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

Lecture 12: Bayes II: Graphical Models, Density Estimation

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

In the previous lecture we noted that without reducing the complexity of representing the joint probability distribution array, and in turn the complexity of making inference from the array (taking marginals, maximizing over variables — MAP), there is not much use to a multivariate probability-based approach. To restate, 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 continue with the pursuit of statistical independence constraints and then address the density estimation problem and the Bayes optimal discriminant function for normal densities (which would be equivalent to the LDA solution we encountered in Class 10).

12.1 Multiple Conditional Independence Statements — Tree Models and Belief Propagation

So far we considered a single statement of conditional independence of the type $P(X \mid Y, Z) = P(X \mid Z)$, i.e., $X \perp Y \mid Z$. We saw that such as statement translates to a factorization result: $P(X, Y \mid Z) = P(X \mid Z)P(Y \mid Z)$. Consider now multiple such statements over a set of random variables $V = \{X_1, ..., X_n\}$ where each variable is of cardinality k (i.e., has k possible discrete values). Instead of considering the conditional independent statements directly, we will (for now) consider their implications, i.e., the factorization result as follows. Let $S_i \subseteq V$ be a subset of the variables associated with X_i, then:

$$P(X_1, ..., X_n) = \prod_{i=1}^{n} P(X_i \mid S_i).$$

Recall that this is always possible even without any constraints since in general:

$$P(X_1, ..., X_n) = P(X_1 \mid X_2, ..., X_n)P(X_2 \mid X_3, ..., X_n) \cdots P(X_{n-1} \mid X_n)P(X_n).$$

We obtain, therefore, a complexity reduction when the cardinality of the subsets S_i are relatively small. The complexity reduction is in the number of parameters (in general k^n) required to represent the the joint probability distribution and in the computational complexity of operations (inference problems) we would like to perform on the distribution array. We will not discuss the space

[1]

complexity (extension of the rank-1 constraints we saw in the single statement scenario) and instead focus here on the computational complexity issues.

As mentioned previously, there are two types of inference operations we would like to perform on the distribution array: (i) compute marginals, say $P(X_5)$ or in general $P(X_A) = \sum_{X_B} P(X_A, X_B)$ where (A, B) is a partitionaing of V, i.e., $V = A \cup B$ and $A \cap B = \emptyset$; and (ii) compute MAP estimates: $\max_{X_A} P(X_A \mid X_B)$. As we shall note later the two operations (sum and max) are interchangeable thus for now we will consider the operation of taking marginals.

Consider $X_1, ..., X_5$ factorized as follows:

$$P(X_1, ..., X_5) = P(X_1)P(X_2 \mid X_1)P(X_3 \mid X_2)P(X_4 \mid X_2)P(X_5 \mid X_4). \tag{12.1}$$

Say we wish to compute the marginal $P(X_5)$:

$$P(X_5) = \sum_{X_1,...,X_4} P(X_1, ..., X_4, X_5),$$

which in general would require k^4 operations. Due to the factorization structure we can reduce this to $4k^2$ by factorizing the summations as follows. Let $m_{12}(X_2)$ be an intermediate sum:

$$m_{12}(X_2) = \sum_{X_1} P(X_1)P(X_2 \mid X_1).$$

We have eliminated the variable X_1 and $m_{12}(X_2)$ is a function of X_2 (a vector with k entries) only. We will use the notation $m_{ij}(X_j)$ to denote summation over X_i leaving a function over X_j. Note that the operation requires k^2 steps because for every value of X_2 we sum over all values of X_1. We are left with:

$$P(X_5) = \sum_{X_2,X_3,X_4} m_{12}(X_2)P(X_3 \mid X_2)P(X_4 \mid X_2)P(X_5 \mid X_4).$$

Let

$$m_{32}(X_2) = \sum_{X_3} P(X_3 \mid X_2),$$

which leaves us with:

$$P(X_5) = \sum_{X_2,X_4} m_{12}(X_2)m_{32}(X_2)P(X_4 \mid X_2)P(X_5 \mid X_4).$$

Now we will eliminate X_2:

$$m_{24}(X_4) = \sum_{X_2} m_{12}(X_2)m_{32}(X_2)P(X_4 \mid X_2),$$

and finally $P(X_5)$ is evaluated:

$$P(X_5) = \sum_{X_4} m_{24}(X_4)P(X_5 \mid X_4). \tag{12.2}$$

Taken together, we spent 4 summation loops each taking k^2 operations. We could have done the same for MAP computation:

$$X_5^* = \max_{X_1,...,X_4} P(X_1)P(X_2 \mid X_1)P(X_3 \mid X_2)P(X_4 \mid X_2)P(X_5 \mid X_4),$$

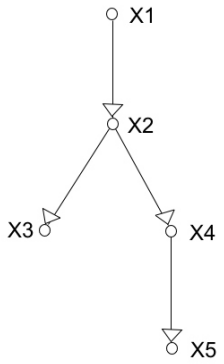

Figure 12.1: Directed graph representation of eqn. (12.1)

by replacing each "sum" with a "max" throughout the process because "max" commutes with products just as "sum" does.

