ECE 175B: Probabilistic Reasoning and Graphical Models
Lecture 14: Inference on Discrete Variable Serial Chains II

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.

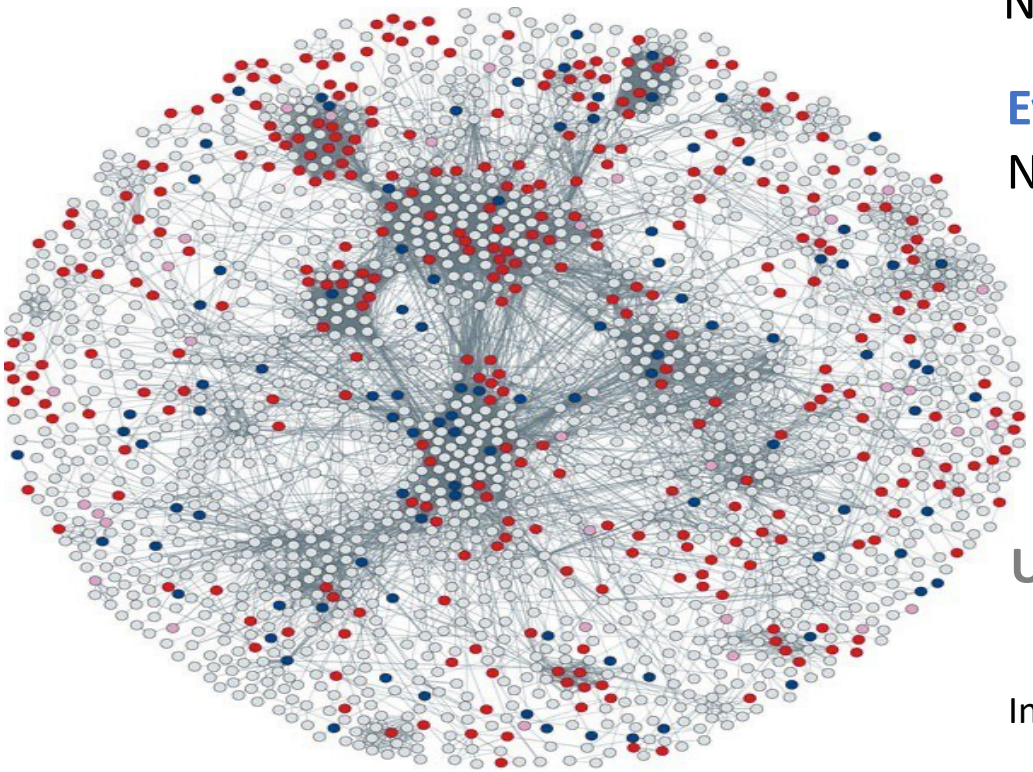
Explanations, Details, and steps in algorithm development that are not presented in these slides can be found in the notes.

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 $\mathbf{Y = y}$ Given Evidence $\mathbf{E = e}$**

  Compute the value of $\mathbf{P(y \mid e)}$

- **Most Probable Value of Outcome $\mathbf{Y = y}$ Given Evidence $\mathbf{E = e}$**

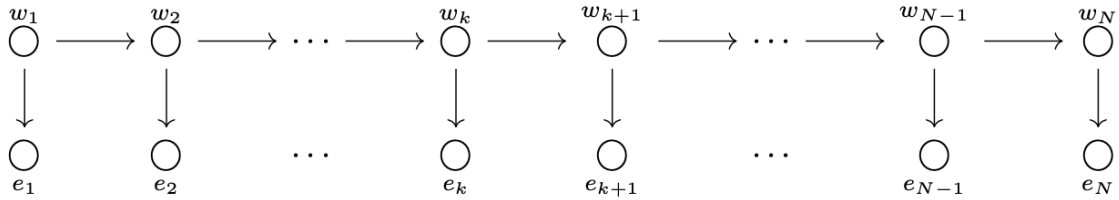
  Compute the most probable outcome

  $$\hat{\mathbf{y}}_{\text{map}} = \arg\max_{\mathbf{y}} \mathbf{P(y \mid e)}$$

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

The Hidden Markov Model (HMM)

A stochastic model of great importance in the *hidden Markov model* (HMM). The HMM is a simple tree that can be analyzed as a serial chain when instantiated on the evidence.



