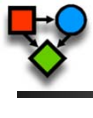


Readings: K&F 3.4, 5.1 ~5.5



Local Probability Models

Lecture 3 – Apr 4, 2011
CSE 515, Statistical Methods, Spring 2011

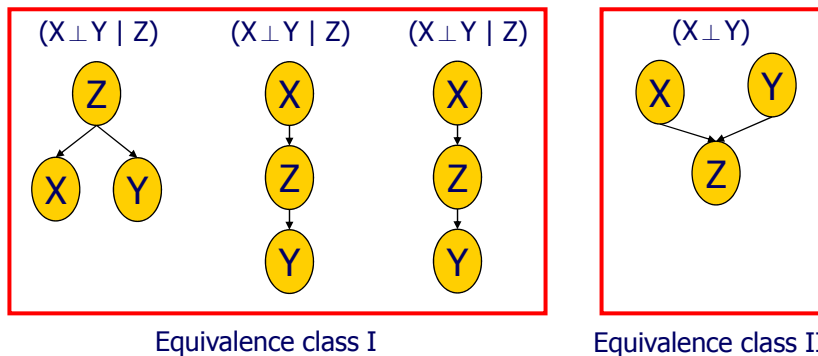
Instructor: Su-In Lee
University of Washington, Seattle

Outline

- Last time
 - Conditional parameterization
 - Bayesian networks
 - Independencies in graphs
 - Local independencies, d-separation, I-equivalence
- Today
 - From distributions to BN graphs
 - Local probability models (CPDs)
 - Tabular
 - Deterministic
 - Context-specific
 - Independence of causal influences
 - Continuous variables

I-equivalence between graphs

- $I(G)$ describe all conditional independencies in G
- Different Bayesian networks can have same Ind.



Two BN graphs G_1 and G_2 are **I-equivalent** if $I(G_1) = I(G_2)$

I-equivalence between graphs

- If P factorizes over a graph in an I-equivalence class
 - P factorizes over all other graphs in the same class
 - P cannot distinguish one I-equivalent graph from another
- Implications for structure learning
 - We cannot find the “correct” structure from within the same equivalent class. -> will revisit later.
- Test for I-equivalence: d-separation

Test for I-equivalence

- **Necessary condition:** same graph skeleton
 - Otherwise, can find active path in one graph but not other
 - But, not sufficient: v-structures
- **Sufficient condition:** same skeleton and v-structures
 - But, not necessary: complete graphs (no independence)

Every two nodes are connected by some edge

- Define $X \rightarrow Z \leftarrow Y$ as **immoral** if X, Y are not directly connected
 - **Necessary and Sufficient:** same skeleton and immoral set of v-structures

Constructing graphs for P

- Can we construct a graph for a distribution P?

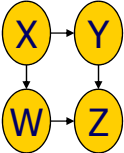
- Any graph which is an I-map for P

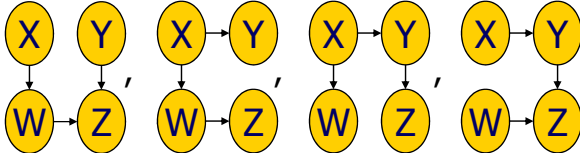
Every two nodes are connected by some edge

- But, this is not so useful: complete graphs
 - Complete graphs imply no independence assumptions
 - Thus, they are I-maps of any distribution

Minimal I-Maps

- A graph G is a **minimal I-map** for P if:
 - G is an I-map for P
 - Removing any edge from G renders it not an I-map for P

- Example: if  is a minimal I-map for P ,

- Then:
- 
- is not I-maps.

Bayesian network definition revisited

- A Bayesian network is a pair (G,P)
 - P factorizes over G
 - P is specified as set of CPDs associated with G 's nodes
 - **Additional requirement: G is a minimal I-map for P**

Constructing minimal I-maps

- Reverse factorization theorem
 - $P(X_1, \dots, X_n) = \prod_{i=1}^n P(X_i | Pa(X_i)) \rightarrow G$ is an I-map of P
- Algorithm for constructing a minimal I-Map
 - Input: (1) fixed **ordering** of nodes X_1, \dots, X_n ; (2) set of independencies that hold in P , denoted by I
 - For each X_i ,
 - Select parents of X_i as minimal subset of $\{X_1, \dots, X_{i-1}\}$, such that $(X_i \perp \{X_1, \dots, X_{i-1} - Pa(X_i)\} \mid Pa(X_i)) \in I$
- (Outline of) Proof of minimal I-map
 - I-map since the factorization above holds by construction
 - Minimal since by construction, removing one edge destroys the factorization

