Bayesian Networks – Representation

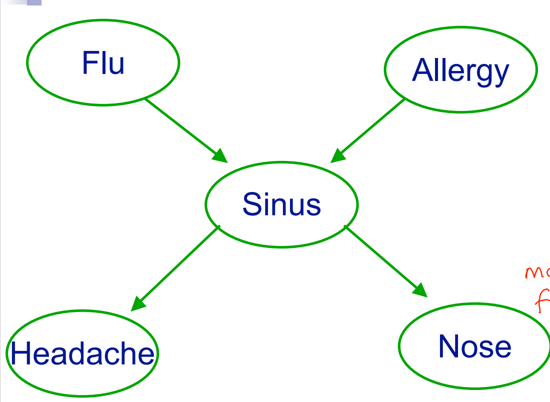
Machine Learning – 10701/15781

Carlos Guestrin

Carnegie Mellon University

October 31st, 2007

Possible queries



- Inference

  $P(F=t \mid H=t, N=f)$

- Most probable explanation

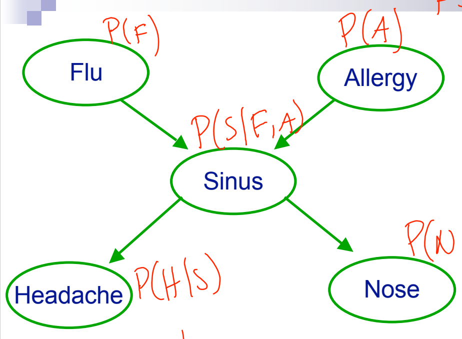
  $\max_{f,a,s} P(f,a,s \mid H=t, N=f)$

- Active data collection

  what should I measure

Factored joint distribution - Preview

Notation: F, A → I am not specify an assignment
f, a → specific assignments
F=t → Flu = true (a particular assignment)



P(F, A, S, H, N)

$2^5 - 1$ (because sums to 1)

r = 32 − 1 = 31

P(F, A, S, H, N) = P(F) · P(A) · P(S|F,A) P(H|S) · P(N|S)

