Bayesian networks
贝叶斯网络

Frequentist vs. Bayesian

客观 vs. 主观

Frequentist（频率主义者）: probability is the long-run expected frequency of occurrence. P(A) = n/N, where n is the number of times event A occurs in N opportunities.

“某事发生的概率是0.1”意味着*0.1*是在无穷多样本的极限条件下能够被观察到的比例

- 在许多情景下不可能进行重复试验

  发生第三次世界大战的概率是多少?

Bayesian: degree of belief. It is a measure of the plausibility（似然性） of an event given incomplete knowledge.

Probability

Probability is a rigorous formalism for uncertain knowledge
概率是对不确定知识一种严密的形式化方法

Joint probability distribution specifies probability of every atomic event
全联合概率分布指定了对随机变量的每种完全赋值，即每个原子事件的概率

Queries can be answered by summing over atomic events
可以通过把对应于查询命题的原子事件的条目相加的方式来回答查询

For nontrivial domains, we must find a way to reduce the joint size

Independence and conditional independence provide the tools

Independence/Conditional Independence

A and *B* are independent iff

$\mathbf{P}(A|B) = \mathbf{P}(A)$ or $\mathbf{P}(B|A) = \mathbf{P}(B)$ or $\mathbf{P}(A, B) = \mathbf{P}(A)\,\mathbf{P}(B)$

A is conditionally independent of *B* given *C*:

$\mathbf{P}(A \mid B, C) = \mathbf{P}(A \mid C)$

在大多数情况下，使用条件独立性能将全联合概率的表示由n的指数关系减为n的线性关系。

Conditional independence is our most basic and robust form of knowledge about uncertain environments.

Probability Theory

Probability theory can be expressed in terms of two simple equations

– Sum Rule （加法规则）

- probability of a variable is obtained by marginalizing （边缘化）or summing out other variables

$$p(a) = \sum_b p(a,b)$$

– Product Rule （乘法规则）

- joint probability expressed in terms of conditionals

$$p(a,b) = p(b \mid a)\, p(a)$$

All probabilistic inference and learning amounts to repeated application of sum and product rule

Outline

- Graphical models （概率图模型）
- Bayesian networks
  - Syntax（语法）
  - Semantics（语义）
- Inference（推导） in Bayesian networks

What are Graphical Models?

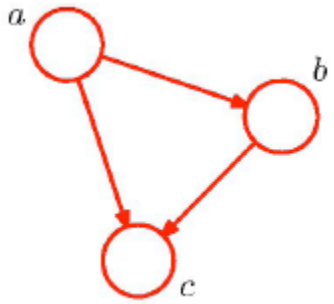
They are *diagrammatic*（图表的）*representations of* probability distributions

– marriage between probability theory and graph theory

- Also called *probabilistic graphical models*
- They augment analysis instead of using pure algebra（代数）

What is a Graph?

- Consists of nodes (also called vertices) and links (also called edges or arcs)



- In a probabilistic graphical model
  - each node represents a random variable (or group of random variables)
  - Links express probabilistic relationships between variables

Graphical Models in CS

- Natural tool for handling uncertainty（不确定性） and complexity（复杂性）
  - which occur throughout applied mathematics and engineering
- Fundamental to the idea of a graphical model is the notion of modularity（模块性）
  - a complex system is built by combining simpler parts.

Why are Graphical Models useful

- Probability theory provides the glue whereby
  - the parts are combined, ensuring that the system as a whole is consistent
  - providing ways to interface models to data.
- Graph theoretic side provides:
  - Intuitively appealing interface
    - by which humans can model highly-interacting sets of variables
  - Data structure
    - that lends itself naturally to designing efficient general-purpose（通用的） algorithms

Graphical models: Unifying Framework

- View classical multivariate（多变量的） probabilistic systems as instances of a common underlying formalism（形式）
  - mixture models（混合模型）, factor analysis（因子分析）, hidden Markov models, Kalman filters（卡尔曼滤波器）, etc.
  - Encountered in systems engineering, information theory, pattern recognition and statistical mechanics

- Advantages of View:
  - Specialized techniques in one field can be transferred between communities and exploited
  - Provides natural framework for designing new systems

Role of Graphical Models in Machine Learning

1. Simple way to visualize（形象化）

   structure of probabilistic model

2. Insights into properties of model

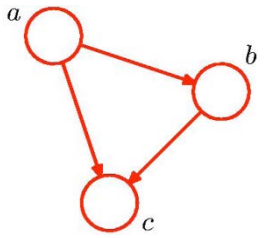
   Conditional independence properties by inspecting graph

3. Complex computations

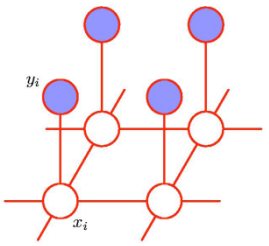
   required to perform inference and learning expressed as graphical manipulations

Graph Directionality

- Directed graphical models
  - Directionality associated with arrows
- Bayesian networks
  - Express causal relationships （因果关系） between random variables
- More popular in AI and statistics



- Undirected graphical models
  - links without arrows
- Markov random fields （马尔科夫随机场）
  - Better suited to express soft constraints between variables
- More popular in Vision and physics



Bayesian networks

一种简单的，图形化的数据结构，用于表示变量之间的依赖关系（条件独立性），为任何全联合概率分布提供一种简明的规范。

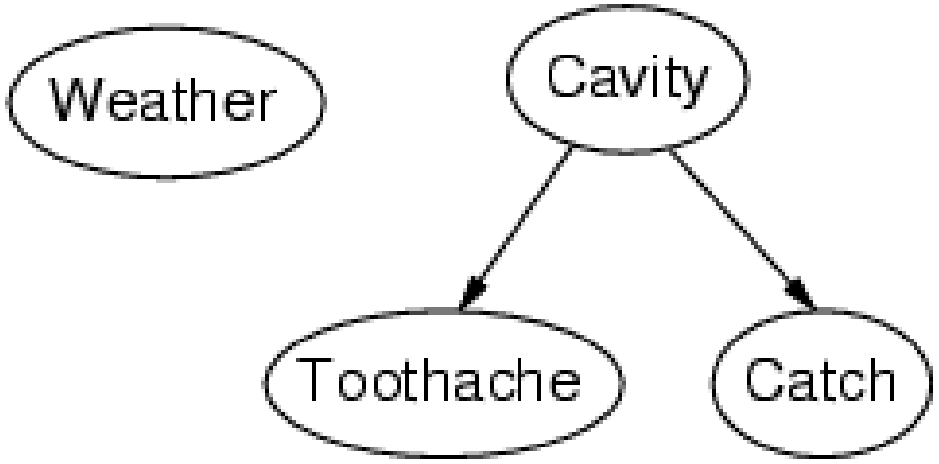
Syntax:

- a set of nodes, one per variable
- a directed（有向）, acyclic（无环） graph (link ≈ "direct influences")
- a conditional distribution for each node given its parents:
  - **P** (X_i | Parents (X_i))—量化其父节点对该节点的影响

In the simplest case, conditional distribution represented as a conditional probability table 条件概率表 (CPT) giving the distribution over X_i for each combination of parent values

Example

Topology（拓扑结构） of network encodes conditional independence assertions:



Weather is independent of the other variables

Toothache and *Catch* are conditionally independent given *Cavity*

Example

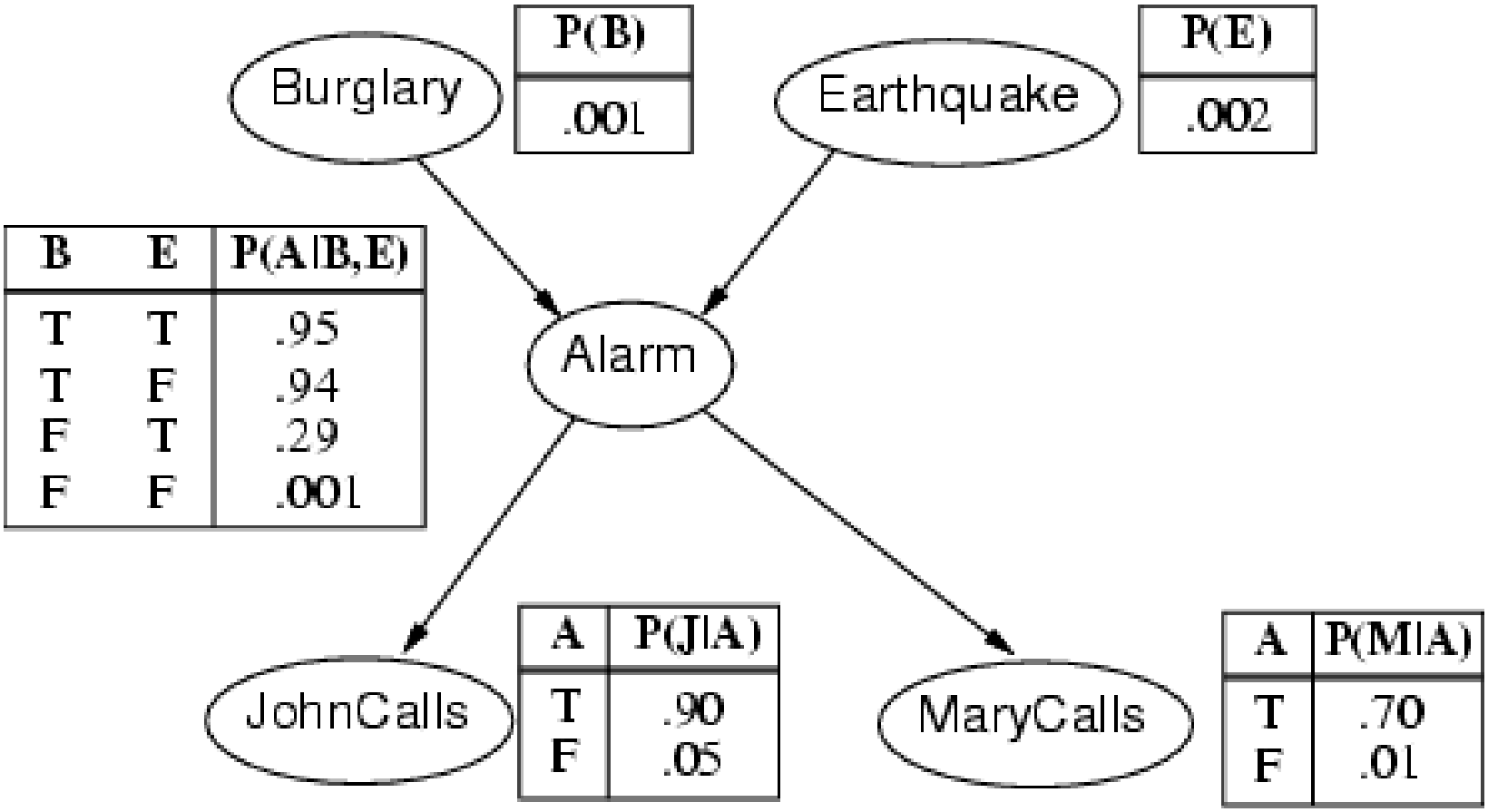
I‘m at work, neighbor John calls to say my alarm is ringing, but neighbor Mary doesn’t call. Sometimes it‘s set off by minor earthquakes. Is there a burglar（夜贼）?

Variables: *Burglary（入室行窃）*, *Earthquake*, *Alarm*, *JohnCalls*, *MaryCalls*

Network topology reflects “causal（因果）" knowledge:

- A burglar can set the alarm off
- An earthquake can set the alarm off
- The alarm can cause Mary to call
- The alarm can cause John to call

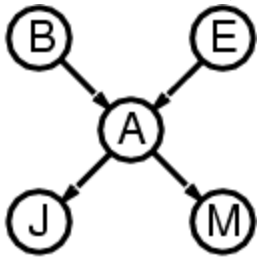
Example contd.



Compactness（紧致性）

A CPT for Boolean X_i with k Boolean parents has 2^k rows for the combinations of parent values

一个具有k个布尔父节点的布尔变量的条件概率表中有2^k个独立的可指定概率



Each row requires one number p for $X_i = true$ (the number for $X_i = false$ is just $1-p$)

If each variable has no more than k parents, the complete network requires $O(n \cdot 2^k)$ numbers

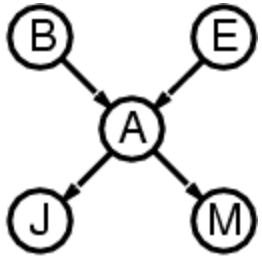
I.e., grows linearly with n, vs. $O(2^n)$ for the full joint distribution

For burglary net, 1 + 1 + 4 + 2 + 2 = 10 numbers (vs. $2^5-1 = 31$)

Global semantics（全局语义）

The full joint distribution is defined as the product of the local conditional distributions:

全联合概率分布可以表示为贝叶斯网络中的条件概率分布的乘积



"Global" semantics defines the full joint distribution as the product of the local conditional distributions:

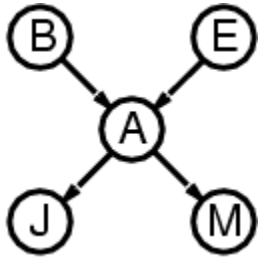
$$P(x_1, \ldots, x_n) = \prod_{i=1}^{n} P(x_i | parents(X_i))$$

e.g., $P(j \wedge m \wedge a \wedge \neg b \wedge \neg e)$

Global semantics（全局语义）

The full joint distribution is defined as the product of the local conditional distributions:

全联合概率分布可以表示为贝叶斯网络中的条件概率分布的乘积



"Global" semantics defines the full joint distribution as the product of the local conditional distributions:

$$P(x_1, \ldots, x_n) = \prod_{i=1}^{n} P(x_i | parents(X_i))$$

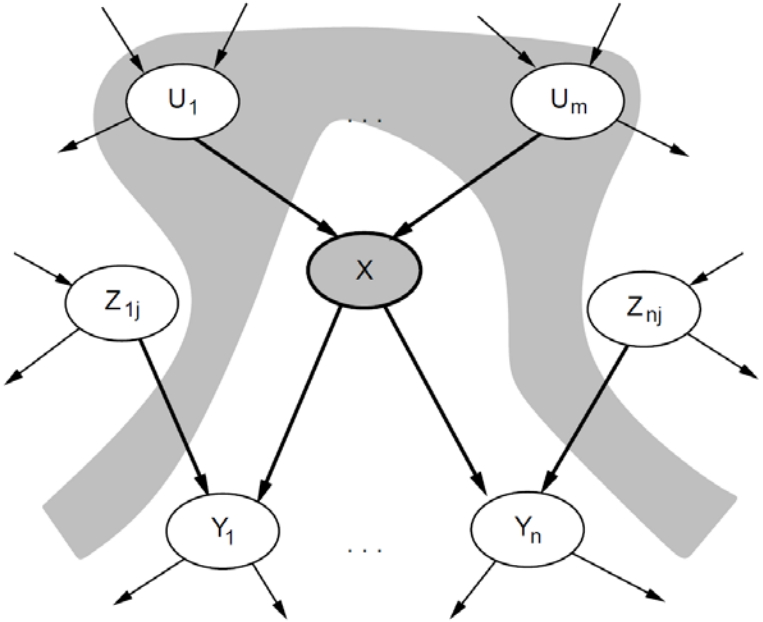
e.g., $P(j \wedge m \wedge a \wedge \neg b \wedge \neg e)$

$$\begin{aligned} &= P(j|a)P(m|a)P(a|\neg b, \neg e)P(\neg b)P(\neg e) \\ &= 0.9 \times 0.7 \times 0.001 \times 0.999 \times 0.998 \\ &\approx 0.00063 \end{aligned}$$

Local semantics

Local semantics: each node is conditionally independent of its nondescendants（非后代） given its parents

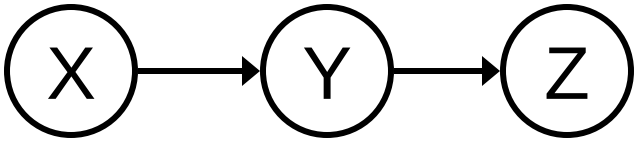
给定父节点，一个节点与它的非后代节点是条件独立的



Theorem: Local semantics ⇔ global semantics

Causal Chains

- A basic configuration



X: Low pressure

Y: Rain

Z: Traffic

$$P(x, y, z) = P(x)P(y|x)P(z|y)$$

  - Is X independent of Z given Y?

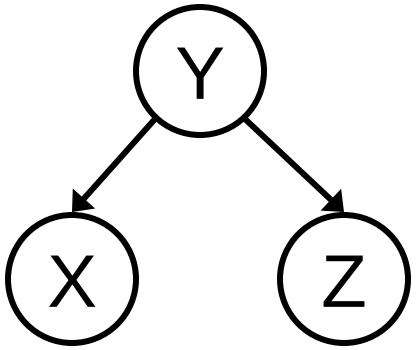
$$P(z|x, y) = \frac{P(x, y, z)}{P(x, y)} = \frac{P(x)P(y|x)P(z|y)}{P(x)P(y|x)}$$

$$= P(z|y)$$ *Yes!*

  - Evidence along the chain "blocks" the influence

Common Cause

- Another basic configuration: two effects of the same cause
  - Are X and Z independent?
  - Are X and Z independent given Y?



$$P(z|x,y) = \frac{P(x,y,z)}{P(x,y)} = \frac{P(y)P(x|y)P(z|y)}{P(y)P(x|y)}$$

$$= P(z|y) \quad \textit{Yes!}$$

Y: Project due

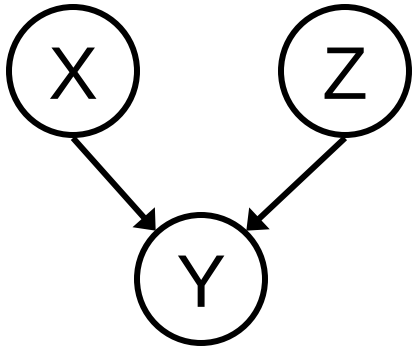
X: Newsgroup busy

Z: Lab full

  - Observing the cause blocks influence between effects.

Common Effect

- Last configuration: two causes of one effect (v-structures)
  - Are X and Z independent?
    - Yes: remember the ballgame and the rain causing traffic, no correlation?
  - Are X and Z independent given Y?
    - No: remember that seeing traffic put the rain and the ballgame in competition?
  - This is backwards from the other cases
    - Observing the effect enables influence between causes.



