67577 – Intro. to Machine Learning	**Fall semester, 2004/5**

Lecture 11: Bayesian Decision Theory

Lecturer: Amnon Shashua *Scribe: Amnon Shashua* [1]

During the next few lectures we will be looking at the inference from training data problem as a *random* process modeled by the joint probability distribution over input (measurements) and output (say class labels) variables. Previously, the notion of a probability distribution was implicit in our analysis in the sense that the PAC model assumed some fixed distribution but other than that there was no attempt to reason about or to directly estimate the distribution. Now we will be taking a different approach of modeling the distribution and making the inference (decision rules) based on the estimated distributions. In general, estimating the underlying distribution is a daunting and unwieldy task, but there are a number of constraints or "tricks of the trade" so to speak that under certain conditions make this task manageable and fairly effective.

To make things simple, we will assume a discrete world, i.e., that the values of our random variables take on a finite number of values. Consider for example two random variables X taking on k possible values $x_1, ..., x_k$ and H taking on two values h_1, h_2. The values of X could stand for a Body Mass Index (BMI) measurement $weight/height^2$ of a person and H stands for the two possibilities h_1 standing for the "person being over-weight" and h_2 as the possibility "person of normal weight". Given a BMI measurement we would like to estimate the probability of the person being over-weight.

The joint probability $P(X, H)$ is a two dimensional array (2-way array) with $2k$ entries (cells). Each training example (x_i, h_j) falls into one of those cells, therefore $P(X = x_i, H = h_j) = P(x_i, h_j)$ holds the ratio between the number of hits into cell (i, j) and the total number of training examples. As a result $\sum_{ij} P(x_i, h_j) = 1$.

The projections of the array onto its vertical and horizontal axes by summing over columns or over rows is called *marginalization* and produces $P(h_j) = \sum_i P(x_i, h_j)$ the sum over the j'th row is the probability $P(H = h_j)$, i.e., the probability of a person being over-weight (or not) before we see any measurement — these are called *priors*. Likewise, $P(x_i) = \sum_j P(x_i, h_j)$ is the probability $P(X = x_i)$ which is the probability of receiving such a BMI measurement to begin with — this is often called *evidence*. Note that, by definition, $\sum_j P(h_j) = \sum_i P(x_i) = 1$. In Fig. 11.1 we have that $P(h_1) = 14/22, P(h_2) = 8/22$ that is there is a higher prior probability of a person being over-weight than being of normal weight. Also $P(x_3) = 7/22$ is the highest meaning that we encounter BMI $= x_3$ with the highest probability.

The *conditional probability* $P(h_j \mid x_i) = P(x_i, h_j)/P(x_i)$ is the ratio between the number of hits in cell (i, j) and the number of hits in the i'th column, i.e., the probability that the outcome is $H = h_j$ given the measurement $X = x_i$. In Fig. 11.1 we have $P(h_2 \mid x_3) = 3/7$. Note that

$$\sum_j P(h_j \mid x_i) = \sum_j \frac{P(x_i, h_j)}{P(x_i)} = \frac{1}{P(x_i)} \sum_j P(x_i, h_j) = P(x_i)/P(x_i) = 1.$$

Likewise, the conditional probability $P(x_i \mid h_j) = P(x_i, h_j)/P(h_j)$ is the number of hits in cell (i, j) normalized by the number of hits in the j'th row and represents the probability of receiving

[1]

h_1	2	5	4	2	1
h_2	0	0	3	3	2
	x_1	x_2	x_3	x_4	x_5

Figure 11.1: Joint probability $P(X, H)$ where X ranges over 5 discrete values and H over two values. Each entry contains the number of hits for the cell (x_i, h_j). The joint probability $P(x_i, h_j)$ is the number of hits divided by the total number of hits (22). See text for more details.

BMI $= x_i$ given the class label $H = h_j$ (over-weight or not) of the person. In Fig. 11.1 we have $P(x_3 \mid h_2) = 3/8$ which is the probability of receiving BMI $= x_3$ given that the person is known to be of normal weight. Note that $\sum_i P(x_i \mid h_j) = 1$.

