Journal of Machine Learning Research 16 (2015) 2751-2795 Submitted 7/12; 12/14; Published 12/15

A View of Margin Losses as Regularizers of Probability Estimates

Hamed Masnadi-Shirazi HMASNADI@SHIRAZU.AC.IR
School of Electrical and Computer Engineering,
Shiraz University,
Shiraz, Iran

Nuno Vasconcelos NUNO@UCSD.EDU
Statistical Visual Computing Laboratory,
University of California, San Diego
La Jolla, CA 92039, USA

Editor: Saharon Rosset

Abstract

Regularization is commonly used in classifier design, to assure good generalization. Classical regularization enforces a cost on classifier complexity, by constraining parameters. This is usually combined with a margin loss, which favors large-margin decision rules. A novel and unified view of this architecture is proposed, by showing that margin losses act as regularizers of posterior class probabilities, in a way that amplifies classical parameter regularization. The problem of controlling the regularization strength of a margin loss is considered, using a decomposition of the loss in terms of a link and a binding function. The link function is shown to be responsible for the regularization strength of the loss, while the binding function determines its outlier robustness. A large class of losses is then categorized into equivalence classes of identical regularization strength or outlier robustness. It is shown that losses in the same regularization class can be parameterized so as to have tunable regularization strength. This parameterization is finally used to derive boosting algorithms with loss regularization (BoostLR). Three classes of tunable regularization losses are considered in detail. Canonical losses can implement all regularization behaviors but have no flexibility in terms of outlier modeling. Shrinkage losses support equally parameterized link and binding functions, leading to boosting algorithms that implement the popular shrinkage procedure. This offers a new explanation for shrinkage as a special case of loss-based regularization. Finally, α-tunable losses enable the independent parameterization of link and binding functions, leading to boosting algorithms of great flexibility. This is illustrated by the derivation of an algorithm that generalizes both AdaBoost and LogitBoost, behaving as either one when that best suits the data to classify. Various experiments provide evidence of the benefits of probability regularization for both classification and estimation of posterior class probabilities.

Keywords: classification, margin losses, regularization, boosting, probability elicitation, generalization, loss functions, link functions, binding functions, shrinkage

1. Introduction

The ability to generalize beyond the training set is a central challenge for classifier design. A binary classifier is usually implemented by thresholding a continuous function, the classifier predictor, of a high-dimensional feature vector. Predictors are frequently affine functions, whose level sets (decision boundaries) are hyperplanes in feature space. Optimal predictors minimize the empirical expectation of a loss function, or risk, on a training set. Modern risks guarantee good generalization by enforcing large margins and parameter regularization. Large margins follow from the use of margin losses, such as the hinge loss of the support vector machine (SVM), the exponential loss of AdaBoost, or the logistic loss of logistic regression and LogitBoost. These are all upper-bounds on the zero-one classification loss of classical Bayes decision theory. Unlike the latter, margin losses assign a penalty to examples correctly classified but close to the boundary. This guarantees a classification margin and improved generalization (Vapnik, 1998). Regularization is implemented by penalizing predictors with many degrees of freedom. This is usually done by augmenting the risk with a penalty on the norm of the parameter vector. Under a Bayesian interpretation of risk minimization, different norms correspond to different priors on predictor parameters, which enforce different requirements on the sparseness of the optimal solution.

While for some popular classifiers, e.g. the SVM, regularization is a natural side-product of risk minimization under a margin loss (Moguerza and Munoz, 2006; Chapelle, 2007; Huang et al., 2014), the relation between the two is not always as clear for other learning methods, e.g. boosting. Regularization can be added to boosting (Buhlmann and Hothorn, 2007; Lugosi and Vayatis, 2004; Blanchard et al., 2003) in a number of ways, including restricting the number of boosting iterations (Raskutti et al., 2014; Natekin and Knoll, 2013; Zhang and Yu, 2005; Rosset et al., 2004; Jiang, 2004; Buhlmann and Yu, 2003), adding a regularization term (Saha et al., 2013; Culp et al., 2011; Xiang et al., 2009; Bickel et al., 2006; Xi et al., 2009), restricting the weight update rule (Lozano et al., 2014, 2006; Lugosi and Vayatis, 2004; Jin et al., 2003) or using divergence measures (Liu and Vemuri, 2011) and has been implemented for both the supervised and semi-supervised settings (Chen and Wang, 2008, 2011). However, many boosting algorithms lack explicit parameter regularization. Although boosting could eventually overfit (Friedman et al., 2000; Rosset et al., 2004), and there is an implicit regularization when the number of boosting iterations is limited (Raskutti et al., 2014; Natekin and Knoll, 2013; Zhang and Yu, 2005; Rosset et al., 2004; Jiang, 2004; Buhlmann and Yu, 2003), there are several examples of successful boosting on very high dimensional spaces, using complicated ensembles of thousands of weak learners, and no explicit regularization (Viola and Jones, 2004; Schapire and Singer, 2000; Viola et al., 2003; Wu and Nevatia, 2007; Avidan, 2007). This suggests that regularization is somehow implicit in large margins, and additional parameter regularization may not always be critical, or even necessary. In fact, in domains like computer vision, large margin classifiers are more popular than classifiers that enforce regularization but not large margins, e.g. generative models with regularizing priors. This suggests that the regularization implicit in large margins is complementary to parameter regularization. However, this connection has not been thoroughly studied in the literature.

In this work, we approach the problem by studying the properties of margin losses. This builds on prior work highlighting the importance of three components of risk mini-

mization: the loss ϕ, the minimum risk C^*_ϕ, and a link function f^*_ϕ that maps posterior class probabilities to classifier predictions (Friedman et al., 2000; Zhang, 2004; Buja et al., 2006; Masnadi-Shirazi and Vasconcelos, 2008; Reid and Williamson, 2010). We consider the subset of losses of invertible link, since this enables the recovery of class posteriors from predictor outputs. Losses with this property are known as proper losses and important for applications that require estimates of classification confidence, e.g. multiclass decision rules based on binary classifiers (Zadrozny, 2001; Rifkin and Klautau, 2004; Gonen et al., 2008; Shiraishi and Fukumizu, 2011). We provide a new interpretation of these losses as regularizers of finite sample probability estimates and show that this regularization has at least two important properties for classifier design. First, it combines multiplicatively with classical parameter regularization, amplifying it in a way that tightens classification error bounds. Second, probability regularization strength is proportional to loss margin for a large class of link functions, denoted generalized logit links. This enables the introduction of tunable regularization losses ϕ_σ, parameterized by a probability regularization gain σ. A procedure to derive boosting algorithms of tunable loss regularization (BoostLR) from these losses is also provided. BoostLR algorithms generalize the GradientBoost procedure (Friedman, 2001), differing only in the example weighting mechanism, which is determined by the loss ϕ_σ.

To characterize the behavior of these algorithms, we study the space $\mathcal{R}$ of proper losses ϕ of generalized logit link. It is shown that any such ϕ is uniquely defined by two components: the link f^*_ϕ and a binding function β_ϕ that maps f^*_ϕ into the minimum risk C^*_ϕ. This decomposition has at least two interesting properties. First, the two components have a functional interpretation: while f^*_ϕ determines the probability regularization strength of ϕ, β_ϕ determines its robustness to outliers. Second, both β_ϕ and f^*_ϕ define equivalence classes in $\mathcal{R}$. It follows that $\mathcal{R}$ can be partitioned into subsets of losses that have either the same outlier robustness or probability regularization properties. It is shown that the former are isomorphic to a set of symmetric scale probability density functions and the latter to the set of monotonically decreasing odd functions. Three loss classes, with three different binding functions, are then studied in greater detail. The first, the class of canonical losses, consists of losses of linear binding function. This includes some of the most popular losses in the literature, e.g. the logistic. While they can implement all possible regularization behaviors, these losses have no additional degrees of freedom. In this sense, they are the simplest tunable regularization losses. This simplicity enables a detailed analytical characterization of their shape and how this shape is affected by the regularization gain. The second, the class of shrinkage losses, is a superset of the class of canonical losses. Unlike their canonical counterparts, shrinkage losses support nonlinear binding functions, and thus more sophisticated handling of outliers. However, they require an identical parameterization of the link and binding function. It is shown that, under this constraint, BoostLR implements the popular shrinkage regularization procedure (Hastie et al., 2001). Finally, the class of α-tunable losses enables independent parameterization of the link and binding functions. This endows the losses in this class, and the associated BoostLR algorithms, with a great deal of flexibility. We illustrate this by introducing an α-tunable loss that generalizes both the exponential loss of AdaBoost and the logistic loss of LogitBoost, allowing BoostLR to behave as either of the two algorithms, so as to best suit the data to classify.

The paper is organized as follows. Section 2 briefly reviews classifier design by risk minimization. The view of margin losses as regularizers of probability estimates is introduced in Section 3. Section 4 characterizes the regularization strength of proper losses of generalized logit link. Tunable regularization losses and binding functions are introduced in Section 5, which also introduces the BoostLR algorithm. The structure of $\mathcal{R}$ is then characterized in Section 6, which introduces canonical, shrinkage, and α-tunable losses. An extensive set of experiments on various aspects of probability regularization is reported in Section 7. Finally, some conclusions are drawn in Section 8.

2. Loss Functions and Risk Minimization

We start by reviewing the principles of classifier design by risk minimization (Friedman et al., 2000; Zhang, 2004; Buja et al., 2006; Masnadi-Shirazi and Vasconcelos, 2008) .

2.1 The Classification Problem

A classifier h maps a feature vector $\mathbf{x} \in \mathcal{X}$ to a class label $y \in \{-1, 1\}$, according to

$$h(\mathbf{x}) = sign[p(\mathbf{x})], \tag{1}$$

where $p : \mathcal{X} \to \mathbb{R}$ is the classifier predictor. Feature vectors and class labels are drawn from probability distributions $P_{\mathbf{X}}(\mathbf{x})$ and $P_{Y|X}(y|\mathbf{x})$ respectively. Given a non-negative loss function $L(\mathbf{x}, y)$, the optimal predictor $p^*(\mathbf{x})$ minimizes the risk

$$R(p) = E_{\mathbf{X},Y}[L(p(\mathbf{x}), y)]. \tag{2}$$

This is equivalent to minimizing the conditional risk

$$E_{Y|\mathbf{X}}[L(p(\mathbf{x}), y)|\mathbf{X} = \mathbf{x}]$$

for all $\mathbf{x} \in \mathcal{X}$. It is frequently useful to express $p(\mathbf{x})$ as a composition of two functions,

$$p(\mathbf{x}) = f(\eta(\mathbf{x})),$$

where $\eta(\mathbf{x}) = P_{Y|\mathbf{X}}(1|\mathbf{x})$ is the posterior probability function, and $f : [0, 1] \to \mathbb{R}$ a link function. The problem of learning the optimal predictor can thus be decomposed into the problems of learning the optimal link $f^*(\eta)$ and estimating the posterior function $\eta(\mathbf{x})$. Since $f^*(\eta)$ can usually be determined analytically, this reduces to estimating $\eta(\mathbf{x})$, whenever $f^*(\eta)$ is a one-to-one mapping.

In classical statistics, learning is usually based on the zero-one loss

$$L_{0/1}(y, p) = \frac{1 - sign(yp)}{2} = \begin{cases} 0, & \text{if } y = sign(p); \\ 1, & \text{if } y \neq sign(p), \end{cases}$$

where we omit the dependence on $\mathbf{x}$ for notational simplicity. The associated conditional risk

$$C_{0/1}(\eta, p) = \eta\frac{1 - sign(p)}{2} + (1 - \eta)\frac{1 + sign(p)}{2} = \begin{cases} 1 - \eta, & \text{if } p = f(\eta) \geq 0; \\ \eta, & \text{if } p = f(\eta) < 0, \end{cases}$$

is the probability of error of the classifier of (1), and is minimized by any f^* such that

$$\begin{cases} f^*(\eta) > 0 & \text{if } \eta > \frac{1}{2} \\ f^*(\eta) = 0 & \text{if } \eta = \frac{1}{2} \\ f^*(\eta) < 0 & \text{if } \eta < \frac{1}{2}. \end{cases} \tag{3}$$

The optimal classifier $h^*(\mathbf{x}) = sign[p^*(\mathbf{x})]$, where $p^* = f^*(\eta)$, is the well known Bayes decision rule, and has minimum conditional (zero-one) risk

$$\begin{aligned} C^*_{0/1}(\eta) &= \eta \left(\frac{1}{2} - \frac{1}{2} sign(2\eta - 1) \right) + (1 - \eta) \left(\frac{1}{2} + \frac{1}{2} sign(2\eta - 1) \right) \\ &= \min\{\eta, 1 - \eta\}. \end{aligned}$$

2.2 Learning from Finite Samples

Practical learning algorithms produce an estimate $\hat{p}^*(\mathbf{x})$ of the optimal predictor by minimizing an empirical estimate of (2), the empirical risk, from a training sample $\mathcal{D} = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\}$

$$R_{emp}(p) = \frac{1}{n} \sum_i L(p(\mathbf{x}_i), y_i). \tag{4}$$

This can be formulated as fitting a model $\hat{\eta}(\mathbf{x}) = [f^*]^{-1}(p(\mathbf{x}; \mathbf{w}))$ to the sample $\mathcal{D}$, where f^* is an invertible link that satisfies (3) and $p(\mathbf{x}; \mathbf{w})$ a parametric predictor. Two commonly used links are

$$f^* = 2\eta - 1 \quad \text{and} \quad f^* = \log \frac{\eta}{1 - \eta}.$$

In this way, the learning problem is reduced to the estimation of the model parameters $\mathbf{w}$ of minimum empirical risk. Most modern learning techniques rely on a linear predictor, implemented on either $\mathcal{X}$ - $p(\mathbf{x}, \mathbf{w}) = \mathbf{w}^T\mathbf{x}$ - or some transformed space - $p(\mathbf{x}, \mathbf{w}) = \mathbf{w}^T\Phi(\mathbf{x})$. For example, logistic regression (Hosmer and Lemeshow, 2000) uses the logit link $f^* = \log \frac{\eta}{1-\eta}$, or equivalently the logistic inverse link $[f^*]^{-1}(v) = \frac{e^v}{1+e^v}$, and learns a linear predictor $p(\mathbf{x}, \mathbf{w}) = \mathbf{w}^T\mathbf{x}$. When a transformation $\Phi(\mathbf{x})$ is used, it is either implemented indirectly with recourse to a kernel function, e.g. kernelized logistic regression (Zhu and Hastie, 2001), or learned. For example, boosting algorithms rely on a transformation $\Phi(\mathbf{x}) = (h_1(\mathbf{x}), \ldots, h_m(\mathbf{x}))$ where $h_i(\mathbf{x})$ is a weak or base classifier selected during training. In this case, the predictor has the form

$$p(\mathbf{x}; \mathbf{w}) = \sum_i w_i h_i(\mathbf{x}). \tag{5}$$

In all cases, given the optimal predictor estimate $\hat{p}^*(\mathbf{x}) = p(\mathbf{x}, \mathbf{w}^*)$, estimates of the posterior probability $\eta(\mathbf{x})$ can be obtained with $\hat{\eta}(\mathbf{x}) = [f^*]^{-1}(\hat{p}^*(\mathbf{x}))$. However, when learning is based on the empirical risk of (4), convergence to the true probabilities is only guaranteed asymptotically and for certain loss functions $L(.,.)$. Even when this is the case, learning algorithms can easily overfit to the training set, for finite samples. The minimum of (4) is achieved for some empirical predictor

$$\hat{p}^*(\mathbf{x}) = p^*(\mathbf{x}) + \epsilon_p(\mathbf{x}), \tag{6}$$

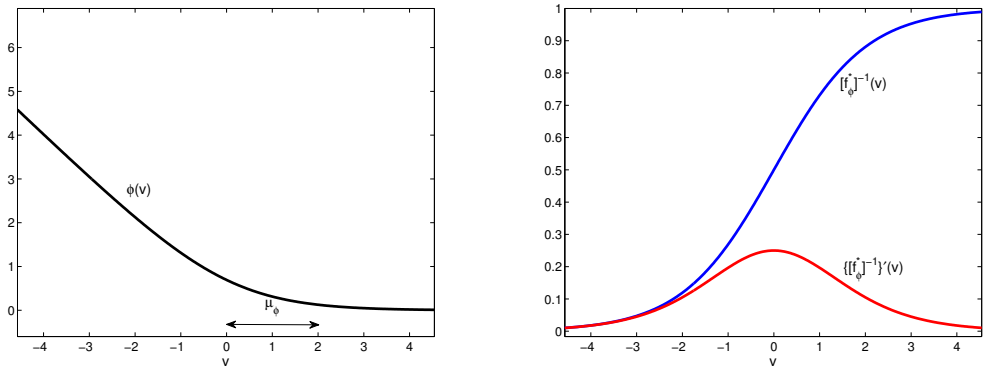


Figure 1: Left: A margin loss function (the logistic loss) of margin parameter μ_ϕ, defined in (25). Right: corresponding inverse link (in blue) and its growth rate (in red).

where $p^*(\mathbf{x})$ is the optimal predictor and $\epsilon_p(\mathbf{x})$ a prediction error, sampled from a zero mean distribution of decreasing variance with sample size. For a given sample size, a predictor with error of smaller variance is said to generalize better. One popular mechanism to prevent overfitting is to regularize the parameter vector $\mathbf{w}$, by imposing a penalty on its norm, i.e. minimizing

$$R_{emp}(p) = \frac{1}{n}\sum_i L(p(\mathbf{x}_i), y_i) + \lambda||\mathbf{w}||_l$$

instead of (4). We refer to this as parameter regularization.

