Bayesian Decision Theory

Berlin Chen

Graduate Institute of Computer Science & Information Engineering

National Taiwan Normal University

References:

1. **E. Alpaydin, *Introduction to Machine Learning* , Chapter 3**
2. Tom M. Mitchell, *Machine Learning*, Chapter 6

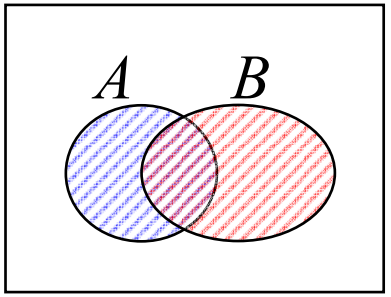
Review: Basic Formulas for Probabilities (1/2)

- **Product Rule**: probability $P(A \cap B)$ of a conjunction of two events *A* and *B*

$$P(A \cap B) = P(A, B) = P(A|B)P(B) = P(B|A)P(A)$$

- **Sum Rule**: probability of a disjunction of two events *A* and *B*

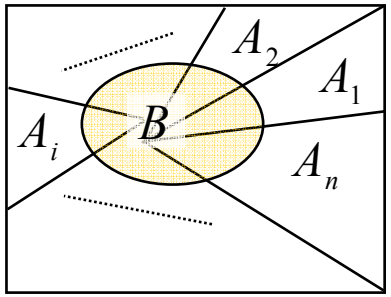
$$P(A \cup B) = P(A) + P(B) - P(A \cap B)$$



- **Theorem of total probability**: if events $A_1, \ldots, A_n$ are mutually exclusive and exhaustive ($\forall i \neq j \;\; P(A_i \cap A_j) = 0$ and $\sum_{i=1}^{n} P(A_i) = 1$)

$$P(B) = \sum_{i=1}^{n} P(B \cap A_i)$$

$$= \sum_{i=1}^{n} P(B|A_i)P(A_i)$$



Review: Basic Formulas for Probabilities (2/2)

- **Chain Rule**: probability of a conjunction of many events $A_1, A_2, \ldots, A_n$

$$P(A_1, A_2, \ldots, A_n)$$
$$= P(A_1)P(A_2|A_1)P(A_3|A_1, A_2)\ldots P(A_n|A_1, A_2, \ldots, A_{n-1})$$

Classification (1/5)

- Illustrative Case 1: Credit Scoring

$$\boldsymbol{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}^t = \begin{bmatrix} income \\ savings \end{bmatrix}^t$$

$C_1 \Rightarrow$ hight - risk

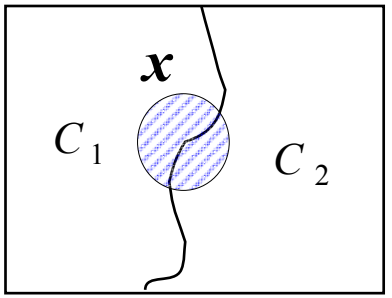
$C_2 = 0 \Rightarrow$ Low - risk

– Given a new application $\boldsymbol{x} = [x_1, x_2]$

$$\text{Choose} \begin{cases} C_1 & \text{if } P(C_1|\boldsymbol{x}) > 0.5 \\ C_2 & \text{otherwise} \end{cases}$$

or equivalent ly

$$\text{Choose} \begin{cases} C_1 & \text{if } P(C_1|\boldsymbol{x}) > P(C_2|\boldsymbol{x}) \\ C_2 & \text{otherwise} \end{cases}$$



Note that $P(C_1|\boldsymbol{x}) + P(C_2|\boldsymbol{x}) = 1$

Classification (2/5)

- Bayes' Classifier
  - We can use the probability theory to make inference from data

$$P(C|\boldsymbol{x}) = \frac{P(\boldsymbol{x}|C)P(C)}{P(\boldsymbol{x})}$$

A kind of diagnosis

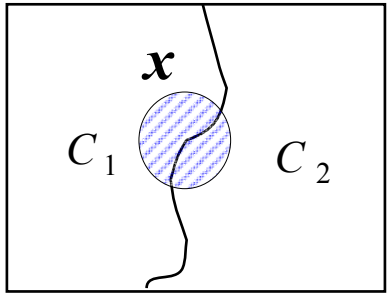
  - $\boldsymbol{x}$: observed data (variable)
  - C : class hypothesis
  - $P(\boldsymbol{x})$: prior probability of $\boldsymbol{x}$
  - $P(C)$: prior probability of C
  - $P(\boldsymbol{x}|C)$: probability of $\boldsymbol{x}$ given C

Classification (3/5)

- Calculate the posterior probability of the concept C (C_1 or C_2) after having the observation $\boldsymbol{x}$
  - Combine the prior and what the data tells using Bayes' rule

$$P(C|\boldsymbol{x}) = \frac{P(\boldsymbol{x}|C)P(C)}{P(\boldsymbol{x})}$$

$$posterior = \frac{likelihood \times prior}{evidence}$$



  - C_1 and C_2 are mutually exclusive and exhaustive classes (concepts)

$$\therefore P(\mathbf{x}) = P(\mathbf{x} \cap C_1) + P(\mathbf{x} \cap C_2)$$
$$= P(\mathbf{x}|C_1)P(C_1) + P(\mathbf{x}|C_2)P(C_2)$$

Classification (4/5)

- Bayes' Classifier: Extended to K mutually exclusive and exhaustive classes

$$\forall_{i \neq j} P\left(C_i \cap C_j\right) = 0 \quad , \text{ and } \quad \sum_{i=1}^{K} P\left(C_i\right) = 1$$

$$P\left(C_i \middle| \mathbf{x}\right) = \frac{P\left(\mathbf{x} \middle| C_i\right) P\left(C_i\right)}{P\left(\mathbf{x}\right)} = \frac{P\left(\mathbf{x} \middle| C_i\right) P\left(C_i\right)}{\sum_{k=1}^{K} P\left(\mathbf{x} \middle| C_k\right) P\left(C_k\right)}$$

Classification (5/5)

- Maximum likelihood classifier
  - The posterior probability that the data belongs to class C_i

$$L_i(\boldsymbol{x}) = P(\boldsymbol{x}|C_i)$$

  - Have the same classification result as that of Bayes' Classifier, if the prior probability $P(C_i)$ is assumed to be equal to each other

$$\max_i L_i(\boldsymbol{x}) = \max_i P(\boldsymbol{x}|C_i) \quad \stackrel{?}{=} \quad \max_i P(C_i|\boldsymbol{x}) = \max_i \frac{P(\boldsymbol{x}|C_i)P(C_i)}{P(\boldsymbol{x})}$$

Classification: Illustrative Case 2 (1/3)

- Does a patient have cancer (C_1) or not (C_2) ?
  - A patient takes a lab test, and result would be $x = "+"$ or $x = "-"$

    1. If the result comes back positive ($x = "+"$)

    2. And we also knew that the test returns a correct positive result (+) in only 98% of the cases in which the disease is actually present ($P(+|C_1) = 0.98$) and a correct negative result (-) in only 97% of the cases in which the disease is not present ($P(-|C_2) = 0.97$)

       Furthermore, 0.008 of the entire population have this cancer ($P(C_1) = 0.008$)

Classification: Illustrative Case 2 (2/3)

- Bayes' Classifier:

$$P(C_1|+) = \frac{P(+|C_1)P(C_1)}{P(+|C_2)P(C_2) + P(+|C_1)P(C_1)}$$

$$= \frac{0.98 \times 0.008}{0.03 \times 0.992 + 0.98 \times 0.008}$$

$$\approx \frac{0.0078}{0.298 + 0.0078}$$

$$\approx 0.21$$

$$P(C_2|+) = \frac{P(+|C_2)P(C_2)}{P(+|C_2)P(C_2) + P(+|C_1)P(C_1)}$$

$$= \frac{0.03 \times 0.992}{0.03 \times 0.992 + 0.98 \times 0.008}$$

$$\approx \frac{0.298}{0.298 + 0.0078}$$

$$\approx 0.79$$

$\Rightarrow$ $P(C_1|+) \approx 0.21$

$P(C_2|+) \approx 0.79$ ✓

Classification: Illustrative Case 2 (3/3)

- Maximum likelihood classifier:

$$P(+|C_1) = 0.98 \quad \checkmark$$
$$P(+|C_2) = 0.03$$

Losses and Risks

- Decisions are not always prefect
  - E.g., “Loan Application”
    - The loss for a high-risk applicant erroneously accepted (false acceptance) may be different from that for an erroneously rejected low-risk applicant (false rejection)
  - Much critical for other cases such as medical diagnosis or earthquake prediction

