

# Graphical models

Sunita Sarawagi  
IIT Bombay

<http://www.cse.iitb.ac.in/~sunita>

# Probabilistic modeling

- Given: several variables:  $x_1, \dots, x_n$ ,  $n$  is large.
- Task: build a joint distribution function  $\Pr(x_1, \dots, x_n)$
- Goal: Answer several kind of projection queries on the distribution

# Probabilistic modeling

- Given: several variables:  $x_1, \dots, x_n$ ,  $n$  is large.
- Task: build a joint distribution function  $\Pr(x_1, \dots, x_n)$
- Goal: Answer several kind of projection queries on the distribution
- Basic premise
  - ▶ Explicit joint distribution is dauntingly large
  - ▶ Queries are simple **marginals** (sum or max) over the joint distribution.

## Example

- Variables are attributes are people.

Age	Income	Experience	Degree	Location
10 ranges	7 scales	7 scales	3 scales	30 places

- An explicit joint** distribution over all columns not tractable:  
number of combinations:  $10 \times 7 \times 7 \times 3 \times 30 = 44100$ .
- Queries: Estimate fraction of people with
  - ▶ Income  $>$  200K and Degree="Bachelors",
  - ▶ Income  $<$  200K, Degree="PhD" and experience  $>$  10 years.
  - ▶ Many, many more.

# Alternatives to an explicit joint distribution

- Assume all columns are independent of each other: **bad assumption**

# Alternatives to an explicit joint distribution

- Assume all columns are independent of each other: **bad assumption**
- Use data to detect pairs of highly correlated column pairs and estimate their pairwise frequencies
  - ▶ Many highly correlated pairs  
income ~~⊥~~ age, income ~~⊥~~ experience, age ~~⊥~~ experience
  - ▶ **Ad hoc methods of combining these into a single estimate**

# Alternatives to an explicit joint distribution

- Assume all columns are independent of each other: **bad assumption**
- Use data to detect pairs of highly correlated column pairs and estimate their pairwise frequencies
  - ▶ Many highly correlated pairs  
income  $\not\perp$  age, income  $\not\perp$  experience, age  $\not\perp$  experience
  - ▶ **Ad hoc methods of combining these into a single estimate**
- Go beyond pairwise correlations: conditional independencies
  - ▶ income  $\not\perp$  age, but income  $\perp\!\!\!\perp$  age | experience
  - ▶ experience  $\perp\!\!\!\perp$  degree, but experience  $\not\perp$  degree | income

Graphical models make explicit an efficient joint distribution from these independencies

# Graphical models

Model joint distribution over **several** variables as a product of smaller factors that is

- ① *Intuitive* to represent and visualize
  - ▶ Graph: represent structure of dependencies
  - ▶ Potentials over subsets: quantify the dependencies
- ② *Efficient* to query
  - ▶ given values of any variable subset, reason about probability distribution of others.
  - ▶ many efficient exact and approximate inference algorithms



# Graphical models

Model joint distribution over **several** variables as a product of smaller factors that is

- ① *Intuitive* to represent and visualize
  - ▶ Graph: represent structure of dependencies
  - ▶ Potentials over subsets: quantify the dependencies
- ② *Efficient* to query
  - ▶ given values of any variable subset, reason about probability distribution of others.
  - ▶ many efficient exact and approximate inference algorithms

Graphical models = graph theory + probability theory.

# Graphical models in use

- Roots in statistical physics for modeling interacting atoms in gas and solids [ 1900]
- Early usage in genetics for modeling properties of species [ 1920]
- AI: expert systems ( 1970s-80s)
- Now many new applications:
  - ▶ Error Correcting Codes: Turbo codes, impressive success story (1990s)
  - ▶ Robotics and Vision: image denoising, robot navigation.
  - ▶ Text mining: information extraction, duplicate elimination, hypertext classification, help systems
  - ▶ Bio-informatics: Secondary structure prediction, Gene discovery
  - ▶ Data mining: probabilistic classification and clustering.

# Part I: Outline

## 1 Representation

- Directed graphical models: Bayesian networks
- Undirected graphical models

## 2 Inference Queries

- Exact inference on chains
- Variable elimination on general graphs
- Junction trees

## 3 Approximate inference

- Generalized belief propagation
- Sampling: Gibbs, Particle filters

## 4 Constructing a graphical model

- Graph Structure
- Parameters in Potentials

## 5 References

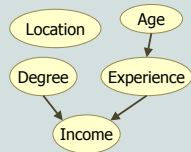
# Representation

Structure of a graphical model: Graph + Potential

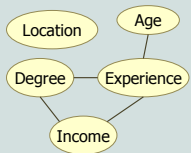
## Graph

- Nodes: variables  $\mathbf{x} = x_1, \dots, x_n$ 
  - ▶ Continuous: Sensor temperatures, income
  - ▶ Discrete: Degree (one of Bachelors, Masters, PhD), Levels of age, Labels of words
- Edges: direct interaction
  - ▶ Directed edges: Bayesian networks
  - ▶ Undirected edges: Markov Random fields

### Directed



### Undirected



# Representation

## Potentials: $\psi_c(\mathbf{x}_c)$

- Scores for assignment of values to subsets  $c$  of directly interacting variables.
- Which subsets? What do the potentials mean?
  - ▶ Different for directed and undirected graphs

# Representation

## Potentials: $\psi_c(\mathbf{x}_c)$

- Scores for assignment of values to subsets  $c$  of directly interacting variables.
- Which subsets? What do the potentials mean?
  - ▶ Different for directed and undirected graphs

## Probability

Factorizes as product of potentials

$$\Pr(\mathbf{x} = x_1, \dots, x_n) \propto \prod \psi_S(\mathbf{x}_S)$$

# Directed graphical models: Bayesian networks

- Graph  $G$ : directed acyclic
  - ▶ Parents of a node:  $\text{Pa}(x_i) =$  set of nodes in  $G$  pointing to  $x_i$

# Directed graphical models: Bayesian networks

- Graph  $G$ : directed acyclic
  - ▶ Parents of a node:  $\text{Pa}(x_i) =$  set of nodes in  $G$  pointing to  $x_i$



# Directed graphical models: Bayesian networks

- Graph  $G$ : directed acyclic
  - ▶ Parents of a node:  $\text{Pa}(x_i)$  = set of nodes in  $G$  pointing to  $x_i$
- Potentials: defined at each node in terms of its parents.

$$\psi_i(x_i, \text{Pa}(x_i)) = \Pr(x_i | \text{Pa}(x_i))$$

# Directed graphical models: Bayesian networks

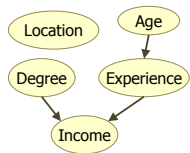
- Graph  $G$ : directed acyclic
  - ▶ Parents of a node:  $\text{Pa}(x_i)$  = set of nodes in  $G$  pointing to  $x_i$
- Potentials: defined at each node in terms of its parents.

