Graphical models

- Represent complex multivariate probabilistic models
  - multivariate -> multiple random variables
    \[ P(X) = P(X_1, X_2, \ldots, X_d) \]
    \[ p(X) = p(X_1, X_2, \ldots, X_d) \]

- Parametric distribution models:
  - Bernoulli (outcome of coin flip)
  - Binomial (outcome of multiple coin flips)
  - Multinomial (outcome of die)
  - Poisson
  - Exponential
  - Gamma distribution
  - Gaussian (this one is multivariate)
Challenges for modeling complex multivariate distributions

How to model/parameterize complex multivariate distributions $P(X)$ with a large number of variables?

**One solution:**
- Decompose the distribution. Reduce the number of parameters, using some form of independence.

**Two graphical models:**
- Bayesian belief networks (BBNs)
- Markov Random Fields (MRFs)

• Learning of these models relies on the decomposition.

Bayesian belief network

**Directed acyclic graph**
- **Nodes** = random variables
- **Links** = direct (causal) dependencies
  Missing links encode different marginal and conditional independences
Graphical structure and independences

1. JohnCalls is independent of Burglary given Alarm
   \[ P(J \mid A, B) = P(J \mid A) \]
   \[ P(J, B \mid A) = P(J \mid A)P(B \mid A) \]

Independences in BBNs

2. Burglary is independent of Earthquake (not knowing Alarm)
   Burglary and Earthquake become dependent given Alarm !!
   \[ P(B, E) = P(B)P(E) \]
Independences in BBNs

3. MaryCalls is independent of JohnCalls given Alarm

\[
P(J \mid A, M) = P(J \mid A)
\]
\[
P(J, M \mid A) = P(J \mid A)P(M \mid A)
\]

Bayesian belief network: parameters

2. Local conditional distributions
   • relate variables and their parents \( P(v \mid pa(v)) \)
Bayesian belief network

Full joint distribution in BBNs

The full joint distribution is defined as a product of local conditional distributions:

\[ P(X_1, X_2, ..., X_n) = \prod_{i=1}^{n} P(X_i \mid pa(X_i)) \]

Example:

Assume the following assignment of values to random variables:

\[ B = T, E = T, A = T, J = T, M = F \]

Then its probability is:

\[
P(B = T, E = T, A = T, J = T, M = F) =
\]

\[
P(B = T)P(E = T)P(A = T \mid B = T, E = T)P(J = T \mid A = T)P(M = F \mid A = T)
\]
Parameter complexity problem

- In the BBN the **full joint distribution** is defined as:
  \[
P(X_1, X_2, \ldots, X_n) = \prod_{i=1}^{n} P(X_i | pa(X_i))
\]
- What did we save?

Alarm example: binary (True, False) variables

# of parameters of the full joint:

\[
2^5 = 32
\]

One parameter depends on the rest:

\[
2^5 - 1 = 31
\]

# of parameters of the BBN:

\?
Bayesian belief network: parameters count

<table>
<thead>
<tr>
<th></th>
<th>True</th>
<th>False</th>
</tr>
</thead>
<tbody>
<tr>
<td>Burglary</td>
<td>P(B)</td>
<td>T</td>
</tr>
<tr>
<td>Earthquake</td>
<td>P(E)</td>
<td>T</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>True</th>
<th>False</th>
</tr>
</thead>
<tbody>
<tr>
<td>JohnCalls</td>
<td>P(J</td>
<td>A)</td>
</tr>
<tr>
<td>MaryCalls</td>
<td>P(M</td>
<td>A)</td>
</tr>
</tbody>
</table>

Total: 20

Parameter complexity problem

- In the BBN the **full joint distribution** is defined as:
  \[ P(X_1, X_2, \ldots, X_n) = \prod_{i=1}^{n} P(X_i \mid pa(X_i)) \]
- What did we save?
  Alarm example: 5 binary (True, False) variables

  # of parameters of the full joint:
  \[ 2^5 = 32 \]

  One parameter depends on the rest:
  \[ 2^5 - 1 = 31 \]

  # of parameters of the BBN:
  \[ 2^3 + 2(2^2) + 2(2) = 20 \]

  One parameter in every conditional depends on the rest:
Bayesian belief network: free parameters

Parameter complexity problem

- In the BBN the full joint distribution is defined as:
  $$P(X_1, X_2, \ldots, X_n) = \prod_{i=1}^{n} P(X_i \mid pa(X_i))$$

- What did we save?

