

Linear Classification

Here we consider the problem of learning from binary labels. As before, we assume a space \mathcal{X} of objects and a feature map $\Phi : \mathcal{X} \rightarrow R^D$ where D is the number of features. We will assume that $\Phi(x)$ is homogeneous, i.e., $\Phi_1(x) = 1$ for all x . We assume training data $D = \langle x_1, y_1 \rangle, \dots, \langle x_N, y_N \rangle$ with y_t being one of the two values 1 or -1 . We will assume that these training pairs have been drawn independently from a distribution (or density) ρ . Our objective is to construct a predictor for y given x which will work well for a new pair drawn from ρ .

1 Training Linear Predictors for Binary Data

Here we consider linear predictors.

$$f_w(x) = w \cdot \Phi(x) \tag{1}$$

We usually interpret $f_w(x)$ as being related to a learned estimate of the probability of y given x . The relationship between $f_w(x)$ and $P_w(y|x)$ may be subtle but generally if $f_w(x) = 0$ then $P(y = 1|x)$ is estimated to be $1/2$ and $P(y = -1|x)$ is assumed to be some monotonic function of $f_w(x)$. If we are forced to predict just one value for y then we can minimize the number of errors by predicting y to be $\text{sign}(f_w(x))$. But if the cost of a false positive is different from the cost of a false negative then we will want to use a decision rule of the form $\text{sign}(f_w(x) - b)$. Higher values of b will lead to a reduction of false positives but an increase in false negatives. For cancer screening we want to catch the cancers so we want a small number of false negatives. For a cheap cancer test we don't mind if most of the detected positives are actually false positives — for each positive screening outcome one can follow-up with a more accurate (but more expensive) test such as a biopsy.

We are interested in learning a parameter vector w from the given training data. We first define the margin $m_t(w)$ as follows.

$$m_t(w) = y_t f_w(x) \tag{2}$$

The parameter vector w will usually be clear from context and we will usually write m_t instead of $m_t(w)$. Note that we have $m_t > 0$ if and only if $y_t = \text{sign}(f_w(x_t))$. If $m_t > 0$ then m_t is the “margin of safety” by which the prediction $\text{sign}(f_w(x_t))$ is correct. If $m_t < 0$ then m_t is a measure of the margin by which $f_w(x_t)$ is wrong.

Here we will consider learning algorithms which set w from the training data using the following formula where L and R are functions that are different in different learning algorithms.

$$w^* = \underset{w}{\operatorname{argmin}} \sum_{t=1}^N L(m_t(w)) + \lambda R(w) \tag{3}$$

In practice the value of λ is tuned so as to maximize the empiric performance of the resulting value of w^* on holdout data (holdout data is labeled data not used as part of the training data). The function L in (3) is called the loss function and the function R of w is called the regularization function and λ is called the regularization parameter. We will consider five possible loss functions and three possible regularizers.

2 The Convexity-Consistency Trade-Off

The loss functions and regularizers can be classified into convex/nonconvex and the loss functions can be classified into consistent/inconsistent. If we have both a convex loss function and a convex regularizer then the optimization problem in (3) is computationally tractable. For the nonconvex cases we consider, the optimization problem in (3) is NP-hard, although in general nonconvexity does not imply NP-hardness.

We also analyze the various loss functions with respect to consistency. Here we focus on consistency for classification error rate and for D finite. In particular, to minimize the geeraization classification error rate would like to find $w_{0,1}^*$ defined as follows.

$$w_{0,1}^* = \operatorname{argmin}_{w \in R^D, \|w\|_2=1} \mathbb{P}_{\langle x, y \rangle \sim \rho} [\operatorname{sign}(f_w(x)) \neq y] \quad (4)$$

We consider the question of whether we are interested in whether (3) yields a consistent estimator of $w_{0,1}^*$, i.e., whether in the limit of infinite training data we get that the direction of w^* converges to the direction of $w_{0,1}^*$. It turns out that no convex loss function can yield a consistent estimator of $w_{0,1}^*$ (for D finite). This is because minimizing 0-1 loss can be shown to be NP-hard while convex optimization can be done in polynomial time in general. So there is a fundamental trade-off between convexity and consistency.