The standard HMM shown above is often interpreted as representing *Markovian dynamics* that models the behavior of a process which is *progressing forward in time*. The *hidden* variables w_k are often referred to as the unobservable *state variables* of the process, while the *visible* variables e_k are the observable *outputs*.

Let $Z_k = (W_k, E_k)$ and for $i \leq j$ let $\mathcal{Z}_i^j = \{Z_i, \cdots, Z_j\}$. The dynamics is said to be Markovian because "the past is independent of the future given the present value" of the *state* $W_k = w_k$:

$$\mathcal{Z}_1^{k-1} \perp\!\!\!\perp \mathcal{Z}_{k+1}^N \,|\, W_k \iff P(\mathbf{z}_1^{k-1}, \mathbf{z}_{k+1}^N \,|\, w_k) = P(\mathbf{z}_1^{k-1} \,|\, w_k)\, P(\mathbf{z}_{k+1}^N \,|\, w_k)$$

In particular this implies the **Markovian State-Transition Property**:

$$\boxed{P(w_{k+1} \,|\, w_k, \mathbf{w}_1^{k-1}) = P(w_{k+1} \,|\, w_k)}$$

I.e., knowledge of the state-value today, $W_k = w_k$, enables one to predict the probability of tomorrow's state-value, $W_{k+1} = w_{k+1}$, independently of how the system evolved to today's value.

Serial-Chain Factorization (SCF)

To sync up with the notation used in our previous lecture and the lecture notes, we set

$$\mathbf{x} = \mathbf{z} = \mathbf{z}_1^N = \{z_1, \cdots, z_N\}, \quad \mathbf{w} = \mathbf{w}_1^N = \{w_1, \cdots, w_N\}, \quad \text{and} \quad \mathbf{e} = \mathbf{e}_1^N = \{e_1, \cdots, e_N\}.$$

The distribution compatible with the HMM is

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

where $w_0 = \emptyset$ and $P(w_1|w_0) = P(w_1)$. Also define the normalization factor $Z = 1$ and

$$\underbrace{\psi_{k,k+1}(w_k, w_{k+1})}_{\text{supressed-evidence notation}} \equiv \underbrace{\psi_{k,k+1}(w_k, w_{k+1}, e_{k+1})}_{\text{expressed-evidence notation}} \triangleq P(e_{k+1},\, w_{k+1}|w_k).$$

This yields the *Serial-Chain Factorizations*

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

with $Z(\mathbf{e}) = ZP(\mathbf{e})$. Note both $P(\mathbf{x}) = P(\mathbf{w}, \mathbf{e})$ and $P(\mathbf{w}|\mathbf{e})$ have *the same unnormalized form*.

Conditional Random Fields (CRFs) and Dynamical Bayesian Networks (DBNs)

- Conditioned on the evidence, the HMM has the structure of a Markov Random Field (MRF). Thus the HMM is a particular instance of a **Conditional Random Field** (CRF). CRFs are the general class of probabilistic graphical models that have the structure of an MRF when conditioned on the evidence (which is now more generally referred to as the *observation*). CRF graphical models are N-node undirected graphs whose nodes are given by $\mathcal{Z}_1^N$. They can be Markov equivalent to undirected, directed, or mixed graphs.

- The HMM is also a special case of a **Dynamical Bayesian Network(DBN)** . DBNs are important in economics, industrial control, financial prediction, investment portfolio optimization, speech processing, communications theory, and many other domains that require the prediction and control of processes that unfold in time.

- Like the HMM, many other CRF and DBN models have evidence-conditional graphical models that result in conditional distributions with the Serial-Chain Factorization (SCF) form. Thus the efficient algorithms derived below have a wide range of applications.

Message Passing Algorithm (MPA) for a Serial Chain – I

To reduce notational clutter, we drop the subscripts on potential functions. Also assume each variable w_k can take any one of the K possible distinct values in the set $\mathbb{X} = \{\xi_1, \cdots, \xi_K\}$.

