Chapter 14 (AIAMA)

Probabilistic Reasoning

Sukarna Barua
Assistant Professor, CSE, BUET

Bayesian Network: Exact Inference

- **Consider the query**:
- We need to compute: [compute for both and]
- Express in terms of joint distributions [we already know how to compute joint distribution]

where is the normalizing constant [we can find the value of later]

Bayesian Network: Exact Inference

- How to compute
- Add all hidden variables to get full joint distribution
  - Query variable:
  - Evidence variables:
  - Hidden variables:

Bayesian Network: Exact Inference

- Finally, we get the following:
  - Compute and [expression has
  - Find [Using the eq.
- **Question**: How many operations (multiplications + additions)?
  - Answer: [For each] Mult: 16, Sum: 4, Total = 20 operations

Bayesian Network: Exact Inference

- Computing using join distribution naively:

- Required operations: 20
- **Can we improve?**
  - Yes, move sums inside [*closest to the factors having the hidden variable*]
  - Sums are evaluated early and gets multiplied once [*rather than every time during joint calculation*]

Bayesian Network: Exact Inference

- Move summations inside:
- Two alternate ways to do it:
  - Option 1:
  - Option 2:
  - Which one is more efficient in terms of operations?

Bayesian Network: Exact Inference

- **Which one is more efficient?** Assume and compute both.
- Equation 1:

  - Required # of multiplications: 9, # of sums = 3, total = 12

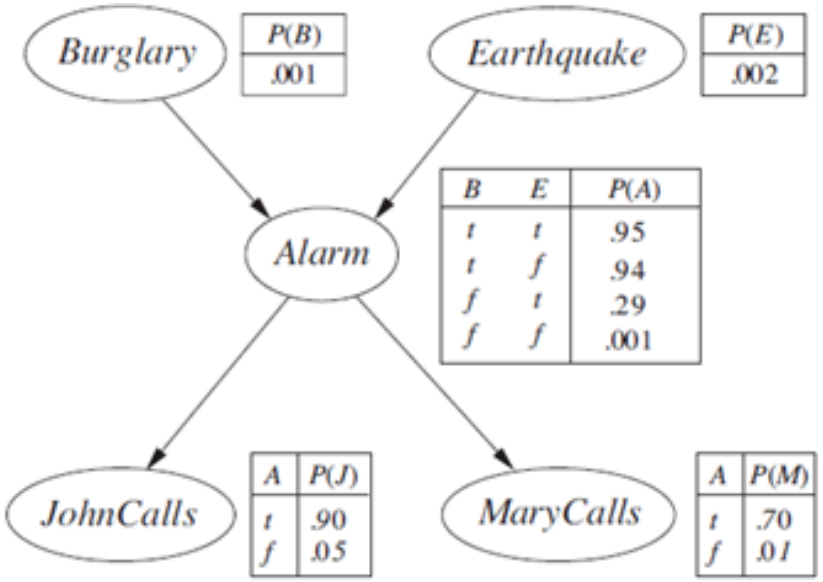
- Equation 2:

  - Required # of multiplications: 11, # of sums = 3, total = 14

- So order matters! However, finding optimal order is difficult!

Bayesian Network: Exact Inference

- Using the CPT data, we calculate:



