CSE 312: Foundations of Computing II
Advanced Topics Session #5

The Multivariate Normal Distribution, Bayesian Networks

Lecturers: Joshua Fan, Alex Tsun

Date: May 16, 2017

1 Jointly Normal Random Variables

For this section, we will treat a constant c as a normal random variable with mean c and variance 0, even though its pdf is undefined. Therefore, we will have $aX + b$ being normal if X is normal, and even if $a = 0$.

Two random variables X and Y are **jointly normal** if and only if they can be expressed in the form

$$(X, Y) = (aU + bV, cU + dV)$$

where $a, b, c, d \in \mathbb{R}$ and U and V are independent and normally distributed.

If X and Y are jointly normal, then $W = sX + tY$ is also normally distributed.

From Wikipedia, if X and Y have some normal distribution, it does not imply that (X, Y) are jointly normal. As a counterexample, let $X \sim N(0,1)$ and $Y = X$ if $|X| > c$ and $Y = -X$ if $|X| \leq c$, for some $c > 0$.

Theorem: For jointly normal random variables (X, Y), if $Cov(X, Y) = 0$ (or equivalently, $Cor(X, Y) = \rho_{X,Y} = 0$), then $X \perp Y$. Note that we showed earlier, that if two random variables S, T are independent, then $Cov(S, T) = Cor(S, T) = 0$. Therefore, this is a special case of the converse being true.

2 The Multivariate Normal Distribution

In the special case with two jointly normal random variables, where $X \sim N(\mu_X, \sigma_X^2)$ and $Y \sim N(\mu_Y, \sigma_Y^2)$ and correlation $\rho \equiv \rho_{X,Y} = \frac{Cov(X,Y)}{\sqrt{Var(X)Var(Y)}} = \frac{Cov(X,Y)}{\sigma_X \sigma_Y}$, we define the joint probability density function of X and Y as

$$f_{X,Y}(x, y) = \frac{1}{2\pi\sigma_X\sigma_Y\sqrt{1-\rho^2}} e^{-\frac{1}{2(1-\rho^2)} z}$$

where

$$z = \frac{(x-\mu_X)^2}{\sigma_X^2} - \frac{2\rho(x-\mu_X)(y-\mu_Y)}{\sigma_X\sigma_Y} + \frac{(y-\mu_Y)^2}{\sigma_Y^2}$$

In the special case where $X \perp Y$ or equivalently $\rho = 0$ from the earlier theorem,

$$f_{X,Y}(x,y) = f_X(x)f_Y(y) = \frac{1}{\sigma_X\sqrt{2\pi}} e^{-\frac{1(x-\mu_X)^2}{2\ \sigma_X^2}} \frac{1}{\sigma_Y\sqrt{2\pi}} e^{-\frac{1(y-\mu_Y)^2}{2\ \sigma_Y^2}}, \qquad x, y \in \mathbb{R}$$

The **mean vector $\boldsymbol{\mu}$** is given by

$$\boldsymbol{\mu} = \begin{bmatrix} \mu_X \\ \mu_Y \end{bmatrix}$$

The **covariance matrix $\boldsymbol{\Sigma}$** is given by

$$\Sigma = \begin{bmatrix} \sigma_X^2 & \rho\sigma_X\sigma_Y \\ \rho\sigma_X\sigma_Y & \sigma_Y^2 \end{bmatrix}$$

In this case, we say that (X, Y) has the **bivariate normal distribution with mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$**, and we write $(X, Y) \sim N_2(\boldsymbol{\mu}, \Sigma)$.

All the standard properties of joint distributions apply – including expectation, marginal and conditional distributions, etc.

Now we extend our analysis to the general case: suppose $(X_1, \ldots . X_n)$ are jointly normally distributed random variables with mean vector $\boldsymbol{\mu} \in \mathbb{R}^n$ and covariance matrix $\Sigma \in \mathbb{R}^{n \times n}$, with $\Sigma_{ij} = Cov(X_i, X_j)$. Then, the joint density function of the random vector $\boldsymbol{X} = (X_1, \ldots, X_n)$ evaluated at a vector $\boldsymbol{x} = (x_1, \ldots, x_n) \in \mathbb{R}^n$ is given by

$$f_{\boldsymbol{X}}(\boldsymbol{x}) = \frac{1}{\sqrt{(2\pi)^n |\Sigma|}} e^{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^T \Sigma^{-1} (\boldsymbol{x}-\boldsymbol{\mu})}$$

In this case, we say $(X_1, \ldots, X_n)$ has the **multivariate normal distribution with mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$**, and we write $(X_1, \ldots, X_n) \sim N_n(\boldsymbol{\mu}, \Sigma)$. In the case where $X_1, \ldots, X_n$ are iid (or equivalently, pairwise uncorrelated), the covariance matrix is diagonal, and the joint density is the product of the individual densities.

