Probabilistic Graphical models

Vicenç Gómez

Donders Institute for Brain, Cognition and Behaviour,
Radboud University, Nijmegen, The Netherlands

Outline

1. Introduction to probabilistic Graphical Models
   - Bayesian networks
   - Markov Random Fields
   - Factor graphs and the Sum-Product algorithm
2. Learning Graphical Models

Most of the material of these slides has been taken from :

- Chapter 8 of C. Bishop's book
- D. Mackay's book
- Tutorial on Graphical Models of Z. Ghahramani (MLSS 2012)

- Unifying language to express many existing problems
- Intersection of many different scientific areas
  - probability theory
  - computer science
  - decision theory
  - optimization
  - ...
- Examples of applications: medical and fault diagnosis, image understanding, reconstruction of biological networks, speech recognition, natural language processing, decoding of messages sent over a noisy communication channel, robot navigation, and many more

- Defines a family of joint probability distributions in terms of a graph
  - directed : Bayesian Network (AI community)
  - undirected : Markov Random Field (stat.physics, computer vision)
  - bipartite factor graph (general class, coding theory)
- Joint probability factorizes as a product of potential functions defined on *small* subsets of variables (nodes in the graph).
- Independencies encoded in the structure of the graph

Computational tasks

- **Inference**: estimate probabilities for a given fixed joint distribution
  - Posterior marginals or belief $p(\mathbf{x}|\mathbf{e})$ over latent variables
  - Probability of evidence $p(\mathbf{e})$
  - Maximum a Posteriori hypothesis (map) $p(\mathbf{z}|\mathbf{e})$
- **Learning**: find best graphical model that explains given data
  - Learning parameters
  - Structure learning

- Defines a family of joint probability distributions in terms of a graph
  - directed : Bayesian Network (AI community)
  - undirected : Markov Random Field (stat.physics, computer vision)
  - bipartite factor graph (general class, coding theory)
- Joint probability factorizes as a product of potential functions defined on *small* subsets of variables (nodes in the graph).
- Independencies encoded in the structure of the graph

Computational tasks

- **Inference**: estimate probabilities for a given fixed joint distribution
  - Posterior marginals or belief $p(\mathbf{x}|\mathbf{e})$ over latent variables
  - Probability of evidence $p(\mathbf{e})$
  - Maximum a Posteriori hypothesis (map) $p(\mathbf{z}|\mathbf{e})$
- **Learning**: find best graphical model that explains given data
  - Learning parameters
  - Structure learning

Introduction: Quick recap on probability theory

Definitions:

- X is a **random variable**, takes values $x \in \mathcal{A}_X = \{a_1, a_2, \ldots, a_i, \ldots, a_I\}$ with probabilities $\mathcal{P}_X = \{p_1, p_2, \ldots, p_i, \ldots, p_I\}$
  - $p(x = a_i) = p_i, p_i \geq 0$
  - $\sum_{a_i \in \mathcal{A}_X} P(x = a_i) = 1$
- Probability of a **subset**: if T is a subset of $\mathcal{A}_X$ then: $P(T) = P(x \in T) = \sum_{a_i \in T} P(x = a_i)$
- if XY is an ordered pair of variables where then $P(x, y)$ is the **joint probability** of x and y
- **Marginal** probability: $P(x = a_i) = \sum_{y \in \mathcal{A}_y} P(x = a_i, y)$
- **Conditional** probability:

$$P(x = a_i | y = b_j) = \frac{P(x = a_i, y = b_j)}{P(y = b_j)}, \text{if } P(y = b_j) \neq 0$$

Introduction: Quick recap on probability theory

Rules of probability:

- Product rule $P(x, y|\mathcal{H}) = P(x|y, \mathcal{H})P(y|\mathcal{H}) = P(y|x, \mathcal{H})P(x|\mathcal{H})$
- Sum rule $P(x|\mathcal{H}) = \sum_y P(x, y|\mathcal{H}) = \sum_y P(x|y, \mathcal{H})P(y|\mathcal{H})$
- Bayes theorem

$$P(y|x, \mathcal{H}) = \frac{p(x|y, \mathcal{H})P(y|\mathcal{H})}{P(x|\mathcal{H})} = \frac{p(x|y, \mathcal{H})P(y|\mathcal{H})}{\sum_{y'} p(x|y', \mathcal{H})P(y'|\mathcal{H})}$$

- Marginal independence: X and Y are independent $X \perp\!\!\!\perp Y|\emptyset$ if and only if

$$P(x, y) = P(x)P(y)$$

- Conditional independence: X and Y are independent given Z $X \perp\!\!\!\perp Y|Z$ if and only if

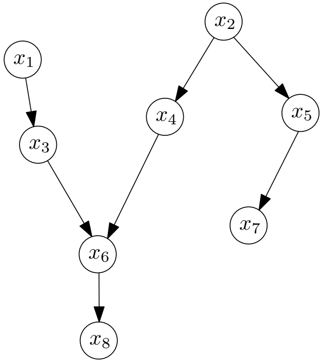
$$P(x, y|z) = P(x|z)P(y|z)$$

Outline

1. Introduction to probabilistic Graphical Models
   - Bayesian networks
   - Markov Random Fields
   - Factor graphs and the Sum-Product algorithm

2. Learning Graphical Models

Asia network

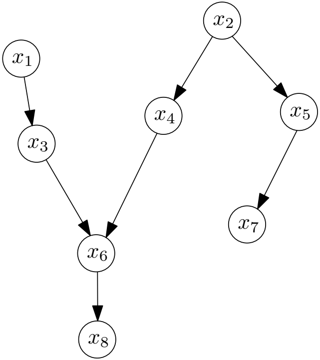


- x_1 : Visit to Asia
- x_2 : Smoker
- x_3 : Has Tuberculosis
- x_4 : Has Lung Cancer
- x_5 : Has Bronquitis
- x_6 : Tuberculosis or Cancer
- x_7 : X-Ray result
- x_8 : Dyspnea

Naive factorization: $p(\mathbf{x}) = p(x_1|x_2, \ldots, x_8)p(x_2|x_3, \ldots, x_8) \ldots p(x_8)$

Requires table with $\mathbf{2^8}$ elements!

