ECE 6504: Advanced Topics in Machine Learning

Probabilistic Graphical Models and Large-Scale Learning

Topics

- Bayes Nets: Inference
  - Marginals, MPE, MAP
  - Variable Elimination

Readings: KF 9.1,9.2; Barber 5.1

Dhruv Batra

Virginia Tech

Administrativia

- HW1
  - Out
  - Due in 2 weeks: ~~Feb 17~~, Feb 19, 11:59pm
  - Please please please please start early
  - Implementation: TAN, structure + parameter learning
  - Please post questions on Scholar Forum.

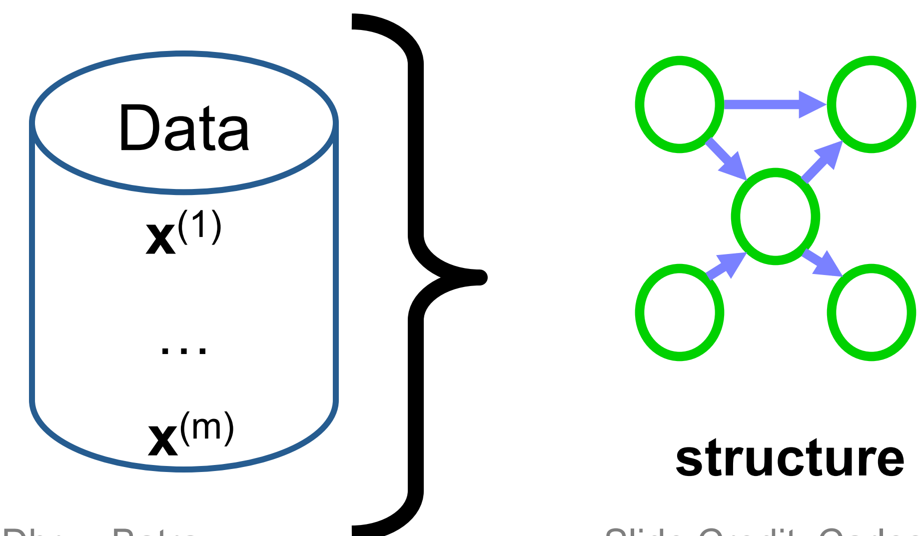
- HW2
  - Out soon
  - Due in 2 weeks: Mar 5, 11:59pm

- Project Proposal
  - Due: Mar 12, 11:59pm
  - <=2pages, NIPS format

Recap of Last Time

Learning Bayes nets

	Known structure	Unknown structure
Fully observable data	Very easy	Hard
Missing data	Somewhat easy (EM)	Very very hard



structure

+

CPTs – $P(X_i | \mathbf{Pa}_{X_i})$

parameters

Main Issues in PGMs

- Representation
  - How do we store $P(X_1, X_2, \ldots, X_n)$
  - What does my model mean/imply/assume? (Semantics)

- Learning
  - How do we learn parameters and structure of $P(X_1, X_2, \ldots, X_n)$ from data?
  - What model is the right for my data?