Alarm example: 5 binary (True, False) variables

# of parameters of the full joint:
$$2^5 = 32$$

One parameter depends on the rest:
$$2^5 - 1 = 31$$

# of parameters of the BBN:
$$2^3 + 2(2^2) + 2(2) = 20$$

One parameter in every conditional depends on the rest:
$$2^2 + 2(2) + 2(1) = 10$$
Inference in Bayesian network

- **Bad news:**
  - Exact inference problem in BBNs is NP-hard (Cooper)
  - Approximate inference is NP-hard (Dagum, Luby)
- **But** very often we can achieve significant improvements
- Assume our Alarm network

![Alarm network diagram](image)

- Assume we want to compute: \( P(J = T) \)

Inference in Bayesian networks

- Full joint uses the decomposition
- **Calculation of marginals:**
  - Requires summation over variables we want to take out
  \[
P(J = T) = \\
= \sum_{b \in T, F} \sum_{e \in T, F} \sum_{a \in T, F} \sum_{m \in T, F} P(B = b, E = e, A = a, J = T, M = m)
\]

- How to compute sums and products more efficiently?
  \[
\sum_s af(x) = a \sum_s f(x)
\]
Variable elimination

- **Variable elimination:**
  - E.g. Query \( P(J = T) \) requires to eliminate A,B,E,M and this can be done in different order

\[
P(J = T) = \sum_{b \in T, F} \sum_{e \in T, F} \sum_{a \in T, F} \sum_{m \in T, F} P(J = T \mid A = a) P(M = m \mid A = a) P(A = a \mid B = b, E = e) P(B = b) P(E = e)
\]
Variable elimination

Assume order: $M, E, B, A$ to calculate $P(J = T)$

$$
= \sum_{bc1,F} \sum_{ecl,F} \sum_{uc1,F} P(J = T | A = a) P(M = m | A = a) P(A = a | B = b, E = e) P(B = b) P(E = e)
$$

$$
= \sum_{bc1,F} \sum_{ecl,F} \sum_{uc1,F} P(J = T | A = a) P(A = a | B = b, E = e) P(B = b) P(E = e) \left[ \sum_{mcl,F} P(M = m | A = a) \right]
$$
Variable elimination

Assume order: M, E, B, A to calculate \( P(J = T) \)

\[
\begin{align*}
\tau_i(A = a, B = b) &= \\
&= \sum_{a \in F} \sum_{b \in T} \sum_{c \in T, F} \sum_{d \in T, F} \sum_{e \in T, F} P(J = T \mid A = a) P(M = m \mid A = a) P(A = a \mid B = b, E = e) P(B = b) P(E = e) \\
&= \sum_{a \in F} \sum_{b \in T} \sum_{c \in T, F} \sum_{d \in T, F} \sum_{e \in T, F} P(J = T \mid A = a) P(A = a \mid B = b, E = e) P(B = b) P(E = e) \left[ \sum_{m \in T, F} P(M = m \mid A = a) \right] \\
&= \sum_{a \in F} \sum_{b \in T} \sum_{c \in T, F} \sum_{d \in T, F} \sum_{e \in T, F} P(J = T \mid A = a) P(A = a \mid B = b, E = e) P(B = b) P(E = e) \tau_i(A = a, B = b) \\
&= \sum_{a \in F} \sum_{b \in T} \sum_{c \in T, F} \sum_{d \in T, F} \sum_{e \in T, F} \left[ P(A = a \mid B = b, E = e) P(E = e) \right] \\
&= \tau_i(A = a, B = b) = \\
&= \sum_{a \in F} \sum_{b \in T} \sum_{c \in T, F} \sum_{d \in T, F} \sum_{e \in T, F} P(A = a \mid B = b, E = e) P(E = e)
\end{align*}
\]

