ECE 175B: Probabilistic Reasoning and Graphical Models
Lecture 13: Inference on Discrete Variable Serial Chains I

Florian Meyer & Ken Kreutz-Delgado

Electrical and Computer Engineering Department
University of California San Diego

UC San Diego
JACOBS SCHOOL OF ENGINEERING

Primary Source Material for this Lecture

This lecture is a summary discussion of lecture notes.

Inference on Discrete-Variable Serial Chains

which can be found on the Canvas page for this course.

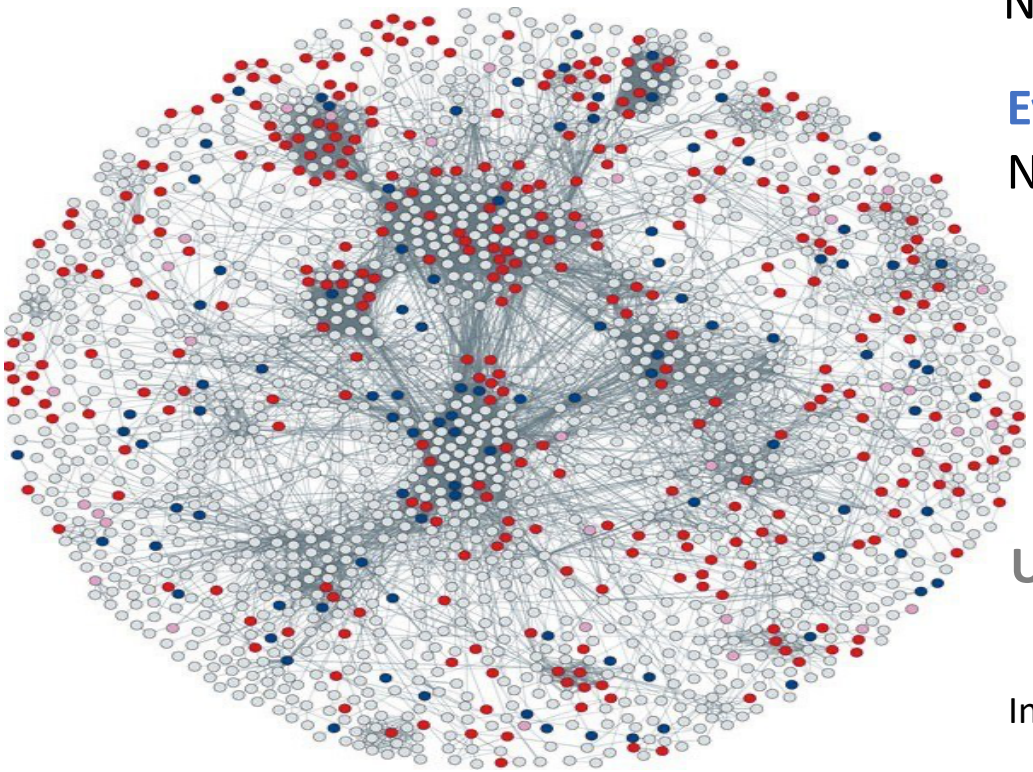
Explanations, Details, and steps in algorithm development that are not presented in these slides can be found in the notes.

All distributions are assumed positive throughout this note, $P > 0$.

Unnormalized probabilities are denoted with a tilde, i.e. as $\tilde{P}(\mathbf{x})$, *etc.*

Querying a Distribution Encoded in a Graph

$P_{\mathbf{X}}(x) = P_{\mathbf{W},\mathbf{E}}(w,e) = P_{\mathbf{V},\mathbf{Y}}(v,y) = P_{\mathbf{Y},\mathbf{E},\mathbf{Z}}(y,e,z)$



Query nodes = Y = "Outcome"

Non-Query nodes = **V** = **X** - **Y**

Evidence nodes = E

Non-Evidence nodes = **W** = **X** - **E**

$\mathbf{Y} \cap \mathbf{E} = \emptyset$

$\mathbf{Z} = \mathbf{X} - (\mathbf{Y} \cup \mathbf{E})$

Unobserved nodes = Z

In the standard manner we set $P_{\mathbf{X}}(x) = P(x)$, etc.

Two Basic Queries

- **Probability of Outcome**

  Compute the value of $P(\mathbf{y}|\mathbf{e})$ for outcome y given e

- **Most Probable Value of Outcome**

  Compute most probable outcome $\hat{\mathbf{y}}_{\text{map}} = \arg\max_{\mathbf{y}} P(\mathbf{y}|\mathbf{e})$ given e

How do we do this in an efficient, computationally tractable manner?

