Note Set 2: Conditional Independence and Graphical Models

Padhraic Smyth,
Department of Computer Science
University of California, Irvine
January 2024

1 Introduction

This set of notes will cover basic concepts in **multivariate probability models**, i.e., probability models involving multiple random variables. We will begin by discussing the important concepts of independence and conditional independence and look at the general framework of graphical models. We will then look at specific examples of models: for joint distributions we will look at naive Bayes models and Markov models, and for joint densities we will look at the multivariate Gaussian model. Note that the ideas we discuss below are applicable equally well to both discrete-valued and continuous variables (and their combinations): in our examples below we will typically to focus on one or the other and it should be clear how to generalize.

2 Independence and Conditional Independence

2.1 Independence

Let A and B be two random variables (we can assume they are discrete to be concrete, but they could also be continuous).

Definition 1 *A and B are* **independent** *iff* $P(a, b) = P(a)P(b), \ \forall \ a, b.$

We see that independence of two random variables implies that their probability distribution **factors** into two parts. For example, if A and B each took 100 values, then the full joint distribution $P(a, b)$ would consist of a table of $100 \times 100 = 10,000$ probabilities. However, if one assumes or knows that A and B are independent, then we can factor the table as a product of $P(a)$ and $P(b)$ and only need $100 + 100 = 200$ probabilities which is far fewer! Of course what we save in terms of complexity (of the model) is offset by the fact that we won't be able to model all of the possible dependencies between a and b values as we would in the case where we used a full unconstrained joint distribution $P(a, b)$ without any independence assumption.

There is another way to define independence using conditional probability that provides some more intuition as to what independence means. The definition of independence we provided above is mathematically equivalent to the following alternate definition:

Definition 2 *Random variables A and B are* **independent** *iff* $P(a|b) = P(a) \ \ \forall a, b$.

This condition $P(a|b) = P(a)$ can be interpreted as stating that the event $P(B = b)$ provides no information about the event $P(A = a)$, i.e., our belief in a is not changed at all if we learn that $B = b$.

(The definition above of independence above could equivalently be stated in the "other direction" as $P(b|a) = P(b) \ \ \forall a, b$).

We can extend the definition of independence to more than 2 random variables. For example, we say that A, B, C are mutually independent if and only if $P(a, b, c) = p(a)p(b)p(c) \ \ \forall \ a, b, c$.

We can ask questions such as 'is independence transitive?' i.e., if A is independent of B, and B is independent of C, then does this imply that A is independent of C? (The answer is no: there is a simple counter-example that shows this implication can't hold in general).

2.2 Models versus the Real-World

It is important to understand that independence is often **assumed** for convenience in situations where we are trying to build a model of the real world. We are all familiar with simple games of chance, such as tossing a coin or throwing a die, where each toss or throw can be assumed to be independent of all previous tosses or throws. In this situation the assumption of independence is reasonable given what we know about the problem.

However, in many real-world problems, the phenomena we are trying to model may be dependent, perhaps in some small and subtle way. For example, the weather in Beijing today might be somewhat predictive of the weather in Seattle in a few days time, e.g., due to the jet stream traveling from west to east around the globe in the Northern hemisphere. As a consequence, while independence might be a useful modeling assumption for a problem like this, it might be too strong of an assumption depending on what our goals are—we will see below that conditional independence is often more reasonable.

Thus, unlike the properties of probability models that were discussed in Note Set 1, independence is **an assumption** that we can make (if we wish to) in building a multivariate probability model, but is not an inherent property of sets of random variables unless we build it into a model.

2.3 Conditional Independence

A more general notion of independence is **conditional independence**, defined as follows:

Definition 3 *A and B are conditionally independent given random variable C iff*

$$P(a, b|c) = P(a|c)P(b|c), \quad \forall\, a, b, c.$$

In words, A and B are independent conditioned on knowing the value of C.

As with independence, this can be equivalently stated in an intuitive manner in terms of conditional probabilities as

$$P(a|b, c) = P(a|c) \quad \text{or} \quad P(b|a, c) = P(b|c).$$

We can interpret this first equation as saying that if we already know $C = c$, then learning that $B = b$ does not change our belief in $A = a$, i.e., knowing b does not change our belief in a if we already know c.

