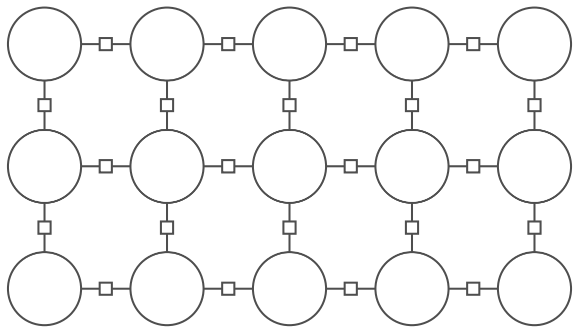
Markov Networks and Bayesian Networks I



Lecture

Markov Networks: Overview

Markov Networks: Gibbs Sampling

Bayesian Networks: Overview

Bayesian Networks: Definitions

Course plan

Constraint satisfaction problems

Search problems

Markov decision processes

Markov networks

Adversarial games

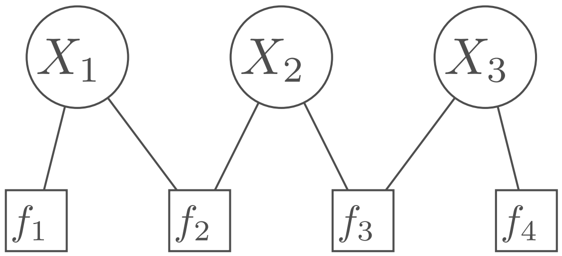
Bayesian networks

Reflex **States** **Variables** **Logic**

Low-level High-level

Machine learning

Review: factor graphs



Definition: factor graph

Variables:

$X = (X_1, \ldots, X_n)$, where $X_i \in \text{Domain}_i$

Factors:

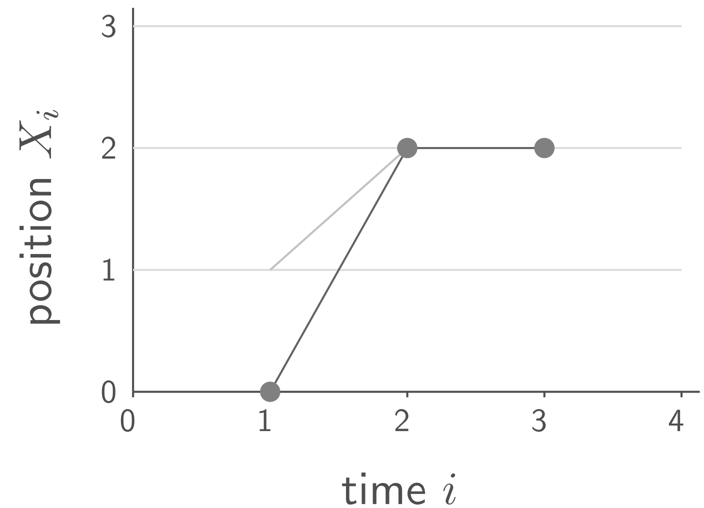
$f_1, \ldots, f_m$, with each $f_j(X) \geq 0$

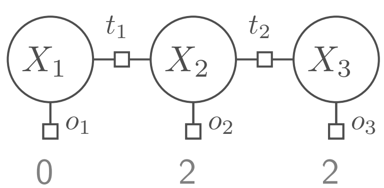
Definition: assignment weight

Each **assignment** $x = (x_1, \ldots, x_n)$ has a **weight**:

$$\text{Weight}(x) = \prod_{j=1}^{m} f_j(x)$$

Example: object tracking





x_1	$o_1(x_1)$
0	2
1	1
2	0

x_2	$o_2(x_2)$
0	0
1	1
2	2

x_3	$o_3(x_3)$
0	0
1	1
2	2

$\lvert x_i - x_{i+1}\rvert$	$t_i(x_i, x_{i+1})$
0	2
1	1
2	0

[demo]

Maximum weight assignment

CSP objective: find the maximum weight assignment

$$\max_x \mathsf{Weight}(x)$$

x_1	x_2	x_3	$\mathsf{Weight}(x)$
0	1	1	4
0	1	2	4
1	1	1	4
1	1	2	4
1	2	1	2
1	2	2	8

Maximum weight assignment: $\{x_1 : 1, x_2 : 2, x_3 : 2\}$ (weight 8)

But this doesn't represent all the other possible assignments...