Probability of Outcome Queries

Here, the question is how to efficiently compute marginalizations like

$$P(\mathbf{y}) = \sum_{\mathbf{v}} P(\mathbf{v}, \mathbf{y}) = \sum_{\mathbf{v}} P(\mathbf{x}) \quad \text{and} \quad P(\mathbf{e}) = \sum_{\mathbf{w}} P(\mathbf{w}, \mathbf{e}) = \sum_{\mathbf{w}} P(\mathbf{x}),$$

and conditional probabilities like

$$P(\mathbf{w} \mid \mathbf{e}) = \frac{P(\mathbf{w}, \mathbf{e})}{P(\mathbf{e})} = \frac{P(\mathbf{x})}{P(\mathbf{e})}$$

and

$$P(\mathbf{y} \mid \mathbf{e}) = \frac{P(\mathbf{y}, \mathbf{e})}{P(\mathbf{e})} = \sum_{\mathbf{w} \in \mathbf{W} \setminus \mathbf{Y}} P(\mathbf{w} \mid \mathbf{e}) = \sum_{\mathbf{w} \setminus \mathbf{y}} P(\mathbf{w} \setminus \mathbf{y}, \mathbf{y} \mid \mathbf{e}).$$

Note that we can write

$$P(\mathbf{w} \mid \mathbf{e}) = \frac{1}{Z(\mathbf{e})} \psi(\mathbf{w}, \mathbf{e}) = \frac{1}{Z(\mathbf{e})} \psi(\mathbf{x}) \quad \text{with} \quad Z(\mathbf{e}) = \sum_{\mathbf{w}} \psi(\mathbf{w}, \mathbf{e})$$

E.g. take $\psi(\mathbf{w}, \mathbf{e}) = P(\mathbf{w}, \mathbf{e})$ and $Z(\mathbf{e}) = P(\mathbf{e})$. Note if no evidence, $\mathbf{E} = \emptyset$, then

$$P(\mathbf{x}) = \frac{1}{Z} \psi(\mathbf{x}) \quad \text{and} \quad Z = \sum_{\mathbf{x}} \psi(\mathbf{x}).$$

General Intractability of the Partition Function Z

Generally the partition function $Z(\mathbf{e}) = \sum_{\mathbf{w}} \psi(\mathbf{w}, \mathbf{e})$ is intractable to compute, making the determination of the probability value

$$P(\mathbf{y}|\mathbf{e}) = \sum_{\mathbf{w} \in \mathbf{W} \setminus \mathbf{Y}} P(\mathbf{w} \,|\, \mathbf{e}) = \frac{1}{Z(\mathbf{e})} \sum_{\mathbf{w} \in \mathbf{W} \setminus \mathbf{Y}} \psi(\mathbf{w}, \mathbf{e}) = \frac{1}{Z(\mathbf{e})} \psi(\mathbf{y}, \mathbf{e}) = \frac{\tilde{P}(\mathbf{y} \,|\, \mathbf{e})}{Z(\mathbf{e})}$$

practically noncomputable. For example suppose $\mathbf{w} = (w_1, \cdots, w_n)$ is a collection of n binary nodes w_i with $n = 100$. Then there are more than 10^{30} terms in the sum defining $Z(\mathbf{e})$.

Sometimes one can nonetheless tractably compute the unnormalized conditional probability $\tilde{P}(\mathbf{y}|\mathbf{e}) = \psi(\mathbf{y}, \mathbf{e})$. In this case one can tractably compute the conditional odds

$$\frac{P(\mathbf{y}' \,|\, \mathbf{e})}{P(\mathbf{y} \,|\, \mathbf{e})} = \frac{\tilde{P}(\mathbf{y}' \,|\, \mathbf{e})}{\tilde{P}(\mathbf{y} \,|\, \mathbf{e})} = \frac{\psi(\mathbf{y}', \mathbf{e})}{\psi(\mathbf{y}, \mathbf{e})}.$$

E.g. in this case one might be able to tractably determine that given the evidence $\mathbf{e}$ the value $\mathbf{Y} = \mathbf{y}'$ is ten times more probably to be seen in an experiment than the value $\mathbf{Y} = \mathbf{y}$.