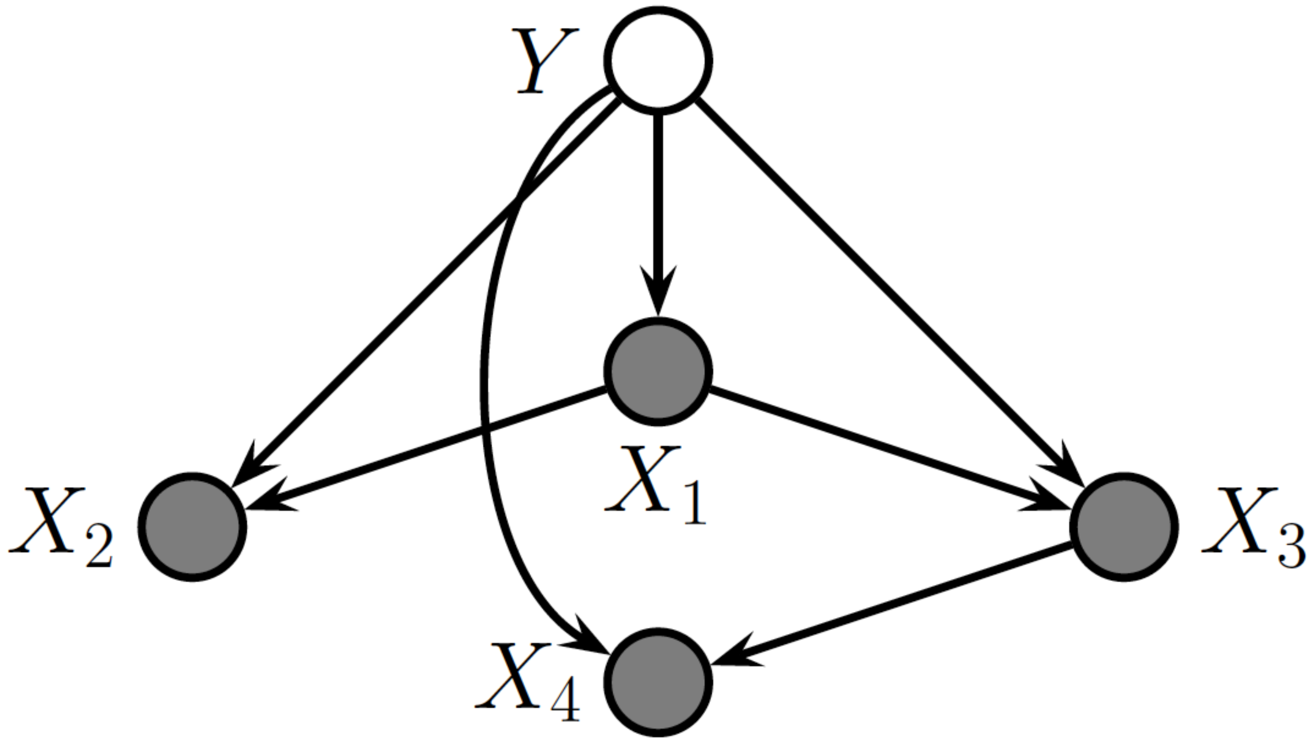
- Inference
  - How do I answer questions/queries with my model? such as
  - Marginal Estimation: $P(X_5 \mid X_1, X_4)$
  - Most Probable Explanation: $\operatorname{argmax} P(X_1, X_2, \ldots, X_n)$

Plan for today

- BN Inference
  - Queries: Marginals, Conditional Probabilities, MAP, MPE
  - Variable Elimination

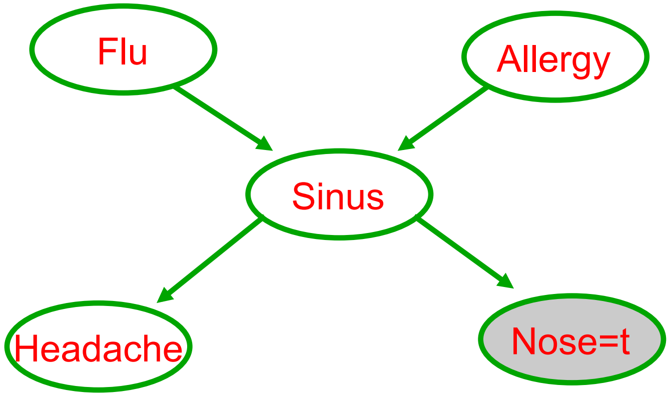
Example

- HW1 Inference:



Tree-Augmented Naïve Bayes (TAN)

Possible Queries



- Evidence: **E**=**e** (e.g. N=t)
- Query variables of interest **Y**

- Conditional Probability: P(**Y** | **E**=**e**)
  - E.g. P(F,A | N=t)
  - Special case: Marginals P(F)

- Maximum a Posteriori: argmax P(All variables | **E**=**e**)
  - argmax_{f,a,s,h} P(f,a,s,h | N = t)

