by the dual (10) subject to constraints (11) is first applied to kernel logistic regression (KLR) (Jaakkola and Haussler, 1999), with agnostic distance

$$\begin{aligned} D_I(P,U) &= \sum_{k=1}^{M}\sum_{\mathbf{x}\in\mathcal{T}} P_k(\mathbf{x})\log\frac{P_k(\mathbf{x})}{U_{ik}} \\ &= \sum_{k=1}^{M}\sum_{\mathbf{x}\in\mathcal{T}} P_k(\mathbf{x})\log P_k(\mathbf{x}) + \mathrm{cst} \end{aligned} \tag{12}$$

derived from the Kullback-Leibler (KL) divergence $\Psi_{KL}(P,U) = P\log(P/U)$ for a uniform agnostic distribution $U_{ik} \equiv 1/M$. The constant term $\mathrm{cst} = N\log M$ in (12) drops in the minimization and is subsequently ignored. The probability function according to (9) is then given by

$$P_k(\mathbf{x}) = \exp\left(\frac{1}{\gamma}[f_k(\mathbf{x}) - z(\mathbf{x}) + \beta_k(\mathbf{x})]\right). \tag{13}$$

By property of $\exp(.)$, $P_k(\mathbf{x}) \geq 0 \quad \forall\mathbf{x}\in\mathcal{X}$, and so the Lagrange multiplier $\beta_k(\mathbf{x})$ in (13) is arbitrary and can be eliminated, $\beta_k(\mathbf{x}) \equiv 0$. The other Lagrange multiplier $z(\mathbf{x})$ in (13) is determined by expressing the normalization condition $\sum_{k=1}^{M} P_k(\mathbf{x}) = 1$ which leads to a logistic model

$$P_k(\mathbf{x}) = \exp\left(\frac{1}{\gamma}f_k(\mathbf{x}))/\sum_{p=1}^{M}\exp(\frac{1}{\gamma}f_p(\mathbf{x})\right). \tag{14}$$

Substituting the KL distance $\Psi_{KL}(.,.)$ for $\Psi(.,.)$ in the general form (10) directly leads to the dual cost function

$$H_e = \sum_{k=1}^{M}\left[\frac{1}{2C}\sum_{i=1}^{N}\sum_{j=1}^{N}\lambda_k^i Q_{ij}\lambda_k^j + \gamma\sum_{i=1}^{N}(y_{ik}-\lambda_k^i/C)\log(y_{ik}-\lambda_k^i/C)\right] \tag{15}$$

subject to the dual constraints (11).

3.1 KLR Primal Reformulation

The dual (15) derived from the general maximum entropy form (1) is identical to the dual formulation of another, closely related primal cost function for kernel logistic regression as formulated in Jaakkola and Haussler (1999). The purpose of this section is to establish the equivalence with a connection to large margin kernel machines and their interpretation in feature space (Schölkopf and Smola, 2001).

Expressing the kernel function $K(\mathbf{x},\mathbf{v}) = \Phi(\mathbf{x})\cdot\Phi(\mathbf{v})$ as an inner-product in a transformed feature space $\Phi(.)$, the decision functions $f_k(\mathbf{x})$ are linked to a set of M hyperplanes

$$\begin{aligned} f_k(\mathbf{x}) &= \sum_{i=1}^{N}\lambda_k^i\, K(\mathbf{x}_i,\mathbf{x}) + b_k \quad (16) \\ &= \sum_{i=1}^{N}\lambda_k^i\, \Phi(\mathbf{x}_i)\cdot\Phi(\mathbf{x}) + b_k \quad (17) \\ &= \mathbf{w}_k\cdot\Phi(\mathbf{x}) + b_k \end{aligned}$$

where $\mathbf{w}_k = \sum_{i=1}^{N} \lambda_k^i \Phi(\mathbf{x}_i)$ represent the parameters of the hyperplanes. The following proposition links the kernel logistic regression dual (15) with its equivalent primal formulation (Jaakkola and Haussler, 1999).

Proposition I: The kernel logistic regression objective function (15) is the dual derived from a primal objective function with regularized loss function

$$\begin{aligned} L_e &= \sum_{k=1}^{M} \frac{1}{2}|\mathbf{w}_k|^2 + C\sum_{i=1}^{N}\sum_{k=1}^{M} y_{ik}\log\frac{y_{ik}}{P_k(\mathbf{x}_i)} \\ &= \sum_{k=1} \frac{1}{2}|\mathbf{w}_k|^2 - C\sum_{i=1}^{N}[\sum_{k=1}^{M} y_{ik} f_k(\mathbf{x}_i) - \log(e^{f_1(\mathbf{x}_i)} + \ldots + e^{f_M(\mathbf{x}_i)})]\ . \end{aligned} \tag{18}$$

where additional constant terms in (18) have been ignored and a unity value for γ has been assumed in the probability model (14). The proof of the proposition is provided in Appendix A.

The primal uses the Kullback-Leibler (KL) divergence $\Psi_{KL}(P,U) = P\log(P/U)$ between distributions y_{ik} and $P_k(\mathbf{x}_i)$ as loss function in the regularized form (18) (Wahba, 1998; Zhu and Hastie, 2002). One of the disadvantages of the kernel logistic dual is that the KL divergence distance metric strongly penalizes solutions far away from the agnostic distribution U, leading to a non-sparse kernel expansion. A sparser kernel expansion is obtained in soft-margin support vector machines for classification. A *Gini* form of entropy as agnostic distance metric provides the connection between support vector machines and probability regression, studied next.

4. *Gini*SVM and Margin Normalization

Instead of KL divergence, a natural choice for an agnostic distance metric D_I is a quadratic form of entropy similar to the quadratic form of the prior distance metric D_Q. A *Gini* quadratic form of entropy, or impurity function, has been used extensively in natural language processing for growing decision trees (Breiman et al., 1984). The *Gini* quadratic entropy forms the basis of the *Gini*-support vector machine (*Gini*SVM) for probability regression (Chakrabartty and Cauwenberghs, 2002).

The *Gini* quadratic form of entropy $\Psi_{Gini}(P,U) = \frac{1}{2}(P-U)^2$ with uniform agnostic distribution $U_{ik} \equiv 1/M$ leads to an agnostic distance metric

$$\begin{aligned} D_I(P,U) &= \frac{1}{2}\sum_{k=1}^{M}\sum_{\mathbf{x}\in\mathcal{T}}(P_k(\mathbf{x}) - U_{ik})^2 \\ &= \frac{1}{2}\sum_{k=1}^{M}\sum_{\mathbf{x}\in\mathcal{T}} P_k(\mathbf{x})^2 + \mathrm{cst} \end{aligned}$$

where the constant term $\mathrm{cst} = -N/2M$ drops in the minimization. Substituting the Gini distance $\Psi_{Gini}(.,.)$ for $\Psi(.,.)$ in the general form (10) leads to the dual *Gini*SVM cost function

$$H_g = \sum_{k=1}^{M}\left[\frac{1}{2C}\sum_{i=1}^{N}\sum_{j=1}^{N}\lambda_k^i Q_{ij}\lambda_k^j + \frac{\gamma}{2}\sum_{i=1}^{N}(y_{ik} - \lambda_k^i/C)^2\right] \tag{19}$$

under constraints (11).

In contrast to the KL distance in the KLR dual (15), the quadratic distance in the *Gini*SVM dual (19) allows sparse kernel expansions, where several of the inference parameters λ_k^i are driven

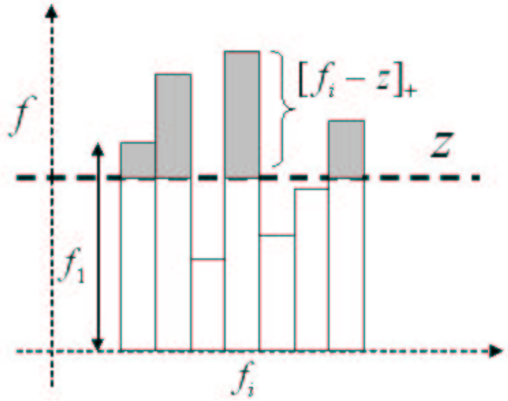


Figure 2: Illustration of reverse water-filling procedure. The level z is adjusted as to maintain the net f_i in excess of z (shaded area) at γ.

to zero by the inequality constraints in (11) corresponding to a majority of labels for which $y_{ik} = 0$. Even sparser kernel expansions could be obtained with soft-margin support vector machines for classification, owing to its slightly different quadratic cost function under linear constraints which further favors sparsity. More significantly, *Gini*SVM produces conditional probability estimates that are based on maximum entropy (1). The probability estimates themselves could be sparse, with $P_k(\mathbf{x}) = 0$ for a number of classes k depending on γ and $\mathbf{x}$, as we show next.