Most Probable Outcome Queries

Given measured (instantiated) evidence $\mathbf{E} = \mathbf{e}$, compute the most probable *a posteriori* outcomes,

$$\widehat{\mathbf{w}}_{\text{map}} = \text{MAP}(\mathbf{W} \,|\, \mathbf{E} = \mathbf{e}) = \arg\max_{\mathbf{w}} P(\mathbf{w} \,|\, \mathbf{e})$$

$$\widehat{\mathbf{y}}_{\text{map}} = \text{MAP}(\mathbf{Y} \,|\, \mathbf{E} = \mathbf{e}) = \arg\max_{\mathbf{y}} P(\mathbf{y} \,|\, \mathbf{e}) = \arg\max_{\mathbf{y}} \sum_{\mathbf{w} \in \mathbf{W} \setminus \mathbf{Y}} P(\mathbf{w} \,|\, \mathbf{e}).$$

The second expression is also referred to as a *marginal* MAP query.

Important fact:

$$\widehat{\mathbf{w}}_{\text{map}} = \arg\max_{\mathbf{w}} P(\mathbf{w} \,|\, \mathbf{e}) = \arg\max_{\mathbf{w}} \psi(\mathbf{w}, \mathbf{e}) = \arg\max_{\mathbf{w}} \psi(\mathbf{x}),$$

Thus **there is no need to compute the partition function Z** in order to determine the MAP estimate $\widehat{\mathbf{w}}_{\text{map}}$.

For this reason, often **one can determine computationally efficient algorithms to determine the MAP estimate.**

Inference on Undirected Trees – I

A singly-connected undirected graph, or *undirected tree*, has only end-to-end connected pairwise cliques and no loops. Because the cliques on a singly connected undirected graph (i.e., on a *tree*) are pairwise, the general form of a compatible distribution must factorize according to

$$P(\mathbf{x}) = P(\mathbf{X} = \mathbf{x}) = \frac{1}{Z} \prod_{(i,j)} \phi_{i,j}(x_i, x_j),$$

where the product is over all edges (i, j) of the graph.

Given instantiated evidence $\mathbf{E} = \mathbf{e} = \{e_1, \cdots, e_s\}$ where $\mathbf{W} = \mathbf{X} \setminus \mathbf{E}$, $\mathbf{X} = (\mathbf{W}, \mathbf{E})$, we have,

$$P(\mathbf{w} \,|\, \mathbf{e}) = \frac{P(\mathbf{w}, \mathbf{e})}{P(\mathbf{e})} = \frac{P(\mathbf{x})}{P(\mathbf{e})}$$

with

$$P(\mathbf{x}) = P(\mathbf{w}, \mathbf{e}) = \frac{1}{Z} \prod_{(g,h)} \phi_{g,h}(e_g, e_h) \prod_{(i,j)} \phi_{i,j}(w_i, e_j) \prod_{(k,\ell)} \phi_{k,\ell}(w_k, w_\ell).$$

Inference on Undirected Trees – II

This gives

$$P(\mathbf{w} \mid \mathbf{e}) \propto \prod_{(i,j)} \phi_{i,j}(w_i, e_j) \prod_{(k,\ell)} \phi_{k,\ell}(w_k, w_\ell) = \prod_i \left(\prod_{j \in (i,j)} \phi_{i,j}(w_i, e_j) \right) \prod_{(k,\ell)} \phi_{k,\ell}(w_k, w_\ell) = \prod_i \phi_i(w_i; \mathbf{e}) \prod_{(k,\ell)} \phi_{k,\ell}(w_k, w_\ell)$$

where

$$\phi_i(w_i; \mathbf{e}) \triangleq \prod_{j \in (i,j)} \phi_{i,j}(w_i, e_j)$$

is the *local evidence* for node i taking the value w_i.

$$P(\mathbf{w} \mid \mathbf{e}) = \frac{1}{Z(\mathbf{e})} \psi(\mathbf{x}) = \frac{\tilde{P}(\mathbf{w} \mid \mathbf{e})}{Z(\mathbf{e})}$$

with

$$\tilde{P}(\mathbf{w} \mid \mathbf{e}) = \psi(\mathbf{x}) = \psi(\mathbf{w}, \mathbf{e}) = \prod_i \phi_i(w_i; \mathbf{e}) \prod_{(k,\ell)} \phi_{k,\ell}(w_k, w_\ell) = \prod_{(k,\ell)} \psi_{k,\ell}(w_k, w_\ell),$$

where (see footnote 8 in lecture notes)