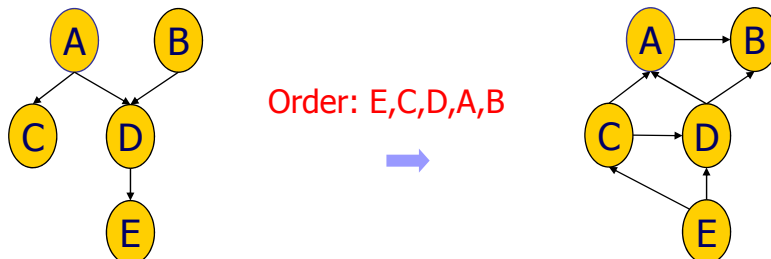
9

Non-uniqueness of minimal I-map

- Applying the same I-Map construction process with **different orders** can lead to different structures

Assume: $I(G) = I(P)$

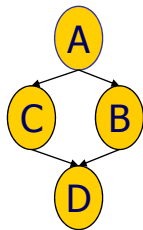
Choosing order: Drastic effects on complexity of minimal I-Map graph
Heuristic: Use causal order



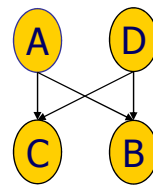
Different independence assumptions (different skeletons, e.g., $(A \perp B)$ holds on left)

Perfect maps

- G is a **perfect map (P-Map)** for P if $I(P)=I(G)$
- Does every distribution have a P-map?
 - No: independencies may be encoded in CPD ($X \perp Y|Z=1$)
 - No: some structures cannot be represented in a BN
 - Independencies in P: ($A \perp D | B,C$), and ($B \perp C | A,D$)



$(B \perp C | A,D)$ does not hold



$(A \perp D)$ also holds

CSE 515 – Statistical Methods – Spring 2011

11

Finding a perfect map

- If P has a P-map, can we find it?
 - Not uniquely, since I-equivalent graphs are indistinguishable
 - Thus, represent I-equivalent graphs and return it
- Recall I-Equivalence
 - **Necessary and Sufficient:** same skeleton and immoral set of v-structures
- Finding P-maps
 - Step I: Find skeleton
 - Step II: Find immoral set of v-structures
 - Step III: Direct constrained edges
 - Detailed algorithm: please read the textbook

CSE 515 – Statistical Methods – Spring 2011

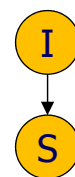
12

Outline

- Last time
 - Conditional parameterization
 - Bayesian networks
 - Independencies in graphs
 - Local independencies, d-separation, I-equivalence
- Today
 - From distributions to BN graphs
 - Local probability models
 - Conditional Probability Distributions (CPDs)
 - Table CPDs
 - Deterministic CPDs
 - Context-specific CPDs
 - Independence of causal influences
 - Continuous variables

Table CPDs

- Entry for each joint assignment of X and $\text{Pa}(X)$
- For each $\text{pa}_x : \sum_{x \in \text{Val}(X)} P(x | \text{pa}_x) = 1$
- Most general representation
- Represents every discrete CPD



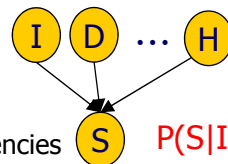
$P(I)$

I	
i^0	i^1
0.7	0.3

$P(S|I)$

I	S	
	s^0	s^1
i^0	0.95	0.05
i^1	0.2	0.8

- Limitations
 - Cannot model continuous RVs
 - # parameters exponential in $|\text{Pa}(X)|$
→ Cannot model large in-degree dependencies
 - Ignores structure within the CPD



$P(S|I,D,\dots,H)$

I,D,...H	S	
	s^0	s^1
$I^0 d^0 \dots h^0$	0.95	0.05
$I^0 d^0 \dots h^1$	0.2	0.8
⋮	⋮	⋮

- How to overcome the limitations?