To appreciate the general feel of what was done here, we can associate a directed graph with the variables represented by the vertices of the graph, and the subsets S_i are the "parents" of the vertex X_i, i.e., there is a directed edge from each vertex of S_i to X_i. The graph associated with our example is displayed in Fig. 12.1 — note that the undirected underlying graph forms a tree (a connected graph with $n-1$ edges, or equivalently a connected graph with no loops). If we limit our discussion to *trees* then the key operation of summing a product has the form:

$$m_{ij}(X_j) = \sum_{X_i} P(X_j \mid X_i) \prod_{l \in N(i), l \neq j} m_{li}(X_i),$$

where $N(i)$ are the neighbors (in the undirected graph) of X_i in the graph. For example, the computation of $m_{24}(X_4)$ contains the product of $m_{12}(X_2)m_{32}(X_2)$ because X_1, X_3 are neighbors of X_2 in the graph. The marginal $P(X_i)$ is the result:

$$P(X_i) \cong \prod_{l \in N(i)} m_{li}(X_i),$$

where equality is up to scale. For example, $P(X_5) = m_{45}(X_5)$ defined in eqn. (12.2) because X_4 is the only neighbor of X_5. It is customary to view m_{ij} as "messages" which vertex X_i "sends" to X_j. A vertex (node) sends a message to a neighboring node once it received messages from *all of its other neighbors*. So for example, m_{24} was sent to X_4 after m_{12} and m_{32} were evaluated. The marginal probability (up to scale) of a node is given by the product of all the incoming messages.

This algorithm is called the *sum product algorithm* or *belief propagation* — due primarily to Pearl (1988) who studied most extensively this type of structural factorization back in the 80s. The complexity of the algorithm is $O(|E|k^2)$ where $|E|$ is the number of edges in the graph. For every $(i,j) \in E$ both m_{ij} and m_{ji} are computed.

The extension to general graphs is much harder and would not be discussed here. It is worthwhile to note that it would have just the same with undirected graph representations. Our example can be represented up to scale by:

$$P(X_1, ..., X_5) \cong P(X_1, X_2)P(X_2, X_3)P(X_2, X_4)P(X_4, X_5),$$

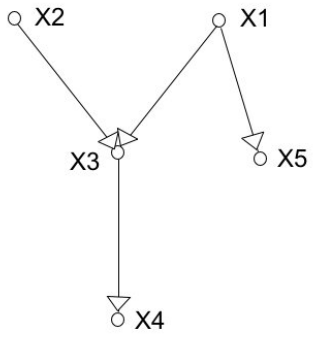

Figure 12.2: Directed graph representation of eqn. (12.3)

where the subsets are *cliques* (fully connected subsets of nodes) in the undirected graph. We associate with each clique a scaled probability distribution over the nodes in the clique. Any directed graph can be transformed to the undirected graph representation by adding edges connecting the parents S_i of node X_i and then undirecting all edges of the graph — this is called the *moral graph* in which all the arguments $P(X_i \mid S_i)$ are contained in a clique. The sum product algorithm as presented above is valid to undirected graphs forming a tree.

We will finish this section with the description of the conditional independent statements induced by the directed graph. If S_i are the parents of X_i, then let N_i be the non-descendants of X_i in the graph. Then:

$$X_i \perp N_i \mid S_i.$$

For example, consider the graph displayed in Fig. 12.2. The factorization structure is:

$$P(X_1, ..., X_5) = P(X_1)P(X_2)P(X_3 \mid X_1, X_2)P(X_4 \mid X_3)P(X_5 \mid X_1). \tag{12.3}$$

The general decomposition, ignoring the graph, is:

$$P(X_1, ..., X_5) = P(X_1)P(X_2 \mid X_1)P(X_3 \mid X_1, X_2)P(X_4 \mid X_1, X_2, X_3)P(X_5 \mid X_1, ..., X_4).$$

