Reminder: class web links and infrastructure

- Check in with our web page (http://j.ee.washington.edu/~bilmes/classes/ee596a_winter_2013/) for up to date announcements, homeworks, etc.
Reminder: class web links and infrastructure

- Check in with our web page (http://j.ee.washington.edu/~bilmes/classes/ee596a_winter_2013/) for up to date announcements, homeworks, etc.

- All homeworks will be due via our dropbox (https://catalyst.uw.edu/collectit/dropbox/bilmes/25379)
Check in with our web page (http://j.ee.washington.edu/~bilmes/classes/ee596a_winter_2013/) for up to date announcements, homeworks, etc.

All homeworks will be due via our dropbox (https://catalyst.uw.edu/collectit/dropbox/bilmes/25379)

All questions should be posted to our discussion board (https://catalyst.uw.edu/gopost/board/bilmes/31332)
Reminder: class web links and infrastructure

- Check in with our web page (http://j.ee.washington.edu/~bilmes/classes/ee596a_winter_2013/) for up to date announcements, homeworks, etc.
- All homeworks will be due via our dropbox (https://catalyst.uw.edu/collectit/dropbox/bilmes/25379)
- All questions should be posted to our discussion board (https://catalyst.uw.edu/gopost/board/bilmes/31332)
- You can contact me anonymously if you wish via anonymous email (https://catalyst.uw.edu/umail/form/bilmes/4144)
Announcements, Assignments, and Reminders

- No class next Wednesday (one week from today), 1/16 (will either have makeup lecture, or video lecture on the web page)
- No class on Monday, 1/21 official holiday.
Readings are in a sub-directory “reading_drafts” directly below our web page (http://j.ee.washington.edu/~bilmes/classes/ee596a_winter_2013/).
Relevant Readings: Readings/handouts

- Readings are in a sub-directory “reading_drafts” directly below our web page (http://j.ee.washington.edu/~bilmes/classes/ee596a_winter_2013/).
- uid is this class name (lower case) and pwd are the quarter/year of the class.
Reading rough drafts are in "doc.pdf", read chapters 1 - 6.
In particular, read chapter 1 and 2 (overview, and about families)
Read chapters 3 (undirected) and chapters 4 (directed) graphical models.
read chapter 5 (on evidence)
read chapter 6 (inference on trees)
Then, read Wainwright/Jordan book chaps 3/4/5
http://dx.doi.org/10.1561/2200000001
Cumulative Outstanding Reading

- Read 8.1 - 8.3 in "doc.pdf"
Review section

- All slides in this section are either from previous lectures, or are a summary of material from previous lectures.
Review section

- All slides in this section are either from previous lectures, or are a summary of material from previous lectures.
- For more details, see the previous lectures.
A goal of graphical model inference - produce generic algorithm

Any $p \in \mathcal{F}(G,\mathcal{M})$

A particular probabilistic query

Observed Data

Correct answer

Produce Graphical Model Inference Procedure

$\text{inference}(p)$ an algorithm for doing Inference
Dynamic Graphical Models

What is different from static graphical model case?

- Any “graph” $G$ and properties $M$ are defined via an expandable but static “template” that, along with rules, allow certain graphs to be instantiated.

- An expanded template corresponds to a graph in a GM in the normal sense, and rules also say how to do expansion for given length. Expansion is in one dimension only.

- Distributions in the same family correspond to variable (unbounded) number of random variables — RVs can expand unboundedly in time, but still a member of the family.

- Graph is shaped differently. Much wider than it is taller, often there is a “time” parameter $T$ that expands the template. For online inference, rules say how to expand a next chunk.
What is different from static graphical model case?

- Typically some form of parameter sharing (otherwise, unbounded time would require unbounded number of parameters).
- There is some form of “temporal Markov property” in one way or another that, loosely speaking, takes the form: The future and past are rendered conditionally independent given the present.
- For probabilistic inference: Still want to deduce computational properties from the static template, rather than have to expand the template to each possible length, and then (re-)deduce an inference algorithm — amortization is again the goal.
Many forms of DGM:

- In each case, a graph template can describe many important properties of members of these families. We will become fairly well versed in almost all of the above.

- In each case, some form of “the past is independent of the future given the present”, for various definitions of past, present, and future.
Template models - summary

- static graphical models
- plate models
- dynamic graphical models
- general template models
Plate models

- A convenience to represent that a portion of a graph can be expanded any number of times, say $M$.

- The plate = graph stub surrounded by a rectangle, and an integer that indicates the number of times that the stub should be repeated.

- With no edges crossing the plate, graph is simply expanded $M$ times corresponding to $M$ set of random variable groups — groups are mutually independent of each other, dependencies exist within each group identical to the plate template.
Plate models

- With edges crossing plate, coupling may occur between stub instantiations.
- This may also occur hierarchically to express multiple groups with a common parameter in a Bayesian setting:
Plate models - self loops

- The stub itself may possess self-loops, something normally disallowed in a Bayesian network.
- This corresponds to edges between the same variable in successive instantiations of the stub. Compare with persistent edges in a dynamic graphical model.
- HMM as a plate model, with $T = 5$:
Definition 2.2.3 (Markov chain)

A collection of discrete-valued random variables \( \{Q_t : t \geq 1\} \) forms an \( n^{th} \)-order Markov chain if

\[
P(Q_t = q_t | Q_{t-1} = q_{t-1}, Q_{t-2} = q_{t-2}, \ldots, Q_1 = q_1) = \tag{2.11}
\]

\[
P(Q_t = q_t | Q_{t-1} = q_{t-1}, Q_{t-2} = q_{t-2}, \ldots, Q_{t-n} = q_{t-n}) \tag{2.12}
\]

for all \( t \geq 1 \), and all \( q_1, q_2, \ldots, q_t \).
We view the event \( \{Q_t = i\} \) as if the chain is “in state \( i \) at time \( t \)” and the event \( \{Q_t = i, Q_{t+1} = j\} \) as a transition from state \( i \) to state \( j \) starting at time \( t \). This notion arises by viewing a Markov chain as a stochastic finite-state automata (FSA).
Markov chain order conversion

- An $n^{th}$-order Markov chain may be converted into equivalent first-order Markov chain via state equivalence:

$$Q'_t \overset{\Delta}{=} \{Q_t, Q_{t-1}, \ldots, Q_{t-n}\}$$

where $Q_t$ is an $n^{th}$-order Markov chain.

- Then $Q'_t$ is a first-order Markov chain because

$$P(Q'_t = q'_t | Q'_{t-1} = q'_{t-1}, Q'_{t-2} = q'_{t-2}, \ldots, Q'_1 = q'_1) = P(Q_{t-n:t} = q_{t-n:t} | Q_{1:t-1} = q_{1:t-1})$$

$$= P(Q_{t-n:t} = q_{t-n:t} | Q_{t-n-1:t-1} = q_{t-n-1:t-1})$$

$$= P(Q'_t = q'_t | Q'_{t-1} = q'_{t-1})$$

- Given a large enough state space, first-order Markov chain may represent any $n^{th}$-order Markov, but with exponential in $n$ state space.

- Can use a single integer to represent $Q'_t$. 
The statistical evolution of a 1st-order Markov chain is determined by the state transition probabilities \( a_{ij}(t) \triangleq P(Q_t = j | Q_{t-1} = i) \).

function both of the states at successive time steps and of the current time \( t \).

sometimes, there is no dependence on \( t \), and this is called \textit{time-homogeneous} (or just \textit{homogeneous}) because \( a_{ij}(t) = a_{ij} \) for all \( t \).

Hence, a transition matrix \( A \) with \((i,j)^{th}\) entry \( a_{ij} \) is sufficient to represent all transition probabilities in a time-homogeneous 1st-order Markov chain.