There is a significant literature on consistency for $D = \infty$. The $D = \infty$ case will be defined when we discuss kernels and Hilbert spaces. For $D = \infty$, if we use Hinge loss with L_2 regularization, and an appropriate schedule for selecting λ as a function of N , then we can be both convex and consistent. For $D = \infty$ one can get around NP-hardness by assuming that $f(x)$ can be set arbitrarily for each x independently. This assumption that is asymptotically valid for $D = \infty$. But this assumption is extremely unrealistic in practice. In practice we have to predict on objects that are different from any object we have trained on. In practice there is always a finite “effective dimension”. For this reason, consistency for finite D is more meaningful than consistency for $D = \infty$. For finite D we cannot have both consistency and convexity.

Here consistency has interpreted with respect to estimating $w_{0,1}^*$. But we may be interested estimating other weight vectors — weight vectors that optimize some other notion of loss. But any realistic loss function is bounded —

in practice the cost of a bad prediction in a fielded system is bounded. Any nonconstant bounded function is nonconvex. For any realistic (bounded) loss we will have a consistency-convexity trade-off.

3 Regularizers

The three regularization functions we consider are the following.

$$L_2(w) = \frac{1}{2} \|w\|^2 \tag{5}$$

$$L_1(w) = \sum_{i=1}^D |w_i| \tag{6}$$

$$L_0(w) = |\{i : w_i \neq 0\}| \tag{7}$$

More generally we can define a p -norm as follows.

$$\|w\|_p = \left(\sum_{i=1}^D |w_i|^p \right)^{1/p} \tag{8}$$

Note that for $p = 2$ we get the usual notion of the length (norm) of w . For $p = 1$ we get L_1 . We also have the following.

$$L_0(w) = \lim_{p \rightarrow 0} \|w\|_p \tag{9}$$

$\|w\|_p$ is convex in w for $p \geq 1$ but is nonconvex for $p < 1$. So we have that L_0 is nonconvex. In practice $\|w\|_p$ is rarely used for p not equal to one of 0, 1, or 2.

Under L_1 regularization the marginal cost of increasing $|w_i|$ is significant even when w_i is very near zero. This results in optimal vectors w^* that are sparse — many weights are zero. Sparseness helps identify which weights are important in predicting the data.

4 0-1 Loss

0-1 loss is defined as follows.

$$L_{01}(m) = \begin{cases} 0 & \text{if } m > 0 \\ 1 & \text{otherwise} \end{cases} \tag{10}$$

In some sense 0-1 loss is the most natural loss function if one cares primarily about the misclassification rate. However 0-1 loss is almost never used in (3).

There are two reasons for this. First, when using 0-1 loss in (3) one must either not regularize, which is ok for $D \ll N$, or use L_0 regularization. One cannot use $\|w\|_p$ regularization with $p > 0$ because this has the property that the regularization cost goes to zero as we scale down w . But for any $\epsilon > 0$ we have that ϵw has the same 0-1 loss as w . So with $\|w\|_p$ regularization, the regularization cost can always be driven to zero without changing the 0-1 loss. When using 0-1 loss it is natural to require that $\|w\| = 1$ (since the empirical loss term does not depend on scaling w). A second problem with 0-1 loss is that both it, and the required L_0 regularization, fail to be convex which makes the optimization problem in (3) more difficult.

Although 0-1 loss is rarely used in practice, it is important to note that 0-1 loss is consistent for finite D in the sense that as the training sample size goes to infinity the direction of w^* will converge on the direction of $w_{0,1}^*$.

5 Sigmoidal Loss

Sigmoidal loss generally refers to a continuous loss function L_s with $L_s(m) \approx 1$ for $m \ll -1$, $L_s(m) \approx 0$ for $m \gg 1$, and $L_s(0) = 1/2$. The following forms are commonly used.

$$L_s(m) = \begin{cases} 0 & \text{if } m \geq 1 \\ (1 - m)/2 & \text{if } -1 \leq m \leq 1 \\ 1 & \text{if } m \leq -1 \end{cases} \quad (11)$$

$$L'_s(m) = \frac{1}{1 + e^m} \quad (12)$$

$$L_{\text{probit}S}(m) = \int_m^\infty \mathcal{N}(0, 1)(x) dx \quad (13)$$

Sigmoidal loss is similar to 0-1 loss except that it can be meaningfully used with $\|w\|_p$ regularizers, and in particular with L_1 and L_2 regularization. Note that for ϵ sufficiently small we have that ϵw yields a sigmoidal loss close to $1/2$. This loss can be smaller if w is larger. So L_1 and L_2 regularization is meaningful. For this reason sigmoidal loss has been more widely used in (3) than has 0-1 loss. There seems to be little difference in practice between the three above variants of sigmoidal loss. But sigmoidal loss is nonconvex. For this reason it remains less popular than the convex loss functions described below.