$$\psi_i(x_i, \text{Pa}(x_i)) = \Pr(x_i | \text{Pa}(x_i))$$

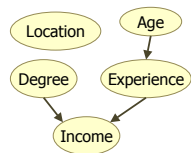
- Probability distribution

$$\Pr(x_1 \dots x_n) = \prod_{i=1}^n \Pr(x_i | \text{pa}(x_i))$$

# Example of a directed graph



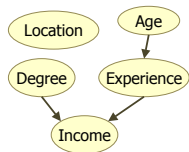
# Example of a directed graph



$$\psi_1(L) = \Pr(L)$$

<b>NY</b>	<b>CA</b>	<b>London</b>	<b>Other</b>
0.2	0.3	0.1	0.4

# Example of a directed graph



$$\psi_1(L) = \Pr(L)$$

NY	CA	London	Other
0.2	0.3	0.1	0.4

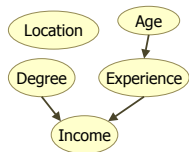
$$\psi_2(A) = \Pr(A)$$

20-30	30-45	> 45
0.3	0.4	0.3

or, a Gaussian distribution

$$(\mu, \sigma) = (35, 10)$$

# Example of a directed graph



$$\psi_1(L) = \Pr(L)$$

NY	CA	London	Other
0.2	0.3	0.1	0.4

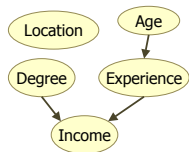
$$\psi_2(A) = \Pr(A)$$

20-30	30-45	> 45
0.3	0.4	0.3

or, a Gaussian distribution

$$(\mu, \sigma) = (35, 10)$$

# Example of a directed graph



$$\psi_1(L) = \Pr(L)$$

NY	CA	London	Other
0.2	0.3	0.1	0.4

$$\psi_2(A) = \Pr(A)$$

20-30	30-45	> 45
0.3	0.4	0.3

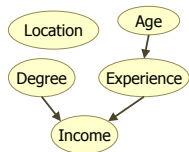
or, a Gaussian distribution

$$(\mu, \sigma) = (35, 10)$$

$$\psi_2(E, A) = \Pr(E|A)$$

	0-10	10-15	> 15
20-30	0.9	0.1	0
30-45	0.4	0.5	0.1
> 45	0.1	0.1	0.8

# Example of a directed graph



$$\psi_1(L) = \Pr(L)$$

NY	CA	London	Other
0.2	0.3	0.1	0.4

$$\psi_2(A) = \Pr(A)$$

20-30	30-45	> 45
0.3	0.4	0.3

or, a Gaussian distribution  
 $(\mu, \sigma) = (35, 10)$

$$\psi_2(E, A) = \Pr(E|A)$$

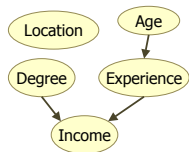
	0-10	10-15	> 15
20-30	0.9	0.1	0
30-45	0.4	0.5	0.1
> 45	0.1	0.1	0.8

$$\psi_2(I, E, D) = \Pr(I|D, A)$$

3 dimensional table, or a  
histogram approximation.



# Example of a directed graph



$$\psi_1(L) = \Pr(L)$$

NY	CA	London	Other
0.2	0.3	0.1	0.4

$$\psi_2(A) = \Pr(A)$$

20-30	30-45	> 45
0.3	0.4	0.3

or, a Gaussian distribution  
 $(\mu, \sigma) = (35, 10)$

$$\psi_2(E, A) = \Pr(E|A)$$

	0-10	10-15	> 15
20-30	0.9	0.1	0
30-45	0.4	0.5	0.1
> 45	0.1	0.1	0.8

$$\psi_2(I, E, D) = \Pr(I|D, A)$$

3 dimensional table, or a histogram approximation.

## Probability distribution

$$\text{Pa}(\mathbf{x} = L, D, I, A, E) = \Pr(L) \Pr(D) \Pr(A) \Pr(E|A) \Pr(I|D, E)$$

## Conditional Independencies

- Given three sets of variables  $X$ ,  $Y$ ,  $Z$ , set  $X$  is conditionally independent of  $Y$  given  $Z$  ( $X \perp\!\!\!\perp Y|Z$ ) iff

$$\Pr(X|Y, Z) = \Pr(X|Z)$$

## Conditional Independencies

- Given three sets of variables  $X$ ,  $Y$ ,  $Z$ , set  $X$  is conditionally independent of  $Y$  given  $Z$  ( $X \perp\!\!\!\perp Y|Z$ ) iff

$$\Pr(X|Y, Z) = \Pr(X|Z)$$

- Local conditional independencies in BN: for each  $x_i$

$$x_i \perp\!\!\!\perp ND(x_i)|Pa(x_i)$$

# Conditional Independencies

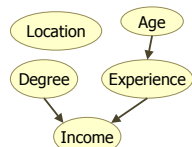
- Given three sets of variables  $X$ ,  $Y$ ,  $Z$ , set  $X$  is conditionally independent of  $Y$  given  $Z$  ( $X \perp\!\!\!\perp Y|Z$ ) iff

$$\Pr(X|Y, Z) = \Pr(X|Z)$$

- Local conditional independencies in BN: for each  $x_i$

$$x_i \perp\!\!\!\perp ND(x_i)|Pa(x_i)$$

- $L \perp\!\!\!\perp E, D, A, I$
- $A \perp\!\!\!\perp L, D$
- $E \perp\!\!\!\perp L, D|A$
- $I \perp\!\!\!\perp A|E, D$



# CI and Factorization

## Theorem

*Local CI  $\implies$  Factorization*

## Proof.

- $x_1, x_2, \dots, x_n$  topographically ordered (parents before children).
- Local CI:  $\Pr(x_i | x_1, \dots, x_{i-1}) = \Pr(x_i | Pa(x_i))$
- Chain rule:

$$\Pr(x_1, \dots, x_n) = \prod_i \Pr(x_i | x_1, \dots, x_{i-1}) = \prod_i \Pr(x_i | Pa(x_i))$$



## Global CIs in a BN

Three sets of variables  $X, Y, Z$ . If  $Z$  **d-separates**  $X$  from  $Y$  in BN then,  $X \perp\!\!\!\perp Y|Z$ .

In a directed graph  $H$ ,  $Z$  d-separates  $X$  from  $Y$  if all paths  $P$  from any  $X$  to  $Y$  is blocked by  $Z$ .

A path  $P$  is blocked by  $Z$  when

- 1  $x_1 \rightarrow x_2 \rightarrow \dots x_k$  and  $x_i \in Z$
- 2  $x_1 \leftarrow x_2 \leftarrow \dots x_k$  and  $x_i \in Z$
- 3  $x_1 \dots \leftarrow x_i \rightarrow \dots x_k$  and  $x_i \in Z$
- 4  $x_1 \dots \rightarrow x_i \leftarrow \dots x_k$  and  $x_i \notin Z$  and  $Desc(x_i) \notin Z$

# Global CIs in a BN

Three sets of variables  $X, Y, Z$ . If  $Z$  **d-separates**  $X$  from  $Y$  in BN then,  $X \perp\!\!\!\perp Y | Z$ .

In a directed graph  $H$ ,  $Z$  d-separates  $X$  from  $Y$  if all paths  $P$  from any  $X$  to  $Y$  is blocked by  $Z$ .

A path  $P$  is blocked by  $Z$  when

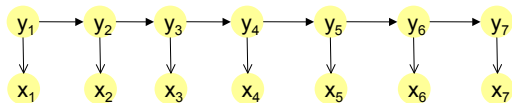
- 1  $x_1 \rightarrow x_2 \rightarrow \dots x_k$  and  $x_i \in Z$
- 2  $x_1 \leftarrow x_2 \leftarrow \dots x_k$  and  $x_i \in Z$
- 3  $x_1 \dots \leftarrow x_i \rightarrow \dots x_k$  and  $x_i \in Z$
- 4  $x_1 \dots \rightarrow x_i \leftarrow \dots x_k$  and  $x_i \notin Z$  and  $Desc(x_i) \not\subset Z$

## Theorem

*The d-separation test identifies the complete set of conditional independencies that hold in all distributions that conform to a given Bayesian network.*

# Popular Bayesian networks

- Hidden Markov Models: **speech recognition, information extraction**



- ▶ State variables: discrete **phoneme, entity tag**
  - ▶ Observation variables: continuous (**speech waveform**), discrete (**Word**)
- Kalman Filters: State variables: continuous
  - ▶ Discussed later
- Topic models for text data
  - 1 Principled mechanism to categorize multi-labeled text documents while incorporating priors in a flexible generative framework
  - 2 Application: news tracking
- QMR (Quick Medical Reference) system
- PRMs: Probabilistic relational networks:

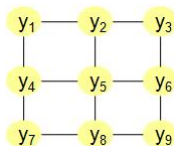


# Undirected graphical models

- Graph  $G$ : arbitrary undirected graph
- Useful when variables interact symmetrically, no natural parent-child relationship
- Example: labeling pixels of an image.
- Potentials  $\psi_C(\mathbf{y}_C)$  defined on arbitrary cliques  $C$  of  $G$ .
- $\psi_C(\mathbf{y}_C)$ : Any arbitrary non-negative value, cannot be interpreted as probability.