A time-homogeneous \( n^{th}\)-order Markov chain \((n \geq 0)\), can represent the transition probabilities using a \((n + 1)^{st}\)-order tensor.
Homogeneous Markov Chain - Transition Matrix/Graph

A =

\[
\begin{pmatrix}
  a_{00} & a_{01} & a_{02} & 0 & 0 & 0 & 0 & 0 \\
  0 & a_{11} & 0 & a_{13} & a_{14} & 0 & 0 & 0 \\
  0 & 0 & a_{22} & a_{23} & 0 & 0 & a_{26} & 0 \\
  0 & 0 & 0 & a_{33} & a_{34} & a_{35} & 0 & 0 \\
  0 & 0 & 0 & 0 & 0 & 0 & a_{46} & 0 \\
  0 & 0 & 0 & 0 & 0 & 0 & 0 & a_{57} \\
  a_{61} & 0 & 0 & 0 & 0 & 0 & 0 & a_{67} \\
  a_{70} & 0 & 0 & 0 & 0 & 0 & 0 & a_{77}
\end{pmatrix}
\]

- Important: don’t confuse graph used to represent transition matrix (pictured above) and graph used for graphical model!
Homogeneous and Bayesian plate model

A time-homogeneous 1st-order Markov chain can be viewed as a Bayesian model with plate notation.

\[ y_1 
  \hline
  y_2 
  \hline
  y_3 
  \hline
  y_4 
  \hline
  y_5
\]
A time-homogeneous 1st-order Markov chain can be viewed as a Bayesian model with plate notation.

For non-random parameters, we have that the transition matrix $A$ is a constant random variable (meaning that $p(A = \bar{A}) = 1$ for some stochastic transition matrix $\bar{A}$).
Higher order Markov chains and plate templates

Q: Can a 2nd order Markov chain be described by a standard plate model?
Markov chains: types of states

- A state $i$ is said to be **transient** if, after visiting the state, it is possible for it to be never visited again, i.e.,:

  \[ p(Q_n = i \text{ for some } n > t | Q_t = i) < 1. \]

- A state $i$ is said to be **null-recurrent** if it is not transient but the expected return time is infinite, i.e.,

  \[ E[\min\{n > t : Q_n = i\} | Q_t = i] = \infty \]  \hspace{1cm} (2.1)

- A state is **positive-recurrent** if it is not transient and the expected return time to that state is finite.

- For a Markov chain with a finite number of states, a state can be only either transient or positive-recurrent.
State Types

- State types, example:
State Types

- State types, example:

  \[ 
  q_0 \rightarrow q_1 \rightarrow q_2 \rightarrow q_3 \rightarrow q_2 \rightarrow q_3 \rightarrow q_0 \]

- What is the type of each state?
State types, example:

- State types, example:

State types, example:

- What is the type of each state?
- $q_0$ is transient while states $q_1$, $q_2$, and $q_3$ are positive-recurrent.
Stationarity for time-homogeneous 1st-order Markov chain

If $Q_t$ is a time-homogeneous stationary first-order Markov chain

then:

$$P(Q_{t_1} = q_1, Q_{t_2} = q_2, \ldots, Q_{t_n} = q_n)$$

$$= P(Q_{t_1+h} = q_1, Q_{t_2+h} = q_2, \ldots, Q_{t_n+h} = q_n)$$

for all $t_i$, $h$, $n$, and $q_i$. 

(2.2)

(2.3)
Stationarity for time-homogeneous 1st-order Markov chain

- If $Q_t$ is a time-homogeneous stationary first-order Markov chain then:

$$P(Q_{t_1} = q_1, Q_{t_2} = q_2, \ldots, Q_{t_n} = q_n)$$

$$= P(Q_{t_1+h} = q_1, Q_{t_2+h} = q_2, \ldots, Q_{t_n+h} = q_n)$$

for all $t_i, h, n$, and $q_i$.

- Using the first order Markov property, the above can be written as:

$$P(Q_{t_n} = q_n|Q_{t_{n-1}} = q_{n-1})$$

$$P(Q_{t_{n-1}} = q_{n-1}|Q_{t_{n-2}} = q_{n-2})$$

$$\ldots P(Q_{t_2} = q_2|Q_{t_1} = q_1)P(Q_{t_1} = q_1)$$

$$= P(Q_{t_n+h} = q_n|Q_{t_{n-1}+h} = q_{n-1})$$

$$P(Q_{t_{n-1}+h} = q_{n-1}|Q_{t_{n-2}+h} = q_{n-2})$$

$$\ldots P(Q_{t_2+h} = q_2|Q_{t_{1}+h} = q_1)P(Q_{t_{1}+h} = q_1)$$
Therefore, a time-homogeneous Markov chain is stationary iff \( \forall t \)

\[
P(Q_{t_1} = q) = P(Q_{t_1+h} = q) = P(Q_t = q) \text{ for all } q \in D_Q \quad (2.10)
\]
Therefore, a time-homogeneous Markov chain is stationary iff $\forall t$

$$P(Q_{t_1} = q) = P(Q_{t_1+h} = q) = P(Q_t = q) \text{ for all } q \in D_Q \quad (2.10)$$

- Let $\xi$ be a distribution on states (so $\xi_i = p(Q_t = i)$).
Therefore, a time-homogeneous Markov chain is stationary iff \( \forall t \)

\[
P(Q_{t_1} = q) = P(Q_{t_1+h} = q) = P(Q_t = q) \text{ for all } q \in D_Q \quad (2.10)
\]

Let \( \xi \) be a distribution on states (so \( \xi_i = p(Q_t = i) \)).

Then, stationary distribution \( \xi \) has the property that \( \xi A = \xi \)

implying that \( \xi \) must be a left eigenvector of the transition matrix \( A \).
Stationarity for time-homogeneous 1st-order Markov chain

Therefore, a time-homogeneous Markov chain is stationary iff \( \forall t \)

\[
P(Q_{t_1} = q) = P(Q_{t_1+h} = q) = P(Q_t = q) \text{ for all } q \in D_Q \quad (2.10)
\]

Let \( \xi \) be a distribution on states (so \( \xi_i = p(Q_t = i) \)).

Then, stationary distribution \( \xi \) has the property that \( \xi A = \xi \)

implying that \( \xi \) must be a left eigenvector of the transition matrix \( A \).

Example: let \( p_1 = [.5, .5] \) be the current distribution over 2-state Markov chain.
Stationarity for time-homogeneous 1st-order Markov chain

- Therefore, a time-homogeneous Markov chain is stationary iff $\forall t$

\[ P(Q_{t_1} = q) = P(Q_{t_1+h} = q) = P(Q_t = q) \text{ for all } q \in D_Q \quad (2.10) \]

- Let $\xi$ be a distribution on states (so $\xi_i = p(Q_t = i)$).
- Then, stationary distribution $\xi$ has the property that $\xi A = \xi$
  implying that $\xi$ must be a left eigenvector of the transition matrix $A$.
- Example: let $p_1 = [.5, .5]$ be the current distribution over 2-state Markov chain. Let $A_1 = [.3, .7; .7, .3]$ be the transition matrix.
Stationarity for time-homogeneous 1st-order Markov chain

- Therefore, a time-homogeneous Markov chain is stationary iff \( \forall t \)

\[
P(Q_{t_1} = q) = P(Q_{t_1+h} = q) = P(Q_t = q) \text{ for all } q \in D_Q \quad (2.10)
\]

- Let \( \xi \) be a distribution on states (so \( \xi_i = p(Q_t = i) \)).
- Then, stationary distribution \( \xi \) has the property that \( \xi A = \xi \)
  implying that \( \xi \) must be a left eigenvector of the transition matrix \( A \).
- Example: let \( p_1 = [.5, .5] \) be the current distribution over 2-state Markov chain. Let \( A_1 = [.3, .7; .7, .3] \) be the transition matrix. The Markov chain is stationary since \( p_1 A_1 = p_1 \).
Stationarity for time-homogeneous 1st-order Markov chain

- Therefore, a time-homogeneous Markov chain is stationary iff $\forall t$

$$P(Q_{t_1} = q) = P(Q_{t_1+h} = q) = P(Q_t = q) \text{ for all } q \in D_Q \quad (2.10)$$

- Let $\xi$ be a distribution on states (so $\xi_i = p(Q_t = i)$).