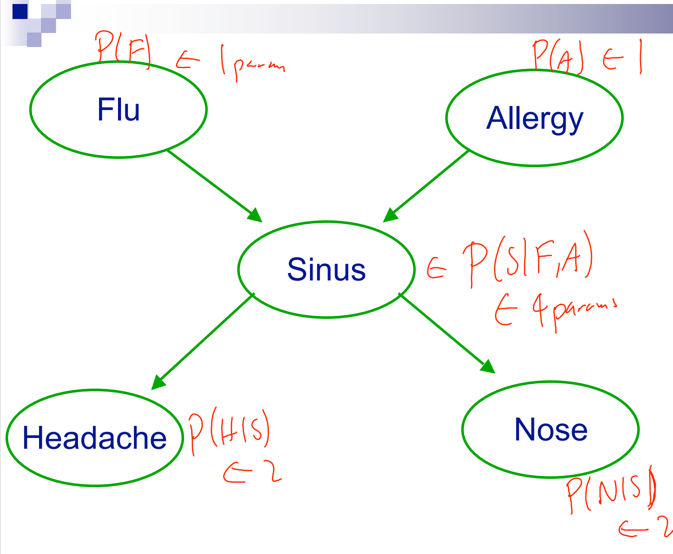
P(F) =

F	
t	0.1
f	0.9

P(H|S): 2 numbers

H \ S	t	f
t	0.8	0.3
f	1−0.8 =0.2	0.7

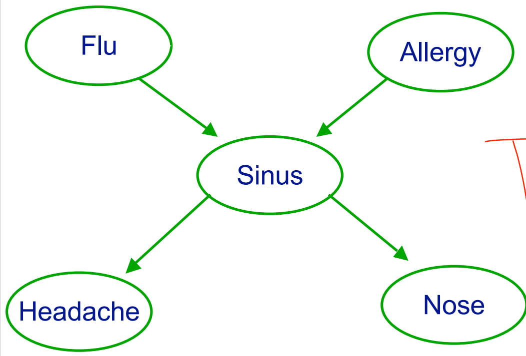
Number of parameters



total: 10

Key: Independence assumptions

The bias

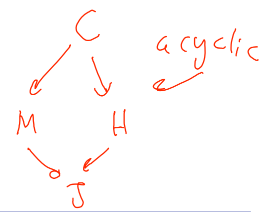


independent | is it true

$A \perp H \mid S$ | $F \perp A \mid S$

$H \perp N \mid S$ | NO! explaining away

$F \perp A$



Knowing sinus separates the variables from each other

A BN is always acyclic | cyclic: , not cycle:

(Marginal) Independence

- Flu and Allergy are (marginally) independent

$F \perp A$

$\Updownarrow$

$P(F,A) = P(F) \cdot P(A)$

- More Generally:

Flu = t	0.2
Flu = f	0.8

Allergy = t	0.3
Allergy = f	0.7

	Flu = t	Flu = f
Allergy = t	0.3×0.2	0.3×0.8
Allergy = f	0.2×0.7	0.7×0.8

Marginally independent random variables

- **Sets** of variables **X**, **Y**
- X is independent of Y if $\forall x \in Val(X), y \in Val(Y)$
  - ~~P²~~(**X=x**⊥**Y=y**), ~~8 **x2**Val(**X**), **y2**Val(**Y**)~~
  - $P(X=x, Y=y) = P(X=x) \cdot P(Y=y)$
- Shorthand: $P(X=x \mid Y=y) = P(X=x)$
  - **Marginal independence:** ~~P²~~ (**X** ⊥ **Y**)
- **Proposition:** *P* satisfies (**X** ⊥ **Y**) if and only if
  - P(**X**,**Y**) = P(**X**) P(**Y**)
  - $P(X \mid Y) = P(X)$

Conditional independence

- Flu and Headache are not (marginally) independent
- Flu and Headache are independent given Sinus infection
- More Generally:

Conditionally independent random variables

- **Sets** of variables **X**, **Y**, **Z**
- X is independent of Y given Z if
  - *P* ²(**X**=**x** ⊥ **Y**=**y**|**Z**=**z**), 8 **x2**Val(**X**), **y2**Val(**Y**), **z2**Val(**Z**)

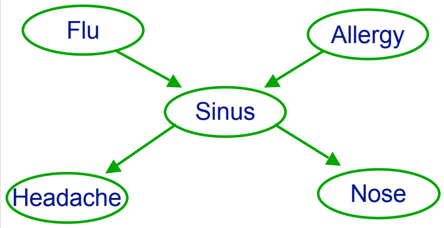
- Shorthand:
  - **Conditional independence:** *P* ² (**X** ⊥ **Y** | **Z**)
  - For *P* ² (**X** ⊥ **Y** | ;), write P ² (**X** ⊥ **Y**)

- **Proposition:** *P* statisfies (**X** ⊥ **Y** | **Z**) if and only if
  - P(**X**,**Y**|**Z**) = P(**X**|**Z**) P(**Y**|**Z**)

Properties of independence

- **Symmetry:**
  - (**X** ⊥ **Y** | **Z**) ⇒ (**Y** ⊥ **X** | **Z**)
- **Decomposition:**
  - (**X** ⊥ **Y**,**W** | **Z**) ⇒ (**X** ⊥ **Y** | **Z**)
- **Weak union:**
  - (**X** ⊥ **Y**,**W** | **Z**) ⇒ (**X** ⊥ **Y** | **Z**,**W**)
- **Contraction:**
  - (**X** ⊥ **W** | **Y**,**Z**) & (**X** ⊥ **Y** | **Z**) ⇒ (**X** ⊥ **Y**,**W** | **Z**)
- **Intersection:**
  - (**X** ⊥ **Y** | **W**,**Z**) & (**X** ⊥ **W** | **Y**,**Z**) ⇒ (**X** ⊥ **Y**,**W** | **Z**)
  - Only for positive distributions!
  - P(α)>0, 8α, α≠;