4.1 Margin Normalization and Reverse Waterfilling

The quadratic entropy form of $\Psi_{Gini}(.,.)$ leads to a linear Legendre transform $\nabla\Psi^{-1}(.)$ and thus a linear, rather than exponential, form of the conditional probability estimates $P_k(\mathbf{x}) = 1/\gamma/[f_k(\mathbf{x}) - z(\mathbf{x}) + \beta_k(\mathbf{x})], k = 1,..,M$. To ensure positive probabilities according to constraints (3), the Lagrange parameters $\beta_k(\mathbf{x})$ produce rectified linear probability estimates

$$P_k(\mathbf{x}) = \frac{1}{\gamma}[f_k(\mathbf{x}) - z(\mathbf{x})]_+ \tag{20}$$

where $[x]_+ = \max(x, 0)$ denotes a hinge function. The remaining Lagrange parameter $z(\mathbf{x})$ is determined through a subtractive normalization procedure which solves for the normalization constraint (4)

$$\sum_{k=1}^{M} [f_k(\mathbf{x}) - z(\mathbf{x})]_+ = \gamma. \tag{21}$$

The conditions (20) and (21) are jointly satisfied by applying a reverse water-filling algorithm commonly found in communication systems (Cover and Thomas, 1991), listed in Algorithm 1 and illustrated in Figure 2. The algorithm recursively computes the normalization factor $z(\mathbf{x})$ such that the net balance of class confidence levels $f_k(\mathbf{x})$ in excess of $z(\mathbf{x})$ equals γ.

We refer to the procedure solving for $z(\mathbf{x})$ given confidence scores $f_k(\mathbf{x})$ in (21) as *margin normalization*, because of similarities between the normalization parameter γ and the margin of multi-class soft-margin support vector machines (Weston and Watkins, 1998). Unlike the *divisive*

Algorithm 1 Reverse water-filling procedure to compute normalization parameter z

```
Require: Set of confidence values $\{f_k(x)\}, k = 1,..,M$.
Ensure: $z = 0, N = 1, T = 0$
  $a = max\{f_k(x)\}$
  $\{s\} \leftarrow \{f_k(x)\} - \{a\}$
  while $T < \gamma$  &  $N < M$ do
    $b = max\{s\}$
    $T \leftarrow T + N(a - b)$
    $a \leftarrow b$
    $\{s\} \leftarrow \{s\} - \{b\}$
    $N \leftarrow N + 1$
  end while
  $z \leftarrow b + N(\gamma - T)$
```

normalization (14) of probabilities in kernel logistic regression, the *subtractive* margin normalization (20)-(21) in *Gini*SVM offers several distinct properties in connection with margin based classifiers:

Monotonicity: *Let $f_k(\mathbf{x}), k = 1,..,M$ a set of GiniSVM decision functions satisfying the reverse water-filling conditions $\sum_{k=1}^{M}[f_k(\mathbf{x}) - z_1(\mathbf{x})]_+ = \gamma_1$ and $\sum_{k=1}^{M}[f_k(\mathbf{x}) - z_2(\mathbf{x})]_+ = \gamma_2$. If $\gamma_1 \geq \gamma_2 > 0$, then $z_1(\mathbf{x}) \leq z_2(\mathbf{x})$.*

Proof: The two reverse water-filling conditions lead to

$$\sum_{k=1}^{M}([f_k(\mathbf{x}) - z_1(\mathbf{x})]_+ - [f_k(\mathbf{x}) - z_2(\mathbf{x})]_+) = \gamma_1 - \gamma_2 > 0.$$

The convexity of the hinge function $[a-b]_+ \geq [a]_+ - [b]_+; a, b \in \mathcal{R}$ leads to

$$[z_2(\mathbf{x}) - z_1(\mathbf{x})]_+ \geq (\gamma_1 - \gamma_2)/M > 0$$

which is equivalent to $z_2(\mathbf{x}) > z_1(\mathbf{x})$.

Sparsity: The effect of rectification (20) in the subtractive normalization (21) is to produce a number $0 \leq m < M$ of classes $k_1, \ldots k_m \in \{1, \ldots M\}$ with zero probabilities $P_{k_j}(\mathbf{z}) = 0$, for which the normalization level $z(\mathbf{x})$ exceeds the confidence level $f_{k_j}(\mathbf{x})$. As a direct consequence of the monotonicity property, decreasing the margin parameter γ leads to a larger number m of classes with zero probabilities. Therefore the margin parameter γ directly controls the sparsity m of the probability estimates, assigning the probability mass to a smaller fraction of more confident classes with larger $f_k(\mathbf{x})$ as γ is decreased. Besides the dependence on γ, the number of zero probability classes m depends on the actual values of $f_k(\mathbf{x})$, and hence on the inputs $\mathbf{x}$.

Margin: In the limit $\gamma \rightarrow 0$, the normalization factor $z(\mathbf{x}) \rightarrow max_k f_k(\mathbf{x})$. Thus as $\gamma \rightarrow 0$, margin normalization acts as 'winner-take-all', $m \rightarrow M-1$, and strongly favors the highest class score. Based on this principle a multi-class probability margin can be defined based on the multi-class decision functions $f_k(x)$ as $f_k(\mathbf{x}) = z(\mathbf{x}) + \gamma$. The effect of the hyper-parameter γ can be seen on a synthetic three-class classification problem and is shown in Figure 3. The hyper-parameter γ determines the smoothness of the decision boundary and controls the location of the margin (shown

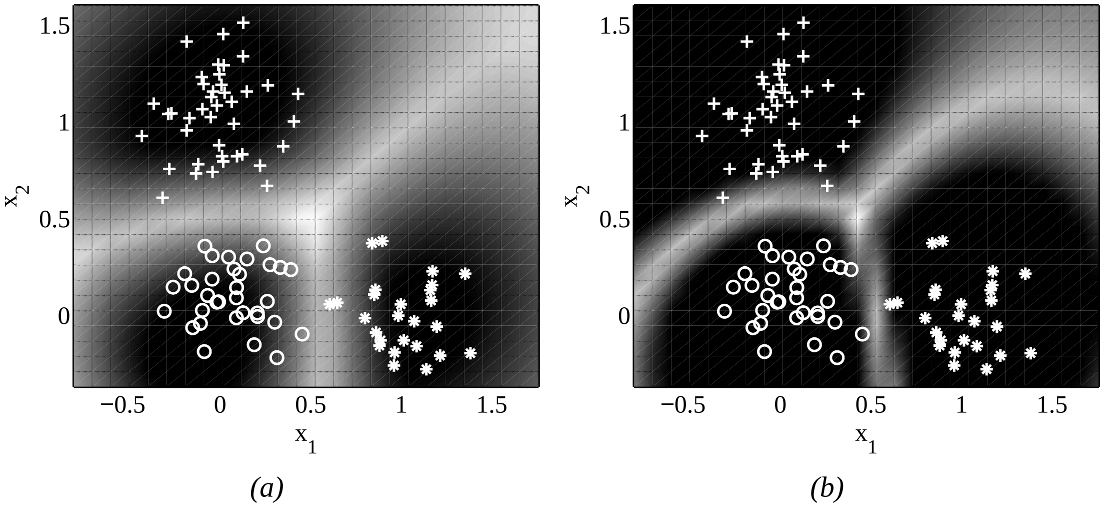


Figure 3: Equal probability contour plots for a three-class problem with the *Gini*SVM solution obtained for (a) $\gamma = 8$ and (b) $\gamma = 0.08$.

by 'white' region). Similar to soft-margin SVM the location of the margin determines the sparsity of the *Gini*SVM solution. This is illustrated in Figure 3(a)-(b). Shades in the figure represent equal probability contours, and the extent of 'white' regions around the decision boundaries illustrates the margin of separation. It can be seen from Figure 3(a)-(b) that reduction in γ has the effect of increasing the size of the margin, and thereby controls the sparsity of the *Gini*SVM solution.

Robustness: The subtractive margin normalization (20) and (21) is inherently robust to impulsive noise, since components $f_k(\mathbf{z})$ in the kernel expansion smaller than a threshold $z(\mathbf{x})$ (at most a margin γ below the largest value) do not contribute to the output.[2] In a physical implementation of margin decoding, adjusting the level γ according to the noise floor leads to significant improvements in decoding performance (Chakrabartty and Cauwenberghs, 2004). The robustness properties of *Gini*SVM in relation to the threshold γ are further analyzed in Section 4.4.

4.2 *Gini*SVM Primal Reformulation

In this section we derive an equivalent primal reformulation of the *Gini*SVM dual (19), analogous to the derivation in Section 3.1. As for the multi-class logistic primal (18), decision functions for classes $k = 1,..,M$ are expressed in terms of a set of M hyperplanes $f_k(\mathbf{x}) = \mathbf{w}_k \dot{} \Phi(x) + b_k$. Given a set of training vectors $\mathbf{x}_i \in \mathcal{R}^D, i = 1,..,N$ and its corresponding prior probability distributions $y_{ik} \in \mathcal{R} : y_{ik} \geq 0; \sum_{k=1}^{M} y_{ik} = 1$, *Gini*SVM in its primal reformulation of minimizes a regularization factor proportional to the $L2$ norm of the weight vectors $\mathbf{w}_k, k = 1,..,M$ and a quadratic loss function

[2] The subtractive normalization is insensitive only to *negative* impulsive noise in $f_k(\mathbf{z})$. Typically, a choice of kernel indicating a match rather than a mismatch in feature space will avoid positive impulsive noise, since random error is much more likely to activate further mismatch rather than an accidental match.

l_g according to

$$L_g = \frac{1}{2}\sum_{k=1}^{M}|\mathbf{w}_k|^2 + \sum_{i=1}^{N}\sum_{k=1}^{M} l_g(\mathbf{w}_k, b_k, z_i) \tag{22}$$

with loss function $l_g(\mathbf{x}_i)$ for each training vector $\mathbf{x}_i$ given by

$$l_g(\mathbf{x}_i) = \frac{\gamma C}{2}\left(y_{ik} - \frac{1}{\gamma}[f_k(\mathbf{x}_i) - z_i]_+\right)^2,$$

and where z_i, $i = 1, \ldots N$ are free parameters entering the minimization of L_g, along with the hyperplane parameters $\mathbf{w}_k$ and b_k, $k = 1, \ldots M$.