- Then, stationary distribution $\xi$ has the property that $\xi A = \xi$ implying that $\xi$ must be a left eigenvector of the transition matrix $A$.

- Example: let $p_1 = [.5, .5]$ be the current distribution over 2-state Markov chain. Let $A_1 = [.3, .7; .7, .3]$ be the transition matrix. The Markov chain is stationary since $p_1 A_1 = p_1$. If the current distribution is $p_2 = [.4, .6]$, however, then $p_2 A_1 \neq p_2$, so the chain is no longer stationary (even with same transition matrix).
Stationarity for time-homogeneous 1st-order Markov chain

- Therefore, a time-homogeneous Markov chain is stationary iff ∀t

\[ P(Q_{t_1} = q) = P(Q_{t_1+h} = q) = P(Q_t = q) \text{ for all } q \in D_Q \quad (2.10) \]

- Let \( \xi \) be a distribution on states (so \( \xi_i = p(Q_t = i) \)).

- Then, stationary distribution \( \xi \) has the property that \( \xi A = \xi \)

implying that \( \xi \) must be a left eigenvector of the transition matrix \( A \).

- Example: let \( p_1 = [.5, .5] \) be the current distribution over 2-state Markov chain. Let \( A_1 = [.3, .7; .7, .3] \) be the transition matrix. The Markov chain is stationary since \( p_1 A_1 = p_1 \). If the current distribution is \( p_2 = [.4, .6] \), however, then \( p_2 A_1 \neq p_2 \), so the chain is no longer stationary (even with same transition matrix).

- There can exist more than 1 stationary distribution.
Stationarity for time-homogeneous 1st-order Markov chain

- Therefore, a time-homogeneous Markov chain is stationary iff $\forall t$

$$P(Q_{t_1} = q) = P(Q_{t_1+h} = q) = P(Q_t = q) \text{ for all } q \in \mathbb{D}_Q \quad (2.10)$$

- Let $\xi$ be a distribution on states (so $\xi_i = p(Q_t = i)$).

- Then, stationary distribution $\xi$ has the property that $\xi A = \xi$ implying that $\xi$ must be a left eigenvector of the transition matrix $A$.

- Example: let $p_1 = [.5, .5]$ be the current distribution over 2-state Markov chain. Let $A_1 = [.3, .7; .7, .3]$ be the transition matrix. The Markov chain is stationary since $p_1 A_1 = p_1$. If the current distribution is $p_2 = [.4, .6]$, however, then $p_2 A_1 \neq p_2$, so the chain is no longer stationary (even with same transition matrix).

- There can exist more than 1 stationary distribution.

- Exercise: Must there exist a stationary distribution in a time-homogeneous Markov chain?
Stationarity implies Homogeneity

If a 1st-order Markov chain is stationary, then

\[ P(Q_t = i, Q_{t-1} = j) = P(Q_{t-1} = i, Q_{t-2} = j) \]  \hspace{1cm} (2.11)

and

\[ P(Q_t = i) = P(Q_{t-1} = i) \] \hspace{1cm} (2.12)
Stationarity implies Homogeneity

- If a 1st-order Markov chain is stationary, then

\[ P(Q_t = i, Q_{t-1} = j) = P(Q_{t-1} = i, Q_{t-2} = j) \]  \hspace{1cm} (2.11)

and

\[ P(Q_t = i) = P(Q_{t-1} = i) \]  \hspace{1cm} (2.12)

- Therefore,

\[ a_{ij}(t) = \frac{P(Q_t = i, Q_{t-1} = j)}{P(Q_{t-1} = j)} \]  \hspace{1cm} (2.13)

\[ = \frac{P(Q_{t-1} = i, Q_{t-2} = j)}{P(Q_{t-2} = j)} \]  \hspace{1cm} (2.14)

\[ = a_{ij}(t-1) \]  \hspace{1cm} (2.15)
Stationarity implies Homogeneity

- If a 1st-order Markov chain is stationary, then

\[ P(Q_t = i, Q_{t-1} = j) = P(Q_{t-1} = i, Q_{t-2} = j) \]  

(2.11)

and

\[ P(Q_t = i) = P(Q_{t-1} = i) \]  

(2.12)

- Therefore,

\[ a_{ij}(t) = P(Q_t = i, Q_{t-1} = j)/P(Q_{t-1} = j) \]  

(2.13)

\[ = P(Q_{t-1} = i, Q_{t-2} = j)/P(Q_{t-2} = j) \]  

(2.14)

\[ = a_{ij}(t - 1) \]  

(2.15)

- and by induction \( a_{ij}(t) = a_{ij}(t + \tau) \) for all \( \tau \), meaning the stochastic transition matrix is the same for all time, \( A \).
Example: inhomogeneous and stationary

- Let \( A_t = [0.3, 0.7; 0.7, 0.3] \) when \( t \) is even and \( A_t = [0.4, 0.6; 0.6, 0.4] \) when \( t \) is odd, so chain is inhomogeneous.
Example: inhomogeneous and stationary

- Let $A_t = [.3, .7; .7, .3]$ when $t$ is even and $A_t = [.4, .6; .6, .4]$ when $t$ is odd, so chain is inhomogeneous.
- If the current state distribution is $p = [.5, .5]$, then $pA_t = p$ for $t$ both even and odd.
Example: inhomogeneous and stationary

- Let $A_t = [.3, .7; .7, .3]$ when $t$ is even and $A_t = [.4, .6; .6, .4]$ when $t$ is odd, so chain is inhomogeneous.
- If the current state distribution is $p = [.5, .5]$, then $pA_t = p$ for $t$ both even and odd.
- Is this stationary?
Example: inhomogeneous and stationary

- Let $A_t = [.3, .7; .7, .3]$ when $t$ is even and $A_t = [.4, .6; .6, .4]$ when $t$ is odd, so chain is inhomogeneous.
- If the current state distribution is $p = [.5, .5]$, then $pA_t = p$ for $t$ both even and odd.
- Is this stationary?
- Note that this is not a stationary distribution. When $t$ is even, we have that $p(Q_t = 0, Q_{t+1} = 1) = 0.5 \times 0.7$ but when $t$ is odd, $p(Q_t = 0, Q_{t+1} = 1) = 0.5 \times 0.6$, so the chain does not exhibit a stationary distribution according to the definition.
Example: inhomogeneous and stationary

Let $A_t = [.3, .7; .7, .3]$ when $t$ is even and $A_t = [.4, .6; .6, .4]$ when $t$ is odd, so chain is inhomogeneous.

If the current state distribution is $p = [.5, .5]$, then $pA_t = p$ for $t$ both even and odd.

Is this stationary?

Note that this is not a stationary distribution. When $t$ is even, we have that $p(Q_t = 0, Q_{t+1} = 1) = 0.5 \times 0.7$ but when $t$ is odd, $p(Q_t = 0, Q_{t+1} = 1) = 0.5 \times 0.6$, so the chain does not exhibit a stationary distribution according to the definition.

