

## Chapter 3

# Supervised and unsupervised learning - 2

### 3.1 Introduction

### 3.2 Discrimination and Classification

Discrimination and classification leads us to separate distinct sets of objects and allocate new objects to previously defined groups. Discriminant analysis is employed as a separative procedure to investigate observed differences. Classification leads us to well-defined rules to assign new objects. Usually, the former tries to describe differential features of objects from known collections. The latter focuses on sorting the objects in two or more labeled classes. However, we must admit that the two techniques overlap a lot. The feature of the observations consists of the measurements on  $p$  associated random variables, say  $\mathbf{x} = (x_1, \dots, x_p)^t$ . Allocation or classification rules are developed as learners. Some examples of classification / separation problems are as follows.

**Example 1 :** The bank wants to classify their customers with good and poor credit risks, to decide on loans. the measurements taken are income, age, number of credit cards, family size etc.

**Example 2 :** The Iris flower data is classified as Setosa, Versicolor and Virginia with measurements on sepal and petal lengths and widths. In future, an unknown specimen is collected and required to assign one of the species classes.

To start with, we denote the classes as  $\pi_1, \dots, \pi_k$ . The goal of the learner is to either provide a partition of the  $X$ -space in regions  $R_1, \dots, R_k$  such that a new observation  $\mathbf{x}$  will be classified as belonging to  $\pi_i$  if  $\mathbf{x} \in R_i$ ; or to provide a vector valued function  $\hat{p}(\mathbf{x}) = (\hat{p}_1, \dots, \hat{p}_k)(\mathbf{x}) \geq 0$  such that  $\sum \hat{p}_i(\mathbf{x}) = 1$ , where each  $\hat{p}_i$  denote the estimated probability of that observation coming from  $\pi_i$ . We may then choose the label  $i$  for which  $\hat{p}_i(\mathbf{x})$  is the maximum.

For the training data, for each item  $\mathbf{x}_i, i = 1, \dots, N$ , we have the associated class label  $g_i$  (taking values  $1, \dots, k$ , or equivalently, the vector  $Y_i$  (taking values  $e_1, \dots, e_k$ ). As mentioned before, the learner can give a probability function  $\hat{Y}(\mathbf{x})$  or a label  $\hat{g}(\mathbf{x})$ , as per the requirement. We discuss the different learners in this aspect. Before that, we build up a decision theoretic framework to compare those classifiers.

### 3.2.1 Decision theoretic approach

It can be argued that whereas the learning samples contain information about the populations, why the same cannot be observed for new samples. The reason can be incomplete knowledge of future performances (e.g. bankruptcy of firms or success of students), potential destruction of objects (lifetimes of bulbs), unavailability or expensiveness of certain information etc. This is precisely the prediction problem targets at, as assigning new rules will be the main challenge of a learner. Clearly, it cannot be completely error-free except for extreme circumstances; since the measured characteristics of the two populations may not be completely distinct. Also, sometimes classifying a  $\pi_1$  object to population  $\pi_2$  represents a more serious error than classifying a  $\pi_2$  object to  $\pi_1$ . For example, in case of diagnosing a fatal disease, the error of predicting someone as disease-free when he is ill is substantially more serious than concluding the presence of the disease when it is absent. Similarly, in jurisdiction, the errors of finding a guilty person innocent is seen as less than incriminating a truly innocent person. To incorporate that, we assign cost functions to the misclassifications.

The observed values of  $\mathbf{x}$  differ from one class to another to some extent. The probability density of  $\mathbf{x}$  differ in the two classes, and we call them as  $f_i(\mathbf{x})$ . Let  $\Omega$  be the sample space of all outcomes  $\mathbf{x}$  and the learner seek to partition  $\Omega$  in  $k$  parts,  $R_1, \dots, R_k$ . The misclassification probabilities are then,

$$P(k|i) = P(\mathbf{x} \in R_k | \pi_i) = \int_{R_k} f_i(\mathbf{x}) d\mathbf{x}$$

