Squared Loss

The popular choice is to minimize the sum of squares of the residuals $f(x) - y$. As we shall see in Section 3.3.1, this corresponds to the assumption that we have additive normal noise corrupting the observations y_i. Consequently we minimize

$$c(x, y, f(x)) = (f(x) - y)^2 \text{ or equivalently } \tilde{c}(\xi) = \xi^2. \tag{3.8}$$

For convenience of subsequent notation, $\frac{1}{2}\xi^2$ rather than ξ^2 is often used.

ε-insensitive Loss and ℓ_1 Loss

An extension of the soft margin loss (3.3) to regression is the *ε-insensitive* loss function [561, 572, 562]. It is obtained by symmetrization of the "hinge" of (3.3),

$$\tilde{c}(\xi) = \max(|\xi| - \varepsilon, 0) =: |\xi|_\varepsilon. \tag{3.9}$$

The idea behind (3.9) is that deviations up to ε should not be penalized, and all further deviations should incur only a linear penalty. Setting $\varepsilon = 0$ leads to an ℓ_1 loss, i.e., to minimization of the sum of absolute deviations. This is written

$$\tilde{c}(\xi) = |\xi|. \tag{3.10}$$

We will study these functions in more detail in Section 3.4.2.

Practical Considerations

For efficient implementations of learning procedures, it is crucial that loss functions satisfy certain properties. In particular, they should be cheap to compute, have a small number of discontinuities (if any) in the first derivative, and be convex in order to ensure the uniqueness of the solution (see Chapter 6 and also Problem 3.6 for details). Moreover, we may want to obtain solutions that are computationally efficient, which may disregard a certain number of training points. This leads to conditions such as vanishing derivatives for a range of function values $f(x)$. Finally, requirements such as outlier resistance are also important for the construction of estimators.

3.2 Test Error and Expected Risk

Now that we have determined how errors should be penalized on specific instances $(x, y, f(x))$, we have to find a method to combine these (local) penalties. This will help us to assess a particular estimate f.

In the following, we will assume that there exists a probability distribution $\mathrm{P}(x, y)$ on $\mathcal{X} \times \mathcal{Y}$ which governs the data generation and underlying functional dependency. Moreover, we denote by $\mathrm{P}(y|x)$ the *conditional* distribution of y given x, and by $d\mathrm{P}(x, y)$ and $d\mathrm{P}(y|x)$ the integrals with respect to the distributions $\mathrm{P}(x, y)$ and $\mathrm{P}(y|x)$ respectively (cf. Section B.1.3).

3.2.1 Exact Quantities

Unless stated otherwise, we assume that the data (x, y) are drawn iid (independent and identically distributed, see Section B.1) from $\mathrm{P}(x, y)$. Whether or not we have

knowledge of the test patterns at training time[4] makes a significant difference in the design of learning algorithms. In the latter case, we will want to minimize the *test error* on that *specific* test set; in the former case, the *expected* error over *all possible* test sets.

Definition 3.2 (Test Error) *Assume that we are not only given the* training *data* $\{x_1, \dots, x_m\}$ *along with target values* $\{y_1, \dots y_m\}$ *but also the* test *patterns* $\{x'_1, \dots x'_{m'}\}$ *on which we would like to predict* y'_i $(i = 1, \dots, m')$*. Since we already know* x'_i*, all we should care about is to minimize the* expected error *on the* test set*. We formalize this in the following definition*

Transduction Problem

$$R_{\text{test}}[f] := \frac{1}{m'} \sum_{i=1}^{m'} \int_{\mathcal{Y}} c(x'_i, y, f(x'_i)) d\mathrm{P}(y|x'_i). \tag{3.11}$$

Unfortunately, this problem, referred to as *transduction*, is quite difficult to address, both computationally and conceptually, see [562, 267, 37, 211]. Instead, one typically considers the case where no knowledge about test patterns is available, as described in the following definition.