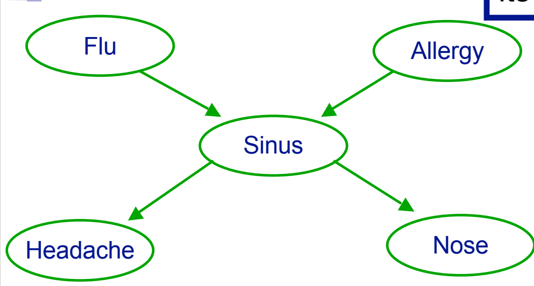
The independence assumption



Local Markov Assumption:
A variable X is independent of its non-descendants given its parents

Explaining away

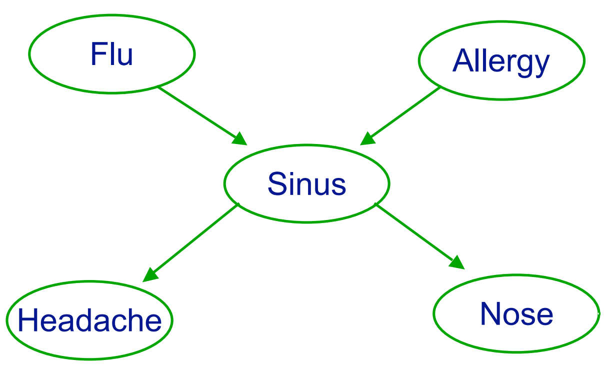
Local Markov Assumption:
A variable X is independent of its non-descendants given its parents



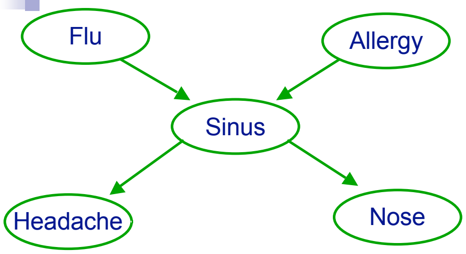
Naïve Bayes revisited

Local Markov Assumption:
A variable X is independent of its non-descendants given its parents

What about probabilities? Conditional probability tables (CPTs)



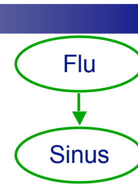
Joint distribution



Why can we decompose? Markov Assumption!

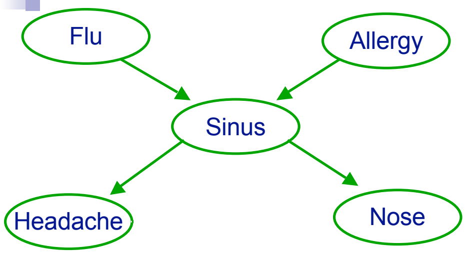
The chain rule of probabilities

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



- More generally:
  - $P(X_1,\ldots,X_n) = P(X_1)$ ¢ $P(X_2|X_1)$ ¢ … ¢ $P(X_n|X_1,\ldots,X_{n-1})$

Chain rule & Joint distribution



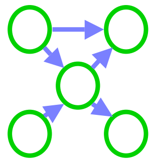
Local Markov Assumption:
A variable X is independent of its non-descendants given its parents

Two (trivial) special cases

Edgeless graph

Fully-connected graph

The Representation Theorem – Joint Distribution to BN

BN: 

Encodes independence assumptions

If conditional independencies in BN are subset of conditional independencies in *P*

Obtain

Joint probability distribution:

$$P(X_1, \ldots, X_n) = \prod_{i=1}^{n} P\left(X_i \mid \mathbf{Pa}_{X_i}\right)$$

A general Bayes net

- Set of random variables
- Directed acyclic graph
  - Encodes independence assumptions
- CPTs
- Joint distribution:

$$P(X_1, \ldots, X_n) = \prod_{i=1}^{n} P\left(X_i \mid \mathbf{Pa}_{X_i}\right)$$

How many parameters in a BN?

