COMP 562: Introduction to Machine Learning

Lecture 9 : Information Theory, Bayesian Networks

Mahmoud Mostapha

Department of Computer Science

University of North Carolina at Chapel Hill

mahmoudm@cs.unc.edu

September 26, 2018

COMP562 - Lecture 9

Plan for today:

- Basics of Information Theory
- Conditional Independence
- Bayesian Networks

Information Theory

> The fundamental problem of communication is that of reproducing at one point either exactly or approximately a message selected at another point.
>
> *(Claude Shannon, 1948)*

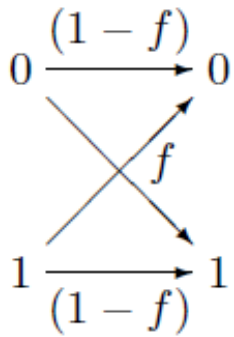
Information Theory tackles problems such as

- communication over noisy channels
- compression

Information Theory

Suppose we have a noisy channel that flips bits with probability f. Let x be message sent over the noisy channel and y received message.

$$x \;\begin{array}{c} 0 \to 0 \\ 1 \to 1 \end{array}\; y \qquad \begin{array}{llll} P(y=0\,|\,x=0) = 1-f; & P(y=0\,|\,x=1) = f; \\ P(y=1\,|\,x=0) = f; & P(y=1\,|\,x=1) = 1-f. \end{array}$$



Basics of Information Theory

Suppose you communicate to your friends using one of the four messages **a**,**b**,**c** and **d**.

You might opt to encode messages as[1]

- $\text{Enc}(\mathbf{a}) = 00$
- $\text{Enc}(\mathbf{b}) = 01$
- $\text{Enc}(\mathbf{c}) = 10$
- $\text{Enc}(\mathbf{d}) = 11$

Note that the length of each message is 2 bits (notation $|\text{Enc}(\mathbf{a})| = 2$)

Regardless of the message you want to communicate we always have to transmit 2 bits.

[1] this table is called a codebook

Expected/average message length

The expected length of message is

$$E_p[|\mathrm{Enc}(M)|] = \sum_{m \in \{\mathbf{a},\mathbf{b},\mathbf{c},\mathbf{d}\}} p(M = m)|\mathrm{Enc}(m)|$$

Since $|Enc(m)| = 2$ for all m, on average for each message you will use ... 2 bits

Basics of Information Theory

Suppose you knew something extra: the probability that a particular message will need to be transmitted.

$$p(M) = \begin{cases} 1/2, & M = \mathbf{a} \\ 1/4, & M = \mathbf{b} \\ 1/8, & M = \mathbf{c} \\ 1/8, & M = \mathbf{d} \end{cases}$$

Could you then take advantage of this information?

Basics of Information Theory

Use short codewords for frequent messages, longer codewords for infrequent messages

- $p(M = a) = 1/2$, $\text{Enc}(\mathbf{a}) = 0$
- $p(M = b) = 1/4$, $\text{Enc}(\mathbf{b}) = 10$
- $p(M = c) = 1/8$, $\text{Enc}(\mathbf{c}) = 110$
- $p(M = a) = 1/8$, $\text{Enc}(\mathbf{d}) = 111$

and expected message length is

$$1/2 * 1 + 1/4 * 2 + 1/8 * 3 + 1/8 * 3 = 1.75.$$

Hence, on average we save 0.25 bits per message using the above codebook.

Entropy

You can show that the codeword length assignment that minimizes the expected message length is $-\log_2 p(m)$.

Given a random variable M distributed according to p entropy is

$$H(M) = \sum_m p(M = m) \left[-\log_2 p(M = m)\right]$$

The entropy can be interpreted as number bits spent in communication using the optimal codebook.

For simplicity, we may write $H(p)$ when it is clear which random variable we are considering.