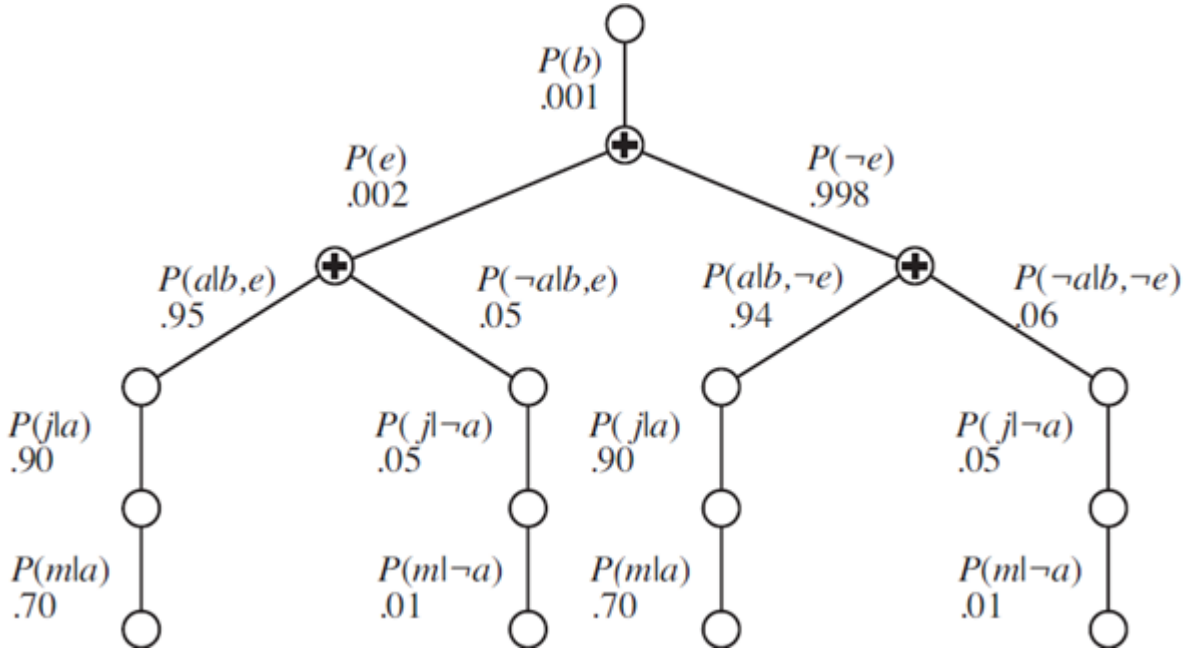
Hence,

,)

[After normalizing as probabilities sum to 1]

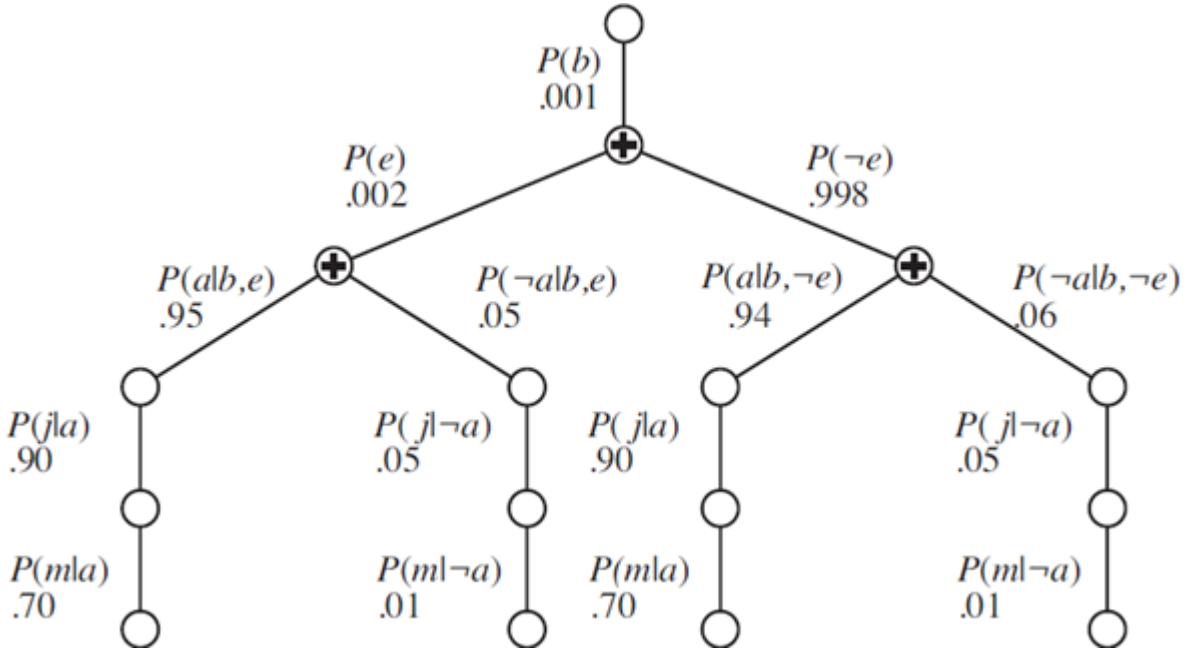
Bayesian Network: Exact Inference

- An evaluation tree is shown for the expression:
  - Blank circled nodes represent multiplication [*Notice the repetition of sub-paths*]



Bayesian Network: Exact Inference

- **Problem:** Repeated subexpression evaluation
- The following products are computed twice in two paths of the tree!



