Exploiting Structure in Probability Distributions

Irit Gat-Viks

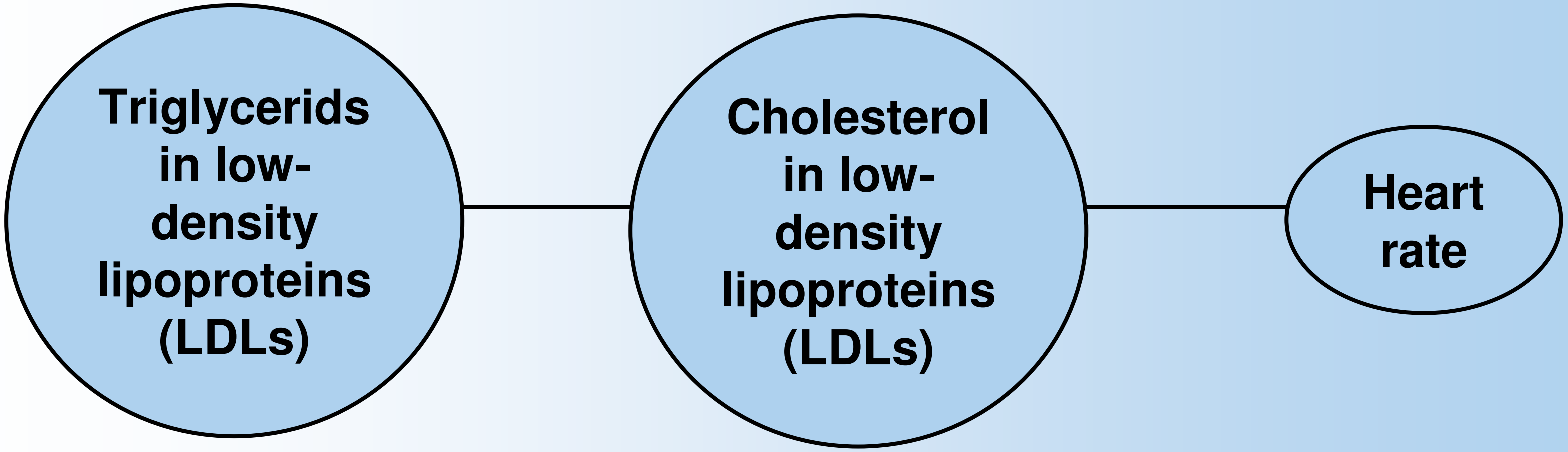
Based on presentation and lecture notes of
Nir Friedman, Hebrew University

General References:

- D. Koller and N. Friedman, *probabilistic graphical models*
- Pearl, *Probabilistic Reasoning in Intelligent Systems*
- Jensen, *An Introduction to Bayesian Networks*
- Heckerman, *A tutorial on learning with Bayesian networks*

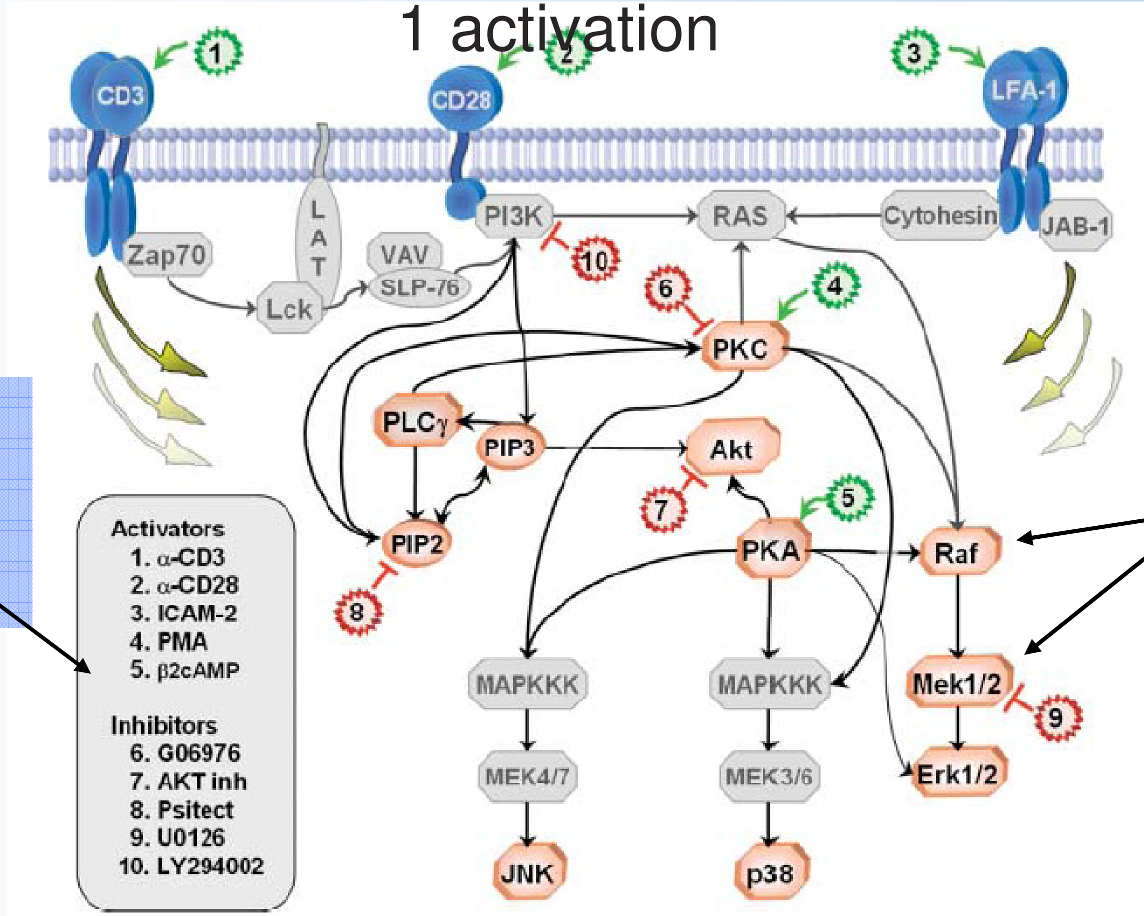
Example I

Relationships in obesity



Example II

The currently accepted consensus network of human primary CD4 T cells, downstream of CD3, CD28, and LFA-1 activation