Conditional entropy

$$H(X) = \sum_m p(X = m) \left[-\log_2 p(M = m) \right]$$

If we have access to extra information Y then we can compute conditional entropy

$$H(X|Y) = \sum_y \sum_x p(X = x|Y = y) \left[-\log_2 p(X = x|Y = y) \right]$$

Entropy – characteristics

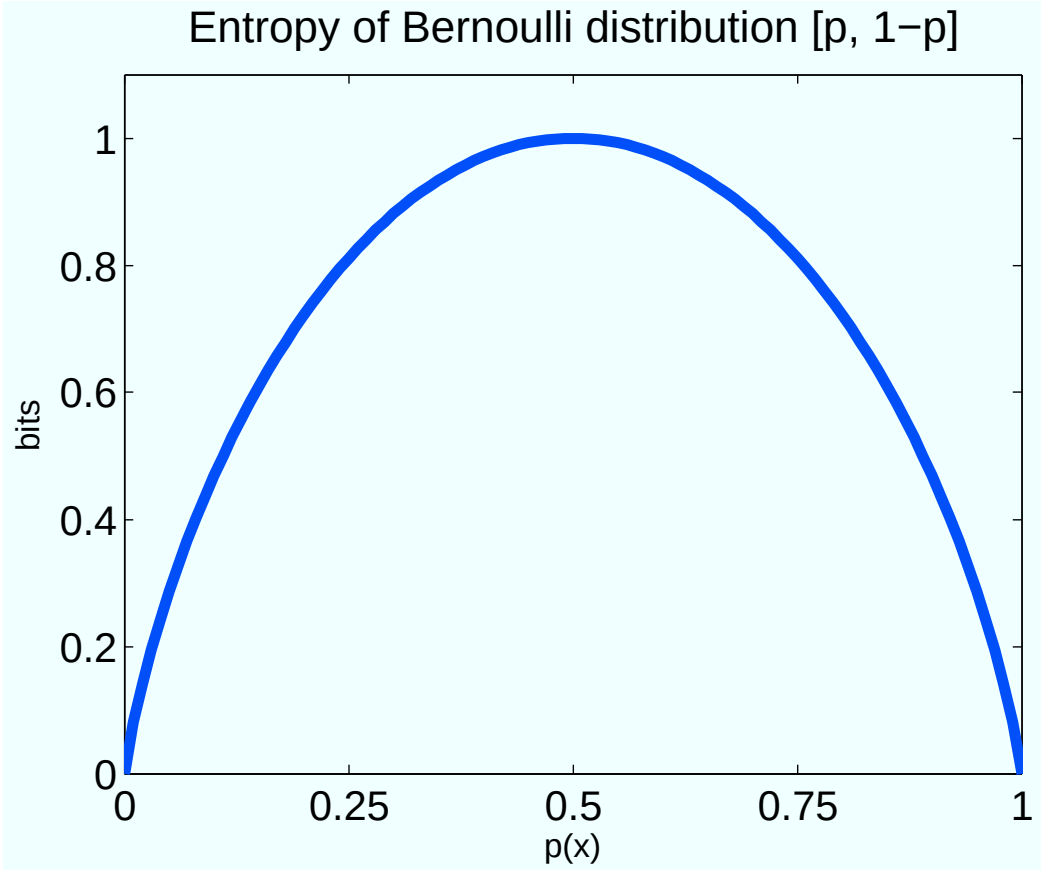
$$H(p) = \sum_m p(m) \left[-\log_2 p(m) \right]$$

Entropy is always non-negative

The more uniform the distribution the higher the entropy (I need 2 bits for 4 messages with prob. $1/4$).

The less uniform the distribution the lower the entropy (I need 0 bits if I always send the same message).