Bayesian Network: Variable Elimination Algorithm

- Steps of variable elimination algorithm:
  - Step 1: Compute factors for each node of Bayesian Network and instantiate/restrict evidence variables.
  - Step 2: For some order of variables ():
    - (2a) If variable is a hidden variable, then multiply all the factors of
    - (2b) Sum-out
  - Step 3: Multiply remaining factors.
  - Step 4: Normalize the final factor to eliminate .

Bayesian Network: Variable Elimination Algorithm

- Consider the following expression:
  - Query variable:
  - Evidence variable:
  - Hidden variable:

- We will use variable elimination algorithm to calculate

Bayesian Network: Variable Elimination Algorithm

- Query:
- Joint probability expression of BN:

- **Step 1**: Compute all the factors of BN.

- The BN has five factors [*note the joint probability expression*]:

Variable Elimination Algorithm: Compute Factors

- Query:
- **Step 1a**: Compute all the factors of BN. [*don't restrict evidences now*]

$f_1(B) = P(B)$

T	0.001
F	0.999

$f_2(E) = P(E)$

T	0.002
F	0.998

$f_3(A, B, E) = P(A|B, E)$

T	T	T	0.95
T	T	F	0.94
T	F	T	0.29
T	F	F	0.001
F	T	T	0.05
F	T	F	0.06
F	F	T	0.71
F	F	F	0.999

$f_4(J, A) = P(J|A)$

T	T	0.90
T	F	0.05
F	T	0.10
F	F	0.95

$f_5(M, A) = P(M|A)$

T	T	0.70
T	F	0.01
F	T	0.30
F	F	0.99

Variable Elimination Algorithm: Restrict Factors

- Query:
- **Step 1b**: Instantiate/restrict evidence variables in all factors [*and* .
  - Restrict *to* true *in* [*Keep only rows where*
  - Note that is removed from factor expression.

$f_4(J, A) = P(J|A)$

T	T	0.90
T	F	0.05
F	T	0.10
F	F	0.95

$f_4(A)$

T	0.90
F	0.05

Variable Elimination Algorithm: Restrict Factors

- Query:
- **Step 1b**: Instantiate/restrict evidence variables in all factors. [*and* .
  - Restrict *M to* true *in* [*Keep only rows where*

$f_5(M, A) = P(M|A)$

T	T	0.70
T	F	0.01
F	T	0.30
F	F	0.99

$f_5(A)$

T	0.70
F	0.01

Variable Elimination Algorithm: Multiply Factors

- Query:
- **Step 2**: Select every variable (in any order) which is hidden [Here, and
  - Multiply all factors of and then sum-out the variable.
  - Suppose we select in this order: E, A.
    - First we select E
    - Multiple all factors having E and get a new factor:
    - Sum-out and get a new factor:
    - Note that after summing-out, is removed from factor-argument.

Bayesian Network: Multiply Factors

- **Step 2a**: Multiple all factors having E and get a new factor:

- Factor multiplication is a point-wise multiplication of elements in two factors:
  - Multiply rows which have matching values of the variables.
  - Number of rows in the new factor where is number of total variables in the new factor:
    - Multiplying and will give us which will have 8 rows.