This type of **conditional** independence is very useful in practical modeling problems. For example, we might assume that two disease symptoms A and B are conditionally independent given the disease C, e.g., the occurrence of headache and fever are conditionally independent given flu. As we mentioned earlier, these independence relations are not necessarily true in the real-world, but we can often **assume** that they hold for the purposes of building an **approximate model**—this is somewhat analogous to approximating a non-linear function with a "linear first-order approximation" in applied mathematical modeling.

Example 1: Daily Temperature in Different Parts of the World: Let X_t and Y_t be the temperatures measured at noon (local time) on day t in Irvine and in Shanghai respectively. Intuitively we don't believe that X_t and Y_t should directly depend on each other given that Shanghai and Irvine are so far away. However, if we wanted to build a perfectly accurate model for the joint density $p(x_t, y_t)$ we would need to assume that they depend on each other. For example, they will both tend to rise and fall in a similar manner depending on the time of year—so learning that it is relatively cool in Irvine will cause you to update your degree of belief about the temperature in Shanghai that day, i.e., the value of one variable carries information about the value of the other.

What's really going on here is that Irvine and Shanghai are "connected" indirectly by seasonality. For example, if we had a 3rd variable D that represents the day of the year in the Western calendar (taking values 1 through 365), we could use the conditional independence model $p(x, y|d) \approx p(x|d)p(y|d)$ and assume that Irvine and Shanghai are in fact **conditionally independent** if we know the day of the year. This is a much more sensible model than the one that connects Irvine and Shanghai directly together.

We can start to see here how we will be able to simplify models by introducing "key variables" that induce conditional independence relations among large sets of other variables (e.g., imagine building a model for daily temperatures across 100's of cities rather than just two). Of course, although we may prefer our conditional independence model better than the other model (particularly in terms of its simplicity), it might not capture all aspects of variation between X and Y.

Building models of real-world phenomena tends to be something of an art, in terms of trading-off parsimony with fidelity to detail. There is a famous quote by the statistician George Box: *"all models are wrong, but some are useful."* What he is saying here is that any probability model will not match reality exactly, so in a sense all of our models of the real-world will be wrong (even if only slightly) at some level. But nonetheless, some models are very useful even though they don't capture every detail of the real-world. For example, in physics, Newtonian models of motion don't take into account relativity, but are still very useful. Similarly we will see with probabilistic machine learning that models that use conditional independence assumptions may be over-simplifying certain aspects of the real-world, but may nonetheless be very useful in practice.

Note that in the model above for temperature, that X and Y are not marginally independent, i.e., $p(x, y) \neq p(x)p(y)$ even though they are conditionally independent. In general, conditional independence does not imply marginal independence. In the example above we can see why this is the case: conditioned on D (the day of the year), the temperatures in Irvine and Shanghai can be assumed to have no information about each other. But if we don't know D, then they have information about each other, i.e., for this problem they are dependent if D is unknown.

Note also that we can extend the definition of conditional independence to multiple random variables, e.g., we say that A, B, C, D are conditionally independent given Z, if and only if

$$P(a, b, c, d|z) = P(a|z)P(b|z)P(c|z)P(d|z).$$

Thus, if we know the value of Z, then none of A, B, C, D carry any additional information about each other's distributions. This is a form of a "naive Bayes" model, which we will discuss later below.

Example 2: First-Order Markov Models: Consider a sequence of random variables $X_1, \ldots, X_t, \ldots, X_T$ where the random variable at position t in the sequence is indexed by $t = 1, \ldots, T$. For simplicity all random variables $X_t, 1 \leq t \leq T$, are assumed to take values from the same set (whether discrete or real-valued). As an example consider X_t representing the tth word in a spoken sentence or a text document—in this case the number of possible values for each X_t is the number of words that the model knows about and could be as large as 50,000 or 100,000 (e.g., in real-world speech recognition or text modeling applications).