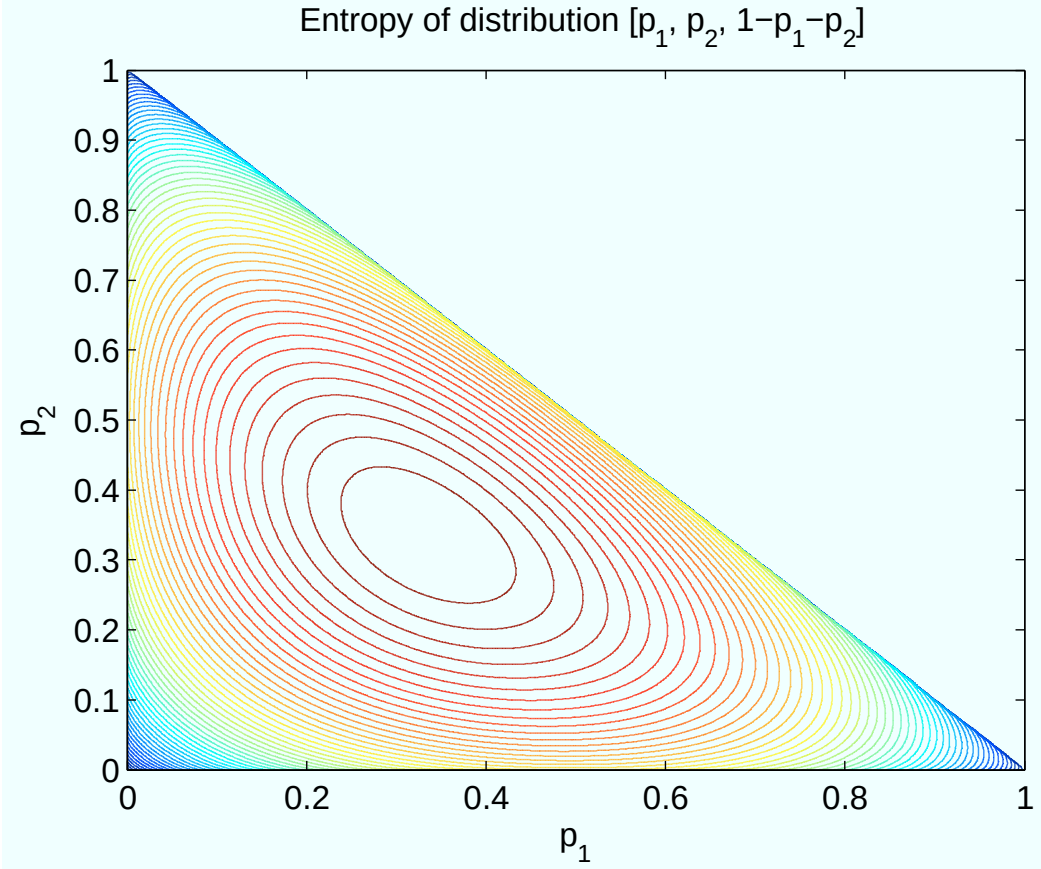
Entropy



Maximized for uniform distribution $p = \frac{1}{2}$

$$H\left(\left[\frac{1}{2}, \frac{1}{2}\right]\right) = \frac{1}{2}\left(-\log_2 \frac{1}{2}\right) + \frac{1}{2}\left(-\log_2 \frac{1}{2}\right) = \frac{1}{2}\log_2 2 + \frac{1}{2}\log_2 2 = 1$$

Entropy



$$H([p_1, p_2, 1-p_1-p_2]) = -p_1 \log_2 p_1 - p_2 \log_2 p_2 - (1-p_1-p_2) \log_2 (1-p_1-p_2)$$

For which p_1 and p_2 is entropy the largest? What is the value of entropy for that distribution?

Entropy recap

- Entropy is evaluated on probability distributions

$$H(p) = -\sum_m p(M = m) \log_2 p(M = m)$$

- Given a distribution of messages, entropy tells us the least number of bits we would need to encode messages to communicate efficiently.
- Length of a code for a message should be $-\log_2 p(m)$. Sometimes this can not be achieved.[2]
- Entropy is maximized for uniform distributions

$$H([\frac{1}{K} \dots \frac{1}{K}]) = \log_2 K$$

[2] If $p(m) \neq 2^{-c}$ we get fractional code lengths.

Cross-entropy

Suppose we were given message probabilities q.

We build our codebook based on q and the optimal average message length is $H(q)$.

But it turns out the q is incorrect and the true message probabilities are p.

Cross-entropy is the average message length under these circumstances

$$H(p, q) = -\sum_{m} p(m) \log q(m)$$

Cross-entropy

Cross entropy

$$H(p, q) = -\sum_m p(m) \log q(m)$$

is always greater or equal than entropy

$$H(p) = -\sum_m p(m) \log p(m).$$

Using wrong codebook is always incurs cost in communication – using longer codes for less frequent messages.

Kulback Leibler divergence

Had we known the true distribution p our average message length would be

$$H(p) = -\sum_m p(m) \log p(m)$$

we didn't and we are now on average using a *longer* message

$$H(p, q) = -\sum_m p(m) \log q(m).$$

How much longer?

$$H(p, q) - H(p) = -\sum_m p(m) \log q(m) + \sum_m p(m) \log p(m)$$

This difference is called Kullback-Leibler divergence

$$\mathrm{KL}(p||q) = H(p, q) - H(p) = -\sum_m p(m) \log q(m) + \sum_m p(m) \log p(m)$$

KL divergence

For discrete pdfs:

$$\mathrm{KL}(p||q) = \sum_x p(x) \log \frac{p(x)}{q(x)}$$

For continuous pdfs:

$$\mathrm{KL}(p||q) = \int_x p(x) \log \frac{p(x)}{q(x)} \mathrm{d}x$$

A couple of observations about KL-divergence

1. $\mathrm{KL}(p||q) \geq 0$
2. $\mathrm{KL}(p||q) = 0$ if and only if $p = q$
3. it is not symmetric $\mathrm{KL}(p||q) \neq \mathrm{KL}(q||p)$

Summary

- Entropy as a measurement of the number of bits needed to communicate efficiently.
- KL-divergence as a number of bits that could be saved by using the right distribution
- KL-divergence as a distance between distributions.