2.3 Margin Losses

Another possibility is to change the loss function, e.g. by replacing the 0-1 loss with a margin loss $L_\phi(y, p(\mathbf{x})) = \phi(yp(\mathbf{x}))$. As illustrated in Figure 1 (left), these losses assign a non-zero penalty to small positive values of the margin yp, i.e. in the range $0 < yp < \mu_\phi$, where μ_ϕ is a parameter, denoted the loss margin. Commonly used margin losses include the exponential loss of AdaBoost, the logistic loss (shown in the figure) of logistic regression, and the hinge loss of SVMs. The resulting large-margin classifiers have better finite sample performance (generalization) than those produced by the 0-1 loss (Vapnik, 1998). The associated conditional risk

$$C_\phi(\eta, p) = C_\phi(\eta, f(\eta)) = \eta\phi(f(\eta)) + (1-\eta)\phi(-f(\eta)) \tag{7}$$

is minimized by the link

$$f^*_\phi(\eta) = \arg\min_f C_\phi(\eta, f) \tag{8}$$

leading to the minimum conditional risk function

$$C^*_\phi(\eta) = C_\phi(\eta, f^*_\phi). \tag{9}$$

Algorithm	$\phi(v)$	$f^*_\phi(\eta)$	$[f^*_\phi]^{-1}(v)$	$C^*_\phi(\eta)$
SVM	$\max(1-v,0)$	$sign(2\eta-1)$	NA	$1-\lvert 2\eta-1\rvert$
Boosting	$\exp(-v)$	$\frac{1}{2}\log\frac{\eta}{1-\eta}$	$\frac{e^{2v}}{1+e^{2v}}$	$2\sqrt{\eta(1-\eta)}$
Logistic Regression	$\log(1+e^{-v})$	$\log\frac{\eta}{1-\eta}$	$\frac{e^{v}}{1+e^{v}}$	$-\eta\log\eta-(1-\eta)\log(1-\eta)$

Table 1: Loss ϕ, optimal link $f^*_\phi(\eta)$, optimal inverse link $[f^*_\phi]^{-1}(v)$, and minimum conditional risk $C^*_\phi(\eta)$ of popular learning algorithms.

Unlike the 0-1 loss, the optimal link is usually unique for margin losses and computable in closed-form, by solving $\eta\phi'(f^*_\phi(\eta)) = (1-\eta)\phi'(-f^*_\phi(\eta))$ for f^*_ϕ. Table 1 lists the loss, optimal link, and minimum risk of popular margin losses.

The adoption of a margin loss can be equivalent to the addition of parameter regularization. For example, a critical step of the SVM derivation is a normalization that makes the margin identical to $1/||\mathbf{w}||$, where $\mathbf{w}$ is the normal of the SVM hyperplane $p(\mathbf{x};\mathbf{w}) = \mathbf{w}^T\mathbf{x}$ (Moguerza and Munoz, 2006; Chapelle, 2007). This renders margin maximization identical to the minimization of hyperplane norm, leading to the interpretation of the SVM as minimizing the hinge loss under a regularization constraint on $\mathbf{w}$ (Moguerza and Munoz, 2006; Chapelle, 2007), i.e.

$$R_{SVM}(\mathbf{w}) = \frac{1}{n}\sum_i \max[0, 1 - yp(\mathbf{x}_i;\mathbf{w})] + \lambda||\mathbf{w}||^2. \tag{10}$$

In this case, larger margins translate directly into the regularization of classifier parameters. This does not, however, hold for all large margin learning algorithms. For example, boosting does not use explicit parameter regularization, although regularization is implicit in early stopping (Raskutti et al., 2014; Natekin and Knoll, 2013; Zhang and Yu, 2005; Rosset et al., 2004; Jiang, 2004; Buhlmann and Yu, 2003). This consists of terminating the algorithm after a small number of iterations. While many bounds have been derived to characterize the generalization performance of large margin classifiers, it is not always clear how much of the generalization ability is due to the loss vs. parameter regularization. In what follows, we show that margin losses can themselves be interpreted as regularizers. However, instead of regularizing predictor parameters, they directly regularize posterior probability estimates, by acting on the predictor output. This suggests a complementary role for loss-based and parameter regularization. We will see that the two types of regularization in fact have a multiplicative effect.

3. Proper Losses and Probability Regularization

We start by discussing the role of margin losses as probability regularizers.

3.1 Regularization Losses

For any margin loss whose link of (8) is invertible, posterior probabilities can be recovered from

$$\eta(\mathbf{x}) = [f_\phi^*]^{-1}(p^*(\mathbf{x})). \tag{11}$$

Whenever this is the case, the loss is said to be proper[1] and the predictor calibrated (DeGroot and Fienberg, 1983; Platt, 2000; Niculescu-Mizil and Caruana, 2005; Gneiting and Raftery, 2007). For finite samples, estimates of the probabilities $\eta(\mathbf{x})$ are obtained from the empirical predictor $\hat{p}^*$ with

$$\hat{\eta}(\mathbf{x}) = [f_\phi^*]^{-1}(\hat{p}^*(\mathbf{x})). \tag{12}$$

Parameter regularization improves estimates $\hat{p}^*(\mathbf{x})$ by constraining predictor parameters. For example, a linear predictor estimate $\hat{p}^*(\mathbf{x};\hat{\mathbf{w}}) = \hat{\mathbf{w}}^T\mathbf{x}$ can be written in the form of (6), with $p^*(\mathbf{x}) = \mathbf{w}^{*T}\mathbf{x}$ and $\epsilon_p(\mathbf{x}) = \mathbf{w}_\epsilon{}^T\mathbf{x}$, where $\mathbf{w}_\epsilon$ is a parameter estimation error. The regularization of (10) reduces $\mathbf{w}_\epsilon$ and the prediction error $\epsilon_p(\mathbf{x})$, improving probability estimates in (12).

Loss-based regularization complements parameter regularization, by regularizing the probability estimates directly. To see this note that, whenever the loss is proper and the noise component ϵ_p of (6) has small amplitude, (12) can be approximated by its Taylor series expansion around p^*

$$\begin{aligned}\hat{\eta}(\mathbf{x}) &\approx [f_\phi^*]^{-1}(p^*(\mathbf{x})) + \{[f_\phi^*]^{-1}\}'(p^*(\mathbf{x}))\epsilon_p(\mathbf{x}) \\ &= \eta(\mathbf{x}) + \epsilon_\eta(\mathbf{x})\end{aligned}$$

with

$$\epsilon_\eta(\mathbf{x}) = \{[f_\phi^*]^{-1}\}'(p^*(\mathbf{x}))\epsilon_p(\mathbf{x}). \tag{13}$$

If $|\{[f_\phi^*]^{-1}\}'(p^*(\mathbf{x}))| < 1$ the probability estimation noise ϵ_η has smaller magnitude than the prediction noise ϵ_p. Hence, for equivalent prediction error ϵ_p, a loss ϕ with inverse link $[f_\phi^*]^{-1}$ of smaller growth rate $|\{[f_\phi^*]^{-1}\}'(v)|$ produces more accurate probability estimates. Figure 1 (right) shows the growth rate of the inverse link of the logistic loss. When the growth rate is smaller than one, the loss acts as a regularizer of probability estimates. From (13), this regularization multiplies any decrease of prediction error obtained by parameter regularization. This motivates the following definition.

Definition 1 *Let $\phi(v)$ be a proper margin loss. Then*

$$\rho_\phi(v) = \frac{1}{|\{[f_\phi^*]^{-1}\}'(v)|} \tag{14}$$

is the regularization strength of $\phi(v)$. If $\rho_\phi(v) \geq 1, \forall v$, then $\phi(v)$ is denoted a regularization loss.

1. When the optimal link is unique, the loss is denoted strictly proper. Because this is the case for all losses considered in this work, we simply refer to the loss as proper.

3.2 Generalization

An alternative way to characterize the interaction of loss-based and parameter-based regularization is to investigate how the two impact classifier generalization. This can be done by characterizing the dependence of classification error bounds on the two forms of regularization. Since, in this work, we will emphasize boosting algorithms, we rely on the following well known boosting bound.

Theorem 1 *(Schapire et al., 1998) Consider a sample S of m examples $\{(\mathbf{x}_1, y_i), \ldots, (\mathbf{x}_m, y_m)\}$ and a predictor $\hat{p}^*(\mathbf{x}; \mathbf{w})$ of the form of (5) where the $h_i(x)$ are in a space $\mathcal{H}$ of base classifiers of VC-dimension d. Then, with probability at least $1 - \delta$ over the choice of S, for all $\theta > 0$,*

$$P_{\mathbf{X},Y}[yp(\mathbf{x}; \mathbf{w}) \leq 0] \leq P_S \left[\frac{y\hat{p}^*(\mathbf{x}; \mathbf{w})}{||\mathbf{w}||_1} \leq \theta \right] + O\left(\frac{1}{\sqrt{m}} \sqrt{\frac{d \log^2(m/d)}{\theta^2} + \log(1/\delta)} \right),$$

where P_S denotes an empirical probability over the sample S.

Given $\mathcal{H}, m, d$ and δ, the two terms of the bound are functions of θ. The first term depends on the distribution of the margins $y_i\hat{p}^*(\mathbf{x}_i; \mathbf{w})$ over the sample. Assume, for simplicity, that S is separable by $\hat{p}^*(\mathbf{x}; \mathbf{w})$, i.e. $y_i\hat{p}^*(\mathbf{x}_i; \mathbf{w}) > 0, \forall i$, and denote the empirical margin by

$$\gamma_s = y_{i^*}\hat{p}^*(\mathbf{x}_{i^*}; \mathbf{w}), \qquad i^* = \arg\min_i y_i\hat{p}^*(\mathbf{x}_i; \mathbf{w}). \tag{15}$$

Then, for any $\epsilon > 0$ and $\theta = \gamma_s/||\mathbf{w}||_1 - \epsilon$, the empirical probability is zero and

$$P_{\mathbf{X},Y}[yp(\mathbf{x}; \mathbf{w}) \leq 0] \leq O\left(\frac{1}{\sqrt{m}} \sqrt{\frac{d \log^2(m/d)}{(\frac{\gamma_s}{||\mathbf{w}||_1} - \epsilon)^2} + \log(1/\delta)} \right).$$

Using (11) and a first order Taylor series expansion of $[f_\phi^*]^{-1}(.)$ around the origin

$$\begin{aligned} \hat{\eta}(\mathbf{x}_{i^*}) &= [f_\phi^*]^{-1}(y_{i^*}\gamma_s) \\ &\approx [f_\phi^*]^{-1}(0) + y_{i^*}\gamma_s\{[f_\phi^*]^{-1}\}'(0) \end{aligned}$$

it follows that

$$\gamma_s \approx \rho_\phi(0)|\hat{\eta}(\mathbf{x}_{i^*}) - 1/2|, \tag{16}$$

and the bound can be approximated by

$$P_{\mathbf{X},Y}[yp(\mathbf{x}; \mathbf{w}) \leq 0] \leq O\left(\frac{1}{\sqrt{m}} \sqrt{\frac{d \log^2(m/d)}{\left(\frac{\rho_\phi(0)}{||\mathbf{w}||_1}|\hat{\eta}(\mathbf{x}_{i^*}) - 1/2| - \epsilon \right)^2} + \log(1/\delta)} \right). \tag{17}$$

Since this is a monotonically decreasing function of the generalization factor

$$\kappa = \frac{\rho_\phi(0)}{||\mathbf{w}||_1}, \tag{18}$$

larger κ lead to tighter bounds on the probability of classification error, i.e. classifiers with stronger generalization guarantees. This confirms the complimentary nature of parameter and probability regularization, discussed in the previous section. Parameter regularization, as in (10), encourages solutions of smaller $||\mathbf{w}||_1$ and thus larger κ. Regularization losses multiply this effect by the regularization strength $\rho_\phi(0)$. This is in agreement with the multiplicative form of (13). In summary, for regularization losses, the generalization guarantees of classical parameter regularization are amplified by the strength of the probability regularization at the classification boundary.

4. Controlling the Regularization Strength of Proper Losses

In the remainder of this work, we study the design of regularization losses. In particular, we study how to control the regularization strength of a proper loss, by manipulating some loss parameter.

4.1 Proper Losses

The structure of proper losses can be studied by relating conditional risk minimization to the classical problem of probability elicitation in statistics (Savage, 1971; DeGroot and Fienberg, 1983). Here, the goal is to find the probability estimator $\hat{\eta}$ that maximizes the expected score

$$I(\eta, \hat{\eta}) = \eta I_1(\hat{\eta}) + (1 - \eta) I_{-1}(\hat{\eta}), \tag{19}$$

of a scoring rule that assigns to prediction $\hat{\eta}$ a score $I_1(\hat{\eta})$ when event $y = 1$ holds and a score $I_{-1}(\hat{\eta})$ when $y = -1$ holds. The scoring rule is proper if its components $I_1(\cdot), I_{-1}(\cdot)$ are such that the expected score is maximal when $\hat{\eta} = \eta$, i.e.

$$I(\eta, \hat{\eta}) \leq I(\eta, \eta) = J(\eta), \ \ \forall \eta \tag{20}$$

with equality if and only if $\hat{\eta} = \eta$. A set of conditions under which this holds is as follows.

Theorem 2 *(Savage, 1971) Let $I(\eta, \hat{\eta})$ be as defined in (19) and $J(\eta) = I(\eta, \eta)$. Then (20) holds if and only if $J(\eta)$ is convex and*

$$I_1(\eta) = J(\eta) + (1 - \eta) J'(\eta) \qquad I_{-1}(\eta) = J(\eta) - \eta J'(\eta). \tag{21}$$

Several works investigated the connections between probability elicitation and risk minimization (Buja et al., 2006; Masnadi-Shirazi and Vasconcelos, 2008; Reid and Williamson, 2010). We will make extensive use of the following result.

Theorem 3 *(Masnadi-Shirazi and Vasconcelos, 2008) Let $I_1(\cdot)$ and $I_{-1}(\cdot)$ be as in (21), for any continuously differentiable convex $J(\eta)$ such that $J(\eta) = J(1 - \eta)$, and $f(\eta)$ any invertible function such that $f^{-1}(-v) = 1 - f^{-1}(v)$. Then*

$$I_1(\eta) = -\phi(f(\eta)) \qquad I_{-1}(\eta) = -\phi(-f(\eta))$$

if and only if

$$\phi(v) = -J\left(f^{-1}(v)\right) - (1 - f^{-1}(v)) J'\left(f^{-1}(v)\right).$$

It has been shown that, for $C_\phi(\eta, p)$, $f^*_\phi(\eta)$, and $C^*_\phi(\eta)$ as in (7)-(9), $C^*_\phi(\eta)$ is concave (Zhang, 2004) and

$$C^*_\phi(\eta) = C^*_\phi(1-\eta) \quad (22)$$

$$[f^*_\phi]^{-1}(-v) = 1 - [f^*_\phi]^{-1}(v). \quad (23)$$

Hence, the conditions of the theorem are satisfied by any continuously differentiable $J(\eta) = -C^*_\phi(\eta)$ and invertible $f(\eta) = f^*_\phi(\eta)$. It follows that, $I(\eta, \hat{\eta}) = -C_\phi(\eta, f)$ is the expected score of a proper scoring rule if and only if the loss has the form

$$\phi(v) = C^*_\phi\left([f^*_\phi]^{-1}(v)\right) + (1 - [f^*_\phi]^{-1}(v))[C^*_\phi]'\left([f^*_\phi]^{-1}(v)\right). \quad (24)$$

In this case, the predictor of minimum risk is $p^* = f^*_\phi(\eta)$, and posterior probabilities can be recovered with (11). Hence, the loss ϕ is proper and the predictor p^* calibrated. In summary, proper losses have the structure of (22)-(24). In this work, we also assume that $C^*_\phi(0) = C^*_\phi(1) = 0$. This guarantees that the minimum risk is zero when there is absolute certainty about the class label Y, i.e. $P_{Y|\mathbf{X}}(1|\mathbf{x}) = 0$ or $P_{Y|\mathbf{X}}(1|\mathbf{x}) = 1$.

4.2 Loss Margin and Regularization Strength

The facts that 1) the empirical margin γ_s of (15) is a function of the loss margin μ_ϕ of Figure 1, and 2) the regularization strength ρ_ϕ is related to γ_s by (16), suggests that μ_ϕ is a natural loss parameter to control ρ_ϕ. A technical difficulty is that a universal definition of μ_ϕ is not obvious, since most margin losses $\phi(v)$ only converge to zero as $v \to \infty$. Although approximately zero for large positive v, they are strictly positive for all finite v. This is, for example, the case of the logistic loss $\phi(v) = \log(1 + e^{-v})$ of Figure 1 and the boosting loss of Table 1. To avoid this problem, we use a definition based on the second-order Taylor series expansion of ϕ around the origin. The construct is illustrated in Figure 2, where the loss margin μ_ϕ is defined by the point where the quadratic expansion reaches its minimum. It can be easily shown that this is the point $v = \mu_\phi$, where

$$\mu_\phi = -\frac{\phi'(0)}{\phi''(0)}. \quad (25)$$

In Appendix A, we show that, under mild conditions (see Lemma 9) on the inverse link $[f^*_\phi]^{-1}(\eta)$ of a twice differentiable loss ϕ

$$\mu_\phi = \frac{\rho_\phi(0)}{2}, \quad (26)$$

and the regularization strength of ϕ is lower bounded by twice the loss margin

$$\rho_\phi(v) \geq 2\mu_\phi. \quad (27)$$

Under these conditions, $\phi(v)$ is a regularization loss if and only if $\mu_\phi \geq \frac{1}{2}$. This establishes a direct connection between margins and probability regularization: larger loss margins produce more strongly regularized probability estimates. Hence, for proper losses of suitable link, the large margin strategy for classifier learning is also a strategy for regularization of probability estimates. In fact, from (26) and (18), the generalization factor of these losses is directly determined by the loss margin, since $\kappa = \frac{2\mu_\phi}{||\mathbf{w}||_1}$.