Definition

Definition: Markov network

A Markov network is a factor graph which defines a joint distribution over random variables $X = (X_1, \dots, X_n)$:

$$\mathbb{P}(X = x) = \frac{\text{Weight}(x)}{Z}$$

where $Z = \sum_{x'} \text{Weight}(x')$ is the normalization constant.

x_1	x_2	x_3	Weight(x)	$\mathbb{P}(X = x)$
0	1	1	4	0.15
0	1	2	4	0.15
1	1	1	4	0.15
1	1	2	4	0.15
1	2	1	2	0.08
1	2	2	8	0.31

$$Z = 4 + 4 + 4 + 4 + 2 + 8 = 26$$

Represents uncertainty!

Marginal probabilities

Example question: where was the object at time step 2 (X_2)?

Definition: Marginal probability

The marginal probability of $X_i = v$ is given by:

$$\mathbb{P}(X_i = v) = \sum_{x : x_i = v} \mathbb{P}(X = x)$$

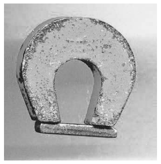
Object tracking example:

x_1	x_2	x_3	Weight(x)	$\mathbb{P}(X = x)$
0	1	1	4	0.15
0	1	2	4	0.15
1	1	1	4	0.15
1	1	2	4	0.15
1	2	1	2	0.08
1	2	2	8	0.31

$\mathbb{P}(X_2 = 1) = 0.15 + 0.15 + 0.15 + 0.15 = 0.62$

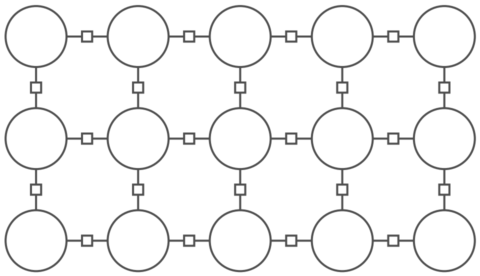
$\mathbb{P}(X_2 = 2) = 0.08 + 0.31 = 0.38$

Note: different than max weight assignment!



Application: Ising model

Ising model: classic model from statistical physics to model ferromagnetism

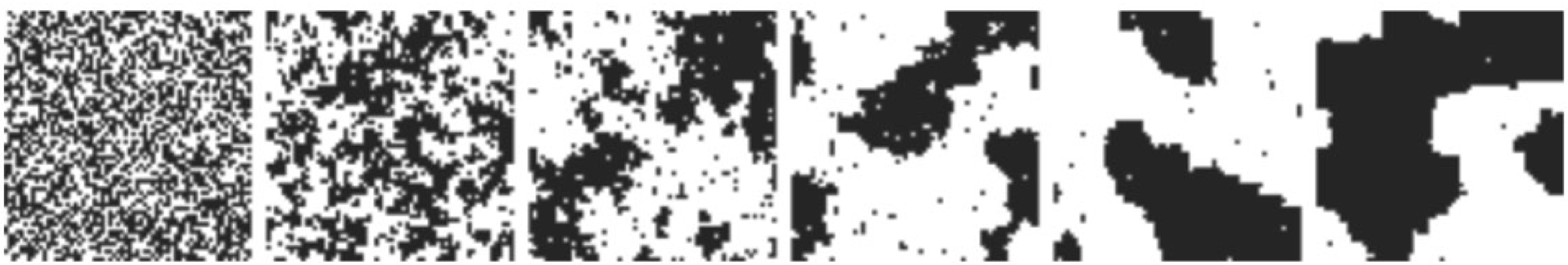


$X_i \in \{-1, +1\}$: atomic spin of site i

$f_{ij}(x_i, x_j) = \exp(\beta x_i x_j)$ wants same spin

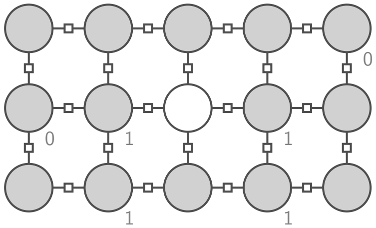
Samples as β increases:

Figure 2 from Perez (1998)