\( \tau_i(A = a, B = b) \)
Variable elimination

Assume order: M, E, B, A to calculate \( P(J = T) \)

\[
\begin{align*}
&= \sum_{bcT,F} \sum_{ecT,F} \sum_{acT,F} \sum_{m_{ecT,F}} P(J = T \mid A = a) P(M = m \mid A = a) P(A = a \mid B = b, E = e) P(B = b) P(E = e) \\
&= \sum_{bcT,F} \sum_{ecT,F} \sum_{acT,F} P(J = T \mid A = a) P(A = a \mid B = b, E = e) P(B = b) P(E = e) \left[ \sum_{m_{ecT,F}} P(M = m \mid A = a) \right] \\
&= \sum_{bcT,F} \sum_{ecT,F} \sum_{acT,F} P(J = T \mid A = a) P(A = a \mid B = b, E = e) P(B = b) P(E = e) \quad 1 \\
&= \sum_{acT,F} \sum_{bcT,F} P(J = T \mid A = a) P(B = b) \left[ \sum_{ecT,F} P(A = a \mid B = b, E = e) P(E = e) \right] \\
&= \sum_{acT,F} \sum_{bcT,F} P(J = T \mid A = a) P(B = b) \tau_t(A = a, B = b)
\end{align*}
\]

Variable elimination

Assume order: M, E, B, A to calculate \( P(J = T) \)

\[
\begin{align*}
&= \sum_{bcT,F} \sum_{ecT,F} \sum_{acT,F} \sum_{m_{ecT,F}} P(J = T \mid A = a) P(M = m \mid A = a) P(A = a \mid B = b, E = e) P(B = b) P(E = e) \\
&= \sum_{bcT,F} \sum_{ecT,F} \sum_{acT,F} P(J = T \mid A = a) P(A = a \mid B = b, E = e) P(B = b) P(E = e) \left[ \sum_{m_{ecT,F}} P(M = m \mid A = a) \right] \\
&= \sum_{bcT,F} \sum_{ecT,F} \sum_{acT,F} P(J = T \mid A = a) P(A = a \mid B = b, E = e) P(B = b) P(E = e) \quad 1 \\
&= \sum_{acT,F} \sum_{bcT,F} P(J = T \mid A = a) P(B = b) \left[ \sum_{ecT,F} P(A = a \mid B = b, E = e) P(E = e) \right] \\
&= \sum_{acT,F} \sum_{bcT,F} P(J = T \mid A = a) P(B = b) \tau_t(A = a, B = b) \\
&= \sum_{acT,F} P(J = T \mid A = a) \left[ \sum_{bcT,F} P(B = b) \tau_t(A = a, B = b) \right]
\end{align*}
\]
**Variable elimination**

**Assume order:** M, E, B, A to calculate \( P(J = T) \)

\[
\begin{align*}
&= \sum_{a \in T \setminus F} \sum_{b \in T \setminus F} \sum_{c \in T \setminus F} \sum_{m \in T \setminus F} P(J = T | A = a) P(M = m | A = a) P(B = b | A = a, B = b, E = e) P(E = e) \\
&= \sum_{a \in T \setminus F} \sum_{b \in T \setminus F} \sum_{c \in T \setminus F} \sum_{m \in T \setminus F} P(J = T | A = a) P(B = b | A = a, B = b, E = e) P(E = e) P(B = b | A = a, B = b, E = e) \\
&= \sum_{a \in T \setminus F} \sum_{b \in T \setminus F} P(J = T | A = a) P(B = b | A = a, B = b, E = e) P(E = e) \\
&= \sum_{a \in T \setminus F} \sum_{b \in T \setminus F} P(J = T | A = a) \tau_1(A = a, B = b) \\
&= \sum_{a \in T \setminus F} P(J = T | A = a) \tau_1(A = a) \\
\end{align*}
\]

