Machine Learning

Topic 13: Bayesian Belief Networks

Bayes Optimal Classifier

- **Bayes Optimal Classification:** The most probable classification of a new instance is obtained by combining the predictions of all hypotheses, weighted by their posterior probabilities:

$$\arg\max_{v \in V} \sum_{h \in H} P(v \mid h) P(h \mid D)$$

V = is the set of all the values a classification can take

H = the set of hypotheses

Bayes Optimal Classifier

- An advantage of Bayesian Decision Theory
  - it gives us a lower bound on the classification error that can be obtained for a given problem.

- **Bayes Optimal Classification:** The most probable classification of a new instance is obtained by combining the predictions of all hypotheses, weighted by their posterior probabilities:

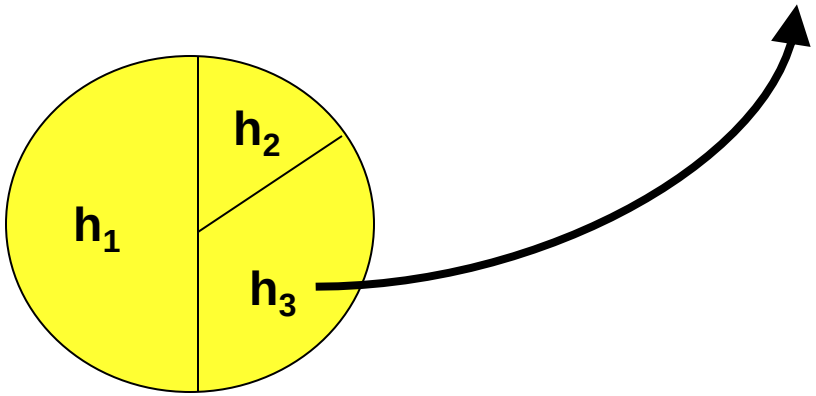
$$\arg\max \sum_{\substack{v \in V \\ h \in H}} P(v \mid h) P(h \mid D)$$

…where V is the set of all the values a classification can take and v is one possible such classification.

Gibbs Classifier

- Bayes optimal classification can be too hard to compute
- Instead, randomly pick a single hypothesis (according to the probability distribution of the hypotheses)
- use this hypothesis to classify new cases

$$\arg\max_{v \in V} P(v \mid h) P(h \mid D)$$



Naïve Bayes Classifier

- Cases described by a conjunction of attribute values
  - These attributes are our "independent" hypotheses
- The target function has a finite set of values, *V*

$$v_{MAP} = \arg\max_{v_j \in V} P(v_j \mid a_1 \wedge a_2 \ldots \wedge a_n)$$

- Could be solved using the joint distribution table
- What if we have 50,000 attributes?
  - Attribute *j* is a Boolean signaling presence or absence of the *jth* word from the dictionary in my latest email.

Naïve Bayes Classifier

$$v_{MAP} = \arg\max_{v_j \in V} P(v_j \mid a_1 \wedge a_2 \ldots \wedge a_n)$$

$$= \arg\max_{v_j \in V} \frac{P(a_1 \wedge a_2 \ldots \wedge a_n \mid v_j) P(v_j)}{P(a_1 \wedge a_2 \ldots \wedge a_n)}$$

$$= \arg\max_{v_j \in V} P(a_1 \wedge a_2 \ldots \wedge a_n \mid v_j) P(v_j)$$

Naïve Bayes Continued

$$v_{MAP} = \arg\max_{v_j \in V} P(a_1 \wedge a_2 ... \wedge a_n \mid v_j) P(v_j)$$

independence step

$$v_{NB} = \arg\max_{v_j \in V} P(a_1 \mid v_j) P(a_2 \mid v_j) ..... P(a_n \mid v_j) P(v_j)$$

$$= \arg\max_{v_j \in V} P(v_j) \prod_i P(a_i \mid v_j)$$

Instead of one table of size 2^{50000} we have 50,000 tables of size 2

Bayesian Belief Networks

- ***Bayes Optimal Classifier***
  - Often too costly to apply (uses full joint probability)
- ***Naïve Bayes Classifier***
  - Assumes conditional independence to lower costs
  - This assumption often overly restrictive
- ***Bayesian belief networks***
  - provide an ***intermediate*** approach
  - allows conditional independence assumptions that apply to ***subsets*** of the variable.

