# Machine Learning (CS 181): 19. Inference in Graphical Models

David C. Parkes and Sasha Rush

Spring 2017



## Contents

1. Introduction
2. Reasoning Patterns, d-Separation
3. Exact Inference
4. Approximate Inference
5. Conclusion

## Contents


1 Introduction

2 Reasoning Patterns, d-Separation

3 Exact Inference

4 Approximate Inference

5 Conclusion




## Overview

- We have seen how to construct (and learn) Bayesian Networks.
- What about reasoning patterns: which variables are conditionally independent?
- What about inference about latent variables:
  - Exact, via variable elimination and generalizations
  - Approximate, via MCMC (Gibbs sampling) and variational methods

# Contents


1 Introduction

2 Reasoning Patterns, d-Separation

3 Exact Inference

4 Approximate Inference

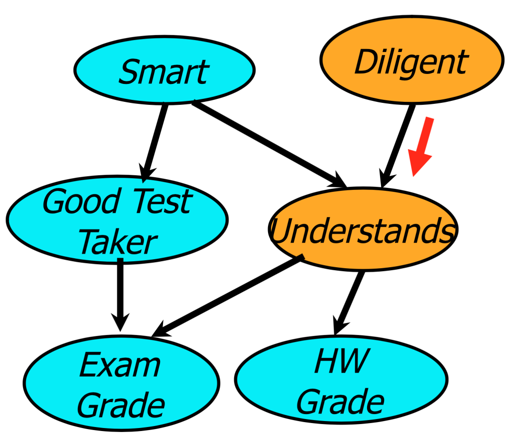
5 Conclusion




# Reasoning Patterns

(Note: assume in running example that a change in a parent has a positive effect; e.g., if GTT true then EG likely to improve).

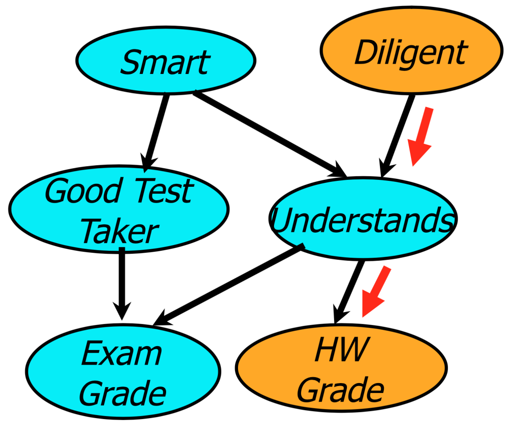
1. Causal. Observe Diligent is true. Does $p(U = true)$ go up, down, or neither?




**Up**. Not independent.

## Reasoning Patterns

2. Chained causal. Observe Diligent is true. Does $p(HG = A)$ go up, down, or neither?

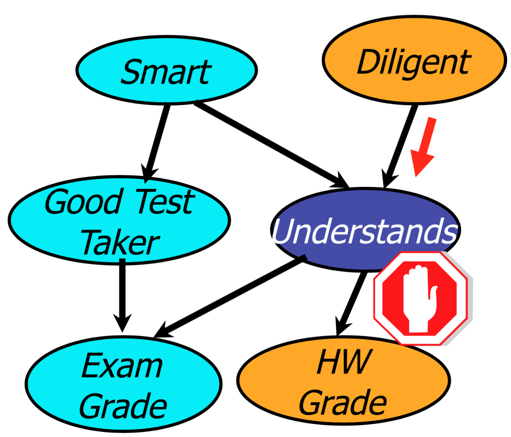



**Up**. Not independent.



## Reasoning Patterns

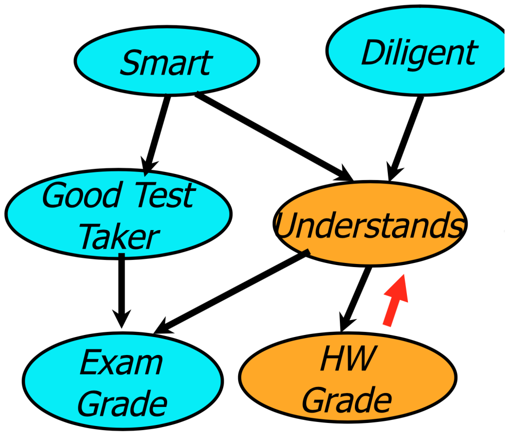
3. Chained causal. Know Understand is true. Now observe Diligent is true. Does $p(HG = A)$ go up, down, or neither?