Aforementioned criterion for stationary ($\xi A = \xi$) requires a homogeneous chain. In general, $\xi A_t = \xi$ alone does not guarantee stationarity.
Example: inhomogeneous and stationary

- Let $A_t = [0.3, 0.7; 0.7, 0.3]$ when $t$ is even and $A_t = [0.4, 0.6; 0.6, 0.4]$ when $t$ is odd, so chain is inhomogeneous.
- If the current state distribution is $p = [0.5, 0.5]$, then $pA_t = p$ for $t$ both even and odd.
- Is this stationary?
- Note that this is not a stationary distribution. When $t$ is even, we have that $p(Q_t = 0, Q_{t+1} = 1) = 0.5 \times 0.7$ but when $t$ is odd, $p(Q_t = 0, Q_{t+1} = 1) = 0.5 \times 0.6$, so the chain does not exhibit a stationary distribution according to the definition.
- Aforementioned criterion for stationary ($\xi A = \xi$) requires a homogeneous chain. In general, $\xi A_t = \xi$ alone does not guarantee stationarity.
- In general, important to realize that stationarity and homogeneity of a Markov chain (or any DGM) are distinct properties. If stationary, then homogeneous. If homogeneous, then might or might not be stationary.
Stationarity and probability flow

- Probability “flow” can help to give intuition behind stationary distributions.

\[ \text{For stationarity, probability outflow must equal probability inflow for every state!} \]

\[ \text{Stationary, or } \xi_A = \xi, \text{ implies that for all } \]

\[ \xi_i = \sum_j \xi_j a_{ji} \quad (2.16) \]

or equivalently,

\[ \xi_i (1 - a_{ii}) = \sum_{j \neq i} \xi_j a_{ji} \quad (2.17) \]

Note that

\[ 1 - a_{ii} = \sum_{j \neq i} a_{ij} \quad (2.18) \]
Stationarity and probability flow

- Probability “flow” can help to give intuition behind stationary distributions.
- For stationarity, probability outflow must equal probability inflow for every state!
Stationarity and probability flow

- Probability “flow” can help to give intuition behind stationary distributions.
- For stationarity, probability outflow must equal probability inflow for every state!
- Stationary, or $\xi A = \xi$, implies that for all $i$

$$\xi_i = \sum_j \xi_j a_{ji} \quad (2.16)$$

Note that $1 - a_{ii} = \sum_{j \neq i} a_{ij} \quad (2.18)$
Stationarity and probability flow

- Probability “flow” can help to give intuition behind stationary distributions.
- For stationarity, probability outflow must equal probability inflow for every state!
- Stationary, or $\xi A = \xi$, implies that for all $i$
  \[
  \xi_i = \sum_j \xi_j a_{ji} \quad (2.16)
  \]
  or equivalently,
  \[
  \xi_i (1 - a_{ii}) = \sum_{j \neq i} \xi_j a_{ji} \quad (2.17)
  \]
Stationarity and probability flow

• Probability “flow” can help to give intuition behind stationary distributions.
• For stationarity, probability outflow must equal probability inflow for every state!
• Stationary, or $\xi A = \xi$, implies that for all $i$

$$\xi_i = \sum_j \xi_j a_{ji} \quad (2.16)$$

• or equivalently,

$$\xi_i (1 - a_{ii}) = \sum_{j \neq i} \xi_j a_{ji} \quad (2.17)$$

• Note that

$$1 - a_{ii} = \sum_{j \neq i} a_{ij} \quad (2.18)$$
Therefore, the stationary condition can be stated as, \( \forall i \)

\[
\sum_{j \neq i} \xi_i a_{ij} = \sum_{j \neq i} \xi_j a_{ji} \tag{2.19}
\]
Stationarity and probability flow

Therefore, the stationary condition can be stated as, \( \forall i \)

\[
\sum_{j \neq i} \xi_i a_{ij} = \sum_{j \neq i} \xi_j a_{ji} \tag{2.19}
\]

Left side interpreted as probability flow out of state \( i \), right side interpreted as flow into state \( i \).
Stationarity and probability flow

Therefore, the stationary condition can be stated as, \( \forall i \)

\[
\sum_{j \neq i} \xi_i a_{ij} = \sum_{j \neq i} \xi_j a_{ji}
\]  

(2.19)

Left side interpreted as probability flow out of state \( i \), right side interpreted as flow into state \( i \).

Example with \( i = 6 \):

![Diagram showing probability flow with states and transitions labeled \( \xi_1, \xi_2, \xi_3, \xi_4,\xi_5,\xi_6,\xi_7,\xi_8,\xi_9 \) and transitions \( a_{16}, a_{26}, a_{36}, a_{46}, a_{56}, a_{67}, a_{68}, a_{69} \).]
Let $A$ be a transition matrix for a 1st-order time-homogeneous Markov chain.

**Proposition 2.3.1**

The transition matrix $A = [a_{ij}]_{i,j}$ is a stochastic matrix, meaning:

(a) $A$ has non-negative entries, so that $a_{ij} \geq 0 \ \forall i,j$

(b) $A$ has row-sums equal to one $\sum_j a_{ij} = 1$. 
Chapman-Kolmogorov equations
Computing transition probabilities over $k$ steps

- Given time-homogeneous 1st-order chain, let

$$a^k_{ij} \triangleq p(Q_{t+k} = j | Q_t = i), \quad (2.20)$$

meaning probability of transitioning from state $i$ to state $j$ in exactly $k$ steps.
Given time-homogeneous 1st-order chain, let

\[ a_{ij}^k \triangleq p(Q_{t+k} = j | Q_t = i), \quad (2.20) \]

meaning probability of transitioning from state \( i \) to state \( j \) in exactly \( k \) steps.

Then

\[ a_{ij}^2 = p(Q_{t+2} = j | Q_t = i) \]

\[ = \sum_{\ell} p(Q_{t+2} = j | Q_{t=1} = \ell) p(Q_{t=1} = \ell | Q_t = i) \quad (2.22) \]

\[ = \sum_{\ell} a_{i\ell} a_{\ell j} \quad (2.23) \]
Chapman-Kolmogorov equations

- We can generalize this immediately to any \( k > 1 \) steps to give:

\[
a_{ij}^k = \sum_{\ell} a_{i\ell}^m a_{\ell j}^n
\]

where \( m, n \geq 0 \) with \( k = m + n \).
Chapman-Kolmogorov equations

- We can generalize this immediately to any $k > 1$ steps to give:

$$a_{ij}^k = \sum_\ell a_{i\ell}^m a_{\ell j}^n$$ (2.24)

where $m, n \geq 0$ with $k = m + n$.

- Can be further generalized to $n^{th}$-order and/or time-inhomogeneous chains. Example, we can define:

$$a_{ij}(t, t + \tau) = p(Q_{t+\tau} = i | Q_t = i)$$ (2.25)

for $\tau > 0$. 
Definition 2.3.2 (period)

The period $d(i)$ of a state is defined as $d(i) = \gcd \{ n : a_{ii}^n > 0 \}$, the greatest common divisor of the time intervals where return is possible.
Period of a state

Definition 2.3.2 (period)
The period \( d(i) \) of a state is defined as \( d(i) = \gcd \{ n : a_{ii}^n > 0 \} \), the greatest common divisor of the time intervals where return is possible.

Definition 2.3.3 (periodic/aperiodic)
A state \( i \) is said to be periodic if \( d(i) > 1 \) and aperiodic if \( d(i) = 1 \).
**Definition 2.3.2 (period)**

The period $d(i)$ of a state is defined as $d(i) = \gcd \{ n : a_{ii}^n > 0 \}$, the greatest common divisor of the time intervals where return is possible.

**Definition 2.3.3 (periodic/aperiodic)**

A state $i$ is said to be **periodic** if $d(i) > 1$ and **aperiodic** if $d(i) = 1$.

**Ex 1** If all states have self-loops, period is 1.

**Ex 2** concentric loops, in one case loops of length 3,5, period of state $i$ is 1.

**Ex 3** another case of length 3,6, period of state $i$ is 3.
Period of a state

- Period of a state is a probabilistic concept
Period of a state

- Period of a state is a probabilistic concept
- if period is > 1, then there will be certain times where state is impossible (e.g., 3,6 case above).
Period of a state

- Period of a state is a probabilistic concept
- If period is \( > 1 \), then there will be certain times where state is impossible (e.g., 3,6 case above).
- If period = 1 (i.e., return times are co-prime), eventually, all states will have positive probability at successive time intervals. Exercise: When is the earliest \( t \) when this happens in general?
Period of a state

- Period of a state is a probabilistic concept
- If period is $> 1$, then there will be certain times where state is impossible (e.g., 3,6 case above).
- If period = 1 (i.e., return times are co-prime), eventually, all states will have positive probability at successive time intervals. **Exercise:** When is the earliest $t$ when this happens in general?
- I.e., let $\mathbb{N} = \{1, 2, \ldots\}$ be the natural numbers, $i\mathbb{N} = \{i, 2i, 3i, \ldots\}$ for $i \in \mathbb{N}$, and $A + B = \{c : \exists a \in A, b \in B \text{ w. } c = a + b\}$. 

