Undirected Graphical Models 3
Parameterizations Revisited

Finer-Grained Parameterizations

A factor graph $F$ is an undirected graph containing two types of nodes:
- Variable nodes (denoted as ovals) and
- Factor nodes (denoted as squares).

The graph only contains edges between variable nodes and factor nodes.

Finer-Grained Parameterizations

A factor graph $F$ is parameterized by a set of factors, where each factor node $V_\phi$ is associated with only one factor $\phi$, whose scope is the set of variables that are neighbors of $V_\phi$ in the graph.

A distribution $P$ factorizes over $F$ if it can be represented as a set of factors of this form.
Finer-grained Parameterization

Rather than encoding factors as complete tables over the scope of the factor, we can use a log-linear model:

\[ \phi(D) = \exp(-\alpha(D)) \]

Where \( \alpha(D) = -\ln \phi(D) \) is an energy function (which you want to minimize)

\[ P(X_1, \ldots, X_n) \propto \exp \left[ - \sum_{i=1}^{m} \epsilon_i(D_i) \right] \]

Note: log representation makes sure the distribution is positive

Finer-grained Parameterizations

Features provide a compact way to specify certain types of interactions

Example: Suppose \( A_1 \) and \( A_2 \) can take on \( l \) possible values \( a'_1, \ldots, a'_l \). \( A_1 \) and \( A_2 \) prefer situations when they take on the same value, and have no preference otherwise. The energy function might take the following:

\[ \epsilon(A_1, A_2) = \begin{cases} -10 & A_1 = A_2 \\ 0 & \text{otherwise} \end{cases} \]
Finer-grained Parameterizations

(example continued)
Two options for representing the factor:
• As a table, it requires \( P \) values
• Log-linear function in terms of a feature \( f(A_1, A_2) \)
  that is an indicator function for the event \( A_1 = A_2 \).
  The energy function looks like:

\[
\varepsilon(A_1, A_2) = 3 * I(A_1 = A_2)
\]

We just replaced a table with a function

Finer-grained Parameterizations

3 representations of the parameterization of a Markov network:
1. Markov network: product over potentials on cliques
2. Factor graph: product of factors
3. Set of features: product over feature weights

Which is most appropriate? Depends on the nature of the problem…

Overparameterization

A distribution \( P \) is a log-linear model over a Markov network \( \mathcal{H} \) if it is associated with:
• A set of features \( F = \{f_1(D_1), \ldots, f_k(D_k)\} \), where each \( D_i \) is a complete subgraph in \( \mathcal{H} \)
• A set of weights \( w_1, \ldots, w_k \)
  Such that

\[
P(X_1, \ldots, X_n) = \frac{1}{Z} \exp \left[ - \sum_{i=1}^k w_i f_i(D_i) \right]
\]
Overparameterization

Markov network parameterization is generally overparameterized
- If the graph is a single clique over n binary variables $X_1, \ldots, X_n$, the clique potential has $2^n$ parameters
- Joint distribution only has $2^n - 1$ independent parameters

But the overparameterization can get even worse…
We’ll discuss this briefly in the next few slides

Overparameterization

A more subtle form of overparameterization:
- Pair of cliques \{A, B\} and \{B, C\}
- Clique potential for \{A, B\} contains information about:
  - The interaction between A and B
  - About A and B individually
- Clique potential for \{B, C\} contains information about:
  - The interaction between B and C
  - About B and C individually

Overparameterization

Information about B can be stored in many different ways (either in one of the cliques or split between the two cliques in arbitrary ways)

Recall:
$$P(X_1, \ldots, X_n) = \frac{1}{Z} \exp[-\varepsilon_1(D_1) - \ldots - \varepsilon_m(D_m)]$$

You can end up with the same sum by “moving” values between different energy functions

Overparameterization

Example:

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>$\varepsilon_1(A,B)$</th>
<th>B</th>
<th>C</th>
<th>$\varepsilon_2(B,C)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>3.4</td>
<td>0</td>
<td>0</td>
<td>-4.61</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>-1.61</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>-2.3</td>
<td>1</td>
<td>1</td>
<td>-4.61</td>
</tr>
</tbody>
</table>

Note: $\varepsilon(D) = -\ln \phi(D)$
### Overparameterization

Alternate parameterization (but still an equivalent distribution): subtract 1 from $\epsilon_1(A,B)$ and add 1 to $\epsilon_2(A,B)$ when $B=0$

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>$\epsilon_1(A,B)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>-4.4</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1.61</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>-1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>2.3</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>B</th>
<th>C</th>
<th>$\epsilon_2(B,C)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>-3.61</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1.5</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>4.61</td>
</tr>
</tbody>
</table>

Note: if the instance has $B=0$, the sum of the energy functions is identical to the original sum.

### Special Cases of Markov Networks

**Ising model**: model for the energy of a physical system involving a system of interacting atoms

- $X_i \in \{+1,-1\}$ (direction of atom’s spin)
- $\epsilon_{ij}(x_i,x_j) = -w_{ij} x_i x_j$
- The energy function defines the following distribution

$$P(\xi) = \frac{1}{Z} \exp \left( - \sum_{i \leq j} w_{ij} x_i x_j - \sum_i u_i x_i \right)$$

- $w_{ij} > 0$, model prefers the spins of two atoms to be aligned
- Individual node potentials (bias nodes to have a particular spin)

*We won’t be covering this in more detail in this course.*
Special Cases of Markov Networks

Metric MRFs: We have a graph of nodes $X_1, ..., X_n$ related by a set of edges $E$, and we wish to assign a label in the space $V = \{v_1, ..., v_K\}$ to each $X_i$.

- In isolation, each node prefers a particular label
- But we want smoothness – neighboring nodes have “similar” values
- The energy function:
  $$E(x_1, ..., x_n) = \sum_i \phi_i(x_i) + \sum_{(i,j) \in E} \phi_{ij}(x_i, x_j)$$
- Goal is to minimize the energy $\arg \min_{x_1, ..., x_n} E(x_1, ..., x_n)$

How do we enforce smoothness? We have some options on the energy function:

**Option 1:**
$$\phi_{i,j}(x_i, x_j) = \begin{cases} 0 & x_i = x_j \\ \lambda_{i,j} & x_i \neq x_j, \lambda_{i,j} \geq 0 \end{cases}$$

**Option 2:**
$$\phi_{i,j}(x_i, x_j) = \min(c \|x_i - x_j\|_2, \text{dist}_{\text{max}})$$

This is a distance metric
This is the 2-norm