**Neither.** $I(HWG, D \mid U)$.

## Reasoning Patterns

4. Evidential. Observe $HG = A$. Does $p(U = true)$ go up, down, or neither?

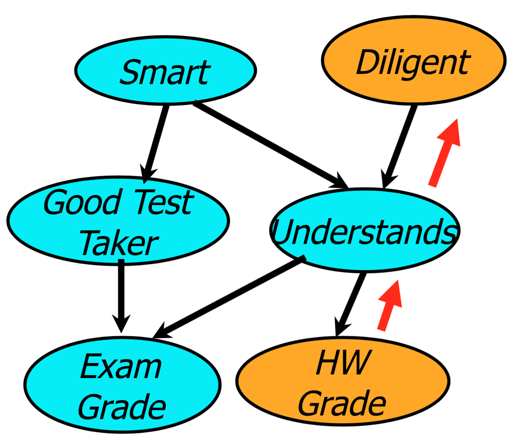



**Up**. Not independent.



## Reasoning Patterns

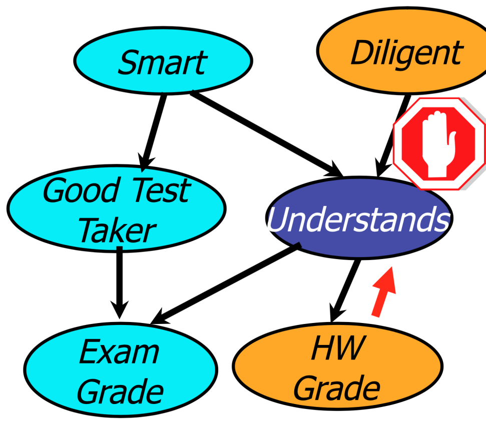
5. Chained evidential. Observe $HG = A$. Does $p(D = true)$ go up, down, or neither?




**Up**. Not independent.

## Reasoning Patterns

6. Chained evidential. Know that $U = true$. Observe $HG = A$. Does $p(D = true)$ go up, down, or neither?

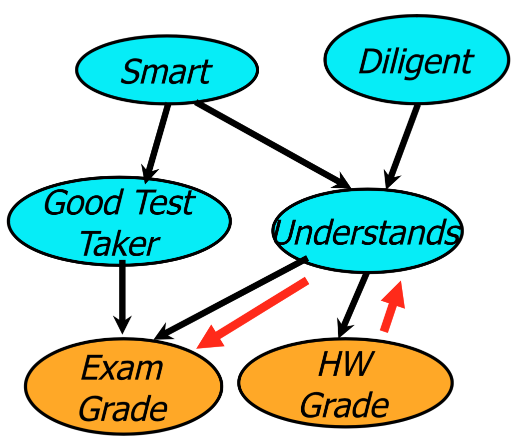



**Neither**. $I(D, HWG \,|\, U)$.



## Reasoning Patterns

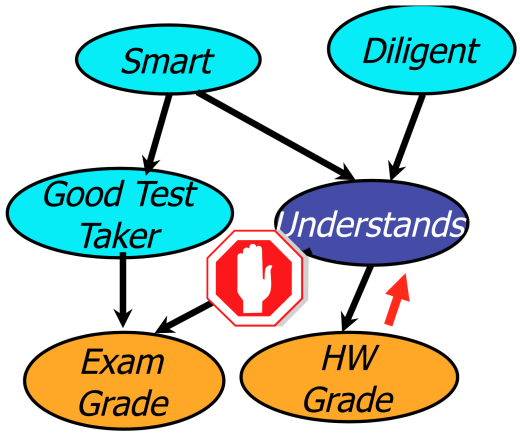
7. Mixed causal-evidential. Observe $HG = A$. Does $p(EG = A)$ go up, down, or neither?