# Undirected graphical models

- Graph  $G$ : arbitrary undirected graph
- Useful when variables interact symmetrically, no natural parent-child relationship
- Example: labeling pixels of an image.
- Potentials  $\psi_C(\mathbf{y}_C)$  defined on arbitrary cliques  $C$  of  $G$ .
- $\psi_C(\mathbf{y}_C)$ : Any arbitrary non-negative value, cannot be interpreted as probability.
- Probability distribution

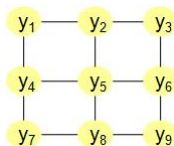


$$\Pr(y_1 \dots y_n) = \frac{1}{Z} \prod_{C \in \mathcal{G}} \psi_C(\mathbf{y}_C)$$

where  $Z = \sum_{\mathbf{y}'} \prod_{C \in \mathcal{G}} \psi_C(\mathbf{y}'_C)$  (partition function)

# Undirected graphical models

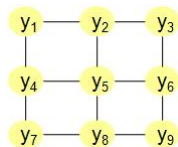
- Graph  $G$ : arbitrary undirected graph
- Useful when variables interact symmetrically, no natural parent-child relationship
- Example: labeling pixels of an image.
- Potentials  $\psi_C(\mathbf{y}_C)$  defined on arbitrary cliques  $C$  of  $G$ .
- $\psi_C(\mathbf{y}_C)$ : Any arbitrary non-negative value, cannot be interpreted as probability.
- Probability distribution



$$\Pr(y_1 \dots y_n) = \frac{1}{Z} \prod_{C \in \mathcal{G}} \psi_C(\mathbf{y}_C)$$

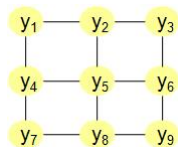
where  $Z = \sum_{\mathbf{y}'} \prod_{C \in \mathcal{G}} \psi_C(\mathbf{y}'_C)$  (partition function)

## Example



$y_i = 1$  (part of foreground), 0 otherwise.

# Example

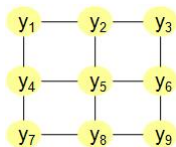


$y_i = 1$  (part of foreground), 0 otherwise.

- Node potentials

- ▶  $\psi_1(0) = 4, \psi_1(1) = 1$
- ▶  $\psi_2(0) = 2, \psi_2(1) = 3$
- ▶ ....
- ▶  $\psi_9(0) = 1, \psi_9(1) = 1$

# Example



$y_i = 1$  (part of foreground), 0 otherwise.

- Node potentials

- ▶  $\psi_1(0) = 4, \psi_1(1) = 1$

- ▶  $\psi_2(0) = 2, \psi_2(1) = 3$

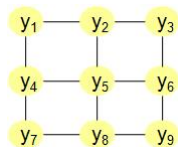
- ▶ ....

- ▶  $\psi_9(0) = 1, \psi_9(1) = 1$

- Edge potentials: Same for all edges

- ▶  $\psi(0,0) = 5, \psi(1,1) = 5, \psi(1,0) = 1, \psi(0,1) = 1$

# Example



- Node potentials

- ▶  $\psi_1(0) = 4, \psi_1(1) = 1$

- ▶  $\psi_2(0) = 2, \psi_2(1) = 3$

- ▶ ....

- ▶  $\psi_9(0) = 1, \psi_9(1) = 1$

- Edge potentials: Same for all edges

- ▶  $\psi(0,0) = 5, \psi(1,1) = 5, \psi(1,0) = 1, \psi(0,1) = 1$

- Probability:  $\Pr(y_1 \dots y_9) \propto \prod_{k=1}^9 \psi_k(y_k) \prod_{(i,j) \in E(G)} \psi(y_i, y_j)$

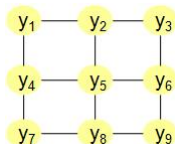
# Conditional independencies (CIs) in an undirected graphical model

Let  $V = \{y_1, \dots, y_n\}$ .

- 1 Local CI:  $y_i \perp\!\!\!\perp V - ne(y_i) - \{y_i\} | ne(y_i)$
- 2 Pairwise CI:  $y_i \perp\!\!\!\perp y_j | V - \{y_i, y_j\}$  if edge  $(y_i, y_j)$  does not exist.
- 3 Global CI:  $X \perp\!\!\!\perp Y | Z$  if  $Z$  separates  $X$  and  $Y$  in the graph.

Equivalent when the distribution is positive.

- 1  $y_1 \perp\!\!\!\perp y_3, y_5, y_6, y_7, y_8, y_9 | y_2, y_4$
- 2  $y_1 \perp\!\!\!\perp y_3 | y_2, y_4, y_5, y_6, y_7, y_8, y_9$
- 3  $y_1, y_2, y_3 \perp\!\!\!\perp y_7, y_8, y_9 | y_4, y_5, y_6$



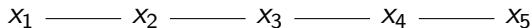


## Relationship between Local-CI and Global-CI

Let  $G$  be a undirected graph over  $V = x_1, \dots, x_n$  nodes and  $P(x_1, \dots, x_n)$  be a distribution. If  $P$  satisfies Global-CIs of  $G$ , then  $P$  will also satisfy the local-CIs of  $G$  but the reverse is not always true. We will show this with an example.

Consider a distribution over 5 binary variables:  $P(x_1, \dots, x_5)$  where  $x_1 = x_2$ ,  $x_4 = x_5$  and  $x_3 = x_2 \text{ AND } x_4$ .

Let  $G$  be



All 5 local CIs in the graph: e.g.  $x_1 \perp\!\!\!\perp \{x_3, x_4, x_5\} | x_2$  etc hold in the graph.

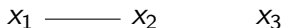
However, the global CI:  $x_2 \perp\!\!\!\perp x_4 | x_3$  does not hold.

## Relationship between Local-Cl and Pairwise-Cl

Let  $G$  be a undirected graph over  $V = x_1, \dots, x_n$  nodes and  $P(x_1, \dots, x_n)$  be a distribution. If  $P$  satisfies Local-CIs of  $G$ , then  $P$  will also satisfy the pairwise-CIs of  $G$  but the reverse is not always true. We will show this with an example.

Consider a distribution over 3 binary variables:  $P(x_1, x_2, x_3)$  where  $x_1 = x_2 = x_3$ . That is,  $P(x_1, x_2, x_3) = 1/2$  when all three are equal and 0 otherwise.

Let  $G$  be



All 2 pairwise CIs in the graph: e.g.  $x_1 \perp\!\!\!\perp \{x_3\} | x_2$  and  $x_2 \perp\!\!\!\perp \{x_3\} | x_1$  hold in the graph.

However, the local CI:  $x_1 \perp\!\!\!\perp x_3$  does not hold.

# Factorization implies Global-CI

## Theorem

*Let  $G$  be a undirected graph over  $V = x_1, \dots, x_n$  nodes and  $P(x_1, \dots, x_n)$  be a distribution. If  $P$  can be factorized as per the cliques of  $G$ , then  $P$  will also satisfy the global-CIs of  $G$*

## Proof.

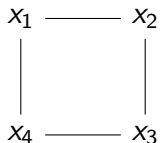
Discussed in class.



## Global-CI does not imply factorization.

But global-CI does not imply factorization. Consider a distribution over 4 binary variables:  $P(x_1, x_2, x_3, x_4)$

Let  $G$  be



Let  $P(x_1, x_2, x_3, x_4) = 1/8$  when  $x_1, x_2, x_3, x_4$  takes values from this set  $=\{0000, 1000, 1100, 1110, 1111, 0111, 0011, 0001\}$ . In all other cases it is zero. One can painfully check that all four global CIs in the graph: e.g.  $x_1 \perp\!\!\!\perp \{x_3\} | x_2, x_4$  etc hold in the graph.

Now let us look at factorization. The factors correspond to the edges in  $\psi(x_1, x_2)$ . Each of the four possible assignment of each factor will get a positive value. But that cannot represent the zero probability for cases like  $x_1, x_2, x_3, x_4 = 0101$ .

# Fractorization and CIs

## Theorem

*(Hammersley Clifford Theorem) If a positive distribution  $P(x_1, \dots, x_n)$  confirms to the pairwise CIs of a UDGM  $G$ , then it can be factorized as per the cliques  $C$  of  $G$  as*

$$P(x_1, \dots, x_n) \propto \prod_{C \in G} \psi_C(\mathbf{y}_C)$$

## Proof.

Skipped. □

# Popular undirected graphical models

- Interacting atoms in gas and solids [ 1900]
- Markov Random Fields in vision for image segmentation
- Conditional Random Fields for information extraction
- Social networks
- Bio-informatics: annotating active sites in a protein molecules.

# Comparing directed and undirected graphs

- Some distributions can only be expressed in one and not the other.



- Potentials
  - ▶ Directed: conditional probabilities, more intuitive
  - ▶ Undirected: arbitrary scores, easy to set.
- Dependence structure
  - ▶ Directed: Complicated d-separation test
  - ▶ Undirected: Graph separation:  $A \perp\!\!\!\perp B \mid C$  iff  $C$  separates  $A$  and  $B$  in  $G$ .
- Often application makes the choice clear.
  - ▶ Directed: Causality
  - ▶ Undirected: Symmetric interactions.