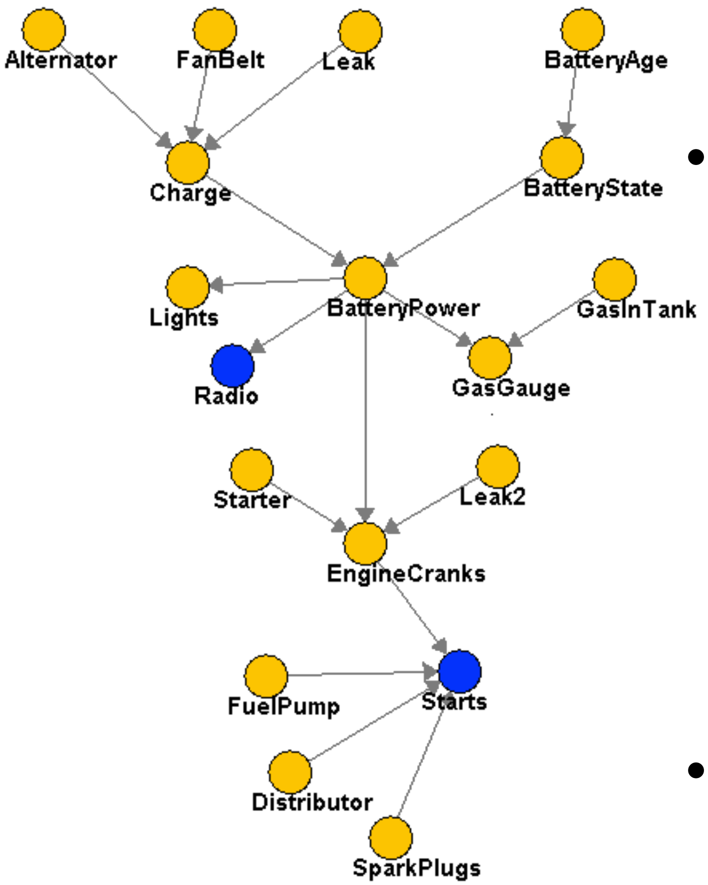
Old-school terminology: MPE

- Marginal-MAP: argmax_y P(**Y** | **E**=**e**)

Old-school terminology: MAP

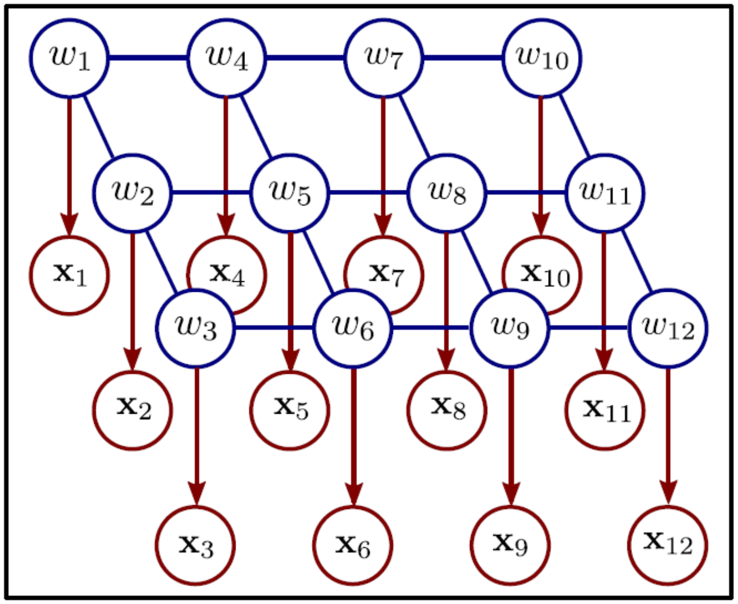
  - = argmax_{y} Σ_o P(**Y**=**y**, **O**=**o** | **E**=**e**)

Car starts BN



- 18 binary attributes
- Inference
  - P(BatteryAge|Starts=f)
- 2^{18} terms, why so fast?