Expected Risk (1/2)

- Def: the Expected Risk for taking action α_i
  - Hypothesize that the example $\boldsymbol{x}$ belongs to class i
  - Suppose the example actually belongs to some class k

$$R(\alpha_i|\boldsymbol{x}) = \sum_{k=1}^{K} \lambda_{ik} P(C_k|\boldsymbol{x})$$

$$\lambda_{ik} = \begin{cases} 0 & \text{if } i = k \\ 1 & \text{if } i \neq k \end{cases}$$

    - A zero-one loss function
    - All correct decisions have no loss and all error are equally costly

- Choose the action α_i with minimum risk

$$\alpha = \arg\min_j R(\alpha_j|\boldsymbol{x})$$

Expected Risk (2/2)

- Choosing the action α_i with minimum risk is equivalent to choosing the class C_i with the highest posterior probability

$$R(\alpha_i|\boldsymbol{x}) = \sum_{k=1}^{K} \lambda_{ik} P(C_k|\boldsymbol{x})$$

$$= \sum_{k=1,k\neq i}^{K} P(C_k|\boldsymbol{x})$$

$$= 1 - P(C_i|\boldsymbol{x})$$

  - Choose the action α_i with

$$\alpha_i = \arg\max_j P(C_j|\boldsymbol{x})$$

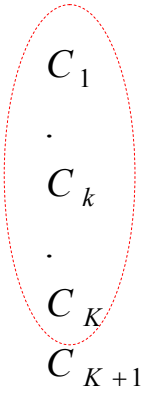
Expected Risk: Reject Action Involved (1/3)

- Manual Decisions ?
  - Wrong decisions (misclassification) may have very high cost
  - Resort to a manual decision when automatic system has low certainty of its decision

- Define an additional action of reject (or doubt) α_{K+1}

$$\lambda_{ik} = \begin{cases} 0 & \text{if } i = k \\ \lambda & \text{if } i = K+1 \\ 1 & \text{otherwise} \end{cases}$$

$\boldsymbol{x} \rightarrow C_i$



  - $\lambda \ (0 \le \lambda \le 1)$ is the loss incurred for choosing the (K+1)st action of reject

Expected Risk: Reject Action Involved (2/3)

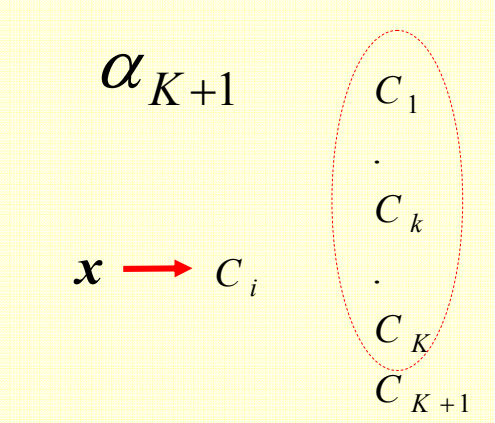
- The risk for choosing the reject ((*K*+1)st) action α_{K+1}

$$R(\alpha_{K+1}|\boldsymbol{x}) = \sum_{k=1}^{K} \lambda P(C_k|\boldsymbol{x})$$

$\alpha_i = \alpha_{K+1}$

$$= \lambda \sum_{k=1}^{K} P(C_k|\boldsymbol{x})$$

$$= \lambda$$



- Recall that the risk for choosing action $\alpha_i \ (1 \le i \le K)$

$$R(\alpha_i|\boldsymbol{x}) = \sum_{k=1}^{K} \lambda_{ik} P(C_k|\boldsymbol{x})$$

$$= \sum_{k=1,k\neq i}^{K} P(C_k|\boldsymbol{x})$$

$$= 1 - P(C_i|\boldsymbol{x})$$

Expected Risk: Reject Action Involved (3/3)

- The optimal decision rule is to:

$$\text{Choose } C_i \text{ if } R(\alpha_i|\boldsymbol{x}) < R(\alpha_j|\boldsymbol{x}) \text{ for all } 1 \le j \le K, j \ne i$$
$$\text{and } R(\alpha_i|\boldsymbol{x}) < R(\alpha_{K+1}|\boldsymbol{x})$$
$$\text{Reject if } R(\alpha_{K+1}|\boldsymbol{x}) < R(\alpha_j|\boldsymbol{x}) \text{ for all } 1 \le j \le K$$

- That is

$$\text{Choose } C_i \text{ if } P(C_i|\boldsymbol{x}) > P(C_j|\boldsymbol{x}) \text{ for all } 1 \le j \le K, j \ne i$$
$$\text{and } P(C_i|\boldsymbol{x}) > 1 - \lambda$$
$$\text{Reject otherwiese}$$

  - When $\lambda = 0 \rightarrow$ always reject the chosen action
  - When $\lambda = 1 \rightarrow$ always accept the chosen action

Discriminant Functions (1/3)

- Classification can be thought of as a set of discriminant functions, $g_i(\boldsymbol{x}), i = 1,...,K$, for each class such that
  - Choose C_i if $g_i(\boldsymbol{x}) = \max_k g_k(\boldsymbol{x})$
- $g_i(\boldsymbol{x})$ can be expressed by using the Bayes's classifier (with minimum risk and no additional action of reject)

$$g_i(\boldsymbol{x}) = -R(\alpha_i|\boldsymbol{x})$$

  - If the zero-one loss function is imposed, $g_i(\boldsymbol{x})$ can also be expressed by

$$g_i(\boldsymbol{x}) = P(C_i|\boldsymbol{x})$$

    - With the same ranking result, we can have $g_i(\boldsymbol{x}) = P(\boldsymbol{x}|C_i)P(C_i)$

Discriminant Functions (2/3)

- The instance space thus can be divided into K decision regions $R_1,..., R_K$, where

$$R_i = \left\{ \boldsymbol{x} \middle| g_i(\boldsymbol{x}) = \max_k g_k(\boldsymbol{x}) \right\}$$

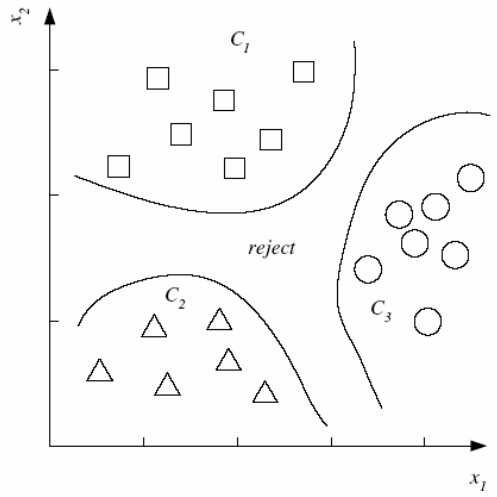


Figure 3.1: Example of decision regions and decision boundaries.

- Ties occur among the largest discriminant functions ?

Discriminant Functions (3/3)

- For two-class problems, we can merely define a single discriminant function

$$g(\boldsymbol{x}) = g_1(\boldsymbol{x}) - g_2(\boldsymbol{x})$$

$$\text{Choose} \begin{cases} C_1 & \text{if } g(\boldsymbol{x}) > 0 \\ C_2 & \text{otherwise} \end{cases}$$

  - The classification system is a dichotomizer

- When class number is larger than or equal to 3, the classification system is a poly-chotomizer

Choosing Hypotheses* : *MAP* Criterion

- In machine learning, we are interested in finding the best (most probable) hypothesis (classifier) h_c from some hypothesis space *H*, given the observed training data set $X_{c_i} = \left\{ \left(x^t, r^t \right) \middle| r^t = c_i, \; t = 1, 2, \cdots, n \right\}$

$$h_{MAP} = \arg \max_{h_{c_i} \in H} P\left(h_{c_i} \middle| \boldsymbol{X}_{c_i} \right)$$

$$= \arg \max_{h_{c_i} \in H} \frac{P\left(\boldsymbol{X}_{c_i} \middle| h_{c_i} \right) P\left(h_{c_i} \right)}{P\left(\boldsymbol{X}_{c_i} \right)}$$

$$= \arg \max_{h_{c_i} \in H} P\left(\boldsymbol{X}_{c_i} \middle| h_{c_i} \right) P\left(h_{c_i} \right)$$