The sequence $X_1, \ldots, X_t, \ldots, X_T$ is defined to have first-order Markov property if the following property holds true:

$$P(x_{t+1}|x_t, x_{t-1}, \ldots, x_1) = P(x_{t+1}|x_t), \quad 1 \leq t \leq T-1$$

which is just another way of stating that the variable X_{t+1} is conditionally independent of all variables X_{t-1} back to X_1, conditioned on X_t. This assumption is sometimes stated as "the future only depends on the present and not on the past" (this is not true in the real-world in general of course, its just an assumption that is sometimes useful).

We can see that for modeling sequences of words that making an assumption such as this one (or some type of assumption of limited dependence) is almost essential from a practical viewpoint. For the case of a first-order Markov assumption it means we only have to model dependencies between pairs of successive words, and not the joint distribution of long sequences of words (of which there are an exponential number of such sequences as the length grows).

However, the price we pay, by only modeling adjacent words, is that we cannot for example model some of the natural structure of language, such as various aspects of grammar and phrasing, which tend to lead to longer-range dependencies. Nonetheless, the simple first-order Markov model can be very useful in practice.

Example 3: Higher-Order Markov Models: We can generalize the Markov model above to a kth order Markov model, where we assume that

$$P(x_{t+1}|x_t, x_{t-1}, \ldots, x_1) = P(x_{t+1}|x_t, \ldots, x_{t-k+1})$$

for some fixed integer $k > 0$. For $k = 1$ we get the first order Markov model. For $k = 2$ we get a 2nd-order Markov model where the next state now depends on the previous *two* states, and so on with 3rd order, 4th order, etc, Markov models. These models are also sometime referred to as n-gram models, where $n = k + 1$.

The conditional independence assumption being made in a kth order Markov model is that the next state is conditionally independent of earlier states given the k preceding state values. (Here we are using the word "state" to refer to the value of the variables X_t, a common convention when describing Markov chains). Modeling of DNA sequences in bioinformatics is (for example) one area where such high-order models can be used: in that case, each position X_t can take one of 4 possible values `A, G, T, C`, and we can afford to build models that are richer than simple first-order dependency.

However, the number of probabilities that we require for our model increases exponentially with k since we need to specify an exponentially increasing number of probabilities as k increases, on the order of m^{k+1}. Various interesting extensions are possible for problems where m is very large (such as modeling words in text), such as clustering words together into groups and modeling the conditional probabilities at the group level, and also searching for specific combinations of values that have higher order dependency (rather than assuming all combinations have high-order dependency) and then representing these dependencies in a tree-structured Markov model.

More recently, sequential models in deep learning have produced significant improvements in sequential language modeling, by using the notion of low-dimensional latent representations (such as embeddings) that can efficiently represent historical information in a sequence that is relevant to predicting what comes next. In effect in these models, the conditional distribution $P(x_{t+1}|x_t, x_{t-1}, \ldots, x_1)$ is approximated as $P(x_{t+1}|f(x_t, x_{t-1}, \ldots, x_1))$ where the $f()$ function produces a real-valued vector summary of the history of the sequence up to time t, e.g., as in the hidden state representations learned by recurrent neural networks and transformer models.

2.4 The Naive Bayes Model

The term **naive Bayes model** is used to refer to a particular type of conditional independence model where we have a set of d random variables[1] $X_1, \ldots, X_d$ that are assumed to be conditionally independent given a discrete random variable C. The term "naive Bayes" was originally used for this model to convey the idea that the model is "naive" in terms of its modeling assumptions.

A typical application of the naive Bayes model in machine learning is where $X_1, \ldots, X_d$ consists of a set of d discrete random variables that we can measure (sometimes called the **features** or **attributes**), with another discrete-valued variable C taking values $\{1, \ldots, m\}$ (called the class variable) that represents a property of the object that is not directly observed and must be inferred. The probabilities of different values of C can be calculated via Bayes rule as follows:

$$\begin{aligned}
P(C = j|x_1, \ldots, x_d) &= \frac{P(x_1, \ldots, x_d|C = j)P(C = j)}{P(x_1, \ldots, x_d)} \\
&\quad \text{(by Bayes' rule)} \\
&= \frac{\left(\prod_{i=1}^{d} P(x_i|C = j)\right)P(C = j)}{P(x_1, \ldots, x_d)} \\
&\quad \text{(invoking the conditional independence assumption)} \\
&= \frac{1}{\alpha}\left(\prod_{i=1}^{d} P(x_i|C = j)\right)P(C = j), \quad 1 \leq j \leq m \\
&\quad \text{(where } \alpha \text{ is a proportionality constant independent of } j\text{).}
\end{aligned}$$