Like 0-1 loss, sigmoidal loss is consistent for finite D — in the limit of infinite data the direction of w^* as defined by (3) converges to the direction of $w_{0,1}^*$.

We will see in later sections that the theoretical analysis of generalization provides direct support for the use of sigmoidal loss and, in particular, $L_{\text{probit}S}$ as defined above.

6 Quadratic Loss

For quadratic loss (also called L_2 loss) we define $L_2(m)$ as follows.

$$\begin{aligned} L_2(m) &= (m - 1)^2 \\ &= (yf_w(x) - y^2)^2 \\ &= y^2(f_w(x) - y)^2 \\ &= (f_w(x) - y)^2 \end{aligned} \tag{14}$$

So the general equation (3), under quadratic loss (14), is the same as regularized least-squares regression. Least-squares regression is, of course, widely used. Consider the following ideal function f^* where f is allowed to be any function of \mathcal{X} .

$$f^* = \operatorname{argmin}_f \mathbb{E}_{\langle x, y \rangle \sim \rho} [L_2(yf(x))] \tag{15}$$

$$f^*(x) = \mathbb{E}[y|x] = \mathbb{E}_{y \sim \rho(\cdot|x)} [y] \tag{16}$$

Under L_2 loss will can think of $f_{w^*}(x)$ as an estimate of $\mathbb{E}[y|x]$.

Now consider w_2^* defined as follows (for D finite).

$$w_2^* = \operatorname{argmin}_w \mathbb{E}_{\langle x, y \rangle \sim \rho} [(f_w(x) - y)^2] \tag{17}$$

In general we have w_2^* is different from w_{01}^* and quadratic loss, while convex, does not yield a consistent estimator of w_{01}^* .

7 Log Loss

Log loss is the following loss function.

$$\begin{aligned} L_{\log}(m) &= \ln(1 + \exp(-m)) \\ &= \ln(1/P_w(y|x)) \end{aligned} \tag{18}$$

$$P_w(y|x) = \frac{1}{Z} \exp\left(\frac{1}{2}y(w \cdot \Phi(x))\right) \tag{19}$$

$$\begin{aligned} P_w(y_t|x_t) &= \frac{\exp(\frac{1}{2}m_t)}{\exp(\frac{1}{2}m_t) + \exp(-\frac{1}{2}m_t)} \\ &= \frac{1}{1 + \exp(-m_t)} \end{aligned}$$

It is important to note that for $m \gg 1$ we have that $L_{\log}(m) \approx 0$ and for $m \ll -1$ we have that $L_{\log}(m) \approx -m$. This will be important in motivating the hinge loss described below.

$$f^* = \operatorname{argmin}_f \mathbb{E}_{\langle x, y \rangle \sim \rho} [L_{\log}(yf(x))] \tag{20}$$

$$f^*(x) = \ln P(y = 1|x) - \ln P(y = -1|x) \tag{21}$$

$$P(y = 1|x) = \frac{1}{1 + e^{-f^*(x)}} \tag{22}$$

Now consider the optimal weight vector defined by log loss.

$$w_{\log}^* = \operatorname{argmin}_w \mathbb{E}_{\langle x, y \rangle \sim \rho} [L_{\log}(yf_w(x))] \tag{23}$$

Log loss is convex but does not yield a consistent estimator of w_{01}^* .

Both log loss and quadratic loss can be viewed as an attempt to estimate $P(y|x)$, which is arguably more important than estimating the direction of w_{01}^* . But both log loss and quadratic loss are unbounded while in practice the cost of an error will be finite. Log loss is much more forgiving of bad predictions and is less sensitive to outliers in the data, i.e., to training points where the loss of w^* is high. Log loss is generally preferred to quadratic loss for $y \in \{-1, 1\}$.

8 Hinge Loss

Hinge Loss is the following.

$$L_{\text{hinge}}(m) = \max(0, 1 - m) \quad (24)$$

Hinge loss can be viewed as a piecewise linear approximation of log loss. Both hinge loss and log loss have the properties that $L(m) \approx 0$ for $m \gg 1$ and $\partial L/\partial m \approx -1$ for $m \ll -1$.

