

# ELEN6887

## Lecture 1: A Probabilistic Approach to Pattern Recognition

R. Castro

1/20/2010

In this lecture we introduce the basic elements of statistical pattern recognition and set the stage for the rest of the course. A probabilistic approach provides a good framework to cope with the uncertainty inherent to any data-set.

### 1 Learning from Data

To formulate the basic learning from data problem, we must specify several basic elements: data spaces, probability measures over the data spaces, loss functions, and statistical risk.

#### 1.1 Data Spaces

Learning from data begins with a specification of two spaces:

$\mathcal{X} \equiv$  Input Space

$\mathcal{Y} \equiv$  Output Space

The input space is also sometimes called the “feature space” or “signal domain.” The output space is also called the “label space,” “outcome space,” “response space,” or “signal range.”

##### Example 1

$\mathcal{X} = \mathbf{R}^d$  *d-dimensional Euclidean space of “feature vectors”*

$\mathcal{Y} = \{0, 1\}$  *two classes or “class labels”*

##### Example 2

$\mathcal{X} = \mathbf{R}$  *one-dimensional signal domain (e.g., time-domain)*

$\mathcal{Y} = \mathbf{R}$  *real-valued signal*

*A classic example is estimating a signal  $f$  in noise:*

$$Y = f(X) + W$$

*where  $X$  is a random sample point on the real line and  $W$  is a noise independent of  $X$ .*

The goal of learning is to devise a good prediction rule, such that, given a feature vector  $X \in \mathcal{X}$  we can predict the corresponding label  $Y \in \mathcal{Y}$  accurately. For this we need to characterize how features and labels are related. We do so using a probabilistic approach.

## 1.2 Probability Measure and Expectation

Define a joint probability distribution on  $\mathcal{X} \times \mathcal{Y}$  denoted  $P_{X,Y}$ . Let  $(X, Y)$  denote a pair of random variables distributed according to  $P_{X,Y}$ . We will also have use for marginal and conditional distributions. Let  $P_X$  denote the marginal distribution on  $X$ , and let  $P_{Y|X}$  denote the conditional distribution of  $Y$  given  $X$ . For any distribution  $P$ , let  $p$  denote its density function with respect to the corresponding dominating measure; e.g., *Lebesgue measure* for continuous random variables or *counting measure* for discrete random variables.

Define the expectation operator:

$$E[f(X, Y)] \equiv \int f(x, y) dP_{X,Y}(x, y) = \int f(x, y) p_{X,Y}(x, y) dx dy.$$

## 1.3 Loss Functions

To measure the quality of a prediction rule we need to have a notion of loss or cost. A loss function is a mapping

$$\ell : \mathcal{Y} \times \mathcal{Y} \mapsto \mathbf{R}$$

**Example 3** In binary classification problems,  $\mathcal{Y} = \{0, 1\}$ . The 0/1 loss function is usually used:  $\ell(y_1, y_2) = 1_{y_1 \neq y_2}$ , where  $1_A$  is the indicator function which takes a value of 1 if condition  $A$  is true and zero otherwise. We typically will compare a true label  $y$  with a prediction  $\hat{y}$ , in which case the 0/1 loss simply counts misclassifications.

**Example 4** In regression or estimation problems,  $\mathcal{Y} = \mathbf{R}$ . The squared error loss function is often employed:  $\ell(y_1, y_2) = (y_1 - y_2)^2$ , the square of the difference between  $y_1$  and  $y_2$ . In application, we are interested in a true value  $y$  in comparison to an estimate  $\hat{y}$ .

**Example 5** In some classification problems it is useful to consider asymmetric loss functions. For example when learning a rule to classify email as spam we often prefer to incorrectly label a spam email as legitimate than the other way around. This can be made quite explicit through a proper choice of loss function

$$\ell(y_1, y_2) = \begin{cases} 10 & \text{if } y_1 = 1, y_2 = 0 \\ 2 & \text{if } y_1 = 0, y_2 = 1 \\ 0 & \text{if } y_1 = y_2 \end{cases} .$$

If  $\hat{y}$  is our classification for a particular email (one if it spam and zero otherwise), and  $y$  is the true label, then  $\ell(\hat{y}, y)$  reflects the fact that we prefer to misclassify spam emails than legitimate emails.