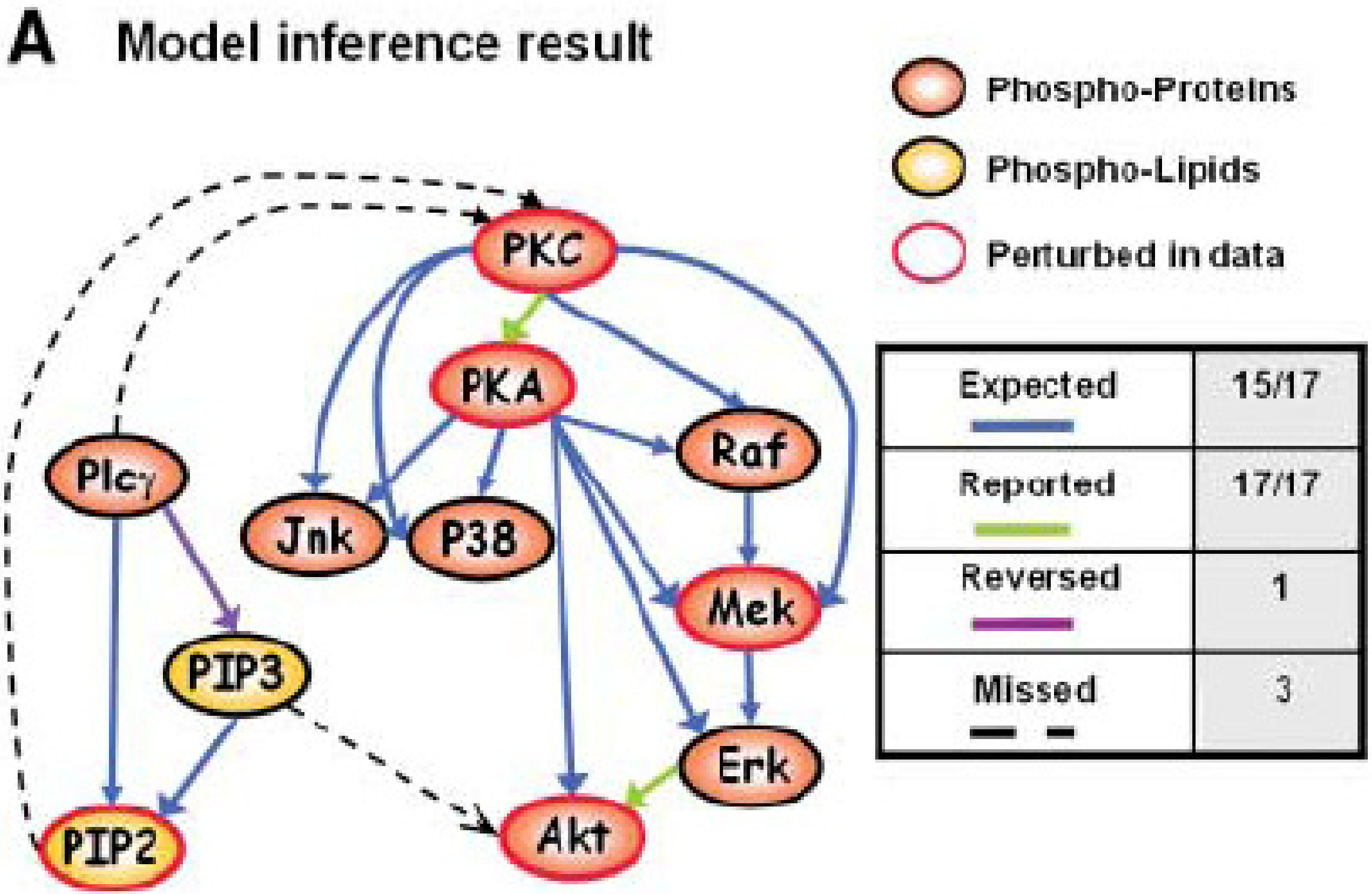


10 Perturbations by small molecules

11 Colored nodes: Measured signaling proteins

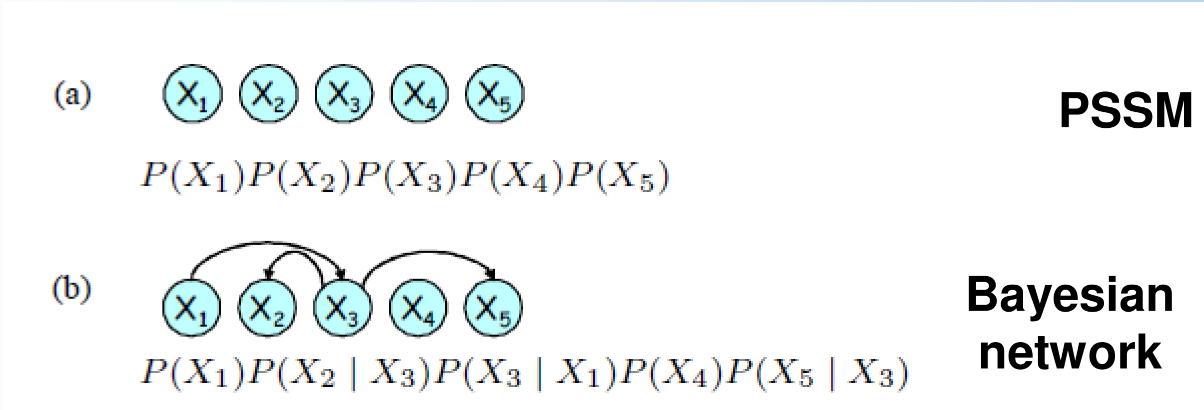
Causal Protein-Signaling Networks Derived from Multiparameter Single-Cell Data. Karen Sachs, *et al. 2005.*

Bayesian network results

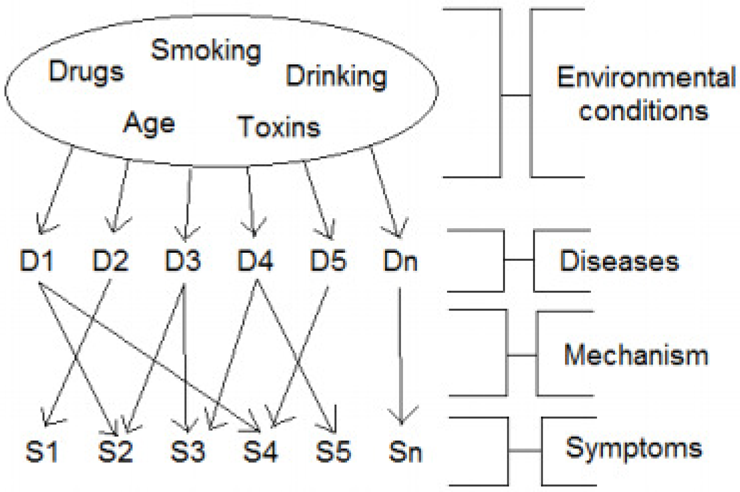


- **PKC->PKA was validated experimentally.**
- **Akt was not affected by Erk in additional experiments**

Example III: Bayesian network models for a transcription factor-DNA binding motif with 5 positions



Example IV: Diagnostic Bayesian network model



Basic Probability Definitions

- **Product Rule**: P(A,B)=P(A | B)*P(B)= P(B | A)*P(A)
- **Independence** between A and B: P(A,B)=P(A)*P(B), or alternatively: P(A|B)=P(A), P(B|A)=P(B).
- **Total probability theorem**: $\bigcup_{i=1}^{n} B_i = \Omega, \ \forall i \neq j \ B_i \bigcap B_j = \phi$

$$P(A) = \sum_{i=1}^{n} P(A, B_i) = \sum_{i=1}^{n} P(B_i) * P(A \mid B_i)$$

Basic Probability Definitions

- Bayes Rule:

$$P(A \mid B) = \frac{P(B \mid A) \cdot P(A)}{P(B)}$$

$$P(A \mid B, C) = \frac{P(B \mid A, C) \cdot P(A \mid C)}{P(B \mid C)}$$

- Chain Rule:

$$P(X_1, \ldots, X_n) =$$
$$P(X_1 \mid X_2, \ldots, X_n) \cdot P(X_2 \mid X_3, \ldots, X_n) \cdot P(X_3 \mid X_4, \ldots, X_n) \cdot \ldots P(X_{n-1} \mid X_n) \cdot P(X_n)$$

Exploiting Independence Property

- G: whether the woman is pregnant
- D: whether the doctor’s test is positive

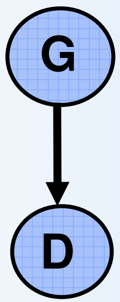
The joint distribution representation P(g,d):

G	D	P(G,D)
0	0	0.54
0	1	0.06
1	0	0.02
1	1	0.38

Factorial representation

Using conditional probability: P(g,d)=P(g)*P(d|g).

The distribution of P(g), P(d|g):



g0	0.6
g1	0.4

P(d|g)	d0	d1
g0	0.9	0.1
g1	0.05	0.95

Example: P(g0,d1)=0.06 vs. P(g0)*P(d1|g0)=0.6*0.1=0.06

Exploiting Independence Property

- H: home test
- Independence assumption: Ind(H;D|G) (i.e., given G, H is independent of D).