Notice that we can compute $P(C = j|x_1, \ldots, x_d)$ as being proportional to a product of simpler individual terms $P(x_i|C = j)$ (by virtue of the conditional independence assumption) and $P(C = j)$.

Sidenote on notation: be aware of the "mixing" of notation here: $C = j$ refers to the jth value of class variable C, whereas x_i refers to some value of variable X_i. This type of overloading of notation is hard to avoid without introducing superscripts (which brings its own problems in terms of complexity of notation), but the interpretation in general should be reasonably clear from the context.

A full joint distribution (without any independence assumption) over $X_1, \ldots, X_d$ and C would require order of mK^d probabilities to specify, if each of the X_i variables took K values and C takes m values. Even for relatively moderate values of K and d this is clearly impractical. On the other hand, with the naive Bayes (conditional independence) assumption we need only on the order of dmK probabilities in total.

[1] Note the change in notation here where the subscript i in X_i indicates variable i, and x_i represents a possible value for variable X_i (rather than the ith value of variable X, as we had in Note Set 1).

Example 4: Medical Diagnosis: The naive Bayes model has been widely used in the past for building simple models for AI problems like automated medical diagnosis. The X_i variables can represent different symptoms or test results for example and the C variable represents the disease variable. As well as the advantage of requiring far fewer parameters than a full joint distribution, the naive Bayes model is also quite easy to interpret, since if we write the product form of equations above in log form (by taking the log of each side) we get:

$$\log P(C = j|x_1, \ldots, x_d) = \sum_{i=1}^{d} \log P(x_i|C = j) + \log P(C = j) - \log K$$

where K is a constant independent of C.

This can be interpreted as a simple **additive model**. If we make a prediction with this type of model, given a set of $x_1, \ldots, x_d$ values, we see which of the terms $\log P(x_i|C = j)$ had the most impact on the final answer. For example, if we measure a value x_i for a feature such that $P(x_i|C = j)$ is close to zero (i.e., according to the model it is highly unlikely that we would observe that value of x_i if the class variable is $C = j$), then $\log P(x_i|C = j)$ will be a very large negative number in this case and this will tend to make $\log P(C = j)$ very negative (and $P(C = j)$ very small) relative to the other possible class values (assuming that the value x_i is not so unlikely when conditioned on the other class values). Thus, the contribution of individual features can be interpreted in the final outcome because of the additive nature of the model.

Of course the limitation of the naive Bayes model is the fact that it ignores any interactions among the features and how these interactions can affect the probability of $C = j$. For example, "exclusive-OR" types of relations cannot be modeled, such as a person being likely to have the disease if they have either symptom X_1 or symptom X_2 but being *very unlikely* to have the disease if they exhibit both symptoms at the same time. Keep in mind, however, that if the occurrence of both symptoms together is relatively rare, or if modeling of $P(C = j|...)$ is dominated by other variables, then not modeling such interactions may have relatively little impact overall in terms of the performance of the model overall.

Example 5: Spam Email Classification: One of the very early successes of machine learning was the application of the naive Bayes model (back around 2005) in both open-source and commercial spam email filters. For many years most spam filter algorithms used a naive Bayes model to classify incoming emails into two classes, "spam" or "not spam", as represented by a binary class variable C. The feature variables $X_1, \ldots, X_d$ represent the occurrence of different words or phrases in the text. For example X_i could represent the presence or absence of the phrase `free offer` in the email, or the number of times this phrase occurs. Selection of which phrases to include in the model is of course somewhat of an art, but there are techniques that can search for good discriminative features to include given a set of training data. Having large sets of training data is of course not a problem given that we are all inundated with both spam and non-spam emails on a daily basis. For real-world spam filters there are of course many other features besides the words in the email that can be taken into account (header information, subject line information, HTML structure, sender address, etc) but even with these additional features naive Bayes models have been frequently (and successfully) used in practice.