  - A Maximum a Posteriori (*MAP*) hypothesis h_{MAP}

Choosing Hypotheses*: *ML* Criterion

- If we further assume that every hypothesis is equally probable a priori, e.g. $P(h_{c_i}) = P(h_{c_j})$. The above equation can be simplified as:

$$h_{ML} = \arg \max_{h_{c_i} \in H} P\left(\boldsymbol{X}_{c_i} \middle| h_{c_i}\right)$$

  - A Maximum Likelihood (*ML*) hypothesis h_{ML}

- $P\left(\boldsymbol{X}_{c_i} \middle| h_{c_i}\right)$ often called "the likelihood of the data set $\boldsymbol{X}_{c_i}$ given h_{c_i}"

Naïve Bayes Classifier (1/3)

- A simplified approach to the Bayes's classifier
  - The attributes $x_1, x_2, \ldots, x_d$ of an instance/example $\boldsymbol{x}$ are assumed to be independent conditioned on a given class hypothesis
    - Naïve Bayes assumption:

$$P(\boldsymbol{x}|C_j) = P(x_1, x_2, \ldots, x_d|C_j) = \prod_{n=1}^{d} P(x_n|C_j)$$

  - Naïve Bayes Classifier:

$$\begin{aligned} C_{MAP} &= \arg\max_{C_k} P(C_k|\boldsymbol{x}) \\ &= \arg\max_{C_k} \frac{P(\boldsymbol{x}|C_k)P(C_k)}{P(\boldsymbol{x})} = \arg\max_{C_k} P(\boldsymbol{x}|C_k)P(C_k) \\ &= \arg\max_{C_k} P(x_1, \ldots, x_d|C_k)P(C_k) \\ &= \arg\max_{C_k} P(C_k)\prod_{n=1}^{d} P(x_n|C_k) \end{aligned}$$

Naïve Bayes Classifier (2/3)

- Illustrative case 1
  - Given a data set with 3-dimensional Boolean examples $\boldsymbol{x} = (x_A, x_B, x_C)$, train a naïve Bayes classifier to predict the classification

Attribute A	Attribute B	Attribute C	Classification D
F	T	F	T
F	F	T	T
T	F	F	T
T	F	F	F
F	T	T	F
F	F	T	F

$P(D = T) = 1/2,\ P(D = F) = 1/2$

$P(A = T|D = T) = 1/3, P(A = F|D = T) = 2/3$
$P(B = T|D = T) = 1/3, P(B = F|D = T) = 2/3$
$P(C = T|D = T) = 1/3, P(B = F|D = T) = 2/3$

$P(A = T|D = F) = 1/3, P(A = F|D = F) = 2/3$
$P(B = T|D = F) = 1/3, P(B = F|D = F) = 2/3$
$P(C = T|D = F) = 2/3, P(B = F|D = F) = 1/3$

  - What is the predicted probability $P(D = T|A = T, B = F, C = T)$?
  - What is the predicted probability $P(D = T|B = T)$?

Naïve Bayes Classifier (3/3)

- Illustrative case 1 (cont.)

(i) $P(D=T|A=T,B=F,C=T)$

$$= \frac{P(A=T,B=F,C=T|D=T)P(D=T)}{P(A=T,B=F,C=T)}$$

$$= \frac{P(A=T,B=F,C=T|D=T)P(D=T)}{P(A=T,B=F,C=T|D=T)P(D=T)+P(A=T,B=F,C=T|D=F)P(D=F)}$$

$$= \frac{\frac{1}{3}\cdot\frac{2}{3}\cdot\frac{1}{3}\cdot\frac{1}{2}}{\frac{1}{3}\cdot\frac{2}{3}\cdot\frac{1}{3}\cdot\frac{1}{2}+\frac{1}{3}\cdot\frac{2}{3}\cdot\frac{2}{3}\cdot\frac{1}{2}} = \frac{2}{2+4} = \frac{1}{3}$$

(ii) $P(D=T|B=T)$

$$= \frac{P(B=T|D=T)P(D=T)}{P(B=T)}$$

$$= \frac{P(B=T|D=T)P(D=T)}{P(B=T|D=T)P(D=T)+P(B=T|D=F)P(D=F)}$$

$$= \frac{\frac{1}{3}\cdot\frac{1}{2}}{\frac{1}{3}\cdot\frac{1}{2}+\frac{1}{3}\cdot\frac{1}{2}} = \frac{1}{1+1} = \frac{1}{2}$$

How to Train a Naïve Bayes Classifier

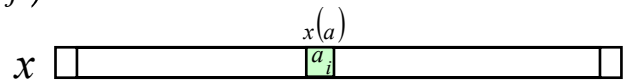
- Naïve_Bayes_Learn(*examples*)

  For each target value v_j

  $\hat{P}(v_j) \leftarrow$ maximum likelihood (ML) estimate of $P(v_j)$ $\quad \frac{|v_j|}{n} or \frac{|v_j|}{\sum_{v_k} |v_k|}$

  For each attribute value a_i of each attribute a

  $\hat{P}(a_i | v_j) \leftarrow$ maximum likelihood (ML) estimate of $P(a_i | v_j)$



$$\frac{\sum_{x \in v_j, x(a)=a_i} 1}{\sum_{x \in v_j} 1} = \frac{\sum_{x \in v_j, x(a)=a_i} 1}{|v_j|}$$

- Classify_New_Instance(*x*)

$$v_{NB} = \arg \max_{v_j \in V} P(v_j) \prod_{a_i \in x} P(a_i | v_j)$$

Naïve Bayes: Example 2

- Consider *PlayTennis* again and new instance

  <*Outlook=sunny*, *Temperature=cool*, *Humidity=high*, *Wind=strong*>

- Want to compute

$$v_{NB} = \arg\max_{v_j \in V=\{yes,no\}} P(v_j) \times P(Outlook = sunny | v_j) \times P(Temperature = cool | v_j) \times P(Humidity = high | v_j) \times P(Wind = Strong | v_j)$$

$$P(yes) \times P(Outlook = sunny | yes) \times P(Temperature = cool | yes) \times P(Humidity = high | yes) \times P(Wind = Strong | yes) = 0.0053$$

$$P(no) \times P(Outlook = sunny | no) \times P(Temperature = cool | no) \times P(Humidity = high | no) \times P(Wind = Strong | no) = 0.206$$

$$\therefore v_{NB} = no$$

Dealing with Data Sparseness

- What if none of the training instances with target value v_j have attribute value a_i ? Then

$$\hat{P}(a_i|v_j) = 0, \text{ and } \ldots$$

$$v_{NB} = \arg\max_{v_j \in V} \hat{P}(v_j) \prod_i \hat{P}(a_i|v_j)$$

  - Typical solution is Bayesian estimate for $\hat{P}(a_i|v_j)$

$$\hat{P}(a_i|v_j) \leftarrow \frac{n_c + mp}{n + m}$$ Smoothing

    - n is number of training examples for which $v = v_j$
    - n_c is number of training examples for which $v = v_j$ and $a = a_i$
    - p is prior estimate for $\hat{P}(a_i|v_j)$
    - m is weight given to prior (i.e., number of "virtual" examples)

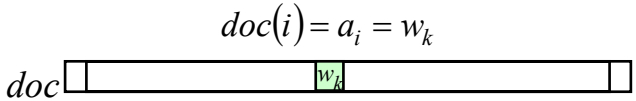
Example: Learning to Classify Text (1/4)

- For instance,
  - Learn which news articles are of interest
  - Learn to classify web pages by topic
- Naïve Bayes is among the most effective algorithms
- What attributes shall we use to represent text documents
  - The word occurs in each document position

Example: Learning to Classify Text (2/4)

- Target Concept: *Interesting* ? Document $\rightarrow \{+,-\}$

  1. Represent each document by vector of words
     - one attribute per word position in document
  2. Learning Use training examples to estimate
     - $P(+)$
     - $P(-)$
     - $P(doc|+)$
     - $P(doc|-)$



- Naïve Bayes conditional independence assumption

$$P(doc|v_j) = \prod_{i=1}^{length(doc)} P(a_i = w_k | v_j)$$

  - Where $P(a_i = w_k | v_j)$ is probability that word in position *i* is w_k, given v_j
  - One more assumption: $P(a_i = w_k | v_j) = P(a_m = w_k | v_j), \forall i, m$

Time Invariant

Example: Learning to Classify Text (3/4)

- Learn_Naïve_Bayes_Text(*Examples*, *V*)