[Diagram showing state transitions]
Period of a state

- Period of a state is a probabilistic concept
- If period is $> 1$, then there will be certain times where state is impossible (e.g., 3, 6 case above).
- If period = 1 (i.e., return times are co-prime), eventually, all states will have positive probability at successive time intervals. Exercise: When is the earliest $t$ when this happens in general?

I.e., let $\mathbb{N} = \{1, 2, \ldots \}$ be the natural numbers, $i\mathbb{N} = \{i, 2i, 3i, \ldots \}$ for $i \in \mathbb{N}$, and $A + B = \{c : \exists a \in A, b \in B \text{ w. } c = a + b\}$.

Then, in the above, we have that $3\mathbb{N} + 5\mathbb{N} = \mathbb{N} \cap \{i : i \geq 3 + 5\}$. 
State “communication”

Some definitions:

- State $i$ communicates with state $j$ ($i \rightarrow j$) if $a^k_{ij} > 0$ for some $k \geq 0$. States intercommunicate if $i \rightarrow j$ and $j \rightarrow i$, written $i \leftrightarrow j$.

- Set of states $C$ is closed if $a_{ij} = 0$ for $i \in C$ and $j \notin C$.

- Set of states $C$ is irreducible if $i \leftrightarrow j$ for all $i, j \in C$. 
State “communication”

Some definitions:

- State $i$ communicates with state $j$ ($i \rightarrow j$) if $a_{ij}^k > 0$ for some $k \geq 0$. States intercommunicate if $i \rightarrow j$ and $j \rightarrow i$, written $i \leftrightarrow j$.
- Set of states $C$ is closed if $a_{ij} = 0$ for $i \in C$ and $j \notin C$.
- Set of states $C$ is irreducible if $i \leftrightarrow j$ for all $i, j \in C$.

Theorem 2.3.4

If $i \leftrightarrow j$ then

(a) $i$ and $j$ have same period

(b) $i$ is transient iff $j$ is transient

(c) $i$ is null-recurrent iff $j$ is null-recurrent
State partitioning

**Theorem 2.3.5**

Set of states can be partitioned uniquely as

\[ S = T \cup C_1 \cup C_2 \cup \ldots \]  \hspace{1cm} (2.26)

where \( T \) is the set of transient states, and \( C_i \) are sets of irreducible closed sets of recurrent states.
State partitioning

**Theorem 2.3.5**

*Set of states can be partitioned uniquely as*

\[ S = T \cup C_1 \cup C_2 \cup \ldots \]  \hspace{1cm} (2.26)

*where T is the set of transient states, and C_i are sets of irreducible closed sets of recurrent states.*

- Intuition: When we run a Markov chain (by choosing the next state according to \( a_{ij} \)), states are either such that they eventually are impossible to reach, or we get into a “rut” where only a subset of the states are reachable, and we can never break out of that rut.
State partitioning

**Theorem 2.3.5**

*Set of states can be partitioned uniquely as*

\[ S = T \cup C_1 \cup C_2 \cup \ldots \]  

(2.26)

*where T is the set of transient states, and C_i are sets of irreducible closed sets of recurrent states.*

- **Intuition:** When we run a Markov chain (by choosing the next state according to \( a_{ij} \)), states are either such that they eventually are impossible to reach, or we get into a “rut” where only a subset of the states are reachable, and we can never break out of that rut.

- **It might be** that no states are transient and there is only one \( C_i \) (e.g., A matrix with all strictly positive entries). The entire chain is then irreducible.
State communication/partitioning

Example 2.3.6

\[ A = \begin{pmatrix}
\frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 & 0 \\
\frac{1}{4} & \frac{3}{4} & 0 & 0 & 0 & 0 \\
\frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & 0 & 0 \\
\frac{1}{4} & 0 & \frac{1}{4} & \frac{1}{4} & 0 & \frac{1}{4} \\
0 & 0 & 0 & 0 & \frac{1}{2} & \frac{1}{2} \\
0 & 0 & 0 & 0 & \frac{1}{2} & \frac{1}{2}
\end{pmatrix} \]  
(2.27)

States \{1, 2\} and \{5, 6\} are irreducible and closed and recurrent. States 3 and 4 are transient since 3 \rightarrow 4 \rightarrow 6 but we can't get back from 6.

The period of all states is 1 (since \(a_{ii} > 0\)).

Exercise: Derive additional Markov chain matrices such that you have variants of the above (i.e., a case with no T, a case with only one C, a case with \(>1\) C, and so on).
Example 2.3.6

\[
A = \begin{pmatrix}
1/2 & 1/2 & 0 & 0 & 0 & 0 \\
1/4 & 3/4 & 0 & 0 & 0 & 0 \\
1/4 & 1/4 & 1/4 & 1/4 & 0 & 0 \\
1/4 & 0 & 1/4 & 1/4 & 0 & 1/4 \\
0 & 0 & 0 & 0 & 1/2 & 1/2 \\
0 & 0 & 0 & 0 & 1/2 & 1/2
\end{pmatrix}
\]  
(2.27)

States \{1, 2\} and \{5, 6\} are irreducible and closed and recurrent.
Example 2.3.6

\[
A = \begin{pmatrix}
1/2 & 1/2 & 0 & 0 & 0 & 0 & 0 \\
1/4 & 3/4 & 0 & 0 & 0 & 0 & 0 \\
1/4 & 1/4 & 1/4 & 1/4 & 0 & 0 \\
1/4 & 0 & 1/4 & 1/4 & 0 & 1/4 \\
0 & 0 & 0 & 0 & 1/2 & 1/2 \\
0 & 0 & 0 & 0 & 1/2 & 1/2 \\
\end{pmatrix}
\] (2.27)

States \{1, 2\} and \{5, 6\} are irreducible and closed and recurrent. States 3 and 4 are transient since 3 \rightarrow 4 \rightarrow 6 but we can’t get back from 6.
State communication/partitioning

Example 2.3.6

\[
A = \begin{pmatrix}
1/2 & 1/2 & 0 & 0 & 0 & 0 \\
1/4 & 3/4 & 0 & 0 & 0 & 0 \\
1/4 & 1/4 & 1/4 & 1/4 & 0 & 0 \\
1/4 & 0 & 1/4 & 1/4 & 0 & 1/4 \\
0 & 0 & 0 & 0 & 1/2 & 1/2 \\
0 & 0 & 0 & 0 & 1/2 & 1/2 \\
\end{pmatrix}
\]

States \( \{1, 2\} \) and \( \{5, 6\} \) are irreducible and closed and recurrent. States 3 and 4 are transient since \( 3 \rightarrow 4 \rightarrow 6 \) but we can’t get back from 6. The period of all states is 1 (since \( a_{ii} > 0 \)).
Markov chains
HMMs
HMMs as GMs
What HMMs can do
Summary
Scratch

State communication/partitioning

Example 2.3.6

\[
A = \begin{pmatrix}
\frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 & 0 \\
\frac{1}{4} & \frac{3}{4} & 0 & 0 & 0 & 0 \\
\frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & 0 & 0 \\
\frac{1}{4} & 0 & \frac{1}{4} & \frac{1}{4} & 0 & \frac{1}{4} \\
0 & 0 & 0 & 0 & \frac{1}{2} & \frac{1}{2} \\
0 & 0 & 0 & 0 & \frac{1}{2} & \frac{1}{2}
\end{pmatrix}
\] (2.27)

States \{1, 2\} and \{5, 6\} are irreducible and closed and recurrent. States 3 and 4 are transient since 3 \rightarrow 4 \rightarrow 6 but we can’t get back from 6. The period of all states is 1 (since \(a_{ii} > 0\)).