Proposition II: Denote the solution to the minimization of L_g as

$$(\mathbf{w}_k^*, b_k^*, z_i^*) = \text{argmin}_{\mathbf{w}_k, b_k, z_i} L_g,$$

then

1. $P_k(\mathbf{x}_i) = \frac{1}{\gamma}[f_k^*(\mathbf{x}_i) - z_i^*]_+$ with $f_k^*(\mathbf{x}_i) = \mathbf{w}_k^* \cdot \mathbf{x}_i + b_k^*$ for a given data $\mathbf{x}_i$ is a valid conditional probability measure over classes $k \in 1, .., M$, where z_i performs the normalization $\sum_{k=1}^{M} P_k(\mathbf{x}_i) = 1$.
2. The dual cost function corresponding to the primal cost function (22) is the *Gini*SVM dual (19).

The proof of the proposition is given in Appendix B.

4.3 Binary *Gini*SVM and Quadratic SVM

In one case of interest the multi-class *Gini*SVM solution simplifies to a binary class problem, where the learner is provided with binary labels $y_i \in \{-1, +1\}$ representing class membership of a feature vector $\mathbf{x}_i \in \mathcal{T}$. Binary *Gini*SVM entails regression of a single probability $P_{+1}(\mathbf{x}) = 1 - P_{-1}(\mathbf{x})$ as a function of a single margin variable $f(\mathbf{x}) = \frac{1}{2}(f_{+1}(\mathbf{x}) - f_{-1}(\mathbf{x}))$. Elimination of the normalization parameter z from the binary version of (20) constrained by (21) yields

$$P_{+1}(\mathbf{x}) = \left[\frac{f(\mathbf{x})}{\gamma} + \frac{1}{2}\right]_0^1 \tag{23}$$

where $[.]_0^1$ denotes a limiter function confining the probability to the [0,1] interval, $[a]_0^1 = [a]^+ - [a - 1]^+$. With the kernel expansion of $f(\mathbf{x})$ expressed in reduced form[3]

$$f(\mathbf{x}) = \sum_{i=1}^{N} \lambda^i y_i K(\mathbf{x}_i, \mathbf{x}) + b, \tag{24}$$

the *Gini*SVM dual objective function (19) and linear constraints (11) reduce to

$$H_b = \min_{\lambda^i} \frac{1}{C}\left[\sum_{i=1}^{N}\sum_{j=1}^{N} \lambda^i \lambda^j y_i y_j K(\mathbf{x}_i, \mathbf{x}_j) - \sum_{i=1}^{N} G(\lambda^i)\right] \tag{25}$$

[3] This choice of kernel expansion is consistent with binary soft-margin SVM and binary KLR, with identical dual formulation under constraints, and with the only difference in the form of the dual potential function $G(\lambda)$ (26).

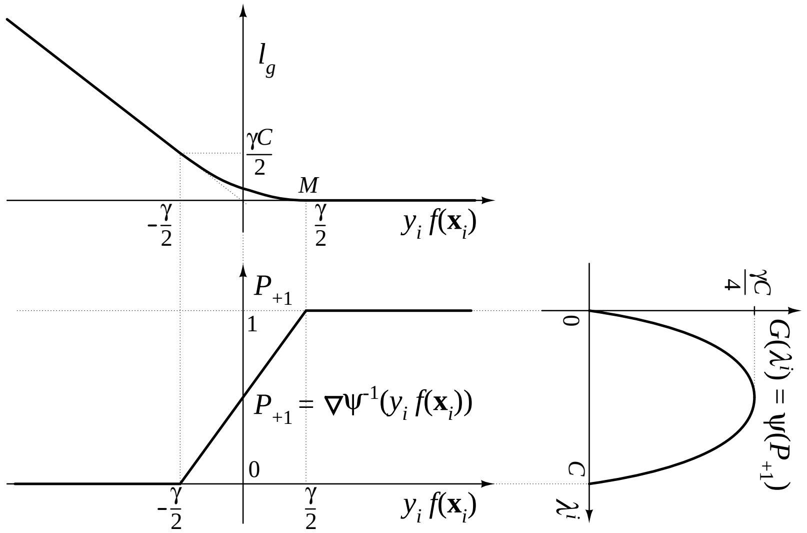


Figure 4: Graphical illustration of the relation between binary *Gini*SVM dual potential function G, inference parameters λ^i, probability estimates P_i, and primal loss function l_g.

under constraints

$$\sum_{i=1}^{M} \lambda^i y_i = 0,$$
$$0 \le \lambda^i \le C$$

with dual potential function

$$G(\lambda) = \gamma C \frac{\lambda}{C}(1 - \frac{\lambda}{C}). \tag{26}$$

The derivation is given in Appendix C.

Figure 4 graphically illustrates the relationship between the binary *Gini*SVM dual potential function G, inference parameters λ^i, probability estimates P_i, and primal reformulation loss function l_g. The dual potential is linked to the probability estimate through the Legendre transform $\nabla\Psi^{-1}(.)$ as described by Equations (5)-(7). Since the dual potential function is symmetric $\Psi(P_{+1}) = \Psi(1 - P_{+1}) = \Psi(P_{-1})$, it follows that the Legendre transform $\nabla\Psi^{-1}(.)$ is antisymmetric and hence the probability estimates are centered around the discrimination boundary, $P_{+1}(f(\mathbf{x})) = 1 - P_{+1}(-f(\mathbf{x}))$ consistent with the functional form (23). Symmetry in the dual potential function thus leads to unbiased probability estimates that are centered around the discrimination boundary, $P_{+1} = P_{-1} = 1/2$ for $f(\mathbf{x}) = 0$.

The Legendre transform also links the dual potential function to the primal reformulation cost function. Note the relationship between the parameter γ scaling the dual potential function, and the location of the margin in the loss function and probability estimate indicated by M in Figure 4. Hence the parameter γ can be seen both to control the strength of the agnostic metric D_I, and to control a measure of margin in the probability regression. The regularization parameter C also scales the dual potential function, but controls regularization by scaling the primal loss function by the same factor C without affecting margin, as in soft-margin SVM classification.

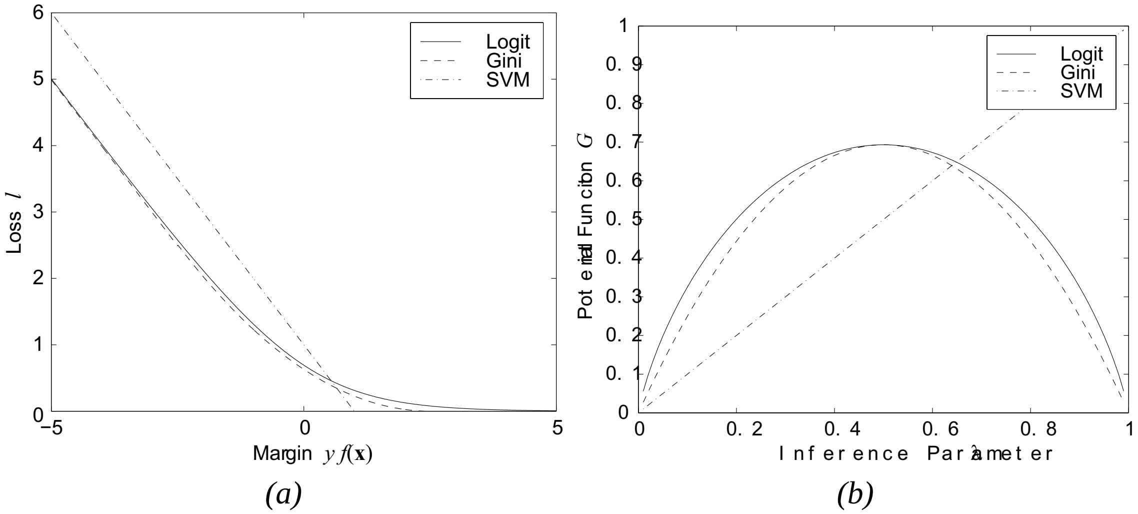


Figure 5: Primal and dual formulation of logistic regression, *Gini*SVM regression and soft-margin SVM classification. (a): Loss function in the primal formulation. (b): Potential function in the dual formulation. The *Gini*SVM loss function and potential function closely approximate those for logistic regression, while offering the sparseness of soft-margin SVM classification. $C = 1$, and $\gamma = 8\log 2$ for *Gini*SVM and $\gamma = 1$ for logistic regression.

A direct comparison can be made between the binary *Gini*SVM formulation and other binary classifiers by inspecting differences in their potential functions $G(u)$, shown in Figure 5(b). The *Gini*SVM potential function is symmetric around the center of the agnostic, uniform distribution $U = 1/2$ where it reaches its maximum. The center corresponds to the origin of the margin variable $y_i f(\mathbf{x}_i)$ in Figure 5(a) which represents the separating hyperplane. Figure 5(b) also shows the binary KLR dual potential function given by Shannon's binary entropy (Jaakkola and Haussler, 1999)

$$G(\lambda) = \gamma C \left(\frac{\lambda}{C} \log(\frac{\lambda}{C}) + (1 - \frac{\lambda}{C}) \log(1 - \frac{\lambda}{C}) \right).$$

Like *Gini*SVM, the binary KLR dual potential function is symmetric with respect to the separating hyperplane in Figure 5(a), and hence also produces unbiased estimates of conditional probabilities. In contrast, the soft-margin SVM potential function $G(\lambda) = \lambda$ is asymmetric with respect to the separating hyperplane and produces biased or skewed conditional probability estimates. The binary *Gini*SVM dual bears similarity to the quadratic SVM dual (Schölkopf et al., 1998), but the quadratic SVM lacks the symmetry of the potential function around the separating hyperplane.