Structured CPDs

- **Key idea:** reduce # parameters by modeling $P(X|Pa_X)$ without explicitly determining all entries of the joint
 - We use constraints instead.
 - Lose expressive power (cannot represent every CPD)
- Many ways depending on the constraints
 - Deterministic, tree-structured, rule-based CPDs

Deterministic CPDs

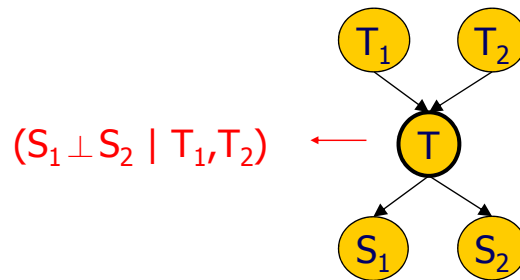
- There is a function $f: \text{Val}(Pa_X) \rightarrow \text{Val}(X)$ such that

$$P(x | pa_x) = \begin{cases} 1 & x = f(pa_x) \\ 0 & \text{otherwise} \end{cases}$$

- Example functions
 - OR, XOR, AND, NAND functions
 - $Z = X+Y$ (continuous variables)

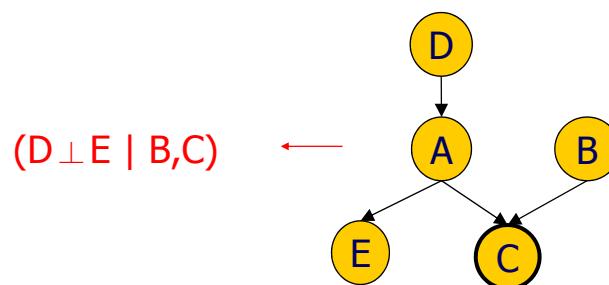
Deterministic CPDs: example I

- Induce **additional conditional independencies**
- Example: T is **any** deterministic function of T_1, T_2



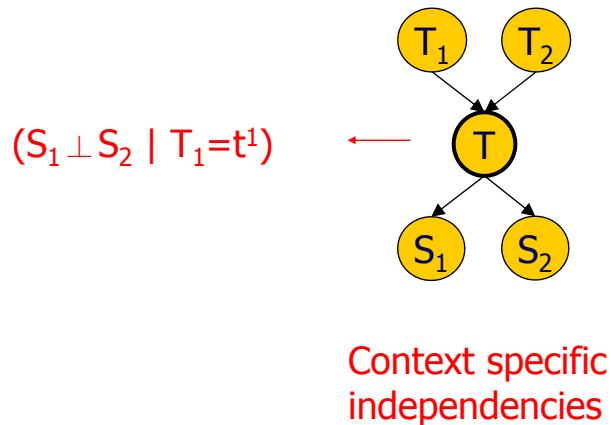
Deterministic CPDs: example II

- Induce **additional conditional independencies**
- Example: C is an **XOR** deterministic function of A,B



Deterministic CPDs: example III

- Induce **additional conditional independencies**
- Example: T is an **OR** deterministic function of T_1, T_2



19

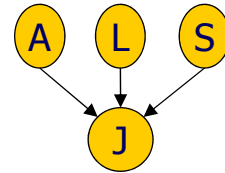
Context specific independencies

- Let X, Y, Z be disjoint random variable sets
- Let C be a set of variables and $c \in \text{Val}(C)$
- X and Y are **contextually independent** given Z and c , denoted $(X \perp_c Y \mid Z, C=c)$ if:

$$P(X \mid Y, Z, c) = P(X \mid Z, c) \text{ whenever } P(Y, Z, c) > 0$$

Tree CPDs

- A natural representation for capturing common elements in CPDs
- Uses a decision tree



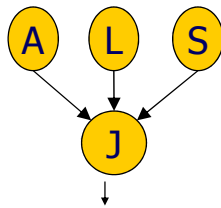
- Example – job offer

- Job offer Val(J)={yes,no}
- Apply in time Val(A)={yes,no}
- Quality of Letter Val(L)={good,bad}
- GPA Score Val(S)={high,low}