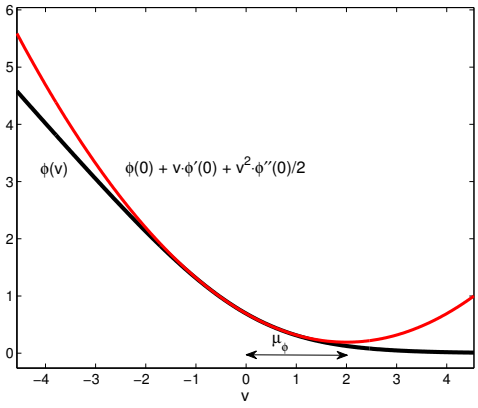


Figure 2: Definition of the loss margin μ_ϕ of a loss ϕ.

4.3 The Generalized Logit Link

As shown in Lemma 9 of Appendix A, the conditions that must be satisfied by the inverse link for (26) and (27) to hold (monotonically increasing, maximum derivative at the origin) are fairly mild. For example, they hold for the scaled logit

$$\gamma(\eta; a) = a \log \frac{\eta}{1-\eta} \qquad \gamma^{-1}(v; a) = \frac{e^{v/a}}{1+e^{v/a}}, \tag{28}$$

which, as shown in Table 1, is the optimal link of the exponential loss when $a = 1/2$ and of the logistic loss when $a = 1$. Since the exponential loss of boosting has margin $\mu_\phi = 1$ and the logistic loss $\mu_\phi = 2$, it follows from the lemma that these are regularization losses. However, the conditions of the lemma hold for many other link functions. In this work, we consider a broad family of such functions, which we denote as the generalized logit.

Definition 2 *An invertible transformation $\pi(\eta)$ is a generalized logit if its inverse, $\pi^{-1}(v)$, has the following properties*

1. *$\pi^{-1}(v)$ is monotonically increasing,*
2. *$\lim_{v\to\infty} \pi^{-1}(v) = 1$*
3. *$\pi^{-1}(-v) = 1 - \pi^{-1}(v)$,*
4. *for finite v, $(\pi^{-1})^{(2)}(v) = 0$ if and only if $v = 0$,*

where $\pi^{(n)}$ is the n^{th} order derivative of π.

In Appendix B, we discuss some properties of the generalized logit and show that all conditions of Lemma 9 hold when $f^*_\phi(\eta)$ is in this family of functions. When combined with Lemma 9, this proves the following result.

Theorem 4 *Let $\phi(v)$ be a twice differentiable proper loss of generalized logit link $f^*_\phi(\eta)$. Then*

$$\mu_\phi = \frac{\rho_\phi(0)}{2} \tag{29}$$

and the regularization strength of $\phi(v)$ is lower bounded by twice the loss margin $\rho_\phi(v) \geq 2\mu_\phi$. $\phi(v)$ is a regularization loss if and only if $\mu_\phi \geq \frac{1}{2}$.

5. Controlling the Regularization Strength

The results above show that it is possible to control the regularization strength of a proper loss of generalized logit link by manipulating the loss margin μ_ϕ. In this section we derive procedures to accomplish this.

5.1 Tunable Regularization Losses

We start by studying the set of proper margin losses whose regularization is controlled by a parameter $\sigma > 0$. These are denoted tunable regularization losses.

Definition 3 *Let $\phi(v)$ be a proper loss of generalized logit link $f^*_\phi(\eta)$. A parametric loss*

$$\phi_\sigma(v) = \phi(v;\sigma) \quad \textit{such that} \quad \phi(v;1) = \phi(v)$$

is the tunable regularization loss generated by $\phi(v)$ if $\phi_\sigma(v)$ is a proper loss of generalized logit link and

$$\mu_{\phi_\sigma} = \sigma\mu_\phi,$$

for all σ such that

$$\sigma \geq \frac{1}{2\mu_\phi}. \tag{30}$$

The parameter σ is the gain of the tunable regularization loss $\phi_\sigma(v)$.

Since, from (29) and (14), the loss margin μ_ϕ only depends on the derivative of the inverse link at the origin, a tunable regularization loss can be generated from any proper loss of generalized logit link, by simple application of Theorem 3.

Lemma 4 *Let $\phi(v)$ be a proper loss of generalized logit link $f^*_\phi(\eta)$. The parametric loss*

$$\phi_\sigma(v) = C^*_{\phi_\sigma}\{[f^*_{\phi_\sigma}]^{-1}(v)\} + (1 - [f^*_{\phi_\sigma}]^{-1}(v))[C^*_{\phi_\sigma}]'([f^*_{\phi_\sigma}]^{-1}(v)), \tag{31}$$

where

$$f^*_{\phi_\sigma}(\eta) \;=\; \sigma f^*_\phi(\eta) \tag{32}$$

$C^*_{\phi_\sigma}(\eta)$ is a minimum risk function (i.e. a continuously differentiable concave function with symmetry $[C^*_{\phi_\sigma}](1-\eta) = [C^*_{\phi_\sigma}](\eta)$) such that $C^*_{\phi_\sigma}(0) = 0$, and (30) holds is a tunable regularization loss.

Proof From (32)

$$[f^*_{\phi_\sigma}]^{-1}(v) = [f^*_\phi]^{-1}\left(\frac{v}{\sigma}\right). \tag{33}$$

Since $[f^*_\phi]^{-1}(v)$ is a generalized logit link it has the properties of Definition 2. Since these properties continue to hold when v is replaced by v/σ, it follows that $f^*_{\phi_\sigma}(v)$ is a generalized logit link. It follows from (31) that $\phi_\sigma(v)$ satisfies the conditions of Theorem 3 and is a proper loss. Since $\mu_{\phi_\sigma} = \frac{\rho_{\phi_\sigma}(0)}{2} = \frac{1}{2\{[f^*_{\phi_\sigma}]^{-1}\}'(0)} = \sigma\mu_\phi$, the parametric loss $\phi_\sigma(v)$ is a tunable regularization loss. ■

In summary, it is possible to generate a tunable regularization loss by simply rescaling the link of a proper loss. Interestingly, this holds independently of how σ parameterizes the minimum risk $[C^*_{\phi_\sigma}](\eta)$. However, not all such losses are useful. If, for example, the process results in

$$\phi_\sigma(v) = \phi(v/\sigma),$$

it corresponds to a simple rescaling of the horizontal axis of Figure 1. The loss $\phi_\sigma(v)$ is thus not fundamentally different from $\phi(v)$. Using this loss in a learning algorithm is equivalent to varying the margin by rescaling the feature space $\mathcal{X}$.

5.2 The Binding Function

To produce non-trivial tunable regularization losses $\phi_\sigma(v)$, we need a better understanding of the role of the minimum risk $[C^*_{\phi_\sigma}](\eta)$. This is determined by the binding function of the loss.

Definition 5 *Let $\phi(v)$ be a proper loss of link $f^*_\phi(\eta)$, and minimum risk $C^*_\phi(\eta)$. The function*

$$\beta_\phi(v) = [C^*_\phi]'\left([f^*_\phi]^{-1}(v)\right) \tag{34}$$

is denoted the binding function of ϕ.

The properties of the binding function are discussed in Appendix C and illustrated in Figure 3. For proper losses of generalized logit link, $\beta_\phi(v)$ is a monotonically decreasing odd function, which determines the behavior of $\phi(v)$ away from the origin and defines a one-to-one mapping between the link f^*_ϕ and the derivative of the risk C^*_ϕ. In this way, β_ϕ "binds" link and risk.

The following result shows that the combination of link and binding function determine the loss up to a constant.

Theorem 5 *Let $\phi(v)$ be a proper loss of generalized logit link $f^*_\phi(\eta)$ and binding function $\beta_\phi(v)$. Then*

$$\phi'(v) \;=\; (1 - [f^*_\phi]^{-1}(v))\beta'_\phi(v). \tag{35}$$

Proof From (24) and the definition of β_ϕ,

$$\phi(v) = C^*_\phi([f^*_\phi]^{-1}(v)) + (1 - [f^*_\phi]^{-1}(v))\beta_\phi(v). \tag{36}$$

Taking derivatives on both sides leads to (35). ■

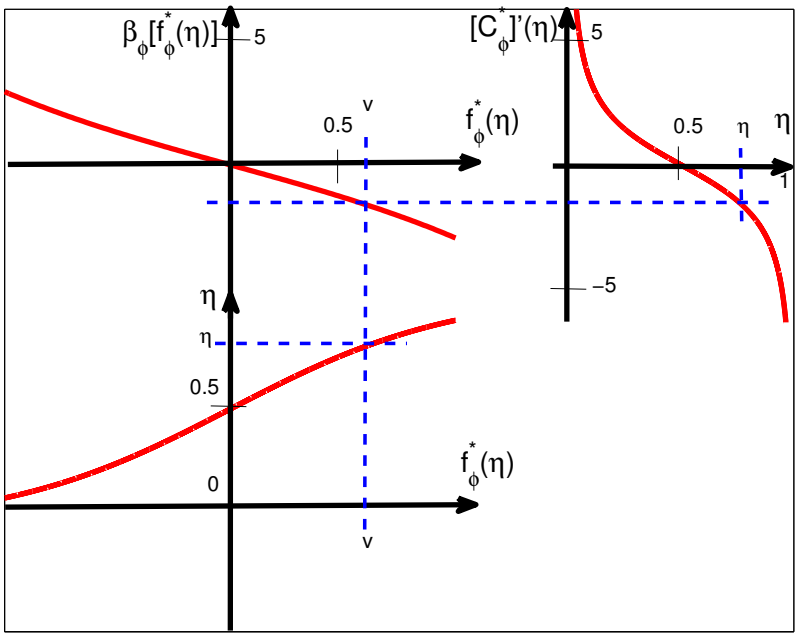


Figure 3: Link $f^*_\phi(\eta)$, risk derivative $[C^*_\phi]'(\eta)$, and binding function $\beta_\phi(f^*_\phi(\eta))$ of a proper loss $\phi(v)$ of generalized logit link.

This result enables the derivation of a number of properties of proper losses of generalized logit link. These are discussed in Appendix D.1, where such losses are shown to be monotonically decreasing, convex under certain conditions on the inverse link and binding function, and identical to the binding function for large negative margins. In summary, a proper loss of generalized logit link can be decomposed into two fundamental quantities: the inverse link, which determines its regularization strength, and the binding function, which determines its behavior away from the origin. Since tunable regularization losses are proper, the combination of this result with Lemma 4 and Definition 5 proves the following theorem.

Theorem 6 *Let $\phi(v)$ be a proper loss of generalized logit link $f^*_\phi(\eta)$. The parametric loss*

$$\phi'_\sigma(v) = (1 - [f^*_{\phi_\sigma}]^{-1}(v))\beta'_{\phi_\sigma}(v), \tag{37}$$

where

$$f^*_{\phi_\sigma}(\eta) \quad = \quad \sigma f^*_\phi(\eta), \tag{38}$$

$\beta_{\phi_\sigma}(v)$ is a binding function (i.e. a continuously differentiable, monotonically decreasing, odd function), and σ is such that (30) holds is a tunable regularization loss.

Algorithm 1: BoostLR

Input: Training set $\mathcal{D} = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\}$, where $y_i \in \{1, -1\}$ is the class label of example $\mathbf{x}$, regularization gain σ, and number T of weak learners in the final decision rule.

Initialization: Set $G^{(0)}(\mathbf{x}_i) = 0$ and $w^{(1)}(\mathbf{x}_i) = -\left(1 - [f^*_{\phi_\sigma}]^{-1}(y_i G^{(0)}(\mathbf{x}_i))\right) \beta'_{\phi_\sigma}\left(y_i G^{(0)}(\mathbf{x}_i)\right) \quad \forall \mathbf{x}_i$.

for $t = \{1, \ldots, T\}$ **do**

choose weak learner

$$g^*(\mathbf{x}) = \arg\max_{g(\mathbf{x})} \sum_{i=1}^{n} y_i w^{(t)}(\mathbf{x}_i) g(\mathbf{x}_i)$$

update predictor G($\mathbf{x}$)

$$G^{(t)}(\mathbf{x}) = G^{(t-1)}(\mathbf{x}) + g^*(\mathbf{x})$$

update weights

$$w^{(t+1)}(\mathbf{x}_i) = -\left(1 - [f^*_{\phi_\sigma}]^{-1}(y_i G^{(t)}(\mathbf{x}_i))\right) \beta'_{\phi_\sigma}\left(y_i G^{(t)}(\mathbf{x}_i)\right) \quad \forall \mathbf{x}_i$$

end for

Output: decision rule $h(\mathbf{x}) = \text{sgn}[G^{(T)}(\mathbf{x})]$.

5.3 Boosting With Tunable Probability Regularization

Given a tunable regularization loss ϕ_σ, various algorithms can be used to design a classifier. Boosting accomplishes this by gradient descent in a space $\mathcal{W}$ of weak learners. While there are many variants, in this work we adopt the GradientBoost framework (Friedman, 2001). This searches for the predictor $G(\mathbf{x})$ of minimum empirical risk on a sample $\mathcal{D} = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\}$,

$$R(G) = \sum_{i=1}^{n} \phi_\sigma(y_i G(\mathbf{x}_i)).$$

At iteration t, the predictor is updated according to

$$G^{(t)}(\mathbf{x}) = G^{(t-1)}(\mathbf{x}) + g^{(t)}(\mathbf{x}), \tag{39}$$

where $g^{(t)}(\mathbf{x})$ is the gradient of $R(G)$ in $\mathcal{W}$, i.e. the weak learner

$$\begin{aligned} g^{(t)}(\mathbf{x}) &= \arg\max_{g} \sum_{i=1}^{n} -y_i \phi'_\sigma(y_i G^{(t-1)}(\mathbf{x}_i)) g(\mathbf{x}_i) \\ &= \arg\max_{g} \sum_{i=1}^{n} y_i w^{(t)}_\sigma(\mathbf{x}_i) g(\mathbf{x}_i), \end{aligned}$$

where

$$w^{(t)}_\sigma(\mathbf{x}_i) = -\phi'_\sigma(y_i G^{(t-1)}(\mathbf{x}_i))$$

is the weight of example $\mathbf{x}_i$ at iteration t. For a tunable regularization loss $\phi_\sigma(v)$ of generalized logit link $f^*_{\phi_\sigma}(\eta)$ and binding function $\beta_{\phi_\sigma}(v)$, it follows from (37) that

$$w^{(t)}_\sigma(\mathbf{x}_i) = -\left(1 - [f^*_{\phi_\sigma}]^{-1}\left(y_i G^{(t-1)}(\mathbf{x}_i)\right)\right)\beta'_{\phi_\sigma}\left(y_i G^{(t-1)}(\mathbf{x}_i)\right). \tag{40}$$

Boosting with these weights is denoted boosting with loss regularization (BoostLR) and summarized in Algorithm 1.

The weighting mechanism of BoostLR provides some insight on how the choices of link and binding function affect classifier behavior. Using $\gamma_i = y_i G^{(t-1)}(\mathbf{x}_i)$ to denote the margin of $\mathbf{x}_i$ for the classifier of iteration $t-1$,

$$w^{(t)}_\sigma(\mathbf{x}_i) = -\phi'_\sigma(\gamma_i) = -\left(1 - [f^*_{\phi_\sigma}]^{-1}(\gamma_i)\right)\beta'_{\phi_\sigma}(\gamma_i). \tag{41}$$

It follows from the discussion of the previous section that 1) the link $f^*_{\phi_\sigma}$ is responsible for the behavior of the weights around the classification boundary and 2) the binding function β_{ϕ_σ} for the behavior at large margins. For example, applying (34) to the links and risks of Table 1 results in

$$\beta(v) = e^{-v} - e^v \qquad \beta'(v) = -e^{-v} - e^v \tag{42}$$

for AdaBoost and

$$\beta(v) = -v \qquad \beta'(v) = -1 \tag{43}$$

for LogitBoost. In result, AdaBoost weights are exponentially large for examples of large negative margin γ_i, while LogitBoost weights remain constant. This fact has been used to explain the much larger sensitivity of AdaBoost to outliers (Maclin and Opitz, 1997; Dietterich, 2000; Mason et al., 2000; Masnadi-Shirazi and Vasconcelos, 2008; Friedman et al., 2000; McDonald et al., 2003; Leistner et al., 2009). Under this view, the robustness of a boosting algorithm to outliers is determined by its binding function. Hence, the decomposition of a loss into link and binding functions translates into a functional decomposition for boosting algorithms. It decouples the generalization ability of the learned classifier, determined by the regularization strength imposed by the link, from its robustness to outliers, determined by the binding function.

6. The Set of Tunable Regularization Losses

The link-binding decomposition can also be used to characterize the structure of the set of tunable regularization losses.