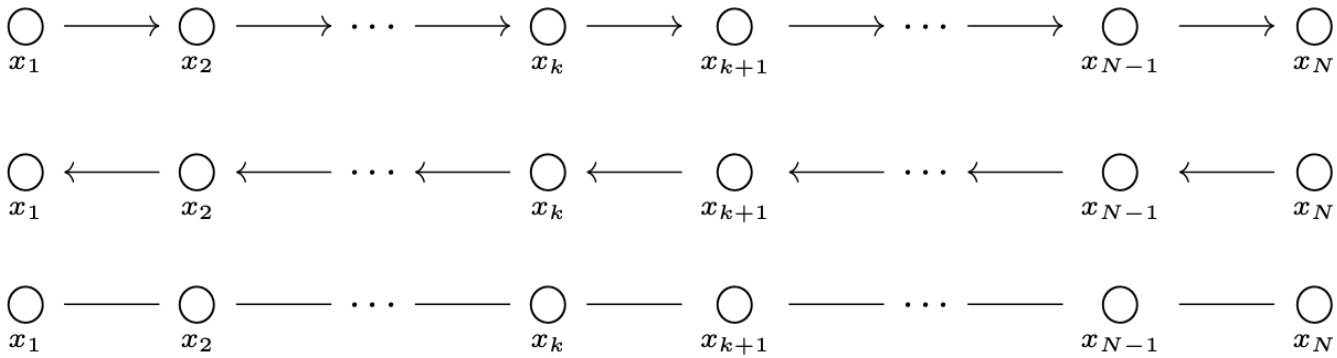
$$\psi_{k,\ell}(w_k, w_\ell) \triangleq \phi_{k,\ell}(w_k, w_\ell) \phi_\ell(w_\ell; \mathbf{e})$$

and

$$Z(\mathbf{e}) = \sum_{\mathbf{w}} \psi(\mathbf{x}) = \sum_{\mathbf{w}} \psi(\mathbf{w}, \mathbf{e}).$$

The Serial Markov Chain

The three serial-chain graphs shown below are *Markov equivalent* and encode exactly the same independence conditions:



The first graph is a Bayesian Network (BN) called a *forward Markov chain*, the second graph is a BN which is a *backward Markov chain*, and the third graph is a Markov Net (MN) which is an *undirected chain* that is "direction neutral".

All three graphs encode a factorization of the form

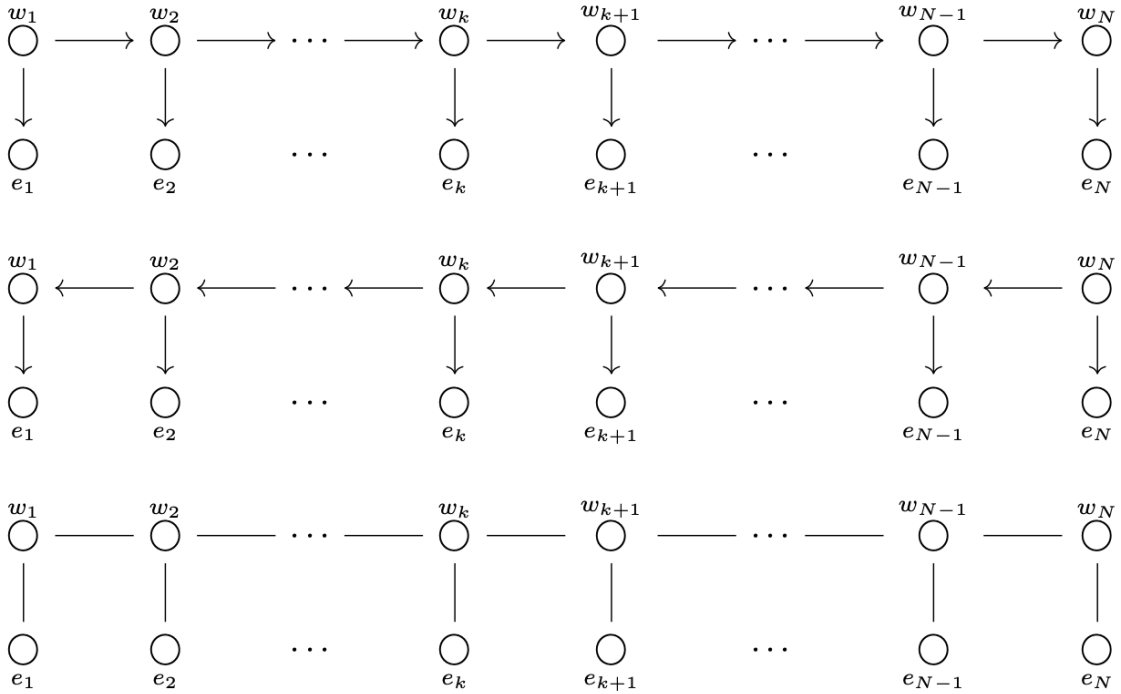
$$P(\mathbf{x}) = \frac{1}{Z}\,\phi(\mathbf{x}) = \frac{1}{Z}\prod_{k=0}^{N-1}\phi_{k,k+1}(x_k, x_{k+1})$$