$P(J|A,L,S)$

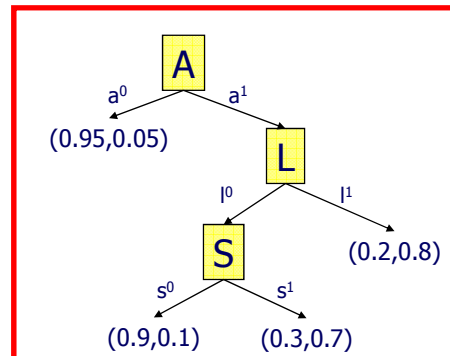
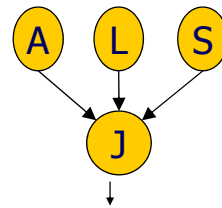
<i>A</i>	<i>L</i>	<i>S</i>	<i>J</i>	
			j^0	j^1
a^0	l^0	s^0	0.95	0.05
a^0	l^0	s^1	0.95	0.05
a^0	l^1	s^0	0.95	0.05
a^0	l^1	s^1	0.95	0.05
a^1	l^0	s^0	0.9	0.1
a^1	l^0	s^1	0.3	0.7
a^1	l^1	s^0	0.2	0.8
a^1	l^1	s^1	0.2	0.8

Tree CPDs: example



<i>A</i>	<i>L</i>	<i>S</i>	<i>J</i>	
			j^0	j^1
a^0	l^0	s^0	0.95	0.05
a^0	l^0	s^1	0.95	0.05
a^0	l^1	s^0	0.95	0.05
a^0	l^1	s^1	0.95	0.05
a^1	l^0	s^0	0.9	0.1
a^1	l^0	s^1	0.3	0.7
a^1	l^1	s^0	0.2	0.8
a^1	l^1	s^1	0.2	0.8

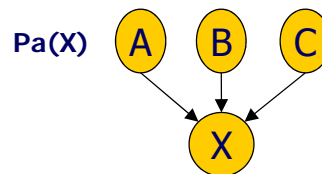
8 parameters



A binary tree + 4 parameters

Rule CPDs

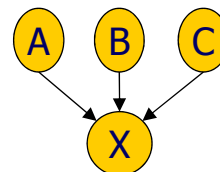
- A **rule** r is a pair $(c;p)$ where c is an assignment to a subset of variables C and $p \in [0,1]$.
 - Let $\text{Scope}[r]=C$
- A **rule-based** CPD $P(X|\text{Pa}(X))$ is a set of rules R s.t.
 - For each rule $r \in R \rightarrow \text{Scope}[r] \in \{X\} \cup \text{Pa}(X)$
 - For each assignment (x,u) to $\{X\} \cup \text{Pa}(X)$ we have **exactly one rule** $(c;p) \in R$ such that c is compatible with (x,u) .
Then, we have $P(X=x | \text{Pa}(X)=u) = p$



Rule CPDs

- Example: Let X be a variable with $\text{Pa}(X) = \{A,B,C\}$

- r1: $(a^1, b^1, x^0; 0.1)$
- r2: $(a^0, c^1, x^0; 0.2)$
- r3: $(b^0, c^0, x^0; 0.3)$
- r4: $(a^1, b^0, c^1, x^0; 0.4)$
- r5: $(a^0, b^1, c^0; 0.5)$
- r6: $(a^1, b^1, x^1; 0.9)$
- r7: $(a^0, c^1, x^1; 0.8)$
- r8: $(b^0, c^0, x^1; 0.7)$
- r9: $(a^1, b^0, c^1, x^1; 0.6)$