- Discrete variables $X_1, \ldots, X_n$
- Graph
  - Defines parents of X_i, $\mathbf{Pa}_{X_i}$
- CPTs – $P(X_i | \mathbf{Pa}_{X_i})$

Real Bayesian networks applications

- Diagnosis of lymph node disease
- Speech recognition
- Microsoft office and Windows
  - http://www.research.microsoft.com/research/dtg/
- Study Human genome
- Robot mapping
- Robots to identify meteorites to study
- Modeling fMRI data
- Anomaly detection
- Fault dianosis
- Modeling sensor network data

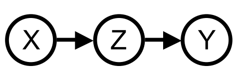
Independencies encoded in BN

- We said: All you need is the local Markov assumption
  - $(X_i \perp NonDescendants_{X_i} \mid \mathbf{Pa}_{X_i})$
- But then we talked about other (in)dependencies
  - e.g., explaining away
- What are the independencies encoded by a BN?
  - Only assumption is local Markov
  - But many others can be derived using the algebra of conditional independencies!!!

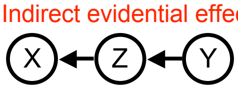
Understanding independencies in BNs – BNs with 3 nodes

Local Markov Assumption:
A variable X is independent of its non-descendants given its parents

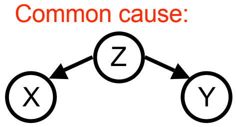
Indirect causal effect:



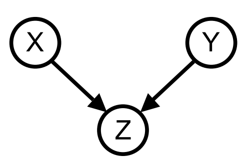
Indirect evidential effect:



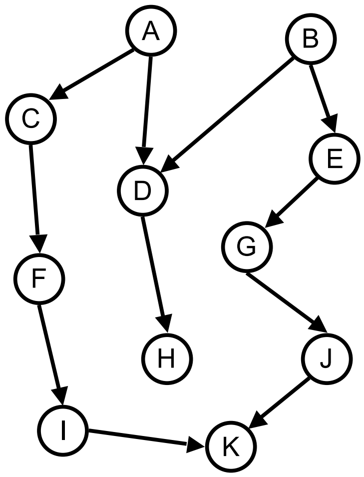
Common cause:



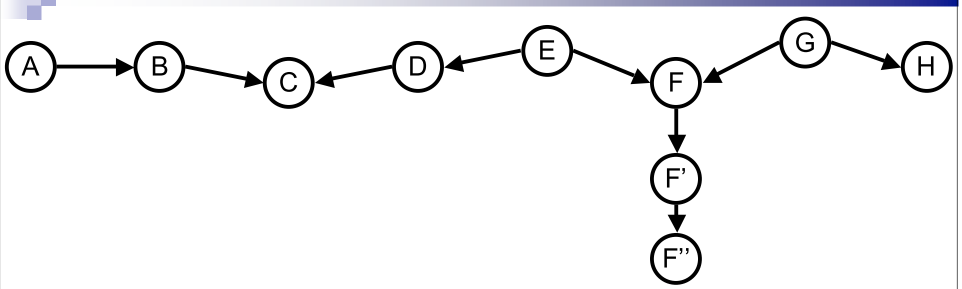
Common effect:



Understanding independencies in BNs – Some examples



An active trail – Example



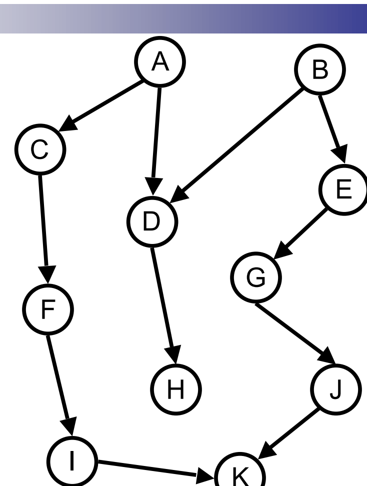
When are A and H independent?