Assume  $p_i$  as the prior probability of the population  $\pi_i$  ( $p_1 + \dots + p_k = 1$ ). Let us denote the pre-specified costs of assigning an object to  $\pi_j$  when it is coming from  $\pi_i$  in truth as  $c(j|i)$  [ $c(i|i) = 0$ ]. The expected cost of misclassification is

$$ECM = \sum_{i=1}^k p_i \sum_{j \neq i} c(j|i) P(j|i) = \sum_{i=1}^k p_i \sum_{j \neq i} c(j|i) \int_{R_j} f_i(\mathbf{x}) d\mathbf{x}$$

The classification rule that minimizes the ECM is defined by allocating  $\mathbf{x}$  to  $\pi_j, j = 1, \dots, k$  for which  $\sum_{i \neq j} p_i f_i(\mathbf{x}) c(j|i)$  is smallest. i.e.

$$R_j = \{ \mathbf{x} : \sum_{i \neq j} p_i f_i(\mathbf{x}) c(j|i) \leq \sum_{i' \neq j} p_{i'} f_{i'}(\mathbf{x}) c(j|i') \text{ for } i' \neq i \}$$

There can be examples when the costs and/or the prior probabilities are unknown or indeterminate, and then we consider them to be equal. (i.e. when prior probabilities are unknown, we assume  $p_i = 1/k$  for all  $i$  and when costs are indeterminate, we take  $c(i|j) = c$  for some constant  $c$ ). Also, there can be scenarios when the true costs unknown, but the cost ratio is assignable (e.g. the cost of not recognizing a fraudulent banking activity when there is one is 100 times severe than wrongly adjudging a genuine transaction, and so on).

In the simple case  $k = 2$ , the rule can be simplified as  $R_1 = \{ \mathbf{x} : \frac{f_1(\mathbf{x})}{f_2(\mathbf{x})} \geq \frac{p_2 c(2|1)}{p_1 c(1|2)} \}$ . Denoting the right side as  $\kappa$ , a new measurement  $\mathbf{x}_0$  is then classified as  $\pi_1$  if  $f_1(\mathbf{x}_0) \geq \kappa f_2(\mathbf{x}_0)$  and as  $\pi_2$  otherwise. The total probability of misclassification

$$TPM = \sum_{i=1}^k p_i \sum_{j \neq i} P(j|i) = \sum_{i=1}^k p_i \sum_{j \neq i} \int_{R_j} f_i(\mathbf{x}) d\mathbf{x}$$

is sometimes considered important (ignoring the costs). This is minimized by the rule which allocates  $\mathbf{x}$  to  $\pi_j$  for which  $\sum_{i \neq j} p_i f_i(\mathbf{x})$  is smallest, or

equivalently,  $p_j f_j(\mathbf{x})$  is largest.

**Bayes Classifier :** The motivation behind the Bayes classification is governed by the interpretation of the classes as parameters. To see this, the classes  $\pi_i$  are considered as coming from a multinomial distribution with parameters  $p_i$ , and the generation of the data is governed by the conditional likelihood  $f_i(\mathbf{x})$  or  $f(\mathbf{x}|\pi_i)$ . The posterior probabilities of the different classes can then be defined as

$$P(\pi_i|\mathbf{x}) = \frac{p_i f_i(\mathbf{x})}{p_1 f_1(\mathbf{x}) + \dots + p_k f_k(\mathbf{x})}$$

Accordingly, the Bayes classifier then chooses the class label  $i$  for which  $P(\pi_i|\mathbf{x})$  is maximum, i.e.  $p_i f_i(\mathbf{x})$  is maximum as well. In essence, the rule has the simple form, allocate  $\mathbf{x}$  to  $\pi_i$  if  $p_i f_i(\mathbf{x})$  is largest. It is therefore equivalent to the classifier which minimizes the total probability of misclassification.

### 3.2.2 Error rates

The error rates or misclassification probabilities are indicative of the performance of any learner. They can be computed directly if the parent populations are known, or estimated from the sample in practice. To start with, we define the optimal error rate (OER) as the smallest possible value of total probability of misclassification, i.e.  $\text{OER} = \min \text{TPM}$ . However, since it is not achieved in practice, one can think of the actual error rates (AER) by the estimated optimal regions. A direct approach to compute the error rates are as follows. Let there be  $n_1$  samples from  $\pi_1$ , among which  $n_{1c}$  are correctly classified to  $\pi_1$  by the learner and  $n_{1m} = n_1 - n_{1c}$  are misclassified. Similarly, out of  $n_2$  samples from  $\pi_2$ ,  $n_{2c}$  and  $n_{2m}$  are classified correctly and incorrectly, respectively. The misclassification rates can be estimated as the apparent error rates

$$APER = \frac{n_{1m} + n_{2m}}{n_1 + n_2}$$

i.e. the fraction of items in the training sample that is misclassified.