# Part I: Outline

- 1 Representation
  - Directed graphical models: Bayesian networks
  - Undirected graphical models
- 2 Inference Queries
  - Exact inference on chains
  - Variable elimination on general graphs
  - Junction trees
- 3 Approximate inference
  - Generalized belief propagation
  - Sampling: Gibbs, Particle filters
- 4 Constructing a graphical model
  - Graph Structure
  - Parameters in Potentials
- 5 References



# Inference queries

- ① *Marginal probability queries over a small subset of variables:*
  - ▶ Find  $\Pr(\text{Income}=\text{'High'} \ \& \ \text{Degree}=\text{'PhD'})$
  - ▶ Find  $\Pr(\text{pixel } y_9 = 1)$

$$\Pr(x_1) = \sum_{x_2 \dots x_n} \Pr(x_1 \dots x_n)$$

# Inference queries

① *Marginal probability queries over a small subset of variables:*

- ▶ Find  $\Pr(\text{Income}=\text{'High'} \ \& \ \text{Degree}=\text{'PhD'})$
- ▶ Find  $\Pr(\text{pixel } y_9 = 1)$

$$\Pr(x_1) = \sum_{x_2 \dots x_n} \Pr(x_1 \dots x_n)$$

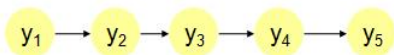
② *Most likely labels of remaining variables: (MAP queries)*

- ▶ Find most likely entity labels of all words in a sentence
- ▶ Find likely temperature at sensors in a room

$$\mathbf{x}^* = \operatorname{argmax}_{x_1 \dots x_n} \Pr(x_1 \dots x_n)$$

# Exact inference on chains

- Given,



- ▶ Graph
- ▶ Potentials:  $\psi_i(y_i, y_{i+1})$
- ▶  $Pr(y_1, \dots, y_n) = \prod_i \psi_i(y_i, y_{i+1})$
- Find,  $Pr(y_i)$  for any  $i$ , say  $Pr(y_5 = 1)$ 
  - ▶ Exact method:  $Pr(y_5 = 1) = \sum_{y_1, \dots, y_4} Pr(y_1, \dots, y_4, 1)$  requires exponential number of summations.
  - ▶ A more efficient alternative...

## Exact inference on chains

$$\begin{aligned}\Pr(y_5 = 1) &= \sum_{y_1, \dots, y_4} \Pr(y_1, \dots, y_4, 1) \\ &= \sum_{y_1} \sum_{y_2} \sum_{y_3} \sum_{y_4} \psi_1(y_1, y_2) \psi_2(y_2, y_3) \psi_3(y_3, y_4) \psi_4(y_4, 1)\end{aligned}$$

## Exact inference on chains

$$\begin{aligned}\Pr(y_5 = 1) &= \sum_{y_1, \dots, y_4} \Pr(y_1, \dots, y_4, 1) \\ &= \sum_{y_1} \sum_{y_2} \sum_{y_3} \sum_{y_4} \psi_1(y_1, y_2) \psi_2(y_2, y_3) \psi_3(y_3, y_4) \psi_4(y_4, 1) \\ &= \sum_{y_1} \sum_{y_2} \psi_1(y_1, y_2) \sum_{y_3} \psi_2(y_2, y_3) \sum_{y_4} \psi_3(y_3, y_4) \psi_4(y_4, 1)\end{aligned}$$

## Exact inference on chains

$$\begin{aligned}\Pr(y_5 = 1) &= \sum_{y_1, \dots, y_4} \Pr(y_1, \dots, y_4, 1) \\ &= \sum_{y_1} \sum_{y_2} \sum_{y_3} \sum_{y_4} \psi_1(y_1, y_2) \psi_2(y_2, y_3) \psi_3(y_3, y_4) \psi_4(y_4, 1) \\ &= \sum_{y_1} \sum_{y_2} \psi_1(y_1, y_2) \sum_{y_3} \psi_2(y_2, y_3) \sum_{y_4} \psi_3(y_3, y_4) \psi_4(y_4, 1) \\ &= \sum_{y_1} \sum_{y_2} \psi_1(y_1, y_2) \sum_{y_3} \psi_2(y_2, y_3) B_3(y_3)\end{aligned}$$

# Exact inference on chains

$$\begin{aligned}\Pr(y_5 = 1) &= \sum_{y_1, \dots, y_4} \Pr(y_1, \dots, y_4, 1) \\ &= \sum_{y_1} \sum_{y_2} \sum_{y_3} \sum_{y_4} \psi_1(y_1, y_2) \psi_2(y_2, y_3) \psi_3(y_3, y_4) \psi_4(y_4, 1) \\ &= \sum_{y_1} \sum_{y_2} \psi_1(y_1, y_2) \sum_{y_3} \psi_2(y_2, y_3) \sum_{y_4} \psi_3(y_3, y_4) \psi_4(y_4, 1) \\ &= \sum_{y_1} \sum_{y_2} \psi_1(y_1, y_2) \sum_{y_3} \psi_2(y_2, y_3) B_3(y_3) \\ &= \sum_{y_1} \sum_{y_2} \psi_1(y_1, y_2) B_2(y_2)\end{aligned}$$

# Exact inference on chains

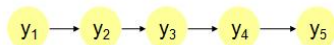
$$\begin{aligned}\Pr(y_5 = 1) &= \sum_{y_1, \dots, y_4} \Pr(y_1, \dots, y_4, 1) \\ &= \sum_{y_1} \sum_{y_2} \sum_{y_3} \sum_{y_4} \psi_1(y_1, y_2) \psi_2(y_2, y_3) \psi_3(y_3, y_4) \psi_4(y_4, 1) \\ &= \sum_{y_1} \sum_{y_2} \psi_1(y_1, y_2) \sum_{y_3} \psi_2(y_2, y_3) \sum_{y_4} \psi_3(y_3, y_4) \psi_4(y_4, 1) \\ &= \sum_{y_1} \sum_{y_2} \psi_1(y_1, y_2) \sum_{y_3} \psi_2(y_2, y_3) B_3(y_3) \\ &= \sum_{y_1} \sum_{y_2} \psi_1(y_1, y_2) B_2(y_2) \\ &= \sum_{y_1} B_1(y_1)\end{aligned}$$



# Exact inference on chains

$$\begin{aligned} \Pr(y_5 = 1) &= \sum_{y_1, \dots, y_4} \Pr(y_1, \dots, y_4, 1) \\ &= \sum_{y_1} \sum_{y_2} \sum_{y_3} \sum_{y_4} \psi_1(y_1, y_2) \psi_2(y_2, y_3) \psi_3(y_3, y_4) \psi_4(y_4, 1) \\ &= \sum_{y_1} \sum_{y_2} \psi_1(y_1, y_2) \sum_{y_3} \psi_2(y_2, y_3) \sum_{y_4} \psi_3(y_3, y_4) \psi_4(y_4, 1) \\ &= \sum_{y_1} \sum_{y_2} \psi_1(y_1, y_2) \sum_{y_3} \psi_2(y_2, y_3) B_3(y_3) \\ &= \sum_{y_1} \sum_{y_2} \psi_1(y_1, y_2) B_2(y_2) \\ &= \sum_{y_1} B_1(y_1) \end{aligned}$$

An alternative view: flow of beliefs  $B_i(\cdot)$  from node  $i + 1$  to node  $i$



# Adding evidence

Given fixed values of a subset of variables  $\mathbf{x}_e$  (evidence), find the

① *Marginal probability queries over a small subset of variables:*

- ▶ Find  $\Pr(\text{Income}=\text{'High'} \mid \text{Degree}=\text{'PhD'})$

$$\Pr(x_1) = \sum_{x_2 \dots x_m} \Pr(x_1 \dots x_n \mid \mathbf{x}_e)$$

# Adding evidence

Given fixed values of a subset of variables  $\mathbf{x}_e$  (evidence), find the

① *Marginal probability queries over a small subset of variables:*

- ▶ Find  $\Pr(\text{Income}=\text{'High'} \mid \text{Degree}=\text{'PhD'})$

$$\Pr(x_1) = \sum_{x_2 \dots x_m} \Pr(x_1 \dots x_n \mid \mathbf{x}_e)$$

② *Most likely labels of remaining variables: (MAP queries)*

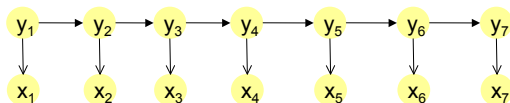
- ▶ Find likely temperature at sensors in a room given readings from a subset of them

$$\mathbf{x}^* = \operatorname{argmax}_{x_1 \dots x_m} \Pr(x_1 \dots x_n \mid \mathbf{x}_e)$$

Easy to add evidence, just change the potential.