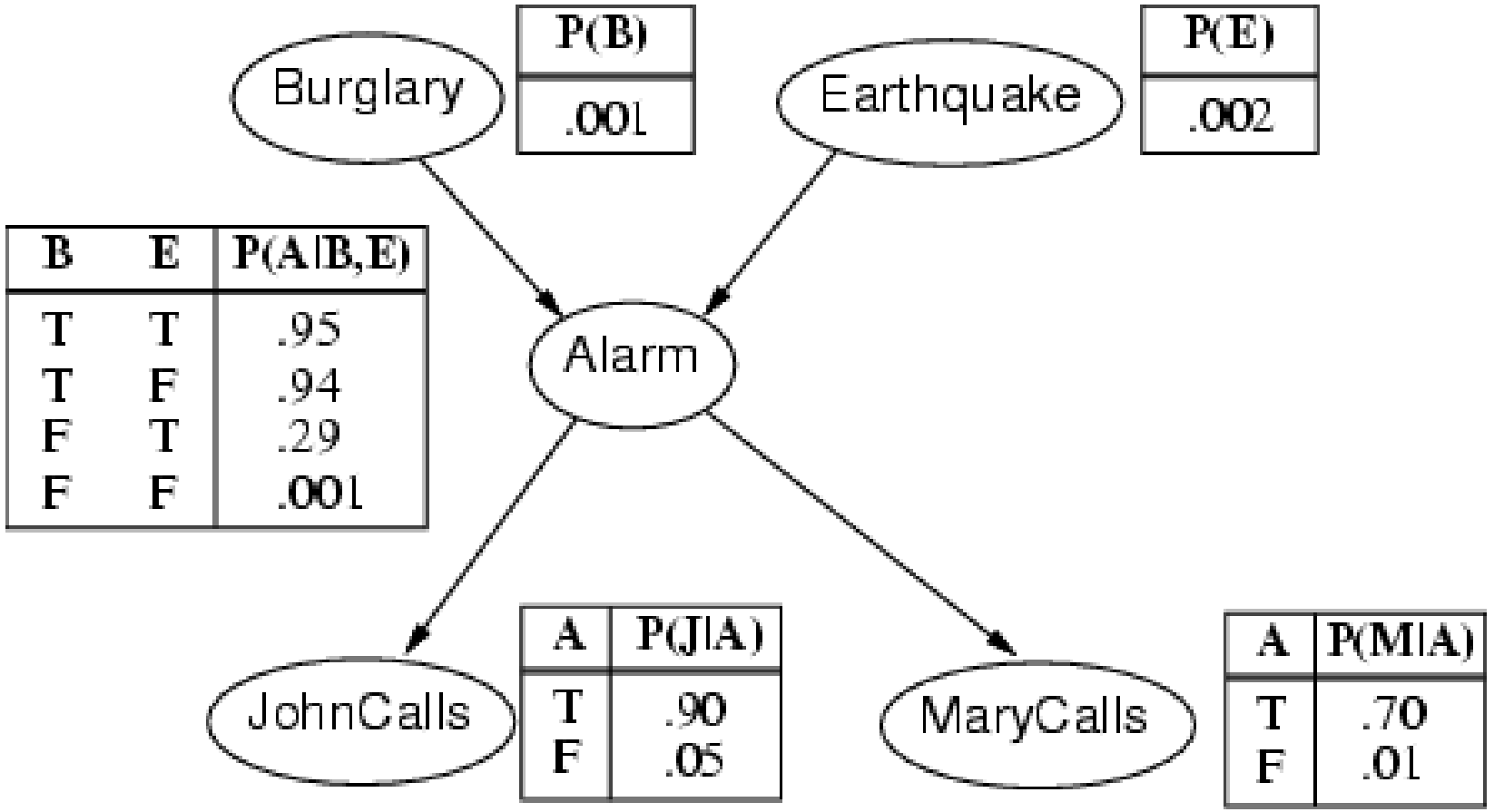
Example

- 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?