Application: Computer Vision



Grid model

Markov random field

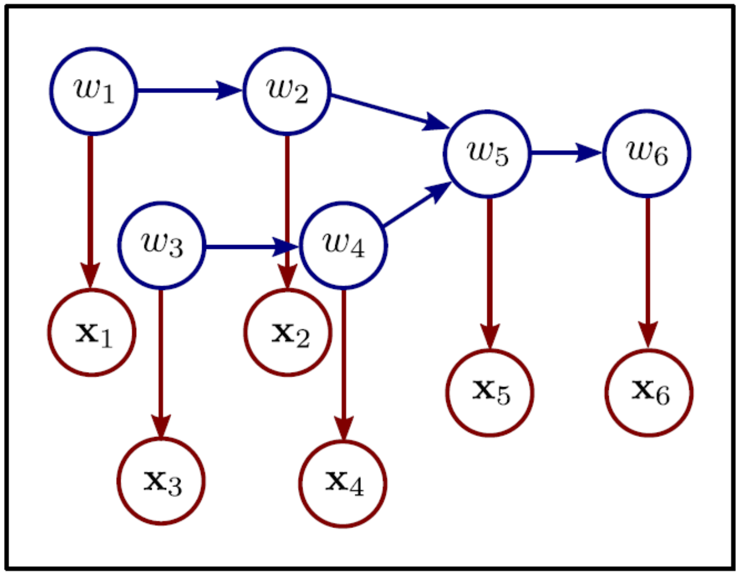
(blue nodes)