However, as we are using the same data to build the learner and validate, the APER tends to underestimate the actual error rate (similar to overfitting

in regression). The notion of dividing the training sample into learning and validation purposes is a remedy. Another option is to use hold-out procedure (due to Lachenbruch), as follows.

1. Hold one observation from  $\pi_1$  and build the learner based on the other  $n_1 + n_2 - 1$  observations. Classify the holdout item using the learner.
2. Repeat this procedure for all  $n_1$  items in  $\pi_1$  and  $n_2$  items in  $\pi_2$ . Count the number of misclassified observations, and find out the fraction among all  $n_1 + n_2$  observations.

### 3.2.3 k Nearest Neighbor technique

The k-NN learner is based on closest training examples in the feature space. Here, the function of interest is estimated by a local approximation and can be also used for regression. It is among the simplest classification learners. An object is classified by a majority vote of its neighbors, being assigned to the class most common amongst its k closest data points. This method can also be used for regression, and modified to yield probabilities for class labels instead of specific labels.

To this end, the training set is stored with the class labels in the first phase. In the classification phase, for any test sample  $\mathbf{x}$  in the feature space  $\mathbb{R}^p$ , define  $N_k(\mathbf{x})$  as the set of closest points  $\mathbf{x}_i$  from  $\mathbf{x}$  (using Euclidean distances, in general). In case of assigning class labels,  $\mathbf{x}$  is then included in  $R_i$ , i.e. given the  $i$ th label, if among all the members of  $N_k(\mathbf{x})$ , the maximum number of items come from the population  $\pi_i$ . In other words, if  $g_i$  denote the class labels of the observations  $\mathbf{x}_i$ , then  $\mathbf{x}$  is assigned the label  $g_i$  such that majority of the  $\mathbf{x}_i$ s in  $N_k(\mathbf{x})$  are from that population. In words, we find the  $k$  observations with  $\mathbf{x}_i$  closest to  $\mathbf{x}$  in feature space, and classify  $\mathbf{x}$  in the class most frequent among them.

In case of regression, the nearest neighbor method fit for  $Y$  is defined as  $\hat{Y}(\mathbf{x}) = \frac{1}{k} \sum_{x_i \in N_k(\mathbf{x})} y_i$ , as the learner at  $\mathbf{x}$ . This directly gives us the predicted value of the output at  $\mathbf{x}$ . Here, we again take the  $k$  observations with  $\mathbf{x}_i$  closest

to  $\mathbf{x}$  in the  $X$  space and average their responses. This can be extended to the case of classification as well. Consider the case where  $y_i$ -s are coming from the set  $\{e_1, \dots, e_p\}$ , such that  $y_i = e_j$  if  $\mathbf{x}_i$  has the label  $g_j$ . Then, the learner  $\hat{Y}(\mathbf{x}) = \frac{1}{k} \sum_{x_i \in N_k(\mathbf{x})} y_i$  yields a vector of probabilities  $(\hat{p}_1(\mathbf{x}), \dots, \hat{p}_p(\mathbf{x}))$  such that each  $\hat{p}_i(\mathbf{x})$  gives the probability of classifying  $\mathbf{x}$  in  $R_i$ .

Now, the choice of  $k$  is crucial in this case. A moderate  $k$  will yield irregular, but fairly connected regions. However, a 1-nearest-neighbor classification will lead us to Voronoi Tessellation of the training data, where each point  $x_i$  has an associated tile where it is the closest input point. In that case, each  $R_i$  will be a union of disjoint sets in the  $X$ -space and the picture will look confusing and difficult to interpret.

Though k-NN learner is the simplest one, because of its arbitrary nature, remedies have been suggested. For example, kernel methods use weights that are smoothly decreasing from the target point, rather than the 0 – 1 nature of the above learner.