**Variable elimination**

**Assume order:** M, E, B, A to calculate \( P(J = T) \)

\[
\begin{align*}
&= \sum_{a \in T \setminus F} \sum_{b \in T \setminus F} \sum_{c \in T \setminus F} \sum_{m \in T \setminus F} P(J = T | A = a) P(M = m | A = a) P(B = b | A = a, B = b, E = e) P(E = e) \\
&= \sum_{a \in T \setminus F} \sum_{b \in T \setminus F} \sum_{c \in T \setminus F} \sum_{m \in T \setminus F} P(J = T | A = a) P(B = b | A = a, B = b, E = e) P(E = e) P(B = b | A = a, B = b, E = e) \\
&= \sum_{a \in T \setminus F} \sum_{b \in T \setminus F} P(J = T | A = a) P(B = b | A = a, B = b, E = e) P(E = e) \\
&= \sum_{a \in T \setminus F} \sum_{b \in T \setminus F} P(J = T | A = a) \tau_1(A = a, B = b) \\
&= \sum_{a \in T \setminus F} P(J = T | A = a) \tau_1(A = a) \\
\end{align*}
\]
Variable elimination

**Assume order:** \( M, E, B, A \) to calculate \( P(J = T) \)

\[
= \sum_{a \in A} \sum_{e \in E} \sum_{a \in A} \sum_{m \in M} P(J = T \mid A = a) P(M = m \mid A = a) P(A = a \mid B = b, E = e) P(B = b) P(E = e)
\]

\[
= \sum_{a \in A} \sum_{e \in E} \sum_{a \in A} \sum_{m \in M} P(J = T \mid A = a) P(A = a \mid B = b, E = e) P(B = b) P(E = e) \left( \sum_{m \in M} P(M = m \mid A = a) \right)
\]

\[
= \sum_{a \in A} \sum_{e \in E} \sum_{a \in A} \sum_{m \in M} P(J = T \mid A = a) P(A = a \mid B = b, E = e) P(B = b) P(E = e) \tau_1(A = a, B = b)
\]

\[
= \sum_{a \in A} \sum_{e \in E} P(J = T \mid A = a) \left[ \sum_{e \in E} P(B = b) \tau_1(A = a, B = b) \right]
\]

\[
= \sum_{a \in A} P(J = T \mid A = a) \tau_1(A = a) = P(J = T)
\]

---

**Variable elimination**

**Assume order:** \( M, E, B, A \) to calculate \( P(J = T) \)

\[
= \sum_{b \in B} \sum_{c \in C} \sum_{a \in A} \sum_{m \in M} P(J = T \mid A = a) P(M = m \mid A = a) P(A = a \mid B = b, E = e) P(B = b) P(E = e)
\]

\[
= \sum_{b \in B} \sum_{c \in C} \sum_{a \in A} \sum_{m \in M} f_1(A) f_2(M, A) f_3(A, B, E) f_4(B) f_4(E)
\]

Conditional probabilities defining the joint factors

Variable elimination inference can be cast in terms of operations defined over factors
Factors

- **Factor**: is a function that maps value assignments for a subset of random variables to $\mathbb{R}$ (reals)
- **The scope of the factor**: a set of variables defining the factor
- **Example**: Assume discrete random variables $x$ (with values $a_1,a_2,a_3$) and $y$ (with values $b_1,b_2$)
  - Factor:
    $$\phi(x, y)$$
  - Scope of the factor: $\{x, y\}$