Select a single state, w_k, as being of interest. Our goal is to compute the value of the posterior distribution $P(w_k|\mathbf{e})$ as efficiently as possible. Note that $P(w_k|\mathbf{e})$ is given by the marginalization,

$$P(w_k|\mathbf{e}) = \sum_{\mathbf{w}\setminus w_k} P(\mathbf{w}|\mathbf{e}) = \sum_{\mathbf{w}\setminus w_k} \frac{P(\mathbf{w},\mathbf{e})}{P(\mathbf{e})} = \frac{\sum\limits_{\mathbf{w}\setminus w_k} P(\mathbf{w},\mathbf{e})}{\sum\limits_{\mathbf{w}} P(\mathbf{w},\mathbf{e})}\,.$$

Directly performing the shown sums is generally computationally intractable. For example, if each variable w_ℓ takes on K discrete values then each sum requires $\mathrm{O}(K^N)$ operations. Thus if $N = 100$ and $K = 2$, then $\mathrm{O}(2^{100}) \sim \mathrm{O}(10^{30})$ operations are required.

Fortunately, the distribution has the SCF will allow us to compute $P(w_k|\mathbf{e})$ in a computationally tractable manner. To start, note that (dropping the subscripts on the ψ's)

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

$$P(w_k|\mathbf{e}) = \sum_{\mathbf{w}\setminus w_k} P(\mathbf{w}|\mathbf{e}) = \frac{1}{ZP(\mathbf{e})} \sum_{\mathbf{w}\setminus w_k} \prod_{k=0}^{N-1} \psi(w_k, w_{k+1}).$$

MPA for a Serial Chain – II

$$
\begin{aligned}
& ZP(\mathbf{e})P(w_k|\mathbf{e}) \\
&= \sum_{\mathbf{w}\setminus w_k} \prod_{k=0}^{N-1} \psi(w_k, w_{k+1}) \\
&= \sum_{w_1} \cdots \sum_{w_{k-1}} \sum_{w_{k+1}} \cdots \sum_{w_N} \psi(w_0, w_1) \cdots \psi(w_{k-1}, w_k)\psi(w_k, w_{k+1}) \cdots \psi(w_{N-1}, w_N) \\
&= \underbrace{\left(\sum_{w_1} \cdots \sum_{w_{k-1}} \psi(w_0, w_1) \cdots \psi(w_{k-1}, w_k) \right)}_{\triangleq\, m^-(w_k) = \text{message to } w_k \text{ from the past}} \underbrace{\left(\sum_{w_{k+1}} \cdots \sum_{w_N} \psi(w_k, w_{k+1}) \cdots \psi(w_{N-1}, w_N) \right)}_{\triangleq\, m^+(w_k) = \text{message to } w_k \text{ from the future}}
\end{aligned}
$$

Also

$$
\sum_{w_k} P(w_k|\mathbf{e}) = 1 \implies ZP(\mathbf{e}) = \sum_{w_k} m^-(w_k)m^+(w_k)
$$

which implies that

$$
\boxed{P(w_k|\mathbf{e}) = \frac{m^-(w_k)\, m^+(w_k)}{\sum_{w_k} m^-(w_k)\, m^+(w_k)}}
$$

Once $m^-(w_k)$ and $m^+(w_k)$ have been computed the required normalization sum is quite tractable. For example, if $K = 2$, there are only two terms in the sum. Note that we have completely side-stepped the need to perform the generally computationally intractable operations required to compute Z and $P(\mathbf{e})$.

MPA for a Serial Chain – III

Reversing the order of the sums and factors comprising $m^-(w_k)$, we have

$$m^-(w_k) = \sum_{w_{k-1}} \sum_{w_{k-2}} \cdots \sum_{w_2} \sum_{w_1} \psi(w_{k-1}, w_k)\psi(w_{k-2}, w_{k-1}) \cdots \psi(w_2, w_3)\psi(w_1, w_2) \underbrace{\psi(w_0, w_1)}_{m^-(w_1)}$$

$$= \underbrace{\sum_{w_{k-1}} \psi(w_{k-1}, w_k) \underbrace{\sum_{w_{k-2}} \psi(w_{k-2}, w_{k-1}) \cdots \underbrace{\sum_{w_2} \psi(w_2, w_3) \underbrace{\sum_{w_1} \psi(w_1, w_2)\, m^-(w_1)}_{m^-(w_2)}}_{m^-(w_3)}}_{m^-(w_{k-1})}}_{m^-(w_k)}$$