3 Directed Graphical Models or Bayesian Networks

Conditional independence is a very useful general framework for structuring a probability model in terms of how our variables are connected in a model. We can take this idea a bit further and use graph theory to provide a formal modeling language for specifying and computing with sets of random variables that have various dependence and independence relations among them.

3.1 Definition of a Directed Graphical Model

Consider a set of d random variables $X_1, \ldots, X_d$, assumed discrete-valued for now, but all of what we will say below also applies to the continuous or mixed cases.

In a directed graphical model[2] we represent each random variable in our model by a node in a directed graph. A directed edge exists in the graph between X_i and X_j if X_j directly depends on X_i. Paths consisting of directed cycles are not allowed in the graph[3]. This means that a more accurate term for directed graphical models is "directed acyclic graphical" models, or DAG models, but we will drop the "acyclic" term in the name for simplicity.

A directed graphical model uses a directed graph to represent a specific factorization of the joint distribution as follows:

$$P(X_1, \ldots, X_d) \;=\; \prod_{i=1}^{d} P(X_i|\text{parents}(X_i)) \tag{1}$$

where parents(X_i) are the parents of the node X_i in the directed graph, i.e., the set of nodes that point to variable X_i. For example, if we have a graph with an edge from A to C and an edge from B to C, and no other edges, then A and B have no parents in this graph, C has two parents A and B, and the joint distribution specified by this graph is

$$\begin{aligned} p(a,b,c) &= P(a|\text{parents}(A))\; P(b|\text{parents}(B))P(c|\text{parents}(C)) \\ &= P(a)P(b)P(c|a,b). \end{aligned}$$

Thus, we can "read off from the graph" the joint distribution as a product of variables and their parents.

[2]"Bayesian network" is another name used to describe "directed graphical models"—they are effectively the same thing.

[3]This constraint comes from the factorization property of joint distributions where we can have terms one variable dependent on another, or vice-versa, but not both dependent on the other in the same factorization.

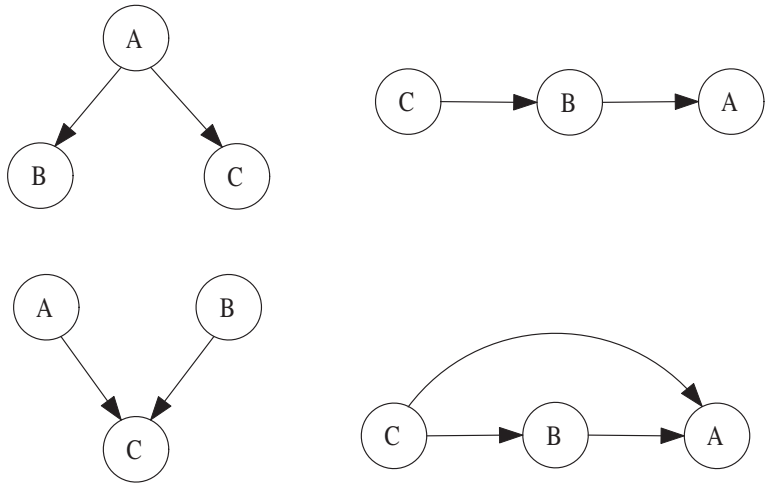

Figure 1: Examples of different graphical models for 3 random variables A, B, and C.

Example 6: Different Directed Graphical Models for 3 Variables: Consider 3 variables A, B, C. We can define a number of different simple graphical models for these three variables as follows (see Figure 1):