4.4 Relation to Robust Estimation and Logistic Regression

The *Gini*SVM dual (19) relates to the kernel logistic regression dual (15) through a lower-bound on Shannon entropy. Using the inequality $\log x \leq x - 1, x \geq 0$, the Shannon entropy term $G_e(P) = -\sum_{k=1}^{M} P_k \log P_k$ is everywhere larger than the *Gini* entropy $G_e(P) \geq 1 - \sum_{ik} P_{ik}^2$. This is illustrated in Figure 5(b) which compares the potential functions for KLR and *Gini*SVM in the binary case. Both expressions of entropy reach their maximum for a uniform distribution, $P = \frac{1}{2}$. It can be shown that the solution obtained by minimizing the *Gini*SVM dual H_g in Equation (19) is an over-estimate of the solution obtained by minimizing kernel logistic dual H_e given by Equation (15). In fact we found that initial iterations decreasing the cost H_g also resulted in a decrease of H_e with deviations evident only near convergence. Thus the *Gini*SVM dual H_g can also be used for approximately solving the kernel logistic dual H_e.

The loss function corresponding to binary *Gini*SVM can be visualized through Figure 5(a) and compared with other primals. For binary-class *Gini*SVM, 'margin' is visualized as the extent over which data points are asymptotically normally distributed. Using the notation for asymptotic normality in Huber (1964), the distribution of distance z from one side of the margin for data of one class[4] is modeled by

$$F(z) = (1-\varepsilon)\mathcal{N}(z,\sigma) + \varepsilon\mathcal{H}(z)$$

where $\mathcal{N}(.,\sigma)$ represents a normal distribution with zero mean and standard deviation σ, $\mathcal{H}(.)$ is the unknown symmetrical contaminating distribution, and $0 \leq \varepsilon \leq 1$. $\mathcal{H}(.)$ could, for instance, represent impulsive noise contributing outliers to the distribution.

Huber (1964) showed that for this general form of $F(z)$ the most robust estimator achieving minimum asymptotic variance minimizes the following loss function:

$$g(z) = \begin{cases} \frac{1}{2}\frac{z^2}{\sigma^2} & ; \quad |z| \leq k\sigma \\ k\frac{|z|}{\sigma} - \frac{1}{2}\sigma k^2 & ; \quad |z| > k\sigma \end{cases} \tag{27}$$

where in general the parameter k depends on ε. For *Gini*SVM, the distribution $F(z)$ for each class is assumed one-sided ($z \leq 0$). In particular, the Huber loss function $g(z)$ in (27) reduces to the binary *Gini*SVM loss function $l_g(yf(\mathbf{x}))$, shown in Figure 4, for $z \leq 0$ with $z = yf(\mathbf{x}) - \gamma/2$, $k\sigma = \gamma$, and $k/\sigma = C$. Therefore the parameter γ in *Gini*SVM can be interpreted as a noise margin in the Huber formulation, consistent with its interpretation as noise threshold in the reverse water-filling procedure for margin normalization. As with soft-margin SVM, points that lie beyond the margin ($z > 0$) are assumed correctly classified, and do not enter the loss function ($g(z) \equiv 0$). The binary *Gini*SVM loss function is a special case of Huber loss used in quadratic SVMs (Schölkopf et al., 1998) which directly generates normalized scores from the margin variable $f(\mathbf{x})$.

5. *Gini*SVM Training Algorithm

*Gini*SVM training entails solving a quadratic optimization problem for which several standard packages and algorithms are available (Platt, 1999b; Cauwenberghs and Poggio, 2001; Osuna et al., 1997). Most of these methods exploit the underlying structure in the classification problem to incorporate heuristics that considerably speed up the convergence of the training algorithm. In this section we describe two algorithms for optimizing *Gini*SVM dual function (19). The first algorithm uses a decomposition algorithm called sequential minimal optimization. The second algorithm uses the polynomial nature of the dual resulting into a novel multiplicative update algorithm based on growth transformation on probabilities.

[4] The distributions for the two classes are assumed symmetrical, with margin on opposite sides, and distance z in opposite directions.

5.1 Sequential Minimal Optimization

Sequential minimal optimization (Platt, 1999b) is an extreme case of a decomposition based quadratic program solver, where a smallest set of inference parameters is chosen each iteration, and optimized subject to the linear constraints. The advantage of SMO is that it can be efficiently implemented without resorting to QP packages, and it scales to very large data sets. In the case of the *Gini*SVM dual function (19), at least four inference parameters need to be chosen to satisfy two sets of equality constraints (11). A randomized version of SMO algorithm is described in Algorithm 2.

Algorithm 2 Randomized SMO algorithm

```
Require: Training data $\mathbf{x}_i, i = 1,..,N$ and labels $y_{ik}, i = 1,..,N, k = 1,..,M$
Ensure: Let $\lambda_k^i = 0$ for $i = 1,..,N, k = 1,..,M$.
  repeat
    • Randomly choose a set of four inference parameters $\lambda_1^1, \lambda_2^1, \lambda_1^2$ and $\lambda_2^2$.
    • Update $\lambda_1^1, \lambda_2^1, \lambda_1^2$ and $\lambda_2^2$ such that the dual (19) is minimized subject to constraints (11).
  until convergence
```

The derivation of the SMO update rule based on the choice of inference parameters is given in Appendix D. Instead of random selection of working sets of inference parameters, heuristics based on the structure of the classification problem can be used to speed up convergence (Platt, 1999b; Keerthi et al., 2001). Standard QP algorithmic methods for SVM training such as caching and shrinking besides chunking (Joachims, 1998) can be applied to further speed up convergence of SMO training.

5.2 Growth Transformation on Generalized Polynomial Dual

In lieu of the inference parameters defined as $\lambda_k^i = C[y_{ik} - P_k(\mathbf{x}_i)]$, the *Gini*SVM dual in (19) can be expressed in terms of probabilities $P_{ik} = P_k(\mathbf{x}_i)$ as

$$H = \frac{C}{2}\sum_{k=1}^{M}\left[\sum_{i=1}^{N}\sum_{j=1}^{N} Q_{ij}\left[y_{ik} - P_{ik}\right]\left[y_{jk} - P_{jk}\right] + \frac{\gamma}{2}\sum_{i=1}^{N} P_{ik}^2\right] \tag{28}$$

with linearity constraints (3) and (4) to ensure valid probabilities, $P_{ik} \geq 0, \forall i, k$ and $\sum_{k=1}^{M} P_{ik} = 1, \forall i$. For the remainder of the derivation the additional equality constraint (2) corresponding to the bias term b will be relaxed. Artifacts due to absence of the bias b can reduced by properly pre-processing and centering the training data or by incorporating an additional input dimension in the kernel function. The optimization function (28) is a non-homogeneous polynomial with normalized probability variables $P_{ik}, \forall i$ and with possibly negative coefficients. We can directly apply results from Baum and Sell (1968) and Gopalakrishnan et al. (1991) to optimize the dual (28).

Theorem 2 (Gopalakrishnan et al.) Let $H(\{P_{ik}\})$ a polynomial of degree d in variables P_{ik} in the domain $D : P_{ik} \geq 0, \sum_{k=1}^{q_i} P_{ik} = 1, i = 1,..,N, k = 1,..,q_i$ such that $\sum_{k=1}^{q_i} P_{ik} \frac{\partial H}{\partial P_{ik}}(P_{ik}) / \neq 0 \quad \forall i$. Define an iterative map according to the following recursion

$$\widehat{P}_{ik} \leftarrow \frac{P_{ik}\left(\frac{\partial H}{\partial P_{ik}}(P_{ik}) + \Gamma\right)}{\sum_{k=1}^{q_i} P_{ik}\left(\frac{\partial H}{\partial P_{ik}}(P_{ik}) + \Gamma\right)} \tag{29}$$

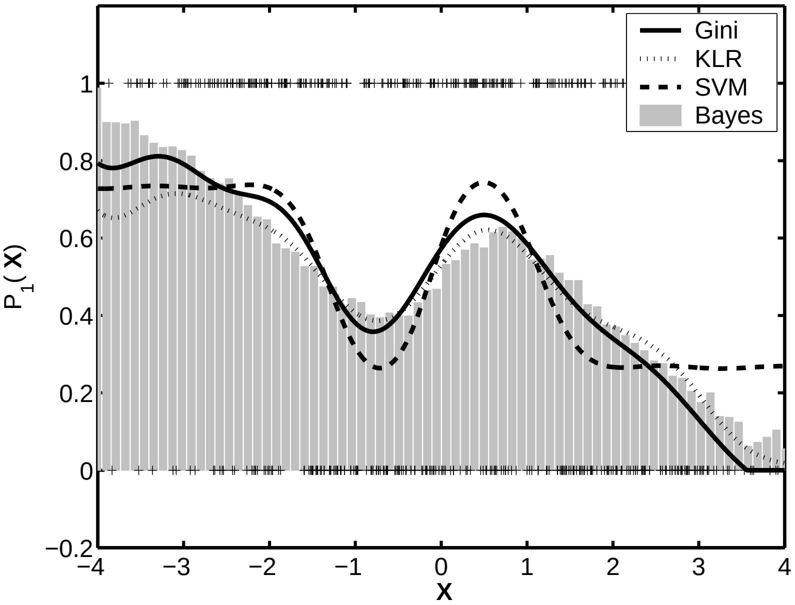


Figure 6: Probability estimates generated by *Gini*SVM, KLR and a calibrated soft-margin SVM for one-dimensional synthetic training data.

where $\Gamma \geq Sd(N+1)^{d-1}$ with S being the smallest coefficient of the polynomial $H(\{P_{ik}\})$. Then $\{\widehat{P}_{ik}\} \in D$ and $H(\{\widehat{P}_{ik}\}) > H(\{P_{ik}\})$.