<table>
<thead>
<tr>
<th>$x$</th>
<th>$y$</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$a_1$</td>
<td>$b_1$</td>
<td>0.5</td>
</tr>
<tr>
<td>$a_1$</td>
<td>$b_2$</td>
<td>0.2</td>
</tr>
<tr>
<td>$a_2$</td>
<td>$b_1$</td>
<td>0.1</td>
</tr>
<tr>
<td>$a_2$</td>
<td>$b_2$</td>
<td>0.3</td>
</tr>
<tr>
<td>$a_3$</td>
<td>$b_1$</td>
<td>0.2</td>
</tr>
<tr>
<td>$a_3$</td>
<td>$b_2$</td>
<td>0.4</td>
</tr>
</tbody>
</table>

Factor Product

**Variables**: $A,B,C$

$$\phi(A, B, C) = \phi(B, C) \circ \phi(A, B)$$

<table>
<thead>
<tr>
<th>$B$</th>
<th>$C$</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$b_1$</td>
<td>$c_1$</td>
<td>0.1</td>
</tr>
<tr>
<td>$b_1$</td>
<td>$c_2$</td>
<td>0.6</td>
</tr>
<tr>
<td>$b_2$</td>
<td>$c_1$</td>
<td>0.3</td>
</tr>
<tr>
<td>$b_2$</td>
<td>$c_2$</td>
<td>0.4</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$A$</th>
<th>$B$</th>
<th>$C$</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$a_1$</td>
<td>$b_1$</td>
<td>$c_1$</td>
<td>0.5*0.1</td>
</tr>
<tr>
<td>$a_1$</td>
<td>$b_1$</td>
<td>$c_2$</td>
<td>0.5*0.6</td>
</tr>
<tr>
<td>$a_1$</td>
<td>$b_2$</td>
<td>$c_1$</td>
<td>0.2*0.3</td>
</tr>
<tr>
<td>$a_1$</td>
<td>$b_2$</td>
<td>$c_2$</td>
<td>0.2*0.4</td>
</tr>
<tr>
<td>$a_2$</td>
<td>$b_1$</td>
<td>$c_1$</td>
<td>0.1*0.1</td>
</tr>
<tr>
<td>$a_2$</td>
<td>$b_1$</td>
<td>$c_2$</td>
<td>0.1*0.6</td>
</tr>
<tr>
<td>$a_2$</td>
<td>$b_2$</td>
<td>$c_1$</td>
<td>0.3*0.3</td>
</tr>
<tr>
<td>$a_2$</td>
<td>$b_2$</td>
<td>$c_2$</td>
<td>0.3*0.4</td>
</tr>
<tr>
<td>$a_3$</td>
<td>$b_1$</td>
<td>$c_1$</td>
<td>0.2*0.1</td>
</tr>
<tr>
<td>$a_3$</td>
<td>$b_1$</td>
<td>$c_2$</td>
<td>0.2*0.6</td>
</tr>
<tr>
<td>$a_3$</td>
<td>$b_2$</td>
<td>$c_1$</td>
<td>0.4*0.3</td>
</tr>
<tr>
<td>$a_3$</td>
<td>$b_2$</td>
<td>$c_2$</td>
<td>0.4*0.4</td>
</tr>
</tbody>
</table>
### Factor Marginalization

**Variables:** A, B, C

\[
\phi(A, C) = \sum_{B} \phi(A, B, C)
\]