Semantic

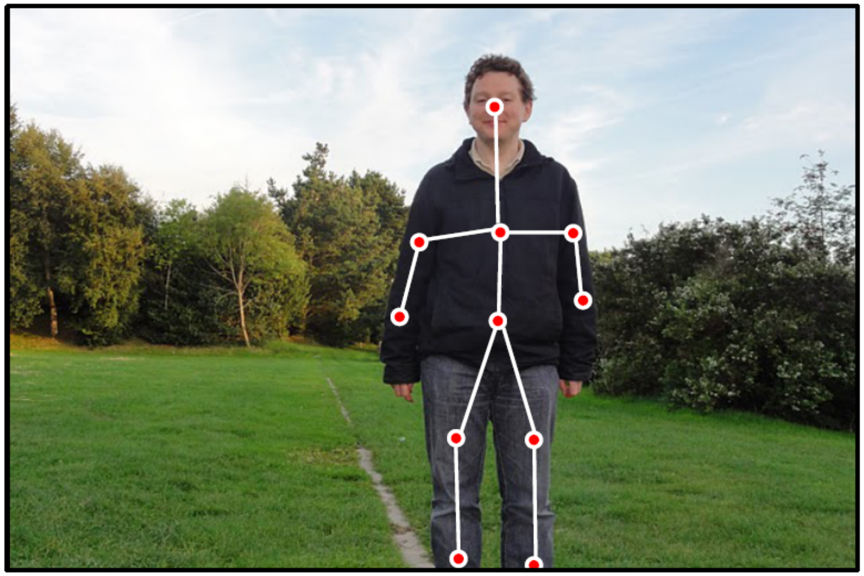
segmentation

Image Credit: Simon JD Prince

Application: Computer Vision

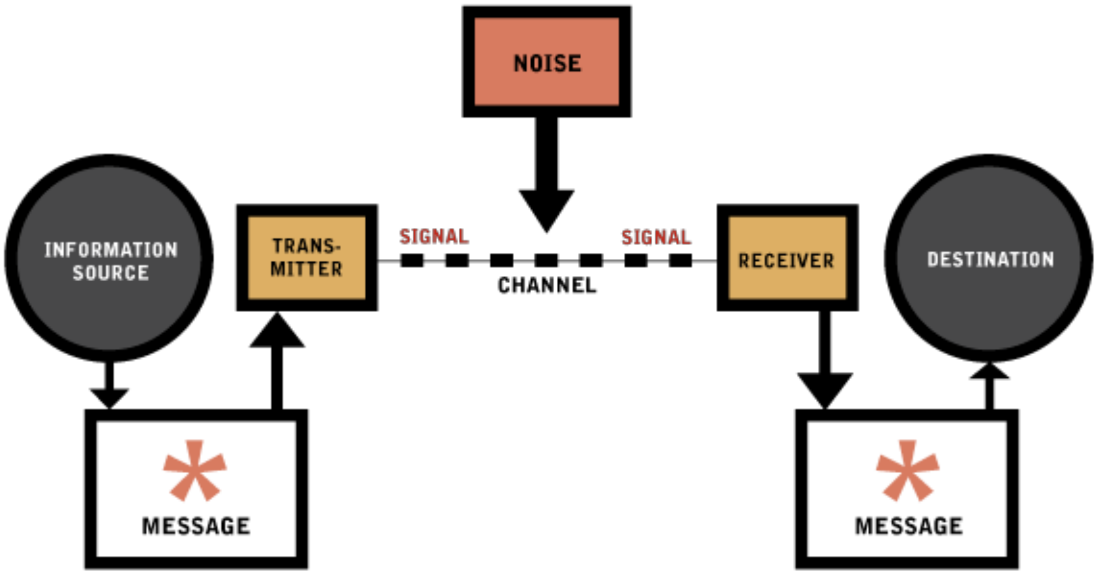


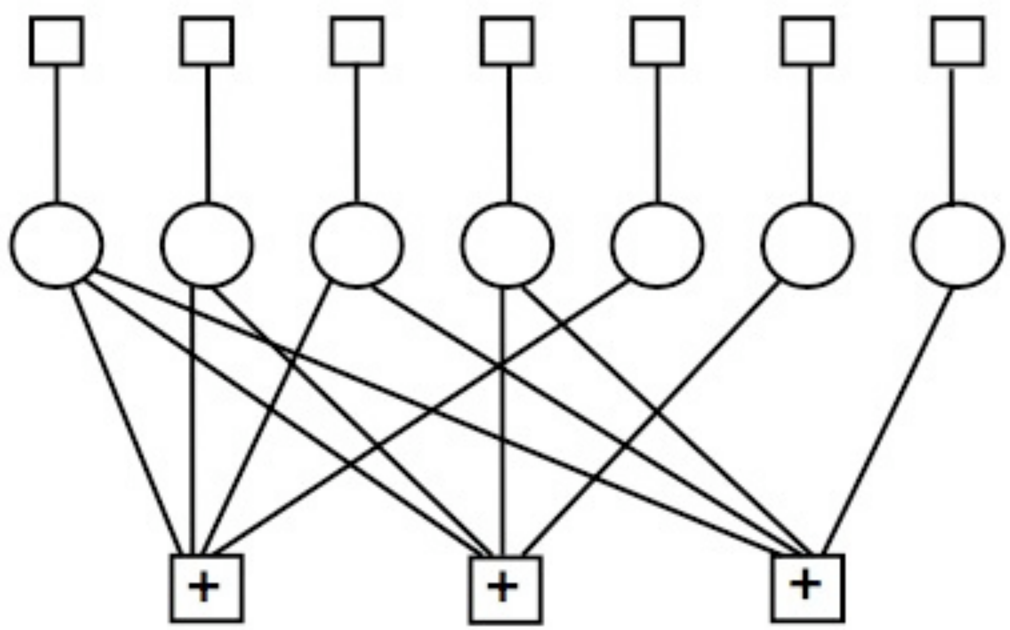
Tree model



Parsing the human body

Application: Coding





Observed Bits

True Bits

Parity Constraints

Application: Medical Diagnosis

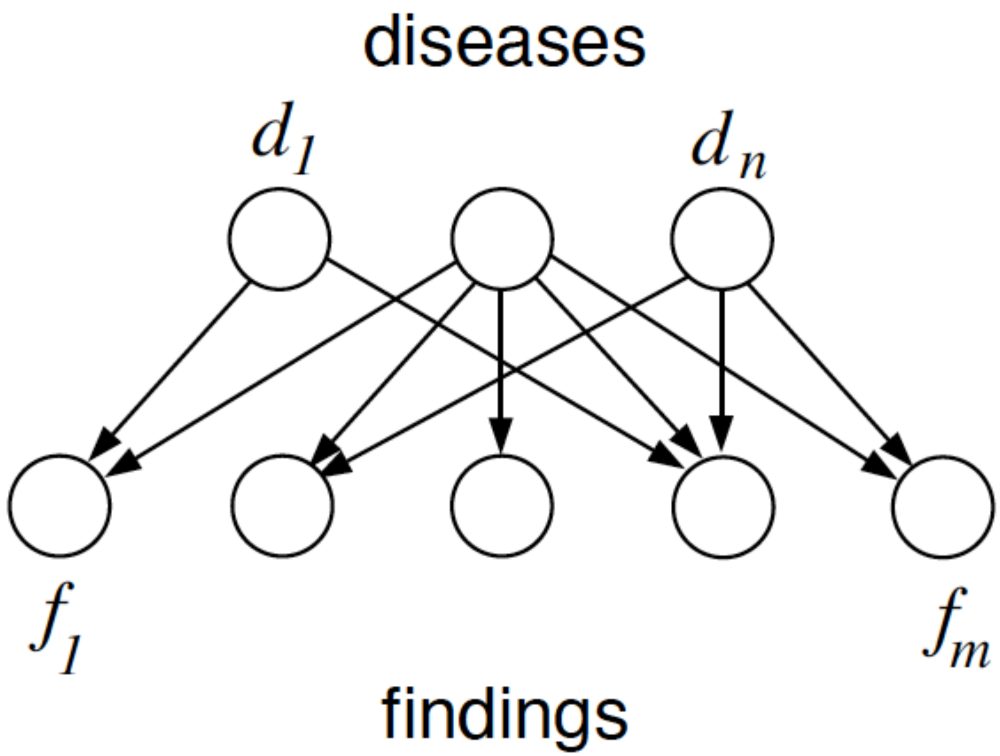


Image Credit: Erik Sudderth

Are MAP and Max of Marginals Consistent?

P(S=f)=0.6
P(S=t)=0.4

P(N|S)

Hardness

- Find P(All variables) — Easy for BN: O(n)

- MAP
  - Find argmax P(All variables | **E**=**e**) — NP-hard
  - Find any assignment P(All variables | **E**=**e**) > p — NP-hard

- Conditional Probability / Marginals
  - Is P(Y=y | **E**=**e**) > 0 — NP-hard