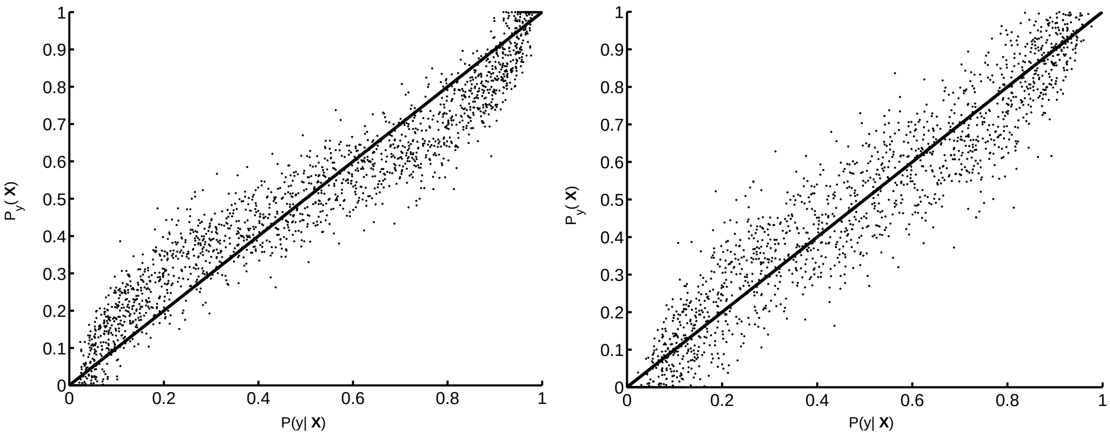
The result can be applied for minimizing the polynomial dual corresponding to Equation (29). Let $P_{ik}^0 = 1/M$ the initial value of the probability distribution for all i,k, and assume the kernel matrix be bounded such that $|Q_{ij}| \leq Q_{max}, \forall i,j$. Also, let P_{ik}^m the value of the probability distribution at m^{th} iteration then

$$P_{ik}^{m+1} \leftarrow P_{ik}^m \delta_{ik}^m / \sum_{k=1}^{M} P_{ik}^m \delta_{ik}^m$$

where

$$\delta_{ik}^m = C \sum_{j=1}^{N} Q_{ij} \, [P_{ik}^m - y_{ik}] + \gamma P_{ik}^m + \Gamma$$

and $\Gamma = C\ (N+1)\ Q_{max}$. At each update the cost function (28) decreases, and the procedure is repeated till convergence. Due to the multiplicative nature of the update some distribution variables P_{ik} can never reach unity or zero; however, in practice it approaches the limits within given margins of precision similar to other implementations of SVM training algorithms. As with other SVM optimization techniques, the speed of large margin growth transformation can be enhanced by using caching and shrinking (Joachims, 1998), as values of the distribution P_{ik} close to unity or zero almost do not change.



(a) *(b)*

Figure 7: Comparison between conditional Bayes probability estimates and scores generated by *Gini*SVM for 10-dimensional synthetic data with *(a)* $\gamma = 0.8$ and *(b)* $\gamma = 0.08$.

6. Experiments and Results

The first set of experiments were designed to characterize the probability scores generated by *Gini*SVM for synthetic data. Figure 6 compares the scores generated by *Gini*SVM, KLR and soft-margin SVM for a synthetic binary classification problem. The one dimensional training data corresponding to two classes were generated using a bimodal Gaussian distribution. A histogram generated by data points randomly sampling the distribution is shown in Figure 6 and the locations of 500 data points used for training are denoted by '+' along $y = 1$ and $y = 0$. For the soft-margin SVM the scores were normalized using Platt's calibration procedure (Platt, 1999a). The Figure 6 shows that the scores generated by KLR, *Gini*SVM and calibrated soft-margin SVM are similar and approximate the sampled distribution which approximates the Bayesian optimum solution. It can be seen that calibrated soft-margin SVM scores do not approximate the true conditional distribution at the boundary of the distribution and would require additional parameterization for producing better estimates.

Figures 7(b) and (c) compare *Gini*SVM scores with sampled conditional scores (Bayes estimates) for synthetic data in 10 dimensions. The data were generated from a multi-variate Gaussian distribution, out of which 100 data points were chosen for training. Figures 7(a) and (b) demonstrate a monotonic relationship between *Gini*SVM scores and Bayes estimate of class conditional probabilities. The sigmoidal relationship trend shown in the scatter plot 7(a) for $\gamma = 0.8$ is attributed to linear approximation of the logistic model (14) by subtractive normalization model (20).

The performance of *Gini*SVM based classifier was evaluated on three benchmark UCI databases and compared with a baseline one-vs-all soft-margin SVM classification method (Weston and Watkins, 1998; Crammer and Singer, 2000). Table 1 summarizes the results obtained for the *Gini*SVM classifier. Data sets are labeled with attributes as (N, D, M) where N denotes its total size, D denotes the dimension of the input vector and M denotes the total number of classes. All training data were normalized between $[-1, 1]$ and a 10-fold cross validation procedure was used to obtain average classification error rate and average number of support vectors. A Gaussian

Iris **(150,3,4)**			**Ecoli** **(336,8,7)**			**Glass** **(214,6,13)**		
(γ,C)	**Err(%)**	**nsv(%)**	**(γ,C)**	**Err(%)**	**nsv(%)**	**(γ,C)**	**Err(%)**	**nsv(%)**
(0.8, 0.5)	3.2 ± 2	40 ± 1.6	(0.8,1)	14 ± 3	60 ± 5	(0.8,1)	30 ± 4.7	95 ± 1.2
(0.8, 5)	4.0 ± 2	19 ± 2.2	(0.8,10)	15 ± 1.8	62 ± 7	(0.8,10)	29 ± 3	89 ± 1.5
(0.08, 5)	4.8 ± 2	14 ± 3	(0.08,1)	12.7 ± 2.7	63 ± 7	(0.08,10)	32 ± 6	82 ± 2.6
Baseline one-vs-all SVM								
(C = 4)	3.4 ± 3	18 ± 1	(C = 10)	14 ± 4	61 ± 6	(C = 5)	30 ± 2	81 ± 3

Table 1: Performance of *Gini*SVM classifier on UCI database.

kernel $K(\mathbf{x},\mathbf{y}) = \exp(-\frac{1}{2\sigma^2}(\mathbf{x}-\mathbf{y})^T(\mathbf{x}-\mathbf{y}))$ was chosen for all experiments. The kernel parameter σ and the regularization parameter C were chosen based on the performance of the baseline SVM on a held-out set. The same kernel parameter was used for training *Gini*SVM classifiers. Table 1 shows the error rate (indicated by **Err**) and the number of support/error vectors (indicated by **nsv**) obtained for different sets of hyper-parameters γ and C. The results indicate that the classification performance of the *Gini*SVM based system is comparable to the baseline one-vs-all SVM system. The results also illustrate the effect of γ on the sparseness of the solution which increases as $\gamma \rightarrow 0$, as explained using the generalized dual framework in Section 4.

6.1 Face Detection and Effects of Parameter Mismatch

The advantage of *Gini*SVM over conventional soft-margin SVM is demonstrated by performing sensitivity analysis on the kernel expansion at completion of training. For this experiment a face detection task was chosen. The classifiers were trained using the face detection database available through CBCL at MIT (Alvira and Rifkin, 2001) and their performance was evaluated on the standard CMU-MIT test set (Rowley et al., 1998). Training of the classifier was performed by utilizing floating point precision arithmetic, whereas evaluation was performed after quantizing the support vectors and inference parameters to $8, 6$ and 4 bits, and adding 1 LSB of uniform random noise. For this experiment the parameter C was determined by optimizing the performance of the classifier on a held-out data set. Receiver operating characteristics (ROC) were obtained by evaluating the performance of the mismatched classifier on the test set. Figure 8 compares ROC curves for the classifier trained with soft-margin SVM, vs. another trained identically with *Gini*SVM for a 2^{nd} order polynomial kernel. The results indicate that *Gini*SVM solution is more robust to mismatch and precision errors in the inference parameters. In fact for this data set, the *Gini*SVM solution quantized to 1 bit is more robust than an equivalent soft-margin SVM solution quantized to 4 bits.