Maximum likelihood and KL minimization

Delta function or indicator function:

$$[x] = \begin{cases} 1, \text{if } x \text{ is true} \\ 0, \textit{otherwise} \end{cases}$$

Data/Empirical distribution:

$$f(\mathbf{x}) = \frac{1}{N} \sum_{i=1}^{N} [\mathbf{x}_i = \mathbf{x}]$$

Importantly:

$$\int_{\mathbf{x}} f(x) g(x) \mathrm{d}\mathbf{x} = \int_{\mathbf{x}} \left(\sum_{i=1}^{N} [\mathbf{x}_i = \mathbf{x}] \right) g(x) \mathrm{d}\mathbf{x} = \frac{1}{N} \sum_{i=1}^{N} g(x_i)$$

Maximizing likelihood is equivalent to minimizing KL

$$\mathrm{ALL}(\theta) = \sum_{i=1}^{N} \frac{1}{N} \log p(\mathbf{x}_i|\theta)$$

KL divergence between empirical distribution $f(\mathbf{x})$ and $p(\mathbf{x}|\theta)$:

$$\begin{aligned}
\mathrm{KL}(f||p) &= \int_{\mathbf{x}} f(\mathbf{x}) \log \frac{f(\mathbf{x})}{p(\mathbf{x})} \mathrm{d}\mathbf{x} \\
&= \int_{\mathbf{x}} \left(\frac{1}{N} \sum_{i=1}^{N} [\mathbf{x}_i = \mathbf{x}] \right) \log \frac{f(\mathbf{x})}{p(\mathbf{x})} \mathrm{d}\mathbf{x} \\
&= \sum_{i=1}^{N} \frac{1}{N} \log \frac{f(\mathbf{x}_i)}{p(\mathbf{x}_i)} = \underbrace{-\sum_{i=1}^{N} \frac{1}{N} \log p(\mathbf{x}_i)}_{-\mathrm{ALL}(\theta)} + \underbrace{\frac{1}{N} \log \frac{1}{N}}_{\text{const.}}
\end{aligned}$$

Maximizing likelihood is equivalent to minimizing KL

$$\mathrm{KL}(f||p) = -\mathrm{ALL}(\theta)$$

and hence

$$\operatorname*{argmin}_{\theta} \mathrm{KL}(f||p) = \operatorname*{argmax}_{\theta} \mathrm{ALL}(\theta) = \operatorname*{argmax}_{\theta} \mathrm{LL}(\theta)$$

Mutual information

Mutual information between random variables is defined as

$$I(X;Y) = \mathrm{KL}(p(X,Y)||p(X)p(Y)) = \sum_x \sum_y p(x,y) \log \frac{p(x,y)}{p(x)p(y)}$$

Mutual information can be expressed as a difference of entropy and conditional entropy.

$$I(X;Y) = H(X) - H(X|Y)$$

Note that mutual information is symmetric

$$I(X;Y) = I(Y;X) = H(Y) - H(Y|X)$$

We will use mutual information your homework to learn Bayesian networks.

Information theory

Relevant concepts:

- Entropy
- Cross-entropy
- Kullback-Leibler divergence
- Mutual information

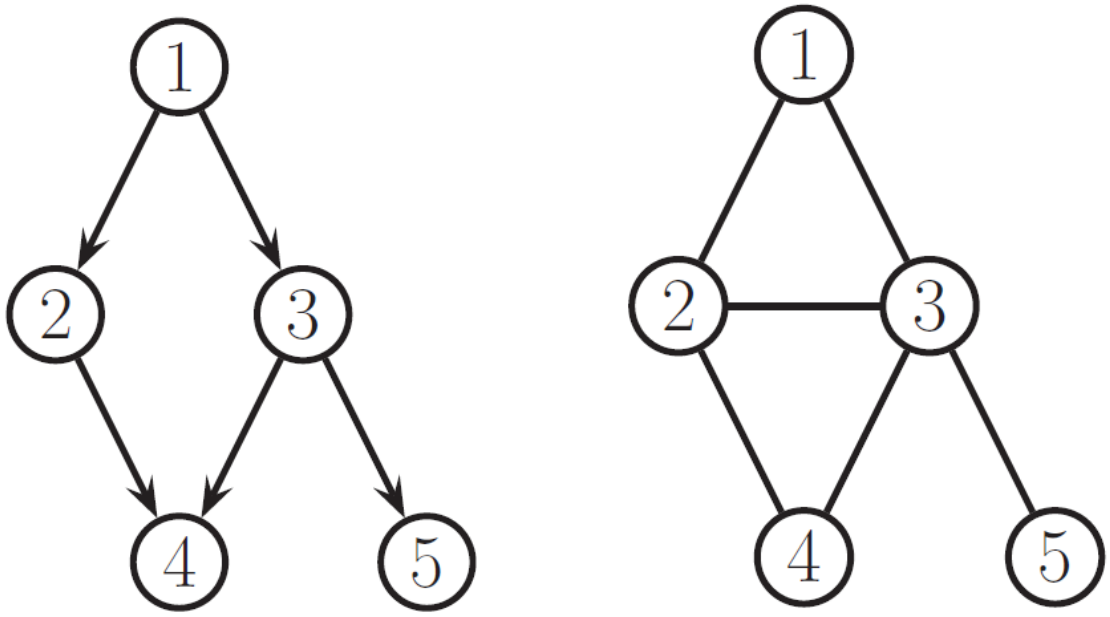
Graphical models

Suppose we observe multiple correlated variables, such as words in a document or pixels in an image.