1. Collect all words and other tokens that occur in *Examples*
   - *Vocabulary* ← all distinct words and other tokens in Examples
2. Calculate the required $P(v_j)$ and $P(w_k|v_j)$ probability terms
   - $docs_j$ ← subset of Examples for which the target value is v_j
   - $P(v_j) \leftarrow \frac{|docs_j|}{|Examples|}$
   - $Text_j$ ← a single document created by concatenating all members of $docs_j$
   - n ← total number of words in $Text_j$ (counting duplicate words multiple times)
   - For each word w_k in *Vocabulary*
     - n_k ← number of times word w_k occurs in
     - $P(w_k|v_j) \leftarrow \frac{n_k + 1}{n + |Vocabulary|}$ Smoothed unigram

Example: Learning to Classify Text (4/4)

- Classify_Naïve_Bayes_Text(*Doc*)
  - *positions* ← all word positions in *Doc* that contain tokens found in *Vocabulary*
  - Return v_{NB}, where

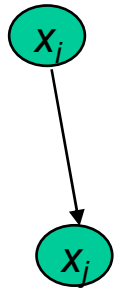
$$v_{NB} = \arg\max_{v_j \in V} P(v_j) \prod_{i \in positions} P(a_i | v_j)$$

Bayesian Networks (1/3)

- Premise
  - Naïve Bayes assumption of conditional independence too restrictive
  - But it is intractable without some such assumptions
  - Bayesian networks describe conditional independence among subsets of variables
    - Allows combining prior knowledge about (in)dependencies among variables with observed training data
- Bayesian Networks also called
  - Bayesian Belief Networks, Bayes Nets, Belief Networks, Probabilistic Networks, Graphical Models etc.

Bayesian Networks (2/3)

- A simple, graphical notation for conditional independence assertions and hence for compact specification of full joint distributions



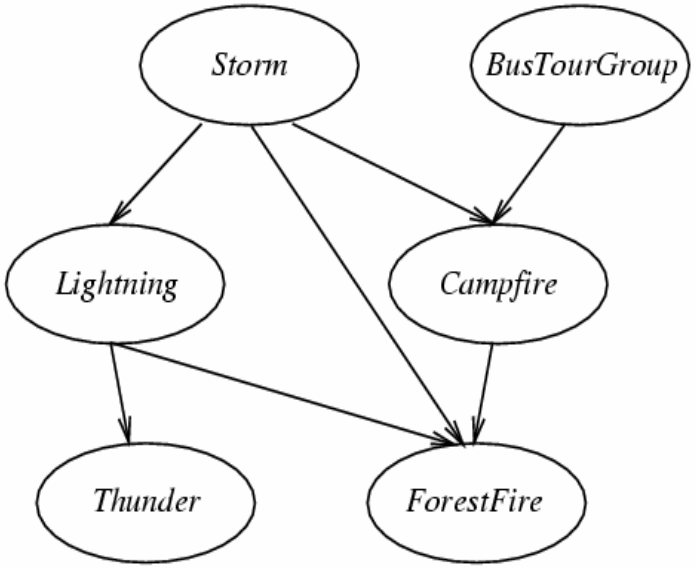
- Syntax
  - A set of nodes, one per variable (discrete or continuous)
    - For discrete variable, they can be either binary or not
  - A directed, acyclic graph (link/arrow ≈ "directly influences")
  - A conditional distribution for each node given its parents

$$P\left(X_i \middle| Parents\left(X_i\right)\right)$$

- In the simplest case, conditional distribution represented as a Conditional Probability Table (CPT) giving the distribution over $P\left(X_i\right)$ for each combination of parent values

Bayesian Networks (3/3)

– E.g., nodes of discrete binary variables



	S,B	$S,\neg B$	$\neg S,B$	$\neg S,\neg B$
C	0.4	0.1	0.8	0.2
$\neg C$	0.6	0.9	0.2	0.8

Campfire

Conditional Probability Table (CPT)

S	B	P(C)
T	T	0.4
T	F	0.1
F	T	0.8
F	F	0.2

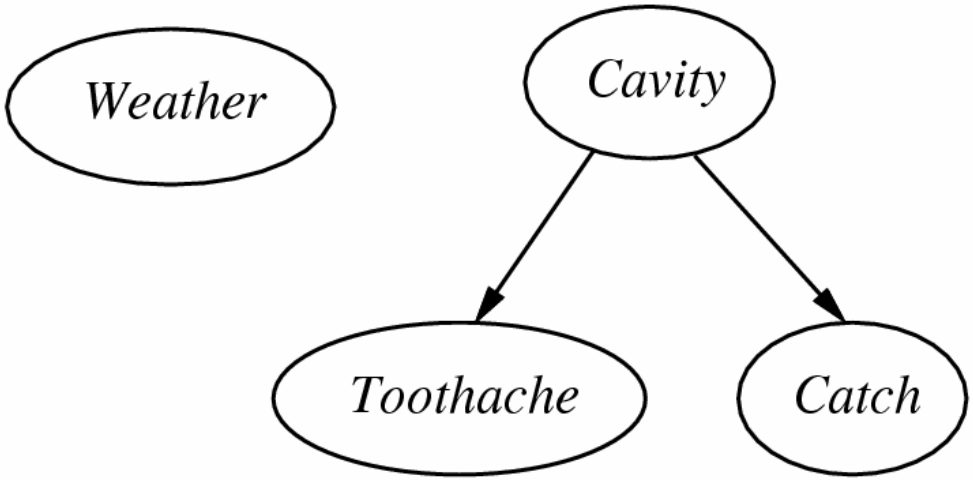
binary random variables

– Each node is asserted to be conditionally independent of its nondescendants, given its immediate predecessors

– Directed acyclic graph

Example 1:Dentist Network

- Topology of network encodes conditional independence assertions



  - Weather is independent of the other variables
  - Toothache and Catch are conditionally independent given Cavity
    - Cavity is a direct cause of Toothache and Catch (Cavity has a direct influence on Toothache and Catch)

Conditional (In)dependence (1/5)

- Definition: *X* is conditionally independent of *Y* given *Z* if the probability distribution governing *X* is independent of the value of *Y* given the value of *Z*; that is, if

$$\left(\forall x_i, y_j, z_k\right)\ P\left(X = x_i \middle| Y = y_j, Z = z_k\right) = P\left(X = x_i \middle| Z = z_k\right)$$

  - More compactly, we can write

$$P\left(X \middle| Y, Z\right) = P\left(X \middle| Z\right)$$

- Conditional independence allows breaking down inference into calculation over small group of variables

Conditional (In)dependence (2/5)

- Example: *Thunder* is conditionally independent of *Rain* given *Lightning*

  P(*Thunder*|*Rain*, *Lightning*)= P(*Thunder*|*Lightning*)

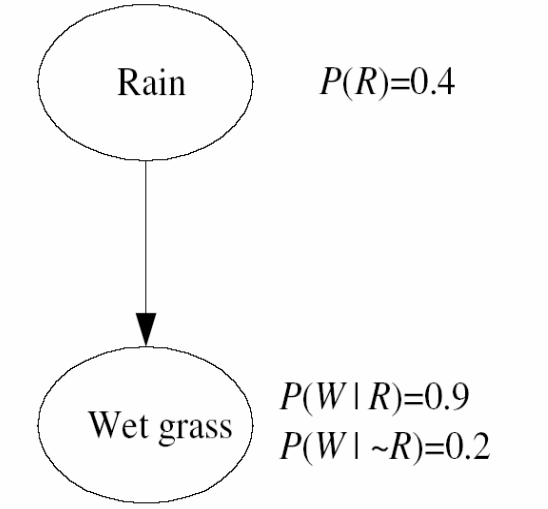
- Recall that Naïve Bayes uses conditional independence to justify

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

X,Y are mutually independent given Z

Conditional (In)dependence (3/5)