Definition 3.3 (Expected Risk) *If we have no knowledge about the test patterns (or decide to ignore them) we should minimize the expected error over all possible training patterns. Hence we have to minimize the expected loss with respect to* P *and c*

$$R[f] := \mathbf{E}\left[R_{\text{test}}[f]\right] = \mathbf{E}\left[c(x, y, f(x))\right] = \int_{\mathcal{X}\times\mathcal{Y}} c(x, y, f(x)) d\mathrm{P}(x, y). \tag{3.12}$$

Here the integration is carried out with respect to the distribution $\mathrm{P}(x, y)$. Again, just as (3.11), this problem is intractable, since we do not know $\mathrm{P}(x, y)$ explicitly. Instead, we are only given the training patterns (x_i, y_i). The latter, however, allow us to replace the unknown distribution $\mathrm{P}(x, y)$ by its empirical estimate.

To study connections between loss functions and density models, it will be convenient to assume that there exists a density $p(x, y)$ corresponding to $\mathrm{P}(x, y)$. This means that we may replace $\int d\mathrm{P}(x, y)$ by $\int p(x, y) dx dy$ and the appropriate measure on $\mathcal{X} \times \mathcal{Y}$. Such a density $p(x, y)$ need not always exist (see Section B.1 for more details) but we will not give further heed to these concerns at present.

3.2.2 Approximations

Unfortunately, this change in notation did not solve the problem. All we have at our disposal is the actual training data. What one usually does is replace $p(x, y)$ by the *empirical density*

Empirical Density

$$p_{\text{emp}}(x, y) := \frac{1}{m} \sum_{i=1}^{m} \delta_{x_i}(x) \delta_{y_i}(y). \tag{3.13}$$

4. The test *outputs*, however, are not available during training.

Here $\delta_{x'}(x)$ denotes the δ-distribution, satisfying $\int \delta_{x'}(x) f(x) dx = f(x')$. The hope is that replacing p by p_{emp} will lead to a quantity that is "reasonably close" to the expected risk. This will be the case if the class of possible solutions f is sufficiently limited [568, 571]. The issue of closeness with regard to different estimators will be discussed in further detail in Chapters 5 and 12. Substituting $p_{\text{emp}}(x, y)$ into (3.12) leads to the empirical risk:

Definition 3.4 (Empirical Risk) *The empirical risk is defined as*

$$R_{\text{emp}}[f] := \int_{\mathcal{X}\times\mathcal{Y}} c(x, y, f(x)) p_{\text{emp}}(x, y) dx dy = \frac{1}{m} \sum_{i=1}^{m} c(x_i, y_i, f(x_i)). \tag{3.14}$$

M-Estimator

This quantity has the advantage that, given the training data, we can readily compute and also minimize it. This constitutes a particular case of what is called an *M-estimator* in statistics. Estimators of this type are studied in detail in the field of empirical processes [554]. As pointed out in Section 3.1, it is crucial to understand that although our particular M-estimator is built from minimizing a loss, this need not always be the case. From a decision-theoretic point of view, the question of which loss to choose is a separate issue, which is dictated by the problem at hand as well as the goal of trying to evaluate the performance of estimation methods, rather than by the problem of trying to define a particular estimation method [582, 166, 43].

Ill-Posed Problems

These considerations aside, it may appear as if (3.14) is the answer to our problems, and all that remains to be done is to find a suitable class of functions $\mathcal{F} \ni f$ such that we can minimize $R_{\text{emp}}[f]$ with respect to $\mathcal{F}$. Unfortunately, determining $\mathcal{F}$ is quite difficult (see Chapters 5 and 12 for details). Moreover, the minimization of $R_{\text{emp}}[f]$ can lead to an ill-posed problem [538, 370]. We will show this with a simple example.