X: Raining

Z: Ballgame

Y: Traffic

Constructing Bayesian networks

Need a method such that a series of locally testable assertions of conditional independence guarantees the required global semantics

需要一种方法使得局部的条件独立关系能够保证全局语义得以成立

1. Choose an ordering of variables $X_1, \ldots, X_n$
2. For i = 1 to n

   add X_i to the network

   select parents from $X_1, \ldots, X_{i-1}$ such that

   $\mathbf{P}(X_i \mid \text{Parents}(X_i)) = \mathbf{P}(X_i \mid X_1, \ldots X_{i-1})$

This choice of parents guarantees the global semantics:

$$\mathbf{P}(X_1, \ldots, X_n) = \prod_{i=1}^{n} \mathbf{P}(X_i | X_1, \ldots, X_{i-1}) \quad \text{(chain rule)}$$

$$= \prod_{i=1}^{n} \mathbf{P}(X_i | Parents(X_i)) \quad \text{(by construction)}$$

Constructing Bayesian networks

要求网络的拓扑结构确实反映了合适的父节点集对每个变量的那些直接影响。

添加节点的正确次序是首先添加“根本原因”节点，然后加入受它们直接影响的变量，以此类推。

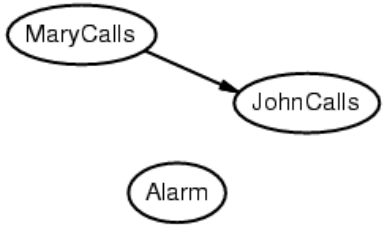
Example

Suppose we choose the ordering *M, J, A, B, E*

P(J | M) = **P**(J)?

Example

Suppose we choose the ordering *M, J, A, B, E*

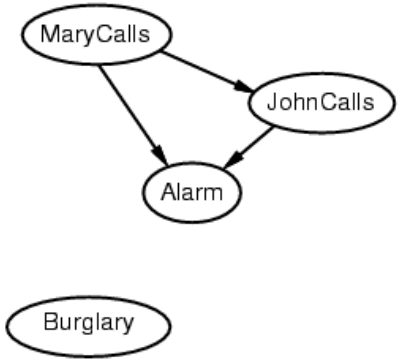


P(J | M) = **P**(J)? **No**

P(A | J, M) = **P**(A | J)? **P**(A | J, M) = **P**(A)?

Example

Suppose we choose the ordering *M, J, A, B, E*



P(J | M) = **P**(J)? **No**

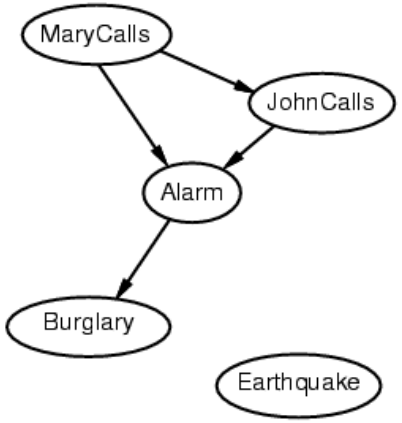
P(A | J, M) = **P**(A | J)? **P**(A | J, M) = **P**(A)? **No**

P(B | A, J, M) = **P**(B | A)?

P(B | A, J, M) = **P**(B)?

Example

Suppose we choose the ordering *M, J, A, B, E*



P(J | M) = **P**(J)? **No**

P(A | J, M) = **P**(A | J)? **P**(A | J, M) = **P**(A)? **No**

P(B | A, J, M) = **P**(B | A)? **Yes**

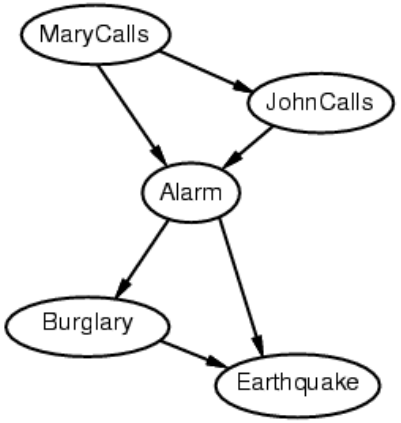
P(B | A, J, M) = **P**(B)? **No**

P(E | B, A ,J, M) = **P**(E | A)?

P(E | B, A, J, M) = **P**(E | A, B)?

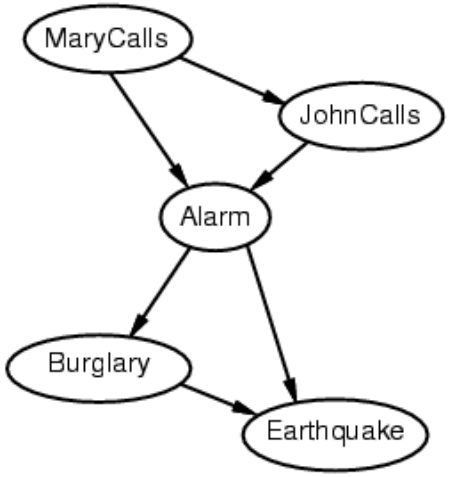
Example

Suppose we choose the ordering *M, J, A, B, E*



P(J | M) = **P**(J)?	**No**
P(A | J, M) = **P**(A | J)? **P**(A | J, M) = **P**(A)?	**No**
P(B | A, J, M) = **P**(B | A)?	**Yes**
P(B | A, J, M) = **P**(B)?	**No**
P(E | B, A ,J, M) = **P**(E | A)?	**No**
P(E | B, A, J, M) = **P**(E | A, B)?	**Yes**

Example contd.



Deciding conditional independence is hard in noncausal（非因果） directions

(Causal models and conditional independence seem hardwired for humans!)

Network is less compact: 1 + 2 + 4 + 2 + 4 = 13 numbers needed

Causality?

- When Bayes' nets reflect the true causal patterns:
  - Often simpler (nodes have fewer parents)
  - Often easier to think about
  - Often easier to elicit from experts

- BNs need not actually be causal
  - Sometimes no causal net exists over the domain (especially if variables are missing)
  - End up with arrows that reflect correlation, not causation

- What do the arrows really mean?
  - Topology may happen to encode causal structure
  - Topology really encodes conditional independence

Inference in Bayesian networks

Inference tasks

Simple queries: compute posterior probability $\mathbf{P}(X_i|E=e)$

e.g., *P(NoGas|Gauge油表=empty, Lights=on, Starts=false)*

Conjunctive queries（联合查询）:

$\mathbf{P}(X_i,X_j|E=e) = \mathbf{P}(X_i|E=e)\mathbf{P}(X_j|X_i,E=e)$

Optimal decisions: decision networks include utility information;

probabilistic inference required for

P(outcome|action, evidence)

Inference by enumeration

Slightly intelligent way to sum out variables from the joint without actually constructing its explicit representation

$$\mathbf{P}(X|\mathbf{e}) = \alpha\mathbf{P}(X,\mathbf{e}) = \alpha\sum_{\mathbf{y}}\mathbf{P}(X,\mathbf{e},\mathbf{y})$$