Asia network

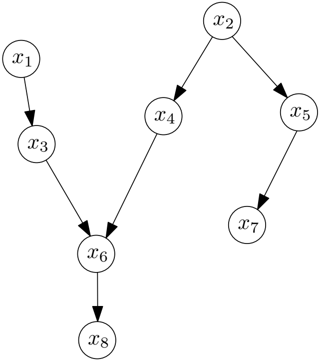


- x_1 : Visit to Asia
- x_2 : Smoker
- x_3 : Has Tuberculosis
- x_4 : Has Lung Cancer
- x_5 : Has Bronquitis
- x_6 : Tuberculosis or Cancer
- x_7 : X-Ray result
- x_8 : Dyspnea

$$p(\mathbf{x}) = p(x_3|x_1)p(x_1)p(x_4|x_2)p(x_5|x_2)p(x_2)p(x_6|x_3,x_4)p(x_7|x_5)p(x_8|x_6)$$

Requires table with $\mathbf{2^3}$ elements!

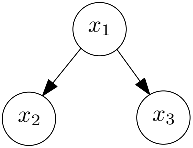
Asia network



- x_1 : Visit to Asia
- x_2 : Smoker
- x_3 : Has Tuberculosis
- x_4 : Has Lung Cancer
- x_5 : Has Bronquitis
- x_6 : Tuberculosis or Cancer
- x_7 : X-Ray result
- x_8 : Dyspnea

In general $p(\mathbf{x}) = \prod_i p(x_i|\text{parents}_i)$

- Given:
  - A directed graphical model
  - Evidence set C
  - Two sets of variables A and B
- Automated way to check independence of A and B given C?
- D-Separation, [Pearl, 1988]
- Based on the three canonical models



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

In general

$$p(x_2, x_3) \neq p(x_2)p(x_3)$$
$$x_2 \not\perp x_3|\emptyset$$

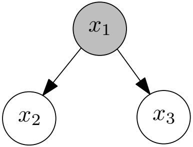
tail-to-tail node

Common parent. Example:

- x_2 : Shoe size, x_3 : Amount of gray hair, x_1 : Age

Bayesian networks

Conditional independence: canonical models (1/3)



$$p(x_1, x_2, x_3) = p(x_2|x_1)p(x_3|x_1)p(x_1)$$

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

$$p(x_2, x_3|x_1) = p(x_2|x_1)p(x_3|x_1)$$

Therefore

$$x_2 \perp\!\!\!\perp x_3 | x_1$$

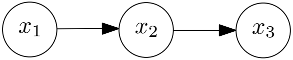
tail-to-tail node

Common parent. Example:

- x_2 : Shoe size, x_3 : Amount of gray hair, x_1 : Age
- Hidden variable explains the observed dependence between x_2 and x_3

Bayesian networks

Conditional independence: canonical models (2/3)



$$p(x_1, x_2, x_3) = p(x_1)p(x_2|x_1)p(x_3|x_2)$$

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

$$= p(x_1)p(x_3|x_1)$$

In general

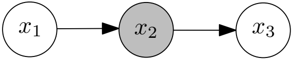
$$p(x_1, x_3) \neq p(x_1)p(x_3)$$

$$x_1 \not\perp\!\!\!\perp x_3|\emptyset$$

head-to-tail node

Markov chain. Example:

- x_1 : Past, x_2 : Present, x_3 : Future



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

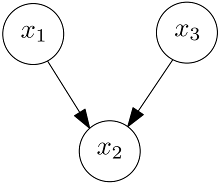
Therefore

$$x_1 \perp\!\!\!\perp x_3 | x_2$$

head-to-tail node

Markov chain. Example:

- x_1 : Past, x_2 : Present, x_3 : Future
- Given the present, past is independent of future



$$p(x_1, x_2, x_3) = p(x_1)p(x_3)p(x_2|x_1, x_3)$$

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

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

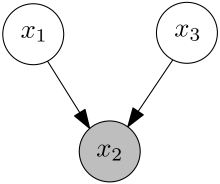
$$p(x_1, x_3) = p(x_1)p(x_3)$$

Therefore $x_1 \perp\!\!\!\perp x_3|\emptyset$

head-to-head node

Multiple parents. "Explaining away" phenomenon:

- x_1 : Battery, x_2 : Sensor, x_3 : Fuel Tank
- Battery and Fuel Tank are marginally unrelated



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

$$p(x_1, x_3|x_2) \neq p(x_1|x_2)p(x_3|x_2)$$

Therefore $\quad x_3 \not\perp\!\!\!\perp x_1 | x_2$

head-to-head node

Multiple parents. "Explaining away" phenomenon:

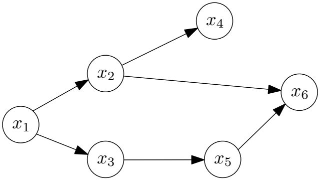
- x_1 : Battery, x_2 : Sensor, x_3 : Fuel Tank
- Battery and Fuel Tank become related once we observe sensor

D-Separation

1. A, B and C non-intersecting subsets of nodes
2. A path from A to B is blocked if it contains a node such that
   - It is a head-to-tail or tail-to-tail node and the node is in C
   - It is a head-to-head node and neither the node, **nor any of its descendants** are in C
3. If all paths from A and B are blocked, A is d-separated from B by C
4. Then $A \perp\!\!\!\perp B | C$

D-Separation

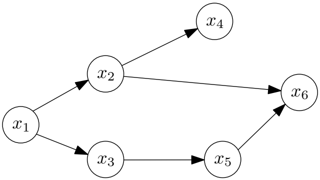
1. A, B and C non-intersecting subsets of nodes
2. A path from A to B is blocked if it contains a node such that
   - It is a head-to-tail or tail-to-tail node and the node is in C
   - It is a head-to-head node and neither the node, **not any of its descendants** are in C
3. If all paths from A and B are blocked, A is d-separated from B by C
4. Then $A \perp\!\!\!\perp B|C$



$x_2 \perp\!\!\!\perp x_3|\emptyset$??