6.2 Speaker Verification Experiments

The benefit of normalized scores generated by *Gini*SVM is demonstrated for the task of text-independent speaker verification. The task entails verifying a particular speaker from possible imposters without any knowledge of the text spoken. A conventional approach uses a classifier to generate scores based on individual speech frames. The scores are integrated over the duration of the utterance and compared against a threshold to accept or reject the speaker. A YOHO speaker verification database was chosen for training and testing the speaker verification system. The YOHO database consists of sets of 4 combination lock phrases spoken by 168 speakers. For each utterance

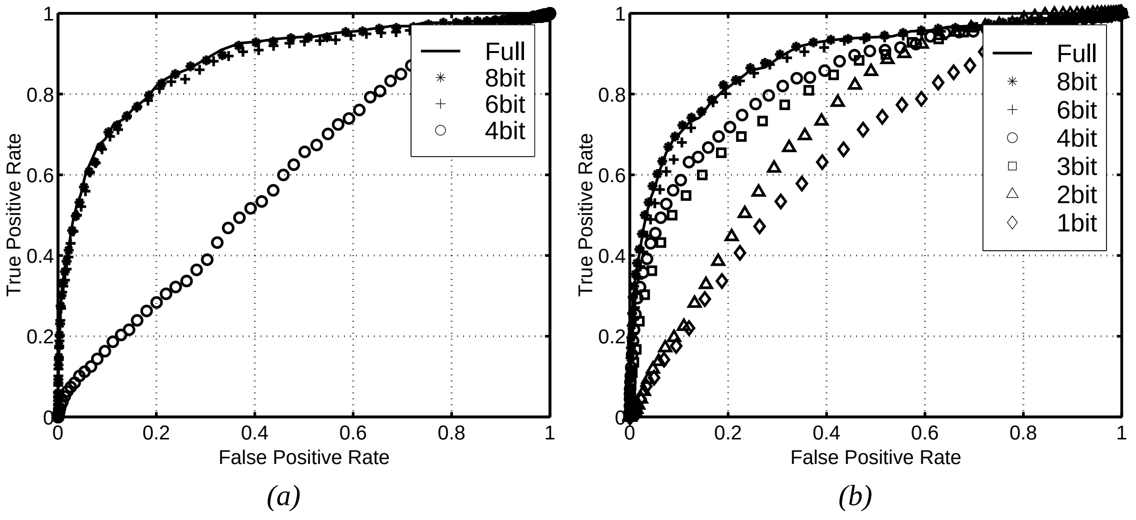


Figure 8: ROC obtained for a face detection system trained with *(a)*: soft-margin SVM and *(b)*: *Gini*SVM training algorithm, for a 2^{nd} order polynomial kernel.

contiguous 25ms speech samples were extracted and Mel-frequency cepstral coefficient (MFCC) features were extracted. The MFCC feature extraction procedure has been extensively studied in the literature and details can be found in Rabiner and Juang (1993). A 39-dimensional feature vector was formed by concatenating the total energy in the speech frame, along with the Δ and $\Delta - \Delta$ MFCC coefficients.

For training, 100 speakers (speaker ID: 101-200) were chosen from the YOHO database and MFCC features were extracted for all speech frames corresponding to each speaker. To reduce the total number of training points, a K-means clustering was performed for each speaker to obtain 1000 cluster points for the correct speaker, and 100 cluster points for each imposter speaker. For each speaker (101-200), this procedure was repeated to obtain a training set of $10,900$ MFCC vectors. Classifiers specific to each speaker were trained using a *Gini*SVM toolkit (http://bach.ece.jhu.edu/svm/ginisvm). For testing utterances corresponding to 100 speakers were chosen from the YOHO test set. Confidence scores generated by *Gini*SVM for each speech frame were integrated over the duration of the utterance to obtain the final cumulative score. Thus each speech frame is treated to be independent and their scores are integrated together without taking into account any time-based correlations.

Figure 9 compares the ROC obtained by a soft-margin SVM based system with a *Gini*SVM based verification system trained for one speaker (id: 148). The speaker with worst verification performance among all was selected. Figure 9 shows that a *Gini*SVM based system exhibits better verification performance compared to an equivalent soft-margin SVM. For each ROC (one per speaker) an equal error rate (EER) parameter was computed. The EER metric is widely used for quantifying performance of a biometric system and is defined as the error rate at which total false positive rate is equal to false rejection rate. Thus, the lower the EER, the more robust is the performance of a biometric system. For this experiments EERs corresponding to each speaker verification system (101-200) were averaged to obtain an equivalent system EER. For a soft-margin SVM and

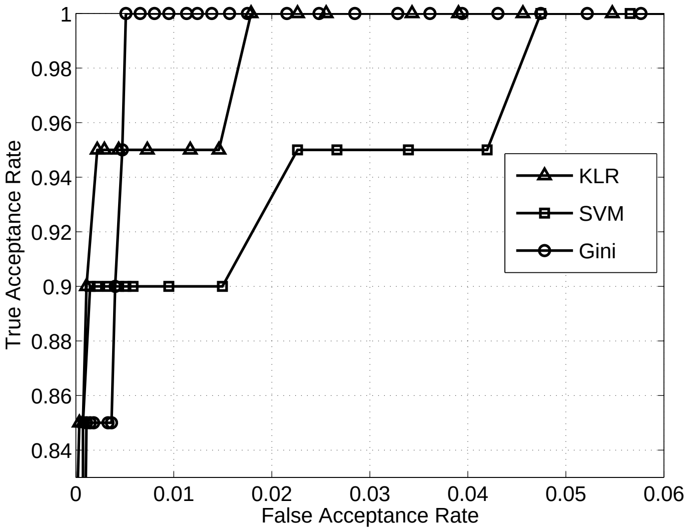


Figure 9: Comparison of ROC obtained for a speaker verification system based on soft-margin SVM and *Gini*SVM classification for speaker id: 148.

KLR, the average EER was computed to be equal to 0.36% and 0.35%, where as the EER for a *Gini*SVM based system was found to 0.28%. This demonstrates that the normalization procedure used by *Gini*SVM improves the accuracy of a text-independent speaker verification system. The verification results are also comparable with other reported results on the YOHO data set (Campbell et al., 2002).

7. Conclusions and Extensions

We introduced a general, maximum entropy based framework for constructing multi-class support vector machines that generate normalized scores. In particular, *Gini*SVM produces direct estimates of conditional probabilities that approximate kernel logistic regression (KLR) at reduced computational cost, incurring quadratic programming under linear constraints as with standard SVM training. Unlike a baseline soft-margin SVM based system with calibrated probabilities, *Gini*SVM produces unbiased probability estimates owing to symmetry in the agnostic distance metric in the maximum entropy formulation. The probability estimates are sparse, where the number of non-zero probabilities is controlled by a single parameter γ, which acts as a margin in the normalization of probability scores. The margin parameter γ is distinct from the regularization parameter C also found in soft-margin SVM and KLR, even though both C and γ weigh the agnostic metric relative to the prior metric in the maximum entropy primal cost function.

For efficient implementation of *Gini*SVM training, we presented a modified sequential minimum optimization (SMO) algorithm, and a multiplicative update algorithm based on growth transformation on probability functions in the dual. The performance of *Gini*SVM probability regression and classification was evaluated on benchmark UCI databases in comparison with KLR and soft-margin SVM. Results on face detection database indicated that the solution obtained by *Gini*SVM training is more robust to mismatch in inference parameters, offering advantages in efficient, reduced precision implementation of SVMs. *Gini*SVM also successfully trained on a task of text-independent speaker verification, by integrating normalized probability scores over time. *Gini*SVM further extends to forward decoding kernel machines for trainable dynamic probabilistic inference on graphs (Chakrabartty and Cauwenberghs, 2002).

The maximum entropy framework for large-margin kernel probability regression introduced for *Gini*SVM is general and can be extended to other classification and regression tasks based on polynomial entropy. Of particular interest are formulations that use symmetric potential functions like the Gini quadratic entropy function.

Acknowledgments

This work was supported by grants from the Catalyst Foundation (*http://www.catalyst-foundation.org*), the National Science Foundation, and the Office of Naval Research.

Appendix A. Kernel Logistic Regression Primal and Dual Formulation

Proof of Proposition I: Define L_e as the regularized log-likelihood/cross entropy for kernel logistic regression (Wahba, 1998; Zhu and Hastie, 2002)

$$L_e = \sum_{k=1}^{M} \frac{1}{2}||\mathbf{w}_k||^2 - C\sum_{i=1}^{N}[\sum_{k=1}^{M} y_{ik} f_k(\mathbf{x}_i) - \log(e^{f_1(\mathbf{x}_i)} + \ldots + e^{f_M(\mathbf{x}_i)})] \; . \tag{30}$$

First order conditions with respect to parameters $\mathbf{w}_k$ and b_k in $f_k(\mathbf{x}) = \mathbf{w}_k.\mathbf{x} + b_k$ yield

$$\begin{aligned} \mathbf{w}_k &= C\sum_{i=1}^{N}[y_{ik} - \frac{e^{f_k(\mathbf{x}_i)}}{\sum_p^M e^{f_p(\mathbf{x}_i)}}]\; \mathbf{x}_i, \\ 0 &= C\sum_{n}^{N}[y_{ik} - \frac{e^{f_k(\mathbf{x}_i)}}{\sum_p^M e^{f_p(\mathbf{x}_i)}}] \; . \end{aligned} \tag{31}$$

Denote

$$\lambda_k^n = C[y_{ik} - \frac{e^{f_k(\mathbf{x}_i)}}{\sum_p^M e^{f_p(\mathbf{x}_i)}}] \tag{32}$$

in the first-order conditions (31) to arrive at the kernel expansion (17) with linear constraint

$$\begin{aligned} f_k(\mathbf{x}) &= \sum_n \lambda_k^n K(\mathbf{x}_i, \mathbf{x}) + b_k, \\ 0 &= \sum_n \lambda_k^n \; . \end{aligned} \tag{33}$$

Note also that $\sum_{k=1}^{M} \lambda_k^n = 0$ by construction.