6.1 Equivalence Classes

A simple consequence of (37) is that the set $\mathcal{R}$ of tunable regularization losses ϕ_σ is the Cartesian product of the set $\mathcal{L}$ of generalized logit links and the set $\mathcal{B}$ of binding functions. It follows that both generalized logit links f_σ and binding functions β_σ define equivalence classes in $\mathcal{R}$. In fact, $\mathcal{R}$ can be partitioned according to

$$\mathcal{R} = \cup_{\beta_\sigma}\mathcal{R}_{\beta_\sigma} \quad \text{where} \quad \mathcal{R}_{\beta_\sigma} = \{\phi_\sigma | \beta_{\phi_\sigma} = \beta_\sigma\}$$

or

$$\mathcal{R} = \cup_{f_\sigma}\mathcal{R}_{f_\sigma} \quad \text{where} \quad \mathcal{R}_{f_\sigma} = \{\phi_\sigma | f^*_{\phi_\sigma} = f_\sigma\}.$$

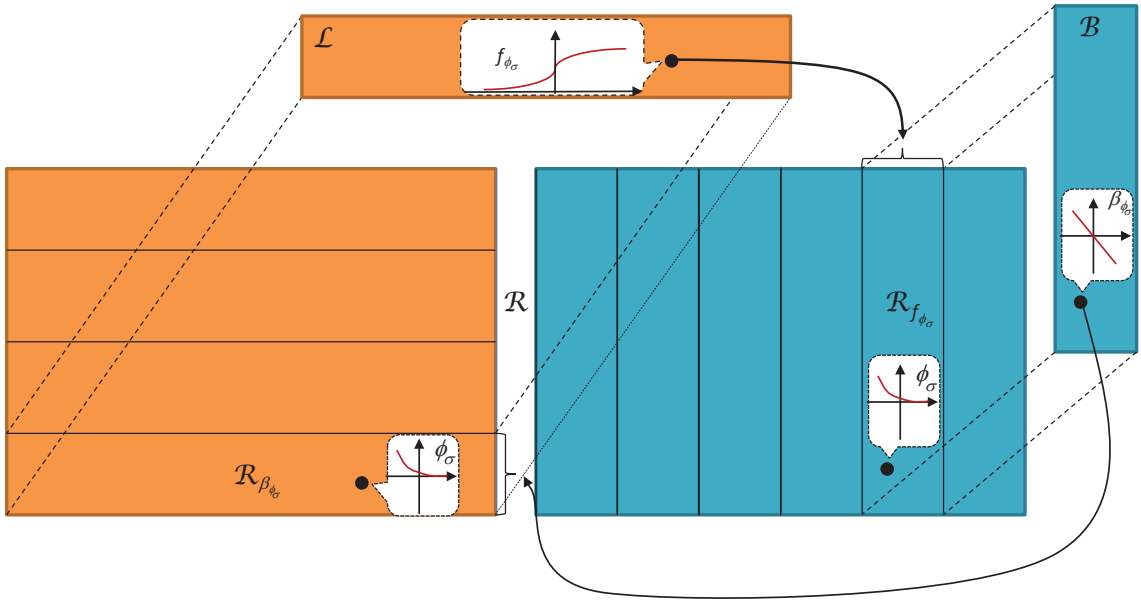


Figure 4: The set $\mathcal{R}$ of tunable regularization losses can be partitioned into equivalence classes $\mathcal{R}_{f_{\phi\sigma}}$, isometric to the set $\mathcal{B}$ of binding functions, or equivalence classes $\mathcal{R}_{\beta_{\phi\sigma}}$, isometric to the set $\mathcal{L}$ of generalized logit links. A tunable regularization loss ϕ_σ is defined by a pair of link f_{ϕ_σ} and binding β_{ϕ_σ} functions.

The sets $\mathcal{R}_{f_\sigma}$ are isomorphic to $\mathcal{B}$, which is itself isomorphic to the set of continuously differentiable, monotonically decreasing, odd functions. The sets $\mathcal{R}_{\beta_\sigma}$ are isomorphic to $\mathcal{L}$, which is shown to be isomorphic, in Appendix B.2, to the set of parametric continuous scale probability density functions (pdfs)

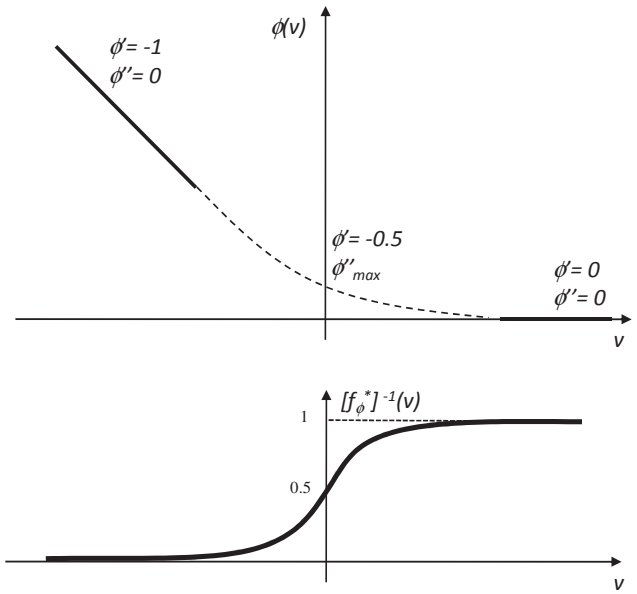
$$\psi_\sigma(v) = \frac{1}{\sigma}\psi\left(\frac{v}{\sigma}\right), \tag{44}$$

where $\psi(v)$ has unit scale, a unique maximum at the origin, and $\psi(-v) = \psi(v)$. The structure of the set of tunable regularization losses is illustrated in Figure 4. The set can be partitioned in two ways. The first is into a set of equivalence classes $\mathcal{R}_{\beta_\sigma}$ isomorphic to the set of pdfs of (44). The second into a set of equivalence classes $\mathcal{R}_{f_\sigma}$ isomorphic to the set of monotonically decreasing odd functions.

6.2 Design of Regularization Losses

An immediate consequence of the structure of $\mathcal{R}$ is that all tunable regularization losses can be designed by the following procedure.

1. select a scale pdf $\psi_\sigma(v)$ with the properties of (44).
2. set $[f^*_{\phi_\sigma}]^{-1}(v) = c_\sigma(v)$, where $c_\sigma(v) = \int_{-\infty}^{v} \psi_\sigma(q)dq$ is the cumulative distribution function (cdf) of $\psi_\sigma(v)$.



Given ψ_σ or c_σ
$\phi_\sigma = \int [c_\sigma(v) - 1] dv$
$[f^*_{\phi_\sigma}]^{-1}(v) = c_\sigma(v)$
$C^*_{\phi_\sigma} = -\int_0^\eta c_\sigma^{-1}(\nu) d\nu$

Given ϕ_σ, $f^*_{\phi_\sigma}$ or $C^*_{\phi_\sigma}$
$\psi_\sigma(v) = \phi''_\sigma(v)$
$\psi_\sigma(v) = \{[f^*_{\phi_\sigma}]^{-1}\}'(v)$
$c_\sigma(v) = [-\{C^*_{\phi_\sigma}\}']^{-1}(v)$

Figure 5: Canonical regularization losses. Left: general properties of the loss and inverse link functions. Right: Relations between losses and scale pdfs.

3. select a binding function $\beta_{\phi_\sigma}(v)$. This can be any parametric family of continuously differentiable, monotonically decreasing, odd functions.

4. define the tunable regularization loss as $\phi'_\sigma(v) = (1 - [f^*_{\phi_\sigma}]^{-1}(v))\beta'_{\phi_\sigma}(v)$.

5. restrict σ according to (30).

Note that the derivative $\phi'_\sigma(v)$ is sufficient to implement the BoostLR algorithm. If desired, it can be integrated to produce a formula for the loss $\phi_\sigma(v)$. This defines the loss up to a constant, which can be determined by imposing the constraint that $\lim_{v\to\infty} \phi_\sigma(v) = 0$. As discussed in the previous section, this procedure enables the independent control of the regularization strength and robustness of the losses $\phi_\sigma(v)$. In fact, it follows from step 2. and (14) that

$$\rho_{\phi_\sigma}(v) = \frac{1}{\psi_\sigma(v)}, \tag{45}$$

i.e. the choice of pdf $\psi_\sigma(v)$ determines the regularization strength of $\phi_\sigma(v)$. The choice of binding function in step 3. then limits $\phi_\sigma(v)$ to an equivalence class $\mathcal{R}_{\beta_\sigma}$ of regularization losses with common robustness properties. We next consider some important equivalence classes.

6.3 Canonical Regularization Losses

We start by considering the set of tunable regularization losses with linear binding function

$$\beta_{\phi_\sigma}(v) = -v. \tag{46}$$

	Generalized Logistic (GLog)	Generalized Gaussian (GGauss)
$\psi_\sigma(v)$	$\frac{e^{\frac{v}{\sigma}}}{\sigma(1+e^{\frac{v}{\sigma}})^2}$	$\frac{1}{4\sigma}e^{-(\frac{\sqrt{\pi}}{4\sigma}v)^2}$
$c_\sigma(v)$	$\frac{e^{v/\sigma}}{1+e^{v/\sigma}}$	$\frac{1}{2}\left[1+erf\left(\frac{\sqrt{\pi}}{4\sigma}v\right)\right]$
$\phi_\sigma(v)$	$\sigma\log\left(1+e^{-\frac{v}{\sigma}}\right)$	$\frac{v}{2}\left[erf\left(\frac{\sqrt{\pi}}{4\sigma}v\right)-1\right]+\frac{2\sigma}{\pi}e^{-\left(\frac{\sqrt{\pi}}{4\sigma}v\right)^2}$
$f^*_{\phi_\sigma}(\eta)$	$\sigma\log\frac{\eta}{1-\eta}$	$\frac{4\sigma}{\sqrt{\pi}}\cdot erf^{-1}(2\eta-1)$
$C^*_{\phi_\sigma}(\eta)$	$-\sigma\eta\log(\eta)-\sigma(1-\eta)\log(1-\eta)$	$-\frac{4\sigma}{\sqrt{\pi}}\int erf^{-1}(2\eta-1)d\eta$
$\rho_\phi(v)$	$\frac{\sigma(1+e^{\frac{v}{\sigma}})^2}{e^{\frac{v}{\sigma}}}$	$4\sigma e^{(\frac{\sqrt{\pi}}{4\sigma}v)^2}$
	Generalized Laplacian (GLaplacian)	**Generalized Boosting (GBoost)**
$\psi_\sigma(v)$	$\frac{1}{4\sigma}e^{-\frac{\lvert v\rvert}{2\sigma}}$	$\frac{2}{\sigma\left(4+(\frac{v}{\sigma})^2\right)^{\frac{3}{2}}}$
$c_\sigma(v)$	$\frac{1}{2}\left[1+sign(v)\left(1-e^{-\frac{\lvert v\rvert}{2\sigma}}\right)\right]$	$\frac{1}{2}+\frac{\frac{v}{\sigma}}{2\sqrt{4+\left(\frac{v}{\sigma}\right)^2}}$
$\phi_\sigma(v)$	$\sigma e^{\frac{-\lvert v\rvert}{2\sigma}}+\frac{1}{2}(\lvert v\rvert-v)$	$\frac{\sigma}{2}\left(\sqrt{4+\left(\frac{v}{\sigma}\right)^2}-\frac{v}{\sigma}\right)$
$f^*_{\phi_\sigma}(\eta)$	$-2\sigma sign(2\eta-1)\log(1-\lvert 2\eta-1\rvert)$	$\sigma\frac{2\eta-1}{\sqrt{\eta(1-\eta)}}$
$C^*_{\phi_\sigma}(\eta)$	$\sigma(1-\lvert 2\eta-1\rvert)[1-\log(1-\lvert 2\eta-1\rvert)]$	$2\sigma\sqrt{\eta(1-\eta)}$
$\rho_\phi(v)$	$4\sigma e^{\frac{\lvert v\rvert}{2\sigma}}$	$\frac{\sigma}{2}\left(4+(\frac{v}{\sigma})^2\right)^{\frac{3}{2}}$

Table 2: Canonical tunable regularization losses

From (37), these losses are uniquely determined by their link function

$$\phi'_\sigma(v) = -(1-[f^*_{\phi_\sigma}]^{-1}(v)). \tag{47}$$

Their properties are discussed in Appendix D.2. As illustrated in Figure 5, they are convex, monotonically decreasing, linear (with slope -1) for large negative v, constant for large positive v, and have slope $-.5$ and maximum curvature at the origin. The only degrees of freedom are in the vicinity of the origin, and determine the loss margin, since $\mu_{\phi_\sigma} = \frac{1}{2\phi''_\sigma(0)}$. Furthermore, because these losses have regularization strength $\rho_{\phi_\sigma}(0) = \frac{1}{\phi''_\sigma(0)}$, they are direct regularizers of probability scores, and regularization losses whenever $\phi''_\sigma(0) \leq 1$. This is reminiscent of a well known result (Bartlett et al., 2006) that Bayes consistency holds for a convex $\phi(v)$ if and only if $\phi'(0) \leq 0$. From Property 4. of Lemma 13, this holds for all regularization losses with the form of (47). The constraint $\phi''_\sigma(0) \leq 1$ is also equivalent to $\frac{\phi''(0)}{\sigma} \leq 1$. This is the condition of (30) for the losses of (47).

When (46) holds, it follows from (34) that $f^*_\phi(\eta) = -[C^*_\phi]'(\eta)$. Buja et al. showed that the empirical risk of (4) is convex when ϕ is a proper loss and this relationship holds . They denoted as canonical risks the risks of (7) for which this is the case (Buja et al., 2006). For consistency, we denote the associated $\phi(v)$ a canonical loss. This is summarized by the following definition.

Definition 6 *A tunable regularization loss $\phi_\sigma(v)$ such that (47) holds for any σ such that $\phi''_\sigma(0) \leq 1$ is a canonical loss.*

We note, however, that what makes canonical losses special is not the guarantee of a convex risk, but that they have the simplest binding function with this guarantee. From Property 2. of Lemma 13, loss convexity does not require a linear binding function. On the other hand, since 1) any risk of convex loss is convex, 2) (57) holds for the linear binding function, and 3) binding functions are monotonically decreasing, the linear binding function is the simplest that guarantees a convex risk.

It should also be noted that the equivalence class of (46) includes many regularization losses. The relations of Figure 5, where $c_\sigma(v)$ is the cumulative distribution function (cdf) of the pdf $\psi_\sigma(v)$ of (44), can be used to derive losses from pdfs or pdfs from losses. Some example tunable canonical regularization losses are presented in Table 2. The generalized logistic (GLog), Gaussian (GGauss), and Laplacian (GLaplacian) losses are tunable losses derived from the logistic, Gaussian, and Laplace pdfs respectively. The GBoost loss illustrates some interesting alternative possibilities for this loss design procedure. In this case, we did not start from the pdf $\psi_\sigma(v)$ but from the minimum risk of boosting (see Table 1). We then used the top equations of Figure 5 to derive the cdf $c_\sigma(v)$ and the bottom equations to obtain $\phi_\sigma(v)$ and $f^*_{\phi_\sigma}(\eta)$. The resulting pdf $\psi_\sigma(v)$ is a special case of the Pearson type VII distribution with zero location parameter, shape parameter $\frac{3}{2}$ and scale parameter 2σ. These losses, their optimal inverse links, and regularization strength are plotted in Figure 6, which also shows how the regularization gain σ influences the loss around the origin, both in terms of its margin properties and regularization strength. Note that, due to (45), canonical losses implement all regularization behaviors possible for tunable regularization losses. This again justifies the denomination of "canonical regularization losses," although such an interpretation does not appear to have been intended by Buja et al.

The combination of BoostLR with a canonical loss is denoted a canonical BoostLR algorithm. For a proper loss ϕ_σ, $G^{(t)}(\mathbf{x})$ converges asymptotically to the optimal predictor $p^*_\sigma(\mathbf{x}) = f^*_{\phi_\sigma}(\eta(\mathbf{x}))$ and the weight function of (40) to

$$w^*(\mathbf{x}_i) \;=\; \begin{cases} 1-\eta(\mathbf{x}_i) & \text{if } y_i = 1 \\ \eta(\mathbf{x}_i) & \text{if } y_i = -1. \end{cases}$$

Hence, the weights of canonical BoostLR converge to the posterior example probabilities. Figure 7 shows the weight functions of the losses of Table 2. An increase in regularization gain σ simultaneously 1) extends the region of non-zero weight away from the boundary, and 2) reduces the derivative amplitude, increasing regularization strength. Hence, larger gains increase both the classification margin and the regularization of probability estimates.

6.4 Shrinkage Losses

Definition 7 *A tunable regularization loss $\phi_\sigma(v)$ such that*

$$\beta'_{\phi_\sigma}(v) = \beta'_\phi\left(\frac{v}{\sigma}\right), \tag{48}$$

for some $\beta_\phi(v) \in \mathcal{B}$ is a shrinkage loss.