D-Separation

1. A, B and C non-intersecting subsets of nodes
2. A path from A to B is blocked if it contains a node such that
   - It is a head-to-tail or tail-to-tail node and the node is in C
   - It is a head-to-head node and neither the node, **not any of its descendants** are in C
3. If all paths from A and B are blocked, A is d-separated from B by C
4. Then $A \perp\!\!\!\perp B|C$

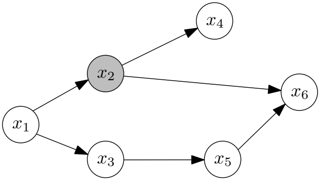


$$x_2 \perp\!\!\!\perp x_3|\emptyset??$$

NO!! path through x_1 is not blocked

D-Separation

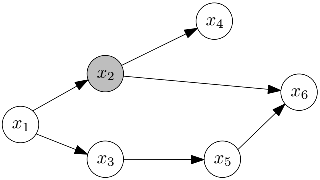
1. A, B and C non-intersecting subsets of nodes
2. A path from A to B is blocked if it contains a node such that
   - It is a head-to-tail or tail-to-tail node and the node is in C
   - It is a head-to-head node and neither the node, **not any of its descendants** are in C
3. If all paths from A and B are blocked, A is d-separated from B by C
4. Then $A \perp\!\!\!\perp B|C$



$$x_4 \perp\!\!\!\perp \{x_1, x_3\}|x_2??$$

D-Separation

1. A, B and C non-intersecting subsets of nodes
2. A path from A to B is blocked if it contains a node such that
   - It is a head-to-tail or tail-to-tail node and the node is in C
   - It is a head-to-head node and neither the node, **not any of its descendants** are in C
3. If all paths from A and B are blocked, A is d-separated from B by C
4. Then $A \perp\!\!\!\perp B|C$

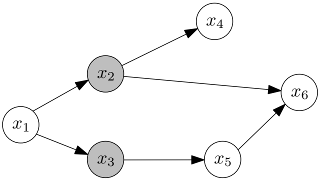


$$x_4 \perp\!\!\!\perp \{x_1, x_3\}|x_2??$$

YES!! paths through x_2 are blocked

D-Separation

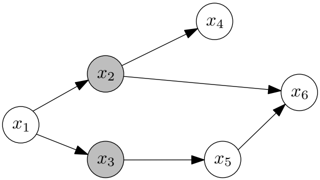
1. A, B and C non-intersecting subsets of nodes
2. A path from A to B is blocked if it contains a node such that
   - It is a head-to-tail or tail-to-tail node and the node is in C
   - It is a head-to-head node and neither the node, **not any of its descendants** are in C
3. If all paths from A and B are blocked, A is d-separated from B by C
4. Then $A \perp\!\!\!\perp B | C$



$$x_1 \perp\!\!\!\perp x_6 | \{x_2, x_3\} ??$$

D-Separation

1. A, B and C non-intersecting subsets of nodes
2. A path from A to B is blocked if it contains a node such that
   - It is a head-to-tail or tail-to-tail node and the node is in C
   - It is a head-to-head node and neither the node, **not any of its descendants** are in C
3. If all paths from A and B are blocked, A is d-separated from B by C
4. Then $A \perp\!\!\!\perp B | C$

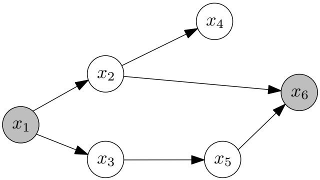


$$x_1 \perp\!\!\!\perp x_6 | \{x_2, x_3\}??$$

YES!! all two paths are blocked

D-Separation

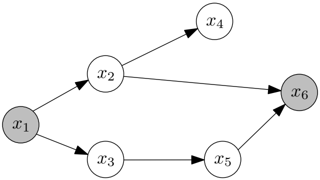
1. A, B and C non-intersecting subsets of nodes
2. A path from A to B is blocked if it contains a node such that
   - It is a head-to-tail or tail-to-tail node and the node is in C
   - It is a head-to-head node and neither the node, **not any of its descendants** are in C
3. If all paths from A and B are blocked, A is d-separated from B by C
4. Then $A \perp\!\!\!\perp B|C$



$$x_2 \perp\!\!\!\perp x_3|\{x_1, x_6\}??$$

D-Separation

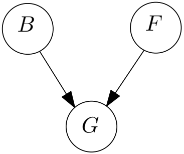
1. A, B and C non-intersecting subsets of nodes
2. A path from A to B is blocked if it contains a node such that
   - It is a head-to-tail or tail-to-tail node and the node is in C
   - It is a head-to-head node and neither the node, **not any of its descendants** are in C
3. If all paths from A and B are blocked, A is d-separated from B by C
4. Then $A \perp\!\!\!\perp B | C$



$$x_2 \perp\!\!\!\perp x_3 | \{x_1, x_6\}??$$

NO!! path through x_6 is opened!

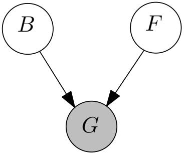
Example: B battery, F fuel tank and G fuel electric sensor



$$P(G=1|B=1,F=1)=0.8$$
$$P(G=1|B=1,F=0)=0.2$$
$$P(G=1|B=0,F=1)=0.2$$
$$P(G=1|B=0,F=0)=0.1$$
$$P(B=1)=0.9$$
$$P(F=1)=0.9$$

Without evidence, the prior probability of the tank being empty is $P(F=0)=0.1$

Example: B battery, F fuel tank and G fuel electric sensor



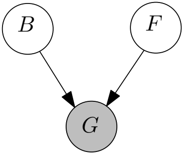
$$P(G=1|B=1,F=1)=0.8$$
$$P(G=1|B=1,F=0)=0.2$$
$$P(G=1|B=0,F=1)=0.2$$
$$P(G=1|B=0,F=0)=0.1$$
$$P(B=1)=0.9$$
$$P(F=1)=0.9$$

Observe sensor $G=0$. What is the probability of the tank being empty?