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>a1</td>
<td>b1</td>
<td>c1</td>
<td>0.2</td>
</tr>
<tr>
<td>a1</td>
<td>b1</td>
<td>c2</td>
<td>0.35</td>
</tr>
<tr>
<td>a1</td>
<td>b2</td>
<td>c1</td>
<td>0.4</td>
</tr>
<tr>
<td>a1</td>
<td>b2</td>
<td>c2</td>
<td>0.15</td>
</tr>
<tr>
<td>a2</td>
<td>b1</td>
<td>c1</td>
<td>0.5</td>
</tr>
<tr>
<td>a2</td>
<td>b1</td>
<td>c2</td>
<td>0.1</td>
</tr>
<tr>
<td>a2</td>
<td>b2</td>
<td>c1</td>
<td>0.3</td>
</tr>
<tr>
<td>a2</td>
<td>b2</td>
<td>c2</td>
<td>0.2</td>
</tr>
<tr>
<td>a3</td>
<td>b1</td>
<td>c1</td>
<td>0.25</td>
</tr>
<tr>
<td>a3</td>
<td>b1</td>
<td>c2</td>
<td>0.45</td>
</tr>
<tr>
<td>a3</td>
<td>b2</td>
<td>c1</td>
<td>0.15</td>
</tr>
<tr>
<td>a3</td>
<td>b2</td>
<td>c2</td>
<td>0.25</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>A</th>
<th>C</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>a1</td>
<td>c1</td>
<td>0.2+0.4=0.6</td>
</tr>
<tr>
<td>a1</td>
<td>c2</td>
<td>0.35+0.15=0.5</td>
</tr>
<tr>
<td>a2</td>
<td>c1</td>
<td>0.1</td>
</tr>
<tr>
<td>a2</td>
<td>c2</td>
<td>0.3</td>
</tr>
<tr>
<td>a3</td>
<td>c1</td>
<td>0.4</td>
</tr>
<tr>
<td>a3</td>
<td>c2</td>
<td>0.7</td>
</tr>
</tbody>
</table>

### Factor division

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>A=1</td>
<td>B=1</td>
<td>0.5</td>
</tr>
<tr>
<td>A=1</td>
<td>B=2</td>
<td>0.4</td>
</tr>
<tr>
<td>A=2</td>
<td>B=1</td>
<td>0.8</td>
</tr>
<tr>
<td>A=2</td>
<td>B=2</td>
<td>0.2</td>
</tr>
<tr>
<td>A=3</td>
<td>B=1</td>
<td>0.6</td>
</tr>
<tr>
<td>A=3</td>
<td>B=2</td>
<td>0.5</td>
</tr>
</tbody>
</table>

Inverse of a factor product

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>A=1</td>
<td>B=1</td>
<td>0.5/0.4=1.25</td>
</tr>
<tr>
<td>A=1</td>
<td>B=2</td>
<td>0.4/0.4=1.0</td>
</tr>
<tr>
<td>A=2</td>
<td>B=1</td>
<td>0.8/0.4=2.0</td>
</tr>
<tr>
<td>A=2</td>
<td>B=2</td>
<td>0.2/0.4=2.0</td>
</tr>
<tr>
<td>A=3</td>
<td>B=1</td>
<td>0.6/0.5=1.2</td>
</tr>
<tr>
<td>A=3</td>
<td>B=2</td>
<td>0.5/0.5=1.0</td>
</tr>
</tbody>
</table>
Markov random fields

An undirected network (also called independence graph)
- Probabilistic models with symmetric dependences
- \( G = (S, E) \)
  - \( S \) set of random variables
  - Undirected edges \( E \) that define dependences between pairs of variables

Example:
variables A,B ..H

\[
\begin{align*}
A & \quad G \\
B & \quad C \\
D & \quad E \\
F & \\
H &
\end{align*}
\]

The full joint of the MRF is defined

\[
P(x) \propto \prod_{c \in cl(x)} \phi_c(x_c)
\]

\( \phi_c(x_c) \) - A potential function (defined over variables in cliques/factors)

Example:

\[
\begin{align*}
A & \quad G \\
B & \quad C \\
D & \quad E \\
F & \\
H &
\end{align*}
\]

Full joint:

\[
P(A, B, \ldots H) \sim \phi_1(A, B, C)\phi_2(B, D, E)\phi_3(A, G)\phi_4(C, F)\phi_5(G, H)\phi_6(F, H)
\]

\( \phi_c(x_c) \) - A potential function (defined over a clique of the graph)
Markov random fields: independence relations

• **Pairwise Markov property**
  – Two nodes in the network that are not directly connected can be made independent given all other nodes

• **Local Markov property**
  – A set of nodes (variables) can be made independent from the rest of nodes variables given its immediate neighbors