  - Find P(Y=y | **E**=**e**) — #P-hard
  - Find |P(Y=y | **E**=**e**) – p| <= ε — NP-hard for any ε<0.5

- Marginal-MAP
  - Find argmax_{y} Σ_o P(**Y**=**y**, **O**=**o** | **E**=**e**) — NP^{PP}-hard

Inference in BNs hopeless?

- In general, yes!
  - Even approximate!
- In practice
  - Exploit structure
  - Many effective approximation algorithms
    - some with guarantees
- Plan
  - Exact Inference
  - Transition to Undirected Graphical Models (MRFs)
  - Approximate inference in the unified setting

Algorithms

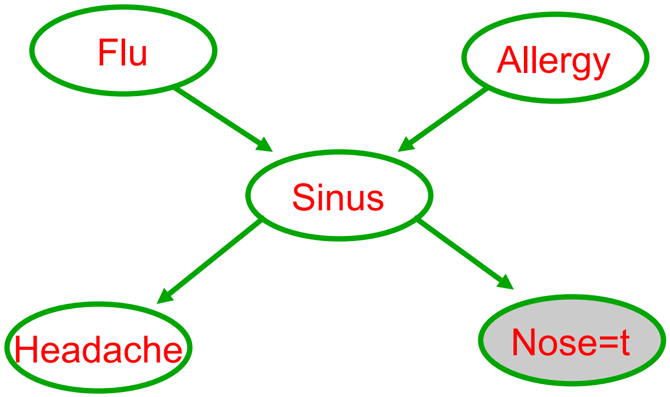
- Conditional Probability / Marginals
  - Variable Elimination
  - Sum-Product Belief Propagation
  - Sampling: MCMC

- MAP
  - Variable Elimination
  - Max-Product Belief Propagation
  - Sampling MCMC