We can understand hinge loss better by first defining f_H^* as follows.

$$f_H^* = \lim_{\lambda \rightarrow 0} \operatorname{argmin}_f \mathbb{E}_{\langle x, y \rangle \sim \rho} [L_{\text{hinge}}(yf(x))] + \lambda \|f\|^2 \quad (25)$$

$$\|f\|^2 = \mathbb{E}_{x \sim \rho} [f^2(x)] \quad (26)$$

One can show that f_H^* is the following.

$$f_H^*(x) = \begin{cases} 1 & \text{if } P(y = 1|x) > 1/2 \\ 0 & \text{if } P(y = 1|x) = 1/2 \\ -1 & \text{if } P(y = 1|x) < 1/2 \end{cases} \quad (27)$$

The function $f_H^*(x)$ is the infinite training data limit in the $D = \infty$ case of a Hilbert space. But for finite D (or for a finite training sample) it will be difficult to approximate f_H^* with a function of the form f_w (or a function that generalizes well). So it also seems insightful to consider the following in the case of finite D .

$$w_H^* = \operatorname{argmin}_w \mathbb{E}_{\langle x, y \rangle \sim \rho} [L_{\text{hinge}}(yf_w(x))] \quad (28)$$

Hinge loss is convex and hence w_H^* is not a consistent estimator of the direction of w_{01}^* for finite D . The vector w_H^* is also different in general form w_{\log}^* , although the difference between hinge loss and log loss seems minor in practice. The optimization problem (3) seems more computationally tractable for hinge loss than for log loss.

9 Common Combinations

I will use the term L - R regression to mean a use of (3) with loss function L and regularizer R . The following combinations of loss functions and regularizers are commonly used.

- $L_{\text{probit}}-L_2$ is called probit regression.
- L_2-L_2 is called ridge regression.
- L_2-L_1 is called lasso.
- $L_{\log}-L_2$ is called logistic regression.
- $L_{\text{hinge}}-L_2$ is called a support vector machine (SVM).

10 Rescaling the Loss Function

We now consider the case where we are interested in the error rate of the learned vector w^* on new data. As mentioned above, to use regularization we cannot directly use 0-1 loss in the training formula (3). Therefore, even though we are ultimately interested in generalization performance as measured by 0-1 loss, the loss L used in (3) must be some other loss such as quadratic loss, log loss, sigmoidal loss or hinge loss.

Consider equation (3) for an arbitrary loss function L . Define L' in terms of $\alpha, \gamma > 0$ as follows.

$$L'(m) = \alpha L(\gamma m)$$

The loss function L' is a rescaling of L involving both an arbitrary rescaling of the margin and an arbitrary rescaling of the loss quantity. It is possible to show that when λ is tuned with holdout data, the rescaled loss L' performs exactly like L . This can be seen as follows for arbitrary $\|w\|_p$ regularization.

$$\begin{aligned} w^* &= \operatorname{argmin}_w \sum_{t=1}^N L(m_t(w)) + \lambda \|w\|_p \\ &= \operatorname{argmin}_w \sum_{t=1}^N \alpha L(m_t(w)) + \alpha \lambda \|w\|_p \\ &= \operatorname{argmin}_w \sum_{t=1}^N \alpha L(\gamma m_t(\frac{w}{\gamma})) + \alpha \lambda \|w\|_p \\ w'^* &= \operatorname{argmin}_{w'} \sum_{t=1}^N \alpha L(\gamma m_t(w')) + \alpha \gamma \lambda \|w'\|_p \\ &= \operatorname{argmin}_w \sum_{t=1}^T L'(m_t) + \lambda' \|w\|_p \\ w^* &= \gamma w'^* \end{aligned}$$

Because w^* and w'^* are in the same direction, they give the same 0 – 1 loss. Hence training with L' and λ' will result in exactly the same 0-1 holdout performance as training with L and λ . This suggests that the exact choice of loss function when optimizing 0-1 holdout loss is not very critical. Two loss functions can be made to look similar when we can rescale both the margin value and the loss value arbitrarily. It also implies that there is no performance advantage in generalizing Hinge loss to be of the form $\max(0, \alpha - \gamma m)$ for parameters α and γ or in generalizing sigmoidal loss to allow sigmoids of different widths.