Note that, since (48) holds for the linear binding function of (46), canonical regularization losses are shrinkage losses. These losses are easily identifiable, since combining (48), (37),

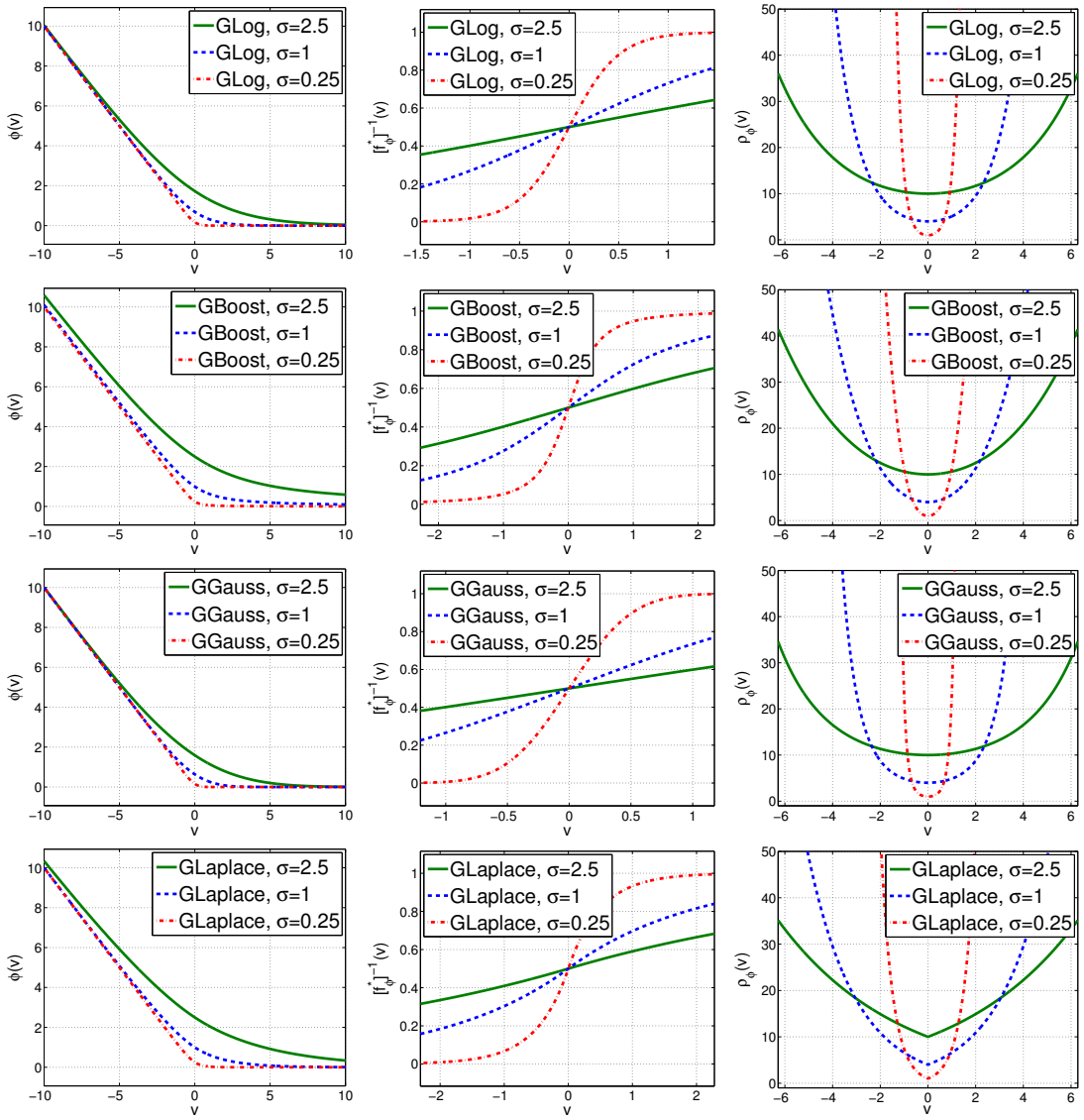


Figure 6: Loss (left), inverse link (middle), and regularization strength (right) functions, for various canonical regularization losses and gains σ. From top to bottom: GLog, GBoost, GGauss and GLaplacian.

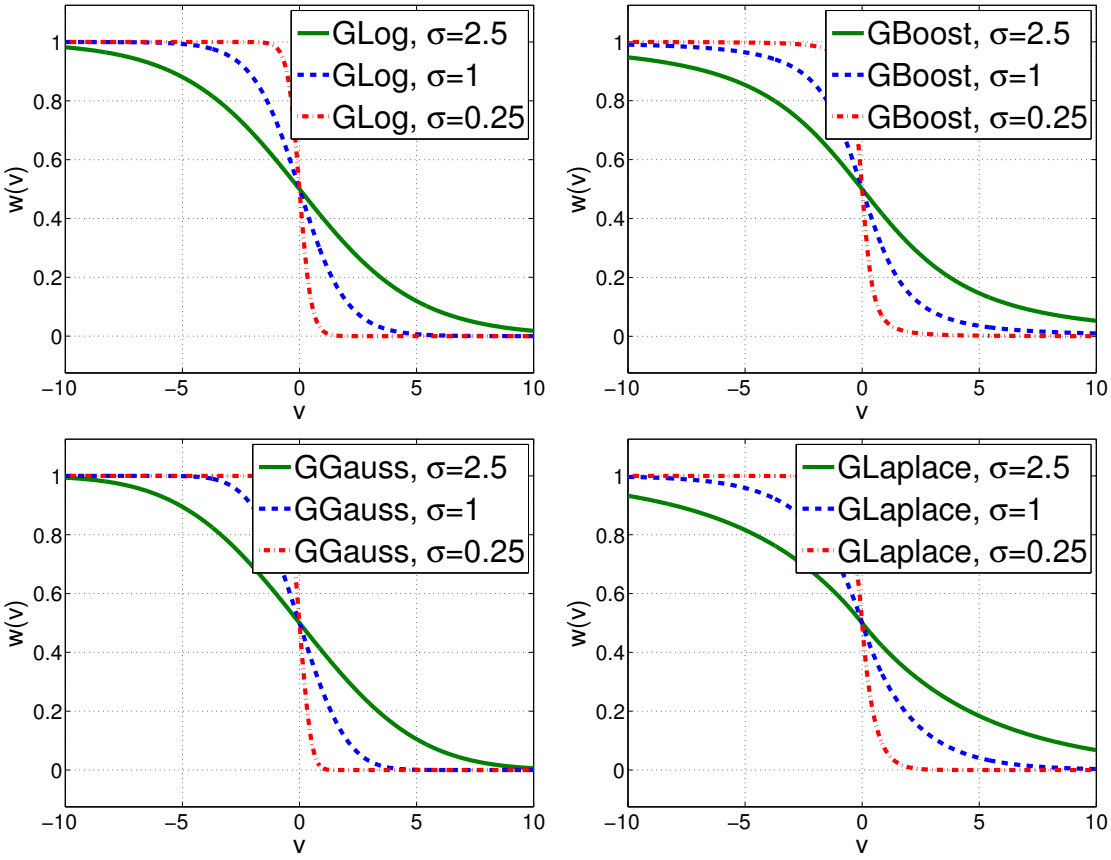


Figure 7: BoostLR weights for various parametric regularization losses and gains. GLog (top left), GBoost (top right), GGauss (bottom left) and GLaplace (bottom right).

and (33) leads to $\phi'_\sigma(v) = \phi'(v/\sigma)$. Hence, ϕ_σ is a shrinkage loss if and only if

$$\phi_\sigma(v) = \sigma\phi\left(\frac{v}{\sigma}\right). \tag{49}$$

This enables the generalization of any proper loss of generalized logit link into a shrinkage loss. For example, using Table 1, it is possible to derive the shrinkage losses generated by the logistic

$$\phi_\sigma(v) = \sigma \log(1 + e^{-\frac{v}{\sigma}})$$

and the exponential loss

$$\phi_\sigma(v) = \sigma e^{-\frac{v}{\sigma}}.$$

The former is the GLog loss of Table 2, but the later is not a canonical regularization loss.

Shrinkage losses also connect BoostLR to shrinkage, a popular regularization heuristic (Hastie et al., 2001). For GradientBoost, this consists of modifying the learning rule

of (39) into

$$G^{(t)}(\mathbf{x}) = G^{(t-1)}(\mathbf{x}) + \lambda g^{(t)}(\mathbf{x}), \tag{50}$$

where $0 < \lambda < 1$ is a learning rate. Shrinkage is inspired by parameter regularization methods from the least-squares regression literature, where similar modifications follow from the adoption of Bayesian models with priors that encourage sparse regression coefficients. This interpretation does not extend to classification, barring the assumption of the least-squares loss and some approximations (Hastie et al., 2001). In any case, it has been repeatedly shown that small learning rates ($\lambda \leq 0.1$) can significantly improve the generalization ability of the learned classifiers. Hence, despite its tenuous theoretical justification, shrinkage is a commonly used regularization procedure.

Shrinkage losses, and the proposed view of margin losses as regularizers of probability estimates, provide a much simpler and more principled justification for the shrinkage procedure. It suffices to note that the combination of (49) and (41) leads to

$$\begin{aligned} w_\sigma^{(t)}(\mathbf{x}_i) &= -\phi'_\sigma(\gamma_i) = -\phi'\left(\frac{\gamma_i}{\sigma}\right) \\ &= -\left(1 - [f^*_\phi]^{-1}\left(\frac{\gamma_i}{\sigma}\right)\right)\beta'_\phi\left(\frac{\gamma_i}{\sigma}\right), \end{aligned}$$

where $\gamma_i = y_i G^{(t-1)}(\mathbf{x}_i)$. Letting $\lambda = 1/\sigma$, this is equivalent to

$$w_\lambda(\mathbf{x}_i) = -\left(1 - [f^*_\phi]^{-1}\left(y_i \lambda G(\mathbf{x}_i)\right)\right)\beta'_\phi\left(y_i \lambda G(\mathbf{x}_i)\right).$$

Hence, the weight function of BoostLR with shrinkage loss ϕ_σ and predictor $G(\mathbf{x})$ is equivalent to the weight function of standard GradientBoost with loss ϕ and shrinked predictor $1/\sigma G(\mathbf{x})$. Since the only other effect of replacing (39) with (50) is to rescale the final predictor $G^{(T)}(\mathbf{x})$, the decision rule $h(\mathbf{x})$ produced by the two algorithms is identical. In summary, GradientBoost with shrinkage and a small learning rate λ is equivalent to BoostLR with a shrinkage loss of large regularization strength $(1/\lambda)$. This justifies the denomination of "shrinkage losses" for the class of regularization losses with the property of (48).

It should be noted, however, that while rescaling the predictor does not affect the decision rule, it affects the recovery of posterior probabilities from the shrinked predictor. The regularization view of shrinkage makes it clear that the probabilities can be recovered with

$$\hat{\eta}(\mathbf{x}) = [f^*_{\phi_\sigma}]^{-1}\left(G^{(T)}(\mathbf{x})\right) = [f^*_\phi]^{-1}\left(\lambda G^{(T)}(\mathbf{x})\right). \tag{51}$$

In the absence of this view, it is not obvious why shrinkage, which is justified as a simple change of learning rate, would require a modified link function for probability recovery. It is also neither clear nor it has been claimed that shrinkage would improve the quality of probability estimates. On the other hand, the discussion above suggests that this is why it works: shrinkage is a procedure for controlling probability regularization strength by manipulation of the loss margin. In fact, since GradientBoost with shrinkage and a small learning rate λ is equivalent to BoostLR with a shrinkage loss of large regularization strength $(1/\lambda)$, Section 3.2 provides a theoretical justification for the empirical evidence that shrinkage improves generalization performance.

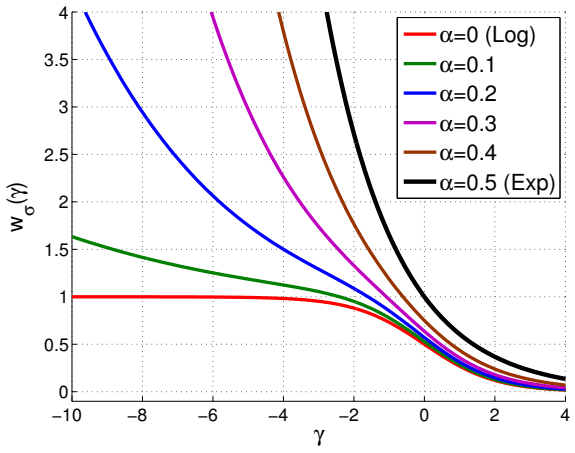


Figure 8: Weight function of the α-tunable regularization loss, for different values of α.

6.5 α-tunable Regularization Losses

From (48), the key to the equivalence between loss-based regularization and shrinkage is the identical parameterization of $[f^*_{\phi_\sigma}]^{-1}(v)$ and $\beta'_{\phi_\sigma}(v)$ in (33) and (48). When this is not the case, BoostLR weights are given by

$$\begin{aligned} w_\sigma(\mathbf{x}_i) &= -\left(1 - [f^*_{\phi_\sigma}]^{-1}(\gamma_i)\right)\beta'_{\phi_\sigma}(\gamma_i) \\ &= -\left(1 - [f^*_{\phi}]^{-1}(\lambda\gamma_i)\right)\beta'_{\phi_\sigma}(\gamma_i) \\ &\neq -\left(1 - [f^*_{\phi}]^{-1}(\lambda\gamma_i)\right)\beta'_{\phi}(\lambda\gamma_i)), \end{aligned}$$

and the shrinkage interpretation no longer holds. One such loss class is defined as follows.

Definition 8 *A tunable regularization loss $\phi_\sigma(v)$ such that*

$$\beta'_{\phi_\sigma}(v) = g(\alpha)\beta'_\phi\left(\alpha\frac{v}{\sigma}\right),$$

where $\beta_\phi(v) \in \mathcal{B}$, $g(\alpha)$ is a constant that depends on α, and $\alpha \geq 0$ is denoted α-tunable.

The additional α parameter enables α-tunable losses to independently control the link and binding functions. In fact, they generalize the previous two loss classes, reducing to shrinkage losses when $\alpha = 1$ and $g(1) = 1$ and canonical losses when $\alpha = 0$ and $g(0)\beta'_\phi(0) = 1$. More generally, the α parameter allows the "interpolation" between pairs of canonical or shrinkage losses of equal generalized logit link. For example, the logistic and exponential losses have the scaled logit of (28) as link function, with $a = 1$ and $a = \frac{1}{2}$, respectively. Since these can be written as $a = \frac{1}{\xi+1}$, for $\xi = 0$ and $\xi = 1$, scaled logits with $\xi \in [0, 1]$ interpolate between the links of the two losses. Similarly, the binding functions of the two losses, given by (42) and (43), are special cases of

$$\beta'_\phi(v) = -\frac{1}{2-b}(e^{-bv} + e^{bv}) \tag{52}$$

with $b = 0$ and $b = 1$. Hence, binding functions with $b = \xi$ and $\xi \in [0, 1]$ interpolate between the binding functions of the two losses. It follows that

$$\phi'(v) = -\left(1 - \frac{e^{(\xi+1)v}}{1 + e^{(\xi+1)v}}\right) \frac{1}{2-\xi}(e^{-\xi v} + e^{\xi v}), \qquad \xi \in [0, 1]$$

interpolates between the derivative of the logistic ($\xi = 0$) and exponential ($\xi = 1$) losses. The derivative of the tunable regularization loss that it generates is

$$\phi'_\mu(v) = -\left(1 - \frac{e^{(\xi+1)\frac{v}{\mu}}}{1 + e^{(\xi+1)\frac{v}{\mu}}}\right) \frac{1}{2-\xi}(e^{-\xi\frac{v}{\mu}} + e^{\xi\frac{v}{\mu}}), \qquad \xi \in [0, 1].$$

Defining $\sigma = \frac{\mu}{\xi+1}$ and $\alpha = \frac{\xi}{1+\xi}$, this can be written as

$$\phi'_\sigma(v) = -\left(1 - \frac{e^{\frac{v}{\sigma}}}{1 + e^{\frac{v}{\sigma}}}\right) \frac{1-\alpha}{2-3\alpha}(e^{-\alpha\frac{v}{\sigma}} + e^{\alpha\frac{v}{\sigma}}), \qquad \alpha \in \left[0, \frac{1}{2}\right], \tag{53}$$

i.e. a α-tunable loss of scaled logit link, $g(\alpha) = \frac{1-\alpha}{2-3\alpha}$, and the binding function of (52). Figure 8 shows the weight function, $w_\sigma(\gamma) = -\phi'_\sigma(\gamma)$, of this loss as a function of the normalized margin $\gamma = v/\sigma$, for different values of α. As α varies, the weight function interpolates between the asymptotically constant weights of LogitBoost (less outlier sensitivity) and the exponential weights of AdaBoost (more sensitive to outliers).

Note that, due to their ability to independently control the link and binding functions, α-tunable losses can always implement this type of interpolation. This can be used to design losses that adapt to the presence of outliers in the data, by cross-validation of α. It should be noted, however, that not all values of $\alpha \geq 0$ lead to sensible loss functions. This is due to the fact that (49) does not hold for these losses. For shrinkage losses, where the property holds, $\phi_\sigma(v) \to 0$ as $v \to \infty$ (whenever $\phi(v)$ has this property), guaranteeing that examples of large positive margin have zero weight. For α-tunable losses, where (49) does not hold, $\beta'_{\phi_\sigma}(v)$ can decrease to $-\infty$ faster than $1 - [f^*_{\phi_\sigma}]^{-1}(v)$ goes to zero, as $v \to \infty$. In this case, examples of large positive margin can receive large positive weight, which is usually undesirable. The losses of (53) have this behavior for $\alpha > 1/2$.

7. Experiments

In this section we discuss various experiments conducted to evaluate different properties of probability regularization.

7.1 Experiments on Two Gaussian Classes

To gain some insight on probability regularization, we considered a simple classification problem, composed of two Gaussian classes of identity covariance, $\mathbf{\Sigma} = \mathbf{I}$, on a two-dimensional space. The means were set to $(0, 0)$ and $(0.7416, 0.7416)$, so as to produce a problem with a Bayes error of 30%. Classifiers were learned with training sets of variable size and evaluated with a test set of $10,000$ examples. All classifiers were learned with BoostLR and the GLog loss, using histogram-based weak learners (Masnadi-Shirazi and

Vasconcelos, 2011; Rasolzadeh et al., 2006; Wu et al., 2004). We started by investigating how the probability estimates varied with the regularization gain σ. The accuracy of the probability estimates was measured by the mean squared error

$$MSE = \frac{1}{n}\sum_{i-1}^{n}[\eta(\mathbf{x}_i) - \hat{\eta}(\mathbf{x}_i)]^2, \tag{54}$$

where $\eta(\mathbf{x}_i)$ and $\hat{\eta}(\mathbf{x}_i)$ are the true and estimated posterior probability for test example $\mathbf{x}_i$. The latter was obtained with (51), where $G^{(T)}(\mathbf{x})$ is the predictor learned by BoostLR. Three regimes were considered. The very small sample regime, where the training set contained $N = 5$ examples per class, the moderate sample size regime, where $N = 40$ and the large sample regime, where $N = 1,000$. Classifiers were learned with BoostLR under the three regimes, for a range of values of σ in the interval $[0.5, 1000]$. Figure 9 shows two complementary views of the MSE data. The top row presents the classical curves of MSE vs. number of boosting iterations T, for different regularization gains. These plots are most useful to assess overfitting, which happens when there is a range of T over which the MSE increases. It is clear that, for both the small and moderate sample sizes, all classifiers eventually overfit as the number of boosting iterations increases, while no overfitting is observed for large sample sizes. The bottom row is most useful to assess the impact of predictor regularization. The data is the same, but these plots show the evolution of the MSE with σ for fixed T. In this case, overfitting occurs on the left of each plot (small values of σ, not enough regularization) and underfitting (too much regularization) on the right.