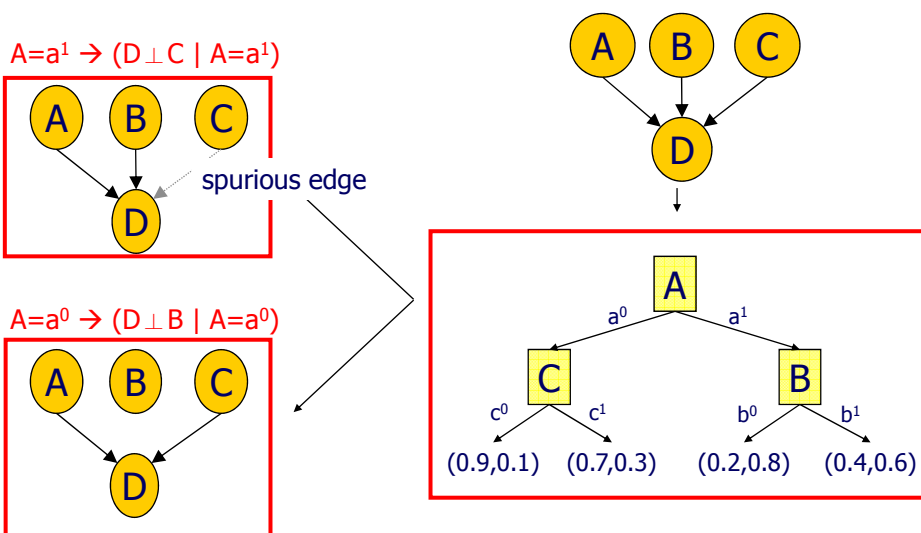


- Note: each assignment maps to exactly one rule
- Rules cannot always be represented compactly within tree CPDs

Tree CPDs and Rule CPDs

- Can represent every discrete function
- Can be easily learned and dealt with in inference
- But, some functions are not represented compactly
 - XOR in tree CPDs: cannot split in one step on a^0, b^1 and a^1, b^0
- Alternative representations exist
 - Complex logical rules

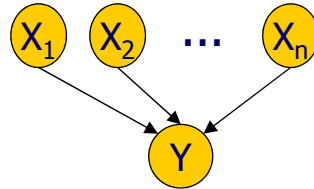
Context specific independencies



Reasoning by cases implies that $(B \perp C \mid A, D)$

Independence of causal influence

- Causes: X_1, \dots, X_n
- Effect: Y



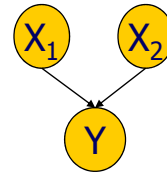
- General case: Y has a complex dependency on X_1, \dots, X_n
- Common case
 - Each X_i influences Y separately
 - Influence of X_1, \dots, X_n is combined to an overall influence on Y

CSE 515 – Statistical Methods – Spring 2011

27

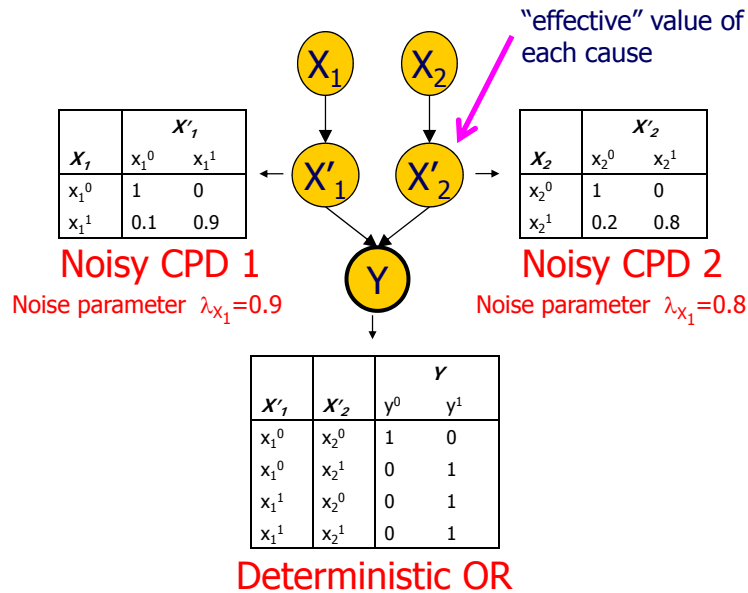
Example 1: Noisy OR

- Two independent causal mechanisms by X_1, X_2
- Key assumptions
 - "OR": $Y=y^1$ cannot happen unless one of X_1, X_2 occurs
 - "Noisy": each causal mechanism is noisy.
 - Independence of the causal mechanisms by X_1, X_2
 - $P(Y=y^0 | X_1=x_1^1, X_2=x_2^1) = P(Y=y^0 | X_1=x_1^0, X_2=x_2^1) P(Y=y^0 | X_1=x_1^1, X_2=x_2^0)$



X_1	X_2	Y	
		y^0	y^1
x_1^0	x_2^0	1	0
x_1^0	x_2^1	0.2	0.8
x_1^1	x_2^0	0.1	0.9
x_1^1	x_2^1	0.02	0.98