Legendre transformation of the primal objective function (30) in $\mathbf{w}_k$ and b_k leads to a dual formulation directly in terms of the coefficients λ_k^n (Jaakkola and Haussler, 1999). Define $z_n = \log(\sum_p^M e^{f_p(\mathbf{x}_i)})$, and $Q_{ij} = K(\mathbf{x}_i, \mathbf{x}_j)$. Then (32) and (33) transform to

$$\sum_l Q_{nl}\lambda_k^l - \log[y_{nk} - \lambda_k^n/C] + b_k - z_n = 0$$

which correspond to first-order conditions of the convex dual functional

$$H_e = \sum_{k=1}^{M} [\frac{1}{2}\sum_{n}^{N}\sum_{l}^{N}\lambda_k^n Q_{nl}\lambda_k^l + C\sum_{n}^{N}(y_{nk} - \lambda_k^n/C)\log(y_{nk} - \lambda_k^n/C)]$$

under constraints

$$\begin{aligned} \sum_n \lambda_k^n &= 0, & (34)\\ \sum_k \lambda_k^n &= 0, & (35)\\ \lambda_k^n &\leq Cy_{nk} \end{aligned}$$

where b_k and z_n serve as Lagrange parameters for the equality constraints (34) and (35).

Appendix B. *GiniSVM* Primal and Dual Formulation

Proof of Proposition II: The hinge function $[.]_+$ can be appropriately modeled by introducing slack variables $\mu_{ik} \geq 0$ into the loss function such that

$$L_g(\mathbf{w}_k, b_k, z_i) = \min_{\mu_{ik}\geq 0} \frac{1}{2}\sum_k |\mathbf{w}_k|^2 + \frac{\gamma C}{2}\sum_{ik}(y_{ik} - \frac{1}{\gamma}[f_k(\mathbf{x}_i) - z_i + \mu_{ik}])^2 + \sum_{ik}\eta_{ik}\mu_{ik}$$

The first order conditions corresponding to the variables $\mathbf{w}_k, b_k, z_i, \mu_{ik}$ are given by

$$\begin{aligned} \partial F/\partial \mathbf{w}_k &= \mathbf{w}_k - C\sum_{i=1}^{N}\left(y_{ik} - \frac{1}{\gamma}[f_k(\mathbf{x}_i) - z_i + \mu_{ik}]\right)\mathbf{x}_i = 0, & (36)\\ \partial F/\partial b_k &= C\sum_{i=1}^{N}\left(y_{ik} - \frac{1}{\gamma}[f_k(\mathbf{x}_i) - z_i + \mu_{ik}]\right) = 0, & (37)\\ \partial F/\partial z_i &= C\sum_k\left(y_{ik} - \frac{1}{\gamma}[f_k(\mathbf{x}_i) - z_i + \mu_{ik}]\right) = 0 & (38)\\ \partial F/\partial \mu_{ik} &= \eta_{ik} - \frac{1}{\gamma}[f_k(\mathbf{x}_i) - z_i + \mu_{ik}] = 0 & (39) \end{aligned}$$

where η_{ik} are the Lagrange multipliers corresponding to the inequality conditions $\mu_{ik} \geq 0$. The complementary slackness criterion (Bertsekas, 1995) for these constraints gives $\eta_{ik} \geq 0$ and $\eta_{ik}\mu_{ik} = 0$ which along with criterion (39) gives

$$\frac{1}{\gamma}[f_k(\mathbf{x}_i) - z_i + \mu_{ik}] = \frac{1}{\gamma}[f_k(\mathbf{x}_i) - z_i]_+ \geq 0 \tag{40}$$

and, according to (38),

$$\sum_k \frac{1}{\gamma}[f_k(\mathbf{x}_i) - z_i]_+ = 1$$

which proves the first part of the proposition.

To prove the second part of the proposition let

$$\lambda_k^i = Cy_{ik} - \frac{1}{\gamma}[f_k(\mathbf{x}_i) - z_i + \mu_{ik}]). \tag{41}$$

Criteria (37), (38) and (40) lead to the constraints (11). Substitution in (36) yields an expansion of $\mathbf{w}_k$ which re-substituted in the primal yields the dual first-order condition

$$\sum_j Q_{ij}\lambda_k^j + b_k - z_i + \mu_k^i - \gamma(y_{ik} - \lambda_k^i/C) = 0.$$

Along with constraints (11) the corresponding dual reduces to the *Gini*SVM dual H_g (19), which completes proof of the proposition.

Appendix C. Binary *Gini*SVM Dual Formulation

For a binary *Gini*SVM the dual cost function (19) becomes

$$H_g = \frac{1}{2C}\sum_{ij}\lambda_{+1}^i\lambda_{+1}^j Q_{ij} + \frac{1}{2C}\sum_{ij}\lambda_{-1}^i\lambda_{-1}^j Q_{ij} + \frac{\gamma}{2}\sum_i (y_{i,+1} - \lambda_{+1}^i/C)^2 + \frac{\gamma}{2}\sum_i (y_{i,-1} - \lambda_{-1}^i/C)^2 \quad (42)$$

and the constraints (11) are written as

$$\lambda_{-1}^i = -\lambda_{+1}^i, \quad (43)$$

$$\sum_{i=1}^N \lambda_{-1}^i = \sum_{i=1}^N \lambda_{+1}^i = 0, \quad (44)$$

$$\lambda_{+1}^i \leq C y_{i,+1}, \quad (45)$$

$$\lambda_{-1}^i \leq C y_{i,-1}. \quad (46)$$

Let $\lambda^i = y_i\lambda_{+1}^i$ where $y_i = (2y_{i,+1} - 1)$. Then $f(\mathbf{x}) = \frac{1}{2}(f_{+1}(\mathbf{x}) - f_{-1}(\mathbf{x}))$ reduces to the kernel expansion (24) with $b = \frac{1}{2}(b_{+1} - b_{-1})$. For binary labels $\mathbf{y}_i = \pm 1$ the equality and inequality constraints (43)-(46) simplify to

$$\sum_{i=1}^N \lambda^i y_i = 0,$$

$$0 \leq \lambda^i \leq C.$$

Further substitution of $\lambda_{+1}^i = y_i\lambda^i$, $\lambda_{-1}^i = -y_i\lambda^i$, $y_{i,+1} = \frac{1}{2}(1+y_i)$ and $y_{i,-1} = \frac{1}{2}(1-y_i)$ into the binary *Gini*SVM dual cost function (42)

$$H_g = \frac{1}{C}\sum_{ij}\lambda^i\lambda^j y_i y_j Q_{ij} - \gamma\sum_{i=1}^N\left(\left(\frac{1}{2}\right)^2 - \left(\frac{1}{2} - \frac{\lambda^i}{C}\right)^2\right)$$

which is equivalent to the form H_b (25).

Appendix D. *Gini*SVM Sequential Minimum Optimization

The following extends the original SMO algorithm (Platt, 1999b) from binary soft-margin SVM to multi-class *Gini*SVM.

Let $\lambda_k^{i*} \in C$ be a set of parameters in the constraint space C given by (11). Each iteration a set of four inference parameters, indexed by class identifiers k_1, k_2 and data identifiers i_1, i_2, are jointly updated. Without loss of generality the parameters of this working set will be referred to as $\lambda_1^{1*}, \lambda_2^{1*}, \lambda_1^{2*}$ and λ_2^{2*}, where the indices correspond to k_1, k_2 and i_1, i_2. The aim of an SMO update is to find a new estimate of these coefficients $\lambda_1^1, \lambda_2^1, \lambda_1^2$ and λ_2^2 such that the new set of coefficients affect a net decrease in the objective function H_g (19), while still satisfying constraints C (11). This is ensured by

$$\begin{aligned}
\lambda_1^1 + \lambda_2^1 &= \zeta_1 = \lambda_1^{1*} + \lambda_2^{1*} \\
\lambda_1^1 + \lambda_1^2 &= \xi_1 = \lambda_1^{1*} + \lambda_1^{2*} \\
\lambda_2^1 + \lambda_2^2 &= \xi_2 = \lambda_2^{1*} + \lambda_2^{2*} \\
\lambda_2^2 + \lambda_1^2 &= \zeta_2 = \lambda_2^{2*} + \lambda_1^{2*}.
\end{aligned}$$

Only three of the above equalities need to be satisfied as the fourth one is automatically satisfied. Decomposing the *Gini*SVM dual in terms of these four coefficients leads to

$$\begin{aligned}
H &= \frac{1}{2}Q_{11}(\lambda_1^1)^2 + Q_{12}\lambda_1^1\lambda_1^2 + \frac{1}{2}Q_{22}(\lambda_1^2)^2 + \lambda_1^1 \sum_{j\neq 1,2} Q_{1j}\lambda_1^j + \lambda_1^2 \sum_{j\neq 1,2} Q_{2j}\lambda_1^j \\
&+ \frac{1}{2}Q_{11}(\lambda_2^1)^2 + Q_{12}\lambda_2^1\lambda_2^2 + \frac{1}{2}Q_{22}(\lambda_2^2)^2 + \lambda_2^1 \sum_{j\neq 1,2} Q_{1j}\lambda_2^j + \lambda_2^2 \sum_{j\neq 1,2} Q_{2j}\lambda_2^j \\
&+ \gamma C(y_{11} - \lambda_1^1/C)^2 + \gamma C(y_{21} - \lambda_1^2/C)^2 + \gamma C(y_{12} - \lambda_2^1/C)^2 + \gamma C(y_{22} - \lambda_2^2/C)^2.
\end{aligned}$$

Substituting

$$\begin{aligned}
\lambda_2^1 &= \zeta_1 - \lambda_1^1, \\
\lambda_1^2 &= \xi_1 - \lambda_1^1, \\
\lambda_2^2 &= \xi_2 - \zeta_1 + \lambda_1^1
\end{aligned}$$

and using the first order condition $\partial H/\partial\lambda_1^1 = 0$, optimal values for λ_1^{1*} are found as

$$\lambda_1^{1*} = \lambda_1^1 + (g_{12} + g_{21} - g_{11} - g_{22})/2\eta \tag{47}$$

where

$$g_{lm} = -2\gamma y_{lm} + \sum_j Q_{lj}\lambda_m^j + 2\gamma/C\lambda_m^l$$

and

$$\eta = Q_{11} + Q_{22} - 2Q_{12} + 4\gamma/C.$$

At each step of the update (47) the *Gini*SVM dual function decreases, and repeated sampling of the four-point working set over the training set ensures proper convergence to the true minimum, barring degeneracies in the cost function. At convergence the parameters $b_k, k = 1,..,M$ are obtained by solving a set of overcomplete equations for data points that lie in the interior of the boundary constraints $\lambda_k^i < Cy_{ik}$. For the interior points denoted by its training index i the following condition is satisfied

$$b_k - z_i + g_{ik} = 0$$

which is over-complete in parameters $b_k, k = 1,..,M$ and $z_i, i = 1,..,I$, where I denotes the total number of training points within the interior of the constraints.