• **Global Markov property**
  – A vertex set A is independent of the vertex set B (A and B are disjoint) given set C if all chains in between elements in A and B intersect C

---

Markov random fields: independence relations

• **Pairwise Markov property**
  – Two nodes in the network that are not directly connected can be made independent given all other nodes

• **Example:**

```
A — G — H
|     |
B — C — F
|     |
D — E
```

C and H are independent given the rest of the nodes
Markov random fields: independence relations

- **Local Markov property**
  - A set of nodes (variables) can be made independent from the rest of nodes variables given its immediate neighbors

- **Example:**

![Graph](image)

C is independent of \{G,H,D,E\} given the neighbors of C that is, variables \{A,B,F\}

---

Markov random fields: independence relations

- **Global Markov property**
  - A vertex set A is independent of the vertex set B (A and B are disjoint) given set C if all chains in between elements in A and B intersect C

- **Example:**

![Graph](image)

A set \{B, C\} is independent of \{G,H\} given the set\{A,F\}
Markov random fields

- regular lattice
  (Ising model)

- Arbitrary graph
Markov random fields

- **Joint probability**
  \[ P(x) \approx \prod_{c \in \mathcal{F}(x)} \phi_c(x_c) \]
  \( \phi_c(x_c) \) - A potential function (defined over cliques/factors)

- **Typical condition on potential functions:**
  - If \( \phi_c(x_c) \) is strictly positive we can rewrite the definition in terms of a log-linear model:
    \[ P(x) = \frac{1}{Z} \exp \left( - \sum_{c \in \mathcal{F}(x)} E_c(x_c) \right) \]
    - Energy function
    - Gibbs (Boltzman) distribution
  \[ Z = \sum_{x \in \mathcal{X}} \exp \left( - \sum_{c \in \mathcal{F}(x)} E_c(x_c) \right) \]
    - A partition function

Types of Markov random fields

- **MRFs with discrete random variables**
  - Clique potentials can be defined by mapping all clique-variable instances to \( \mathbb{R} (= \text{factors}) \)
  - Example: Assume two binary variables A,B with values \{a1,a2,a3\} and \{b1,b2\} are in the same clique \( c \). Then:
    \[ \phi_c(A, B) \]
    | a1 | b1 | 0.5 |
    | a1 | b2 | 0.2 |
    | a2 | b1 | 0.1 |
    | a2 | b2 | 0.3 |
    | a3 | b1 | 0.2 |
    | a3 | b2 | 0.4 |
Types of Markov random fields

- **Gaussian Markov Random Field**
  \[ x \sim N(\mu, \Sigma) \]
  \[ p(x \mid \mu, \Sigma) = \frac{1}{(2\pi)^{d/2} |\Sigma|^{1/2}} \exp \left[ -\frac{1}{2} (x - \mu)^T \Sigma^{-1} (x - \mu) \right] \]

- **Precision matrix** \( \Sigma^{-1} \)
- **Variables in** \( x \) **are connected in the network only if they have a nonzero entry in the precision matrix**
  - All zero entries are not directly connected

---

MRF variable elimination inference

**Example:**

\[ P(B) = \sum_{A,C,D,...H} P(A,B,...H) \]

\[ = \frac{1}{Z} \sum_{A,C,D,...H} \phi_1(A,B,C) \phi_2(B,D,E) \phi_3(A,G) \phi_4(C,F) \phi_5(G,H) \phi_6(F,H) \]

Eliminate \( E \)

\[ = \frac{1}{Z} \sum_{A,C,D,F,G,H} \phi_1(A,B,C) \left[ \sum_{E} \phi_2(B,D,E) \right] \phi_3(A,G) \phi_4(C,F) \phi_5(G,H) \phi_6(F,H) \]

\[ \tau_1(B,D) \]