- How can we compactly represent the joint distribution $p(\mathbf{x}|\theta)$?
- How can we use this distribution to infer one set of variables given another in a reasonable amount of computation time?
- How can we learn the parameters of this distribution with a reasonable amount of data?

Graphical models provide an intuitive way of representing and visualising the relationships between many variables

Representing knowledge through graphical models



- Nodes correspond to random variables
- Edges represent statistical dependencies between the variables

A graphical model is a way to represent a joint distribution by making **conditional independence** assumptions.

Graphical models = statistics $\times$ graph theory $\times$ computer science

Conditional independence

Marginal independence you are familiar with

$$\begin{aligned} p(X|Y,Z) &= p(X) \\ p(X,Y) &= p(X)p(Y) \end{aligned}$$

Conditional independence

$$\begin{aligned} p(X|Y,Z) &= p(X|Z) \\ p(X,Y|Z) &= p(X|Z)p(Y|Z) \end{aligned}$$

And shorthand for this relationship is

$$X \perp Y|Z$$

Note that the relationship is symmetric.

Examples of conditional independence

Shoe size $\perp$ Gray hair | Age

Temperature inside $\perp$ Temperature outside | Air conditioning is working

Pepsi or Coke $\perp$ Sweetness | Restaurant chain

Federal funds rate $\perp$ State of economy | Federal Reserve meeting notes

Dice roll outcome $\perp$ Previous dice rolls | Dice is not loaded

Benefits of capturing conditional independence

The obvious benefit is in compact representation.

Suppose we have a probability distribution $p(X, Y, Z)$ and random variables X, Y, Z can each assume 5 states.

To represent the distribution we need to store 5^3 probabilities.[3]

If we know that

$$X \perp Z|Y$$

then we can write

$$p(X, Y, Z) = p(X|Y, Z)p(Y|Z)p(Z) = p(X|Y)p(Y|Z)p(Z)$$

and instead of storing one large table we store 3 significantly smaller tables.[4]

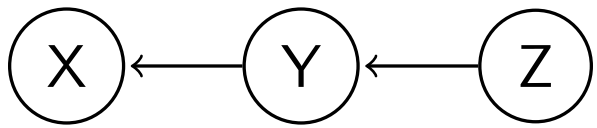
---

[3] ok $5^3 - 1$ because they need to sum to 1

[4] 124 vs 44 entries, alternatively 125 vs. 55

Benefits of capturing conditional independence

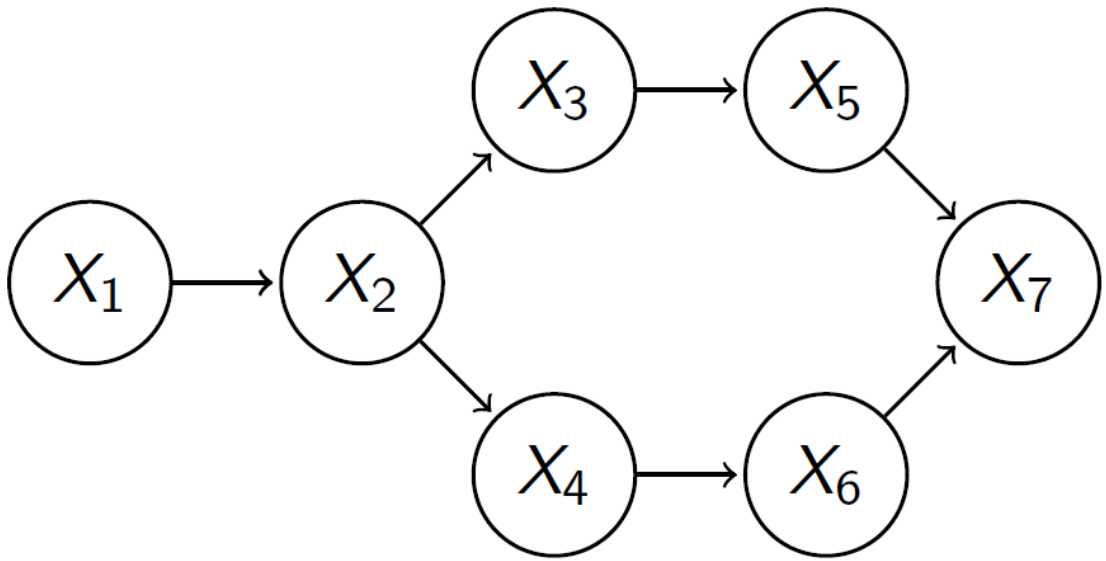
We can also capture such independencies in a graphical form.