References

E.L. Allwein, R.E. Schapire, and Y. Singer. Reducing multiclass to binary: A unifying approach for margin classifiers. *Journal of Machine Learning Research,* 1:113-141, 2000.

M. Alvira and R. Rifkin. An empirical comparison of SNoW and SVMs for face detection. *CBCL paper 193 /AI Memo 2001-2004*, MIT, 2001.

R. Auckenthaler, M. Carey and H. Lloyd-Thomas. Score normalization for text-independent speaker verification system. *Digital Signal Processing,* 10(1):42-54, 2000.

L.E. Baum and G. Sell. Growth transformations for functions on manifolds. *Pacific J. Math.*, 27(2):211-227, 1968.

D. Bertsekas. *Non-linear Programming.* Athena Scientific, MA, 1995.

B. Boser, I. Guyon and V. Vapnik. A training algorithm for optimal margin classifier. *Proc. 5th Ann. ACM Workshop on Computational Learning Theory (COLT),* pages 144-52, 1992.

L. Breiman, J.H. Friedman and R. Olshen. *Classification and Regression Trees.* Wadsworth and Brooks, Pacific Grove CA, 1984.

C. Burges. A tutorial on support vector machines for pattern recognition. U. Fayyad, Ed., *Proc. Data Mining and Knowledge Discovery,* pages 1-43, 1998.

W.M. Campbell, K.T. Assaleh and C.C. Broun. Speaker with polynomial classifiers. *IEEE Trans. Speech and Audio Proc.,* 10(4):205-212, May 2002.

G. Cauwenberghs and T. Poggio. Incremental and decremental support vector machine learning. *Adv. Neural Information Processing Systems 10,* Cambridge MA: MIT Press, 2001.

S. Chakrabartty and G. Cauwenberghs. Forward decoding kernel machines: A hybrid HMM/SVM approach to sequence recognition. *IEEE Int. Conf. of Pattern Recognition: SVM workshop. (ICPR'2002),* 2002.

S. Chakrabartty and G. Cauwenberghs. Margin propagation and forward decoding in analog VLSI. *Proc. IEEE Int. Symp. Circuits and Systems (ISCAS'2004)*, 2004.

S. Chakrabartty and G. Cauwenberghs. Sub-microwatt analog VLSI support vector machine for pattern classification and sequence estimation. *Adv in Neural Information Processing Systems 17*, Cambridge: MIT Press, 2005.

T.M. Cover and J.A. Thomas, *Elements of Information Theory.* John Wiley and Sons, 1991.

K. Crammer and Y. Singer. The learnability and design of output codes for multiclass problems. *Proc. 13th Ann. Conf. Computational Learning Theory (COLT),* 2000.

T.G. Dietterich and G. Bakiri. Solving multiclass learning problems via error-correcting output codes. *Journal of Artificial Intelligence Research*, 2:263-286, 1995.

F. Girosi, M. Jones and T. Poggio. Regularization theory and neural networks architectures. *Neural Computation*, 7:219-269, 1995.

P.S. Gopalakrishnan, D. Kanevsky, A. Nadas and D. Nahamoo. An inequality for rational functions with applications to some statistical estimation problems. *IEEE Trans. Information Theory*, 37(1):107-113, 1991.

Y. Gu and T. Thomas. A text-independent speaker verification system using support vector machines classifier. *Proc. Eur. Conf. Speech Communication and Technology (Eurospeech'01)*, pages 1765-1769, 2001.

C. Hsu and C. Lin. A comparison of methods for multi-class support vector machines. *IEEE Trans. Neural Networks*, 13(2):415-425, 2002.

P.J. Huber. Robust Estimation of Location Parameter. *Annals of Mathematical Statistics*, volume 35, 1964.

T. Jaakkola and D. Haussler. Probabilistic kernel regression models. *Proc. 7th Int. Workshop on Artificial Intelligence and Statistics*, 1999.

E. Jaynes. Information theory and statistical mechanics. *Physics Review*, 106:620-630, 1957.

T. Jebara. Discriminative, generative and imitative learning. PhD Thesis, MIT Media Laboratory, 2001.

T. Joachims. Text categorization with support vector machines. Technical Report LS-8 23, Univ. of Dortmund, 1997.

T. Joachims. Making large-scale support vector machine learning practical, In Schölkopf, Burges and Smola, Eds., *Advances in Kernel Methods: Support Vector Machines*, Cambridge MA: MIT Press, 1998.

M.I. Jordan and R.A. Jacobs. Hierarchical mixtures of experts and the EM algorithm. *Proc. Int. Joint Conference on Neural Networks*, 2:1339-1344, 1993.

S.S. Keerthi, S.K. Shevade, C. Bhattacharyya and K.R.K. Murthy. Improvements to Platt's SMO algorithm for SVM classifier design. *Neural Computation,* 13:637-649, 2001.

J.T.Y. Kwok. Moderating the outputs of support vector machine classifiers. *IEEE Transactions on Neural Networks*, 10(5):1018-1031, 1999.

T. Nayak and C.R. Rao. Cross entropy, dissimilarity measures and characterizations of quadratic entropy. *IEEE Trans. Information Theory*, IT-31:589-593, 1985.

N. Lawrence, M. Seeger and R. Herbrich. Fast sparse gaussian process methods: The informative vector machine. *Neural Information Processing Systems 15*, pages 609-616, 2003.

M. Oren, C. Papageorgiou, P. Sinha, E. Osuna and T. Poggio. Pedestrian detection using wavelet templates. *Computer Vision and Pattern Recognition (CVPR),*pages 193-199, 1997.

E. Osuna, R. Freund and F. Girosi. Training support vector machines: An application to face detection. *Computer Vision and Pattern Recognition*, pages 130-136, 1997.

D.S. Pietra and D.V. Pietra. Statistical Modeling by ME. IBM Internal Report, 1993.

J. Platt. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In A. Smola et al., Eds., *Adv. Large Margin Classifiers,* Cambridge MA: MIT Press, 1999.

J. Platt. Fast training of support vector machine using sequential minimal optimization. In Scholkopf, Burges and Smola, Eds., *Adv. Kernel Methods,* Cambridge MA: MIT Press, 1999.

M. Pontil and A. Verri. Properties of support vector machines. *Neural Computation,* 10:977-996, 1998.

M. Pontil and A. Verri. Support Vector Machines for 3-D Object Recognition. *IEEE Transactions of Pattern Analysis and Machine Intelligence,* 20:637-646, 1998.

L. Rabiner and B.H. Juang. *Fundamentals of Speech Recognition,* Englewood Cliffs, NJ: Prentice-Hall, 1993.

R.T. Rockefeller. *Convex Analysis.* Princeton Landmarks in Mathematics and Physics, Princeton University Press, 1970.

H.A. Rowley, S.A. Baluja and T. Kanade. Neural network based face detection. *IEEE Transactions of Pattern Analysis and Machine Intelligence,* 20(1):23-38, 1998.

M. Schmidt and H. Gish. Speaker identification via support vector classifiers. *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing (ICASSP'96),* 1:105-108, 1996.

B. Schölkopf, C. Burges and A. Smola Eds., *Adv. Kernel Methods-Support Vector Learning,* MIT Press, Cambridge MA, 1998.

B. Schölkopf and A. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond,* MIT Press, Cambridge MA, 2001.

M.E. Tipping. Sparse Bayesian learning and the relevance vector machine. *Journal of Machine Learning Research,* 1:211-244, 2001.

V. Vapnik. *The Nature of Statistical Learning Theory.* New York: Springer-Verlag, 1995.

G. Wahba. Support vector machines, reproducing kernel Hilbert spaces and randomized GACV. *Adv. Kernel Methods-Support Vector Learning,* B. Schölkopf, C.J.C. Burges and A.J. Smola, Eds., Cambridge MA: MIT Press, 1998.

J. Weston and C. Watkins. Multi-class support vector machines. Technical Report CSD-TR-9800-04, Department of Computer Science, Royal Holloway, Univ. London, 1998.

J. Zhu and T. Hastie. Kernel logistic regression and import vector machine. *Adv. Neural Information Processing Systems (NIPS'2001),* Cambridge, MA: MIT Press, 2002.