Overall, the plots demonstrate the complementarity between loss-based probability regularization and classic parameter regularization (due to early stopping, i.e. limiting the number of weak learners in the final ensemble). This is most clear in the moderate sample regime, where many of the curves of the middle column of Figure 9 (top) have the same minimum. Varying the gain σ shifts this minimum, i.e. makes it occur at different numbers of boosting iterations. Hence, when a regularization loss is used, there is less need for early stopping (parameter regularization). This explains the empirical observation that boosted classifiers can do well even with little parameter regularization (e.g. boosted object detectors with thousands of weak learners commonly used in computer vision (Viola and Jones, 2004)). The problem with early stopping is that it can be insufficient for small samples. This is visible in the left column of Figure 9 (top), where there is too little data and boosting overfits *even in the earliest iterations*. The same happens for the moderate sample size (middle column of Figure 9 top) when the regularization gain is small. In these cases, by amplifying parameter based regularization, loss-based regularization can substantially improve the quality of probability estimates. For example, larger σ lead to significant gains in estimation accuracy, for all numbers of boosting iterations, in the left column of Figure 9 (bottom). As σ increases, the best early-stopped MSE ($T = 2$) decreases from roughly 20% to about 5%. Hence, for small samples, loss-based regularization is much more effective than early stopping.

In summary, loss-based regularization is a more flexible way to control the generalization ability of the boosted classifier than early stopping. Hence, in all remaining experiments, we fix the number of boosting iterations and cross-validate the regularization gain. This regularization strategy has one additional property of interest. As can be seen in the bottom

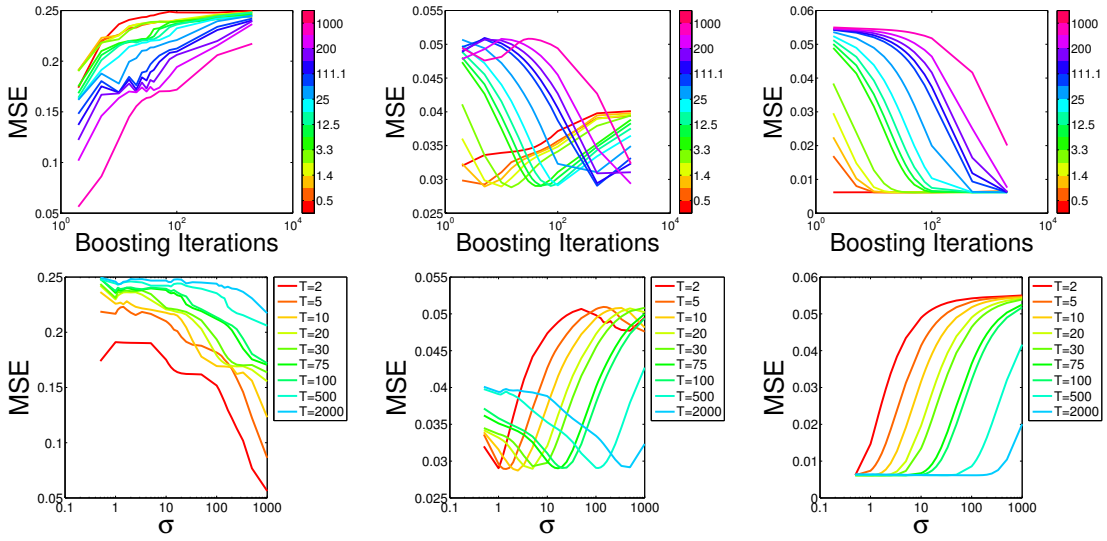


Figure 9: Top: MSE as a function of the number of boosting iterations T for different regularization gains. Bottom: MSE as a function of regularization gain σ for different numbers of boosting iterations T. From left to right: small, moderate sized, and large samples.

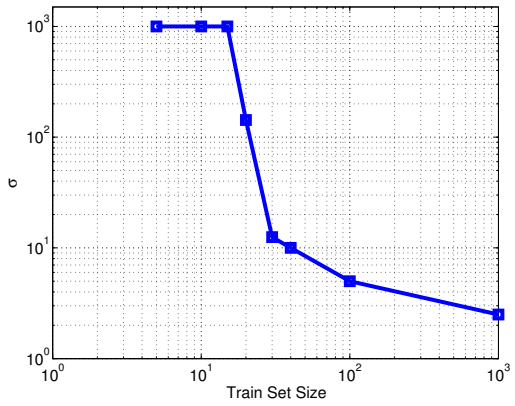


Figure 10: Cross-validated regularization gain as a function of training set size.

row of Figure 9, when the number of iterations T is fixed, the best performing regularization gain decreases with the sample size. This suggests that, when T is fixed, the cross-validated σ can be seen as a diagnostic of whether the classifier would benefit from the collection of further training data. Small samples (left of the figure) require large σ, while a small σ is sufficient for large samples (right). This effect is illustrated in Figure 10, which presents a plot of the cross-validated regularization gain as a function of training set size. Note the monotonic relation between the two variables, suggesting that regularization gain can be used as a diagnostic for data scarcity. While a large σ suggests that it is worth collecting more training data, a small σ indicates that such an effort is likely not justified. This can help learning practitioners perform cost-benefit analysis of their data collection efforts.

7.2 The Role of the Link Function

The next set of experiments used ten binary UCI data sets of relatively small size: (#1) sonar, (#2) breast cancer prognostic, (#3) breast cancer diagnostic, (#4) original Wisconsin breast cancer, (#5) Cleveland heart disease, (#6) tic-tac-toe, (#7) echo-cardiogram, (#8) Haberman's survival, (#9) Pima-diabetes, and (#10) liver disorder. These experiments aimed to evaluate the impact of of the choice of regularization (link) function on calibration and classification accuracy. Since, as discussed in Section 6.3, canonical losses implement all regularization behaviors possible for tunable regularization losses, we only considered the losses of Table 2 in these experiments. Each data set was split into five folds, four of which were used for training and one for testing. This created four train-test pairs per data set, over which the results were averaged. In all experiments, three of the four training folds were used for classifier training and one as validation set for parameter selection.

BoostLR was run for 50 iterations, using histogram-based weak learners and regularization gains $\sigma \in [0.3, 500]$. Classification accuracy was measured with test error. Since the true posterior probabilities are not known for the UCI data sets, calibration cannot be evaluated with (54). A measure of calibration commonly used when this is the case is the cross-entropy between the distributions of the true η and estimated posterior probabilities $\hat{\eta}$ (Niculescu-Mizil and Caruana, 2005). Assuming the quantization of all probabilities into K probability bins, this is defined as

$$H(\eta, \hat{\eta}) = -\sum_{k=1}^{K} p(\eta = k) \log p(\hat{\eta} = k) = -E_{\eta}[\log p(\hat{\eta})].$$

For large samples, the cross-entropy can be estimated with

$$H(\eta, \hat{\eta}) = -\sum_{i=1}^{N} \frac{1}{N} \log p(\hat{\eta}(x_i)).$$

This measure is largest for poorly calibrated classifiers that produce bimodally distributed posterior estimates, concentrated around $\hat{\eta} = 0$ and $\hat{\eta} = 1$, and smallest for well calibrated classifiers whose distribution of posteriors is less concentrated, and spread more evenly between zero and one (Niculescu-Mizil and Caruana, 2005; Mease and Wyner, 2008).

Figure 11 presents curves of the average calibration and classification ranks of the predictor designed with the GLog loss for each σ. Similar curves were obtained for all losses

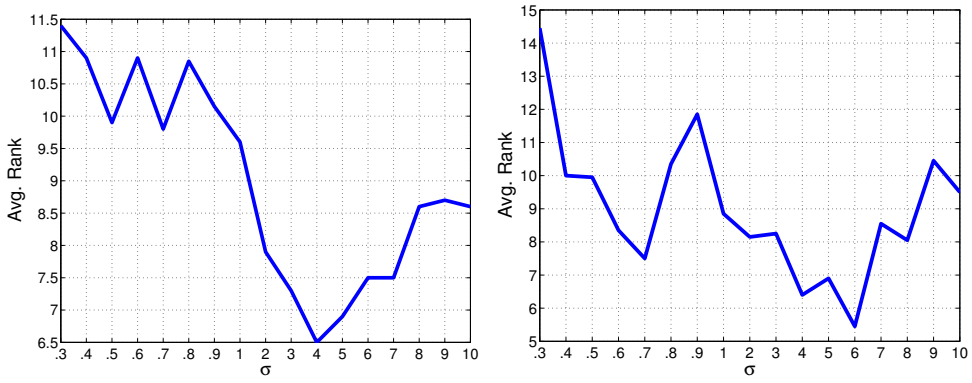


Figure 11: Average calibration (left) and classification (right) rank as a function of regularization gain for the GLog loss on the UCI data.

of Table 2. To produce these plots, a predictor was trained per data set, for 17 values of $\sigma \in [0.3, 10]$. The results were then ranked, and rank 1 (17) assigned to the value of σ of smallest (largest) cross-entropy or classification error. The ranks of each σ were then averaged over the ten data sets (Demšar, 2006). Note that the curves of classification accuracy and cross entropy rank have similar shape, although the rank curve is smoother for cross-entropy. This is because the classifier produces binary decisions by thresholding the predictor output. Nevertheless, the two plots support the conclusion that the best values of σ for these data sets are in the range of $4 \leq \sigma \leq 6$. Note that the average calibration rank for this range (between 6.5 and 7.5), is substantially better than that (more than 9.5) of the logistic loss of Figure 1 (which is identical to GLog with $\sigma = 1$). For classification, the difference is similar (between 5.5 and 6.5 for $4 \leq \sigma \leq 6$, around 9 for $\sigma = 1$). In summary, regularization strength can have a significant impact in both classification and calibration performance. The fact that best results occur for relatively large regularization gains is not surprising, given that these data sets are relatively small.

We next attempted to quantify the intrinsic regularization gain of each data set, i.e. the regularization gain that leads to best performance on that data set across all losses, and the benefits of using that regularization over the standard values (e.g. $\sigma = 1$ for the logistic loss in LogitBoost). For this, we averaged the performance of all BoostLR classifiers learned with the four losses of Table 2, for each value of σ and data set. We then determined the gain σ_{opt} of smallest average classification error per data set. This can be seen as a loss-independent measure of the intrinsic regularization gain of the data set. The associated classification error is a loss-independent estimate of the performance of a classifier tuned to this intrinsic regularization value. These results are summarized in Table 3 (top). For comparison, we also present the results of AdaBoost, LogitBoost (GLog loss with $\sigma = 1$), the average performance of BoostLR with the four losses of Table 2 when the bandwidth is constrained to $\sigma = 1$, and the drop in classification error due to the tuning to the intrinsic regularization gain of the data set. To compute this drop, we defined as ϵ_1 the average error of the BoostLR methods with the intrinsic gain, as ϵ_2 the smallest error of all other

UCI data set#	#1	#2	#3	#4	#5	#6	#7	#8	#9	#10
Classification										
AdaBoost	**11.4**	15.2	9.2	6	11.4	21.6	7.4	23.2	42.8	26.6
LogitBoost($\sigma = 1$)	12.4	15.4	8.6	5.6	11.4	46	7.2	25	40.4	**26.4**
Avg. BoostLR($\sigma = 1$)	13.25	16.4	8.06	5.53	11.6	47.95	7.15	24.6	40.65	27.4
Avg. BoostLR(σ_{opt})	11.6	**14.95**	**6.93**	**4.86**	**11.1**	**13.25**	**6.7**	**14.6**	**38.8**	26.5
Drop(%)	−1.75	1.64	14.08	12.11	2.63	38.65	6.29	37.06	3.96	−0.37
Calibration										
AdaBoost	4.70	4.40	5.31	5.58	3.89	3.453	3.77	**3.593**	3.43	3.54
LogitBoost($\sigma = 1$)	4.73	4.06	5.16	5.49	3.68	**3.414**	3.71	3.609	3.42	3.58
Avg. BoostLR($\sigma = 1$)	4.25	3.88	5.20	5.63	3.77	3.419	3.68	3.599	3.41	3.65
Avg. BoostLR(σ_{opt})	**3.71**	**3.83**	**4.48**	**4.82**	**3.58**	**3.414**	**3.50**	3.595	**3.39**	**3.53**
Drop(%)	58.2	8.8	37.3	30.8	29.2	0.0	48.2	−0.7	26.3	5.3

Table 3: Intrinsic gain of regularization, in terms of classification error (top) and probability estimation accuracy (bottom), on various UCI data sets. Avg. BoostLR(σ) is the average error of classifiers learned with the margin losses of Table 2, for regularization bandwidth σ. σ_{opt} is the bandwidth of smallest average error.

methods, and the drop as $(1 - \frac{\epsilon_1}{\epsilon_2}) \times 100\%$. Note that BoostLR($\sigma_{opt}$) outperformed all other approaches in 8 out of the 10 data sets, virtually tied the best approach in one, and performed slightly worse than the best method (AdaBoost) in another. On four of the data sets its relative drop in classification error was larger than 10% and in two larger than 30%. Note also that the averaging over the four losses does not give an unfair advantage to BoostLR (σ_{opt}), since the same average for BoostLR($\sigma = 1$) has performance equivalent to LogitBoost (which uses one of the four losses of unit gain). A similar analysis is presented in the bottom half of Table 3 for calibration performance. In this case, the drop is defined as $(1 - \frac{H_1 - H}{H_2 - H}) \times 100\%$ where H_1 is the average cross entropy of the BoostLR methods with the intrinsic gain, H_2 the smallest cross entropy of all other methods and H the entropy (minimum possible cross entropy value) of the problem. BoostLR (σ_{opt}) outperformed all other approaches in 8 out of the 10 data sets with a relative drop in cross entropy of more than 10% on six, more than 30% on four and more than 40% on two data sets. These results show that, for an equal amount of parameter regularization (all classifiers have the same number of weak learners) there can be substantial gains in tuning the regularization strength of the loss.

We next evaluated the performance of the individual regularization losses. Since they are canonical, this is equivalent to comparing the associated link $f^*_{\phi_\sigma}$ or regularization strength ρ_{ϕ_σ} functions of (45). Given that the the number of boosting iterations is the same for all methods, i.e. all classifiers have the same amount of parameter regularization, this comparison is indicative of the effectiveness of the different link functions as probability regularizers. The top half of Table 4 presents the average test error obtained for each UCI data set and loss. Also shown are the baseline results of AdaBoost and LogitBoost (GLog loss with $\sigma = 1$). The last two columns present two statistics, reporting to the number of wins of each algorithm. This is the number of data sets in which the algorithm outperformed a set

UCI data set#	#1	#2	#3	#4	#5	#6	#7	#8	#9	#10	W_1	W_2
Classification												
AdaBoost	11.4	**11.4**	9.4	6.4	14	28	6.6	21.8	41.2	28.2	-	1
LogitBoost	11.6	12.4	10	6.6	13.4	48.6	6.8	21.2	39.6	28.4	-	0
GLog	**11.2**	**11.4**	**8**	5.6	**12.4**	11.8	7	18.8	38.2	**27**	9	5
GBoost	12.6	11.6	21	18.6	17.6	**7.2**	**6**	21.8	**37.6**	28.6	3	3
GGauss	13.6	14.4	9	6	13	8.8	7.6	**18.4**	38.4	30.6	6	1
GLaplace	12	12.8	9	**5**	**12.4**	8.2	6.6	20.8	40.6	31.6	6	2
BoostLR wins	1	1	3	3	3	4	2	3	3	1	-	-
Drop (%)	1.7	0	14.9	21.9	7.5	74.3	9.1	13.2	5.0	4.2	-	-
Calibration												
AdaBoost	4.59	4.19	5.47	3.94	5.77	3.61	4.71	3.48	3.442	3.461	-	0
LogitBoost	4.75	3.85	5.47	3.861	5.65	**3.57**	4.64	3.426	3.438	3.48	-	1
GLog	4.20	3.46	4.59	3.80	5.42	3.67	3.89	3.421	**3.40**	3.49	8	1
GBoost	**3.77**	4.60	5.33	**3.69**	**5.21**	3.65	3.83	3.406	3.41	**3.44**	8	4
GGauss	4.07	**3.44**	4.70	3.71	5.49	3.62	3.87	3.429	3.439	3.53	6	1
GLaplace	3.81	3.48	**4.58**	3.76	5.31	3.63	**3.81**	3.41	3.42	3.45	9	2
BoostLR wins	4	3	4	4	4	0	4	3	3	2	-	-
Drop (%)	64.52	76.53	41.56	30.18	19.11	−6.50	62.76	22.23	28.52	13.01	-	-

Table 4: Cross validated classification error (top) and cross entropy (bottom) for each loss function and UCI data set. W_1 : number of wins over AdaBoost and LogitBoost. W_2 : number of wins over all methods.

of competitors. The two statistics differ in the composition of this set. W_1 compares the performance of each tunable regularization loss to the AdaBoost and LogitBoost baselines, evaluating how frequently each version of BoostLR outperforms the well established boosting methods. W_2 uses all other algorithms in the table as competitors, measuring how many times each algorithm achieved the best performance among all methods considered. Finally, the last two rows report similar statistics per data set. The row before last reports the number of BoostLR algorithms that outperformed both AdaBoost and LogitBoost. The last row presents the drop in test error between the established boosting methods and BoostLR. To compute this drop, we found the smallest test error ϵ_1 of Ada and LogitBoost, the smallest test error ϵ_2 of all BoostLR methods, and defined the drop as $(1 - \epsilon_2/\epsilon_1) \times 100\%$.