- Bayesian Network also can be thought of as a causal graph that illustrates causalities between variables
  - We can make a diagnostic inference from the it $P(R|W) = ?$



prediction

diagnosis

$$P(R|W) = \frac{P(W|R)P(R)}{P(W)}$$

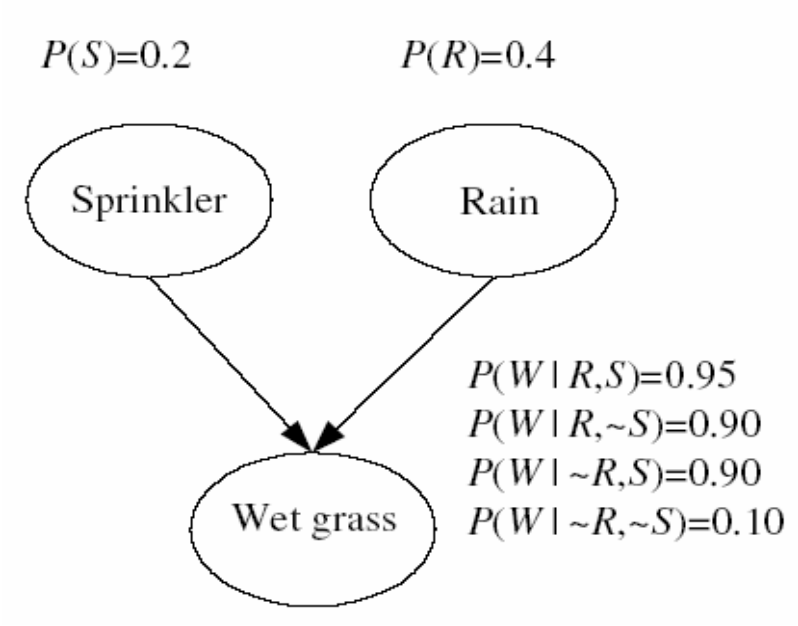
$$= \frac{P(W|R)P(R)}{P(W|R)P(R) + P(W|\neg R)P(\neg R)}$$

$$= \frac{0.9 \times 0.4}{0.9 \times 0.4 + 0.2 \times 0.6}$$

$$\approx 0.75 \quad (> P(R) = 0.4)$$

Conditional (In)dependence (4/5)

- Suppose that sprinkler is included as another cause of wet grass



– Predictive inference

$$
\begin{aligned}
& P(W|S) \\
&= P(W,R|S) + P(W,\neg R|S) \\
&= P(W|R,S)P(R|S) + P(W|\neg R,S)P(\neg R|S) \\
&= P(W|R,S)P(R) + P(W|\neg R,S)P(\neg R) \\
&= 0.95 \times 0.4 + 0.9 \times 0.6 \\
&\approx 0.92
\end{aligned}
$$

– Diagnostic inference (I)

$$
\begin{aligned}
& P(S|W) \\
&= \frac{P(W|S)P(S)}{P(W)} \\
&\approx \frac{0.92 \times 0.2}{0.52} \\
&\approx 0.35 \quad (> P(S) = 0.2)
\end{aligned}
$$

$$
\begin{aligned}
P(W) &= P(W|R,S)P(R,S) + P(W|\neg R,S)P(\neg R,S) \\
&\quad + P(W|R,\neg S)P(R,\neg S) + P(W|\neg R,\neg S)P(\neg R,\neg S) \\
&= P(W|R,S)P(R)P(S) + P(W|\neg R,S)P(\neg R)P(S) \\
&\quad + P(W|R,\neg S)P(R)P(\neg S) + P(W|\neg R,\neg S)P(\neg R)P(\neg S) \\
&= 0.95 \times 0.4 \times 0.2 + 0.9 \times 0.6 \times 0.2 + 0.9 \times 0.4 \times 0.8 + 0.1 \times 0.6 \times 0.8 \\
&= 0.52
\end{aligned}
$$

Conditional (In)dependence (5/5)

– Diagnostic inference (II)

$$P(S \mid R, W)$$

independent

$$= \frac{P(W \mid R, S)P(S \mid R)}{P(W \mid R)}$$

$$= \frac{P(W \mid R, S)P(S)}{P(W \mid R)}$$

$$= \frac{0.95 \times 0.2}{0.91}$$

$$\approx 0.21 \qquad (> P(S) = 0.2)$$

$$P(W \mid R)$$
$$= P(W \mid R, S)P(S \mid R) + P(W \mid R, \neg S)P(\neg S \mid R)$$
$$= P(W \mid R, S)P(S) + P(W \mid R, \neg S)P(\neg S)$$
$$= 0.95 \times 0.2 + 0.9 \times 0.8$$
$$= 0.91$$

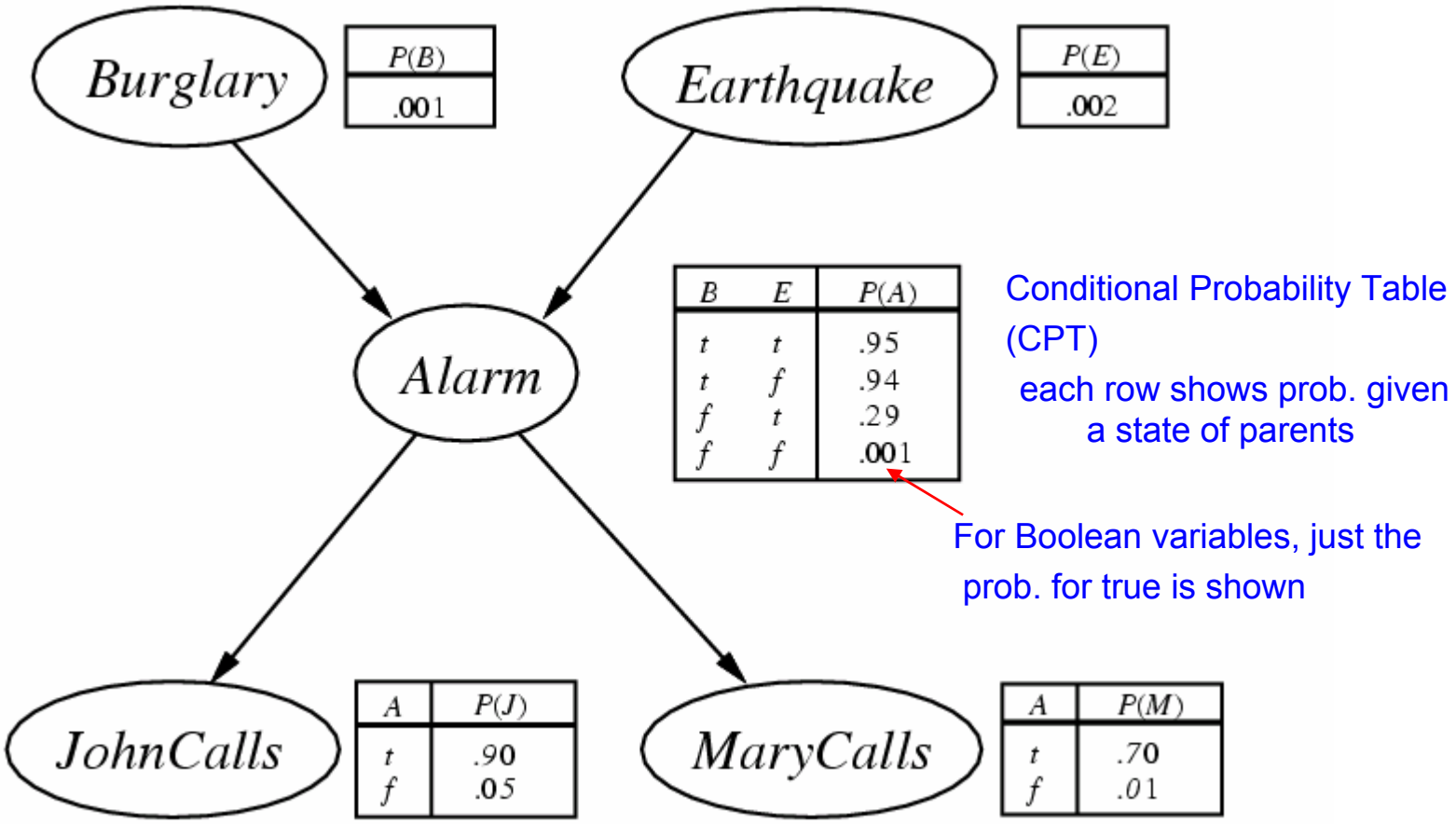
Example 2: Burglary Network (1/2)

- You're at work, neighbor John calls to say your alarm is ringing, but neighbor Mary doesn't call. Sometimes it's set off by minor earthquakes. Is there a burglar?

$$P(Burglary = T | JohnCall = T, MaryCall = F)?$$

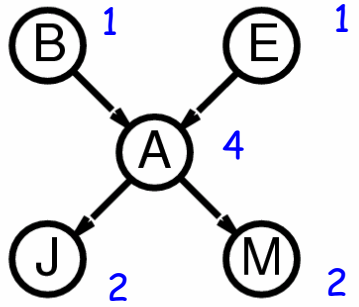
  - Variables: Burglar, Earthquake, Alarm, JohnCalls, MaryCalls
  - Network topology reflects "causal" knowledge
    - A burglar can set the alarm off
    - An earthquake can set the alarm off
    - The alarm can cause Mary to call
    - The alarm can cause John to call
  - But
    - John sometimes confuses the telephone ringing with the alarm
    - Mary likes rather loud music and sometimes misses the alarm