- The independence model, $P(a, b, c) = P(a)P(b)P(c)$ consists of the empty graph (since none of the variables have any parents).
- We can have 3 different "naive Bayes"-type conditional independence models, with any 2 of the variables being conditionally independent given the other, and the conditioning variable being the parent of the other 2.
- We can also have simple Markov chains, e.g., $P(a, b, c) = P(a|b)P(b|c)P(c)$ where C points to B which points to A.
- Another model is the "multiple cause" model, e.g., where the "causes" A and B point to a common "symptom" C. This is useful for example in modeling two medical conditions that can occur independently in a patient, and that have a common symptom.
- The model with no conditional independence relations at all (sometimes referred to as the **saturated model**) can be described in graphical form by recalling the factorization property from Note Set 1 that holds for all distributions and densities, e.g., $P(a, b, c) = P(a|b, c)P(b|c)P(c)$. This is equivalent to a graph where B and C point to A, C points to B, and nothing points to C. We could describe this same saturated model in different ways depending on the ordering. In fact for any arbitrary ordering of the variables the saturated model is equivalent to a graph where each variable has as parents *all* of the variables that precede it in the ordering.

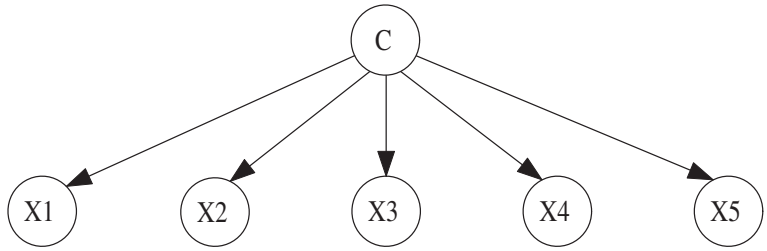

Figure 2: The graphical model representing a naive Bayes model with a class variable C and 5 features $X_1, \ldots, X_5$.

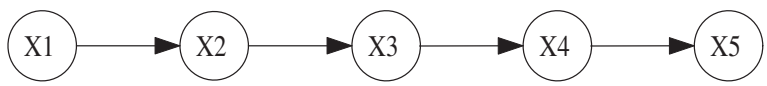

Figure 3: The graphical model for a Markov chain defined on 5 variables.

The graph-based framework is useful because it allows us represent, visualize and communicate the **structure** of a multivariate probability model in a systematic manner. There is also a rich theory that links the computational complexity of (a) performing calculations with the probability model, such as marginalization (summing out certain variables), with (b) the intrinsic structure of the graph. For example, graphs that can be represented as trees, or very sparse graphs, tend to be much more efficient for computation versus graphs where multiple directed paths exist between pairs of nodes and/or dense graphs in general. We will not dwell on the rich theoretical framework of graphical models here but instead use graphical models as a convenient mechanism for describing probability models that have structure in them.

Example 7: The Directed Graphical Model for Naive Bayes: The graphical model for naive Bayes is very simple (Figure 2), a single class variable as the single parent of d child nodes, with a directed edge from C to each of $X_1, \ldots, X_d$. From the general definition in Equation 1 of the joint distribution implied by the structure of a graphical model, we have

$$P(C, X_1, \ldots, X_d) = P(C) \prod_{i=1}^{d} P(X_i|C)$$

Example 8: The Directed Graphical Model for a Markov Chain: We defined earlier the Markov chain property for a sequence of random variables $X_1, \ldots, X_t, \ldots, X_T$. The graphical model is again very simple (Figure 3), consisting of a graph in the form of a chain with a directed edge from each X_t to X_{t+1} and with X_1 having no parents. Once again, from the general definition in Equation 1 of the joint distribution, implied by the structure of a graphical model, we have

$$P(X_1, \ldots, X_T) = P(X_1) \prod_{t=2}^{T} P(X_t | X_{t-1}) \tag{2}$$

The graphical model formalism provides a convenient mechanism for describing sets of conditional dependence relations. Given the structure of a graphical model, the precise way to characterize independence relations from the graph is to use a concept called **d-separation**. For general graphs this is a bit more complex than we need to delve into in these notes[4].

However, if we take the simpler case of graphs where each node has at most a single parent (e.g., chains, trees) then there is an easy way to read off conditional independence relations. In such graphs, lets say that in the graphical model that some variable X_j is on a path (directed or not) between some variable X_i and some subset S of variables. If this is the case we can then say that X_i is conditionally independent of all variables in the subset S given X_j. For example, from Figure 3 we can state that variable X_1 is conditionally independent of the subset $S = \{X_4, X_5\}$ given X_3, and so on. An intuitive idea is to think of the graph as a physical system where nodes are connected via edges: if we fix a node to a particular value then it "blocks" the nodes on either side of it, i.e., we can change the variables on either side, but this effect won't be propagated through the blocked node to the other side.