- Variables: *Burglary*, *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

Example from Russell & Norvig

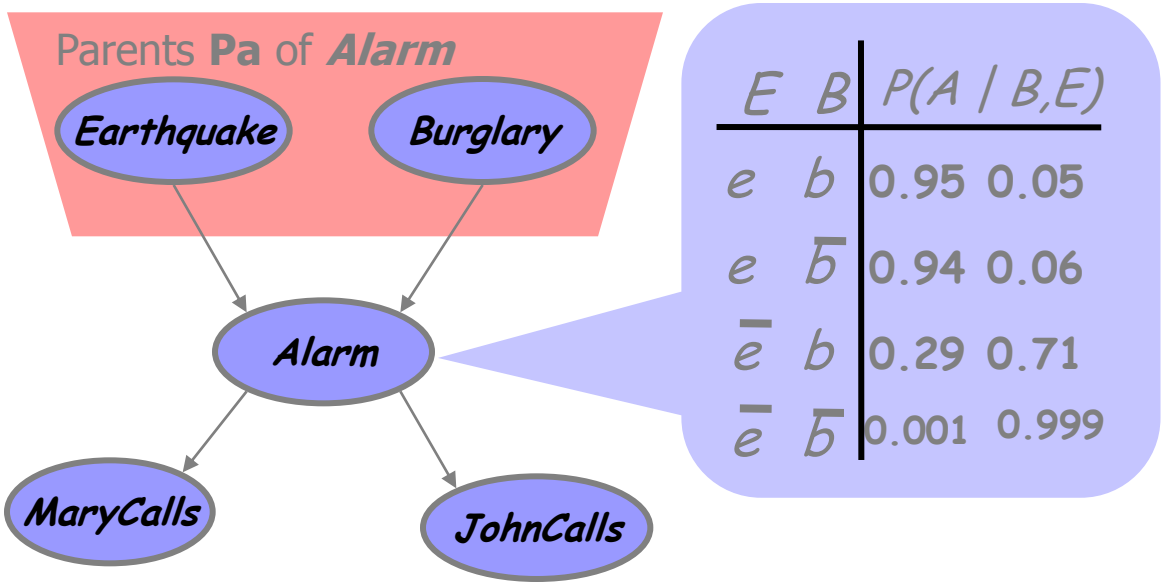


Bayesian Networks [Pearl 91]

$P(B,E,A,M,J)$

Qualitative part:
Directed acyclic graph (DAG)

- Nodes - random vars.
- Edges - direct influence



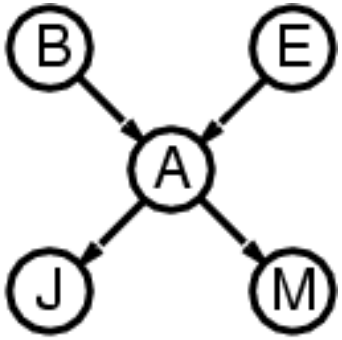
Together:
Define a unique distribution in a factored form

Quantitative part:
Set of conditional probability distributions

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

Compactness

- A CPT for Boolean X_i with k Boolean 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, 2 + 2 + 8+ 4 + 4 = 20 numbers (vs. $2^5-1 = 31$)

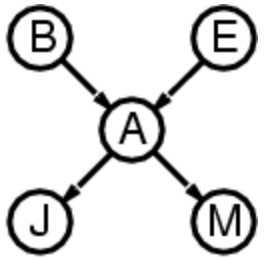


Semantics

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

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

Example:



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

= **P** (j | a) **P** (m | a) **P** (a | $\neg$b, $\neg$e) **P** ($\neg$b) **P** ($\neg$e)

Conditional Independence

- Recall two variables are independent iff...

$$P(A,B) = P(A)P(B)$$

- Two variables A and B are conditionally independent iff...