在贝叶斯网络中可以将全联合分布写成条件概率乘积的形式：

$$\mathbf{P}(X_1,...,X_n) = \prod_{i=1}^{n}\mathbf{P}(X_i|Parents(X_i))$$

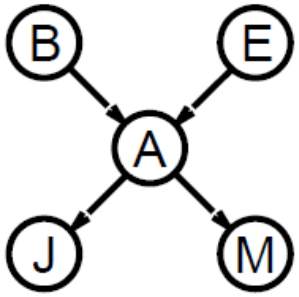
在贝叶斯网络中可以通过计算条件概率的乘积并求和来回答查询。

Inference by enumeration

Slightly intelligent way to sum out variables from the joint without actually constructing its explicit representation

Simple query on the burglary network:

$$\begin{aligned}&\mathbf{P}(B|j,m)\\&= \mathbf{P}(B,j,m)/P(j,m)\\&= \alpha\mathbf{P}(B,j,m)\\&= \alpha \sum_e \sum_a \mathbf{P}(B,e,a,j,m)\end{aligned}$$

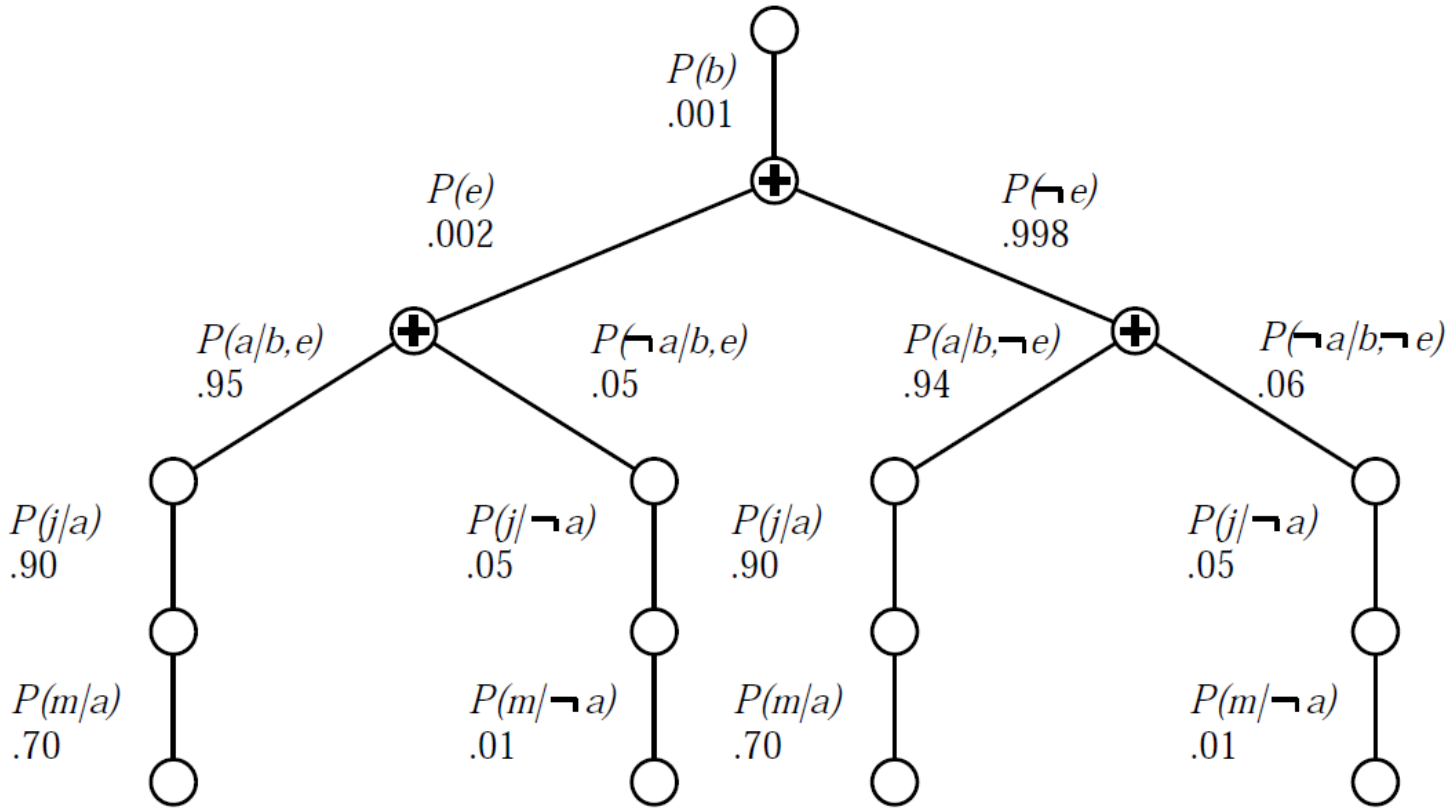


Rewrite full joint entries using product of CPT entries:

$$\begin{aligned}&\mathbf{P}(B|j,m)\\&= \alpha \sum_e \sum_a \mathbf{P}(B)P(e)\mathbf{P}(a|B,e)P(j|a)P(m|a)\\&= \alpha\mathbf{P}(B) \sum_e P(e) \sum_a \mathbf{P}(a|B,e)P(j|a)P(m|a)\end{aligned}$$

Recursive depth-first enumeration: $O(n)$ space, $O(d^n)$ time

Evaluation tree



Enumeration is inefficient: repeated computation

e.g., computes $P(j|a)P(m|a)$ for each value of e

Inference by variable elimination

Variable elimination（变量消元）: carry out summations right-to-left, storing intermediate results (factors：因子) to avoid recomputation

$$
\begin{aligned}
&\mathbf{P}(B|j,m) \\
&\quad = \alpha \underbrace{\mathbf{P}(B)}_{B} \Sigma_e \underbrace{P(e)}_{E} \Sigma_a \underbrace{\mathbf{P}(a|B,e)}_{A} \underbrace{P(j|a)}_{J} \underbrace{P(m|a)}_{M} \\
&\quad = \alpha \mathbf{P}(B) \Sigma_e P(e) \Sigma_a \mathbf{P}(a|B,e) P(j|a) f_M(a) \\
&\quad = \alpha \mathbf{P}(B) \Sigma_e P(e) \Sigma_a \mathbf{P}(a|B,e) f_J(a) f_M(a) \\
&\quad = \alpha \mathbf{P}(B) \Sigma_e P(e) \Sigma_a f_A(a,b,e) f_J(a) f_M(a) \\
&\quad = \alpha \mathbf{P}(B) \Sigma_e P(e) f_{\bar{A}JM}(b,e) \text{ (sum out } A\text{)} \\
&\quad = \alpha \mathbf{P}(B) f_{\bar{E}\bar{A}JM}(b) \text{ (sum out } E\text{)} \\
&\quad = \alpha f_B(b) \times f_{\bar{E}\bar{A}JM}(b)
\end{aligned}
$$