The factorization structure is due to the following conditional independent statements:

$$\begin{aligned} &X_1 \perp X_2 \\ &X_4 \perp \{X_1, X_2, X_5\} \mid X_3 \\ &X_5 \perp \{X_2, X_3, X_4\} \mid X_1 \end{aligned}$$

12.2 Parametric Distributions: Density Estimation

So far we considered constraints induced by conditional independent statements among the random variables as a means to reduce the space and time complexity of the multivariate distribution array. Another approach would be to assume some form of parametric form governing the entries of the array — the most popular assumption is Gaussian distribution $P(X_1, ..., X_n) \sim N(\mu, E)$ with mean

vector μ and covariance matrix E. The parameters of the density function are denoted by $\theta = (\mu, E)$ and for every vector $\mathbf{x} \in R^n$ we have:

$$P(\mathbf{x} \mid \theta) = \frac{1}{(2\pi)^{n/2}|E|^{1/2}} \exp^{-\frac{1}{2}(\mathbf{x}-\mu)^\top E^{-1}(\mathbf{x}-\mu)}.$$

Assume we are given an i.i.d sample of k points $S = \{\mathbf{x}_1, ..., \mathbf{x}_k\}$, $\mathbf{x}_i \in R^n$, and we would like to find the Bayes optimal θ:

$$\theta^* = \text{argmax}_\theta P(S \mid \theta),$$

by maximizing the likelihood (here we are assuming that the the priors $P(\theta)$ are equal, thus the maximum likelihood and the MAP would produce the same result). Because the sample was drawn i.i.d. we can assume that:

$$P(S \mid \theta) = \prod_{i=1}^{k} P(\mathbf{x}_i \mid \theta).$$

Let $L(\theta) = \log P(S \mid \theta) = \sum_i \log P(\mathbf{x}_i \mid \theta)$ and since Log is monotonously increasing we have that $\theta^* = \text{argmax}_\theta L(\theta)$. The parameter estimation would be recovered by taking derivatives with respect to θ, i.e., $\nabla_\theta L = 0$. We have:

$$L(\theta) = -\frac{1}{2}\log|E| - \sum_{i=1}^{k}\frac{n}{2}\log(2\pi) - \sum_i \frac{1}{2}(\mathbf{x}_i - \mu)^\top E^{-1}(\mathbf{x}_i - \mu). \tag{12.4}$$

We will start with a simple scenario where $E = \sigma^2 I$, i.e., all the covariances are zero and all the variances are equal to σ^2. Thus, $E^{-1} = \sigma^{-2} I$ and $|E| = \sigma^{2n}$. After substitution (and removal of items which do not depend on θ) we have:

$$L(\theta) = -nk\log\sigma - \frac{1}{2}\sum_i \frac{\|\mathbf{x}_i - \mu\|^2}{\sigma^2}.$$

The partial derivative with respect to μ:

$$\frac{\partial L}{\partial \mu} = \sigma^{-2}\sum_i(\mu - \mathbf{x}_i) = 0$$

from which we obtain:

$$\mu = \frac{1}{k}\sum_{i=1}^{k}\mathbf{x}_i.$$

The partial derivative with respect to σ is:

$$\frac{\partial L}{\partial \sigma} = \frac{nk}{\sigma} - \sigma^{-3}\sum_i \|\mathbf{x}_i - \mu\|^2 = 0,$$

from which we obtain:

$$\sigma^2 = \frac{1}{kn}\sum_{i=1}^{k}\|\mathbf{x}_i - \mu\|^2.$$

Note that the reason for dividing by n is due to the fact that $\sigma_1^2 = ... = \sigma_n^2 = \sigma^2$, so that:

$$\frac{1}{k}\sum_{i=1}^{k}\|\mathbf{x}_i - \mu\|^2 = \sum_{j=1}^{n}\sigma_j^2 = n\sigma^2.$$

In the general case, E is a full rank symmetric matrix, then the derivative of eqn. (12.4) with respect to μ is:

$$\frac{\partial L}{\partial \mu} = E^{-1} \sum_i (\mu - \mathbf{x}_i) = 0,$$

and since E^{-1} is full rank we obtain $\mu = (1/k) \sum_i \mathbf{x}_i$. For the derivative with respect to E we note two auxiliary items:

$$\frac{\partial |E|}{\partial E} = |E| E^{-T}, \qquad \frac{\partial}{\partial E} trace(AE^{-1}) = -(E^{-1} A E^{-1})^\top.$$