**Up.** Not independent.

## Reasoning Patterns

8. Mixed causal-evidential. We know $U = true$. Observe $HG = A$. Does $p(EG = A)$ go up, down, or neither?

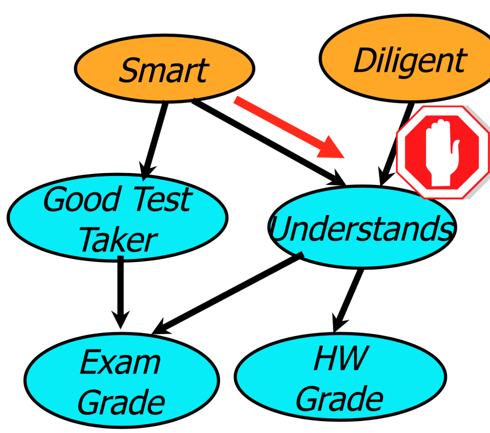



**Neither.** $I(EG, HWG \mid U)$.



## Reasoning Patterns

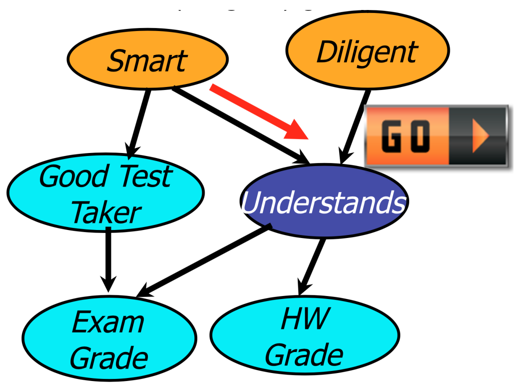
9. Inter-causal reasoning. We observe $S = true$. Does $p(D = true)$ go up, down, or neither?




**Neither.** Independent.

## Reasoning Patterns

10. Inter-causal reasoning. We know that $U = true$. We observe $S = true$. Does $p(D = true)$ go up, down, or neither?



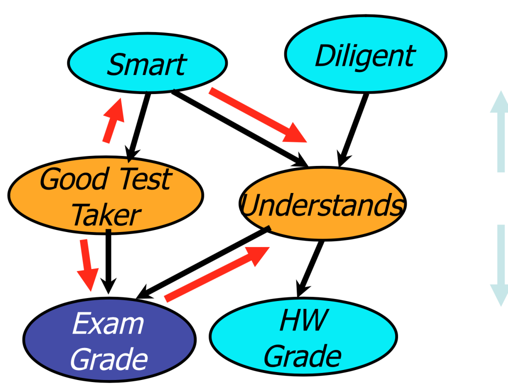

**Down.** not independent, conditioned on Understands!

(this is known as explaining away!)



## Reasoning Patterns

11. Conflicting pattern. We know $EG = A$. We observe $GTT = true$. Does $p(U = true)$ go up, down, or neither?

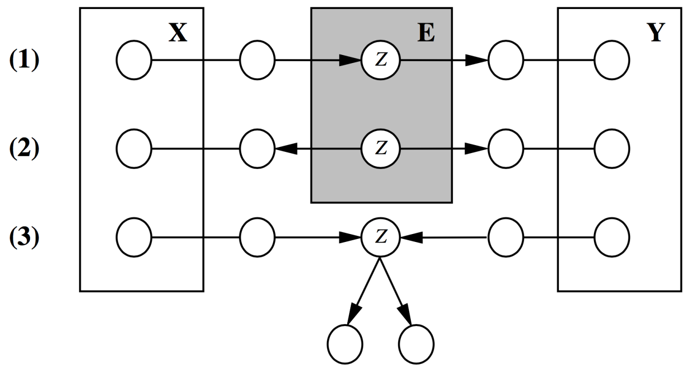



**We don't know.**

## A Sufficient Test for Conditional Independence

One set of variables is conditionally independent of another set given evidence if every undirected path between the two sets is blocked.

Example, illustrating $I(X, Y \mid E)$:




P. Domingos

Paths (1) and (2) are blocked because $Z$ has 'non-converging arrows' and $Z$ is in the evidence. Path (3) is blocked because $Z$ has 'converging arrows' and neither $Z$ nor its descendants are in the evidence.



## d-Separation

**Definition (Directed separation)**

$X_A$ and $X_B$ are d-separated by evidence $X_E$ if every undirected path from a node in $X_A$ to a node in $X_B$ is blocked by $X_E$.

**Definition (Blocked)**

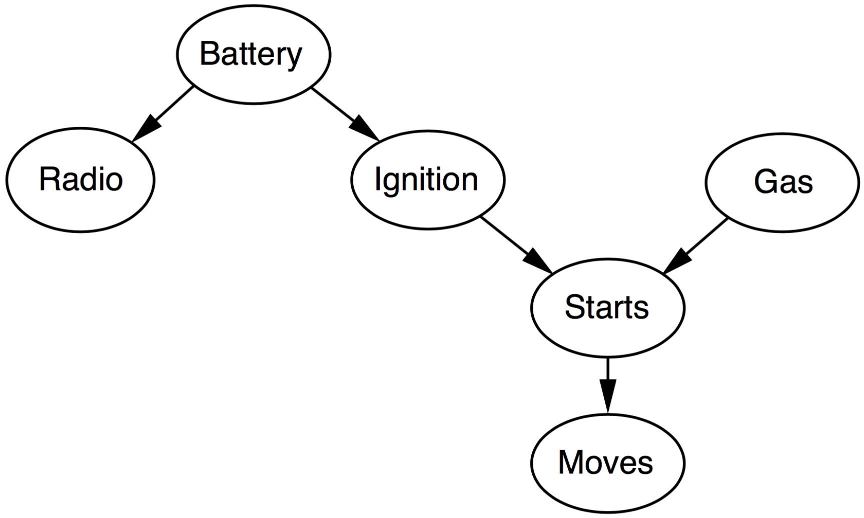
A path is blocked by evidence $X_E$ if either:

- there is a node $Z$ with 'non-converging arrows' on the path, and $Z \in X_E$, or
- there is a node $Z$ with 'converging arrows' on the path, and neither $Z$ nor its descendants are in $X_E$.

**Theorem**

*If $X_A$ and $X_B$ are d-separated by $X_E$, then $I(X_A, X_B \mid X_E)$.*

## Example: Starting a Car

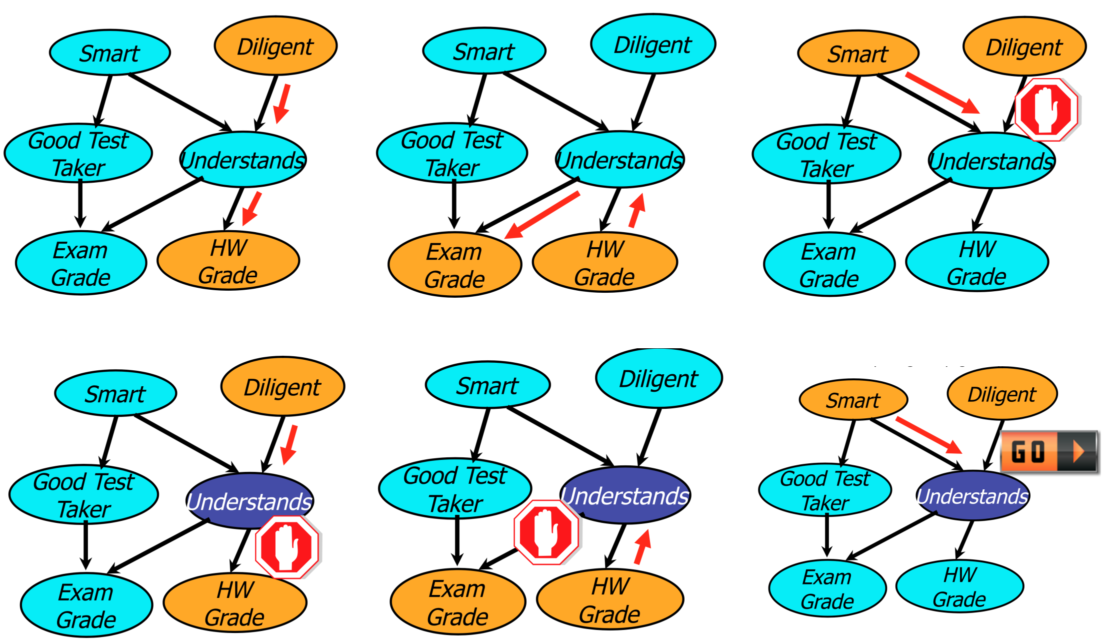



P. Domingos

Are Gas and Radio independent? Given Battery? Ignition? Starts? Moves?



## Checking d-separation on the Reasoning Patterns




(a) (b) (c)

## Contents


1 Introduction

2 Reasoning Patterns, d-Separation

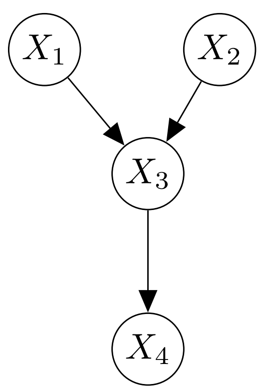
3 Exact Inference

4 Approximate Inference

5 Conclusion




## Exact Inference (1 of 9)




Suppose we want to calculate the marginal probability:

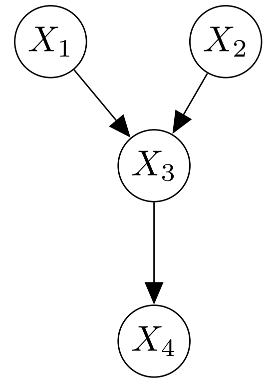
$$p(x_4) = \sum_{x_1,x_2,x_3} p(x_1)p(x_2)p(x_3 \,|\, x_1, x_2)p(x_4 \,|\, x_3)$$

Let $k =$ max domain size. This requires $k^4$ steps ($k^3$ steps for each $x_4$.) Generally, with $m = \#$ variables, we have $k^m$ steps.