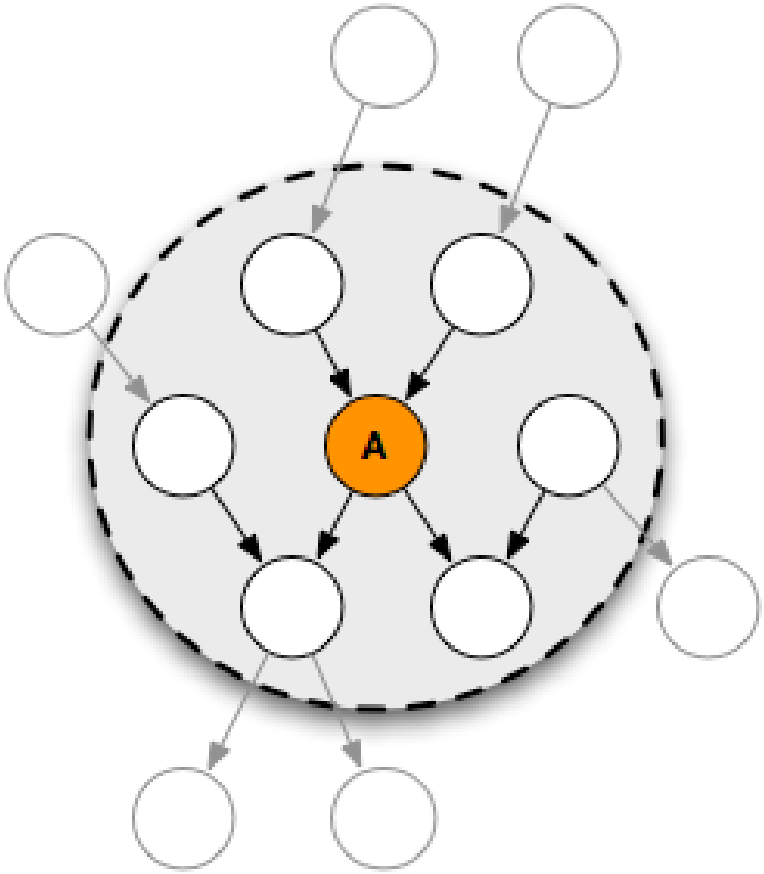
$$P(A,B|C) = P(A|C)P(B|C)$$

- This implies

$$P(A|B,C) = P(A|C)$$

Markov Blanket

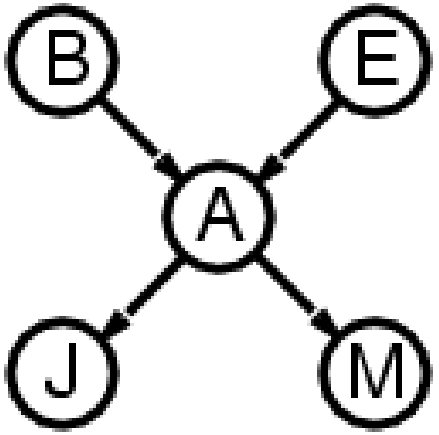
- In a Bayesian network, the **Markov blanket** for node *A* MB(A) is *A*'s parents, its children, and its children's other parents.
- Every set of nodes ***B*** in the network is conditionally independent of *A* when we know values for all nodes in *A*'s Markov blanket.



$$P(A \mid MB(A), B) = P(A \mid MB(A))$$

So what?

- So…this means we can direct our questions.
- To find out P(J) we don't need to know the values of B, E, M …as long as we know the value of A.
- What about P(A)?



Learning BB Networks: 3 cases

1. The network structure is given in advance and all the variables are fully observable in the training examples.

   Trivial Case: just estimate the conditional probabilities.

2. The network structure is given in advance but only some of the variables are observable in the training data.

   Similar to learning the weights for the hidden units of a Neural Net: Gradient Ascent Procedure

3. The network structure is not known in advance.

   Use a heuristic search or constraint-based technique to search through potential structures.