Bayesian networks

Example of inference

Example: B battery, F fuel tank and G fuel electric sensor

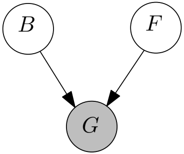


$$P(G=1|B=1,F=1)=0.8$$
$$P(G=1|B=1,F=0)=0.2$$
$$P(G=1|B=0,F=1)=0.2$$
$$P(G=1|B=0,F=0)=0.1$$
$$P(B=1)=0.9$$
$$P(F=1)=0.9$$

$$P(G=0)=\sum_{B=\{0,1\}}\sum_{F=\{0,1\}} p(G=0|B,F)p(B)p(F)=0.315$$

$$P(G=0|F=0)=\sum_{B=\{0,1\}} p(G=0|B,F=0)p(B)p(B)=0.81$$

Example: B battery, F fuel tank and G fuel electric sensor

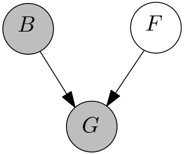


$$P(G=1|B=1,F=1)=0.8$$
$$P(G=1|B=1,F=0)=0.2$$
$$P(G=1|B=0,F=1)=0.2$$
$$P(G=1|B=0,F=0)=0.1$$
$$P(B=1)=0.9$$
$$P(F=1)=0.9$$

$$P(F=0|G=0)=\frac{P(G=0|F=0)P(F=0)}{P(G=0)}\approx 0.257$$

$$P(F=0|G=0)>P(F=0)$$

Example: B battery, F fuel tank and G fuel electric sensor

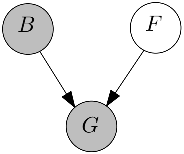


$$P(G=1|B=1,F=1)=0.8$$
$$P(G=1|B=1,F=0)=0.2$$
$$P(G=1|B=0,F=1)=0.2$$
$$P(G=1|B=0,F=0)=0.1$$
$$P(B=1)=0.9$$
$$P(F=1)=0.9$$

Suppose that we check the battery and it is flat $B=0$. What is the new probability of the fuel being empty?

$$P(F=0|G=0,B=0)=?$$

Example: B battery, F fuel tank and G fuel electric sensor

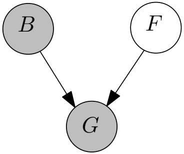


$$P(G=1|B=1,F=1)=0.8$$
$$P(G=1|B=1,F=0)=0.2$$
$$P(G=1|B=0,F=1)=0.2$$
$$P(G=1|B=0,F=0)=0.1$$
$$P(B=1)=0.9$$
$$P(F=1)=0.9$$

Suppose that we check the battery and it is flat $B=0$. What is the new probability of the fuel being empty?

$$P(F=0|G=0,B=0)=\frac{P(G=0|B=0,F=0)P(F=0)}{\sum_{F=\{0,1\}}P(G=0|B=0,F)P(F)}\approx 0.111$$

Example: B battery, F fuel tank and G fuel electric sensor

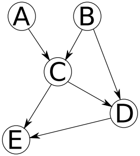


$$P(G=1|B=1,F=1)=0.8$$
$$P(G=1|B=1,F=0)=0.2$$
$$P(G=1|B=0,F=1)=0.2$$
$$P(G=1|B=0,F=0)=0.1$$
$$P(B=1)=0.9$$
$$P(F=1)=0.9$$

Suppose that we check the battery and it is flat $B=0$. What is the new probability of the fuel being empty?

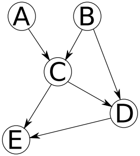
$$P(F=0|G=0,B=0)=\frac{P(G=0|B=0,F=0)P(F=0)}{\sum_{F=\{0,1\}}P(G=0|B=0,F)P(F)}\approx 0.111$$

$$P(F=0|G=0,B=0)<P(F=0|G=0) \qquad \mathbf{F} \not\perp\!\!\!\perp \mathbf{B}|\mathbf{G}$$



$$p(A, B, C, D, E) =$$
$$p(A)p(B)p(C|A, B)p(D|B, C)p(E|C, D)$$

Inference: Evaluate the probability distribution over some set of variables, given values of another set of variables Ex: $p(A|C = c)$? (binary variables)



$$p(A, B, C, D, E) = p(A)p(B)p(C|A, B)p(D|B, C)p(E|C, D)$$

Inference: Evaluate the probability distribution over some set of variables, given values of another set of variables Ex: $p(A|C = c)$? (binary variables)

Naive:

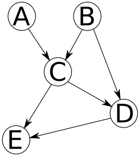
$$p(A, C = c) = \sum_{B,D,E} p(A, B, C = c, D, E) \qquad \text{[16 terms]}$$

$$p(C = c) = \sum_{A} p(A, C = c) \qquad \text{[2 terms]}$$

$$p(A|C = c) = \frac{p(A, C = c)}{p(C = c)} \qquad \text{[2 terms]} \qquad \rightarrow \text{total terms: } 20$$

Bayesian networks

Inference



$$p(A,B,C,D,E) =$$
$$p(A)p(B)p(C|A,B)p(D|B,C)p(E|C,D)$$

Inference: Evaluate the probability distribution over some set of variables, given values of another set of variables Ex: $p(A|C=c)$? (binary variables)

More efficiently:

$$p(A, C=c) = \sum_{B,D,E} p(A)p(B)p(C=c|A,B)p(D|B,C=c)p(E|C=c,D)$$
$$= \sum_{B} p(A)p(B)p(C=c|A,B) \sum_{D} p(D|B,C=c) \sum_{E} p(E|C=$$
$$= \sum_{B} p(A)p(B)p(C=c|A,B) \qquad [4 \text{ terms}]$$

Outline

1. Introduction to probabilistic Graphical Models
   - Bayesian networks
   - Markov Random Fields
   - Factor graphs and the Sum-Product algorithm

