

## Chapter 3

# Supervised and unsupervised learning - 1

### 3.1 Introduction

The science of learning plays a key role in the field of statistics, data mining, artificial intelligence, intersecting with areas in engineering, finance and industry. Typically, a statistician observes an outcome measurement (quantitative or categorical) and a bunch of input variables and the goal is to predict the outcome for future observations. Based on a training data where we have both the outcome and the input measurements, we try to build a prediction model (known as learner) for new unseen objects. In a supervised learning problem, the presence of such an outcome variable guides the learner. However, in a more challenging scenario, we observe the features only and the outcome is not measured. The task is to find patterns in the data to organize them for future purposes. This will be called unsupervised learning. We consider the following examples to introduce the learning in different fields.

**Example 1 :** The information from 4601 email messages are collected to predict whether the message was a junk email or a spam. The learner is to be built to identify and filter out the spams. The true outcome (email/spam) is available for the training data, as well as the relative frequencies of 57 common words and characters (you / your / free / ! / remove etc.). This words are chosen in such a way that the relative frequencies in the two types differs maximally for them.

**Example 2 :** Stamey et al collected the data to examine the correlation between Prostate Specific Antigen (PSA) as an indicator of Prostate cancer and some clinical measurements in 97 men about to receive a radical prostatectomy. The learner needs to predict the PSA (or its log) from the input measurements, namely cancer-volume, prostate wt, age, benign prostatic hyperplasia amount etc. This is a regression problem.

**Example 3 :** USPS targets to sort out handwritten envelopes using their ZIP codes. The digitized (and standardized) images of the digits 0-9 are collected from numerous envelopes in  $16 \times 16$  greyscale maps, each pixels having intensity range 0-255. The learner targets to identify the digits from the image. (Due to the zero error specification, some digits can be classified as 'not clear' and sorted manually later).

**Example 4 :** A gene expression dataset is collected with expression levels from 64 cancer tumors from different patients. There are 6830 genes in total (a typical HDLSS problem), and the gene expression data is displayed in heat map (green - low, red - high). The goal can be to have a regression problem with the gene expressions as output and the genes and samples as inputs, or an unsupervised learning problem where we want to cluster the samples with 6830 dimensions in a specific way.

### 3.1.1 Supervised learning

Supervised learning is a technique for creating a function for a training data. The training data consists of pairs of input objects (a vector of characteristics) and desired output. The output of the function can be a continuous value (regression) or can predict a class label (classification). The task of the learner is to predict the value of the outcome for any valid input object after having seen a number of training examples. In a global model, the goal is to estimate a function  $g$ , given a set of points  $(\mathbf{x}, g(\mathbf{x}))$ .

The basic problem of supervised learning deals with predicting the response

variables from the independent variables. When the output is quantitative, the problem is known as regression. The categorical variable output will lead us to classification and separation. In essence, the input  $X$  is a collection of  $p$  associated variables, and for each  $X$ , an observed value  $Y$ , of the output, is the supervisor. The goal is to build a learner, guided by the training set based on  $N$  samples of the pair  $(Y, X)$ , so that it can predict the value  $\hat{Y}(x)$  from a future observation  $x$ . In case of regression,  $Y$  is quantitative, whereas classification problems are indicated by discrete values of  $Y$ . However, the classification problems can be seen as regression problems, where  $Y$  takes value in  $k$  dimension vector spaces ( $k$  denote the number of classes). Here,  $Y = (y_1, \dots, y_k)^t$  such that  $y_i = 1$  if the observation falls in  $i$ th class and 0 otherwise. In essence,  $Y$  takes the vector values  $e_i$  accordingly as the observation is in  $i$ th class. The learner can either predict the class  $i$  for a new observation  $x$  or it can churn out a vector  $(p_1, \dots, p_k)(x)$  where  $p_i(x)$  denote the probability of coming from the  $i$ th class. Given such a predictor, the decision can then be to choose the class  $i$  that has the maximum probability  $p_i(x)$ . In this chapter, we will mainly focus on classification techniques, and mention about regression only where required.

### 3.1.2 Unsupervised learning

Unsupervised learning is a method of learning where a model is fit to observations. It is distinguished from supervised learning by the fact that there is no a priori output. A data set of input objects is gathered and the learner treats them as a set of random variables. A joint density model is then built for the data set. Typically one has a set of  $N$  observations  $X_1, \dots, X_N$  having a joint density  $p(X)$ , all random  $p$ -vectors. The goal is to directly infer the properties of this density without the supervising variable. This provides some added difficulty to the characterization. however, it is sometimes an advantage to know that  $X$  represents all the variables under consideration and we don't need to infer how  $p(X)$  will change, conditioning on the changing values of

other variables.

In low dimensions ( $p \leq 3$ ), there are a variety of effective ways to directly estimate the density  $p(X)$  at all  $X$  values, including kernel estimation, spline estimation. Those local non-parametric methods fail in high dimensions due to the curse of dimensionality.

**Curse of dimensionality** : Suppose we look for a local hyper-cubical neighborhood about a target-point to capture a fraction  $r$  of the observations. For simplicity, we consider a  $p$  dimensional unit hypercube and assume that the data points are scattered uniformly in the region. As the required neighborhood need to encase a fraction  $r$  of the unit volume, the expected edge length will be  $e_p(r) = r^{1/p}$ . For  $p = 10$ , To capture only 1% of the volume, the required length is  $e_{10}(.01) = .63$ , about 63% of the entire range of each output. Such neighborhoods are no longer local. Moreover, if we consider  $r$  to be small as well, so that we cover only a small range of each variable, we have fewer and fewer observations to average, and the estimate will have high bias.

Another problem that will arise is that all sample points are close to the edge of the data. Consider  $N$  points uniformly distributed on a unit sphere centered at the origin. The median distance from the origin to the closest data point can be computed as  $(1 - \frac{1}{2}^{1/N})^{1/p}$ . For  $p = 10$ , even if  $N = 500$ , the median distance is .52, i.e. more than half-way to the boundary. Clearly if one wants to estimate the density near the origin, it will have very few observations to rely upon.

Finally, as the sampling density is of the order  $N^{1/p}$ , it will take an enormously large amount of data to have a dense sample (i.e. a sample with high density at an input point). All this sparsity related problems makes the non-parametric estimation of the sample density a difficult and unreliable business.

The goal of unsupervised learning will be to characterize the collection of  $X$  values where  $p(X)$  is large. However, the validity of the techniques cannot be measured directly. The common techniques include identifying low-dimensional manifolds within the  $X$ space that represent high density (PCA,

multidimensional scaling etc), finding multiple regions in the  $X$  space containing modes (cluster analysis, mixture modeling). We will discuss the cluster analysis in detail and a few others will be mentioned as well.