  - Integer Programming
    - Linear Programming Relaxation
  - Combinatorial Optimization (Graph-cuts)

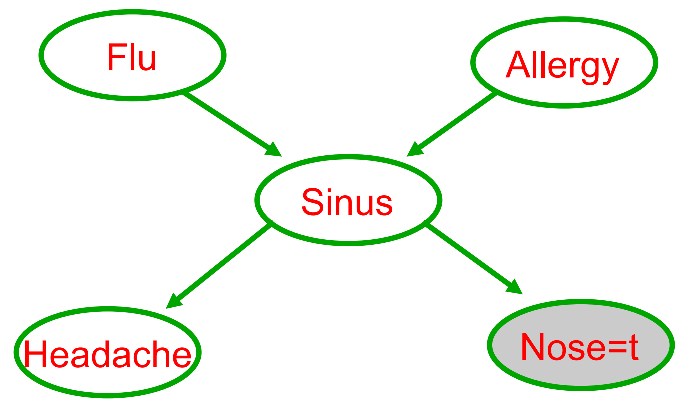
Marginal Inference Example

- Evidence: **E**=**e** (e.g. N=t)
- Query variables of interest **Y**



- Conditional Probability: P(**Y** | **E**=**e**)
  – P(F | N=t)
  – Derivation on board

Marginal Inference Example



Inference seems exponential in number of variables!

Actually, inference in graphical models is NP-hard ☹

Variable elimination algorithm

- Given a BN and a query P(**Y**|**e**) ≈ P(**Y**,**e**)

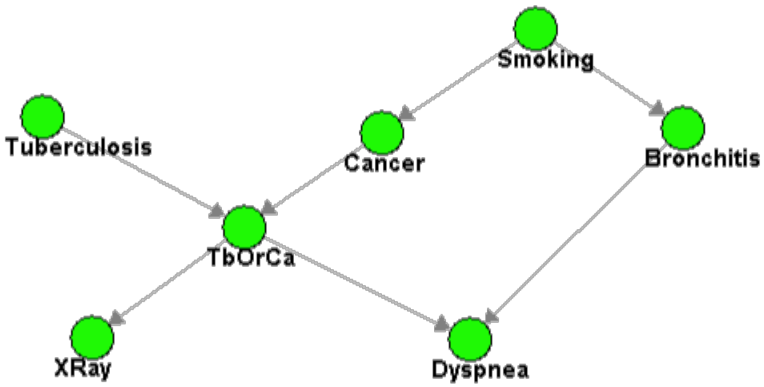
IMPORTANT!!!

- Choose an ordering on variables, e.g., X_1, …, X_n
- For i = 1 to n, If $X_i \notin \{\mathbf{Y},\mathbf{E}\}$
  – Collect factors $f_1,\ldots,f_k$ that include X_i
  – Generate a new factor by eliminating X_i from these factors

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

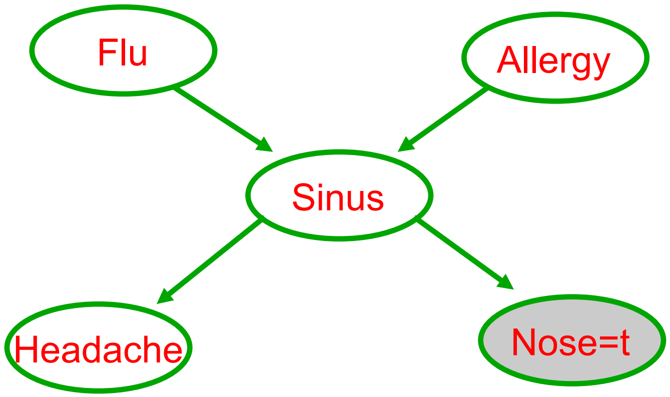
  – Variable X_i has been eliminated!
- Normalize P(**Y**,**e**) to obtain P(**Y**|**e**)

Complexity of variable elimination – Graphs with loops



Exponential in number of variables in largest factor generated

Pruning irrelevant variables



Prune all non-ancestors of query variables
More generally: Prune all nodes not on active trail between evidence and query vars