2. Learning Graphical Models

Factorization : over maximal cliques in the graph

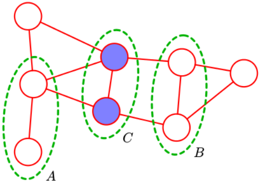
$$p(\mathbf{x}) = \frac{1}{Z} \prod_C \psi_C(\mathbf{x}_C) \qquad Z = \sum_{\mathbf{x}} \prod_C \psi_C(\mathbf{x}_C)$$

where $\psi_C(\mathbf{x}_C)$ is the potential over clique C and Z is the partition function (exponentially large sum)

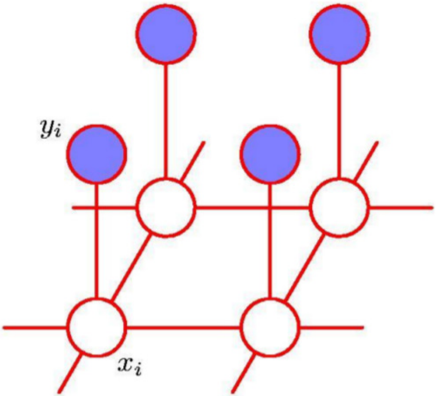
Exponential representation : $\psi_C(\mathbf{x}_C) = \exp(-E(\mathbf{x}_C))$

Independences Easier! If A and B become disconnected after removing C

$$A \perp\!\!\!\perp B | C$$



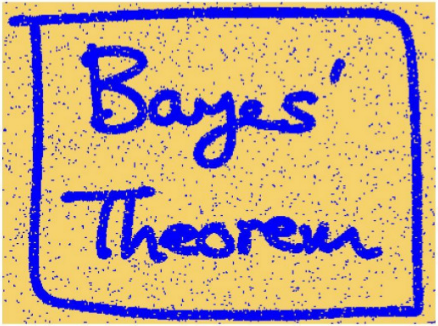
Example: image denoising



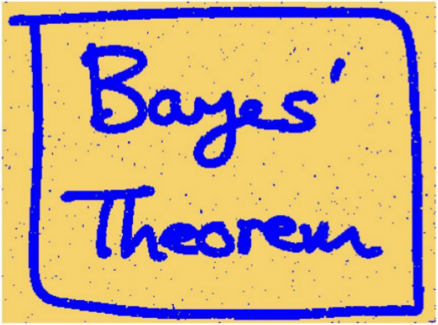
$$E(\mathbf{x},\mathbf{y}) = h\sum_i x_i - \beta \sum_{\{i,j\}} x_i x_j - \eta \sum_i x_i y_i$$

$$p(\mathbf{x},\mathbf{y}) = \frac{1}{Z}\exp\{-E(\mathbf{x},\mathbf{y})\}$$

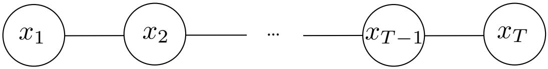
Example: image denoising



Restored Image (ICM)



Restored Image (Graph cuts)

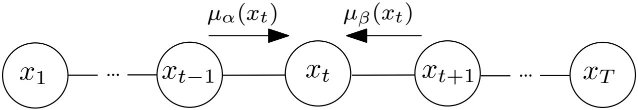


Joint probability distribution :

$$p(\mathbf{x}) = \frac{1}{Z}\psi_{1,2}(x_1, x_2)\psi_{2,3}(x_2, x_3)\ldots\psi_{T-1,T}(x_{T-1}, x_T)$$

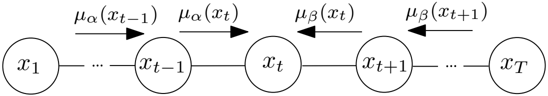
Estimate single-node marginal $p(x_t)$:

$$p(x_t) = \sum_{x_1}\ldots\sum_{x_{t-1}}\sum_{x_{t+1}}\ldots\sum_{x_T} p(\mathbf{x})$$



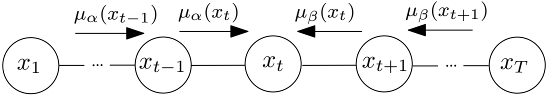
$$p(x_t) = \frac{1}{Z} \underbrace{\left[\sum_{x_{t-1}} \psi_{t-1,t}(x_{t-1}, x_t) \ldots \left[\sum_{x_1} \psi_{1,2}(x_1, x_2)\right] \ldots\right]}_{\mu_\alpha(x_t)} \cdot$$

$$\underbrace{\left[\sum_{x_{t+1}} \psi_{t,t+1}(x_t, x_{t+1}) \ldots \left[\sum_{x_T} \psi_{T-1,T}(x_{T-1}, x_T)\right] \ldots\right]}_{\mu_\beta(x_t)}$$



$$\mu_\alpha(x_t) = \sum_{x_{t-1}} \psi_{t-1,t}(x_{t-1}, x_t) \left[\sum_{x_{t-2}} \dots \right]$$
$$= \sum_{x_{t-1}} \psi_{t-1,t}(x_{t-1}, x_t) \mu_\alpha(x_{t-1})$$

$$\mu_\beta(x_t) = \sum_{x_{t+1}} \psi_{t,t+1}(x_t, x_{t+1}) \left[\sum_{x_{t+2}} \dots \right]$$
$$= \sum_{x_{t+1}} \psi_{t,t+1}(x_t, x_{t+1}) \mu_\beta(x_{t+1})$$



$$\mu_\alpha(x_2) = \sum_{x_1} \psi_{1,2}(x_1, x_2) \qquad \mu_\alpha(x_{T-1}) = \sum_{x_T} \psi_{T-1,T}(x_{T-1}, x_T)$$

$$Z = \sum_{x_t} \mu_\alpha(x_t)\mu_\beta(x_t)$$

Computing local marginals in a chain

1. Compute forward messages $\mu_\alpha(x_t)$
2. Compute backward messages $\mu_\beta(x_t)$
3. Compute Z at any node x_t
4. Compute $p(x_t) = \frac{1}{Z}\mu_\alpha(x_t)\mu_\beta(x_t)$