CS 3750 Advanced Machine Learning
MRF variable elimination inference

Example (cont):

\[ P(B) = \sum_{A,C,D,...,H} P(A, B, ..., H) \]

\[ = \frac{1}{Z} \sum_{A,C,D,F,G,H} \phi_1(A,B,C) \tau_1(B, D) \phi_3(A,G)\phi_3(C,F)\phi_5(G,H)\phi_6(F,H) \]

Eliminate D

\[ = \frac{1}{Z} \sum_{A,C,F,G,H} \phi_1(A,B,C) \left[ \sum_{D} \tau_1(B, D) \right] \phi_3(A,G)\phi_4(C,F)\phi_5(G,H)\phi_6(F,H) \]

\[ \tau_2(B) \]

MRF variable elimination inference

Example (cont):

\[ P(B) = \sum_{A,C,D,...,H} P(A, B, ..., H) \]

\[ = \frac{1}{Z} \sum_{A,C,F,G,H} \phi_1(A,B,C) \tau_2(B) \phi_3(A,G)\phi_4(C,F)\phi_5(G,H)\phi_6(F,H) \]

Eliminate H

\[ = \frac{1}{Z} \sum_{A,C,F,G} \phi_1(A,B,C) \tau_2(B) \phi_3(A,G)\phi_4(C,F) \left[ \sum_{H} \phi_5(G,H)\phi_6(F,H) \right] \]

\[ \tau_3(F,G,H) \]

\[ \tau_4(F,G) \]
MRF variable elimination inference

Example (cont):

\[ P(B) = \sum_{A,C,D,...H} P(A, B,...H) \]

\[ = \frac{1}{Z} \sum_{c,f,g} \phi_1(A, B, C) \tau_2(B) \phi_3(A, G) \phi_4(C, F) \tau_4(F, G) \]

Eliminate F

\[ = \frac{1}{Z} \sum_{A,C,G} \phi_1(A, B, C) \tau_2(B) \phi_3(A, G) \left[ \sum_F \phi_4(C, F) \tau_4(F, G) \right] \]

\[ \tau_5(C, F, G) \]

\[ \tau_6(G, C) \]

MRF variable elimination inference

Example (cont):

\[ P(B) = \sum_{A,C,D,...H} P(A, B,...H) \]

\[ = \frac{1}{Z} \phi_1(A, B, C) \tau_2(B) \phi_3(A, G) \tau_6(C, G) \]

Eliminate G

\[ = \frac{1}{Z} \sum_{A,C} \phi_1(A, B, C) \tau_2(B) \left[ \sum_F \phi_3(A, G) \tau_6(C, G) \right] \]

\[ \tau_7(A, C, G) \]

\[ \tau_8(A, C) \]
MRF variable elimination inference

Example (cont):
\[ P(B) = \sum_{A,C,D,...H} P(A,B,...H) \]
\[ = \frac{1}{Z} \sum_{A,C} \phi_i(A,B,C) \tau_2(B) \tau_8(A,C) \]

Eliminate C

\[ = \frac{1}{Z} \sum_A \tau_2(B) \left[ \sum_C \phi_i(A,B,C) \tau_8(A,C) \right] \]
\[ \cdot \tau_9(A,B,C) \]
\[ \cdot \tau_{10}(A,B) \]

MRF variable elimination inference

Example (cont):
\[ P(B) = \sum_{A,C,D,...} P(A,B,...H) \]
\[ = \frac{1}{Z} \tau_2(B) \tau_{10}(A,B) \]
\[ = \frac{1}{Z} \tau_2(B) \sum_A \tau_{10}(A,B) \]

Eliminate A

\[ = \frac{1}{Z} \tau_2(B) \sum_A \tau_{10}(A,B) \]
\[ \cdot \tau_{11}(B) \]
\[ = \frac{1}{Z} : \tau_2(B) \tau_{11}(B) \]
Variable elimination

**Depends on the order of variables to eliminate**

**Question:** can we optimize the structures ahead of times so that we can make inferences efficiently and without worrying about the specific variable order?

- **Structures that support efficient inferences:**
  - Chains, and trees

![Diagram of variable elimination](image)