Application: image denoising

Example: image denoising



- $X_i \in \{0, 1\}$ is pixel value in location i
- Subset of pixels are observed

$o_i(x_i) = [x_i = \text{observed value at } i]$

- Neighboring pixels more likely to be same than different

$t_{ij}(x_i, x_j) = [x_i = x_j] + 1$

Summary

$$\boxed{\text{Markov networks} = \text{factor graphs} + \text{probability}}$$

- Normalize weights to get probability distribution
- Can compute marginal probabilities to focus on variables

CSPs	Markov networks
variables	random variables
weights	probabilities
maximum weight assignment	marginal probabilities

Lecture

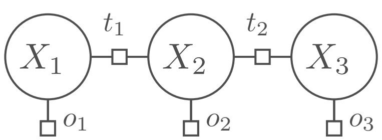
Markov Networks: Overview

Markov Networks: Gibbs Sampling

Bayesian Networks: Overview

Bayesian Networks: Definitions

Review: Markov networks



Definition: Markov network

A Markov network is a factor graph which defines a joint distribution over random variables $X = (X_1, \dots, X_n)$:

$$\mathbb{P}(X = x) = \frac{\text{Weight}(x)}{Z}$$

where $Z = \sum_{x'} \text{Weight}(x')$ is the normalization constant.

Objective: compute marginal probabilities $\mathbb{P}(X_i = v) = \sum_{x:x_i=v} \mathbb{P}(X = x)$

x_1	x_2	x_3	Weight(x)	$\mathbb{P}(X = x)$
0	1	1	4	0.15
0	1	2	4	0.15
1	1	1	4	0.15
1	1	2	4	0.15
1	2	1	2	0.08
1	2	2	8	0.31

$Z = 4 + 4 + 4 + 4 + 2 + 8 = 26$

$\mathbb{P}(X_2 = 1) = 0.15 + 0.15 + 0.15 + 0.15 = 0.62$
$\mathbb{P}(X_2 = 2) = 0.08 + 0.31 = 0.38$

Gibbs sampling

Algorithm: Gibbs sampling

Initialize x to a random complete assignment

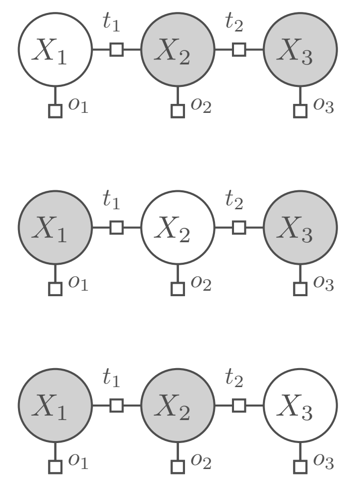
Loop through $i = 1, \dots, n$ until convergence:

  Set $x_i = v$ with prob. $\mathbb{P}(X_i = v \mid X_{-i} = x_{-i})$

    (X_{-i} denotes all variables except X_i)

  Increment $\text{count}_i(x_i)$

Estimate $\hat{\mathbb{P}}(X_i = x_i) = \frac{\text{count}_i(x_i)}{\sum_v \text{count}_i(v)}$



Example: sampling one variable

$\text{Weight}(x \cup \{X_2 : 0\}) = 1$ prob. 0.2

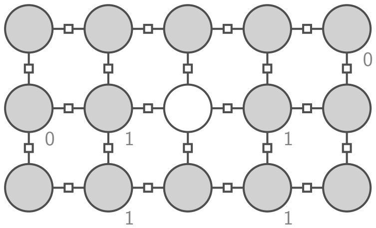
$\text{Weight}(x \cup \{X_2 : 1\}) = 2$ prob. 0.4

$\text{Weight}(x \cup \{X_2 : 2\}) = 2$ prob. 0.4

[demo]

Application: image denoising

Example: image denoising



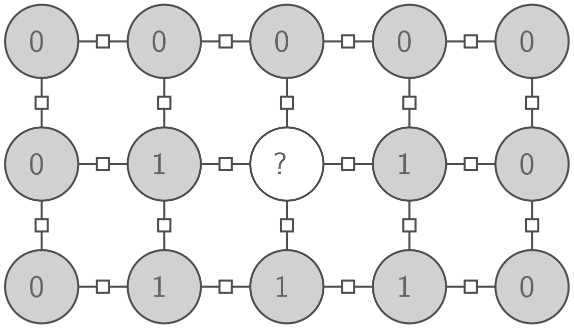
- $X_i \in \{0, 1\}$ is pixel value in location i
- Subset of pixels are observed