Outline

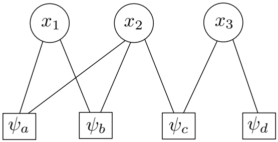
1. Introduction to probabilistic Graphical Models
   - Bayesian networks
   - Markov Random Fields
   - Factor graphs and the Sum-Product algorithm

2. Learning Graphical Models

Bipartite Factor Graphs

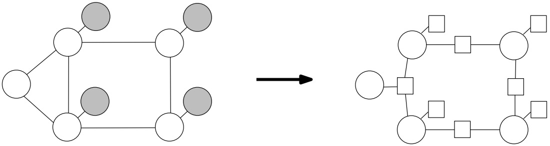
General class of graphical models

Factor graphs subsume both Bayesian networks and MRFs



Factorization: $p(\mathbf{x}) = \psi_a(x_1, x_2)\psi_b(x_1, x_2)\psi_c(x_2, x_3)\psi_d(x_3)$

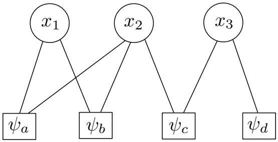
- MRF: factors correspond to maximal cliques potentials $\mathbf{x}_s$



Bipartite Factor Graphs

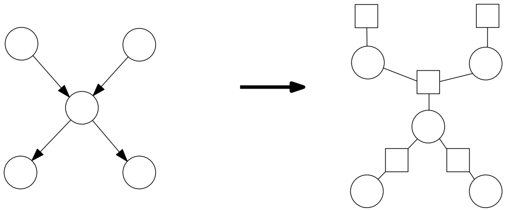
General class of graphical models

Factor graphs subsume both Bayesian networks and MRFs



Factorization: $p(\mathbf{x}) = \psi_a(x_1, x_2)\psi_b(x_1, x_2)\psi_c(x_2, x_3)\psi_d(x_3)$

- BN: factors correspond to conditional probability tables



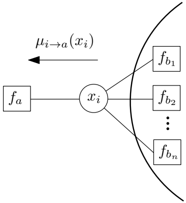
Sum-Product (belief propagation) algorithm

- Generic algorithm to compute local marginals in a factor graph
- Rediscovered several times: Gallager, J. Pearl, Kalman, ...

Iterates the following messages:

variable to factor :

$$\mu_{i\to a}(x_i) = \prod_{b\in\mathcal{N}(i)\setminus a} \mu_{b\to i}(x_i)$$



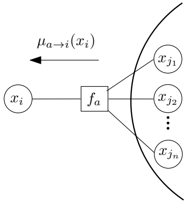
Sum-Product (belief propagation) algorithm

- Generic algorithm to compute local marginals in a factor graph
- Rediscovered several times: Gallager, J. Pearl, Kalman, ...

Iterates the following messages:

factor to variable

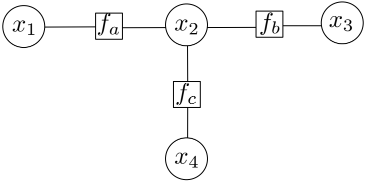
$$\mu_{a\to i}(x_i) = \sum_{\mathbf{x}_a \setminus \{i\}} f_a(\mathbf{x}_a) \prod_{j\in\mathcal{N}(a)\setminus i} \mu_{j\to a}(x_j)$$



Bipartite factor graphs

Inference

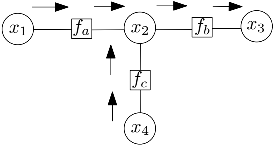
Example of inference using Belief Propagation



$$p(\mathbf{x}) = f_a(x_1, x_2) f_b(x_2, x_3) f_c(x_2, x_4)$$

Bipartite factor graphs

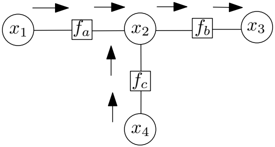
Example of inference using Belief Propagation



$$\mu_{x_1 \to f_a}(x_1) = 1$$

Bipartite factor graphs

Example of inference using Belief Propagation

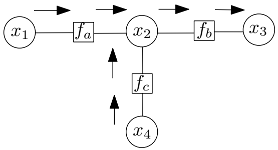


$$\mu_{x_1 \to f_a}(x_1) = 1$$

$$\mu_{f_a \to x_2}(x_2) = \sum_{x_1} f_a(x_1, x_2)$$

Bipartite factor graphs

Example of inference using Belief Propagation



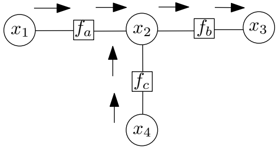
$$\mu_{x_1 \to f_a}(x_1) = 1$$

$$\mu_{f_a \to x_2}(x_2) = \sum_{x_1} f_a(x_1, x_2)$$

$$\mu_{x_4 \to f_c}(x_4) = 1$$

Bipartite factor graphs

Example of inference using Belief Propagation



$$\mu_{x_1 \to f_a}(x_1) = 1$$

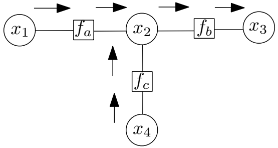
$$\mu_{f_a \to x_2}(x_2) = \sum_{x_1} f_a(x_1, x_2)$$