Several conclusions can be drawn from the table. First, statistic W_1 shows that BoostLR with either the GLog, GGauss, or GLaplace losses, beats both AdaBoost and LogitBoost in at least half of the data sets. Best performance was achieved by GLog, which beat the established methods in 9 out 10 data sets. Second, statistic W_2 shows that, while BoostLR with the GLog loss (logistic link) has the overall best performance, different links perform best for different data sets (3 overall wins for GLaplace, 2 for GBoost, and 1 for GGauss). Third, the gains of tunable loss regularization vary substantially from data set to data set. This is clear from the last two rows of the table, where BoostLR is shown to have modest improvements (less that 5% drop in error rate) for 3 data sets, significant gains (between 5 and 20% drop) in 5, and massive gains (above 20%) in 2. In general, the magnitude of the gain is correlated with the number of BoostLR variants that beat AdaBoost and

LogitBoost, e.g. the more variants beat the established methods the largest the drop in classification error. This suggests that the regularization gains of AdaBoost and LogitBoost are severely mistuned for these data sets.

The bottom half of Table 4 presents a similar analysis for calibration performance, using the cross entropy criteria. In this case the drop is defined as $(1 - \frac{H_1 - H}{H_2 - H}) \times 100\%$, where H_1 is the smallest cross entropy of Ada and LogitBoost, H_2 the smallest cross entropy of all BoostLR methods and H the entropy (minimum possible cross entropy value) of the problem. The cross entropy criteria produced similar results in terms of number of wins, but the drop in relative cross entropy was much more substantial, with a drop of more than 10% on nine data sets, more than 20% on seven, more than 40% on four and more than 60% on three.

We next evaluated the impact of the link function in the recovery of posterior probabilities. For this, we performed a comparison between BoostLR with shrinkage loss and GradientBoost + shrinkage. As discussed in Section 6.4, while the two algorithms produce identical classifiers, the posterior probability estimates are not the same. GradientBoost relies on (12), BoostLR uses (51). The probabilities recovered, using the GLog loss, on the ten UCI data sets were compared. In the first set of experiments, the regularization gain of BoostLR was fixed at $\sigma = 10$ and the learning rate of shrinkage at $\lambda = 0.1$. The calibration performance of both algorithms is shown, for each data set, in the top half of Table 5. BoostLR has considerably better calibration on all ten data sets. We also compared the results achieved with cross-validation of the regularization gain of BoostLR and the learning rate of shrinkage. As shown on the bottom half of Table 5, BoostLR has better calibration on seven of the ten data sets. In summary, even for shrinkage losses, where BoostLR and GradientBoost with shrinkage produce identical classifiers, the fact that BoostLR uses the correct link for probability recovery enables it to achieve superior calibration performance.

7.3 The Role of the Binding Function

The following set of experiments aimed to evaluate the impact of the binding function. For this, we considered the scenario where BoostLR differs from GradientBoost with shrinkage even for classification, by using the α-tunable loss of (53). As discussed in Section 6.5, the additional α parameter of this loss enables independent control of binding and link functions. This allows the loss to adapt to the outlier content of the data. To evaluate the benefits of this adaptation, we compared the classification and calibration performance of BoostLR with the loss of (53) to that of AdaBoost with shrinkage. All experiments relied on five-fold cross-validation. For both algorithms the regularization gain σ was cross-validated among 10 values in $[1, 10]$. The α parameter of BoostLR was cross-validated among 5 values in $[0, 1/2]$. Various percentages of outliers were added to the ten UCI data sets by randomly flipping labels of training examples. The classification and calibration performance of the two algorithms are presented in Figure 12. The figure depicts the average rank of the classifiers learned by the two methods, over the ten UCI data sets, as a function of the percentage of outliers. BoostLR has better calibration (smaller rank) for all outlier percentages. This illustrates the benefits of α-tuning for noisy data. For classification, the same holds for all outlier percentages other than 15%. The reversal of ranks for this percentage can be explained by the noisier nature of the classification data

UCI data set#	#1	#2	#3	#4	#5	#6	#7	#8	#9	#10
Fixed $\sigma = 10$ ($\lambda = 0.1$)										
BoostLR	**4.13**	**3.91**	**4.56**	**5.37**	**3.47**	**3.58**	**3.73**	**3.84**	**3.65**	**4.13**
Shrinkage	4.65	4.49	5.30	5.74	4.63	4.97	4.16	4.35	4.97	4.19
Cross validated σ and λ										
BoostLR	**4.19**	**3.89**	**4.59**	**5.38**	**3.46**	3.45	**3.80**	3.62	**3.40**	3.53
Shrinkage	4.66	4.52	5.30	5.70	3.85	**3.42**	3.87	**3.59**	3.43	**3.46**

Table 5: Calibration performance (cross-entropy) of BoostLR and GradientBoost with shrinkage on the UCI data.

(due to the hard decision made by the classifier). Even though the BoostLR classifiers are better calibrated, the classification error is larger. We note that better results should be possible with α-tunable losses that implement binding functions expressly designed to achieve outlier robustness, e.g. that of the Savage loss (Masnadi-Shirazi and Vasconcelos, 2008). This is left for future work. The goal here was not to produce the classifier of greatest possible robustness, only to investigate the benefits of independently controlling the link and binding functions.

7.4 Experiments on Larger Data sets

The data sets used in the previous section are of relatively small size. To investigate the benefits of loss regularization for larger data sets, we considered the ADULT, LETTER.p1 and LETTER.p2 data sets, which are widely used for comparing ensemble methods (Niculescu-Mizil and Caruana, 2005; Caruana et al., 2004). Missing values in the ADULT training and testing sets were omitted, leading to 30,162 training examples, of which 7,508 are positive and 22,654 negative. The test set consists of 15,060 examples, of which 3,700 are positive and 11,360 negative. The LETTER data was converted into two binary data sets (Caruana et al., 2004). The LETTTER.p1 data set treats the confusable letter "O" as the positive class, and the remaining 25 letters of the alphabet as the negative class, resulting in a highly unbalanced classification problem. LETTER.p2 uses the first 13 letters of the alphabet as the negative class and the last 13 as the positive class, resulting in a balanced but difficult problem. Both datasets contain 4,000 training and 16,000 test examples. As before, all classifiers were learned with BoostLR, using histogram weak learners, and cross-validation of the regularization gain. The performance of the GLog and GLaplacian losses was compared to that of the exponential loss, used by AdaBoost, and GLog with unit gain, used by LogitBoost. Each boosting algorithm was run for 100 iterations.

Table 6 presents the error achieved by each method, and the corresponding regularization gain. Note that 1) best performance was never attained with the logistic loss (GLog with $\sigma = 1$) of LogitBoost, or the exponential loss of AdaBoost, 2) each of the two losses of tuned gain outperformed both standard boosting losses, and 3) in each case the gains were substantial. Note also that the optimal σ was always smaller than one. This is explained by the larger size of the datasets used in this experiment. The optimality of small σ in this experiment and larger σ in the experiments of the previous section is in agreement with

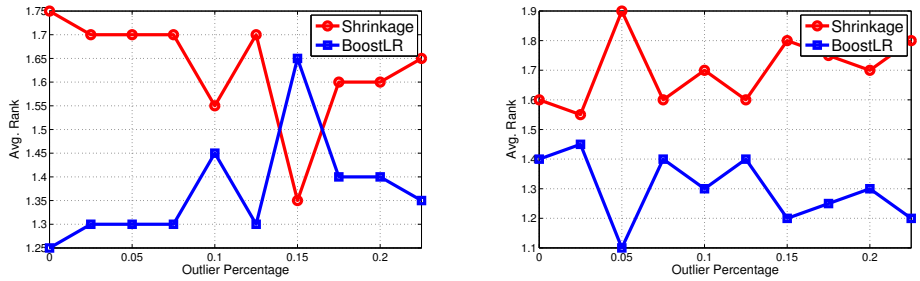


Figure 12: Average classification (left) and calibration (right) rank as a function of percentage of outliers on the UCI data, for BoostLR and AdaBoost with shrinkage.

the observations of Section 7.1. To further investigate this point, we considered reduced versions of LETTER.p2, by randomly subsampling training examples. More precisely, the training set was subsampled by a factor of 2 (DIV2) and 4 (DIV4). The size of the test set was not changed. Table 7 presents 1) the optimal regularization gain for each loss, and 2) the difference between the number of testing errors produced by the exponential and each of the regularization losses, for each training set size. Note how 1) the regularization gain increases for smaller datasets, eventually becoming larger than one, and 2) the classification gains are larger for the smaller datasets. As previously noted in Section 7.1, these results suggest that large margins are important for small datasets but do not add much, to classifier performance, for large ones.

8. Conclusion

Large margins and parameter regularization are commonly used to assure classifier generalization. Large margins are implemented with risks based on margin losses, regularization by inclusion, in these risks, of terms that encourage parameter sparsity. In this work, we have shown that margin losses can also be viewed as regularizers of posterior class probability estimates. In fact, an analysis of both 1) probability estimation error, and 2) generalization bounds, has shown that, for proper losses of generalized logit link, loss-based regularization amplifies the strength of parameter regularization by a factor equal to the loss margin. These losses were also shown to have a simple decomposition in terms of a link and a binding function. The link determines the loss behavior around the classification boundary and is responsible for its regularization strength. The binding function determines the loss behavior for large margins and is responsible for its outlier robustness. In this way, link and binding functions partition the space of losses into equivalence classes of identical probability regularization or outlier robustness. These equivalence classes are isomorphic to the set of symmetric scale probability densities of unique maximum at the origin and the set of monotonically decreasing odd functions, respectively. Each equivalence class contains many tunable regularization losses, parameterized by a regularization gain σ.

Tunable regularization losses can be used to derive boosting algorithms with loss regularization (BoostLR) of tunable strength. Three classes of losses were considered in this

UCI data set	ADULT		LETTER1		LETTER2	
	error	σ	error	σ	error	σ
GLog	**2406**	0.25	427	0.33	**2831**	0.5
GLaplacian	2680	0.45	**420**	0.25	2844	0.3
Exponential	2696		529		2940	
Logit ($\sigma = 1$)	2673		464		2867	

Table 6: Optimal regularization gain and corresponding classification error on the large UCI datasets.

LETTER2	DIV1	DIV2	DIV4
GLog	109	179	260
	$\sigma = 0.5$	$\sigma = 1.66$	$\sigma = 2$
GLaplacian	96	178	186
	$\sigma = 0.3$	$\sigma = 1$	$\sigma = 2$

Table 7: Optimal σ as a function of training set size and corresponding classification error gain over exponential loss.

work: 1) canonical losses, which have linear binding functions and no flexibility in terms of outlier modeling, 2) shrinkage losses, which support equally parameterized link and binding function pairs, and 3) α-tunable losses, which enable independent parameterization of link and binding function. BoostLR algorithms with shrinkage losses were then shown to implement the well known shrinkage procedure. This offers an alternative explanation of shrinkage as regularization of posterior probability estimates, explaining its success in terms of large margins and generalization bounds. On the other hand, the flexibility of α-tunable losses enabled the derivation of a boosting algorithm that generalizes both AdaBoost and LogitBoost, behaving as either of them according to the data to classify.

Extensive experiments on a series of synthetic and UCI datasets showed that, when the regularization gain is optimized, BoostLR can substantially outperform previous boosting algorithms, with respect to both classification error and probability calibration. These results challenge the popular belief that large-margin classifiers are not capable of producing calibrated probability estimates. They also shed some light on the synergies between loss-based and parameter regularization in boosting algorithms, where parameter regularization is usually implemented by early stopping. For small samples, which demand strong regularization, this can be insufficient, and a large loss regularization gain required. For large samples, where little regularization is necessary, the bias introduced by the combination of parameter and loss regularization can be too large. Better results can be obtained by weakening the regularization. This can be accomplished by using a smaller σ.

Appendix A. Relations Between Loss Margin and Regularization Strength

In this appendix, we determine the conditions under which the loss margin μ_ϕ of (25) is a measure of the regularization strength of the loss ϕ.

Lemma 9 *Let $\phi(v)$ be a twice differentiable proper loss of monotonically increasing inverse link $[f^*_\phi]^{-1}(\eta)$. Then (26) holds. Furthermore, $[f^*_\phi]^{-1}(\eta)$ has an inflection point at the origin. If this inflection point is the maximum of $\{[f^*_\phi]^{-1}\}'(v)$, then the regularization strength is lower bounded by twice the loss margin, as in (27), and $\phi(v)$ is a regularization loss if and only if $\mu_\phi \geq \frac{1}{2}$.*

Proof If ϕ is proper, it follows from (24) that

$$\begin{aligned}
\phi'(v) &= \left(1 - [f_\phi^*]^{-1}(v)\right) [C_\phi^*]'' \left([f_\phi^*]^{-1}(v)\right) \{[f_\phi^*]^{-1}\}'(v) \\
\phi''(v) &= -\left(\{[f_\phi^*]^{-1}\}'(v)\right)^2 [C_\phi^*]'' \left([f_\phi^*]^{-1}(v)\right) \\
&+ \left(1 - [f_\phi^*]^{-1}(v)\right) [C_\phi^*]^{(3)} \left([f_\phi^*]^{-1}(v)\right) \left(\{[f_\phi^*]^{-1}\}'(v)\right)^2 \\
&+ \left(1 - [f_\phi^*]^{-1}(v)\right) [C_\phi^*]'' \left([f_\phi^*]^{-1}(v)\right) \{[f_\phi^*]^{-1}\}''(v).
\end{aligned}$$

From (22) and (23), $[f_\phi^*]^{-1}(0) = 1/2$, $[C_\phi^*]^{(3)}(\eta) = -[C_\phi^*]^{(3)}(1-\eta)$, and $\{[f_\phi^*]^{-1}\}''(v) = -\{[f_\phi^*]^{-1}\}''(-v)$, and it follows that

$$\{[f_\phi^*]^{-1}\}''(0) = 0 \tag{55}$$

$$[C_\phi^*]^{(3)}\{[f_\phi^*]^{-1}(0)\} = 0, \tag{56}$$

from which $\phi'(0) = \frac{1}{2}[C_\phi^*]'' \left(\frac{1}{2}\right) \{[f_\phi^*]^{-1}\}'(0)$, $\phi''(0) = -\left(\{[f_\phi^*]^{-1}\}'(0)\right)^2 [C_\phi^*]'' \left(\frac{1}{2}\right)$, and

$$\mu_\phi = \frac{\{[f_\phi^*]^{-1}\}'(0)}{2\left(\{[f_\phi^*]^{-1}\}'(0)\right)^2} = \frac{\rho_\phi(0)}{2}.$$

Furthermore, from (55), $[f_\phi^*]^{-1}$ has an inflection point at the origin. From (14), if this point is a maximum of $\{[f_\phi^*]^{-1}\}'$, then $\rho_\phi(v) \geq \rho_\phi(0)$ for all v, (27) holds, and the theorem follows. ■

Appendix B. The Generalized Logit Link

In this appendix, we discuss some properties of the generalized logit link that are used in the remaining results of this work.

B.1 Properties

We start by noting that the conditions of Definition 2 are a set of sufficient conditions for a function to be the link of a proper loss. The monotonicity of Property 1. is sufficient for the invertibility of π. While it is not necessary that π^{-1} be increasing, this guarantees that the probability estimates $\eta = \pi^{-1}(p)$ increase with p. Property 2. and 3. suffice for π to be a link of some proper loss. Property 3. is the condition of (23). When combined with 1. and 2. it constrains $\pi^{-1}(v)$ to be in $[0, 1]$. This guarantees that η is a probability. While Property 3. is necessary, this is not the case of Property 2. For example,

$$\pi^{-1}(v) = \frac{1+v}{2}, \qquad v \in [-1, 1]$$

is a valid inverse link. However, the use of such a link requires that $p(\mathbf{x}) \in [-1, 1]$ for $\eta(\mathbf{x}) = \pi^{-1}(p(\mathbf{x}))$ to be a probability. This constraint on $p(\mathbf{x})$ has to be enforced by learning algorithms, complicating the underlying optimization. We are aware of no benefit

in adopting such a link over a generalized logit. Property 2. eliminates all links of this type. Finally, Property 4. is necessary and sufficient for π^{-1} to have a unique inflection point at the origin. Note that the if statement follows from Property 3. but not the only if. A "staircase" of sigmoids could satisfy 1.-3. and have multiple inflection points. Property 7. of the following lemma shows that this suffices for the inverse of the generalized logit to have maximum derivative at the origin. It follows that all conditions of Lemma 9 hold when $f_\phi^*(\eta)$ is a generalized logit link, proving Theorem 4.