# Inference in HMMs

- Given,



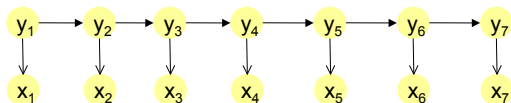
- ▶ Graph
  - ▶ Potentials:  $\Pr(y_i|y_{i-1}), \Pr(x_i|y_i)$
  - ▶ Evidence variables:  $\mathbf{x} = x_1 \dots x_n = o_1 \dots o_n$ .
- Find most likely values of the hidden state variables.

$$\mathbf{y} = y_1 \dots y_n$$

$$\operatorname{argmax}_{\mathbf{y}} \Pr(\mathbf{y}|\mathbf{x} = \mathbf{o})$$

# Inference in HMMs

- Given,

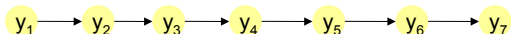


- ▶ Graph
  - ▶ Potentials:  $\Pr(y_i|y_{i-1}), \Pr(x_i|y_i)$
  - ▶ Evidence variables:  $\mathbf{x} = x_1 \dots x_n = o_1 \dots o_n$ .
- Find most likely values of the hidden state variables.

$$\mathbf{y} = y_1 \dots y_n$$

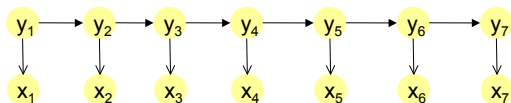
$$\operatorname{argmax}_{\mathbf{y}} \Pr(\mathbf{y}|\mathbf{x} = \mathbf{o})$$

- Define  $\psi_i(y_{i-1}, y_i) = \Pr(y_i|y_{i-1}) \Pr(x_i = o_i|y_i)$
- Reduced graph only a single chain of  $y$  nodes.



# Inference in HMMs

- Given,

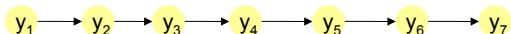


- ▶ Graph
  - ▶ Potentials:  $\Pr(y_i|y_{i-1}), \Pr(x_i|y_i)$
  - ▶ Evidence variables:  $\mathbf{x} = x_1 \dots x_n = o_1 \dots o_n$ .
- Find most likely values of the hidden state variables.

$$\mathbf{y} = y_1 \dots y_n$$

$$\operatorname{argmax}_{\mathbf{y}} \Pr(\mathbf{y}|\mathbf{x} = \mathbf{o})$$

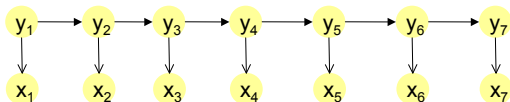
- Define  $\psi_i(y_{i-1}, y_i) = \Pr(y_i|y_{i-1}) \Pr(x_i = o_i|y_i)$
- Reduced graph only a single chain of  $y$  nodes.



- Algorithm same as earlier, just replace “Sum” with “Max”

# Inference in HMMs

- Given,

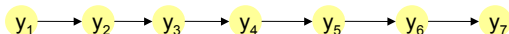


- ▶ Graph
  - ▶ Potentials:  $\Pr(y_i|y_{i-1})$ ,  $\Pr(x_i|y_i)$
  - ▶ Evidence variables:  $\mathbf{x} = x_1 \dots x_n = o_1 \dots o_n$ .
- Find most likely values of the hidden state variables.

$$\mathbf{y} = y_1 \dots y_n$$

$$\operatorname{argmax}_{\mathbf{y}} \Pr(\mathbf{y}|\mathbf{x} = \mathbf{o})$$

- Define  $\psi_i(y_{i-1}, y_i) = \Pr(y_i|y_{i-1}) \Pr(x_i = o_i|y_i)$
- Reduced graph only a single chain of  $y$  nodes.



- Algorithm same as earlier, just replace “Sum” with “Max”

This is the well-known Viterbi algorithm

# Variable elimination on general graphs

- Given, arbitrary sets of potentials  $\psi_C(x_C)$ ,  $C =$  cliques in a graph  $G$ .
- Find,  $Z = \sum_{x_1, \dots, x_n} \prod_C \psi_C(x_C)$

$x_1, \dots, x_n =$  good ordering of variables

$\mathcal{F} = \psi_C(x_C)$ ,  $C =$  cliques in a graph  $G$ .

**for**  $i = 1 \dots n$  **do**

$\mathcal{F}_i =$  factors in  $\mathcal{F}$  that contain  $x_i$

$M_i =$  product of factors in  $\mathcal{F}_i$

$m_i = \sum_{x_i} M_i$

$\mathcal{F} = \mathcal{F} - \mathcal{F}_i \cup \{m_i\}$

**end for**



## Example: Variable elimination

- Given,  $\psi_{12}(x_1, x_2)$ ,  $\psi_{24}(x_2, x_4)$ ,  $\psi_{23}(x_2, x_3)$ ,  $\psi_{45}(x_4, x_5)$ , ,  $\psi_{35}(x_3, x_5)$ .
- Find,  $Z = \sum_{x_1, \dots, x_5} \psi_{12}(x_1, x_2) \psi_{24}(x_2, x_4) \psi_{23}(x_2, x_3) \psi_{45}(x_4, x_5) \psi_{35}(x_3, x_5)$ .
- ①  $x_1: \prod\{\psi_{12}(x_1, x_2)\} \rightarrow M_1(x_1, x_2) \xrightarrow{\sum_{x_1}} m_1(x_2)$

## Example: Variable elimination

- Given,  $\psi_{12}(x_1, x_2)$ ,  $\psi_{24}(x_2, x_4)$ ,  $\psi_{23}(x_2, x_3)$ ,  $\psi_{45}(x_4, x_5)$ , ,  $\psi_{35}(x_3, x_5)$ .
  - Find,  $Z = \sum_{x_1, \dots, x_5} \psi_{12}(x_1, x_2) \psi_{24}(x_2, x_4) \psi_{23}(x_2, x_3) \psi_{45}(x_4, x_5) \psi_{35}(x_3, x_5)$ .
- $x_1$ :  $\prod\{\psi_{12}(x_1, x_2)\} \rightarrow M_1(x_1, x_2) \xrightarrow{\sum_{x_1}} m_1(x_2)$
  - $x_2$ :  $\prod\{\psi_{24}(x_2, x_4), \psi_{23}(x_2, x_3), m_1(x_2)\} \rightarrow M_2(x_2, x_3, x_4) \xrightarrow{\sum_{x_2}} m_2(x_3, x_4)$

## Example: Variable elimination

- Given,  $\psi_{12}(x_1, x_2)$ ,  $\psi_{24}(x_2, x_4)$ ,  $\psi_{23}(x_2, x_3)$ ,  $\psi_{45}(x_4, x_5)$ , ,  $\psi_{35}(x_3, x_5)$ .
- Find,  $Z = \sum_{x_1, \dots, x_5} \psi_{12}(x_1, x_2) \psi_{24}(x_2, x_4) \psi_{23}(x_2, x_3) \psi_{45}(x_4, x_5) \psi_{35}(x_3, x_5)$ .

$$\textcircled{1} \quad x_1: \prod\{\psi_{12}(x_1, x_2)\} \rightarrow M_1(x_1, x_2) \xrightarrow{\sum_{x_1}} m_1(x_2)$$

$$\textcircled{2} \quad x_2: \prod\{\psi_{24}(x_2, x_4), \psi_{23}(x_2, x_3), m_1(x_2)\} \rightarrow M_2(x_2, x_3, x_4) \xrightarrow{\sum_{x_2}} m_2(x_3, x_4)$$

$$\textcircled{3} \quad x_3: \prod\{\psi_{35}(x_3, x_5), m_2(x_3, x_4)\} \rightarrow M_3(x_3, x_4, x_5) \xrightarrow{\sum_{x_3}} m_3(x_4, x_5)$$

## Example: Variable elimination

- Given,  $\psi_{12}(x_1, x_2)$ ,  $\psi_{24}(x_2, x_4)$ ,  $\psi_{23}(x_2, x_3)$ ,  $\psi_{45}(x_4, x_5)$ , ,  $\psi_{35}(x_3, x_5)$ .
- Find,  $Z = \sum_{x_1, \dots, x_5} \psi_{12}(x_1, x_2) \psi_{24}(x_2, x_4) \psi_{23}(x_2, x_3) \psi_{45}(x_4, x_5) \psi_{35}(x_3, x_5)$ .