We often assume features in machine learning or random variables of interest that we measure are multivariate Gaussian (i.e., multivariate normal). Thus it is important to recognize and know basic properties of this distribution. The two most important multivariate distributions by far are the multivariate normal and multinomial distributions.

3 Bayesian Networks

Bayesian networks comprise an important probabilistic model that is used to make inferences about a complex system for which we have statistical data. For example, they are used for

- medical diagnosis
- analyzing genetic expression data
- predicting when/where crime or terrorism is likely to occur
- protecting endangered species by modeling their environment
- estimating risk (for example, what's the probability you'll get into an automobile accident)
- information retrieval
- forecasting the weather
- interpreting legal evidence
- etc…

They are an incredibly powerful tool, and you will learn much more about them if you take courses such as CSE 473 (Artificial Intelligence).

Notation alert: To save space, I sometimes abbreviate $P(X = x)$ as $p(x)$. Similarly, I will abbreviate $P(X = x \cap Y = y)$ as $p(x, y)$.

4 Modelling a joint distribution

In most of the examples we've covered in class, we were analyzing the distribution of a single random variable (such as the outcome of a die roll, whether a person has a disease or not, etc.). However, most real-world situations involve more than one random variable. How do we handle this?

If the random variables are discrete, we can start with the **joint distribution**, which assigns a probability to **every possible combination** of outcomes. For example, let's say we're modelling flu/allergy symptoms and we have five random variables: **Flu, Allergy, Sinus, Headache, Nose**. Each of these random variables is binary (true/false). The joint distribution could look something like this:

Flu	$Allergy$	$Sinus$	$Headache$	$Nose$	$p(f, a, s, h, n)$
T	T	T	T	T	0.01
T	T	T	T	F	0.03
T	T	T	F	T	0.0001
T	T	T	F	F	0.19
T	T	F	T	T	0.08
…	…	…	…	…	…

Exercise: How many probabilities do we need to store? What do the probabilities in the right column need to sum to?

Solution: There are 5 binary random variables (each with 2 possible outcomes), so there are $2^5 = 32$ possible combinations of outcomes. Since we've included every possible setting of each random variable, the probabilities must sum to **1**. Since all the probabilities sum to 1, in practice we only need to store **31** probabilities: the 32nd probability is found by subtracting all other probabilities from 1.

The number of probabilities is **exponential** in the number of random variables. If we have just 50 binary random variables, then we need $2^{50} \approx 1.12 \times 10^{15}$ probabilities. We probably don't have enough data to even guess these probabilities! It seems like we're just completely screwed here. Even if we use the chain rule:

$$\begin{aligned} p(f,a,s,h,n) &= p(n|f,a,s,h)p(f,a,s,h) \\ &= p(n|f,a,s,h)p(h|f,a,s)p(f,a,s) \\ &= p(n|f,a,s,h)p(h|f,a,s)p(f|a,s)p(a|s)p(s) \end{aligned}$$

We're still stuck with complex probabilities (probability of something, GIVEN a ton of other variables). It would be nice if we could get rid of some of those variables in the conditions. If you recall, we faced the same problem in Naïve Bayes, and we addressed it with **conditional independence**!

5 Conditional independence saves the day!

Recall a definition of conditional independence: random variables X and Y are **conditionally independent** given Z if

$$\forall x, y, z: \; p(x|z,y) = p(x|z)$$

Consider a simple example: we have 3 binary random variables:

- **Rain** (true iff it's raining)
- **Traffic** (true iff there is traffic)
- **Umbrella** (true iff I am using my umbrella)

Exercise: just using your intuition, are there conditional independences in this situation? Which variables?
Solution: rain causes both traffic and umbrella usage, so the probability of each increases if we're given that it is raining. However, **Traffic and Umbrella are conditionally independent given Rain**. Once we know that it's raining or not raining, there is no relation between them: the correlation is purely caused by the fact that both are associated with rain.