Lemma 10 *A generalized logit π has the following properties*

1. $\pi^{-1}(v) \in (0,1)$
2. $\lim_{v\to-\infty} \pi^{-1}(v) = 0$
3. $\pi^{-1}(0) = .5$
4. $(\pi^{-1})^{(n)}(-v) = (-1)^{n+1}(\pi^{-1})^{(n)}(v)$
5. $(\pi^{-1})^{(n)}(0) = 0$, *whenever n is even*
6. $\lim_{v\to\pm\infty}(\pi^{-1})^{(n)}(v) = 0, n \geq 1.$
7. $(\pi^{-1})'(v)$ *has a unique maximum at the origin.*

Proof Properties 1.-5. are a straightforward consequence of Properties 1.-3. of Definition 2. Property 6. follows from the fact that π^{-1} is monotonically increasing and lower and upper bounded by 0 and 1, respectively. Property 7. then follows from the fact that $(\pi^{-1})'$ is positive for all v and only has one critical point at the origin, by Property 4. of Definition 2. ■

B.2 Parametric Generalized Logit Links

In this section we show that the set $\mathcal{L}$ of generalized logit links is isomorphic to a set of probability density functions.

Lemma 11 *The set $\mathcal{L}$ of parametric generalized logit links of (38) is isomorphic to the set of parametric continuous scale probability density functions (pdfs)*

$$\psi_\sigma(v) = \frac{1}{\sigma}\psi\left(\frac{v}{\sigma}\right),$$

where $\psi(v)$ has unit scale, a unique maximum at the origin, and $\psi(-v) = \psi(v)$.

Proof Let $c(v) = \int \psi(v)dv$ be the cdf of a continuous scale pdf $\psi(v)$. Then $c(v)$ satisfies Properties 1. and 2. of Definition 2. Property 3. is also met if $\psi(v)$ has symmetry $\psi(-v) = \psi(v)$, and Property 4. if $\psi(v)$ has a unique maximum at the origin. Finally, from the continuity of $\psi(v)$, $c(v)$ has an inverse and $c^{-1}(v)$ is a generalized logit link. Since any generalized logit link with the properties of Definition 2 defines one such cdf, the set of

generalized logit links is isomorphic to the set of continuous scale pdfs $\psi(v)$ of symmetry $\psi(-v) = \psi(v)$ and a unique maximum at the origin.

Let $\psi(v)$ be the pdf corresponding to $f^*_\phi(\eta)$, i.e. $[f^*_\phi]^{-1}(v) = \int_{-\infty}^{v} \psi(q)dq$. Then, for any σ, it follows from (38) that

$$[f^*_{\phi_\sigma}]^{-1}(v) = [f^*_\phi]^{-1}\left(\frac{v}{\sigma}\right)$$

is the cdf of $\psi_\sigma(v)$, as defined in (44). Since this procedure can be repeated for any link function $f^*_\phi(\eta)$, $\mathcal{L}$ is isometric to the set of these pdfs. ■

Appendix C. The Binding Function

In this appendix, we discuss the properties of the binding function.

Lemma 12 *Let $\beta_\phi(v)$ be the binding function of a proper loss $\phi(v)$ of generalized logit link $f^*_\phi(\eta)$, and minimum risk $C^*_\phi(\eta)$. Then*

1. *the behavior of $\phi(v)$ for $v \to \pm\infty$ is determined by $\beta_\phi(v)$.*
2. *$\beta_\phi(v)$ is monotonically decreasing.*
3. *the mapping $[C^*_\phi]'(\eta) = \beta_\phi\left(f^*_\phi(\eta)\right)$ is one-to-one.*
4. *$\beta_\phi(v)$ is an odd function, i.e. $\beta_\phi(-v) = -\beta_\phi(v)$.*

Proof To prove Property 1. we note that, combining (31) with Properties 2. of Definition 2 and Lemma 10, and $C^*_\phi(0) = C^*_\phi(1) = 0$, it follows that

$$\lim_{v\to\pm\infty} \phi(v) = \lim_{v\to\pm\infty} (1 - [f^*_\phi]^{-1}(v))[C^*_\phi]'\{[f^*_\phi]^{-1}(v)\}.$$

The property follows from the fact that $\lim_{v\to\pm\infty}(1-[f^*_\phi]^{-1}(v)) \in \{0, 1\}$ and (34). Property 2 follows from the fact that

$$\beta'_\phi(v) = [C^*_\phi]''\left([f^*_\phi]^{-1}(v)\right)\{[f^*_\phi]^{-1}\}'(v)$$

C^*_ϕ is concave (Theorem 3) and $\{[f^*_\phi]^{-1}\}'(v) > 0$ (Property 1 of Definition 2). Property 3 then follows from (34) and Property 2. Finally, Property 4 follows from

$$\begin{aligned}
\beta_\phi(-v) &= [C^*_\phi]'\left([f^*_\phi]^{-1}(-v)\right) \\
&= [C^*_\phi]'\left(1 - [f^*_\phi]^{-1}(v)\right) \\
&= -[C^*_\phi]'\left([f^*_\phi]^{-1}(v)\right) = -\beta_\phi(v).
\end{aligned}$$

where we have used (22) and (23). ■

Appendix D. Properties of Proper Losses

In this appendix, we derive various properties of proper losses.

D.1 Proper Losses of Generalized Logit Link

The following lemma summarizes various properties of proper losses with generalized logit link.

Lemma 13 *Let $\phi(v)$ be a proper loss of generalized logit link $f^*_\phi(\eta)$ and binding function $\beta_\phi(v)$. Then, the following properties hold.*

1. *$\phi(v)$ is monotonically decreasing*
2. *$\phi(v)$ is convex if and only if*

$$\frac{\beta''_\phi(v)}{\beta'_\phi(v)} < \frac{\{[f^*_\phi]^{-1}\}'(v)}{(1-[f^*_\phi]^{-1}(v))}, \quad \forall v \tag{57}$$

3. $\lim_{v\to-\infty}\phi(v) = \lim_{v\to-\infty}\beta_\phi(v)$
4. $\phi'(0) = \frac{1}{2}\beta'_\phi(0)$
5. $\phi''(0) = -\frac{\beta'_\phi(0)}{\rho_\phi(0)}$.

Proof Property 1. follows from (35) and the facts that $(1-[f^*_\phi]^{-1}(v)) > 0$ (Properties 1. and 2. of Definition 2) and $\beta'_\phi(v) < 0$ (Property 2. of Lemma 12). To prove Property 2. we take derivatives on both sides of (35),

$$\phi''(v) \;=\; -\{[f^*_\phi]^{-1}\}'(v)\beta'_\phi(v) + (1-[f^*_\phi]^{-1}(v))\beta''_\phi(v).$$

It follows that $\phi(v)$ is convex if and only if, for all v, $\{[f^*_\phi]^{-1}\}'(v)\beta'_\phi(v) < (1-[f^*_\phi]^{-1}(v))\beta''_\phi(v)$. Since $(1-[f^*_\phi]^{-1}(v)) > 0$ and $\beta'_\phi(v) < 0$, this is identical to (57). Property 3. follows from (36) and Property 2. of Lemma 10, since $\lim_{v\to-\infty}\phi(v) = C^*_\phi(0) + \lim_{v\to-\infty}(1-[f^*_\phi]^{-1}(v))\beta_\phi(v)$, $C^*_\phi(0) = 0$, and $\lim_{v\to\infty}(1-[f^*_\phi]^{-1}(v)) = 1$. Property 4. is a simple consequence of (23), which implies that $[f^*_\phi]^{-1}(0) = \frac{1}{2}$. Finally, Property 5. follows from $\phi''(0) = -\{[f^*_\phi]^{-1}\}'(0)\beta'_\phi(0) + \frac{1}{2}\beta''_\phi(0)$ and Property 4. of Lemma 12, which implies that $\beta''_\phi(0) = 0$. ∎

D.2 Canonical Regularization Losses

The following lemma summarizes various properties of canonical regularization losses.

Lemma 14 *Let $\phi_\sigma(v)$ be a tunable regularization loss of binding function as in (46). The following properties hold.*

1. $\phi''_\sigma(v) > 0, \forall v$
2. $\lim_{v\to\infty} \phi'_\sigma(v) = 0$
3. $\lim_{v\to-\infty} \phi'_\sigma(v) = -1$
4. $\phi'_\sigma(0) = -1/2$
5. *ϕ''_σ is maximum at the origin.*
6. *the loss margin and regularization strength are related by* $2\mu_{\phi_\sigma} = \rho_{\phi_\sigma}(0) = \frac{1}{\phi''_\sigma(0)}$.

Proof Properties 1. and 2. follow from (47) and Properties 1. and 2. of Definition 2. Properties 3. to 5. follow from Properties 2., 3., and 7. of Lemma 10. Property 6. follows from $\mu_{\phi_\sigma} = \sigma\mu_\phi$ and the combination of (29), Property 5. of Lemma 13, and (46). ■

References

S. Avidan. Ensemble tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29:261–271, 2007.

P. Bartlett, M. Jordan, and J.D. McAuliffe. Convexity, classification, and risk bounds. *Journal of the American Statistical Association*, 101:138–156, 2006.

P.J. Bickel, Y. Ritov, and A. Zakai. Some theory for generalized boosting algorithms. *Journal of Machine Learning Research*, 7:705–732, 2006.

G. Blanchard, G. Lugosi, and N. Vayatis. On the rate of convergence of regularized boosting classifiers. *Journal of Machine Learning Research*, 4:861–894, 2003.

P. Buhlmann and T. Hothorn. Boosting algorithms: Regularization, prediction and model fitting. *Statistical Science*, 22:477–505, 2007.

P. Buhlmann and B. Yu. Boosting with the L2 Loss: Regression and Classification. *Journal of the American Statistical Association*, 98:324–339, 2003.

A. Buja, W. Stuetzle, and Y. Shen. Loss functions for binary class probability estimation and classification: Structure and applications. 2006.

R. Caruana, A. Niculescu-mizil, G. Crew, and A. Ksikes. Ensemble selection from libraries of models. In *International Conference on Machine Learning*, pages 137–144, 2004.

O. Chapelle. Training a support vector machine in the primal. *Neural Computation*, 19: 1155–1178, 2007.

K. Chen and S. Wang. Regularized boost for semi-supervised learning. In *Advances in Neural Information Processing Systems*, volume 20, pages 281–288. MIT Press, 2008.

K. Chen and S. Wang. Semi-supervised learning via regularized boosting working on multiple semi-supervised assumptions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33:129–143, 2011.

M. Culp, K. Johnson, and G. Michailidis. On adaptive regularization methods in boosting. *Journal of Computational Graphics and Statistics*, 20:804–937, 2011.

M.H. DeGroot and S.E. Fienberg. The comparison and evaluation of forecasters. *The Statistician*, 32:14–22, 1983.

J. Demšar. Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, 7:1–30, 2006.

T.G. Dietterich. An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting, and randomization. *Machine Learning*, 40:139–157, 2000.

J. Friedman. Greedy function approximation: A gradient boosting machine. *Annals of Statistics*, 29(5):1189–1232, 2001.

J. Friedman, T. Hastie, and R. Tibshirani. Additive logistic regression: A statistical view of boosting. *Annals of Statistics*, 28:337–407, 2000.

T. Gneiting and A.E. Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, 102:359–378, 2007.

M. Gonen, A. G. Tanugur, and E. Alpaydm. Multiclass posterior probability support vector machines. *IEEE Transactions on Neural Networks*, 19(1):130–139, 2008.

Hastie, Tibshirani, and Friedman. *The Elements of Statistical Learning*. Springer-Verlag Inc, New York, 2001.

D. Hosmer and S. Lemeshow. *Applied Logistic Regression (2nd ed.)*. John Wiley Sons Inc, New York, 2000.

X. Huang, L. Shi, and J. Suykens. Support vector machine classifier with pinball loss. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(5):984–997, 2014.

W. Jiang. Process consistency for adaboost. *Annals of Statistics*, 32:13–29, 2004.

R. Jin, Y. Liu, L. Si, J. Carbonell, and A.G. Hauptmann. A new boosting algorithm using input-dependent regularizer. In *Proceedings of Twentieth International Conference on Machine Learning*, 2003.

C. Leistner, A. Saffari, P.M. Roth, and H. Bischof. On robustness of on-line boosting - a competitive study. In *IEEE International Conference on Computer Vision and Pattern Recognition Workshop on On-line Computer Vision*, 2009.

M. Liu and B.C. Vemuri. Robust and Efficient Regularized Boosting Using Total Bregman Divergence. In *IEEE Proceedings of the 24th Conference on Computer Vision and Pattern Recognition*, pages 2897–2902, 2011.

A.C. Lozano, S.R. Kulkarni, and R.E. Schapire. Convergence and consistency of regularized boosting algorithms with stationary β-mixing observations. In *Advances in Neural Information Processing Systems*, volume 18, pages 819–826. MIT Press, 2006.

A.C. Lozano, S.R. Kulkarni, and R.E. Schapire. Convergence and consistency of regularized boosting with weakly dependent observations. *IEEE Transactions on Information Theory*, 60(1):651–660, 2014.

G. Lugosi and N. Vayatis. On the bayes-risk consistency of regularized boosting methods. *Annals of Statistics*, 32:30–55, 2004.

R. Maclin and D. Opitz. An empirical evaluation of bagging and boosting. In *In Proceedings of the Fourteenth National Conference on Artificial Intelligence*, pages 546–551. AAAI Press, 1997.

H. Masnadi-Shirazi and N. Vasconcelos. On the design of loss functions for classification: theory, robustness to outliers, and savageboost. In *Advances in Neural Information Processing Systems*, pages 1049–1056. MIT Press, 2008.

H. Masnadi-Shirazi and N. Vasconcelos. Cost-sensitive boosting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33:294–309, 2011.

L. Mason, J. Baxter, P. Bartlett, and M. Frean. Boosting Algorithms as Gradient Descent. In *Advances in Neural Information Processing Systems*, pages 512–518. MIT Press, 2000.

R. McDonald, D. Hand, and I. Eckley. An empirical comparison of three boosting algorithms on real data sets with artificial class noise. In *International Workshop on Multiple Classifier Systems*, 2003.

D. Mease and A.J. Wyner. Evidence contrary to the statistical view of boosting. *Journal of Machine Learning Research*, 9:131–156, 2008.

J.M. Moguerza and A. Munoz. Support vector machines with applications. *Statistical Science*, 21:322–336, 2006.

A. Natekin and A. Knoll. Gradient boosting machines, a tutorial. *Frontiers in Neurorobotics*, 7(21), 2013.

A. Niculescu-Mizil and R. Caruana. Obtaining calibrated probabilities from boosting. In *Uncertainty in Artificial Intelligence*, pages 413–419, 2005.

J. Platt. Probabilistic outputs for support vector machines and comparison to regularized likelihood methods. In *Advances in Large Margin Classifiers*, pages 61–74, 2000.

G. Raskutti, M.J. Wainwright, and B. Yu. Early stopping and non-parametric regression: An optimal data-dependent stopping rule. *Journal of Machine Learning Research*, 15: 335–366, 2014.

B. Rasolzadeh, L. Petersson, and N. Pettersson. Response binning: Improved weak classifiers for boosting. In *IEEE Intelligent Vehicle Symposium*, 2006.

M. Reid and R. Williamson. Composite binary losses. *Journal of Machine Learning Research*, 11:2387–2422, 2010.

R. Rifkin and A. Klautau. In defense of one-vs-all classification. *Journal of Machine Learning Research*, 5:101–141, 2004.

S. Rosset, J. Zhu, T. Hastie, and R. Schapire. Boosting as a regularized path to a maximum margin classifier. *Journal of Machine Learning Research*, 5:941–973, 2004.

B. Saha, G. Kunapuli, N. Ray, J. Maldjian, and S. Natarajan. Ar-boost: Reducing overfitting by a robust data-driven regularization strategy. In *European Conference on Machine Learning*, pages 1–16, 2013.

L.J. Savage. The elicitation of personal probabilities and expectations. *Journal of the American Statistical Association*, 66:783–801, 1971.

R. E. Schapire, Y. Freund, P. Bartlett, and W. S. Lee. Boosting the margin: A new explanation for the effectiveness of voting methods. *Annals of Statistics*, 26:1651–1686, 1998.

R.E. Schapire and Y. Singer. Boostexter: A boosting-based system for text categorization. *Machine Learning*, 39:135–168, 2000.

Y. Shiraishi and K. Fukumizu. Statistical approaches to combining binary classifiers for multi-class classification. *Neurocomputing*, 74(5):680–688, 2011.

V.N. Vapnik. *Statistical Learning Theory*. John Wiley Sons Inc, 1998.

P. Viola and M. Jones. Robust real-time face detection. *International Journal Computer Vision*, 57:137–154, 2004.

P. Viola, M. Jones, and D. Snow. Detecting pedestrians using patterns of motion and appearance. In *Ninth IEEE International Conference on Computer Vision*, volume 2, pages 734–741, 2003.

B. Wu and R. Nevatia. Simultaneous object detection and segmentation by boosting local shape feature based classifier. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 18–23, 2007.

B. Wu, H. Ai, C. Huang, and S. Lao. Fast rotation invariant multi-view face detection based on real adaboost. In *Sixth IEEE International Conference on Automatic Face and Gesture Recognition*, pages 79–84, 2004.

Y. Xi, Z. Xiang, P. Ramadge, and R. Schapire. Speed and sparsity of regularized boosting. *Journal of Machine Learning Research*, 5:615–622, 2009.

Z. Xiang, Y. Xi, U. Hasson, and P. Ramadge. Boosting with spatial regularization. In *Advances in Neural Information Processing Systems*, pages 2107–2115. MIT Press, 2009.

B. Zadrozny. Reducing multiclass to binary by coupling probability estimates. In *In Proc. Neural Information Processing Systems*, 2001.

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

T. Zhang and B. Yu. Boosting with early stopping: Convergence and consistency. *Annals of Statistics*, 33:1538–1579, 2005.

J. Zhu and T. Hastie. Kernel logistic regression and the import vector machine. In *Journal of Computational and Graphical Statistics*, pages 1081–1088. MIT Press, 2001.