# Exact Inference (2 of 9)

Use variable elimination procedure, build intermediate $g$ terms:

$$
\begin{aligned}
p(x_4) &= \sum_{x_1,x_2,x_3} p(x_1)p(x_2)p(x_3 \mid x_1,x_2)p(x_4 \mid x_3) \\
&= \sum_{x_2,x_3} p(x_2)p(x_4 \mid x_3) \underbrace{\sum_{x_1} p(x_1)p(x_3 \mid x_1,x_2)}_{g_1(x_2,x_3)} \\
&= \sum_{x_3} p(x_4 \mid x_3) \underbrace{\sum_{x_2} p(x_2)g_1(x_2,x_3)}_{g_2(x_3)} \\
&= \sum_{x_3} p(x_4 \mid x_3)g_2(x_3) = g_3(x_4)
\end{aligned}
$$

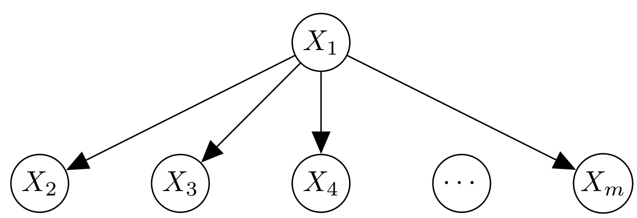



Now: $k^2(k) + k(k) + k(k)$ steps vs $k^4$ steps. Order here is $x_1, x_2, x_3$: leaves first, working towards query.



# Exact Inference (3 of 9)

order of elimination matters




If eliminate $x_1$ first, get

$$p(x_m) = \sum_{x_2,\ldots,x_{m-1}} \sum_{x_1} p(x_1)p(x_2 \mid x_1)\ldots p(x_m \mid x_1) = \sum_{x_2,\ldots,x_{m-1}} g_1(x_2,\ldots,x_m)$$

With ‘leaves-first’ order $x_2,\ldots,x_{m-1},x_1$, get

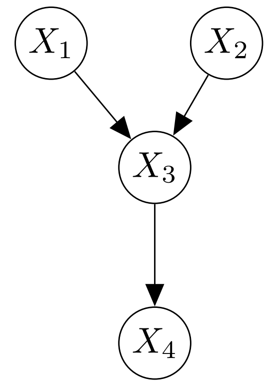
$$
\begin{aligned}
p(x_m) &= \sum_{x_3,\ldots,x_{m-1},x_1} p(x_1)p(x_3 \mid x_1)\ldots p(x_m \mid x_1) \underbrace{\sum_{x_2} p(x_2 \mid x_1)}_{g_1(x_1)} \\
&= \sum_{x_4,\ldots,x_{m-1},x_1} p(x_1)\ldots p(x_m \mid x_1) \underbrace{\sum_{x_3} p(x_3 \mid x_1)g_1(x_1)}_{g_2(x_1)} = \cdots
\end{aligned}
$$

This requires $mk^2$ steps vs $k^m$ steps (!).

# Exact Inference (4 of 9)

- Cost of variable elimination is exponential in the number of variables mentioned by the intermediate factors $g(\cdot)$.
- Example ($g_1$ mentions two variables):

$$
\begin{aligned}
p(x_4) &= \sum_{x_1,x_2,x_3} p(x_1)p(x_2)p(x_3 \,|\, x_1,x_2)p(x_4 \,|\, x_3) \\
&= \sum_{x_2,x_3} p(x_2)p(x_4 \,|\, x_3) \underbrace{\sum_{x_1} p(x_1)p(x_3 \,|\, x_1,x_2)}_{g_1(x_2,x_3)}
\end{aligned}
$$

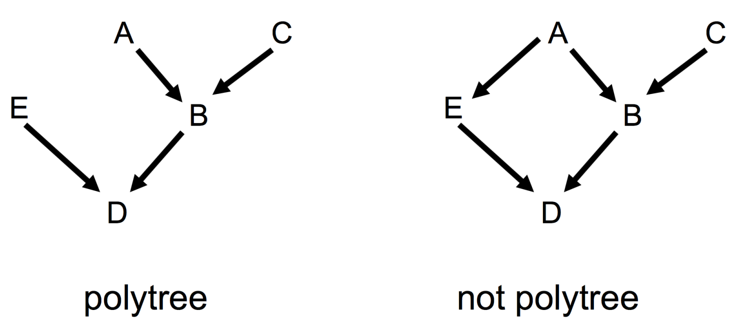



- The tree width of a BN is the minimum over all elimination orders of the largest number of mentions in intermediate factors.