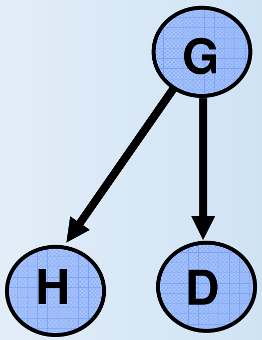
Joint distribution

Factorial representation

$$P(d,h,g)=P(d,h|g)*P(g)=P(d|g)*\ P(h|g)*P(g)$$

Product rule

Ind(H;D|G)



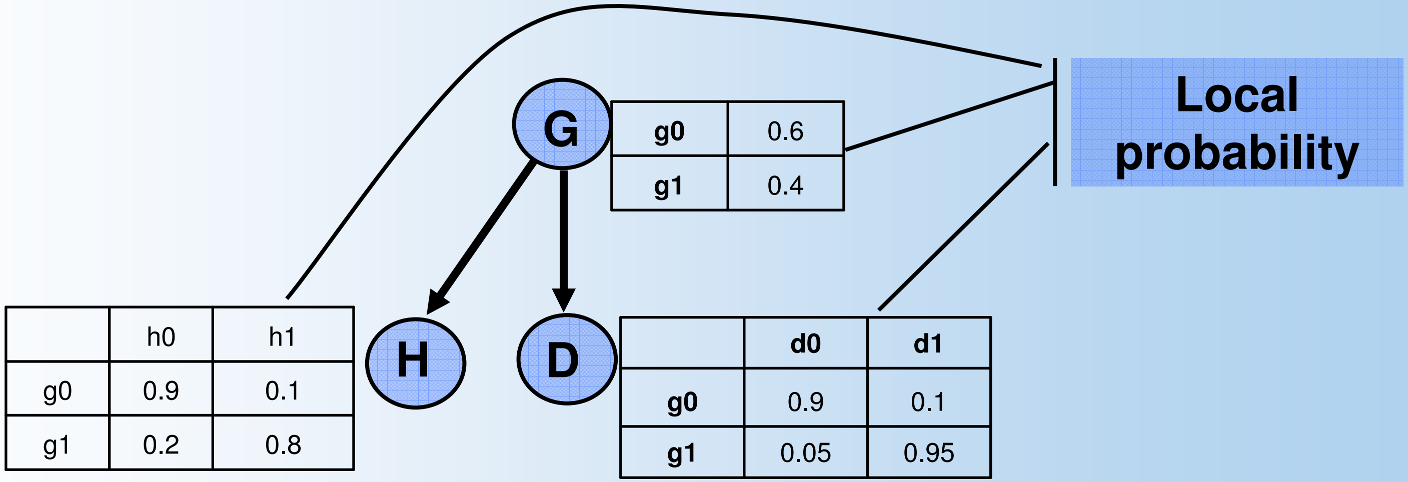
g0	0.6
g1	0.4

	h0	**h1**
g0	0.9	0.1
g1	0.2	0.8

	d0	**d1**
g0	0.9	0.1
g1	0.05	0.95

Exploiting Independence Property

	representation of P(d,g,h)	
	joint distribution	factored distribution
No. of parameters	7	5
Adding new variable H	changing the distribution entirely	**Modularity**: reuse the local probability model. (Only new local probability model for H.)



g0	0.6
g1	0.4

	h0	h1
g0	0.9	0.1
g1	0.2	0.8

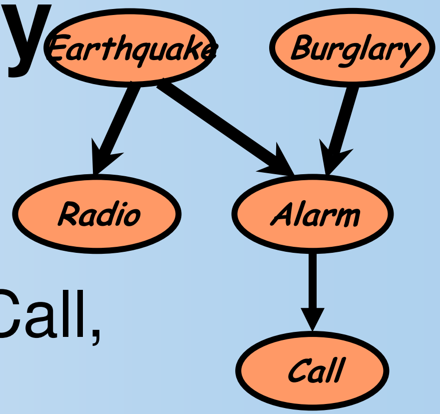
	d0	**d1**
g0	0.9	0.1
g1	0.05	0.95

=> **Bayesian networks**: Exploiting independence properties of the distribution in order to allow a compact and natural representation.

Outline

- Introduction
- Bayesian Networks
  - » <u>Representation & Semantics</u>
  - Inference in Bayesian networks
  - Learning Bayesian networks

Representing the Uncertainty



- A story with five random variables:
  - Burglary, Earthquake, Alarm, Neighbor Call, Radio Announcement
  - Specify joint distribution with $2^5=32$ parameters

maybe…

- An expert system for monitoring intensive care patients
  - Specify joint distribution over 37 variables with (at least) 2^{37} parameters

no way!!!

Probabilistic Independence: a Key for Representation and Reasoning

- Recall that if X and Y are **independent** given Z then

$$P(X \mid Z, Y) = P(X \mid Y)$$

- In our story…if
  - *burglary* and *earthquake* are **independent**
  - *alarm sound* and *radio* are **independent** given *earthquake*
  - *burglary* and *radio* are **independent** given *earthquake*
- then instead of 15 parameters we need 8

$$P(A, R, E, B) = P(A \mid R, E, B) \cdot P(R \mid E, B) \cdot P(E \mid B) \cdot P(B)$$