The Bayes formula arises from:

$$P(x_i \mid h_j)P(h_j) = P(x_i, h_j) = P(h_j \mid x_i)P(x_i),$$

from which we get:

$$P(h_j \mid x_i) = \frac{P(x_i \mid h_j)P(h_j)}{P(x_i)}.$$

The left hand side $P(h_j \mid x_i)$ is called the *posterior* probability and $P(x_i \mid h_j)$ is called the *class conditional likelihood*. The Bayes formula provides a way to estimate the posterior probability from the prior, evidence and class likelihood. It is useful in cases where it is natural to compute (or collect data of) the class likelihood, yet it is not quite simple to compute directly the posterior. For example, given a measurement "12" we would like to estimate the probability that the measurement came from tossing a pair of dice or from spinning a roulette table. If $x = 12$ is our measurement, and h_1 stands for "pair of dice" and h_2 for "roulette" then it is natural to compute the class conditional: $P(\text{"12"} \mid \text{"}\textit{pair of dice}\text{"}) = 1/36$ and $P(\text{"12"} \mid \text{"}\textit{roulette}\text{"}) = 1/38$. Computing the posterior directly is much more difficult. As another example, consider medical diagnosis. Once it is known that a patient suffers from some disease h_j, it is natural to evaluate the probabilities $P(x_i \mid h_j)$ of the emerging symptoms x_i. As a result, in many inference problems it is natural to use the class conditionals as the basic building blocks and use the Bayes formula to invert those to obtain the posteriors.

If we draw the posteriors $P(h_1 \mid x)$ and $P(h_2 \mid x)$ using the probability distribution array in Fig. 11.1 we will see that $P(h_1 \mid x) > P(h_2 \mid x)$ for all values of X smaller than a value which is in between x_3 and x_4. Therefore the decision which will minimize the probability of misclassification would be to choose the class with the maximal posterior:

$$h^* = \text{argmax}_j P(h_j \mid x),$$

which is known as the Maximal A Posteriori (MAP) decision principle. Since $P(x)$ is simply a normalization factor, the MAP principle is equivalent to:

$$h^* = \text{argmax}_j P(x \mid h_j)P(h_j).$$

In the case where information about the prior $P(h)$ is not known or it is known that the prior is uniform, the we obtain the Maximum Likelihood (ML) principle:

$$h^* = \text{argmax}_j P(x \mid h_j).$$

The MAP principle is a particular case of a more general principle, known as "proper Bayes", where a *loss* is incorporated into the decision process. Let $l(h_i, h_j)$ be the loss incurred by deciding on class h_i when in fact h_j is the correct class. For example, the "0/1" loss function is:

$$l(h_i, h_j) = \left\{ \begin{array}{cc} 1 & i \neq j \\ 0 & i = j \end{array} \right\}$$

The least-squares loss function is: $l(h_i, h_j) = \|h_i - h_j\|^2$ typically used when the outcomes are vectors in some high dimensional space rather than class labels. We define the *expected risk*:

$$R(h_i \mid x) = \sum_j l(h_i, h_j) P(h_j \mid x).$$

The proper Bayes decision policy is to minimize the expected risk:

$$h^* = \text{argmin}_j R(h_j \mid x).$$

The MAP policy arises in the case $l(h_i, h_j)$ is the 0/1 loss function:

$$R(h_i \mid x) = \sum_{j \neq i} P(h_j \mid x) = 1 - P(h_i \mid x),$$

Thus,

$$\text{argmin}_j R(h_j \mid x) = \text{argmax}_j P(h_j \mid x).$$

11.1 Independence Constraints

At this point we may pause and ask what have we obtained? well, not much. Clearly, the inference problem is captured by the joint probability distribution and we do not need all these formulas to see this. How do we obtain the necessary data to fill in the probability distribution array to begin with? Clearly without additional simplifying constraints the task is not practical as the size of these kind of arrays are exponential in the number of variables. There are three families of simplifying constraints used in the literature:

- statistical independence constraints,
- parametric form of the class likelihood $P(x_i \mid h_j)$ where the inference becomes a density estimation problem,
- structural assumptions — latent (hidden) variables, graphical models.

Today we will focus on the first of these simplifying constraints — statistical independence properties.

Consider two random variables X and Y. The variables are statistically independent $X \perp Y$ if $P(X \mid Y) = P(X)$ meaning that information about the value of Y does not add anything about X. The independence condition is equivalent to the constraint: $P(X, Y) = P(X)P(Y)$. This can be easily proven: if $X \perp Y$ then $P(X, Y) = P(X \mid Y)P(Y) = P(X)P(Y)$. On the other hand, if $P(X, Y) = P(X)P(Y)$ then

$$P(X \mid Y) = \frac{P(X, Y)}{P(Y)} = \frac{P(X)P(Y)}{P(Y)} = P(X).$$

Let the values of X range over $x_1, ..., x_k$ and the values of Y range over $y_1, ..., y_l$. The associated $k \times l$ 2-way array, $P(X = x_i, Y = y_j)$ is represented by the outer product $P(x_i, y_j) = P(x_i)P(y_j)$ of two vectors $P(X) = (P(x_1), ..., P(x_k))$ and $P(Y) = (P(y_1), ..., P(y_l))$. In other words, the 2-way array viewed as a *matrix* is of rank 1 and is determined by $k + l$ (minus 2 because the sum of each vector is 1) parameters rather than kl (minus 1) parameters.