$$p(X, Y, Z) = p(X|Y)p(Y|Z)p(Z)$$

Bayesian networks

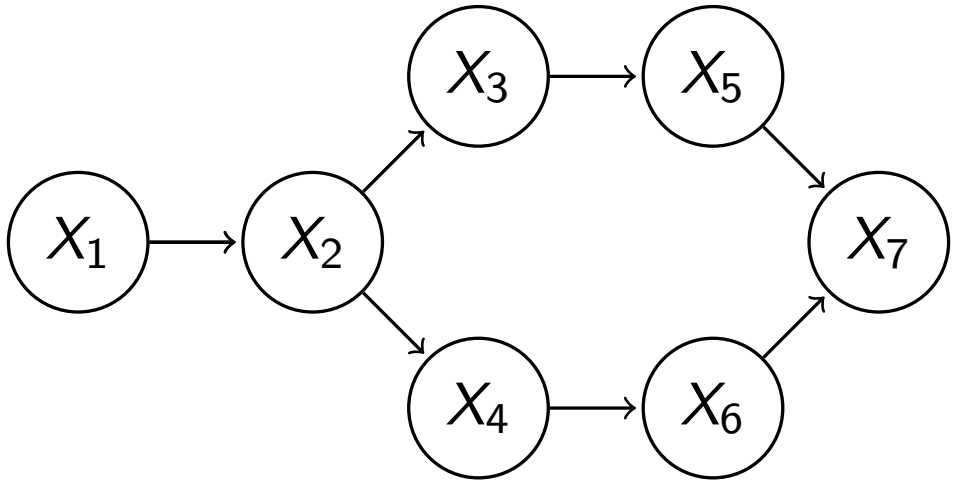
A directed graphical model is a graphical model whose graph is a directed acyclic graph (DAG). Also known as Bayesian network or belief network or causal network.



In DAGs the nodes can be ordered such that parents come before children. This is called a **topological ordering**.

Specifying a graphical model

Each of these nodes corresponds to a random variable.



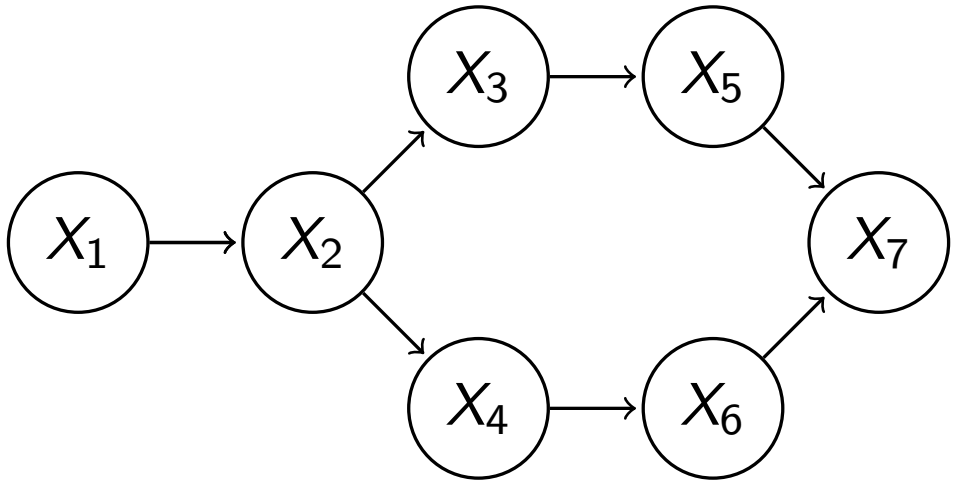
and each node has a conditional probability associated with it

$$p(X_i|X_{\mathbf{pa}(i)})$$

where $\mathbf{pa}(i)$ is a list of parent nodes of node i, e.g.