Now let's use this to simplify the joint distribution.

$$\begin{aligned} p(Traffic, Rain, Umbrella) &= p(Rain)p(Traffic|Rain)p(Umbrella|Rain, Traffic) \\ &= p(Rain)p(Traffic|Rain)p(Umbrella|Rain) \end{aligned}$$

(The last line follows since Umbrella is conditionally independent of Traffic, given Rain.)

6 Bayesian network semantics

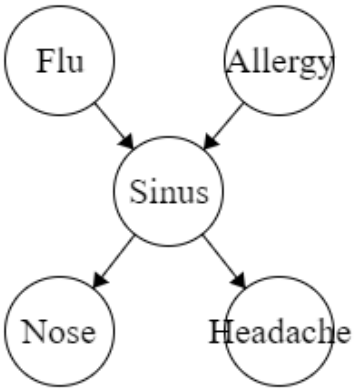
Bayesian networks are a "technique for describing complex joint distributions (models) using simple, local distributions (conditional probabilities)". They consist of both a directed acyclic graph, and probability distributions for each node.

- **Nodes** represent random variables (with domains). They can be assigned (observed) or unassigned (unobserved)
- **Arcs** represent direct influence between variables. Intuitively, you can think of them as causation: an arrow pointing from variable A to B suggests that A directly influences B. Formally, they encode conditional independences as we'll see later.

Exercise: draw the graph for a Bayesian network in the above example with Flu, Allergy, Sinus, Headache, Nose. Assume we know these facts from our prior knowledge:

- The flu causes sinus inflammation
- Allergies cause sinus inflammation
- Sinus inflammation causes a runny nose
- Sinus inflammation causes headaches

Solution:



This is the graph portion of a Bayes' net. But in addition to this graph, which specifies the relationships among the random variables, we also have a conditional probability table for each random variable. This table stores the probability of the variable having each value **given all possible configurations of the parents**. For example, the table for "Sinus" would store the probability of "Sinus = true" given every combination of the parents' values (in this case, Flu and Allergy).

This seems nice – instead of conditioning on every other variable, we only need to condition on the variable's parents. But can we recover the joint distribution from this?

7 How the heck does this represent the joint distribution? Factoring!

By definition, all Bayesian networks make the following **conditional independence assumption:**

Local Markov Assumption: **A variable X is independent of its non-descendants, given its parents.**
(In other words, if we know the values of X's parents, the values of all other variables give us no additional information about the probability of X.)

Theorem 1: If a Bayesian network satisfies the Local Markov Assumption, and the variables are sorted in topological order, then for each variable x_i:

$$p(x_i|x_1, x_2, \dots x_{i-1}) = p\left(x_i \middle| Parents(x_i)\right)$$

Proof: We can split $x_1, x_2, \ldots x_{i-1}$ into parents of x_i and non-parents.

$$p(x_i|x_1, x_2, \ldots x_{i-1}) = p\big(x_i\big|Parents(x_i), NonParents(x_i)\big)$$

Since the variables are sorted in topological order, none of the previous variables $x_1, x_2, \ldots x_{i-1}$ are descendants of x_i. Now, we apply the Local Markov Assumption.

$$p\big(x_i\big|Parents(x_i), NonParents(x_i)\big) = p(x_i|Parents(x_i))$$

Whoa! So using this conditional independence assumption, we can in fact get rid of a lot of the variables we're conditioning on! Now, let's simplify the joint distribution.

Theorem 2: If a Bayesian network has variables $x_1, x_2, \ldots x_n$ and the Local Markov Assumption is satisfied,

$$p(x_1, x_2, \ldots x_n) = \prod_{i=1}^{n} p\big(x_i\big|Parents(x_i)\big)$$

Proof: If we start from the joint distribution again, by the chain rule we have:

$$p(x_1, x_2, x_3, \ldots x_n) = p(x_1)p(x_2|x_1)\ldots p(x_n|x_1, \ldots x_{n-1}) = \prod_{i=1}^{n} p(x_i|x_1, \ldots x_{i-1})$$

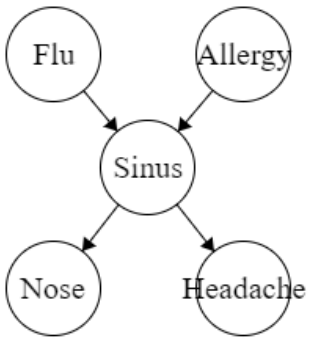
Using Theorem 1, this is equal to

$$\prod_{i=1}^{n} p\big(x_i\big|Parents(x_i)\big)$$

The Conditional Probability Tables in the Bayesian network store exactly this information (probability of a random variable given values for its parents)! Thus, we can get back any probability in the joint distribution!