For example, for the forward Markov chain take $\phi_{k,k+1}(x_k, x_{k+1}) = P(x_{k+1} \mid x_k)$ to obtain

$$P(\mathbf{x}) = P(X_N \mid X_{N-1}) \cdots P(x_{k+1} \mid x_k) \cdots P(x_2 \mid x_1)P(x_1) = \frac{1}{Z}\prod_{k=0}^{N-1}\phi_{k,k+1}(x_k, x_{k+1})$$

The Hidden Markov Model (HMM) – I

The very important *hidden Markov model* (HMM) is a simple tree that can be analyzed as a chain when instantiated on the evidence. The "standard" HMM is shown by the simple directed tree shown at the top below:



The standard HMM is often interpreted as representing Markovian dynamics that is *moving forward in time*. In an HMM the variables w_k are called the *hidden units* while e_k are the *visible units* (aka measurable or observable units). The tree in the middle represents dynamics moving backward in time, while the bottom tree is agnostic to time ordering and says that each "interior" node depends on its three neighbors. **All three are Markov Equivalent**.

The Hidden Markov Model (HMM) – II

All three of these graphical representations can be used to model the same evidence-conditional Markov process via factorization.

$$P(\mathbf{x}) = P(\mathbf{w}, \mathbf{e}) = \frac{1}{Z}\psi(\mathbf{w}, \mathbf{e}) = \frac{1}{Z}\prod_{k=0}^{N-1} \underbrace{\phi_{k,k+1}(w_k, w_{k+1})\phi_{k+1}(w_{k+1}, e_{k+1})}_{\psi_{k,k+1}(w_k, w_{k+1}, e_{k+1})},$$

where we define

$$\psi_{k,k+1}(w_k, w_{k+1}, e_{k+1}) = \phi_{k,k+1}(w_k, w_{k+1})\phi_{k+1}(w_{k+1}, e_{k+1}).$$

As an example, given the factorization for the forward Markov model take:

$$Z = 1$$

$$\phi_{0,1}(w_0, w_1) = \phi_1(w_1) = P(w_1) \qquad (\, w_0 = \emptyset \,)$$
$$\phi_{k,k+1}(w_k, w_{k+1}) = P(w_{k+1}|w_k), \quad k \geq 1$$
$$\phi_k(w_k, e_k) = P(e_k|w_k), \quad k \geq 1$$

and

$$\psi_{k,k+1}(w_k, w_{k+1}, e_{k+1}) = \phi_{k,k+1}(w_k, w_{k+1})\phi_{k+1}(w_{k+1}, e_{k+1}), \quad 0 \leq k \leq N-1.$$

Note that

$$\psi_{k,k+1}(w_k, w_{k+1}, e_{k+1}) = \begin{cases} P(e_1|w_1)P(w_1) = P(e_1, w_1) & k = 0 \\ P(e_{k+1}|w_{k+1})P(w_{k+1}|w_k) = P(e_{k+1}, w_{k+1}|w_k) & 1 \leq k \leq N-1 \end{cases}$$

Serial Chain Factorizations

If we notationally suppress the dependence on the evidence $\mathbf{e}$, setting

$$\psi_{k,k+1}(w_k, w_{k+1}) = \psi_{k,k+1}(w_k, w_{k+1}, e_{k+1}),$$

we have

$$\boxed{P(\mathbf{x}) = P(\mathbf{w}, \mathbf{e}) = \frac{1}{Z} \prod_{k=0}^{N-1} \psi_{k,k+1}(w_k, w_{k+1})}$$

which implies

$$\boxed{P(\mathbf{w} \,|\, \mathbf{e}) = \frac{1}{Z(\mathbf{e})} \prod_{k=0}^{N-1} \psi_{k,k+1}(w_k, w_{k+1})}$$

where $Z(\mathbf{e}) = ZP(\mathbf{e})$.

These forms look like the factorization for a regular chain discussed earlier in this lecture. Henceforth we will call factorizations of this form **Serial Chain Factorizations**.

In the following lecture, we'll continue our discussion of the material in lecture notes. We will exploit the structural form of the serial chain factorization to develop the message-passing and belief-propagation algorithms for efficiently computing outcome probability values.