Thus, there is an equivalence between the structure of the graph, the conditional independence relations implied by the graph structure, and the factorization of the joint probability into a product of local probability tables involving a variable and its parents.

Note that it is tempting to attach a semantic interpretation to the direction of the arrows in a directed graphical model. It is true that in some situations the directionality has a natural interpretation, such as in modeling sequences of words or modeling family relationships in genetic pedigree models, or more generally for problems that involve some natural temporal ordering. However in many problems in machine learning the directionality need not have any obvious semantic (or causal) interpretation. For example, we could represent a Markov chain by factoring the joint distribution eihter forwards or background (i.e., the arrows could go in either direction and still represent the same Markov properties). Thus, in general, the arrows in directed graphical models represent stochastic dependencies between variables rather than necessarily a causal relationship between parent and child variables.

[4]For optional extra reading, students interested in more details about d-separation in general can refer to Section 8.5 in Mathematics for Machine Learning or various online tutorials (e.g., `https://networkx.org/documentation/stable/reference/algorithms/d_separation.html`).

4 The Multivariate Gaussian Model

4.1 Definitions and Properties

The most widely-known multivariate model for real-valued variables is the multivariate Gaussian model, which we discuss below. Consider a set of real-valued random variables $X_1, X_2, \ldots, X_d$, where $x_1, x_2, \ldots, x_d$ will be used to denote generic values for this set of random variables. We will use the shorthand notation $\underline{x} = (x_1, x_2, \ldots, x_d)$ to denote a d-dimensional vector of values. The standard convention is to assume that the vector is a column vector, i.e., has dimension $d \times 1$.

The multivariate Gaussian density function is defined as:

$$p(\underline{x}) = p(x_1, x_2, \ldots, x_d) = \frac{1}{(2\pi)^{\frac{d}{2}} |\mathbf{\Sigma}|^{\frac{1}{2}}} e^{-\frac{1}{2}(\underline{x}-\underline{\mu})^T \mathbf{\Sigma}^{-1} (\underline{x}-\underline{\mu})}$$

where $\underline{\mu} = (\mu_1, \ldots, \mu_d) = (E[x_1], \ldots, E[x_d])$ is a $d \times 1$ dimensional vector of mean values and $\mathbf{\Sigma}$ is a $d \times d$ covariance matrix with entries $\sigma_{ij} = cov(X_i, X_j)$ and with determinant $|\Sigma|$. The diagonal terms in the covariance matrix, $\sigma_{ii} = cov(X_i, X_i)$ are the individual variances $\sigma_i^2 = Var(x_i)$ for each of the X_i variables.

Although at first this expression looks rather formidable, it can be broken down into a simpler version, namely

$$p(\underline{x}) = \frac{1}{C}\, e^{-dist_{\mathbf{\Sigma}}(\underline{x}, \underline{\mu})}$$

where C is just a normalization constant and

$$dist_{\mathbf{\Sigma}}(\underline{x}, \underline{\mu}) = \frac{1}{2}(\underline{x} - \underline{\mu})^T \mathbf{\Sigma}^{-1} (\underline{x} - \underline{\mu})$$