Active trails formalized

- A path $X_1 - X_2 - \cdots - X_k$ is an **active trail** when variables $\boldsymbol{O} \subseteq \{X_1,\ldots,X_n\}$ are observed if for each consecutive triplet in the trail:
  - $X_{i-1} \rightarrow X_i \rightarrow X_{i+1}$, and X_i is **not observed** ($X_i \notin \boldsymbol{O}$)
  - $X_{i-1} \leftarrow X_i \leftarrow X_{i+1}$, and X_i is **not observed** ($X_i \notin \boldsymbol{O}$)
  - $X_{i-1} \leftarrow X_i \rightarrow X_{i+1}$, and X_i is **not observed** ($X_i \notin \boldsymbol{O}$)
  - $X_{i-1} \rightarrow X_i \leftarrow X_{i+1}$, and X_i **is observed** ($X_i \in \boldsymbol{O}$), or **one of its descendents**

Active trails and independence?

- **Theorem**: Variables $\mathbf{X_i}$ **and** $\mathbf{X_j}$ **are independent given** $\boldsymbol{Z} \subseteq \{X_1,\ldots,X_n\}$ if the is **no active trail** between X_i and X_j when variables $\boldsymbol{Z} \subseteq \{X_1,\ldots,X_n\}$ are observed



The BN Representation Theorem

If conditional independencies in BN are subset of conditional independencies in *P* → Obtain → **Joint probability distribution:**

$$P(X_1,\ldots,X_n) = \prod_{i=1}^{n} P\left(X_i \mid \mathbf{Pa}_{X_i}\right)$$

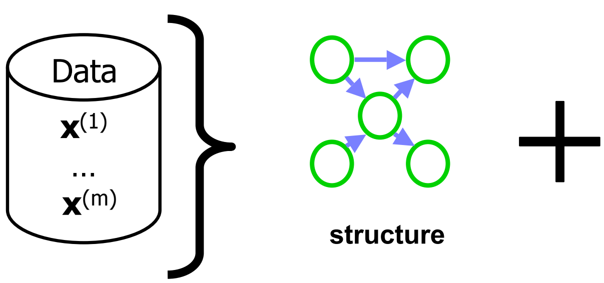
Important because:
Every *P* has at least one BN structure *G*

If joint probability distribution: $P(X_1,\ldots,X_n) = \prod_{i=1}^{n} P\left(X_i \mid \mathbf{Pa}_{X_i}\right)$ → Obtain → **Then conditional independencies in BN are subset of conditional independencies in *P***

Important because:
Read independencies of *P* from BN structure *G*

Learning Bayes nets

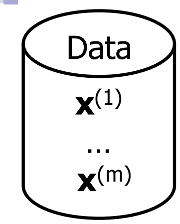
	Known structure	Unknown structure
Fully observable data		
Missing data		



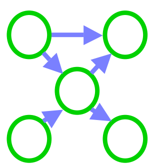
CPTs – $P(X_i \mid \mathbf{Pa}_{X_i})$

parameters

Learning the CPTs



For each discrete variable X_i



MLE: $P(X_i = x_i \mid X_j = x_j) = \frac{\text{Count}(X_i = x_i, X_j = x_j)}{\text{Count}(X_j = x_j)}$

What you need to know

- Bayesian networks
  - A compact **representation** for large probability distributions
  - Not an algorithm
- Semantics of a BN
  - Conditional independence assumptions
- Representation
  - Variables
  - Graph
  - CPTs
- Why BNs are useful
- Learning CPTs from fully observable data
- Play with applet!!! ☺

Announcements

- Recitation this week
  - Bayesian networks

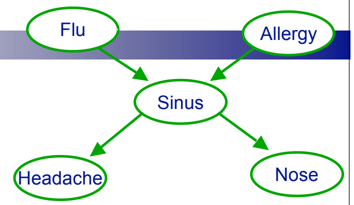
- Pick up your midterm from Monica

Bayesian Networks – Inference

Machine Learning – 10701/15781
Carlos Guestrin
Carnegie Mellon University

October 31st, 2007

General probabilistic inference



- Query: $P(X \mid e)$

- Using Bayes rule:

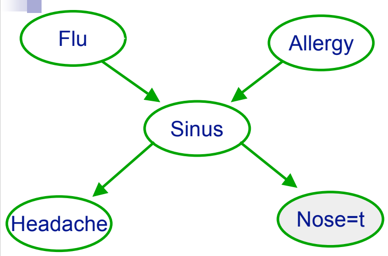
$$P(X \mid e) = \frac{P(X, e)}{P(e)}$$

- Normalization:

$$P(X \mid e) \propto P(X, e)$$

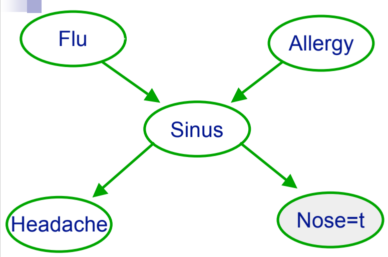
Marginalization

Probabilistic inference example



Inference seems exponential in number of variables!
Actually, inference in graphical models is NP-hard ☹

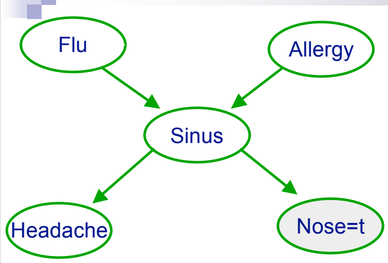
Fast probabilistic inference example – Variable elimination



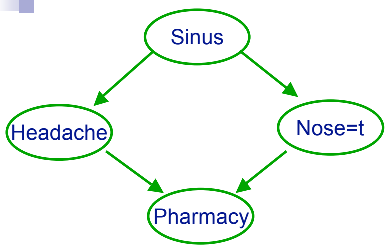
(Potential for) Exponential reduction in computation!

Understanding variable elimination – Exploiting distributivity

Understanding variable elimination – Order can make a HUGE difference



Understanding variable elimination – Another example



Variable elimination algorithm

- Given a BN and a query $P(X|e) \propto P(X,e)$
- Instantiate evidence e
- Choose an ordering on variables, e.g., $X_1, \ldots, X_n$

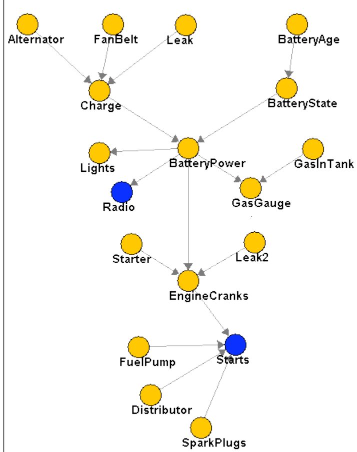
IMPORTANT!!!

- For i = 1 to n, If $X_i \notin \{X,e\}$
  - Collect factors $f_1,\ldots,f_k$ that include X_i
  - Generate a new factor by eliminating X_i from these factors

$$g = \sum_{X_i} \prod_{j=1}^{k} f_j$$

  - Variable X_i has been eliminated!
- Normalize $P(X,e)$ to obtain $P(X|e)$

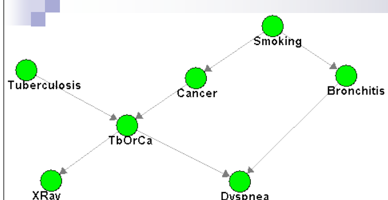
Complexity of variable elimination – (Poly)-tree graphs



Variable elimination order:
Start from "leaves" up –
find topological order, eliminate variables in reverse order

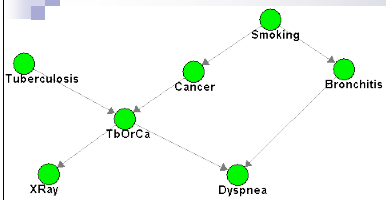
Linear in number of variables!!! (versus exponential)

Complexity of variable elimination – Graphs with loops



Exponential in number of variables in largest factor generated

Complexity of variable elimination –Tree-width



Moralize graph:
Connect parents into a clique and remove edge directions

Complexity of VE elimination:
("Only") exponential in tree-width
Tree-width is maximum node cut +1