versus

$$P(A, R, E, B) = P(A \mid E, B) \cdot P(R \mid E) \cdot P(E) \cdot P(B)$$

Need a language to represent independence statements

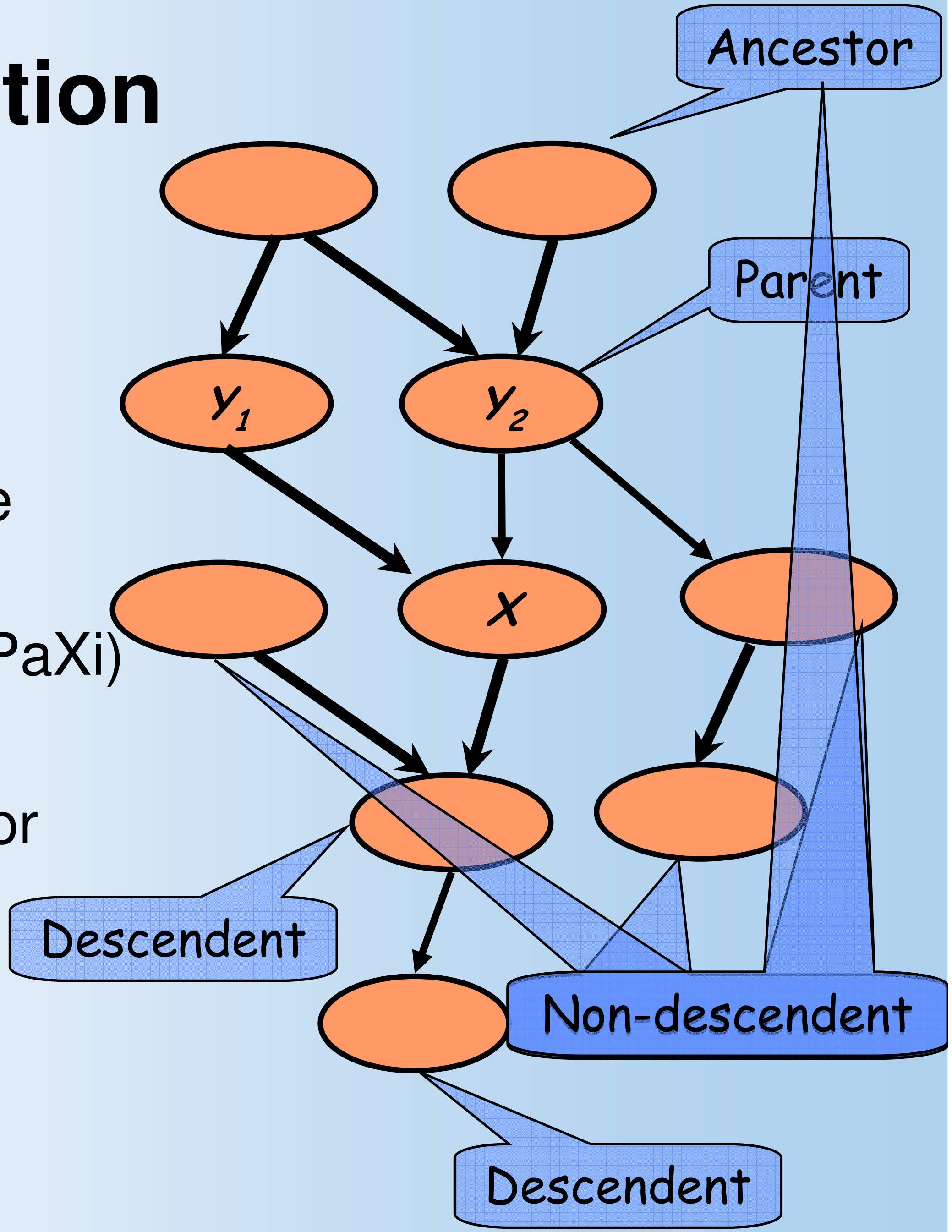
Markov Assumption

Generalizing:

◆A child is **conditionally independent** from its non-descendents, given the value of its parents.

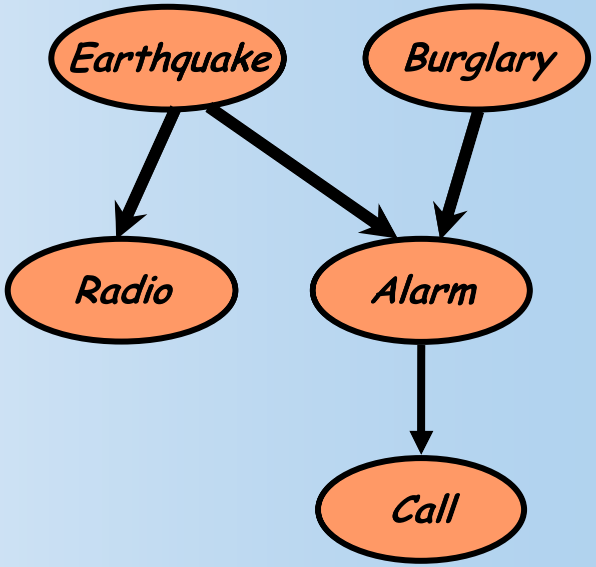
Ind(Xi ; NonDescendantXi | PaXi)

◆It is a natural assumption for many **causal** processes

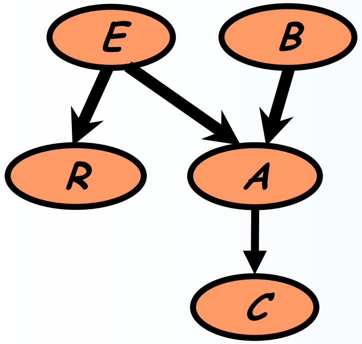


Markov Assumption (cont.)

- Examples:
  - R is independent of A, B, C, given E
  - A is independent of R, given B and E
  - C is independent of B, E, R, given A



Bayesian Network Semantics



Qualitative part
conditional independence statements in BN structure

+

Quantitative part
local probability Models (e.g., multinomial, linear Gaussian)

=

Unique joint distribution over domain

◆ Compact & efficient representation:

- nodes have $\leq k$ parents $\Rightarrow$ $O(2^k n)$ vs. $O(2^n)$ params
- parameters pertain to local interactions

P(C,A,R,E,B) = P(B)*P(E|B)*P(R|E,B)*P(A|R,B,E)*P(C|A,R,B,E)

versus

P(C,A,R,E,B) = P(B)*P(E) * P(R|E) * P(A|B,E) * P(C|A)

➔ **In general:** $P(x_1,\ldots,x_n) = \prod_{i=1,..,n} P(x_i \mid Pa_{xi})$

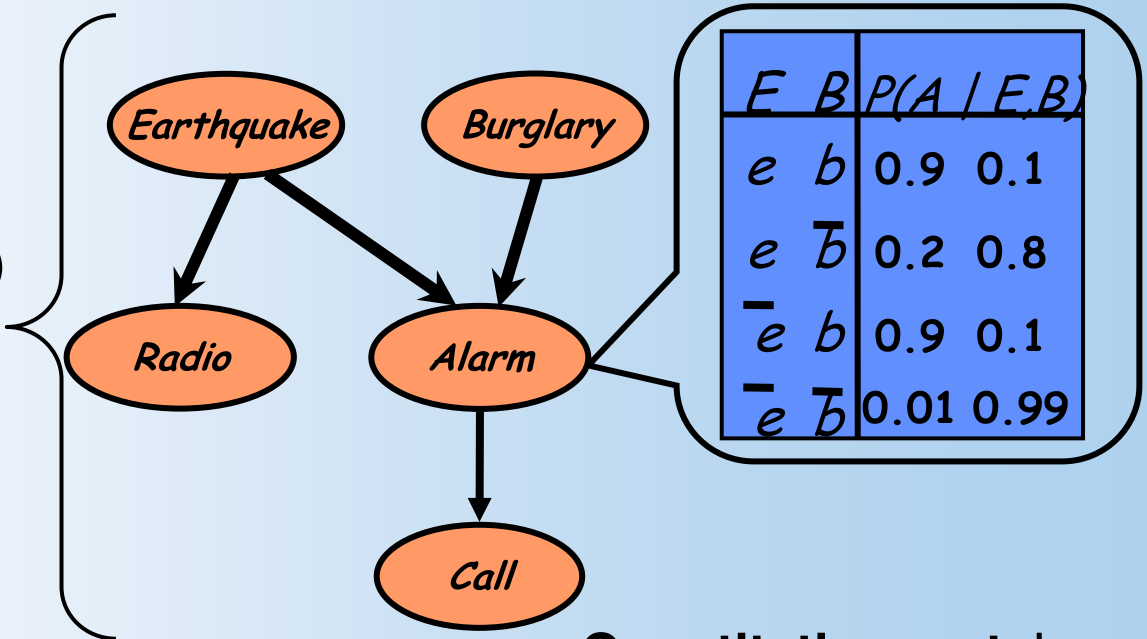
Bayesian networks

Efficient representation of probability distributions via conditional independence

<u>Qualitative part</u>: statistical independence statements

Directed acyclic graph (DAG)

- Nodes - random variables of interest (exhaustive and mutually exclusive states)
- Edges - direct influence



- **<u>Quantitative part</u>**: Local probability models. Set of conditional probability distributions.

Outline

- Introduction
- Bayesian Networks
  - Representation & Semantics
  - » Inference in Bayesian networks
  - Learning Bayesian networks

Inference in Bayesian networks

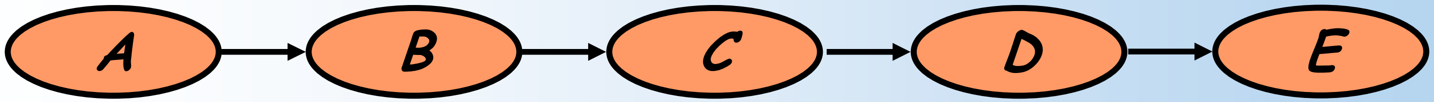
- A Bayesian network represents a probability distribution.
- Can we answer queries about this distribution?