$o_i(x_i) = [x_i = \text{observed value at } i]$

- Neighboring pixels more likely to be same than different

$t_{ij}(x_i, x_j) = [x_i = x_j] + 1$

Gibbs sampling for image denoising



$$t_{ij}(x_i, x_j) = [x_i = x_j] + 1$$

Scan through image and update each pixel given rest:

v	weight	$\mathbb{P}(X_i = v \mid X_{-i} = x_{-i})$
0	$2 \cdot 1 \cdot 1 \cdot 1$	0.2
1	$1 \cdot 2 \cdot 2 \cdot 2$	0.8

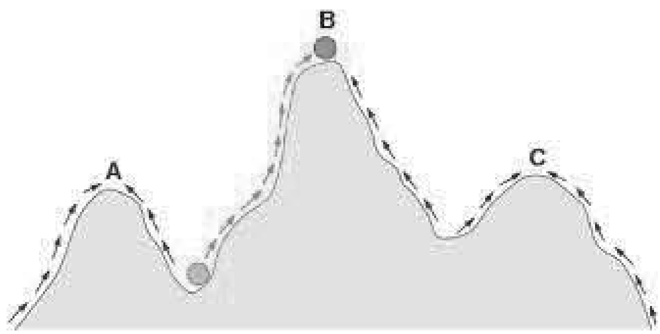
Image denoising demo

[see web version]

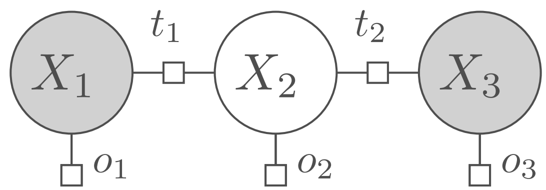
Search versus sampling

Iterated Conditional Modes	Gibbs sampling
maximum weight assignment	marginal probabilities
choose best value	sample a value
converges to local optimum	marginals converge to correct answer*

*under technical conditions (sufficient condition: all weights positive), but could take exponential time



Summary



- Objective: compute marginal probabilities $\mathbb{P}(X_i = x_i)$
- Gibbs sampling: sample one variable at a time, count visitations
- More generally: Markov chain Monte Carlo (MCMC) powerful toolkit of randomized procedures

Lecture

Markov Networks: Overview

Markov Networks: Gibbs Sampling

Bayesian Networks: Overview

Bayesian Networks: Definitions

Course plan

Search problems

Markov decision processes

Adversarial games

Constraint satisfaction problems

Markov networks

Bayesian networks

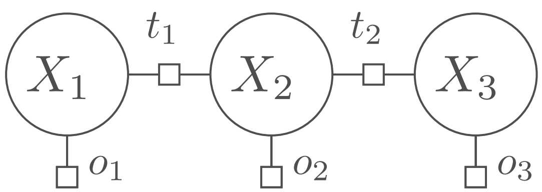
Reflex **States** **Variables** **Logic**

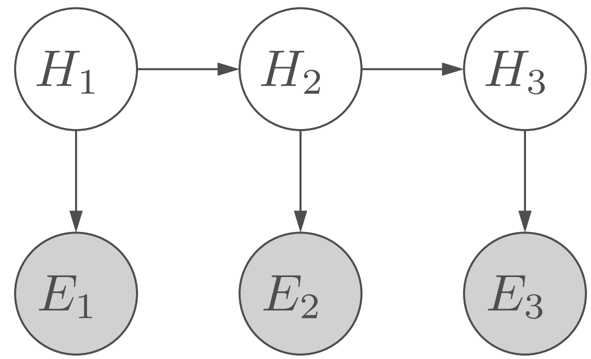
Low-level → High-level

Machine learning

Markov networks versus Bayesian networks

Both define a joint probability distribution over assignments





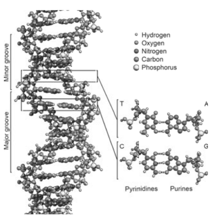
Markov networks	Bayesian networks
arbitrary factors	local conditional probabilities
set of preferences	generative process

Applications

Topic modeling: unsupervised discovery of topics in text

Vision as inverse graphics: recover semantic description given image

Error correcting codes: recover data over a noisy channel



DNA matching: identify people based on relatives

Why Bayesian networks?