- $x_1: \prod\{\psi_{12}(x_1, x_2)\} \rightarrow M_1(x_1, x_2) \xrightarrow{\sum_{x_1}} m_1(x_2)$
- $x_2: \prod\{\psi_{24}(x_2, x_4), \psi_{23}(x_2, x_3), m_1(x_2)\} \rightarrow M_2(x_2, x_3, x_4) \xrightarrow{\sum_{x_2}} m_2(x_3, x_4)$
- $x_3: \prod\{\psi_{35}(x_3, x_5), m_2(x_3, x_4)\} \rightarrow M_3(x_3, x_4, x_5) \xrightarrow{\sum_{x_3}} m_3(x_4, x_5)$
- $x_4: \prod\{\psi_{45}(x_4, x_5), m_3(x_4, x_5)\} \rightarrow M_4(x_4, x_5) \xrightarrow{\sum_{x_4}} m_4(x_5)$

## Example: Variable elimination

- Given,  $\psi_{12}(x_1, x_2)$ ,  $\psi_{24}(x_2, x_4)$ ,  $\psi_{23}(x_2, x_3)$ ,  $\psi_{45}(x_4, x_5)$ , ,  $\psi_{35}(x_3, x_5)$ .
- Find,  $Z = \sum_{x_1, \dots, x_5} \psi_{12}(x_1, x_2) \psi_{24}(x_2, x_4) \psi_{23}(x_2, x_3) \psi_{45}(x_4, x_5) \psi_{35}(x_3, x_5)$ .

- $x_1: \prod\{\psi_{12}(x_1, x_2)\} \rightarrow M_1(x_1, x_2) \xrightarrow{\sum_{x_1}} m_1(x_2)$
- $x_2: \prod\{\psi_{24}(x_2, x_4), \psi_{23}(x_2, x_3), m_1(x_2)\} \rightarrow M_2(x_2, x_3, x_4) \xrightarrow{\sum_{x_2}} m_2(x_3, x_4)$
- $x_3: \prod\{\psi_{35}(x_3, x_5), m_2(x_3, x_4)\} \rightarrow M_3(x_3, x_4, x_5) \xrightarrow{\sum_{x_3}} m_3(x_4, x_5)$
- $x_4: \prod\{\psi_{45}(x_4, x_5), m_3(x_4, x_5)\} \rightarrow M_4(x_4, x_5) \xrightarrow{\sum_{x_4}} m_4(x_5)$
- $x_5: \prod\{m_4(x_5)\} \rightarrow M_5(x_5) \xrightarrow{\sum_{x_5}} Z$

## Choosing a variable elimination order

- Complexity of VE  $O(nm^w)$  where  $w$  is the maximum number of variables in any factor.
- Wrong elimination order can give rise to very large intermediate factors.
- Example: eliminating  $x_2$  first will give a factor of size 4.
- Given an example where the penalty can be really severe (?)
- Choosing the optimal elimination order is NP hard for general graphs.
- Polynomial time algorithm exists for chordal graphs.
  - ▶ A graph is chordal or triangulated if all cycles of length greater than three have a shortcut.
- Optimal triangulation of graphs is NP hard. (Many heuristics)

# Junction tree algorithm

- An **optimal** general-purpose algorithm for **exact** marginal/MAP queries
- Simultaneous computation of many queries
- Efficient data structures
- Complexity:  $O(m^w N)$   $w$  = size of the largest clique in (triangulated) graph,  $m$  = number of values of each discrete variable in the clique. → **linear for trees**.
- Basis for many approximate algorithms.
- Many popular inference algorithms special cases of junction trees
  - ▶ Viterbi algorithm of HMMs
  - ▶ Forward-backward algorithm of Kalman filters

# Junction tree

Junction tree JT of a triangulated graph  $G$  with nodes  $x_1, \dots, x_n$  is a tree where

- Nodes = cliques of  $G$
- Edges ensure that if any two nodes contain a variable  $x_i$  then  $x_i$  is present in every node in the unique path between them (Running intersection property).



# Junction tree

Junction tree JT of a triangulated graph  $G$  with nodes  $x_1, \dots, x_n$  is a **tree** where

- Nodes = cliques of  $G$
- Edges ensure that if any two nodes contain a variable  $x_i$  then  $x_i$  is present in every node in the unique path between them (**Running intersection property**).

## Constructing a junction tree

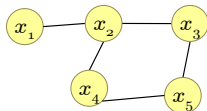
Efficient polynomial time algorithms exist for creating a JT from a triangulated graph.

- 1 Enumerate a covering set of cliques
- 2 Connect cliques to get a tree that satisfies the running intersection property.

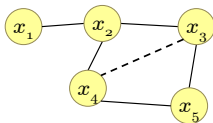
If graph is non-triangulated, triangulate first using heuristics, optimal triangulation is NP-hard.

# Creating a junction tree from a graphical model

1. Starting graph



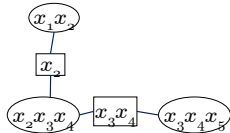
2. Triangulate graph



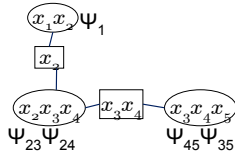
3. Create clique nodes



4. Create tree edges such that variables connected.



5) Assign potentials to exactly one subsumed clique node.



# Finding cliques of a triangulated graph

## Theorem

*Every triangulated graph has a **simplicial** vertex, that is, a vertex whose neighbors form a complete set.*

Input: Graph  $G$ .  $n =$  number of vertices of  $G$

**for**  $i = 1, \dots, n$  **do**

$\pi_i =$  pick any simplicial vertex in  $G$

$C_i = \{\pi_i\} \cup \text{Ne}(\pi_i)$

remove  $\pi_i$  from  $G$

**end for**

Return maximal cliques from  $C_1, \dots, C_n$

# Connecting cliques to form junction tree

Separator variables = intersection of variables in the two cliques joined by an edge.

## Theorem

*A clique tree that satisfies the running intersection property maximizes the number of separator variables.*

Input: Cliques:  $C_1, \dots, C_k$

Form a complete weighted graph  $H$  with cliques as nodes and edge weights = size of the intersection of the two cliques it connects.

$T$  = maximum weight spanning tree of  $H$

Return  $T$  as the junction tree.

# Belief propagation on junction trees

- Each node  $c$ 
  - ▶ sends *belief*  $B_{c \rightarrow c'}(\cdot)$  to each of its neighbors  $c'$ 
    - ★ once it has beliefs from every other neighbor  $N(c) - \{c'\}$ .
  - ▶  $B_{c \rightarrow c'}(\cdot) =$  belief that clique  $c$  has about the distribution of labels to common variables  $s = c \cap c'$

$$B_{c \rightarrow c'}(\mathbf{x}_s) = \sum_{\mathbf{x}_{c-s}} \psi_c(\mathbf{x}_c) \prod_{d \in N(c) - \{c'\}} B_{d \rightarrow c}(\mathbf{x}_{d \cap c})$$

Replace “sum” with “max” for MAP queries.

# Belief propagation on junction trees

- Each node  $c$ 
  - ▶ sends *belief*  $B_{c \rightarrow c'}(\cdot)$  to each of its neighbors  $c'$ 
    - ★ once it has beliefs from every other neighbor  $N(c) - \{c'\}$ .
  - ▶  $B_{c \rightarrow c'}(\cdot)$  = belief that clique  $c$  has about the distribution of labels to common variables  $s = c \cap c'$

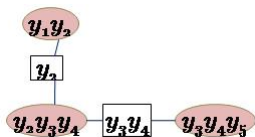
$$B_{c \rightarrow c'}(\mathbf{x}_s) = \sum_{\mathbf{x}_{c-s}} \psi_c(\mathbf{x}_c) \prod_{d \in N(c) - \{c'\}} B_{d \rightarrow c}(\mathbf{x}_{d \cap c})$$

Replace “sum” with “max” for MAP queries.