Examples:

- P(Y|Z=z)
- Most probable estimation $MPE(W \mid Z = z) = \arg\max_w P(w, z)$
- Maximum a posteriori $MAP(Y \mid Z = z) = \arg\max_y P(y \mid z)$

Inference in Bayesian networks

- Goal: compute P(E=e,A=a) in the following Bayesian network:

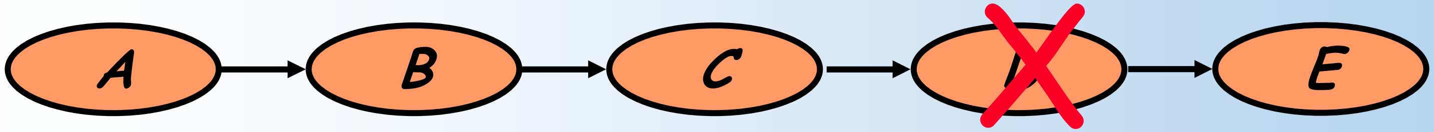


- Using definition of probability, we have

$$P(a,e) = \sum_b \sum_c \sum_d P(a,b,c,d,e)$$

$$= \sum_b \sum_c \sum_d P(a)P(b \mid a)P(c \mid b)P(d \mid c)P(e \mid d)$$

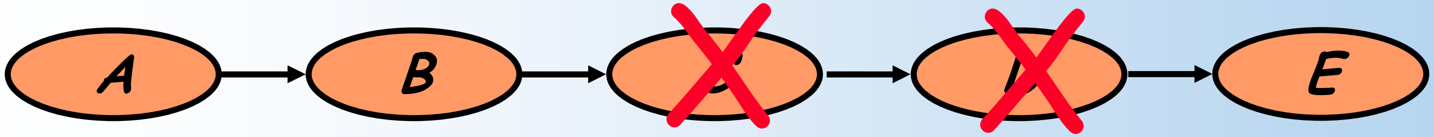
Inference in Bayesian networks



- Eliminating *d*, we get

$$
\begin{aligned}
P(a,e) &= \sum_b \sum_c \sum_d \overbrace{P(a)P(b\mid a)P(c\mid b)}P(d\mid c)P(e\mid d) \\
&= \sum_b \sum_c P(a)P(b\mid a)P(c\mid b)\underbrace{\sum_d P(d\mid c)P(e\mid d)} \\
&= \sum_b \sum_c P(a)P(b\mid a)P(c\mid b)\underbrace{F1(e,c)}_{\text{P(e|c)}}
\end{aligned}
$$

Inference in Bayesian networks



◆ Eliminating *c*, we get

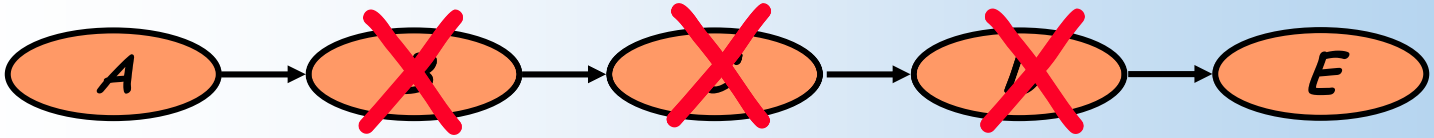
$$P(a,e) = \sum_b \sum_c P(a)P(b \mid a)P(c \mid b)P(e \mid c)$$

$$= \sum_b P(a)P(b \mid a)\underbrace{\sum_c P(c \mid b)P(e \mid c)}$$

$$= \sum_b P(a)P(b \mid a)F2(e,b)$$

p(e|b)

Inference in Bayesian networks



- Finally, we eliminate *b*

$$P(a,e) = \sum_b P(a)P(b \mid a)\,p(e \mid b)$$

$$= P(a)\underbrace{\sum_b P(b \mid a)\,p(e \mid b)}$$

$$= P(a)F3(e,a)$$

p(e|a)

Variable Elimination Algorithm

General idea:

- Write query in the form

$$P(x_1) = \sum_{x_k} \cdots \sum_{x_3} \sum_{x_2} \prod_i P(x_i \mid pa_i)$$

- Iteratively
  - Move all irrelevant terms outside of innermost sum
  - Perform innermost sum, getting a new term
  - Insert the new term into the product

- In case of evidence $P(x_1|\text{evidence } x_j)$, use: $P(x_i \mid x_j) = P(x_i, x_j) / P(x_j)$

Complexity of inference

Naïve exact inference

- **exponential** in the number of variables in the network

Variable elimination complexity

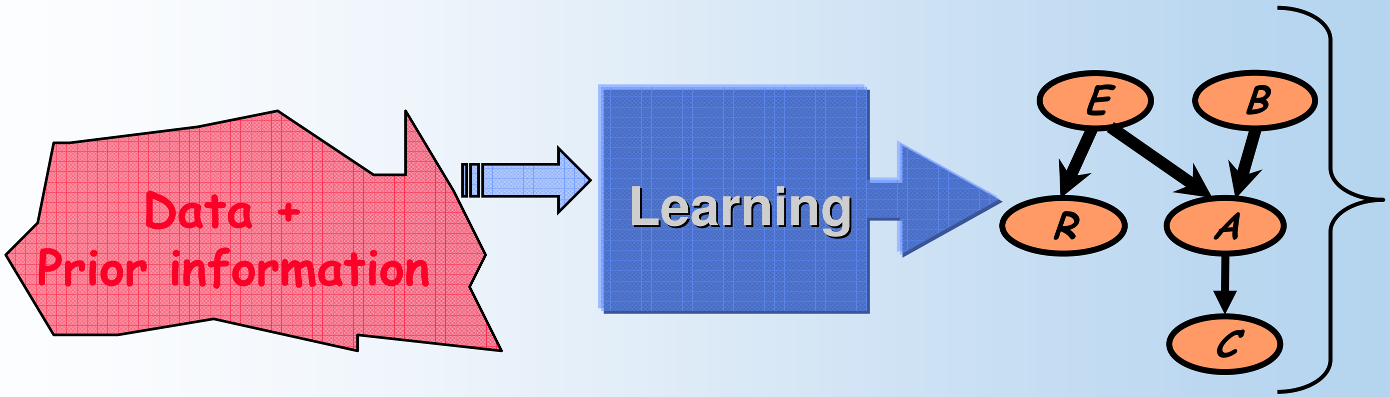
- **exponential** in the size of largest factor
- **polynomial** in the number of variables in the network
- Variable elimination computation depend on order of elimination (many heuristics, e.g., clique tree algorithm).

Outline

- Introduction
- Bayesian Networks
  - Representation & Semantics
  - Inference in Bayesian networks
  - » Learning Bayesian networks
    - Parameter Learning
    - Structure Learning

Learning

- Process
  - **Input:** *dataset* and *prior information*
  - ***Output:*** *Bayesian network*



The Learning Problem

	Known Structure	Unknown Structure
Complete Data	Statistical parametric estimation (closed-form eq.)	Discrete optimization over structures (discrete search)
Incomplete Data	Parametric optimization (EM, gradient descent...)	Combined (Structural EM, mixture models…)

- We will focus on complete data for the rest of the talk
  - The situation with incomplete data is more involved

Outline

- Introduction
- Bayesian Networks
  - Representation & Semantics
  - Inference in Bayesian networks
  - Learning Bayesian networks
    - <u>Parameter Learning</u>
    - Structure Learning

Learning Parameters

- Key concept: the **likelihood function**

$$L(\theta : D) = P(D \mid \theta) = \prod_m P(x[m] \mid \theta)$$

  - measures how the probability of the data changes when we change parameters

- Estimation:
  - MLE: choose parameters that maximize likelihood
  - Bayesian: treat parameters as an unknown quantity, and marginalize over it