defines a non-negative scalar distance[5] between the vector $\underline{x}$ and the mean $\underline{\mu}$. This distance is a generalization of the standard notion of Euclidean distance. If $\mathbf{\Sigma} = I$, the identity matrix, we get $(\underline{x} - \underline{\mu})^T (\underline{x} - \underline{\mu})$, which is the square of the Euclidean distance $\sum_{i=1}^{d} (x_i - \mu_i)^2$ between $\underline{x}$ and $\underline{\mu}$. If $\mathbf{\Sigma}$ is diagonal, then $\mathbf{\Sigma}^{-1}$ is also diagonal, and the distance becomes a weighted Euclidean distance, $\sum_{i=1}^{d} w_i (x_i - \mu_i)^2$ where each weight $w_i = \frac{1}{\sigma_i^2}$. Thus, the contribution to the overall distance is scaled by the inverse of the variance in each dimension—this is equivalent to pre-scaling all the X_i variables to have unit variance, so that the contributions to the overall distance from each dimension would have equal weight from a variance viewpoint. More generally, when $\mathbf{\Sigma}$ is not diagonal, $dist(\underline{x}, \underline{\mu}; \mathbf{\Sigma})$ defines a distance measure that takes into account the covariance (or correlation) among the variables, e.g., so that if two variables X_i and X_j were very highly correlated then the contribution of the distances in dimensions x_i and x_j would be down-weighted since they are really measuring the same thing.

In general, the Gaussian multivariate density function has the following properties:

- it has a single (unimodal) peak at $\underline{x} = \underline{\mu}$ (where $dist_{\mathbf{\Sigma}}(\underline{x}, \underline{\mu}) = 0$);

[5] The square root of this quantity is referred to as the Mahalonobis distance.

- the height of the density function decreases exponentially as a function of $dist_{\mathbf{\Sigma}}(\underline{x}, \underline{\mu})$, as $\underline{x}$ moves further away from $\underline{\mu}$;
- the iso-contours of the density function (loci of points in $\underline{x}$ space that all have the same value of $p(\underline{x})$) will be ellipsoidal in the general case (or "hyperellipsoidal" for $d > 2$). If $\mathbf{\Sigma}$ is diagonal, the major axes of the hyperellipse will be axis-parallel, and if $\mathbf{\Sigma} = I$ then the isocontours are circles;
- any subset of X_i's are also Gaussian, e.g., $p(X_1, X_3)$ is Gaussian as is $p(X_2)$, etc. Conditionals are also Gaussian, e.g., $p(X_1|x_2)$ is Gaussian if $p(X_1, X_2)$ is Gaussian.

4.2 Graphical Models with Real-Valued Variables

In general we can define conditional independence relations and graphical models for sets of real-valued random variables just we did in our examples for discrete random variables. For example, we can impose conditional independence structure on the multivariate Gaussian model above if we wish to.

A simple example would be a type of Gaussian model with Markov structure, where each random variable X_{j+1} only has a single parent X_j. The joint density of such a model, equivalent to what we saw earlier for a discrete-valued Markov chain, can be written as:

$$p(X_1, \ldots, X_d) = p(X_1) \prod_{j=2}^{d} p(X_j|X_{j-1}) \tag{3}$$

The pairwise dependencies $p(X_j|X_{j-1})$ would now be conditional Gaussian densities rather than conditional probability tables. The graphical model has the same structure as Figure 3 but the nodes now represent real-valued random variables and the directed edges represent conditional densities.

We can think of this Gaussian-Markov model as being analogous to a physical chain of coupled variables, where the pairwise correlations govern the "strength" of the dependence between each pair of variables in the chain. And if we "wiggle" the first element of the chain X_1, we expect that this information will propagate all the way through the chain to X_d, even if only very weakly. This implies (intuitively) that the covariance matrix Σ of this chain-structured multivariate graphical model is itself dense (i.e., wiggling any variable will have some effect on all others in the chain). So we can't use the covariance matrix necessarily to tell us directly about conditional independence relations (this is one reason why thresholding covariance values in a covariance matrix is not necessarily a good way to uncover structural dependencies between variables). Interestingly, however, it turns out that conditional independence reveals itself via the presence of zeros in the inverse of the covariance matrix $\mathbf{\Sigma}^{-1}$, for reasons that are somewhat beyond the scope of this set of notes.

Many of the large-scale applications of graphical models tend to involve sets of variables that are all discrete, or all Gaussian (if real-valued), or some mix of the two. The theory and methodology for graphical models with non-Gaussian distributions is tricky (both theoretically and computationally), although techniques such as copulas and kernel density methods have been pursued. One exception is if the non-Gaussian

real-valued variables are at the "edge" of the model and/or are always observed (i.e., we never need to marginalize over them): in those situations we can generally use whatever type of distributional assumption we wish.