Exercise: Derive additional Markov chain matrices such that you have variants of the above (i.e., a case with no \(T\), a case with only one \(C\), a case with \(> 1\) \(C\), and so on).
One aspect of temporal models we wish to study is how does information transfer decay over time.
One aspect of temporal models we wish to study is how does information transfer decay over time.

I.e., for fixed order MC, how bad is the past future|present property?

Lemma 2.3.7

If \( \xi \) is a stationary distribution, then \( \xi = \xi A^\tau \) for any \( \tau > 1 \).
Long-term evolution

- One aspect of temporal models we wish to study is how does information transfer decay over time.
- I.e., for fixed order MC, how bad is the past \( \perp \) future | present property?
- Example: protein folding, many long term interactions, how poor would it be to make such a Markov assumption? Another example: statistical machine translation (permutations, expansions, contractions).
Long-term evolution

- One aspect of temporal models we wish to study is how does information transfer decay over time.
- I.e., for fixed order MC, how bad is the past $\perp$ future $\mid$ present property?
- Example: protein folding, many long term interactions, how poor would it be to make such a Markov assumption? Another example: statistical machine translation (permutations, expansions, contractions).
- Chapman-Kolmogorov allows us to begin to look at that. Let $\xi_t$ be the distribution at time $t$, so that $\xi_{t+1} = \xi_t A$. 
Long-term evolution

- One aspect of temporal models we wish to study is how does information transfer decay over time.
- I.e., for fixed order MC, how bad is the past \perp \text{future} \mid \text{present} property?
- Example: protein folding, many long term interactions, how poor would it be to make such a Markov assumption? Another example: statistical machine translation (permutations, expansions, contractions).
- Chapman-Kolmogorov allows us to begin to look at that. Let $\xi^t$ be the distribution at time $t$, so that $\xi^{t+1} = \xi^t A$. Then

**Lemma 2.3.7**

$$\xi^{t+\tau} = \xi^t A^\tau$$  \hspace{1cm} (2.28)

where $A^\tau = A \times A \times \cdots \times A$ is the $\tau$th power of $\tau$. \hspace{1cm} \text{\tau times}
One aspect of temporal models we wish to study is how does information transfer decay over time.

I.e., for fixed order MC, how bad is the \( \text{past} \perp \text{future} \mid \text{present} \) property?

Example: protein folding, many long term interactions, how poor would it be to make such a Markov assumption? Another example: statistical machine translation (permutations, expansions, contractions).

Chapman-Kolmogorov allows us to begin to look at that. Let \( \xi_t \) be the distribution at time \( t \), so that \( \xi_{t+1} = \xi_t A \). Then

**Lemma 2.3.7**

\[
\xi^{t+\tau} = \xi^t A^\tau \tag{2.28}
\]

where \( A^\tau = A \times A \times \cdots \times A \) is the \( \tau \)th power of \( \tau \).

If \( \xi \) is a stationary distribution, then \( \xi = \xi A^\tau \) for any \( \tau > 1 \).
Note: For every component $C_i$ in the above decomposition, there is a unique stationary distribution. If the entire chain is irreducible, there is a single unique stationary distribution. In fact, we have

**Theorem 2.3.8 (Perron-Frobenius)**

*If $A$ is the $N \times N$ transition matrix of a finite irreducible chain with period $d$, then*

(a) $\lambda_1 = 1$ is an eigenvalue of $A$

(b) The $d$ complex roots of unity, $\lambda_k = W_d^k$, $k \in \{0, \ldots, d - 1\}$ with $W_d = \exp(2\pi\sqrt{-1}/d)$ are eigenvalues of $A$

(c) The remaining eigenvalues $\lambda_{d+1}, \ldots, \lambda_N$ satisfy $|\lambda_j| < 1$. 
Transition Matrices Factorization

- When the eigenvalues $\lambda_1, \ldots, \lambda_N$ are distinct, there exists a matrix $B$ where

$$A^k = B^{-1} \Lambda^n B = B^{-1} \begin{pmatrix} \lambda_1^n & 0 & \cdots & 0 \\ 0 & \lambda_2^n & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_N^n \end{pmatrix} B$$

(rows of $B$ are left eigenvectors of $A$.)
Transition Matrices Factorization

- When the eigenvalues $\lambda_1, \ldots, \lambda_N$ are distinct, there exists a matrix $B$ where

$$A^k = B^{-1} \Lambda^n B = B^{-1} \begin{pmatrix} \lambda_1^n & 0 & \ldots & 0 \\ 0 & \lambda_2^n & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & \lambda_N^n \end{pmatrix} B$$  \hspace{1cm} (2.29)$$

rows of $B$ are left eigenvectors of $A$.

- Therefore, if $d = 1$, we have

$$A^n \rightarrow B^{-1} \begin{pmatrix} 1 & 0 & \ldots & 0 \\ 0 & 0 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & 0 \end{pmatrix} B \text{ as } n \rightarrow \infty$$  \hspace{1cm} (2.30)$$
We also have the following theorem

**Theorem 2.3.9**

If $A$ is the transition matrix of a finite irreducible chain with period $d$, then

$$A^k \to 1\xi^\top$$

(2.31)

where $\xi$ is the unique stationary distribution associated with the Markov chain, and the rate of convergence, moreover, is geometric.
We also have the following theorem

**Theorem 2.3.9**

If $A$ is the transition matrix of a finite irreducible chain with period $d$, then

$$A^k \rightarrow 1\xi^T$$

(2.31)

where $\xi$ is the unique stationary distribution associated with the Markov chain, and the rate of convergence, moreover, is geometric.

Hence, regardless of the initial distribution where we start, we will approach a (unique if irreducible) stationary distribution of the Markov chain, and we’ll do so quite fast. That is, we will forget from where we started from.
An $k^{th}$-order Markov chain is a $k$-tree.
An $k^{th}$-order Markov chain is a $k$-tree.

Hence, exact inference costs (with the goal a distribution over a clique or subclique), would be $O(r^{k+1})$. 
An $k^{th}$-order Markov chain is a $k$-tree.

Hence, exact inference costs (with the goal a distribution over a clique or subclique), would be $O(r^{k+1})$.

The structure (and “meaning”) of the Markov chain is entirely encoded in the transition matrix $[a_{ij}]_{i,j}$. 
Many applications of Markov chains

- Random walk on an integer point line, with transition matrix $A = I + e_i^T e_i$, where $e_i$ is the $i$-th standard basis vector.

- Markov Chain Monte Carlo (MCMC) - goal is to generate samples of some difficult distribution $p(x)$. $p(x)$ corresponds to a stationary distribution of the Markov chain.

- Stochastic Finite State Automata - any discrete time series could be modeled by this. E.g., Language (n-gram modeling in speech/natural language processing).

- Various Genomic & Proteomic Sequencing

- And so on.

Many applications listed in Grimmett & Stirzaker text.