## 1.4 Statistical Risk

At this point we have all the main ingredients we need to formally state the goal of a learning procedure. The basic problem in learning is to determine a mapping  $f : \mathcal{X} \mapsto \mathcal{Y}$  that takes an input  $x \in \mathcal{X}$  and predicts the corresponding output  $y \in \mathcal{Y}$ . The performance of a given map  $f$  is measured by its expected loss or *risk*:

$$R(f) \equiv E[\ell(f(X), Y)]$$

The risk tells us how well, on average, the predictor  $f$  performs with respect to the chosen loss function. A key quantity of interest is the minimum risk value, defined as

$$R^* = \inf_f R(f)$$

where the infimum is taking over all measurable functions.

## 1.5 The Learning Problem

Suppose that  $(X, Y)$  are distributed according to  $P_{X,Y}$  ( $(X, Y) \sim P_{X,Y}$  for short). Our goal is to find a map so that  $f(X) \approx Y$  with high probability. Ideally, we would chose  $f$  to minimize the risk  $R(f) = E[\ell(f(X), Y)]$ . However, in order to compute the risk (and hence optimize it) we need to know the joint distribution  $P_{X,Y}$ . In many problems of practical interest, the joint distribution is unknown, and directly minimizing the risk is not possible.

Suppose that we have some “training examples”, that is, samples from the distribution  $P_{X,Y}$ . Specifically, consider  $n$  samples  $X_i, Y_{i=1}^n$  distributed independently and identically (i.i.d.) according to the otherwise unknown  $P_{X,Y}$ . These are called *training data*, and denote the collection by  $D_n \equiv X_i, Y_{i=1}^n$ . Let’s also define a collection of candidate mappings  $\mathcal{F}$ . We will use the training data  $D_n$  to pick a mapping  $\hat{f}_n \in \mathcal{F}$  that we hope will be a good predictor. This is sometimes called the *Model Selection* problem. Note that the selected model  $f_n$  is a function of the training data:

$$\hat{f}_n(X) = f(X; D_n) .$$

The “hat” and the subscript  $n$  make the dependence on the training data explicit avoiding notational clutter. The risk of  $\hat{f}_n$  is given by

$$R(\hat{f}_n) = E[\ell(\hat{f}_n(X), Y)|D_n] = E_{X,Y}[\ell(\hat{f}_n(X), Y)] .$$

Note that  $\hat{f}_n$  is a random variable, and with the definition above the risk of  $R(\hat{f}_n)$  is also a random variable<sup>1</sup>.

For the most part of this course we will be interested in guaranteeing that  $R(\hat{f}_n)$  is small with very high probability over the distribution of the training data (again, recall that  $R(\hat{f}_n)$  is a random variable). Quite frequently we will mostly be interested in the *expected risk*, computed over random realizations of the training data:

$$E[R(\hat{f}_n)] = E_{D_n}[R(\hat{f}_n)] .$$

We hope that  $\hat{f}_n$  produces a small expected risk.

The notion of expected risk can be interpreted as follows. We would like to define an algorithm (a model selection process) that performs well on average, over any random sample of  $n$  training data. The expected risk is a measure of the expected performance of the algorithm with respect to the chosen loss function. That is, we are not gauging the risk of a particular map  $f \in \mathcal{F}$ , but rather we are measuring the performance of the algorithm that takes any realization of training data and selects an appropriate model in  $\mathcal{F}$ .

This course is concerned with determining “good” model spaces  $\mathcal{F}$  and useful and effective model selection algorithms. Perhaps surprisingly one can do this even making only mild assumptions on the distribution  $P_{X,Y}$ .

---

<sup>1</sup>The notation  $E[A|B]$  indicates a conditional expectation, the expectation of  $A$  given  $B$ , therefore this is a random variable, in fact  $E[A|B] = g(B)$  for some function  $g$ .