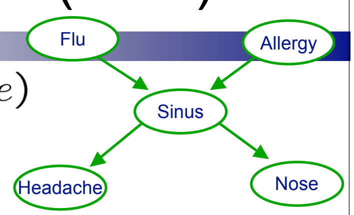
Example: Large tree-width with small number of parents

Compact representation ⇏ Easy inference ☹

Choosing an elimination order

- Choosing best order is NP-complete
  - Reduction from MAX-Clique
- Many good heuristics (some with guarantees)
- Ultimately, can't beat NP-hardness of inference
  - Even optimal order can lead to exponential variable elimination computation
- In practice
  - Variable elimination often very effective
  - Many (many many) approximate inference approaches available when variable elimination too expensive

Most likely explanation (MLE)



- Query: $\underset{x_1,\ldots,x_n}{\text{argmax}}\, P(x_1,\ldots,x_n \mid e)$

- Using Bayes rule:

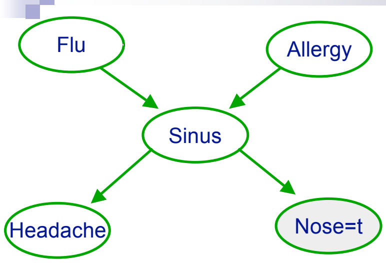
$$\underset{x_1,\ldots,x_n}{\text{argmax}}\, P(x_1,\ldots,x_n \mid e) = \underset{x_1,\ldots,x_n}{\text{argmax}}\, \frac{P(x_1,\ldots,x_n,e)}{P(e)}$$

- Normalization irrelevant:

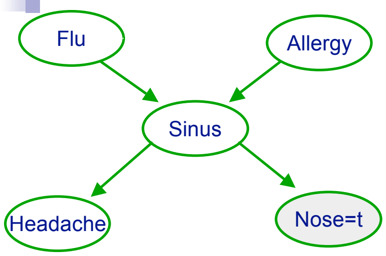
$$\underset{x_1,\ldots,x_n}{\text{argmax}}\, P(x_1,\ldots,x_n \mid e) = \underset{x_1,\ldots,x_n}{\text{argmax}}\, P(x_1,\ldots,x_n,e)$$

Max-marginalization

Example of variable elimination for MLE – Forward pass



Example of variable elimination for MLE – Backward pass



MLE Variable elimination algorithm – Forward pass

- Given a BN and a MLE query $\max_{x_1,\ldots,x_n} P(x_1,\ldots,x_n,e)$
- Instantiate evidence e
- Choose an ordering on variables, e.g., X_1, …, X_n
- For i = 1 to n, If $X_i \notin \{e\}$
  - Collect factors $f_1,\ldots,f_k$ that include X_i
  - Generate a new factor by eliminating X_i from these factors

$$g = \max_{x_i} \prod_{j=1}^{k} f_j$$

  - Variable X_i has been eliminated!

MLE Variable elimination algorithm – Backward pass

- $\{x_1^*, \ldots, x_n^*\}$ will store maximizing assignment
- For i = n to 1, If $X_i \notin \{e\}$
  - Take factors $f_1, \ldots, f_k$ used when X_i was eliminated
  - Instantiate $f_1, \ldots, f_k$, with $\{x_{i+1}^*, \ldots, x_n^*\}$
    - Now each f_j depends only on X_i
  - Generate maximizing assignment for X_i:

$$x_i^* \in \underset{x_i}{\text{argmax}} \prod_{j=1}^{k} f_j$$

What you need to know

- Bayesian networks
  - A useful compact **representation** for large probability distributions
- Inference to compute
  - Probability of X given evidence e
  - Most likely explanation (MLE) given evidence e
  - Inference is NP-hard
- Variable elimination algorithm
  - Efficient algorithm ("only" exponential in tree-width, not number of variables)
  - Elimination order is important!
  - Approximate inference necessary when tree-width to large
    - not covered this semester
  - Only difference between probabilistic inference and MLE is "sum" versus "max"