MLE principle for Binomial Data

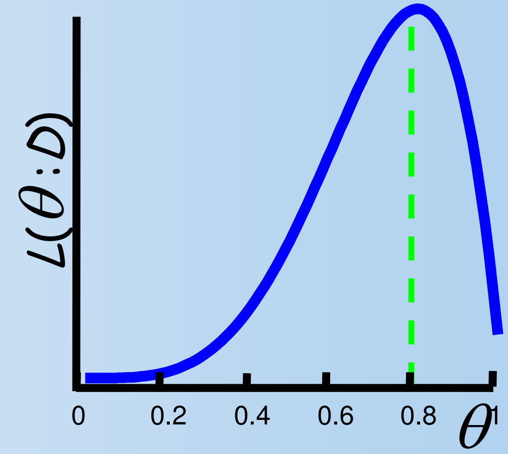
- Data: *H, H, T, H, H*. Θ is the unknown probability P(H).
- Likelihood function: $L(\Theta : D) = \prod_{k=0,1} \theta_k^{N_k}$

$$L(\theta : D) = \theta \cdot \theta \cdot (1 - \theta) \cdot \theta \cdot \theta$$

- <u>Estimation task:</u> Given a sequence of samples x[1], x[2]…x[M], we want to estimate the probability P(H)= θ and P(T)=1-θ.
- MLE principle: choose parameter that maximize the likelihood function.
- Applying the MLE principle we get

$$\hat{\theta} = \frac{N_H}{N_H + N_T}$$

- MLE for $P(X = H)$ is 4/5 = 0.8



MLE principle for Multinomial Data

- Suppose X can have the values 1,2,…,k.
- We want to learn the parameters $\theta_1,\ldots,\theta_{k}$.
- $N_1,\ldots,N_k$ - The number of times each outcome is observed.

- Likelihood function:

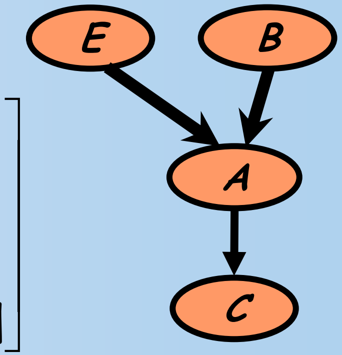
Count of k[th] outcome in D

$$L(\Theta : D) = \prod_{k=1}^{K} \theta_k^{N_k}$$

Probability of k[th] outcome

- The MLE is: $\hat{\theta}_i = \dfrac{N_i}{\sum_{l=1,\ldots,k} N_l}$

MLE principle for Bayesian networks



- Training data has the form: $D = \begin{bmatrix} E[1] & B[1] & A[1] & C[1] \\ . & . & . & . \\ . & . & . & . \\ E[M] & B[M] & A[M] & C[M] \end{bmatrix}$
- Assume i.i.d. samples

$$L(\Theta : D) = \prod_m P(E[m], B[m], A[m], C[m] : \Theta)$$

$$= \prod_m \begin{pmatrix} P(E[m] : \Theta) \\ P(B[m] : \Theta) \\ P(A[m] \mid B[m], E[m] : \Theta) \\ P(C[m] \mid A[m] : \Theta) \end{pmatrix} = \begin{matrix} \prod_m P(E[m] : \Theta) \\ \prod_m P(B[m] : \Theta) \\ \prod_m P(A[m] \mid B[m], E[m] : \Theta) \\ \prod_m P(C[m] \mid A[m] : \Theta) \end{matrix}$$

By definition of network

$$\begin{bmatrix} E[1] & B[1] & A[1] & C[1] \\ . & . & . & . \\ . & . & . & . \\ E[M] & B[M] & A[M] & C[M] \end{bmatrix}$$

$$\begin{bmatrix} E[1] & B[1] & A[1] & C[1] \\ . & . & . & . \\ . & . & . & . \\ E[M] & B[M] & A[M] & C[M] \end{bmatrix}$$

MLE principle for Bayesian networks

- Generalizing for any Bayesian network:

$$L(\Theta : D) = \prod_i \prod_m P(x_i[m] \mid Pa_i[m] : \Theta_i) = \prod_i L_i(\Theta_i : D)$$

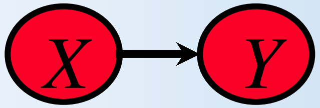
$$L_i(\theta_i : D) = \prod_m P(x_i[m] \mid Pa_i[m] : \theta_i)$$

$$= \prod_{pa_i} \prod_{x_i} P(x_i \mid pa_i : \theta_i)^{N(x_{i,pa_i})} = \prod_{pa_i} \prod_{x_i} \theta_{x_i \mid pa_i}{}^{N(x_{i,pa_i})}$$

- The likelihood decomposes according to the network structure.
- **Decomposition $\Rightarrow$ Independent estimation problems**
  (If the parameters for each family are not related)
- For each value pai of the parent of Xi we get independent multinomial problem.

- The **MLE** is $\hat{\theta}_{x_i \mid pa_i} = \dfrac{N(x_i, pa_i)}{N(pa_i)}$

Continuous (Gaussian) variables

Discrete X 

$Y \mid X \sim N(\mu, \sigma^2)$

$$L(\Theta : D) = \prod_i \prod_m P(x_i[m] \mid Pa_i[m] : \Theta_i) = \prod_i L_i(\Theta_i : D)$$

$$L_i(\theta_i : D) = \prod_m P(x_i[m] \mid Pa_i[m] : \theta_i)$$

$$\boxed{X_i \mid Pa_i \sim N(\mu_{i,Pa_i}, \sigma^2_{i,Pa_i})}$$

- The likelihood decomposes according to the network structure.
- **Decomposition $\Rightarrow$ Independent estimation problems**
  (If the parameters for each family are not related)
- For each value pai of the parent of Xi we get independent maximization problem.

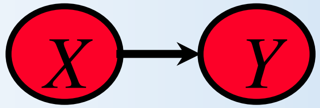
- The **MLE** is

$$\hat{\mu}_{i,Pa_i} = \frac{\sum_{m:Pa_i[m]=pa_i} x_i[m]}{N(pa_i)}$$

$$\hat{\sigma}^2_{i,Pa_i} = \frac{\sum_{m:Pa_i[m]=pa_i} (x_i[m] - \hat{\mu}_{i,Pa_i})^2}{N(pa_i)}$$

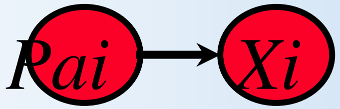
Continuous (Gaussian) variables

$X \sim N(\mu, \sigma_X^2)$



$Y \mid X \sim N(ax + b, \sigma^2)$

$Pa_i \sim N(\mu_{Pa_i}, \sigma_{Pa_i}^2)$



$X_i \mid Pa_i \sim N(a \cdot pa_i + b, \sigma^2)$

$$L(\Theta : D) = \prod_i \prod_m P(x_i[m] \mid Pa_i[m] : \Theta_i) = \prod_i L_i(\Theta_i : D)$$

$$L_i(\theta_i : D) = \prod_m P(x_i[m] \mid Pa_i[m] : \theta_i)$$

- The likelihood decomposes $\Rightarrow$ **Independent estimation problems**

The **MLE** is

$$X_i \mid Pa_i \sim N(a \cdot pa_i + b, \sigma^2)$$

Statistical background - regression

$$Assume\ Y = aX + b + Z, \quad \text{where } Z \sim N(0, \sigma_z^2), \qquad \text{Ind(X,Z)}.$$

σ_x^2, μ_x are population variance and mean of X.