Noisy OR: elaborate representation



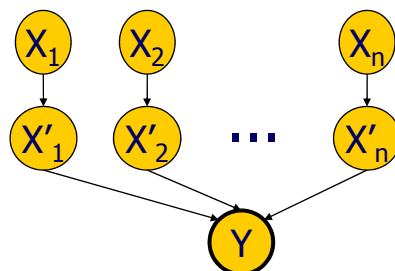
29

Noisy OR: general case

- Y is a binary variable with n binary parents X_1, \dots, X_n
- CPD $P(Y | X_1, \dots, X_n)$ is a noisy OR if there are (n+1) noise parameters $\lambda_0, \lambda_1, \dots, \lambda_n$ such that

$$P(Y = y^0 | X_1, \dots, X_n) = (1 - \lambda_0) \prod_{i: X_i = x_i^1} (1 - \lambda_i)$$

$$P(Y = y^1 | X_1, \dots, X_n) = 1 - \left[(1 - \lambda_0) \prod_{i: X_i = x_i^1} (1 - \lambda_i) \right]$$



$$i \neq j: (X_i \perp X_j | Y = y^0)$$

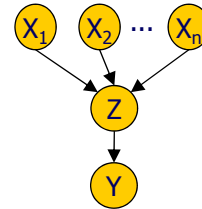
Generalized linear models (GLMs)

- A soft version of a linear threshold function

- Example: **logistic CPD**

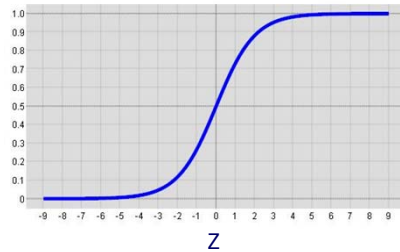
- Binary variables X_1, \dots, X_n, Y
- (Additive) combined effect of X 's $z = w_o + \sum_{i=1}^n w_i \mathbf{1}(X_i = 1)$
- Logistic CPD:

$$P(Y = y^1 | X_1, \dots, X_k) = \text{logit} \left(w_o + \sum_{i=1}^n w_i \mathbf{1}(X_i = 1) \right)$$



Logit function (smooth step function)

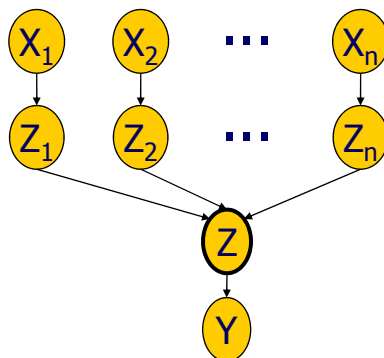
$$\text{logit}(z) = \frac{e^z}{1 + e^z}$$



31

General formulation

- Let Y be a random variable with parents X_1, \dots, X_n
- The CPD $P(Y | X_1, \dots, X_n)$ exhibits **independence of causal influence (ICI)** if it can be described via a network structure shown below:



Logistic
 $Z_i = w_i \mathbf{1}(X_i = 1)$
 $Z = \sum Z_i$
 $Y = \text{logit}(Z)$

Noisy OR
 Z_i has noise model
 Z is an OR function
 Y is the identity CPD

- The CPD $P(Z | Z_1, \dots, Z_n)$ is deterministic
- **Key advantage:** $O(n)$ parameters

32

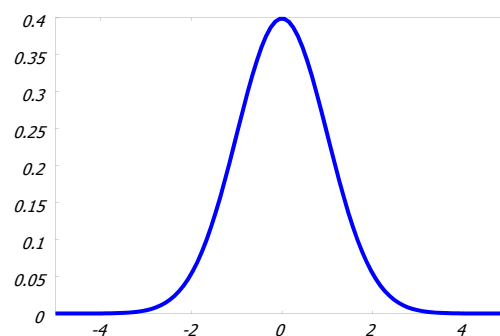
Continuous variables

- One solution: discretize
 - Often requires too many value states
 - Loses domain structure
- Other solutions: use continuous function for $P(X|\text{Pa}(X))$
 - Can combine continuous and discrete variables, resulting in **hybrid networks**
 - Inference and learning may become more difficult

Gaussian density functions

- Among the most common continuous representations
- **Univariate** case:

$$P(X) \sim N(\mu, \sigma^2) \quad \text{if} \quad p(X) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$