Complexity of exact inference

Singly connected networks单联通网络 (or polytrees多树):

— any two nodes are connected by at most one (undirected) path

— time and space cost of variable elimination are $O(d^k n)$

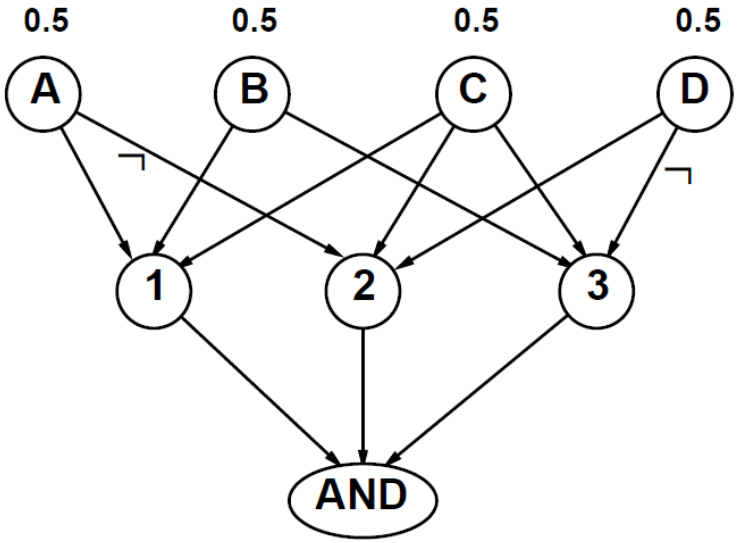
多树上的变量消元的时间和空间复杂度都与网络规模呈线性关系。

Multiply connected networks多联通网络:

— can reduce 3SAT to exact inference ⇒ NP-hard

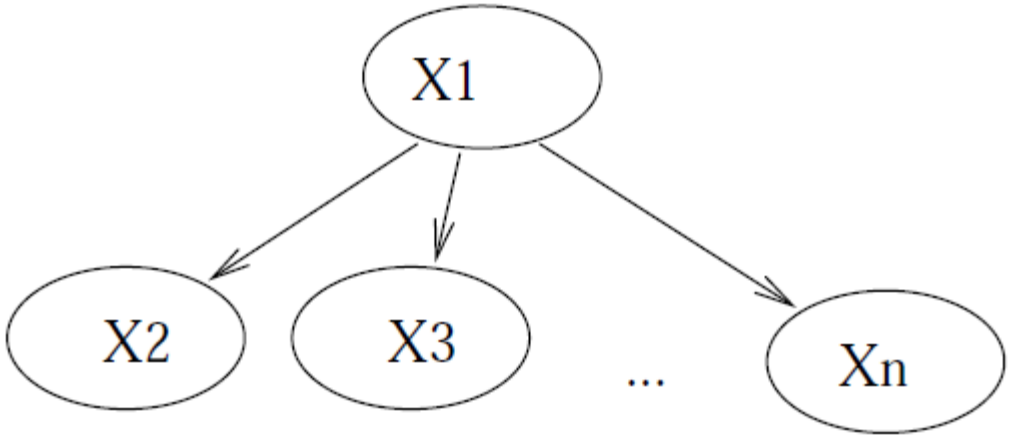
— equivalent to counting 3SAT models ⇒ #P-complete

1. A v B v C
2. C v D v ¬A
3. B v C v ¬D



Example: Naïve Bayes model

There is a single parent variable and a collection of child variables whose values are conditionally independent from one another given the parent.

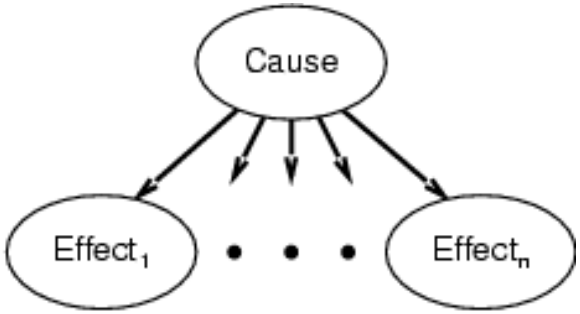


$$\begin{aligned} &\mathrm{P}(X_1 = x_1, \ldots, X_n = x_n) \\ &\quad = \mathrm{P}(X_1 = x_1)\, \mathrm{P}(X_2 = x_2 | X_1 = x_1) \cdots \mathrm{P}(X_n = x_n | X_1 = x_1) \end{aligned}$$

Naïve Bayes model

$$\mathbf{P}(Cause, Effect_1, ..., Effect_n) = \mathbf{P}(Cause) \prod_i \mathbf{P}(Effect_i \mid Cause)$$

$$\mathbf{P}(Cause|Effect_1, ..., Effect_n) = \mathbf{P}(\mathbf{Effects}, Cause)/\mathbf{P}(\mathbf{Effects})$$
$$= \alpha \mathbf{P}(Cause, \mathbf{Effects}) = \alpha \mathbf{P}(Cause) \prod_i \mathbf{P}(Effect_i \mid Cause)$$



Total number of parameters（参数） is linear in *n*

Example: Spam detection

Imagine the problem of trying to automatically detect spam e-mail messages（垃圾邮件）. A simple approach to get started is to look only at the "Subject:" headers in the e-mail messages and attempt to recognize spam by checking some simple computable features（特征）. The two simple features we will consider are:

Caps: Whether the subject header is entirely capitalized

Free: Whether the subject header contains the word `free', either in upper case or lower case

e.g.: a message with the subject header "NEW MORTGAGE RATE" is likely to be spam. Similarly, for "Money for Free", "FREE lunch", etc.