- Handle **heterogeneously** missing information, both at training and test time

- Incorporate **prior** knowledge (e.g., Mendelian inheritance, laws of physics)

- Can **interpret** all the intermediate variables

- Precursor to **causal** models (can do interventions and counterfactuals)

Roadmap: Bayesian Networks

Modeling

Definitions

Probabilistic programming

Inference

Probabilistic inference

Forward-backward

Particle filtering

Learning

Supervised learning

Smoothing

EM algorithm

Lecture

Markov Networks: Overview

Markov Networks: Gibbs Sampling

Bayesian Networks: Overview

Bayesian Networks: Definitions

Review: probability

Random variables: sunshine $S \in \{0, 1\}$, rain $R \in \{0, 1\}$

Joint distribution (probabilistic database):

$\mathbb{P}(S, R) =$

s	r	$\mathbb{P}(S = s, R = r)$
0	0	0.20
0	1	0.08
1	0	0.70
1	1	0.02

Marginal distribution:

(aggregate rows)

$\mathbb{P}(S) =$

s	$\mathbb{P}(S = s)$
0	0.28
1	0.72

Conditional distribution:

(select rows, normalize)

$\mathbb{P}(S \mid R = 1) =$

s	$\mathbb{P}(S = s \mid R = 1)$
0	0.8
1	0.2

Review: probability

Variables: S (sunshine), R (rain), T (traffic), A (autumn)

Joint distribution (probabilistic database):

$$\mathbb{P}(S, R, T, A)$$

Marginal conditional distribution (probabilistic inference):

- **Condition** on evidence (traffic, autumn): $T = 1, A = 1$
- Interested in **query** (rain?): R

$$\boxed{\begin{array}{c}\mathbb{P}(\underbrace{R}_{\text{query}} \mid \underbrace{T = 1, A = 1}_{\text{condition}}) \\ (S \text{ is } \textbf{marginalized out})\end{array}}$$

A puzzle

Problem: earthquakes, burglaries, and alarms

Earthquakes and **burglaries** are independent events (probability ϵ).

Either will cause an **alarm** to go off.

Suppose you get an **alarm**.

Does hearing that there's an **earthquake** increase, decrease, or keep constant the probability of a **burglary**?

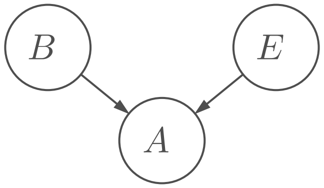
Joint distribution:

$$\mathbb{P}(E, B, A)$$

Questions:

$$\mathbb{P}(B = 1 \mid A = 1) \qquad \textbf{?} \qquad \mathbb{P}(B = 1 \mid A = 1, E = 1)$$

Bayesian network (alarm)



b	$p(b)$
1	ϵ
0	$1-\epsilon$

e	$p(e)$
1	ϵ
0	$1-\epsilon$

b	e	a	$p(a \mid b, e)$
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

$$p(b) = \epsilon \cdot [b = 1] + (1 - \epsilon) \cdot [b = 0]$$

$$p(e) = \epsilon \cdot [e = 1] + (1 - \epsilon) \cdot [e = 0]$$

$$p(a \mid b, e) = [a = (b \vee e)]$$

$$\mathbb{P}(B = b, E = e, A = a) \stackrel{\text{def}}{=} p(b)p(e)p(a \mid b, e)$$

Probabilistic inference (alarm)

Joint distribution

b	e	a	$\mathbb{P}(B = b, E = e, A = a)$
0	0	0	$(1 - \epsilon)^2$
0	0	1	0
0	1	0	0
0	1	1	$(1 - \epsilon)\epsilon$
1	0	0	0
1	0	1	$\epsilon(1 - \epsilon)$
1	1	0	0
1	1	1	ϵ^2

Questions:

$$\mathbb{P}(B = 1) = \epsilon(1 - \epsilon) + \epsilon^2 = \epsilon$$

$$\mathbb{P}(B = 1 \mid A = 1) = \frac{\epsilon(1-\epsilon)+\epsilon^2}{\epsilon(1-\epsilon)+\epsilon^2+(1-\epsilon)\epsilon} = \frac{1}{2-\epsilon}$$