# Exact Inference (5 of 9)

Inference is easy for polytrees.



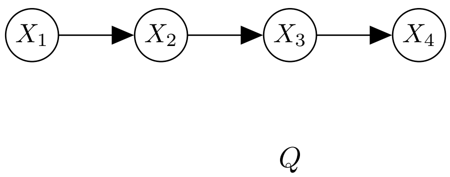

Let $d = $ max $\#$ parents

**Theorem**

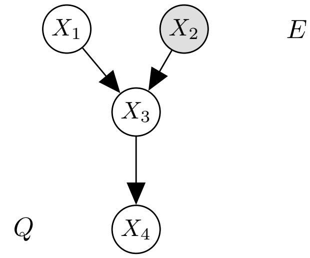
*For Bayesian Networks that are polytrees ($\equiv$ no undirected cycles) then 'leaves first ordering' is optimal and gives $O(mk^{d+1})$ steps.*

Linear in the size of the representation!

# Exact Inference (6 of 9)




(a)




(b)

Additional observations:

(a) We can prune vars that are not ancestors to $Q$ or $E$:

$$p(x_3) = \sum_{x_1,x_2,x_4} p(x_1)p(x_2 \mid x_1)p(x_3 \mid x_2)p(x_4 \mid x_3)$$
$$= \sum_{x_1,x_2} p(x_1)p(x_2 \mid x_1)p(x_3 \mid x_2) \underbrace{\sum_{x_4} p(x_4 \mid x_3)}_{=1}$$

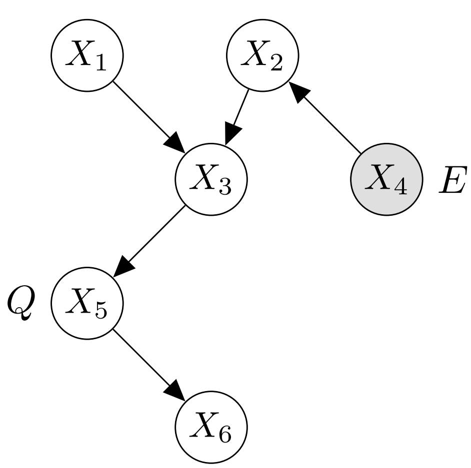
(b) For $p(x_Q \mid \mathbf{x}_E)$, we can instantiate the evidence $\mathbf{x}_E$ in the BN and then reduce the network.

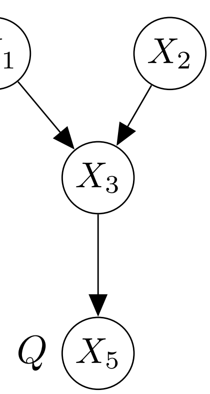


# Exact Inference (7 of 9)

General polytree inference procedure:

- Prune any non-ancestors of query or evidence variables
- Instantiate evidence variables
- Find leaves, and do variable elimination in order of leaves, working back towards the query



## Exact Inference (8 of 9)

- Exact inference is #P-hard in general BNs.
  - #P problems are counting problems, e.g., number of subsets of lists of integers that add to zero.
  - Solving in poly time would imply $P = NP$.
- NP-hard to determine whether there exists an elimination order where no intermediate function mentions more than $\ell$ variables.
  - NP problems are decision problems for which 'yes'-instances are easy to verify, e.g., "is there a solution to a traveling salesperson problem with cost $\leq c$?" NP-hard are the hardest problems in NP.
  - Conjectured that P $\neq$ NP.
- Typical to use a greedy heuristic, select as next var to eliminate the one that generates a $g$ function with as few vars as possible.



## Exact Inference (9 of 9)

- Variable elimination is for computing the marginal probability of one variable, e.g. $p(x_4 \,|\, \mathbf{x}_E)$.
- What if we want to perform multiple inference tasks with the same evidence?
- Use the sum-product message passing algorithm on polytrees. This is a generalization of the 'forward-backward' algorithm. (Generalizes, via junction-tree algorithm to general networks.)

# Contents


1 Introduction

2 Reasoning Patterns, d-Separation

3 Exact Inference

4 Approximate Inference

5 Conclusion




# Approximate Inference (1 of 9)

Because exact inference on general BNs is #P-hard, it is also important to have methods of approximate inference.