$$p(X_7|X_5, X_6)$$

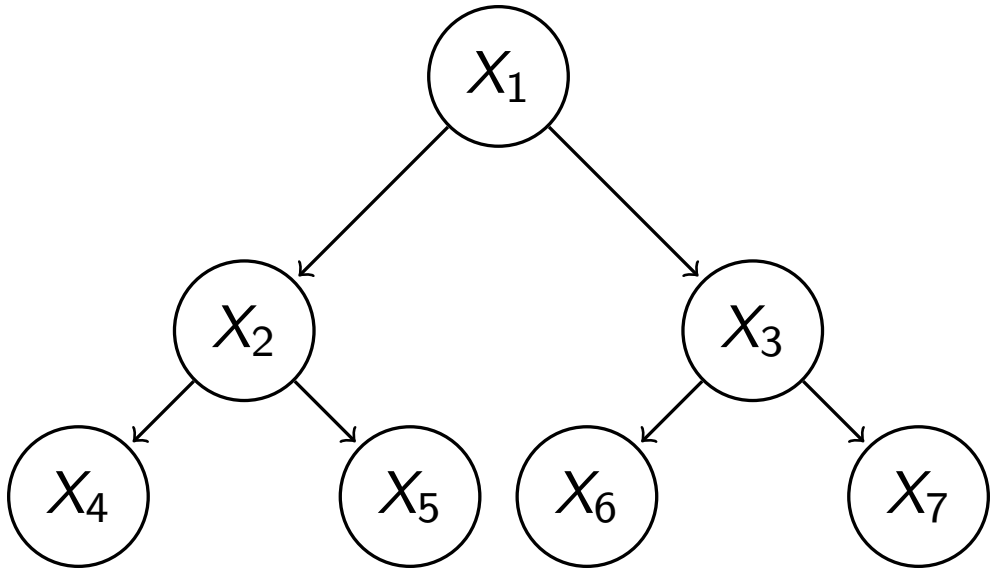
Specifying a graphical model



This graphical model specifies a joint distribution

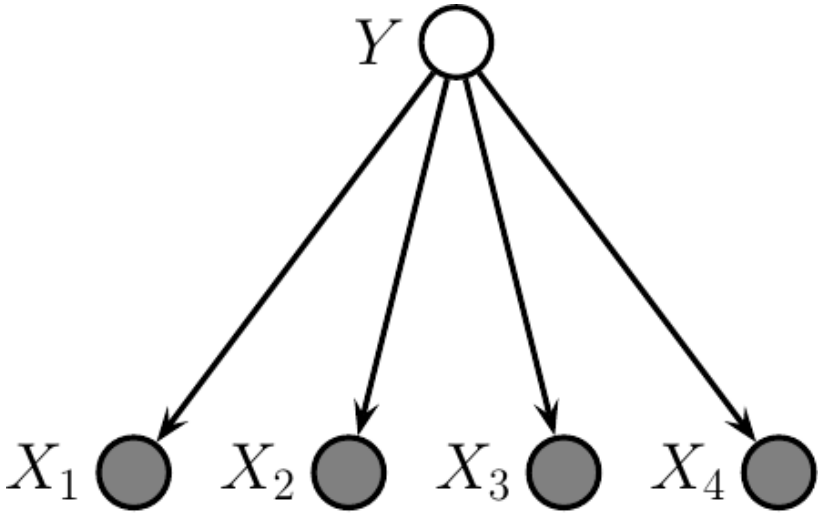
$$
\begin{aligned}
p(X_1, X_2, X_3, X_4, X_5, X_6, X_7) &= \prod_i p(X_i | X_{\mathbf{pa}(i)}) \\
&= p(X_1)\ p(X_2|X_1)\ p(X_3|X_2)\ p(X_4|X_2) \\
&\quad\ p(X_5|X_3)\ p(X_6|X_4)\ p(X_7|X_5, X_6)
\end{aligned}
$$

Another example of a graphical model

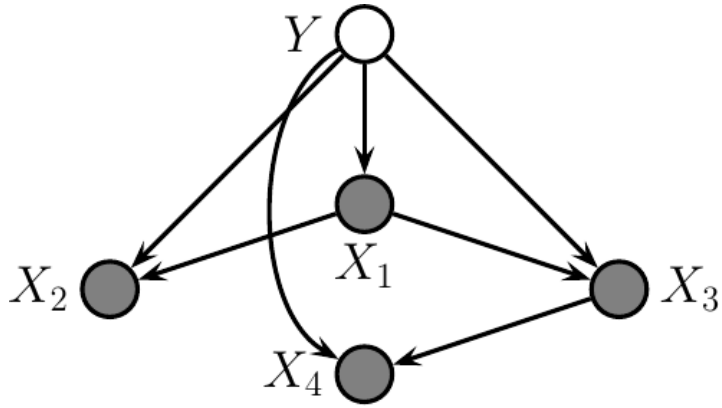


$$
\begin{aligned}
p(X_1, X_2, X_3, X_4, X_5, X_6, X_7) &= \prod_i p(X_i | X_{\mathbf{pa}(i)}) \\
&= p(X_1)p(X_2|X_1)p(X_3|X_1)p(X_4|X_2) \\
&\quad p(X_5|X_2)p(X_6|X_3)p(X_7|X_3)
\end{aligned}
$$