Compute marginal probability of any variable  $x_i$  as

- 1  $c =$  clique in JT containing  $x_i$
- 2  $\Pr(x_i) \propto \sum_{\mathbf{x}_{c-x_i}} \psi_c(\mathbf{x}_c) \prod_{d \in N(c)} B_{d \rightarrow c}(\mathbf{x}_{d \cap c})$

# Example



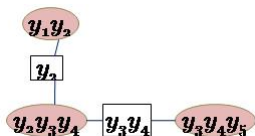
$$\psi_{234}(\mathbf{y}_{234}) = \psi_{23}(\mathbf{y}_{23})\psi_{34}(\mathbf{y}_{34})$$

$$\psi_{345}(\mathbf{y}_{345}) = \psi_{35}(\mathbf{y}_{35})\psi_{45}(\mathbf{y}_{45})$$

$$\psi_{234}(\mathbf{y}_{12}) = \psi_{12}(\mathbf{y}_{12})$$

- 1 Clique “12” sends belief  $B_{12 \rightarrow 234}(y_2) = \sum_{y_1} \psi_{12}(\mathbf{y}_{12})$  to its only neighbor.

# Example



$$\psi_{234}(\mathbf{y}_{234}) = \psi_{23}(\mathbf{y}_{23})\psi_{34}(\mathbf{y}_{34})$$

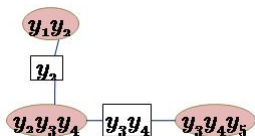
$$\psi_{345}(\mathbf{y}_{345}) = \psi_{35}(\mathbf{y}_{35})\psi_{45}(\mathbf{y}_{45})$$

$$\psi_{234}(\mathbf{y}_{12}) = \psi_{12}(\mathbf{y}_{12})$$

- 1 Clique “12” sends belief  $B_{12 \rightarrow 234}(y_2) = \sum_{y_1} \psi_{12}(\mathbf{y}_{12})$  to its only neighbor.
- 2 Clique “345” sends belief  $B_{345 \rightarrow 234}(\mathbf{y}_{34}) = \sum_{y_5} \psi_{234}(\mathbf{y}_{345})$  to “234”



# Example



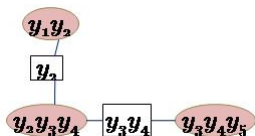
$$\psi_{234}(\mathbf{y}_{234}) = \psi_{23}(\mathbf{y}_{23})\psi_{34}(\mathbf{y}_{34})$$

$$\psi_{345}(\mathbf{y}_{345}) = \psi_{35}(\mathbf{y}_{35})\psi_{45}(\mathbf{y}_{45})$$

$$\psi_{234}(\mathbf{y}_{12}) = \psi_{12}(\mathbf{y}_{12})$$

- 1 Clique “12” sends belief  $B_{12 \rightarrow 234}(y_2) = \sum_{y_1} \psi_{12}(\mathbf{y}_{12})$  to its only neighbor.
- 2 Clique “345” sends belief  $B_{345 \rightarrow 234}(\mathbf{y}_{34}) = \sum_{y_5} \psi_{234}(\mathbf{y}_{345})$  to “234”
- 3 Clique “234” sends belief  $B_{234 \rightarrow 345}(\mathbf{y}_{34}) = \sum_{y_2} \psi_{234}(\mathbf{y}_{234}) B_{12 \rightarrow 234}(y_2)$  to “345”

# Example



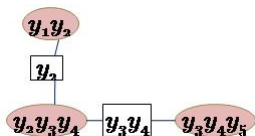
$$\psi_{234}(\mathbf{y}_{234}) = \psi_{23}(\mathbf{y}_{23})\psi_{34}(\mathbf{y}_{34})$$

$$\psi_{345}(\mathbf{y}_{345}) = \psi_{35}(\mathbf{y}_{35})\psi_{45}(\mathbf{y}_{45})$$

$$\psi_{234}(\mathbf{y}_{12}) = \psi_{12}(\mathbf{y}_{12})$$

- 1 Clique "12" sends belief  $B_{12 \rightarrow 234}(y_2) = \sum_{y_1} \psi_{12}(\mathbf{y}_{12})$  to its only neighbor.
- 2 Clique "345" sends belief  $B_{345 \rightarrow 234}(\mathbf{y}_{34}) = \sum_{y_5} \psi_{345}(\mathbf{y}_{345})$  to "234"
- 3 Clique "234" sends belief  $B_{234 \rightarrow 345}(\mathbf{y}_{34}) = \sum_{y_2} \psi_{234}(\mathbf{y}_{234}) B_{12 \rightarrow 234}(y_2)$  to "345"
- 4 Clique "234" sends belief  $B_{234 \rightarrow 12}(y_2) = \sum_{y_4} \psi_{234}(\mathbf{y}_{234}) B_{345 \rightarrow 234}(\mathbf{y}_{34})$  to "12"

# Example



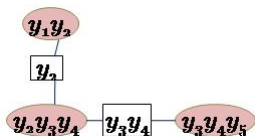
$$\psi_{234}(\mathbf{y}_{234}) = \psi_{23}(\mathbf{y}_{23})\psi_{34}(\mathbf{y}_{34})$$

$$\psi_{345}(\mathbf{y}_{345}) = \psi_{35}(\mathbf{y}_{35})\psi_{45}(\mathbf{y}_{45})$$

$$\psi_{234}(\mathbf{y}_{12}) = \psi_{12}(\mathbf{y}_{12})$$

- 1 Clique "12" sends belief  $B_{12 \rightarrow 234}(y_2) = \sum_{y_1} \psi_{12}(\mathbf{y}_{12})$  to its only neighbor.
- 2 Clique "345" sends belief  $B_{345 \rightarrow 234}(\mathbf{y}_{34}) = \sum_{y_5} \psi_{345}(\mathbf{y}_{345})$  to "234"
- 3 Clique "234" sends belief  $B_{234 \rightarrow 345}(\mathbf{y}_{34}) = \sum_{y_2} \psi_{234}(\mathbf{y}_{234}) B_{12 \rightarrow 234}(y_2)$  to "345"
- 4 Clique "234" sends belief  $B_{234 \rightarrow 12}(y_2) = \sum_{y_4} \psi_{234}(\mathbf{y}_{234}) B_{345 \rightarrow 234}(\mathbf{y}_{34})$  to "12"

# Example



$$\psi_{234}(\mathbf{y}_{234}) = \psi_{23}(\mathbf{y}_{23})\psi_{34}(\mathbf{y}_{34})$$

$$\psi_{345}(\mathbf{y}_{345}) = \psi_{35}(\mathbf{y}_{35})\psi_{45}(\mathbf{y}_{45})$$

$$\psi_{234}(\mathbf{y}_{12}) = \psi_{12}(\mathbf{y}_{12})$$

- 1 Clique "12" sends belief  $B_{12 \rightarrow 234}(y_2) = \sum_{y_1} \psi_{12}(\mathbf{y}_{12})$  to its only neighbor.
- 2 Clique "345" sends belief  $B_{345 \rightarrow 234}(\mathbf{y}_{34}) = \sum_{y_5} \psi_{234}(\mathbf{y}_{345})$  to "234"
- 3 Clique "234" sends belief  $B_{234 \rightarrow 345}(\mathbf{y}_{34}) = \sum_{y_2} \psi_{234}(\mathbf{y}_{234}) B_{12 \rightarrow 234}(y_2)$  to "345"
- 4 Clique "234" sends belief  $B_{234 \rightarrow 12}(y_2) = \sum_{y_4} \psi_{234}(\mathbf{y}_{234}) B_{345 \rightarrow 234}(\mathbf{y}_{34})$  to "12"

$$\Pr(y_1) \propto \sum_{y_2} \psi_{12}(\mathbf{y}_{12}) B_{234 \rightarrow 12}(y_2)$$

# Part I: Outline

## 1 Representation

- Directed graphical models: Bayesian networks
- Undirected graphical models

## 2 Inference Queries

- Exact inference on chains
- Variable elimination on general graphs
- Junction trees

## 3 Approximate inference

- Generalized belief propagation
- Sampling: Gibbs, Particle filters

## 4 Constructing a graphical model

- Graph Structure
- Parameters in Potentials

## 5 References

# Why approximate inference

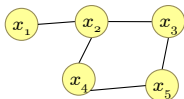
- Exact inference is NP hard. Complexity:  $O(w^m)$ 
  - ▶  $w$  = tree width = size of the largest clique in (triangulated) graph-1,
  - ▶  $m$  = number of values of each discrete variable in the clique.
- Many real-life graphs produce large cliques on triangulation
  - ▶ A  $n \times n$  grid has a tree width of  $n$
  - ▶ A Kalman filter on  $K$  parallel state variables influencing a common observation variable, has a tree width of size  $K + 1$

# Generalized belief propagation

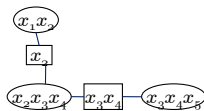
- Approximate junction tree with a cluster graph where
  - 1 Nodes = arbitrary clusters, not cliques in triangulated graph.  
Only ensure all potentials subsumed.
  - 2 Separator nodes on edges = *subset* of intersecting variables.
- Special case: factor graphs.