Multivariate Gaussian density functions

- A **multivariate Gaussian** distribution over X_1, \dots, X_n has
 - $n \times 1$ mean vector μ
 - $n \times n$ positive definite **covariance matrix** Σ
(positive definite: $\forall x \in \mathfrak{R}^n : x^T \Sigma x > 0$)
 - Joint density function:

$$p(x) = \frac{1}{(2\pi)^{n/2} |\Sigma|^{1/2}} \exp\left[-\frac{1}{2}(x - \mu)^T \Sigma^{-1}(x - \mu)\right]$$

- $\mu_i = E[X_i]$
- $\Sigma_{ii} = \text{Var}[X_i]$
- $\Sigma_{ij} = \text{Cov}[X_i, X_j] = E[X_i X_j] - E[X_i]E[X_j]$ ($i \neq j$)

Gaussian density functions

- Marginal distributions are easy to compute

$$P(\mathbf{X}, \mathbf{Y}) = N\left(\begin{bmatrix} \mu_X \\ \mu_Y \end{bmatrix} ; \begin{bmatrix} \Sigma_{XX} & \Sigma_{XY} \\ \Sigma_{YX} & \Sigma_{YY} \end{bmatrix}\right)$$

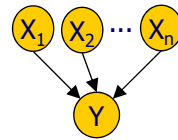


$$P(\mathbf{X}) = N(\mu_X ; \Sigma_{XX})$$

- Independencies can be determined from parameters
 - If $\mathbf{X} = X_1, \dots, X_n$ have a joint normal distribution $N(\mu; \Sigma)$ then $(X_i \perp X_j)$ iff $\Sigma_{ij} = 0$ (for $i \neq j$)
 - Does not hold in general for non-Gaussian distributions

Linear Gaussian CPDs

- Y is a continuous variable with parents X_1, \dots, X_n
- Y has a **linear Gaussian model** if it can be described using parameters β_0, \dots, β_n and σ^2 such that
 - $P(Y | x_1, \dots, x_n) = N(\beta_0 + \beta_1 x_1 + \dots + \beta_n x_n; \sigma^2)$
 - Vector notation: $P(Y | \mathbf{x}) = N(\beta_0 + \beta^T \mathbf{x}; \sigma^2)$
- Pros
 - Simple
 - Captures many interesting dependencies
- Cons
 - Fixed variance (variance cannot depend on parents values)



Linear Gaussian Bayesian network

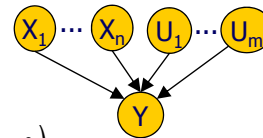
- A **linear Gaussian Bayesian network** is a Bayesian network where
 - All variables are continuous
 - All of the CPDs are linear Gaussians
- **Key result:** linear Gaussian models are equivalent to multivariate Gaussian density functions
- Proof?

Hybrid models

- Models of continuous and discrete variables
 - Continuous variables with discrete parents
 - Discrete variables with continuous parents

- Conditional Linear Gaussians**

- Y continuous variable
- $\mathbf{X} = \{X_1, \dots, X_n\}$ continuous parents
- $\mathbf{U} = \{U_1, \dots, U_m\}$ discrete parents
- $\forall \mathbf{u} \in U : P(Y | \mathbf{u}, \mathbf{x}) = N(a_{u,0} + \sum_{i=1}^n a_{u,i} x_i; \sigma_u^2)$



- A **conditional Linear Bayesian network** is one where
 - Discrete variables have only discrete parents
 - Continuous variables have only CLG CPDs

Hybrid models

- Continuous parents for discrete children

- Threshold models
$$P(Y = y^1 | x) = \begin{cases} 0.9 & x < 10 \\ 0.05 & \text{otherwise} \end{cases}$$

- Linear sigmoid (logit function)

$$P(Y = y^1 | x_1, \dots, x_k) = \text{logit}(w_o + \sum_{i=1}^n w_i x_i)$$

Summary: CPD models

- Deterministic functions
- Context specific dependencies
 - Tree CPDs
 - Rule CPDs
- Independence of causal influence
 - Noisy OR
 - Logistic function
- CPDs capture additional domain structure

Acknowledgement

- These lecture notes were generated based on the slides from Prof Eran Segal.