Example 2: Burglary Network (2/2)



Compactness

- A CPT for Boolean X_i with k Boolean (true/false) parents has 2^k rows for the combinations of parent values
- 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?) 2^0 2^0 2^2 2^1 2^1



Chain rule

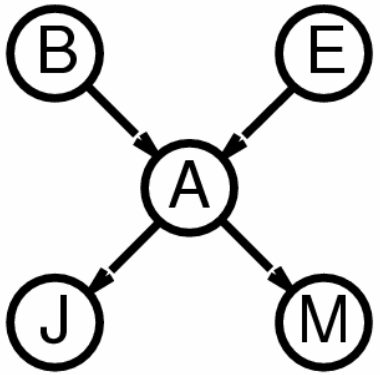
$$P(B,E,A,J,M)$$
$$= P(B)P(E|B)P(A|B,E)P(J|B,E,A)P(M|B,E,A,J)$$
$$\approx P(B)P(E)P(A|B,E)P(J|A)P(M|A)$$

Global Semantics

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

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

  - The Bayesian Network is semantically
    - A representation of the joint distribution
    - A encoding of a collection of conditional independence statements

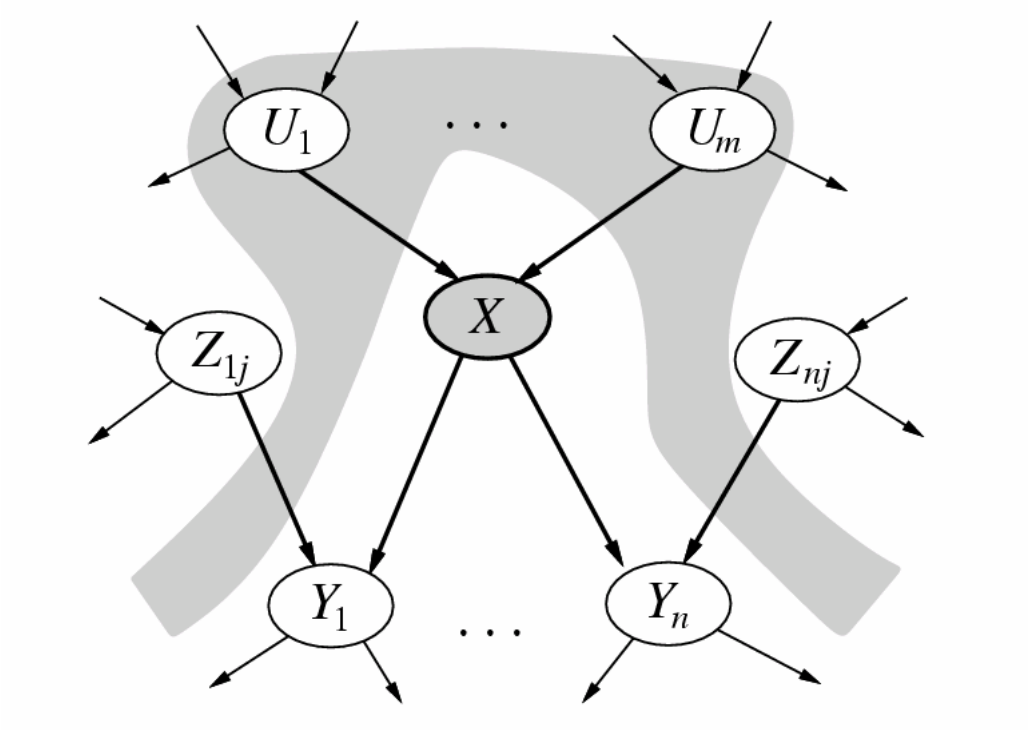


- E.g.,

$$P(J \wedge M \wedge A \wedge \neg B \wedge \neg E)$$
$$\approx P(J|A)P(M|A)P(A|\neg B \wedge \neg E)P(\neg B)P(\neg E)$$
$$= 0.90 \times 0.70 \times 0.001 \times 0.999 \times 0.998$$
$$= 0.00062$$

Local Semantics

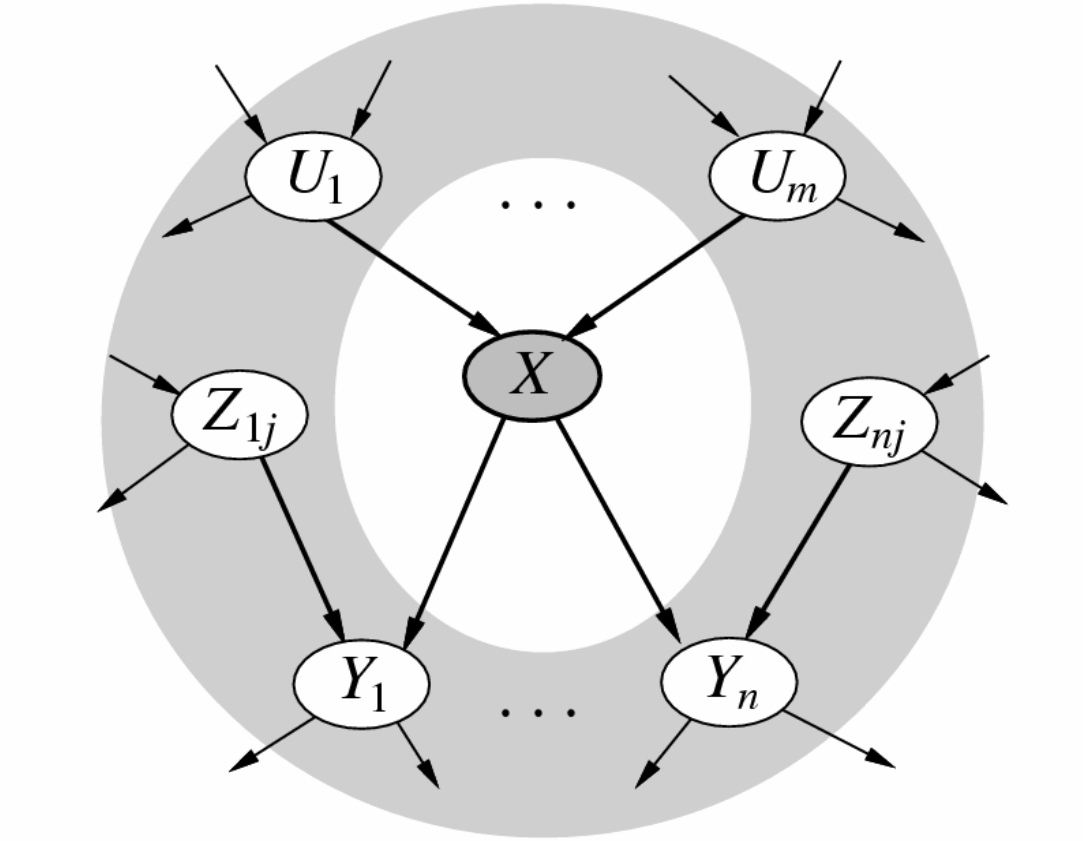
- Local semantics: each node is conditionally independent of its nondescendants given its parents



  - Local semantics ⟺ global semantics

Markov Blanket

- Each node is conditionally independent of all others
  given its parents + children + children's parents



Constructing Bayesian Networks

- Need a method such that a series of locally testable assertions of conditional independence guarantees the required global semantics
  1. Choose an ordering of variables $X_1,..X_i..,X_n$
  2. For *i*=1 to *n*

     add X_i to the network and select parents from $X_1,..X_{i-1}$ such that

     $Parents(X_i) \subseteq \{X_1,..X_{i-1}\}$

     $$P(X_i|X_1,..X_{i-1}) = P(X_i|Parents(X_i))$$

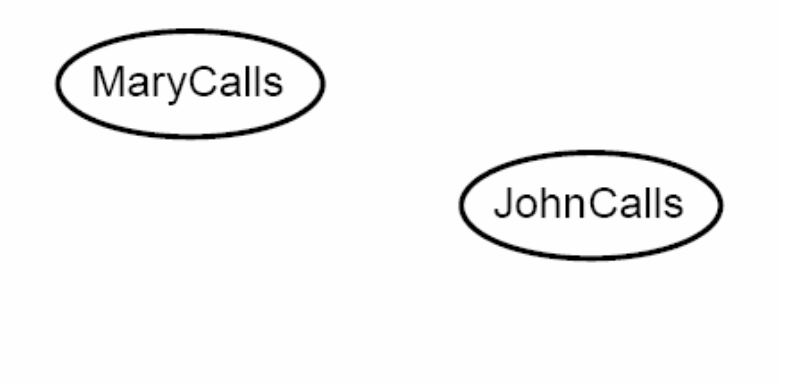
     This choice of parents guarantees the global semantics