Likewise, if $X_1 \perp X_2 \perp .... \perp X_n$ are n statistically independent random variables where X_i ranges over k_i discrete and distinct values, then the n-way array $P(X_1, ..., X_n) = P(X_1) \cdot ... \cdot P(X_n)$ is an outer-product of n vectors and is therefore determined by $k_1 + ... + k_n$ (minus n) parameters instead of $k_1 k_2 ... k_n$ (minus 1) parameters[2]. Viewed as a tensor, the joint probability is a rank 1 tensor. The main point is that the statistical independence assumption reduced the representation of the multivariate joint distribution from *exponential to linear size.*

Since our variables are typically divided to measurement variables and an output/class variable H (or in general $H_1, ..., H_l$), it is useful to introduce another, weaker form, of independence known as *conditional independence*. Variables X, Y are conditionally independent given H, denoted by $X \perp Y \mid H$, iff $P(X \mid Y, H) = P(X \mid H)$ meaning that given H, the value of Y does not add any information about X. This is equivalent to the condition $P(X, Y \mid H) = P(X \mid H)P(Y \mid H)$.

Consider two examples. Let X be the outcome of the first coin toss (the values of X are 0/1) and let Y be the second toss of the coin. Assume there is a possibility that the coin is biased towards heads (the value "0") but we are not certain. Then, X and Y are not independent because $P(Y = 0 \mid X = 0) > P(Y = 0)$ meaning that observing $X = 0$ (first toss came out "head") causes us to increase our belief in a biased coin and thus increase the probability that the second toss will be head as well. So, there is a third variable $H = \{true, false\}$ representing the event "coin is biased towards head". Once we know the value of H then any evidence about the value of X would not change our belief about Y.

As another example, Joe and Mo live on opposite sides of the city. Joe goes to work by train and Mo by car. Let X be the event "Joe is late to work" and Y be the event "Mo is late for work". Clearly X and Y are not independent because there could be other factors. For example, a train strike will cause Joe to be late, but because of the strike there would be extra traffic (people using their car instead of the train) thus causing Mo to be pate as well. Therefore, a third variable H standing for the event "train strike" would decouple X and Y.

From a computational standpoint, the conditional independence assumption has a similar effect to the unconditional independence. Let X range over k distinct value, Y range over r distinct values and H range over s distinct values. Then $P(X, Y, H)$ is a 3-way array of size $k \times r \times s$. Given that $X \perp Y \mid H$ means that $P(X, Y \mid H = h_i)$, a 2-way "slice" of the 3-way array along the H axis is represented by the outer-product of two vectors $P(X \mid H = h_i)P(Y \mid H = h_i)$. As a result the 3-way array is represented by $s(k + r - 2)$ parameters instead of $skr - 1$. Likewise, if $X_1 \perp .... \perp X_n \mid H$ then the n-way array $P(X_1, ..., X_n \mid H = h_i)$ (which is a slice along the H axis of the $(n + 1)$-array $P(X_1, ..., X_n, H)$) is represented by an outer-product of n vectors, i.e., by $k_1 + .. + k_n - n$ parameters.

As an example, consider n coins represented by random variables $X_1, ..., X_n \in \{0, 1\}$ where each coin has a bias $0 < q_i < 1$ for head (value "0"). An observation is a toss of the n coins producing an n-dimensional binary vector. Without additional constraints we have 2^n possibilities, and for each there is a probability of occurrence, i.e. $P(X_1, ..., X_n \mid q_1, ..., q_n)$ is a n-way array of size 2^n. But coin tosses are independent, i.e., the result of coin i does not convey any information about the result

[2] I am a bit over simplifying things because we are ignoring here the fact that the entries of the array should be non-negative. This means that there are additional non-linear constraints which effectively reduce the number of parameters — but nevertheless it stays exponential.

of other coins), thus $P(X_1, ..., X_n)$ is an outer-product of n two-dimensional vectors $(q_i, 1 - q_i)$:

$$P(X_1, ..., X_n) = P(X_1) \cdots P(X_n) = \begin{pmatrix} q_1 \\ 1 - q_1 \end{pmatrix} \otimes ... \otimes \begin{pmatrix} q_n \\ 1 - q_n \end{pmatrix},$$

where $\otimes$ stand for the Cartesian product. Let us simplify this further and assume that $q_1 = ... = q_n = q$, i.e., all the biases are the same. Now, as we explained above, the tosses are not ndependent, but they are conditionally independent on q. But, in addition to conditional independence we have that the *order* does not matter and therefore:

$$P(X_1, ..., X_n \mid q) = \begin{pmatrix} q \\ 1 - q \end{pmatrix}^{\otimes n},$$

which is now represented by a single parameter q. Each entry in the n-way array is uniquely determined by the number λ of heads in the n-toss sequence represented by the entry:

$$P(X_1 = x_1, ..., X_n = x_n \mid q) = q^\lambda (1 - q)^{n-\lambda},$$

where λ is the number of heads in the binary vector $(x_1, ..., x_n)$. The maximum likelihood principle, is to find the largest entry along the q-axis at the coordinate $x_1, ..., x_n$.