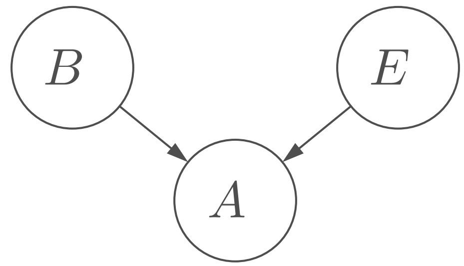
$$\mathbb{P}(B = 1 \mid A = 1, E = 1) = \frac{\epsilon^2}{\epsilon^2+(1-\epsilon)\epsilon} = \epsilon$$

[demo]

News flash: earthquakes decrease burglaries!*

*This is not a causal statement!

Explaining away



Key idea: explaining away

Suppose two causes positively influence an effect. Conditioned on the effect, further conditioning on one cause reduces the probability of the other cause.

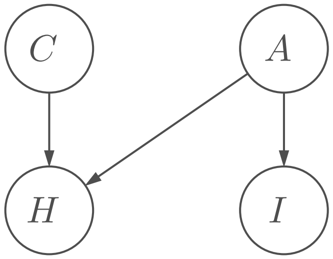
$$\mathbb{P}(B = 1 \mid A = 1, E = 1) < \mathbb{P}(B = 1 \mid A = 1)$$

Note: happens even if causes are independent!

Medical diagnosis

Problem: cold or allergies?

You are coughing and have itchy eyes. Do you have a cold?



Random variables:

cold C, allergies A, cough H, itchy eyes I

Joint distribution:

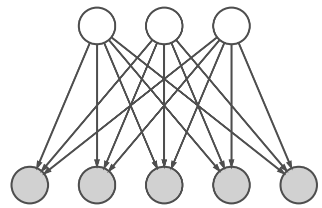
$\mathbb{P}(C = c, A = a, H = h, I = i) = p(c)p(a)p(h \mid c, a)p(i \mid a)$

Questions:

$$\mathbb{P}(C = 1 \mid H = 1) = 0.28 \qquad \mathbb{P}(C = 1 \mid H = 1, I = 1) = 0.13$$

[demo]

Bayesian network (definition)



Definition: Bayesian network

Let $X = (X_1, \dots, X_n)$ be random variables.

A **Bayesian network** is a directed acyclic graph (DAG) that specifies a joint distribution over X as a product of local conditional distributions, one for each node:

$$\mathbb{P}(X_1 = x_1, \dots, X_n = x_n) \stackrel{\text{def}}{=} \prod_{i=1}^{n} p(x_i \mid x_{\text{Parents}(i)})$$

Probabilistic inference (definition)

Input

Bayesian network: $\mathbb{P}(X_1, \dots, X_n)$

Evidence: $E = e$ where $E \subseteq X$ is subset of variables

Query: $Q \subseteq X$ is subset of variables

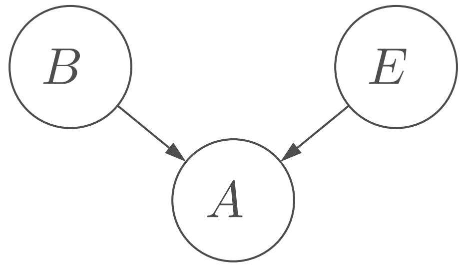
Output

$\mathbb{P}(Q \mid E = e) \longleftrightarrow \mathbb{P}(Q = q \mid E = e)$ for all values q

Example: if coughing and itchy eyes, have a cold?

$$\mathbb{P}(C \mid H = 1, I = 1)$$

Summary



- Random variables capture state of world
- Directed edges between variables represent dependencies
- Local conditional distributions $\Rightarrow$ joint distribution
- Probabilistic inference: ask questions about world
- Captures reasoning patterns (e.g., explaining away)

Summary: Markov and Bayesian Networks I

- Markov Networks: Factor graphs + Probability

- Gibbs sampling is an algorithm for estimating marginal probabilities

- Bayesian Networks, represent generative processes, related to Factor graphs and Markov Networks

- Bayesian Networks Definitions: explaining away

- Next: Inference in Bayesian networks