Using the fact that $\mathbf{x}^\top \mathbf{y} = trace(\mathbf{x}\mathbf{y}^\top)$ we can transform $\mathbf{z}^\top E^{-1} \mathbf{z}$ to $trace(\mathbf{z}\mathbf{z}^\top E^{-1})$ for any vector $\mathbf{z}$. Given that E^{-1} is symmetric, then:

$$\frac{\partial}{\partial E} trace(\mathbf{z}\mathbf{z}^\top E^{-1}) = -E^{-1} \mathbf{z}\mathbf{z}^\top E^{-1}.$$

Substituting $\mathbf{z} = \mathbf{x} - \mu$ we obtain:

$$\frac{\partial L}{\partial E} = -kE^{-1} + E^{-1} \left(\sum_i (\mathbf{x}_i - \mu)(\mathbf{x}_i - \mu)^\top \right) E^{-1} = 0,$$

from which we obtain:

$$E = \frac{1}{k} \sum_{i=1}^{k} (\mathbf{x}_i - \mu)(\mathbf{x}_i - \mu)^\top.$$

12.3 Bayes Classifier for 2-class Normal Distributions

For the last topic in this lecture consider again the 2-class inference problem. We have encountered this problem over and over in this course using Perceptron, SVM and LDA. In the Bayes framework, if $H = \{h_1, h_2\}$ denotes the "class member" variable with two possible outcomes, then the MAP decision policy calls for making the decision based on data $\mathbf{x}$:

$$h^* = \text{argmax}_{h_1, h_2} \{P(h_1 \mid \mathbf{x}), P(h_2 \mid \mathbf{x})\},$$

or in other words the class h_1 would be chosen if $P(h_1 \mid \mathbf{x}) > P(h_2 \mid \mathbf{x})$. The *decision surface* (as a function of $\mathbf{x}$) is therefore described by:

$$P(h_1 \mid \mathbf{x}) - P(h_2 \mid \mathbf{x}) = 0.$$

We saw that in SVM the decision surface (a hyperplane or a non-linear hypersurface using kernel functions) is determined by a selected subset of the training vectors (the so called "support vectors" laying at the boundary between the two classes) whereas in LDA the decision surface (hyperplane) is determined by the means and covariances of the two sets in a way that satisfies certain properties of the projection of the data points onto the normal vector to the hyperplane.

The questions we ask here is what would the Bayes optimal decision surface be like if we assume that the two classes are normally distributed with different means and the same covariance matrix? What we will see is that under the condition of equal priors $P(h_1) = P(h_2)$ the decision surface is a hyperplane — and not only that, it is the same hyperplane produced by LDA.

Claim 1 *If* $P(h_1) = P(h_2)$ *and* $P(\mathbf{x} \mid h_1) \sim N(\mu_1, E)$ *and* $P(\mathbf{x} \mid h_1) \sim N(\mu_2, E)$*, the the Bayes optimal decision surface is a hyperplane* $\mathbf{w}^\top(\mathbf{x}-\mu) = 0$ *where* $\mu = (\mu_1+\mu_2)/2$ *and* $\mathbf{w} = E^{-1}(\mu_1-\mu_2)$*. In other words, the decision surface is described by:*

$$\mathbf{x}^\top E^{-1}(\mu_1 - \mu_2) - \frac{1}{2}(\mu_1 + \mu_2)E^{-1}(\mu_1 - \mu_2) = 0. \tag{12.5}$$

Proof: The decision surface is described by $P(h_1 \mid \mathbf{x}) - P(h_2 \mid \mathbf{x}) = 0$ which is equivalent to the statement that the ratio of the posteriors is 1, or equivalently that the log of the ratio is zero, and using Bayes formula we obtain:

$$0 = \log \frac{P(\mathbf{x} \mid h_1)P(h_1)}{P(\mathbf{x} \mid h_2)P(h_2)} = \log \frac{P(\mathbf{x} \mid h_1)}{P(\mathbf{x} \mid h_2)}.$$

In other words, the decision surface is described by

$$\log P(\mathbf{x} \mid h_1) - \log P(\mathbf{x} \mid h_2) = -\frac{1}{2}(\mathbf{x} - \mu_1)^\top E^{-1}(\mathbf{x} - \mu_1) + \frac{1}{2}(\mathbf{x} - \mu_2)^\top E^{-1}(\mathbf{x} - \mu_2) = 0.$$

After expanding the two terms we obtain eqn. (12.5). □

The conclusion is that since LDA assumes that the data is represented by the class means and sums over the class covariance matrices, it is not surprising that LDA is "optimal" when the two classes are sampled from Normal distributions. The "optimality" is in fact Bayes optimal.