$$\mu_{x_4 \to f_c}(x_4) = 1$$

$$\mu_{f_c \to x_2}(x_2) = \sum_{x_4} f_c(x_2, x_4)$$

Bipartite factor graphs

Example of inference using Belief Propagation



$$\mu_{x_1 \to f_a}(x_1) = 1$$

$$\mu_{f_a \to x_2}(x_2) = \sum_{x_1} f_a(x_1, x_2)$$

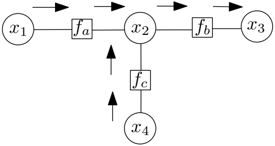
$$\mu_{x_4 \to f_c}(x_4) = 1$$

$$\mu_{f_c \to x_2}(x_2) = \sum_{x_4} f_c(x_2, x_4)$$

$$\mu_{x_2 \to f_b}(x_2) = \mu_{f_a \to x_2}(x_2)\mu_{f_c \to x_2}(x_2)$$

Bipartite factor graphs

Example of inference using Belief Propagation



$$\mu_{x_1 \to f_a}(x_1) = 1$$

$$\mu_{f_a \to x_2}(x_2) = \sum_{x_1} f_a(x_1, x_2)$$

$$\mu_{x_4 \to f_c}(x_4) = 1$$

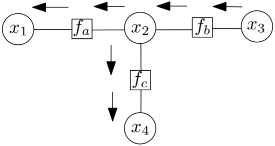
$$\mu_{f_c \to x_2}(x_2) = \sum_{x_4} f_c(x_2, x_4)$$

$$\mu_{x_2 \to f_b}(x_2) = \mu_{f_a \to x_2}(x_2)\mu_{f_c \to x_2}(x_2)$$

$$\mu_{f_b \to x_3}(x_3) = \sum_{x_2} f_b(x_2, x_3)\mu_{x_2 \to f_b}(x_2)$$

Bipartite factor graphs

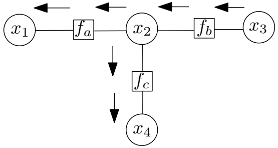
Example of inference using Belief Propagation



$$\mu_{x_3 \to f_b}(x_3) = 1$$

Bipartite factor graphs

Example of inference using Belief Propagation

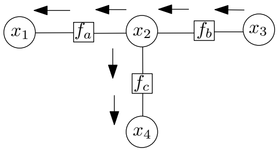


$$\mu_{x_3 \to f_b}(x_3) = 1$$

$$\mu_{f_b \to x_2}(x_2) = \sum_{x_3} f_b(x_2, x_3)$$

Bipartite factor graphs

Example of inference using Belief Propagation



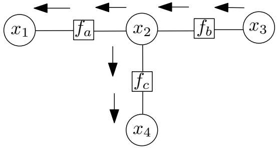
$$\mu_{x_3 \to f_b}(x_3) = 1$$

$$\mu_{f_b \to x_2}(x_2) = \sum_{x_3} f_b(x_2, x_3)$$

$$\mu_{x_2 \to f_a}(x_2) = \mu_{f_b \to x_2}(x_2)\mu_{f_c \to x_2}(x_2)$$

Bipartite factor graphs

Example of inference using Belief Propagation



$$\mu_{x_3 \to f_b}(x_3) = 1$$

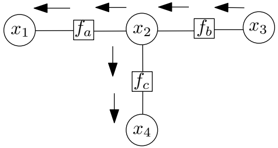
$$\mu_{f_b \to x_2}(x_2) = \sum_{x_3} f_b(x_2, x_3)$$

$$\mu_{x_2 \to f_a}(x_2) = \mu_{f_b \to x_2}(x_2) \mu_{f_c \to x_2}(x_2)$$

$$\mu_{f_a \to x_1}(x_1) = \sum_{x_2} f_a(x_1, x_2) \mu_{x_2 \to f_a}(x_2)$$

Bipartite factor graphs

Example of inference using Belief Propagation



$$\mu_{x_3 \to f_b}(x_3) = 1$$

$$\mu_{f_b \to x_2}(x_2) = \sum_{x_3} f_b(x_2, x_3)$$

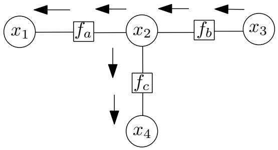
$$\mu_{x_2 \to f_a}(x_2) = \mu_{f_b \to x_2}(x_2) \mu_{f_c \to x_2}(x_2)$$

$$\mu_{f_a \to x_1}(x_1) = \sum_{x_2} f_a(x_1, x_2) \mu_{x_2 \to f_a}(x_2)$$

$$\mu_{x_2 \to f_c}(x_2) = \mu_{f_a \to x_2}(x_2) \mu_{f_b \to x_2}(x_2)$$

Bipartite factor graphs

Example of inference using Belief Propagation



$$\mu_{x_3 \to f_b}(x_3) = 1$$

$$\mu_{f_b \to x_2}(x_2) = \sum_{x_3} f_b(x_2, x_3)$$

$$\mu_{x_2 \to f_a}(x_2) = \mu_{f_b \to x_2}(x_2)\mu_{f_c \to x_2}(x_2)$$

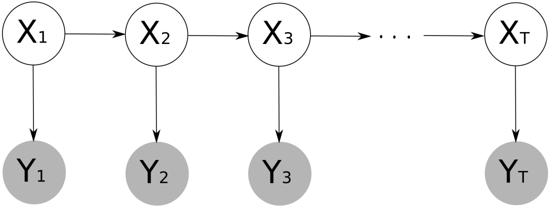
$$\mu_{f_a \to x_1}(x_1) = \sum_{x_2} f_a(x_1, x_2)\mu_{x_2 \to f_a}(x_2)$$

$$\mu_{x_2 \to f_c}(x_2) = \mu_{f_a \to x_2}(x_2)\mu_{f_b \to x_2}(x_2)$$

$$\mu_{f_c \to x_4}(x_4) = \sum_{x_2} f_c(x_2, x_4)\mu_{x_2 \to f_c}(x_2)$$

Probabilistic Inference

Hidden Markov models and Linear Gaussian state-space models



$$p(X_{1,\ldots,T}, Y_{1,\ldots,T}) = p(X_1)p(Y_1|X_1)\prod_{t=2}^{T} p(X_t|X_{t-1})p(Y_t|X_t)$$

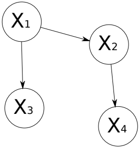
- In HMMs, the states X_t are discrete
- In linear Gaussian SSMs, the states are real Gaussian vectors
- Both HMMs and SSMs can be represented as singly connected DAGs
- The **forward-backward algorithm** in HMMs and the **Kalman smoothing algorithm** in SSMs are both instances of belief propagation / factor graph representation