Bayesian Network: Multiply Factors

- For variable
  - **Step 2a**: Multiple all factors having E and get a new factor:
  - **Multiply**:

$f_2(E) = P(E)$

T	0.002
F	0.998

×

$f_3(A, B, E) = P(A|B, E)$

T	T	T	0.95
T	T	F	0.94
T	F	T	0.29
T	F	F	0.001
F	T	T	0.05
F	T	F	0.06
F	F	T	0.71
F	F	F	0.999

⇨

$f_6(A, B, E)$

T	T	T	0.0019
T	T	F	0.93812
T	F	T	0.00058
T	F	F	0.000998
F	T	T	0.0001
F	T	F	0.05988
F	F	T	0.00142
F	F	F	0.997002

Bayesian Network: Multiply Factors

- For variable E:
  - **Step 2b**: Sum-out resulting factor for E and get a new factor:
  - **Sum-out** : [*sum corresponding rows that have and for E*]

$f_6(A, B, E)$

T	T	T	0.0019
T	T	F	0.93812
T	F	T	0.00058
T	F	F	0.000998
F	T	T	0.0001
F	T	F	0.05988
F	F	T	0.00142
F	F	F	0.997002

$f_7(A, B)$

T	T	0.94002
T	F	0.001578
F	T	0.05998
F	F	0.998422

Bayesian Network: Multiply Factors

- For variable A
  - **Step 2a**: Multiple all factors having A and get a new factor:
  - **Multiply**:

$f_4(A)$

T	0.90
F	0.05

×

$f_5(A)$

T	0.70
F	0.01

⇨

$f_4(A) \times f_5(A)$

T	0.63
F	0.0005

×

$f_7(A, B)$

T	T	0.94002
T	F	0.001578
F	T	0.05998
F	F	0.998422

⇨

$f_8(A, B)$

T	T	0.592213
T	F	0.00099414
F	T	0.00002999
F	F	0.000499211

Bayesian Network: Multiply Factors

- For variable A:
  - **Step 2b**: Sum-out resulting factor for E and get a new factor:
  - **Sum-out** : [*sum corresponding rows that have and for E*]

$f_8(A, B)$

T	T	0.592213
T	F	0.00099414
F	T	0.00002999
F	F	0.000499211

$f_9(B)$

T	0.59224299
F	0.001493351

Bayesian Network: Multiply Remaining Factors

- Query:
- **Step 3:** Multiply remaining factors:

$f_1(B) = P(B)$

T	0.001
F	0.999

×

$f_9(B)$

T	0.59224299
F	0.001493351

⇨

$f_{10}(B)$

T	0.000592
F	0.001492

Bayesian Network: Normalize Result

- Query:
- **Step 4**: Normalize the final factor to eliminate .

$f_{10}(B)$

T	0.000592
F	0.001492

⇨

$f_{11}(B)$

T	0.284172
F	0.715828

- **Answer**: , 16

Bayesian Network: Variable Elimination Algorithm

- Algorithm pseudocode [RN book]
  - Note: Step 1 (initialization) is missing in the pseudocode [it is implied in the MAKE-FACTOR function which is not a good approach!]

function ELIMINATION-ASK(X, **e**, bn) **returns** a distribution over X
  inputs: X, the query variable
    e, observed values for variables **E**
    bn, a Bayesian network specifying joint distribution $\mathbf{P}(X_1, \ldots, X_n)$

  $factors \leftarrow [\,]$
  for each var **in** ORDER(bn.VARS) **do**
    $factors \leftarrow [\text{MAKE-FACTOR}(var, \mathbf{e}) | factors]$
    if var is a hidden variable **then** $factors \leftarrow$ SUM-OUT($var, factors$)
  return NORMALIZE(POINTWISE-PRODUCT($factors$))

Bayesian Network: Irrelevant Variable

- Consider the query:

- Incorporating all hidden variables of the Bayesian Network, we get:

Bayesian Network: Irrelevant Variable

- A simple rearrangement of the expression would yield us:

$$\mathbf{P}(J \mid b) = \alpha\, P(b) \sum_{e} P(e) \sum_{a} P(a \mid b, e) \mathbf{P}(J \mid a) \sum_{m} P(m \mid a)\ .$$

- Note that the inner term will evaluate to 1. So it is eliminated from the expression entirely [Irrelevant variable]

- *Variable elimination algorithm will automatically sum-out from the expression!*
  - Why? [At some point, will be considered as a hidden variable and the summation over will eliminate it]

Bayesian Network: Irrelevant Variable

- In general, we can remove any leaf node that is not a query variable or an evidence variable.
- After removal of initial leaves, there may be some more leaf nodes, and these too may be irrelevant and removed [continue this process].
- *Every variable that is not an ancestor of a query variable or evidence variable is irrelevant to the query.*