Example: Spam detection

The model is based on the following three random variables, *Caps*, *Free* and *Spam*, each of which take on the values Y (for Yes) or N (for No)

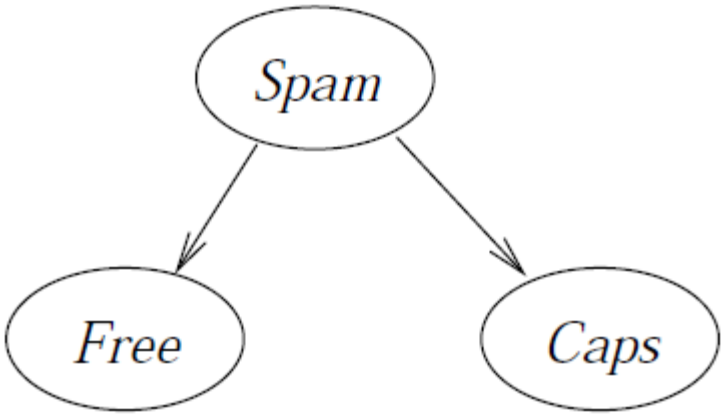
Caps = Y if and only if the subject of the message does not contain lowercase letters

Free = Y if and only if the word `free' appears in the subject (letter case is ignored)

Spam = Y if and only if the message is spam

P(Free , Caps, Spam)= P(Spam) P(Caps|Spam) P(Free|Spam)

Example: Spam detection



P(*Free , Caps, Spam*)= **P**(*Spam*) **P**(*Caps*|*Spam*) **P**(*Free*|*Spam*)

Example: Spam detection

Free	*Caps*	*Spam*	# messages
Y	Y	Y	20
Y	Y	N	1
Y	N	Y	5
Y	N	N	0
N	Y	Y	20
N	Y	N	3
N	N	Y	2
N	N	N	49
		Total:	100

Spam	P(*Spam*)
Y	$\frac{20+5+20+2}{100} = 0.47$
N	$\frac{1+0+3+49}{100} = 0.53$

Caps	*Spam*	P(*Caps*|*Spam*)
Y	Y	$\frac{20+20}{20+5+20+2} \approx 0.8511$
Y	N	$\frac{1+3}{1+0+3+49} \approx 0.0755$
N	Y	$\frac{5+2}{20+5+20+2} \approx 0.1489$
N	N	$\frac{0+49}{1+0+3+49} \approx 0.9245$

Free	*Spam*	P(*Free*|*Spam*)
Y	Y	$\frac{20+5}{20+5+20+2} \approx 0.5319$
Y	N	$\frac{1+0}{1+0+3+49} \approx 0.0189$
N	Y	$\frac{20+2}{20+5+20+2} \approx 0.4681$
N	N	$\frac{3+49}{1+0+3+49} \approx 0.9811$

Example: Spam detection

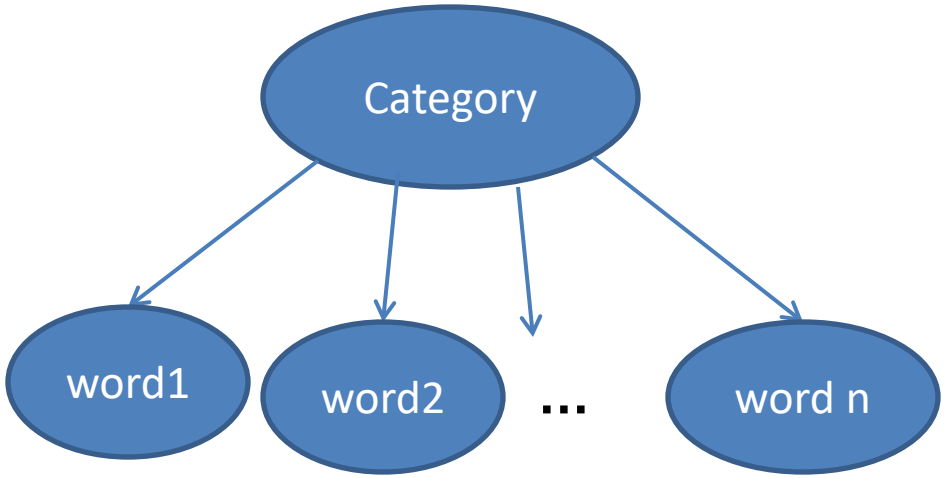
$$
\begin{aligned}
&\mathrm{P}(\mathit{Free} = Y, \mathit{Caps} = N, \mathit{Spam} = N) \\
&\quad = \quad \mathrm{P}(\mathit{Spam} = N)\ \ \mathrm{P}(\mathit{Caps} = N | \mathit{Spam} = N)\ \ \mathrm{P}(\mathit{Free} = Y | \mathit{Spam} = N) \\
&\quad \approx \quad 0.53 \times 0.9245 \times 0.0189 \\
&\quad \approx \quad 0.0093
\end{aligned}
$$

Example: Learning to classify text documents (13.18)

文本分类是在文档所包含的文本基础上，把给定的文档分配到固定类别集合中某一个类别的任务。这个任务中常常用到朴素贝叶斯模型。在这些模型中，查询变量是文档类别，“结果”变量则是语言中每个词是否出现。我们假设文档中的词的出现都是独立的，其出现频率由文档类别确定。

a. 准确地解释当给定一组类别已经确定的文档作为“训练数据”时，这样的模型是如何构造的。
b. 准确地解释如何对新文档进行分类。
c. 这里独立性假设合理吗？请讨论。

Example: Learning to classify text documents



The model consists of the prior probability P(Category) and the conditional probabilities P(word i|Category)

- P(Category=c) is estimated as the fraction of all documents that are of category c
- P(word i = true|Category=c) is estimated as the fraction of documents of category c that contain word i