Thus:

1. $\mu_y = \mathrm{a}\mu_x + \mathrm{b}$

$$\boxed{2.\ Y \mid X \sim N(aX + b, \sigma_z^2).}$$

Using least-squares estimation of a and b:

$$\hat{a}, \hat{b} = \arg\min \sum_i (y_i - (ax_i + b))^2$$

Solving max likelihood estimation of Y | X:

$$\hat{a}, \hat{b} = \arg\max \log \mathrm{L(Y \mid X)}$$

$$= \arg\max \frac{n}{2} \ln \frac{1}{2\pi\sigma^2} - \sum_i (y_i - (ax_i + b))^2 / 2\sigma^2$$

$$= \arg\min \sum_i (y_i - (ax_i + b))^2$$

Statistical background - regression

$$\hat{a},\hat{b} = \arg\min \sum_i (y_i - (ax_i + b))^2$$

$$\hat{a} = \frac{\mathrm{cov}(X,Y)}{\sigma_x^2} = \frac{\rho_{xy}\sigma_x\sigma_y}{\sigma_x^2} = \frac{\rho_{xy}\sigma_y}{\sigma_x}$$

$$\hat{b} = \mathrm{E(Y)} - \hat{a}\,\mathrm{E(X)} = \mu_y - \hat{a}\,\mu_x$$

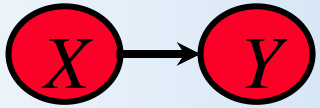
Express Y | X using population parameters $\mu_y \mu_x \sigma_y^2 \sigma_x^2 \rho_{xy}$

$$\mathrm{E}(Y \mid X) = \hat{a}\,X + \hat{b} = \hat{a}\,X + \mu_y - \hat{a}\,\mu_x = \mu_y + \hat{a}(X - \mu_x) = \mu_y + \frac{\rho_{xy}\sigma_y}{\sigma_x}(X - \mu_x)$$

$$Var(Y \mid X) = \ldots = \sigma_y^2(1 - \rho_{xy}^2)$$

Continuous (Gaussian) variables

$X \sim N(\mu, \sigma_X^2)$

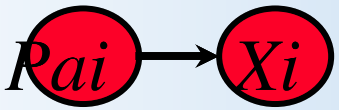


$Y \mid X \sim N(ax + b, \sigma^2)$

$$\mathrm{E}(Y \mid X) = \mu_y + \frac{\rho_{xy}\sigma_y}{\sigma_x}(X - \mu_x)$$

$$Var(Y \mid X) = \sigma_y^2(1 - \rho_{xy}^2)$$

$Pa_i \sim N(\mu_{Pa_i}, \sigma_{Pa_i}^2)$



$X_i \mid Pa_i \sim N(a \cdot pa_i + b, \sigma^2)$

$$\mathrm{E}(X_i \mid Pa_i) = \mu_{X_i} + \frac{\rho_{Pa_i X_i}\sigma_{X_i}}{\sigma_{Pa_i}}(Pa_i - \mu_{Pa_i})$$

$$Var(X_i \mid Pa_i) = \sigma_{X_i}^2(1 - \rho_{Pa_i X_i}^2)$$

Learning Parameters

- Key concept: the **likelihood function**

$$L(\theta : D) = P(D \mid \theta) = \prod_m P(x[m] \mid \theta)$$

  - measures how the probability of the data changes when we change parameters

- Estimation:
  - MLE: choose parameters that maximize likelihood
  - Bayesian: treat parameters as an unknown quantity, and marginalize over it

The Bayesian Approach to learning

- Find the posterior!

$$P(X[M+1]=H \mid D) == \int P(X[M+1]=H \mid \theta, D) P(\theta \mid D) d\theta =$$

$$= \int P(X[M+1]=H \mid \theta) P(\theta \mid D) d\theta =$$

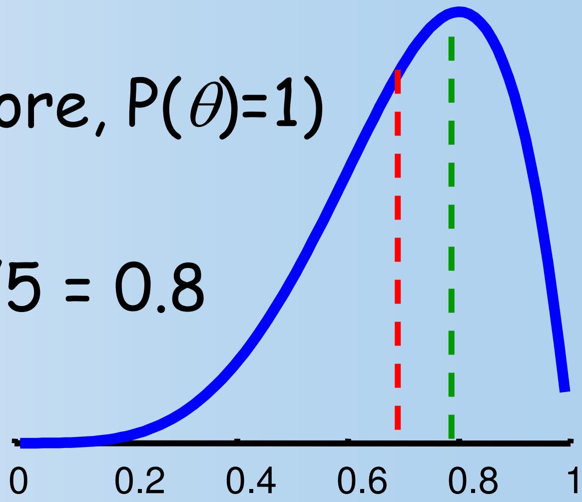
$$\int P(X[M+1]=H \mid \theta) \frac{P(\theta) P(D \mid \theta)}{P(D)} d\theta =$$

$$\frac{\int P(X[M+1]=H \mid \theta) P(\theta) P(D \mid \theta) d\theta}{\int P(\theta) P(D \mid \theta) d\theta} =$$

$$\frac{\int \theta P(\theta) P(D \mid \theta) d\theta}{\int P(\theta) P(D \mid \theta) d\theta}$$

Bayesian approach for Binomial Data

- P(H)= θ.
- **Prior:** uniform for θ in [0,1]. (therefore, P(θ)=1)
- Data: $(N_H, N_T) = (4,1)$
- MLE for $P(X = H)$ is $N_H/(N_H + N_T)$=4/5 = 0.8
- Bayesian prediction is:



$$P(x[M+1] = H \mid D) = \frac{\int \theta P(\theta) P(D \mid \theta) d\theta}{\int P(\theta) P(D \mid \theta) d\theta} =$$

$$\frac{\int \theta \cdot 1 \cdot \theta^{N_H} (1-\theta)^{N_T} d\theta}{\int 1 \cdot \theta^{N_H} (1-\theta)^{N_T} d\theta} = \cdots = \frac{5}{7} = 0.7142\ldots$$

Bayesian approach for Multinomial Data

- Recall that the likelihood function is

$$L(\Theta : D) = \prod_{k=1}^{K} \theta_k^{N_k}$$

- **Dirichlet prior** with hyperparameters $\alpha_1, \dots, \alpha_K$

$$P(\Theta) = \frac{(\sum_{j=1}^{k} \alpha_j - 1)!}{(\alpha_1 - 1)!(\alpha_2 - 1)!....(\alpha_k - 1)!} \prod_{k=1}^{K} \theta_k^{\alpha_k - 1}$$

$\Rightarrow$ the posterior: **Dirichlet** with hyperparameters $\alpha_1 + N_1, \dots, \alpha_K + N_K$

$$P(\Theta | D) = \frac{P(\Theta) P(D | \Theta)}{P(D)} = \frac{c(\alpha)}{P(D)} \prod_{k=1}^{K} \theta_k^{\alpha_k - 1} \prod_{k=1}^{K} \theta_k^{N_k} =$$

$$\frac{(\sum_{j=1}^{k} \alpha_j + N_j - 1)!}{(\alpha_1 + N_1 - 1)!(\alpha_2 + N_2 - 1)!....(\alpha_k + N_k - 1)!} \prod_{k=1}^{K} \theta_k^{\alpha_k + N_k - 1}$$

Bayesian approach for Multinomial Data

- If $P(\Theta)$ is Dirichlet with hyperparameters $\alpha_1,\ldots,\alpha_K$
- The posterior is also Dirichlet:

$P(\Theta|D)$ is Dirichlet with hyperparameters $\alpha_1+N_1 \ldots,\alpha_K+N_k$

and thus we get

$$P(X[M+1]=k \mid D)=\int \theta_k \cdot P(\theta \mid D)d\theta=\frac{\alpha_k+N_k}{\sum_{\ell}(\alpha_\ell+N_\ell)}$$