It is evident that a **forward message passing recursion** is given by

$$\boxed{m^-(w_k) = \sum_{w_{k-1}} \psi(w_{k-1}, w_k)\, m^-(w_{k-1}), \quad k = 1, \cdots, N}$$

with $\psi(w_{k-1}, w_k) = P(e_k|w_k)P(w_k|w_{k-1})$ and initial condition

$$\boxed{m^-(w_0) = 1 \iff m^-(w_1) = \psi(w_0, w_1) = P(e_1|w_1)P(w_1)}$$

MPA for a Serial Chain – IV

In a similiar manner, we obtain

$$
\begin{aligned}
& m^+(w_k) \\
&= \sum_{w_{k+1}} \sum_{w_{k+2}} \cdots \sum_{w_{N-1}} \sum_{w_N} \psi(w_k, w_{k+1}) \psi(w_{k+1}, w_{k+2}) \cdots \psi(w_{N-2}, w_{N-1}) \psi(w_{N-1}, w_N) \cdot \underbrace{1}_{m^+(w_N)} \\
&= \sum_{w_{k+1}} \psi(w_k, w_{k+1}) \underbrace{\sum_{w_{k+2}} \psi(w_{k+1}, w_{k+2}) \cdots \underbrace{\sum_{w_{N-1}} \psi(w_{N-2}, w_{N-1}) \underbrace{\sum_{w_N} \psi(w_{N-1}, w_N)\, m^+(w_N)}_{m^+(w_{N-1})}}_{m^+(w_{N-2})}}_{m^+(w_{k+1})} \\
&\qquad \underbrace{\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad}_{m^+(w_k)}
\end{aligned}
$$

We see that a **backward message passing recursion** is given by,

$$\boxed{m^+(w_k) = \sum_{w_{k+1}} \psi(w_k, w_{k+1})\, m^+(w_{k+1}), \quad k = N-1, \cdots, 0}$$

with $\psi(w_k, w_{k+1}) = P(e_{k+1}|w_{k+1}) P(w_{k+1}|w_k)$ and boundary condition

$$\boxed{m^+(w_N) = 1}$$

MPA for a Serial Chain – V

The forward/backward iterations needed to compute the messages $m^-(w_k)$ and $m^+(w_k)$ are quite efficient. If the variables w_k take K discrete values, then the K values $m^-(w_k = \xi_i)$, $i = 1, \cdots, K$, can be stacked into a K-dimensional vector $\mathbf{m}_k^-$ and the K^2 values of $\psi(w_{k-1}, w_k)$ can be assembled into a $K \times K$ matrix $\mathbf{\Psi}_{k-1,k}$, where

$$\left(\mathbf{\Psi}_{k-1,k}\right)_{ij} = \psi_{k-1,k}(\xi_i, \xi_j, e_k).$$

The forward message passing algorithm can then be written as,

$$\underbrace{\mathbf{m}_k^-}_{K\times 1} = \underbrace{\mathbf{\Psi}_{k-1,k}^T}_{K\times K} \underbrace{\mathbf{m}_{k-1}^-}_{K\times 1}.$$

Similarly, defining the vector $\mathbf{m}_k^+$ with elements $m^+(w_k = \xi_i)$, $i = 1, \cdots, K$, allows the backwar message passing algorithm to be written as

$$\mathbf{m}_k^+ = \mathbf{\Psi}_{k,k+1}\mathbf{m}_{k+1}^+.$$

The shown matrix-vector multiplications each require $\mathrm{O}(K^2)$ operations. To compute $P(w_k|\mathbf{e})$ for *every* node $k = 1, \cdots, N$, the matrix-vector operations are performed N times twice; once as the algorithms sweep across the N-node chain forward to compute $m^-(w_k)$ and backward to compute $m^+(w_k)$ for $k = 1, \cdots, N$. These two sweeps each require $\mathrm{O}(NK^2)$ operations. The number of required additions is $\mathrm{O}(NK)$. Thus a total of $\mathrm{O}(NK^2)$ operations are required to compute $m^-(w_k)$ and $m^+(w_k)$. For example, if $N = 100$ and $K = 2$ on the order of a thousand operations are needed. Since an additional $\mathrm{O}(NK)$ operations are required to compute $P(w_k|\mathbf{e})$ for $k = 1, \cdots, N$, the total operations count to compute $P(w_k|\mathbf{e})$ for all k is $\mathrm{O}(NK^2)$, which is a stunning reduction in computational complexity from the $\mathrm{O}(N^K)$ cost of a direct, naive computation.