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

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

Example for Constructing Bayesian Network (1/6)

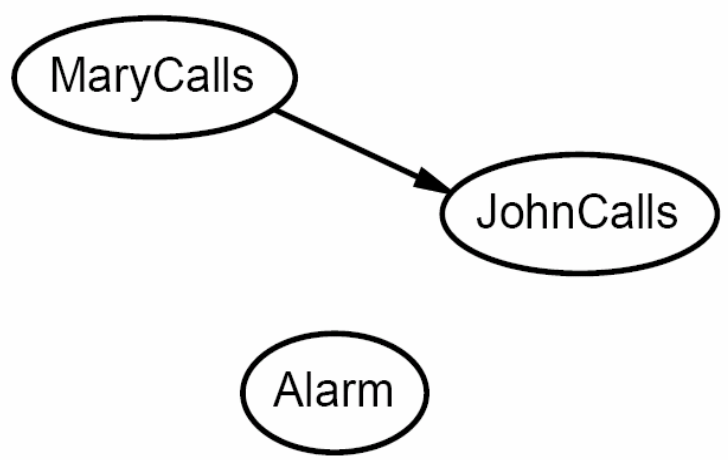
- Suppose we choose the ordering: *M, J, A, B, E*



  - $P(J|M) = P(J)$?

Example for Constructing Bayesian Network (2/6)

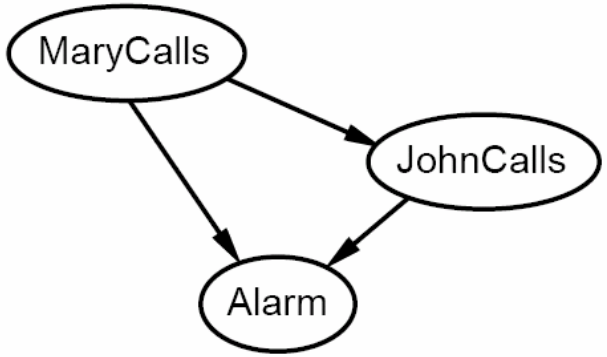
- Suppose we choose the ordering: *M, J, A, B, E*



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

Example for Constructing Bayesian Network (3/6)

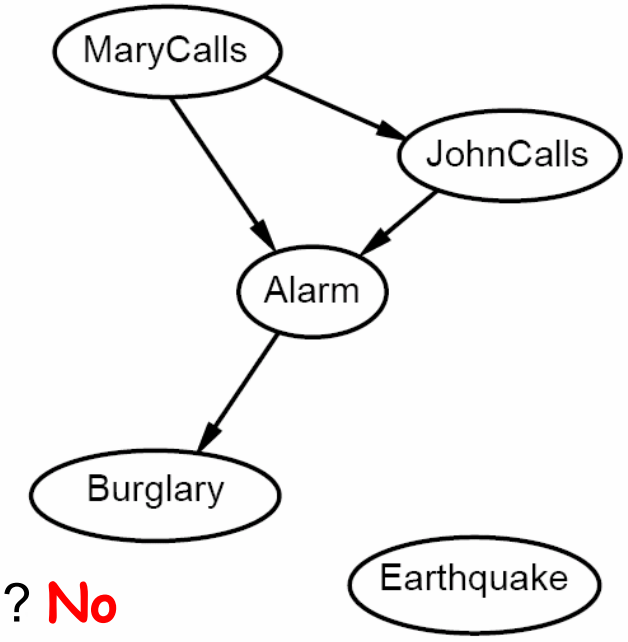
- Suppose we choose the ordering: *M, J, A, B, E*



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

Example for Constructing Bayesian Network (4/6)

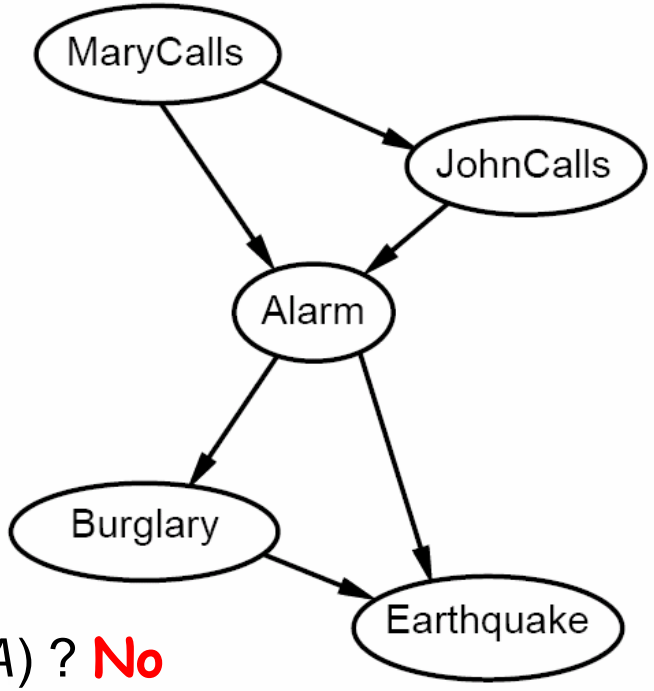
- Suppose we choose the ordering: *M, J, A, B, E*



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

Example for Constructing Bayesian Network (5/6)

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

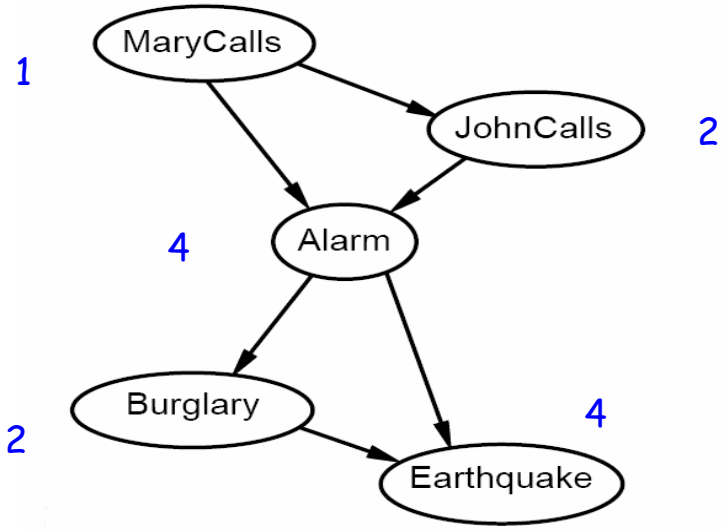


  - $P(J|M) = P(J)$? **No**
  - $P(A|J,M) = P(A|J)$? **No** $P(A|J,M) = P(A)$? **No**
  - $P(B|A,J,M) = P(B|A)$? **Yes**
  - $P(B|A,J,M) = P(B)$? **No**
  - $P(E|B,A,J,M) = P(E|A)$? **No**
  - $P(E|B,A,J,M) = P(E|B,A)$? **Yes**

Example for Constructing Bayesian Network (6/6)

- Summary
  - Deciding conditional independence is hard in noncausal directions

    (Causal models and conditional independence seem hardwired for humans!)
  - Assessing conditional probabilities is hard in noncausal directions
  - Network is less compact: 1 + 2 + 4 + 2 + 4 = 13 numbers needed



Inference Tasks

- Simple queries: compute posterior marginal $P(X_i|E=e)$
  - E.g.,

$$P(Burglary|JohnCalls = true, MarryCalls = true)$$

- Conjunctive queries:

$$P(X_i, X_j|E=e) = P(X_i|X_j, E=e)P(X_j|E=e)$$

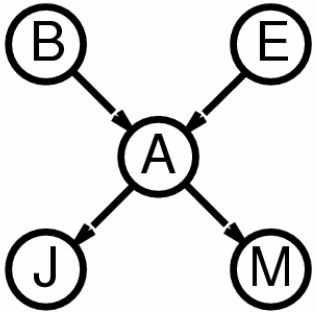
- Optimal decisions: probabilistic inference

$$P(Outcome|Action, Evidence)$$

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

$$P(B|j,m) = \frac{P(B,j,m)}{P(j,m)}$$
$$= \alpha P(B,j,m) \qquad (\alpha = 1/P(j,m))$$
$$= \alpha \sum_e \sum_a P(B,e,a,j,m)$$