Two common approaches:

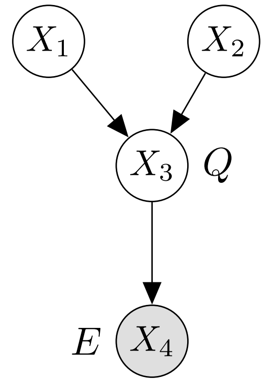
- Stochastic approximations via Markov Chain Monte Carlo methods.
- Variational methods.

We give a sketch of the ideas.

## Approximate inference (2 of 9)

One idea: rejection sampling to estimate posterior, $p(\mathbf{x}_Q \mid \mathbf{x}_E)$:

- Sample $\mathbf{x}$ from the joint distribution $p(\mathbf{x})$ (recall: use topological order)
- Reject any sample where evidence $\mathbf{x}_E$ is not satisfied. Use other samples to estimate posterior.



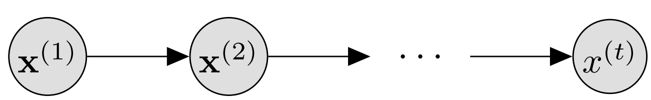

Pro: very simple. Con: fraction of samples rejected grows exponentially as the size of $E$ grows.



## Approximate inference (3 of 9)

Markov chain Monte Carlo (MCMC) methods:

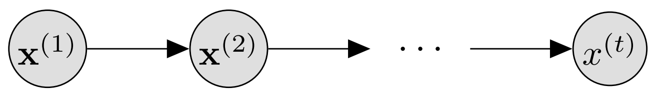
- An approach for generating samples from the posterior distribution
- Construct a Markov chain, where each state ($\mathbf{x}^{(t)}$ at step $t$) corresponds to an instantiation of the variables.




- Define the transition model such that the stationary distr. of the Markov chain (the distribution the state will be in at $T$, as $T \to \infty$) is equal to the posterior.

# Approximate inference (4 of 9)

- Construct a Markov chain, where each state ($\mathbf{x}^{(t)}$ at step $t$) corresponds to an instantiation of the variables.
- Let $P^{(t)}$ denote the distribution on states after $t$ steps. Idea is that $P^{(t)}$ will converge, for large $t$, to the posterior.
- The next state is sampled $q(\mathbf{x}^{(t+1)} \mid \mathbf{x}^{(t)})$. Define $q$ such that:
  - stationary distr. of chain is equal to posterior
  - convergence is fast
  - $q$ is tractable to sample from

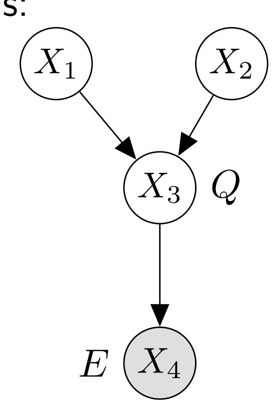





# Approximate inference (5 of 9)

Gibbs sampling is a useful MCMC method for BNs:

- Fix evidence variables throughout. Initialize rest of variables arbitrarily.
- Sample each of the non-evidence variables at random, sampling each variable given the current values of the other variables.

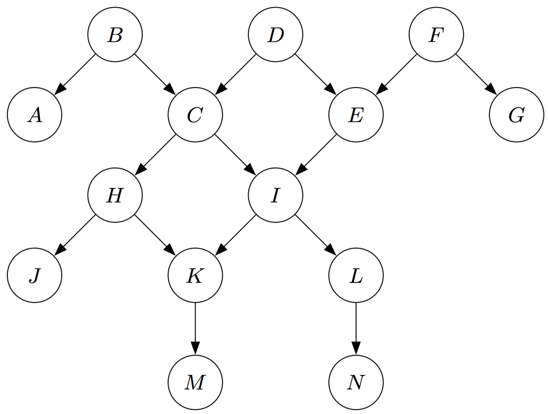



Need: $p(x_3 \mid x_1, x_2, x_4), p(x_2 \mid x_1, x_3, x_4), p(x_1 \mid x_2, x_3, x_4)$.
How can we compute these conditional distributions?

## Approximate inference (6 of 9)

A: via the Markov blanket of a variable. This is the set of parents, children and childrens' parents.

**Theorem:** Each variable is conditionally independent of all others given its Markov blanket (via d-separation arguments.)




T. Nielsen and F. Verner Jensen

The Markov blanket of $I$ is $\{C, E, H, K, L\}$. Leads to fast calculation of conditional distr. on any variable, given values of rest of variables.