This is the key concept behind Bayesian networks – using conditional independences can simplify our representation a lot! (If you count, we can represent the 5-variable flu/allergy distribution with just 10 probabilities, instead of 31. It's even more dramatic with more variables.)

Exercise: for the following Bayes' net, factor the joint distribution $p(f, a, s, n, h)$ into probabilities that the Bayesian network stores.

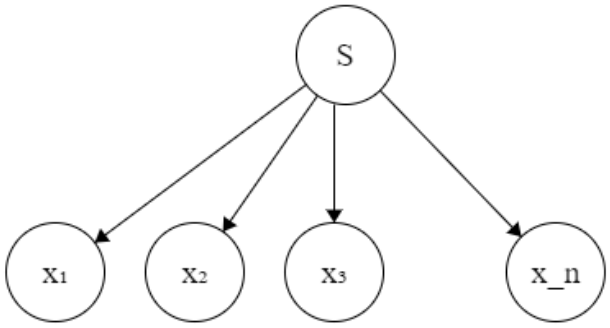


Solution:

$$p(f,a,s,n,h)$$
$$= p(f|Parents(F))p(a|Parents(A))p(s|Parents(S))p(n|Parents(n))p(h|Parents(h))$$
$$= p(f)p(a)p(s|f,a)p(n|s)p(h|s)$$

Exercise: Recall that in Naïve Bayes, we assume words $x_1, x_2, \ldots x_n$ in an email are conditionally independent of each other, given that we know whether the email is spam (event S). Draw a Bayesian network that reflects this, and factor the joint distribution accordingly. (Naïve Bayes is a special example of a Bayesian network.)

Solution:



$$p(S, x_1, x_2, \ldots x_n) = p(S)p(x_1|S)p(x_2|S)\ldots p(x_n|S) = p(S)\prod_{i=1}^{n} p(X_i|S)$$

(Whether the email is spam influences the probability of each word appearing, so there is an arrow from S to each word. Note that by the Local Markov Assumption, each x_i is independent from its non-descendants (other words) given its parent (S), which is the exact conditional independence assumption Naïve Bayes uses.)

8 Inference

Great, now we can find the probability of any combination of values for the random variables! But what about more routine inference tasks? For example, let's say we're given that Nose is true. What is the probability of Flu taking on a particular value?

Applying the definition of conditional probability:

$$P(F = x_F|N = true) = \frac{P(F = x_F, N = true)}{P(N = true)}$$

For illustration let's focus on the numerator. We sum over all possible values A, S, and H can take on.

$$P(F = x_F, N = true) = \sum_{x_A, x_S, x_H} P(F = x_F, N = true, A = x_A, S = x_S, H = x_H)$$

Remember from above (Theorem 2): in a Bayesian network, the joint distribution can be factored into the probability of each variable given its parents.

$$\sum_{x_A, x_S, x_H} P(F = x_F)P(A = x_A)P(S = x_S|F = x_F, A = x_A)P(H = x_H|S = x_S)P(N = true|S = x_S)$$

These terms are available in the Bayesian network's Conditional Probability Tables. However, in the worst case, we need to sum over an exponential number of combinations of variable values.

An optimization can be made: factor common terms out of the summations.

$$P(F = x_F)\sum_{X_A} P(A = x_A)\sum_{X_S} P(S = x_S|F = x_F, A = x_A)P(N = true|S = x_S)\sum_{x_H} P(H = x_H|S = x_S)$$

This is an example of an approach called **variable elimination**. There are other more sophisticated methods for inference (such as junction trees), but unfortunately, all are still exponential time in terms of the number of variables (in general). For complex Bayesian networks, we can instead perform approximate inference using **sampling-based methods**, where we generate a bunch of samples according to the Bayesian network probabilities, and then count the fraction of them that satisfy our criteria. This won't give us the exact probabilities, but it's sometimes the only way to estimate anything in a reasonable amount of time. This is an active area of research in Computer Science and Statistics: people are exploring ways to make probabilistic inference on Bayesian networks more efficient.

For more info on Bayesian networks, check out these slides:
http://courses.cs.washington.edu/courses/cse473/16sp/slides/cse473sp16-BayesNets.pdf
https://courses.cs.washington.edu/courses/cse446/17wi/slides/bayesnets-annotated.pdf