11 problems

Problem 1. This problem is on quadratic discriminant analysis. Consider a sample of pairs $\langle x_1, y_1 \rangle, \dots, \langle x_T, y_T \rangle$ with $\Phi(x_t) \in R^2$ and $y_t \in \{-1, 1\}$. Let $\Phi(x)$ be the pair of numbers $\langle f_1(x), f_2(x) \rangle$. Let S^+ and S^- be defined as follows.

$$\begin{aligned} S^+ &= \{t : 1 \leq t \leq T \wedge y_t = 1\} \\ S^- &= \{t : 1 \leq t \leq T \wedge y_t = -1\} \end{aligned}$$

Now consider the following quantities.

$$\begin{aligned} \mu_1^+ &= \frac{1}{|S^+|} \sum_{t \in S^+} f_1(x_t) \\ \mu_2^+ &= \frac{1}{|S^+|} \sum_{t \in S^+} f_2(x_t) \\ \Sigma_{i,j}^+ &= \frac{1}{|S^+|} \sum_{t \in S^+} (f_i(x_t) - \mu_i^+)(f_j(x_t) - \mu_j^+) \\ \mu_1^- &= \frac{1}{|S^-|} \sum_{t \in S^-} f_1(x_t) \\ \mu_2^- &= \frac{1}{|S^-|} \sum_{t \in S^-} f_2(x_t) \\ \Sigma_{i,j}^- &= \frac{1}{|S^-|} \sum_{t \in S^-} (f_i(x_t) - \mu_i^-)(f_j(x_t) - \mu_j^-) \end{aligned}$$

We let μ^+ be the vector $\langle \mu_1^+, \mu_2^+ \rangle$ and similarly for μ^- . Suppose that we model $P(y = 1)$ as $|S^+|/T$; model the conditional the conditional probability $P(\Phi(x)|y = 1)$ as a Gaussian with mean μ^+ and covariance Σ^+ ; and model $P(\Phi(x)|y = -1)$ similarly with mean μ^- and covariance Σ^- . Now suppose that we have a particular sample satisfying the following.

$$\begin{aligned} |S^+| &= \frac{1}{2}T \\ \mu^+ &= \langle 0, 0 \rangle \\ \mu^- &= \langle 1, 0 \rangle \\ \langle z, w \rangle (\Sigma^+)^{-1} \langle z, w \rangle &= 2z^2 + w^2 \\ \langle z, w \rangle (\Sigma^-)^{-1} \langle z, w \rangle &= z^2 + 2w^2 \end{aligned}$$

Write the condition $P(y = 1|\Phi(x)) \geq P(y = -1|\Phi(x))$ as a quadratic condition on $f_1(x)$ and $f_2(x)$. Draw the decision boundy and label the region in which we have $P(y = 1|\Phi(x)) \geq P(y = -1|\Phi(x))$.

Problem 2. Again consider the sample from problem 1. Define S^+ and S^- as in problem 1.

a. Define a new feature space $\Psi(x) \in R^d$ (for some d) such that the quadratic decision boundary of problem 1 can be written as $\beta_D \cdot \Psi(x) \geq 0$. Give both the feature map Ψ and the vector β_D .

b. Define β^* as follows where Ψ is defined as in part a.

$$\beta^* = \operatorname{argmax}_{\beta} \sum_{t=1}^T \operatorname{sign}(y_t(\beta^T \Psi(x_t)))$$

Give an argument (physicists proof) that in the limit as $T \rightarrow \infty$ we should have that β^* is at least as good as β_D and usually better.

Problem 3. (Understanding L1 and L2 norms) The L2 norm of a vector $x \in \mathcal{R}^n$ is defined as $\|x\|_2 = \sqrt{\sum_i x_i^2}$ and the L1 norm is $\|x\|_1 = \sum_i |x_i|$.

a. For $x, y \in \mathcal{R}^n$, use the Cauchy-Schwarz inequality, which states that $x \cdot y \leq \|x\|_2 \|y\|_2$, to show that:

$$\|x\|_2 \leq \|x\|_1 \leq \sqrt{n} \|x\|_2$$

b. If x lives in the unit sphere (i.e. $\|x\|_2 = 1$), show that both the upper and lower bounds are achievable with points in the sphere. Namely, show there exists an x in the sphere such that $\|x\|_1 = 1$ and that there exists an x in the sphere where $\|x\|_1 = \sqrt{n}$. Hence, the L1 diameter of the sphere can be quite large, \sqrt{n} .

c. If x lives in the simplex (meaning that all the coordinates are positive and sum to one, i.e. it is a probability distribution), again show that the upper and lower bounds are achievable with points in the simplex. Hence, this implies the L2 diameter of the simplex is bounded by 1.