Example: naive Bayes classifiers as Bayesian networks



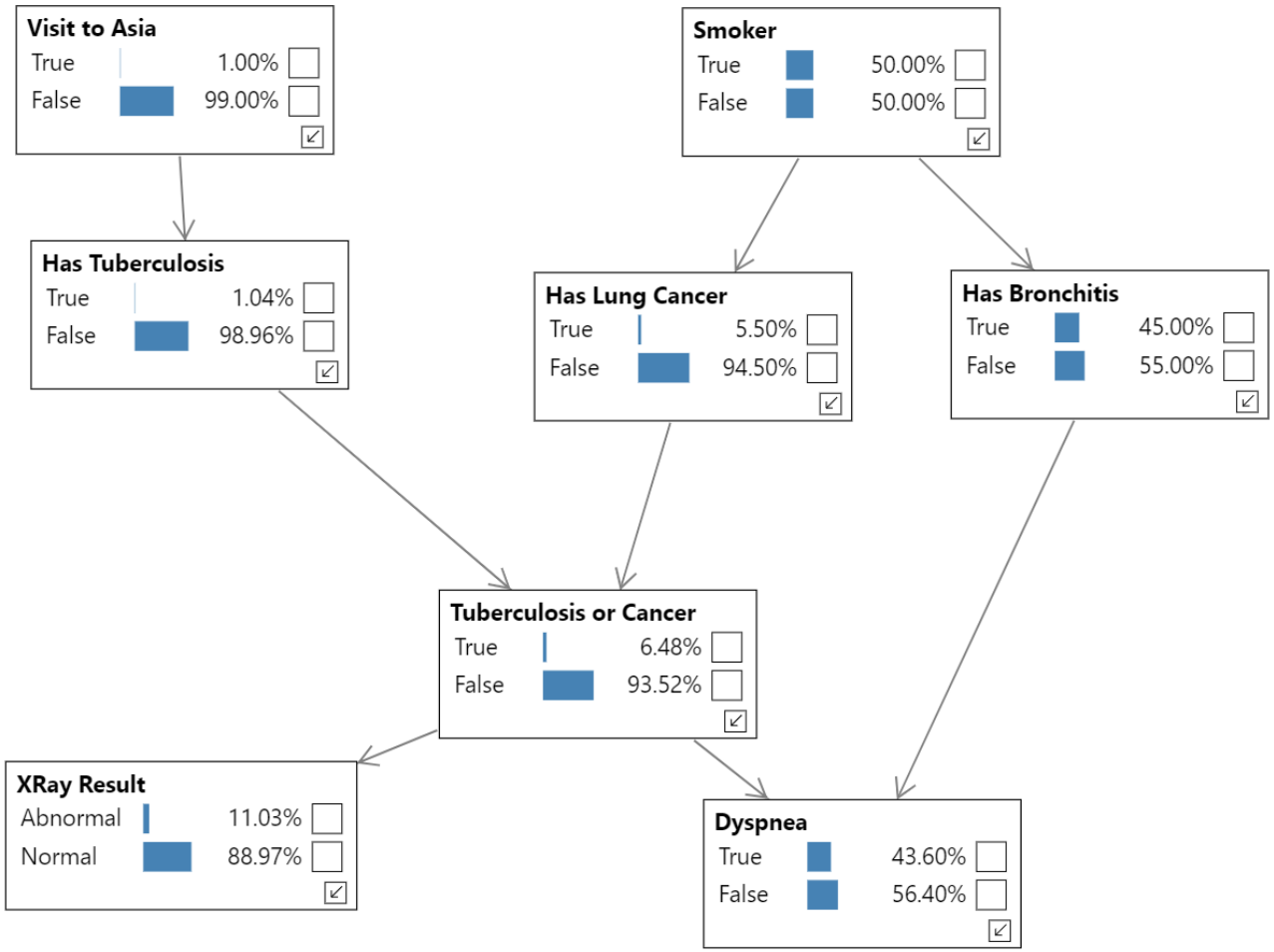
A naive Bayes classifier represented as a directed graphical model. We assume there are D = 4 features, for simplicity. Shaded nodes are observed, unshaded nodes are hidden.



Tree-augmented naive Bayes (TAN) classifier for D = 4 features. In general, the tree topology can change depending on the value of y.

$$p(y, \mathbf{x}) = p(y) \prod_{j=1}^{D} p(x_j|y)$$

Example: Bayesian network for medical diagnosis



Asia Bayesian Network (Interactive Demo)

Example: Bayesian network on Instagram hashtags

We downloaded images and captions from Instagram that were tagged #buterfly.

These captions were converted into binary vector of hashtag presence/absences.

Bayesian network was constructed on these data.

For HW2, you will do that and much more on a different Instagram dataset.

Today

- Information Theory
- Bayesian Networks