Example of an Ill-Posed Problem

Assume that we want to solve a regression problem using the quadratic loss function (3.8) given by $c(x, y, f(x) = (y - f(x))^2$. Moreover, assume that we are dealing with a linear class of functions,[5] say

$$\mathcal{F} := \left\{ f \,\middle|\, f(x) = \sum_{i=1}^{n} \alpha_i f_i(x) \text{ with } \alpha_i \in \mathbb{R} \right\}, \tag{3.15}$$

where the f_i are functions mapping $\mathcal{X}$ to $\mathbb{R}$.

We want to find the minimizer of R_{emp}, i.e.,

$$\underset{f\in\mathcal{F}}{\text{minimize}}\, R_{\text{emp}}[f] = \underset{\boldsymbol{\alpha}\in\mathbb{R}^n}{\text{minimize}} \frac{1}{m} \sum_{i=1}^{m} \left(y_i - \sum_{j=1}^{n} \alpha_j f_j(x_i) \right)^2. \tag{3.16}$$

5. In the simplest case, assuming $\mathcal{X}$ is contained in a vector space, these could be functions that extract coordinates of x; in other words, $\mathcal{F}$ would be the class of linear functions on $\mathcal{X}$.

Computing the derivative of $R_{\text{emp}}[f]$ with respect to $\boldsymbol{\alpha}$ and defining $F_{ij} := f_i(x_j)$, we can see that the minimum of (3.16) is achieved if

$$F^\top \mathbf{y} = F^\top F \boldsymbol{\alpha}. \tag{3.17}$$

A sufficient condition for (3.17) is $\boldsymbol{\alpha} = \left(F^\top F\right)^{-1} F^\top \mathbf{y}$ where $\left(F^\top F\right)^{-1}$ denotes the (pseudo-)inverse of the matrix.

Condition of a Matrix

If $F^\top F$ has a bad condition number (i.e. the quotient between the largest and the smallest eigenvalue of $F^\top F$ is large), it is numerically difficult [423, 530] to solve (3.17) for $\boldsymbol{\alpha}$. Furthermore, if $n > m$, i.e. if we have more basis functions f_i than training patterns x_i, there will exist a subspace of solutions with dimension at least $n - m$, satisfying (3.17). This is undesirable both practically (speed of computation) and theoretically (we would have to deal with a whole class of solutions rather than a single one).

One might also expect that if $\mathcal{F}$ is too rich, the discrepancy between $R_{\text{emp}}[f]$ and $R[f]$ could be large. For instance, if F is an $m \times m$ matrix of full rank, $\mathcal{F}$ contains an f that predicts all target values y_i correctly on the training data. Nevertheless, we cannot expect that we will also obtain zero prediction error on unseen points. Chapter 4 will show how these problems can be overcome by adding a so-called regularization term to $R_{\text{emp}}[f]$.

3.3 A Statistical Perspective

Given a particular pattern $\tilde{x}$, we may want to ask what risk we can expect for it, and with which *probability* the corresponding loss is going to occur. In other words, instead of (or in addition to) $\mathbf{E}\left[c(\tilde{x}, y, f(\tilde{x})\right]$ for a fixed $\tilde{x}$, we may want to know the distribution of y given $\tilde{x}$, i.e., $\mathrm{P}(y|\tilde{x})$.

(Bayesian) statistics (see [338, 432, 49, 43] and also Chapter 16) often attempt to estimate the density corresponding to the random variables (x, y), and in some cases, we may really *need* information about $p(x, y)$ to arrive at the desired conclusions given the training data (e.g., medical diagnosis). However, we always have to keep in mind that if we model the density p first, and subsequently, based on this approximation, compute a minimizer of the expected risk, we will have to make two approximations. This could lead to inferior or at least not easily predictable results. Therefore, wherever possible, we should avoid solving a more general problem, since additional approximation steps might only make the estimates worse [561].

3.3.1 Maximum Likelihood Estimation

All this said, we still may want to compute the conditional density $p(y|x)$. For this purpose we need to model how y is generated, based on some underlying dependency $f(x)$; thus, we specify the functional form of $p(y|x, f(x))$ and maximize