Sum-Product algorithm

- Generic algorithm to compute local marginals in a factor graph
- Sum-Product is exact on tree graphs
- Can be an approximate algorithm on loopy graphs (LBP)
- Convergence is not guaranteed
- Variational interpretation: fixed points of LBP are stationary points of a free energy function
- Exact inference in loopy graphs
  - Compile the graph into a tree (cluster graph)
  - Run message passing on it
  - Complexity exponential in maximum clique size

Learning graphical Models

Learning the parameters



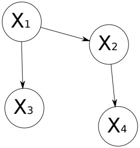
$$p(X_1)p(X_2|X_1)p(X_3|X_1)p(X_4|X_3)$$

θ_2	X_2		
X_1	0.2	0.3	0.5
	0.1	0.6	0.3

- Assume each variable X_i is discrete and can take on K_i values
- The parameters can be represented as 4 tables: θ_1 has K_1, θ_2 has entries $K_1 \times K_2$, etc...
- **Conditional Probability Tables** (CPTs) with the following semantics:

$$p(x_1 = k) = \theta_{1,k}, \qquad p(x_2 = k'|x_1 = k) = \theta_{2,k,k'}$$

- If node i has M parents, θ_i: $M+1$ dimensional table
  or 2-dimensional table with $(\prod_{j \in \text{pa}(i)} K_j \times K_i)$ entries by collapsing all the states of the parents of node i. Note that $\sum_{k'} \theta_{i,k,k'} = 1$
- Assume a data set $\mathcal{D} = \{\mathbf{x}^n\}_{n=1}^N$



$$p(X_1)p(X_2|X_1)p(X_3|X_1)p(X_4|X_3)$$

θ_2	X_2		
X_1	0.2	0.3	0.5
	0.1	0.6	0.3

- Assume each variable X_i is discrete and can take on K_i values
- The parameters can be represented as 4 tables: θ_1 has K_1, θ_2 has entries $K_1 \times K_2$, etc...
- **Conditional Probability Tables** (CPTs) with the following semantics:

$$p(x_1 = k) = \theta_{1,k}, \qquad p(x_2 = k'|x_1 = k) = \theta_{2,k,k'}$$

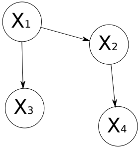
- If node i has M parents, θ_i: $M+1$ dimensional table
  or 2-dimensional table with $(\prod_{j \in \mathrm{pa}(i)} K_j \times K_i)$ entries by collapsing all the states of the parents of node i. Note that $\sum_{k'} \theta_{i,k,k'} = 1$
- Assume a data set $\mathcal{D} = \{\mathbf{x}^n\}_{n=1}^N$ How do we learn θ from $\mathcal{D}$?

Learning graphical Models

Learning the parameters

Assume a data set $\mathcal{D} = \{\boldsymbol{x}^n\}_{n=1}^N$

How do we learn θ from $\mathcal{D}$?



$$p(\boldsymbol{x}|\boldsymbol{\theta}) = p(x_1|\theta_1)p(x_2|x_1,\theta_2)p(x_3|x_1,\theta_3)p(x_4|x_3,\theta_4)$$

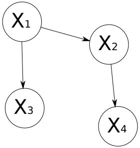
- Likelihood: $p(\mathcal{D}|\boldsymbol{\theta}) = \prod_{n=1}^N p(\boldsymbol{x}^{(n)}|\boldsymbol{\theta})$
- Log-Likelihood: $\log p(\mathcal{D}|\boldsymbol{\theta}) = \sum_{n=1}^N \sum_i \log p(x_i^{(n)}|x_{\text{pa(i)}}^{(n)}, \theta_i)$
- This decomposes into sum of functions of θ_i (optimized separately)

$$\theta_{i,k,k'} = \frac{n_{i,k,k'}}{\sum_{k''} n_{i,k,k''}}$$

$n_{i,k,k''}$ is # times in $\mathcal{D}$ where $x_i = k'$ and $x_{\text{pa(i)}} = k$ (k joint configuration of the parents)

Assume a data set $\mathcal{D} = \{\boldsymbol{x}^n\}_{n=1}^N$

How do we learn θ from $\mathcal{D}$?



$$p(\boldsymbol{x}|\boldsymbol{\theta}) = p(x_1|\theta_1)p(x_2|x_1,\theta_2)p(x_3|x_1,\theta_3)p(x_4|x_3,\theta_4)$$

- Likelihood: $p(\mathcal{D}|\boldsymbol{\theta}) = \prod_{n=1}^N p(\boldsymbol{x}^{(n)}|\boldsymbol{\theta})$
- Log-Likelihood: $\log p(\mathcal{D}|\boldsymbol{\theta}) = \sum_{n=1}^N \sum_i \log p(x_i^{(n)}|x_{\mathsf{pa(i)}}^{(n)}, \theta_i)$
- This decomposes into sum of functions of θ_i (optimized separately)

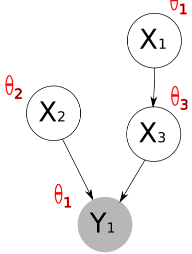
$$\theta_{i,k,k'} = \frac{n_{i,k,k'}}{\sum_{k''} n_{i,k,k''}}$$

$n_{i,k,k''}$ is # times in $\mathcal{D}$ where $x_i = k'$ and $x_{\mathsf{pa(i)}} = k$ (k joint configuration of the parents)

ML solution: Simply calculate frequencies!

Goal : Maximize parameter log-likelihood given observables

$$\mathcal{L}(\theta) = \log p(Y|\theta) = \log \sum_X p(Y, X|\theta)$$



The Expectation - Maximization (EM) algorithm (intuition)

Iterate between applying the following two steps:

- **The E-Step**: fill-in the hidden/missing variables
- **The M-Step**: apply complete data learning to filled-in data. Previous slide formula