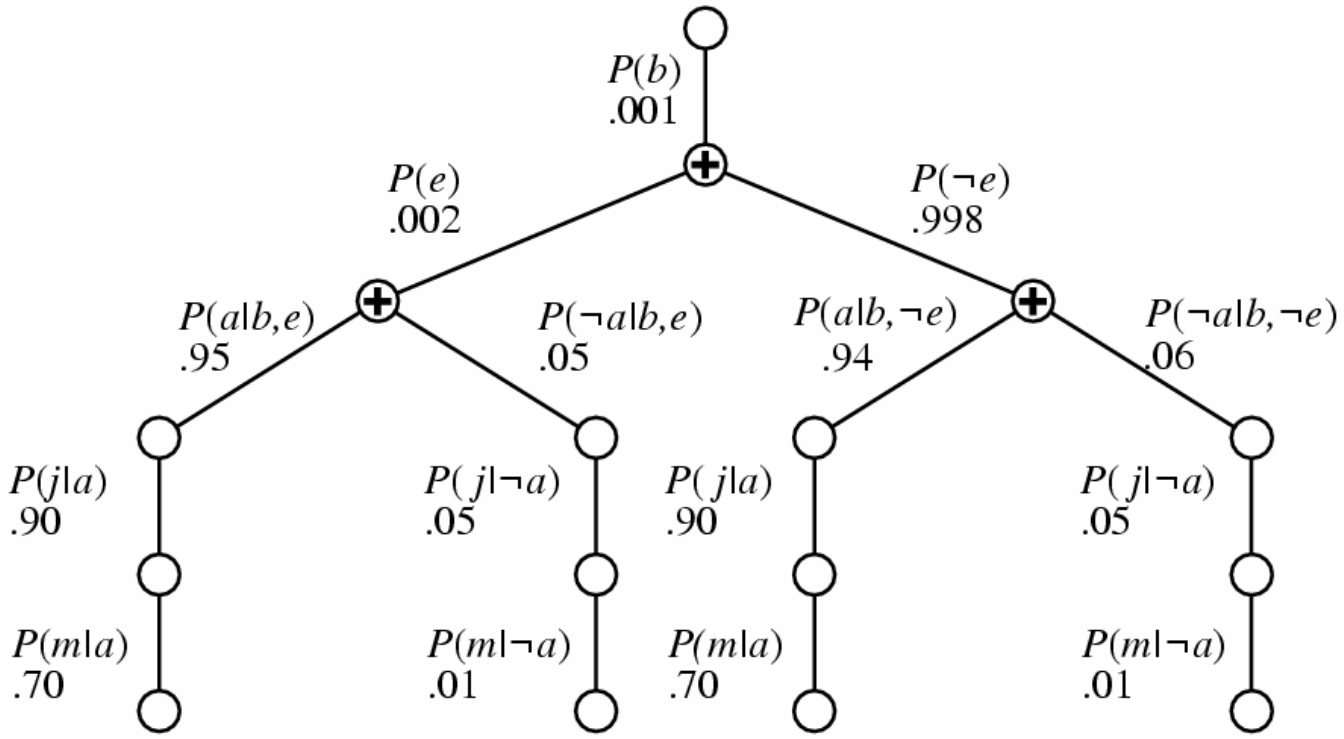


- Rewrite full joint entries using product of CPT entries:

$$P(B|j,m)$$
$$= \alpha \sum_e \sum_a P(B,e,a,j,m)$$
$$= \alpha \sum_e \sum_a P(B)P(e)P(a|B,e)P(j|a)P(m|a)$$
$$= \alpha P(B) \sum_e P(e) \sum_a P(a|B,e)P(j|a)P(m|a)$$

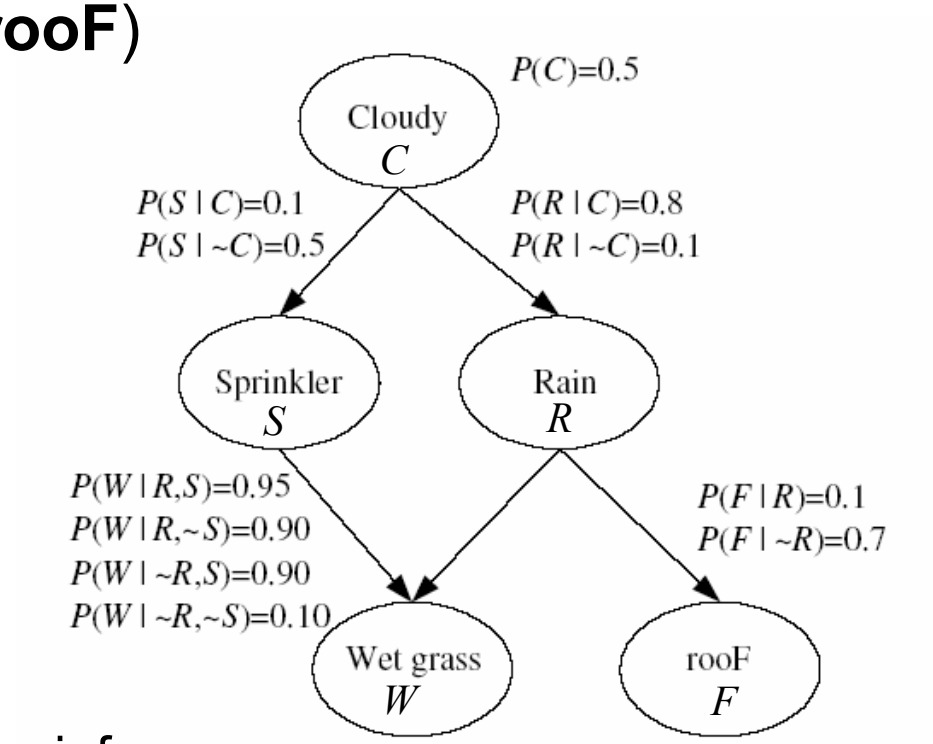
Evaluation Tree

- Enumeration is inefficient: repeated computation\al
  - E.g., computes $P(j|a)P(m|a)$ for each value of e



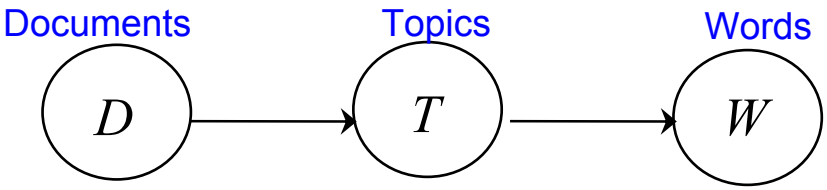
HW-4: Bayesian Networks

- A new binary variable concerning cat making noise on the roof (**rooF**)



  - Predictive inferences
    - $P(F|C) = ?$
    - $P(F|S) = ?$

Bayesian Networks for Information Retrieval



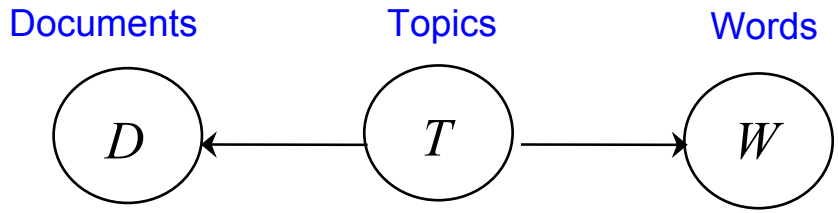
$$P(d_j, w_i) = \sum_{k=1}^{T} P(d_j, t_k, w_i)$$

$$= \sum_{k=1}^{T} P(d_j) P(t_k | d_j) P(w_i | t_k)$$

$$= P(d_j) \sum_{k=1}^{T} P(t_k | d_j) P(w_i | t_k)$$

$$\Rightarrow P(w_i | d_j) = \frac{P(d_j, w_i)}{P(d_j)} = \sum_{k=1}^{T} P(t_k | d_j) P(w_i | t_k)$$

Bayesian Networks for Information Retrieval



$$P(d_j, w_i) = \sum_{k=1}^{T} P(d_j, t_k, w_i)$$

$$= \sum_{k=1}^{T} P(t_k) P(d_j | t_k) P(w_i | t_k)$$

$$\Rightarrow P(w_i | d_j) = \frac{P(d_j, w_i)}{P(d_j)} = \sum_{k=1}^{T} \frac{P(d_j, t_k)}{P(d_j)} P(w_i | t_k) = \sum_{k=1}^{T} P(t_k | d_j) P(w_i | t_k)$$