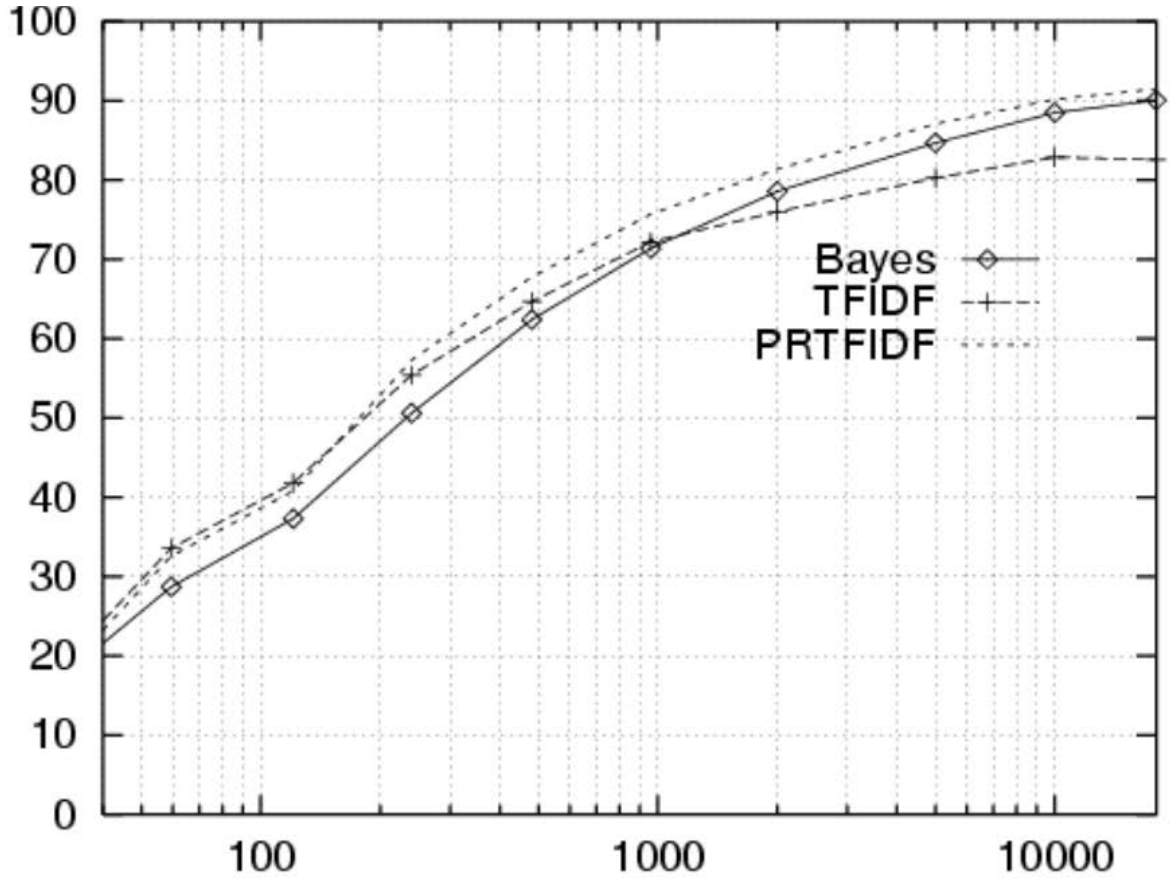
Twenty Newsgroups

Given 1000 training documents from each group. Learn to classify new documents according to which newsgroup it came from

comp.graphics comp.os.ms-windows.misc comp.sys.ibm.pc.hardware comp.sys.mac.hardware comp.windows.x	rec.autos rec.motorcycles rec.sport.baseball rec.sport.hockey	sci.crypt sci.electronics sci.med sci.space
misc.forsale	talk.politics.misc talk.politics.guns talk.politics.mideast	talk.religion.misc alt.atheism soc.religion.christian

Naïve Bayes: 89% classification accuracy

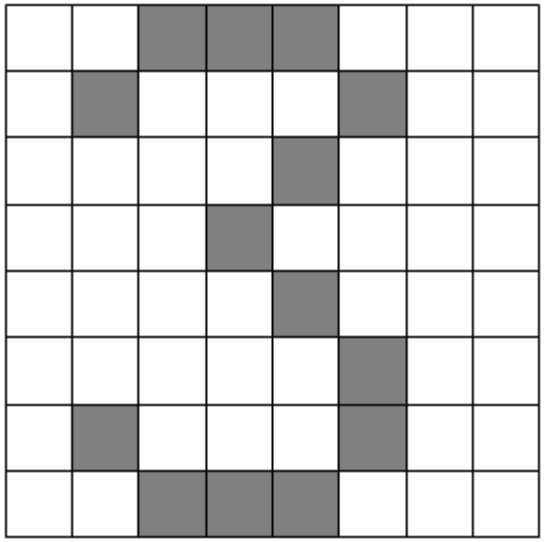
Learning Curve for 20 Newsgroups



Accuracy vs. Training set size (1/3 withheld for test)

Example: A Digit Recognizer

- Input: pixel grids



- Output: a digit 0-9

Naïve Bayes for Digits

Simple version:

- One feature F_{ij} for each grid position <i,j>
- Possible feature values are on / off, based on whether intensity is more or less than 0.5 in underlying image
- Each input maps to a feature vector, e.g.

$$\rightarrow \langle F_{0,0} = 0 \;\; F_{0,1} = 0 \;\; F_{0,2} = 1 \;\; F_{0,3} = 1 \;\; F_{0,4} = 0 \;\; \ldots F_{15,15} = 0 \rangle$$

- Here: lots of features, each is binary

Naïve Bayes model:

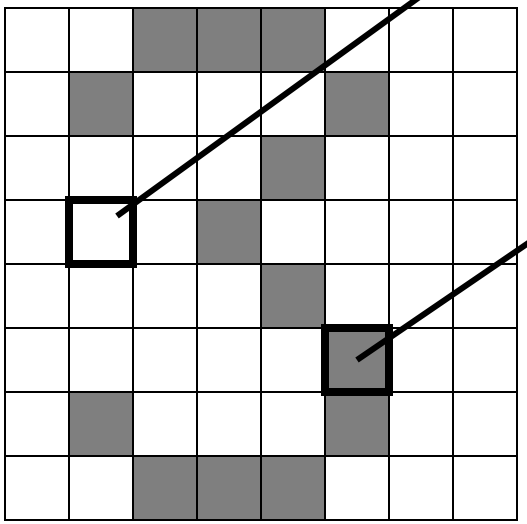
$$P(Y|F_{0,0} \ldots F_{15,15}) \propto P(Y) \prod_{i,j} P(F_{i,j}|Y)$$

What do we need to learn?

Examples: CPTs

$P(Y)$

1	0.1
2	0.1
3	0.1
4	0.1
5	0.1
6	0.1
7	0.1
8	0.1
9	0.1
0	0.1



$P(F_{3,1} = on|Y)$

1	0.01
2	0.05
3	0.05
4	0.30
5	0.80
6	0.90
7	0.05
8	0.60
9	0.50
0	0.80

$P(F_{5,5} = on|Y)$

1	0.05
2	0.01
3	0.90
4	0.80
5	0.90
6	0.90
7	0.25
8	0.85
9	0.60
0	0.80

Comments on Naïve Bayes

Makes probabilistic inference tractable by making a strong assumption of conditional independence.

Tends to work fairly well despite this strong assumption.

Experiments show it to be quite competitive with other classification methods on standard datasets.

Particularly popular for text categorization, e.g. spam filtering.

Summary

- Bayesian networks provide a natural representation for (causally induced) conditional independence
- Topology + CPTs = compact representation of joint distribution
- Generally easy for domain experts to construct

- Exact inference by variable elimination:
  - polytime on polytrees, NP-hard on general graphs
  - space = time, very sensitive to topology
- Naïve Bayes model

作业

- 14.3(a,b,c)，14.4， 14.7(a,b,c)