## Approximate inference (7 of 9)

Still, Gibbs sampling can be too slow for large BNs because the successive samples are highly correlated, and thus it can take a large number of samples to achieve an unbiased estimate of the posterior.

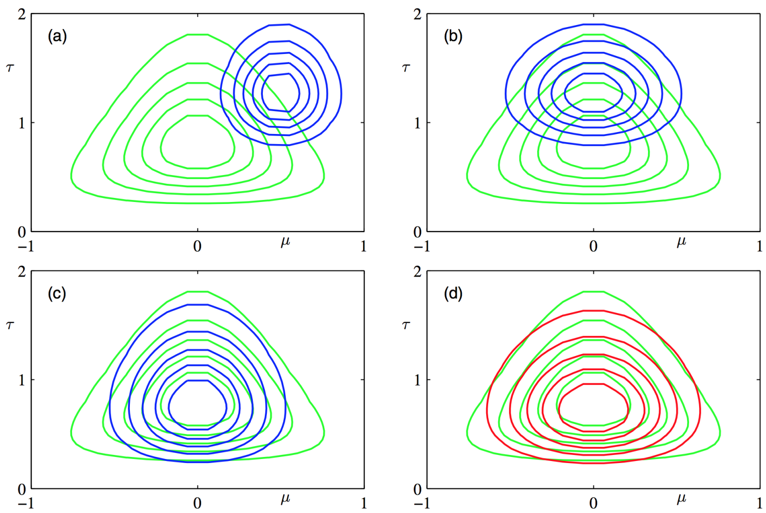
# Approximate inference (8 of 9)

Leads to variational methods. Estimate posterior.

$$\min_{\mathbf{w}} ||p'(\mathbf{x}_Q; \mathbf{w}), p(\mathbf{x}_Q \mid \mathbf{x}_E)||$$

where $p'$ is a simpler distribution, and for some measure of distance.

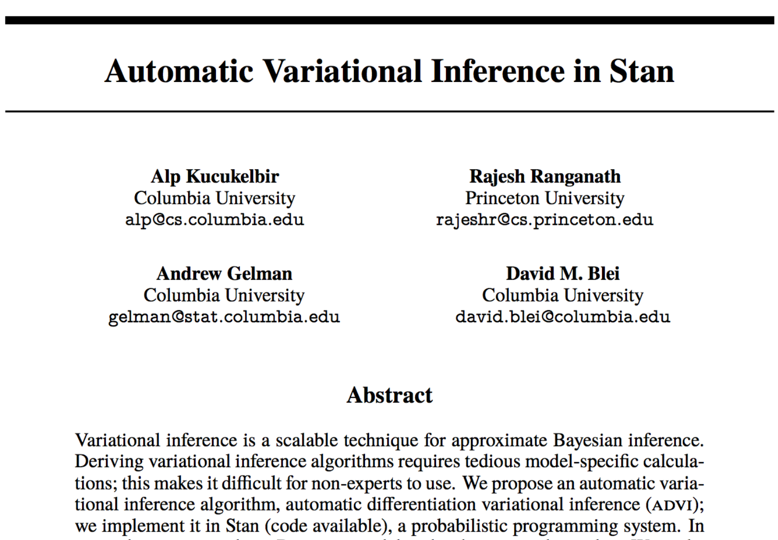
Choose family $p'$ to allow for fast optimization, but close approximation.






# Approximate inference (8 of 9)

Variational approximations are a very active area at the moment, and being coupled with probabilistic programming languages such as Stan.



**Automatic Variational Inference in Stan**


**Alp Kucukelbir**
Columbia University
alp@cs.columbia.edu

**Rajesh Ranganath**
Princeton University
rajeshr@cs.princeton.edu

**Andrew Gelman**
Columbia University
gelman@stat.columbia.edu

**David M. Blei**
Columbia University
david.blei@columbia.edu



**Abstract**

Variational inference is a scalable technique for approximate Bayesian inference. Deriving variational inference algorithms requires tedious model-specific calculations; this makes it difficult for non-experts to use. We propose an automatic variational inference algorithm, automatic differentiation variational inference (ADVI); we implement it in Stan (code available), a probabilistic programming system. In

## Contents


1 Introduction

2 Reasoning Patterns, d-Separation

3 Exact Inference

4 Approximate Inference

5 Conclusion




## Conclusion

- Bayesian networks provide a compact representation of distributions on lots of variables.
- We can understand conditional independence via d-separation.
- For exact inference in polytrees, variable elimination is fast and effective.
- For approximate inference, both MCMC via Gibbs sampling and variational methods are in wide effect.