We will revisit Markov chains repeatedly in this course, since many of the properties of DGMs have properties generalized from a Markov chain.
Markov Chains, some applications

- ∃ many applications of Markov chains
  - Random walk on a integer point line, with $a_{ij} = 0$ unless $j = i \pm 1$. 
Markov Chains, some applications

- ∃ many applications of Markov chains
  - Random walk on a integer point line, with $a_{ij} = 0$ unless $j = i \pm 1$.
  - Markov Chain Monte Carlo - goal is to generate samples of some
difficult distribution $p(x)$. $p(x)$ corresponds to a stationary
distribution of the Markov chain.
∃ many applications of Markov chains

- Random walk on a integer point line, with $a_{ij} = 0$ unless $j = i \pm 1$.
- Markov Chain Monte Carlo - goal is to generate samples of some difficult distribution $p(x)$. $p(x)$ corresponds to a stationary distribution of the Markov chain.
- Stochastic Finite State Automata - any discrete time series could be modeled by this. E.g., Language ($n$-gram modeling in speech/natural language processing).
∃ many applications of Markov chains

- Random walk on a integer point line, with $a_{ij} = 0$ unless $j = i \pm 1$.
- Markov Chain Monte Carlo - goal is to generate samples of some difficult distribution $p(x)$. $p(x)$ corresponds to a stationary distribution of the Markov chain.
- Stochastic Finite State Automata - any discrete time series could be modeled by this. E.g., Language ($n$-gram modeling in speech/natural language processing).
- Various Genomic & Proteomic Sequencing
∃ many applications of Markov chains

- Random walk on a integer point line, with $a_{ij} = 0$ unless $j = i \pm 1$.
- Markov Chain Monte Carlo - goal is to generate samples of some difficult distribution $p(x)$. $p(x)$ corresponds to a stationary distribution of the Markov chain.
- Stochastic Finite State Automata - any discrete time series could be modeled by this. E.g., Language ($n$-gram modeling in speech/natural language processing).
- Various Genomic & Proteomic Sequencing
- And so on.
Exists many applications of Markov chains

- Random walk on a integer point line, with $a_{ij} = 0$ unless $j = i \pm 1$.
- Markov Chain Monte Carlo - goal is to generate samples of some difficult distribution $p(x)$. $p(x)$ corresponds to a stationary distribution of the Markov chain.
- Stochastic Finite State Automata - any discrete time series could be modeled by this. E.g., Language ($n$-gram modeling in speech/natural language processing).
- Various Genomic & Proteomic Sequencing
- And so on.
- Many applications listed in Grimmett & Stirzaker text.
∃ many applications of Markov chains

- Random walk on a integer point line, with $a_{ij} = 0$ unless $j = i \pm 1$.
- Markov Chain Monte Carlo - goal is to generate samples of some difficult distribution $p(x)$. $p(x)$ corresponds to a stationary distribution of the Markov chain.
- Stochastic Finite State Automata - any discrete time series could be modeled by this. E.g., Language ($n$-gram modeling in speech/natural language processing).
- Various Genomic & Proteomic Sequencing
- And so on.
- Many applications listed in Grimmett & Stirzaker text.
- We will revisit Markov chains repeatedly in this course, since many of the properties of DGMs have properties generalized from a Markov chain.
Hidden Markov Models (HMMs)

- Hidden Markov Models (HMMs) are a ubiquitously used model in many fields, including speech recognition, natural language processing, bioinformatics, financial markets, and many time-series problems.
Hidden Markov Models (HMMs)

- Hidden Markov Models (HMMs) are a ubiquitously used model in many fields, including speech recognition, natural language processing, bioinformatics, financial markets, and many time-series problems.
- HMMs are powerful and diverse enough to be interesting, but still simple enough so that they are often computationally feasible (it is also a good example to get a strong feel for exact probabilistic inference).
Hidden Markov Models (HMMs) are a ubiquitously used model in many fields, including speech recognition, natural language processing, bioinformatics, financial markets, and many time-series problems. HMMs are powerful and diverse enough to be interesting, but still simple enough so that they are often computationally feasible (it is also a good example to get a strong feel for exact probabilistic inference). HMMs can be described in many ways, including using one of a variety of different graphical models (e.g., Bayesian networks or Markov random fields, which means that they are decomposable).
Hidden Markov Models (HMMs) are a ubiquitously used model in many fields, including speech recognition, natural language processing, bioinformatics, financial markets, and many time-series problems.

HMMs are powerful and diverse enough to be interesting, but still simple enough so that they are often computationally feasible (it is also a good example to get a strong feel for exact probabilistic inference).

HMMs can be described in many ways, including using one of a variety of different graphical models (e.g., Bayesian networks or Markov random fields, which means that they are decomposable).

As we will see, HMMs are already triangulated (after moralization if necessary), no triangulation step for forming a junction tree.
Hidden Markov Models (HMMs) are a ubiquitously used model in many fields, including speech recognition, natural language processing, bioinformatics, financial markets, and many time-series problems.

HMMs are powerful and diverse enough to be interesting, but still simple enough so that they are often computationally feasible (it is also a good example to get a strong feel for exact probabilistic inference).

HMMs can be described in many ways, including using one of a variety of different graphical models (e.g., Bayesian networks or Markov random fields, which means that they are decomposable).

As we will see, HMMs are already triangulated (after moralization if necessary), no triangulation step for forming a junction tree.

As we will see, since the clique sizes are size 2, HMMs are easy to deal with ($O(TN^2)$ complexity), at least in theory.
A note on HMM notation

It is common to use either $Q$ and $Q_{1:T}$ or $Y$ and $Y_{1:T}$ as the underlying Markov chain in an HMM.
A note on HMM notation

- It is common to use either $Q$ and $Q_{1:T}$ or $Y$ and $Y_{1:T}$ as the underlying Markov chain in an HMM.
- Speech recognition traditionally has used $Q$ and $Q_{1:T}$.
A note on HMM notation

- It is common to use either $Q$ and $Q_{1:T}$ or $Y$ and $Y_{1:T}$ as the underlying Markov chain in an HMM.
- Speech recognition traditionally has used $Q$ and $Q_{1:T}$.
- Machine learning has typically used $Y$ and $Y_{1:T}$. 
A note on HMM notation

- It is common to use either $Q$ and $Q_{1:T}$ or $Y$ and $Y_{1:T}$ as the underlying Markov chain in an HMM.
- Speech recognition traditionally has used $Q$ and $Q_{1:T}$.
- Machine learning has typically used $Y$ and $Y_{1:T}$.
- We will use both, but there should be no confusion about this (if there is, ask).
What if we don’t observe the Markov chain but it is hidden. Let \( y = y_{1:T} \) be a length \( T \) 1st-order time-homogeneous Markov chain (for the time being, all Markov chains will be such unless otherwise indicated)
What if we don’t observe the Markov chain but it is hidden. Let $y = y_{1:T}$ be a length $T$ 1st-order time-homogeneous Markov chain (for the time being, all Markov chains will be such unless otherwise indicated)

If a sequence $x = x_{1:T}$ is related by $x = f(y)$, with $y = y_{1:T}$ then we have that $x$ is a function of a Markov chain.
HMMs - random functions of a Markov chain

- What if we don’t observe the Markov chain but it is hidden. Let $y = y_{1:T}$ be a length $T$ 1st-order time-homogeneous Markov chain (for the time being, all Markov chains will be such unless otherwise indicated).

- If a sequence $x = x_{1:T}$ is related by $x = f(y)$, with $y = y_{1:T}$ then we have that $x$ is a function of a Markov chain.

- If we have a set functions $\{f_1, f_2, \ldots, f_s\}$, and we have a random variable $S$ with $p(S = s)$, then if we define $X = f_S(Y)$, then $X$ is a probabilistic function of a Markov chain.
HMMs - random functions of a Markov chain

- What if we don’t observe the Markov chain but it is hidden. Let $y = y_1:T$ be a length $T$ 1st-order time-homogeneous Markov chain (for the time being, all Markov chains will be such unless otherwise indicated)

- If a sequence $x = x_1:T$ is related by $x = f(y)$, with $y = y_1:T$ then we have that $x$ is a function of a Markov chain.

- If we have a set functions $\{f_1, f_2, \ldots, f_s\}$, and we have a random variable $S$ with $p(S = s)$, then if we define $X = f_S(Y)$, then $X$ is a probabilistic function of a Markov chain.

- We observe $\bar{x}$ which presumably has been generated by choosing one $f_i$ randomly based on some random assignment to $Y$. $Y$ is therefore hidden.
What if we don’t observe the Markov chain but it is hidden. Let $y = y_{1:T}$ be a length $T$ 1st-order time-homogeneous Markov chain (for the time being, all Markov chains will be such unless otherwise indicated)

If a sequence $x = x_{1:T}$ is related by $x = f(y)$, with $y = y_{1:T}$ then we have that $x$ is a function of a Markov chain.