$$q^* = \text{argmax}_q q^\lambda (1 - q)^{n-\lambda} = \text{argmax}_q \log[q^\lambda (1 - q)^{n-\lambda}].$$

Taking the partial derivative with respect to q and setting it to zero:

$$\frac{\partial}{\partial q}[\lambda \log q + (n - \lambda) \log(1 - q)] = \frac{\lambda}{q} + \frac{n - \lambda}{1 - q} = 0,$$

produces the result:

$$q = \frac{\lambda}{n}.$$

We see that the low-rank and the symmetry constraints reduced the necessary number of observations from an exponential measure to a *single* measure (a toss of the n coins). Moreover, low rank constraints and non-parametric multivariate distribution are two aspects of the same problem — however unlike PCA here we need to enforce non-negativity in the factorization. These type of tensor factorizations are part of advanced topics which are beyond the scope of our discussion.

Consider another application of conditional dependence which is the Bayes incremental rule. Suppose we have processed n examples $X^{(n)} = \{X_1, ..., X_n\}$ and computed somehow $P(H \mid X^{(n)})$. We are given a new measurement X and wish to compute (update) the posterior $P(H \mid X^{(n)}, X)$. We will use the chain rule[3]:

$$P(X \mid Y, Z) = \frac{P(Z \mid X, Y)P(X \mid Y)}{P(Z \mid Y)}$$

to obtain:

$$P(H \mid X^{(n)}, X) = \frac{P(X \mid X^{(n)}, H)P(H \mid X^{(n)})}{P(X \mid X^{(n)})}$$

[3]this is based on the rule $P(X_1, ..., X_n) = P(X_1 \mid X_2, ..., X_n)P(X_2 \mid X_3, ..., X_n) \cdots P(X_{n-1} \mid X_n)P(X_n)$

from conditional independence, $P(X \mid X^{(n)}, H) = P(X \mid H)$. The term $P(X \mid X^{(n)})$ can expanded as follows:

$$\begin{aligned} P(X \mid X^{(n)}) &= \sum_i \frac{P(X, X^{(n)} \mid H = h_i)P(H = h_i)}{P(X^{(n)}} = \sum_i \frac{P(X \mid H = h_i)P(X^{(n)} \mid H = h_i)P(H = h_i)}{P(X^{(n)}} \\ &= \sum_i P(X \mid H = h_i)P(H = h_i \mid X^{(n)}) \end{aligned}$$

After substitution we obtain:

$$P(H = h_i \mid X^{(n)}, X) = \frac{P(X \mid H = h_i)P(H = h_i \mid X^{(n)})}{\sum_j P(X \mid H = h_j)P(H = h_j \mid X^{(n)})}.$$

The old posterior $P(H \mid X^{(n)})$ is now the prior for the updated formula. Consider the following example[4]: We have a coin which could be either fair or biased towards Head at a probability of 0.6. Let $H = h_1$ be the event that the coin is fair, and $H = h_2$ that the coin is biased. We start with prior probabilities $P(h_1) = 0.75$ and $P(h_2) = 0.25$ (we have a higher initial belief that the coin is fair). Suppose our first coin toss is a Head, i.e., $X_1 =$ "0". Then,

$$P(h_1 \mid x_1) = \frac{P(x_1 \mid h_1)P(h_1)}{P(x_1)} = \frac{0.5 * 0.75}{0.5 * 0.75 + 0.6 * 0.25} = 0.714$$

and $P(h_2 \mid x_1) = 0.286$. Our posterior belief that the coin is fair has gone down after a Head toss. Assume we have another measurement $X_2 =$ "0", then:

$$P(h_1 \mid x_1, x_2) = \frac{P(x_2 \mid h_1)P(h_1 \mid x_1)}{normalization} = \frac{0.5 * 0.714}{0.5 * 0.714 + 0.6 * 0.286} = 0.675,$$

and $P(h_2 \mid x_1, x_2) = 0.325$, thus our belief that the coin is fair continues to go down after Head tosses.

[4] adopted from Ron Rivest's 1994 class notes.