MPA for a Serial Chain – VI

Given (generally evidence-dependent) potentials along a serial chain

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

the **Message Passing Algorithm (MPA)** is quite simple:

$$m^-(w_k) = \sum_{w_{k-1}} \psi(w_{k-1}, w_k)\, m^-(w_{k-1}), \quad k = 1, \cdots, N$$

$$m^+(w_k) = \sum_{w_{k+1}} \psi(w_k, w_{k+1})\, m^+(w_{k+1}), \quad k = N-1, \cdots, 0$$

$$P(w_k|\mathbf{e}) = \frac{m^-(w_k)\, m^+(w_k)}{\sum_{w_k} m^-(w_k)\, m^+(w_k)}$$

For the HMM we have the evidence based potentials

$$\psi(w_{k-1}, w_k) = P(e_k|w_k)P(w_k|w_{k-1}) = P(e_k,\ w_k|w_{k-1})$$

Because the MPA has both a forward pass and a backward pass along the chain, it is commonly referred to by communications engineers as the **forward-backward algorithm.** It is sometimes called the **BCJR algorithm** after the initials of the four authors who proposed this algorithm (for symbol decoding in communications systems) **in 1974**.

Belief Propagation Algorithm (BPA) for a Serial Chain – I

Belief propagation is a special case of message passing. When the messages are equal to, or proportional to, probabilities or to invertible functions of probabilities such as log-probabilities, we refer to them as *beliefs*. In this case we write,

$$\text{bel}^-(w_k) = m^-(w_k) \quad \text{and} \quad \text{bel}^+(w_k) = m^+(w_k)\,.$$

With these name changes, the **Belief Propagation Algorithm** (BPA) for serial chains is essentially given by the MPA, but **only after** appropriate forms for the (generally evidence-dependent) potentials $\psi(w_k, w_{k+1})$ have been established to ensure that the beliefs are indeed proportional to invertible functions of probabilities. This is easily done for the HMM where $\psi(w_k, w_{k+1})$ is given by

$$\psi(w_k, w_{k+1}) = P(e_k|w_k)P(w_k|w_{k-1}) = P(e_k,\, w_k|w_{k-1})$$

HMM BPA:

$$bel^-(w_k) = \sum_{w_{k-1}} P(e_k|w_k)P(w_k|w_{k-1})\, bel^-(w_{k-1})\,, \quad k = 1, \cdots, N$$

$$bel^+(w_k) = \sum_{w_{k+1}} P(e_{k+1}|w_{k+1})P(w_{k+1}|w_k)\, bel^+(w_{k+1})\,, \quad k = N-1, \cdots, 0$$

$$P(w_k|\mathbf{e}) = \frac{bel^-(w_k)\, bel^+(w_k)}{\sum_{w_k} bel^-(w_k)\, bel^+(w_k)}$$

To verify this, we show that the HMM belief functions $\text{bel}^-(w_k)$ and $\text{bel}^+(w_k)$ are themselves probabilities.

BPA for a Serial Chain – II

As detailed in the lecture notes, two induction proofs demonstrate that $\text{bel}^-(w_k)$ and $\text{bel}^+(w_k)$ are probabilities, where the basis steps of the induction proofs are given by the boundary conditions

$$\text{bel}^-(w_1) = P(e_1, w_1) = P(e_1|w_1)P(w_1) \quad \text{and} \quad \text{bel}^+(w_N) = 1,$$

which are obviously probabilities. The induction steps of the proofs show that

$$\begin{aligned}\text{bel}^-(w_k) &= P(e_1, \cdots, e_k, w_k) = P(e_1, \cdots, e_k|w_k)P(w_k)\\ \text{bel}^+(w_k) &= P(e_{k+1}, \cdots, e_N|w_k)\end{aligned}$$

Since given the "present value" of w_k the "past and present data is independent of the future data", we have

$$\text{bel}^-(w_k)\text{bel}^+(w_k) = P(e_1, \cdots, e_k|w_k)P(e_{k+1}, \cdots, e_N|w_k)P(w_k) = P(\mathbf{e}|w_k)P(w_k) = P(\mathbf{e}, w_k).$$

Therefore

$$P(w_k|\mathbf{e}) = \frac{\text{bel}^-(w_k)\,\text{bel}^+(w_k)}{\sum_{w_k}\text{bel}^-(w_k)\,\text{bel}^+(w_k)} = \frac{P(\mathbf{e}, w_k)}{\sum_{w_k}P(\mathbf{e}, w_k)} = \frac{P(w_k, \mathbf{e})}{P(\mathbf{e})}$$

as required.