Learning Parameters for Bayesian networks : Summary

- For multinomials: counts $N(x_i, pa_i)$
- Parameter estimation

$$\hat{\theta}_{x_i|pa_i} = \frac{N(x_i, pa_i)}{N(pa_i)} \qquad \tilde{\theta}_{x_i|pa_i} = \frac{\alpha(x_i, pa_i) + N(x_i, pa_i)}{\alpha(pa_i) + N(pa_i)}$$

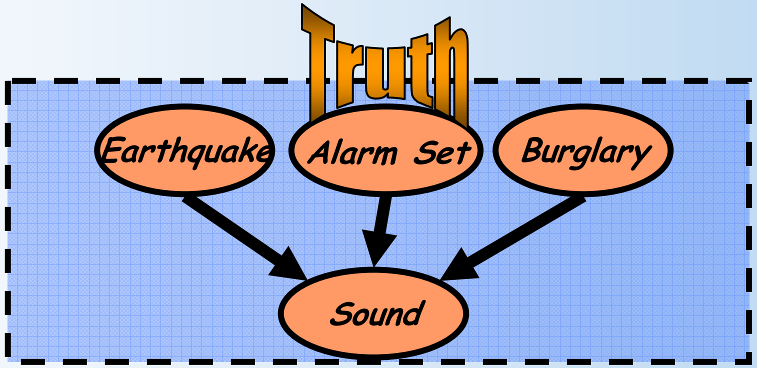
MLE Bayesian (Dirichlet)

- Both can be implemented in an on-line manner by accumulating counts.

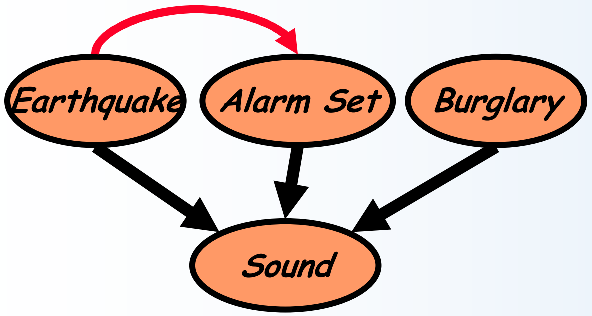
Outline

- Introduction
- Bayesian Networks
  - Representation & Semantics
  - Inference in Bayesian networks
  - Learning Bayesian networks
    - Parameter Learning
    - »Structure Learning

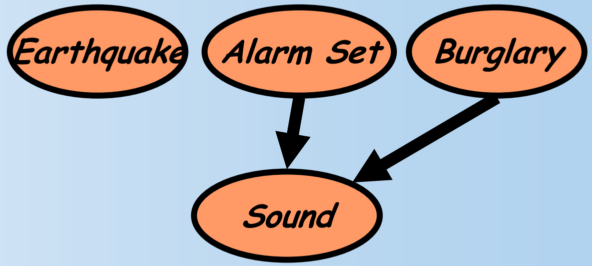
Learning Structure: Motivation



Adding an arc



Missing an arc



Optimization Problem

Input:

- Training data
- Scoring function (including priors)
- Set of possible structures

Output:

- A network (or networks) that maximize the score

Key Property:

- **Decomposability**: the score of a network is a sum of terms.

Scores

For example. The BDE score:

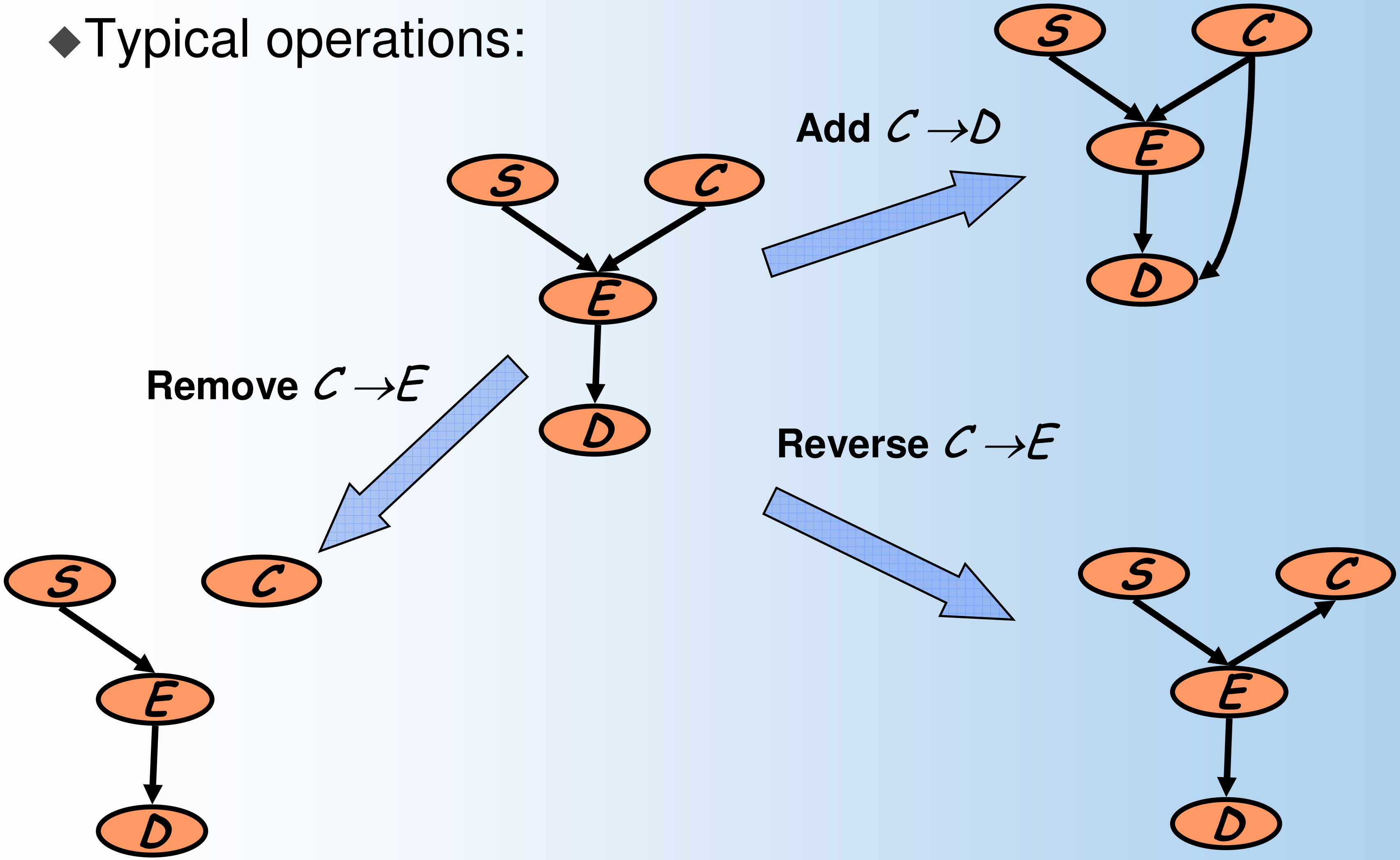
$$Score(G:D) = P(G \mid D) \propto P(D \mid G)P(G)$$
$$= \int P(D \mid G,\theta)P(\theta \mid G)d\theta P(G)$$

When the data is complete, the score is decomposable:

$$Score(G:D) = \sum_i Score(X_i \mid Pa_i^G : D)$$

Heuristic Search (cont.)

- Typical operations:



Heuristic Search

- We address the problem by using heuristic search
- Traverse the space of possible networks, looking for high-scoring structures
- Search techniques:
  - Greedy hill-climbing
  - Simulated Annealing
  - ...

Outline

- Introduction
- Bayesian Networks
  - Representation & Semantics
  - Inference in Bayesian networks
  - Learning Bayesian networks
  - Conclusion
- Applications