Constructing Bayesian networks

- 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

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

This choice of parents guarantees:

$$\mathbf{P}(X_1, \ldots, X_n) = \prod_{i=1}^{n} \mathbf{P}(X_i \mid X_1, \ldots, X_{i-1})$$

(chain rule)

$$= \prod_{i=1} \mathbf{P}(X_i \mid Parents(X_i))$$

(by construction)

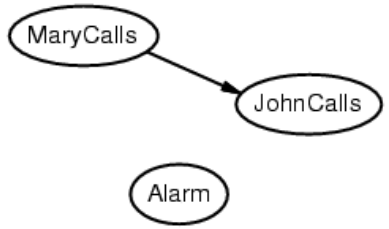
Example

- 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*
-



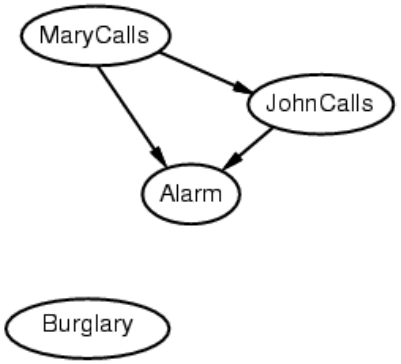
***P**(J | M) = **P**(J)?*

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

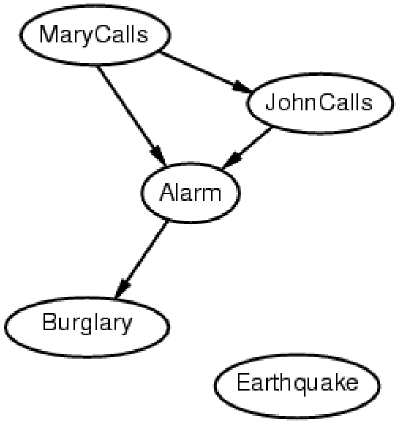
P*(A | J, M) =* ***P****(A | J)?* ***P****(A | J, M) =* ***P****(A)?* **No**

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**

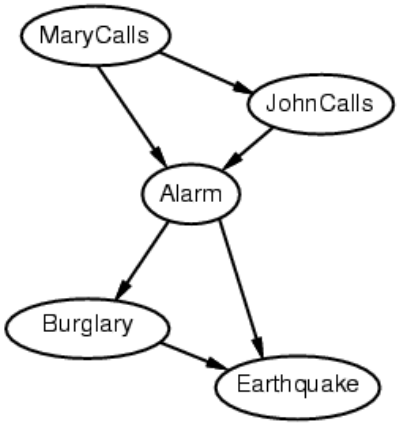
P(*B* | *A*, *J*, *M*) = ***P***(*B*)? **No**

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**

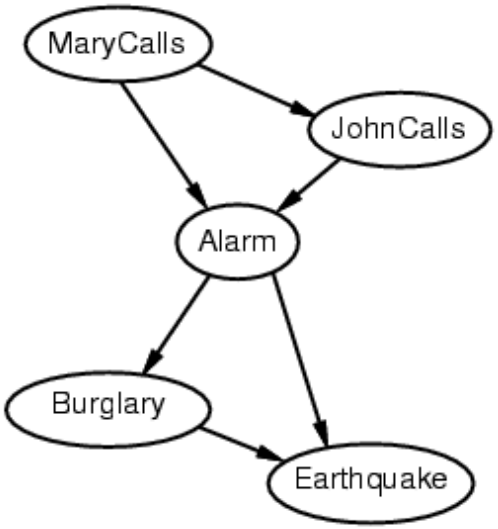
P*(B | A, J, M) =* ***P****(B)?* **No**

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 directions
-
  - Causal models and conditional independence seem hardwired for humans!
  -
- Network is less compact



Inference in BB Networks

- A Bayesian Network can be used to compute the probability distribution for any subset of network variables given the values or distributions for any subset of the remaining variables.
- Unfortunately, exact inference of probabilities in general for an arbitrary Bayesian Network is known to be NP-hard.
- In theory, approximate techniques (such as Monte Carlo Methods) can also be NP-hard, though in practice, many such methods are shown to be useful.

For more on Bayesian networks

- For more on Bayesian networks see the books in this Bayesian belief network…