## Example cluster graph

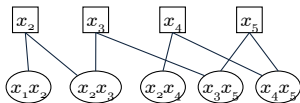
Starting graph



Junction tree.



Cluster graph



# Belief propagation in cluster graphs

- Graph can have loops, tree-based two-phase method not applicable.
- Many variants on scheduling order of propagating beliefs.
  - ▶ Simple loopy belief propagation [?]
  - ▶ Tree-reweighted message passing [?, ?]
  - ▶ Residual belief propagation [?]
- Many have no guarantees of convergence. Specific tree-based orders do [?]
- Works well in practice, default method of choice.



# MCMC (Gibbs) sampling

- Useful when all else fails, guaranteed to converge to the optimal over infinite number of samples.
- Basic premise: easy to compute conditional probability  $\Pr(x_i | \text{fixed values of remaining variables})$

## Algorithm

- Start with some initial assignment, say  $\mathbf{x}^1 = [x_1, \dots, x_n] = [0, \dots, 0]$
- For several iterations
  - ▶ For each variable  $x_i$   
Get a new sample  $\mathbf{x}^{t+1}$  by replacing value of  $x_i$  with a new value sampled according to probability  $\Pr(x_i | x_1^t, \dots, x_{i-1}^t, x_{i+1}^t, \dots, x_n^t)$

## Others

- Combinatorial algorithms for MAP [?].
- Greedy algorithms: relaxation labeling.
- Variational methods like mean-field and structured mean-field.
- LP and QP based approaches.

# Part I: Outline

- 1 Representation
  - Directed graphical models: Bayesian networks
  - Undirected graphical models
- 2 Inference Queries
  - Exact inference on chains
  - Variable elimination on general graphs
  - Junction trees
- 3 Approximate inference
  - Generalized belief propagation
  - Sampling: Gibbs, Particle filters
- 4 Constructing a graphical model
  - Graph Structure
  - Parameters in Potentials
- 5 References

# Graph Structure

- ① Manual: Designed by domain expert
  - ▶ Used in applications where dependency structure is well-understood
  - ▶ Example: QMR systems, Kalman filters, Vision (Grids), HMM for speech recognition and IE.
- ② Learned from examples
  - ▶ NP hard to find the optimal structure.
  - ▶ Widely researched, mostly posed as a branch and bound search problem.
  - ▶ Useful in dynamic situations

# Parameters in Potentials

- ① Manual: Provided by domain expert
  - ▶ Used in infrequently constructed graphs, example QMR systems
  - ▶ Also where potentials are an easy function of the attributes of connected graphs, example: vision networks.
- ② Learned: from examples
  - ▶ More popular since difficult for humans to assign numeric values
  - ▶ Many variants of parameterizing potentials.
    - ① Each potential entry a parameter, example, HMMs
    - ② Potentials: combination of shared parameters and data attributes: example, CRFs. (Discussed in structured learning tutorial)

# Learning potentials

Given sample  $D = \{\mathbf{x}^1, \dots, \mathbf{x}^N\}$  of data generated from a distribution  $P(\mathbf{x})$  represented by a graphical model with known structure  $G$ , learn potentials  $\psi_C(\mathbf{x}_C)$ .

Two dimensions:

- 1 All variables observed or not.
  - 1 Fully observed: each training sample  $\mathbf{x}^i$  has all  $n$  variables observed.
  - 2 Partially observed: a subset of the variables are observed.
- 2 Potentials coupled with a log-partition function or not.
  - 1 No: **Closed form solutions**
  - 2 Yes: Potentials attached to arbitrary overlapping subset of variables in a UDGM. Example = edge potentials in a grid graph. **iterative solution as in the case of learning with shared parameters** Discussed later.

# Potential learning: fully observed, decoupled potentials

- 1 Potentials in a Bayesian network  $P(\mathbf{x}) = \prod_i \Pr(x_i | Pa(x_i))$
- 2 potentials attached to maximal cliques in UDGM.

$$\Pr(\mathbf{x}) = \frac{\prod_{C \in \text{Cliques}} \Pr(\mathbf{x}_C)}{\prod_{S \in \text{Separators}} \Pr(\mathbf{x}_S)}$$

Maximum likelihood estimation with constraints on potentials to make them behave like probabilities:

$$\Pr(\mathbf{x}_C) = \frac{\sum_{i=1}^N [[\mathbf{x}_C^i == \mathbf{x}_C]]}{N}$$

# Potential learning: fully observed, decoupled potentials

- 1 Potentials in a Bayesian network  $P(\mathbf{x}) = \prod_i \Pr(x_i | Pa(x_i))$
- 2 potentials attached to maximal cliques in UDGM.

$$\Pr(\mathbf{x}) = \frac{\prod_{C \in \text{Cliques}} \Pr(\mathbf{x}_C)}{\prod_{S \in \text{Separators}} \Pr(\mathbf{x}_S)}$$

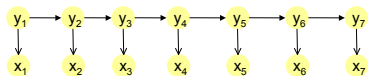
Maximum likelihood estimation with constraints on potentials to make them behave like probabilities:

$$\Pr(\mathbf{x}_C) = \frac{\sum_{i=1}^N [[\mathbf{x}_C^i == \mathbf{x}_C]]}{N}$$

$$\Pr(x_j | pa(x_j)) = \frac{\sum_{i=1}^N [[x_j^i == x_j, \mathbf{x}_{Pa(j)}^i = pa(x_j)]]}{\sum_{i=1}^N [[\mathbf{x}_{Pa(j)}^i = pa(x_j)]]}$$



# Partially observed, decoupled potentials



## EM Algorithm

Input: Graph  $G$ , Data  $D$  with observed subset of variables  $\mathbf{x}$  and hidden variables  $\mathbf{z}$ .

Initially ( $t = 0$ ): Assign random variables of parameters

$$\Pr(x_j | pa(x_j))^t$$

**for**  $i = 1, \dots, T$  **do**

**E-step**

**for**  $i = 1, \dots, N$  **do**

Use inference in  $G$  to estimate conditionals  $\Pr_i(\mathbf{z}_c | \mathbf{x}^i)^t$  for all variable subsets  $(i, pa(i))$  involving any hidden variable.

**end for**

**M-step**

$$\Pr(x_j | pa(x_j) = \mathbf{z}_c)^t = \frac{\sum_{i=1}^N \Pr_i(\mathbf{z}_c | \mathbf{x}^i) \mathbb{I}[[x_j^i == x_j]]}{\sum_{i=1}^N \Pr_i(\mathbf{z}_c | \mathbf{x}^i)^t}$$

**end for**

# Part I: Outline

- 1 Representation
  - Directed graphical models: Bayesian networks
  - Undirected graphical models
- 2 Inference Queries
  - Exact inference on chains
  - Variable elimination on general graphs
  - Junction trees
- 3 Approximate inference
  - Generalized belief propagation
  - Sampling: Gibbs, Particle filters
- 4 Constructing a graphical model
  - Graph Structure
  - Parameters in Potentials
- 5 References

## More on graphical models

- Koller and Friedman, Probabilistic Graphical Models: Principles and Techniques. MIT Press, 2009.
- Wainwright's article in FnT for Machine Learning. 2009.
- Kevin Murphy's brief online introduction (<http://www.cs.ubc.ca/~murphyk/Bayes/bnintro.html>)
- Graphical models. M. I. Jordan. Statistical Science (Special Issue on Bayesian Statistics), 19, 140-155, 2004. (<http://www.cs.berkeley.edu/~jordan/papers/statsci.ps.gz>)
- Other text books:
  - ▶ R. G. Cowell, A. P. Dawid, S. L. Lauritzen and D. J. Spiegelhalter. "Probabilistic Networks and Expert Systems". Springer-Verlag. 1999.
  - ▶ J. Pearl. "Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference." Morgan Kaufmann. 1988.
  - ▶ Graphical models by Lauritzen, Oxford science publications F. V. Jensen. "Bayesian Networks and Decision Graphs". Springer. 2001.