If we have a set functions $\{f_1, f_2, \ldots, f_s\}$, and we have a random variable $S$ with $p(S = s)$, then if we define $X = f_S(Y)$, then $X$ is a probabilistic function of a Markov chain.

We observe $\bar{x}$ which presumably has been generated by choosing one $f_i$ randomly based on some random assignment to $Y$. $Y$ is therefore hidden.

Hence, the name “Hidden” Markov model.
HMMs - random functions of a Markov chain

- HMMs are less general than this this, as we don’t allow every $x_t$ to be a function of all of $y$.
HMMs - random functions of a Markov chain

- HMMs are less general than this, as we don’t allow every \( x_t \) to be a function of all of \( y \).
- Suppose we have a set of \( |D_S| \times T \) functions \( f_{i,t} \), and \( T \) independent random variables \( S_t \) then given Markov chain \( Y \), we can define \( X_t = f_{S_t}(Y_t) \). This at last corresponds our picture.
HMMs - random functions of a Markov chain

- HMMs are less general than this, as we don’t allow every $x_t$ to be a function of all of $y$.
- Suppose we have a set of $|D_S|^T$ functions $f_{i,t}$, and $T$ independent random variables $S_t$ then given Markov chain $Y_t$, we can define $X_t = f_{S_t}(Y_t)$. This at last corresponds our picture.

Alternatively, can draw with explicit rep. of the variables $\{S_t : t\}$.

Can think of $x_t$ as a deterministic function of both $y_t$ and $s_t$, and all of the randomness relating $x$ and $y$ is captured by the variables $s$. 
HMMs as Mixture Distributions

- We can also view HMMs as a sequence of mixture distributions.
We can also view HMMs as a sequence of mixture distributions. A mixture distribution is simply \( p(x) = \sum_c p(x, c) = \sum_c p(x|c)p(c) \)

\[ X \quad \overset{\text{left}}{\leftrightarrow} \quad C \]
We can also view HMMs as a sequence of mixture distributions. A mixture distribution is simply
\[ p(x) = \sum_c p(x, c) = \sum_c p(x|c)p(c) \]

Suppose we have a set of random variables \( X_1, \ldots, X_T \) each of which is an individual independent mixture distribution, with mixture variable \( Y_t \). We get:
HMMs as Mixture Distributions

- We can also view HMMs as a sequence of mixture distributions.
- A mixture distribution is simply $p(x) = \sum_c p(x, c) = \sum_c p(x|c)p(c)$

Suppose we have a set of random variables $X_1, \ldots, X_T$ each of which is an individual independent mixture distribution, with mixture variable $Y_t$. We get:

Left: “plate” notation, where the set of random variables is repeated $T$ times. Right: $T = 5$. 
HMMs as Mixture Distributions

- We can also view HMMs as a sequence of mixture distributions.
- A mixture distribution is simply $p(x) = \sum_c p(x, c) = \sum_c p(x|c)p(c)$

\[
\begin{align*}
X & \quad \overset{\text{C}}{\longrightarrow} \\
& 
\end{align*}
\]

- Suppose we have a set of random variables $X_1, \ldots, X_T$ each of which is an individual independent mixture distribution, with mixture variable $Y_t$. We get:

\[
\begin{array}{c}
y_t \\
x_t \\
T
\end{array} 
\quad 
\begin{array}{cccccc}
y_1 & y_2 & y_3 & y_4 & y_5 \\
x_1 & x_2 & x_3 & x_4 & x_5
\end{array}
\]

- Left: “plate” notation, where the set of random variables is repeated $T$ times. Right: $T = 5$.
- So in the collection of mixtures, we have that $Y_i \perp \perp Y_j$ for all $i \neq j$. The mixtures are exchangeable.
HMMs as Mixture Distributions

- We might want to relax the assumption that the $Y_i$'s are independent.
HMMs as Mixture Distributions

- We might want to relax the assumption that the $Y_i$'s are independent.

- One of the simplest ways to add interaction between the mixture variables is to say that the $Y_{1:T}$ variables form a tree, and a simple tree is a chain, leading to an HMM.
We might want to relax the assumption that the $Y_i$’s are independent.

One of the simplest ways to add interaction between the mixture variables is to say that the $Y_{1:T}$ variables form a tree, and a simple tree is a chain, leading to an HMM.

However, in such case we still have a mixture distribution in that

$$p(x_{1:T}) = \sum_{y_{1:T}} p(x_{1:T}, y_{1:T})$$ (2.32)

where there is naively an exponential number of mixture components.
HMM: As generative process

- Classic urn/ball example. We have $N$ urns, each urn has a set of different colored balls in them.
- We choose urns according to a 1st-order Markov chain. Each time we choose an urn, we sample (with replacement) a ball from that urn.
- The output is the sequence of ball colors, but not the sequence of urn choices.
- A goal might be, based on the sequence of ball colors, and knowing the corresponding distributions, deduce the most probable sequence of urns that could possibly have generated that ball sequence (this is the recognition problem).
HMM: As generative process urns/balls

Prof. Jeff Bilmes
EE596A/Winter 2013/DGMs – Lecture 2 - Jan 9th, 2013
HMM: As generative process urns/balls (cont. II)

**Figure:** A 4-state HMM viewed as balls in urns. Each HMM state corresponds to an urn and the directed edges between urns give the state transition matrix, where missing edges correspond to zeros in the matrix. In each state, there is a distribution on colored balls. For example, in state one, we have
\[
\begin{align*}
\Pr(\text{color} = \text{yellow}|\text{state} = 1) &= \frac{4}{7} \\
\Pr(\text{color} = \text{red}|\text{state} = 1) &= \frac{2}{7} \\
\Pr(\text{color} = \text{green}|\text{state} = 1) &= \frac{1}{7} \\
\Pr(\text{color} = \text{blue}|\text{state} = 1) &= 0
\end{align*}
\]
That is, in each state (urn), the probability of drawing a color is equal to the number of balls of that color in that urn divided by the total number of balls.
Definition 2.4.1

Hidden Markov Model  A hidden Markov model (HMM) is a collection of random variables consisting of a set of $T$ discrete scalar variables $Q_{1:T}$ and a set of $T$ other variables $X_{1:T}$ which may be either discrete or continuous (and either scalar- or vector-valued). These variables, collectively, possess the following conditional independence properties:

\[
\{Q_{t:T}, X_{t:T}\} \perp \perp \{Q_{1:t-2}, X_{1:t-1}\} | Q_{t-1}
\]

(2.33)

and

\[
X_t \perp \perp \{Q_{t-r}, X_{\neg t}\} | Q_t
\]

(2.34)

for each $t \in 1 : T$. No other conditional independence properties are true in general, unless they follow from Equations 2.33 and 2.34. The length $T$ of these sequences is itself an integer-valued random variable having a complex distribution.
HMM: As a set of independence properties

- $Q_t$ takes values from finite set, so $Q_t \in D_Q$ where $D_Q$ is called the state space, cardinality $|D_Q|$.

- Equation 2.33 states that the future is conditionally independent of the past given the present. Therefore, $Q_t \perp \perp Q_{1:t-2} | Q_{t-1}$ which means the variables $Q_{1:T}$ form a discrete-time, discrete-valued, first-order Markov chain.

- Also, $Q_t \perp \perp \{Q_{1:t-2}, X_{1:t-1}\} | Q_{t-1}$ so that $X_{\tau}$ is unable, given $Q_{t-1}$, to affect $Q_t$ for $\tau < t$. Does **not** imply, given $Q_{t-1}$, that $Q_t$ is unaffected by future variables.
HMM: As a set of independence properties

What the definition doesn’t require:

- does not limit the number of states $|D_Q|$ in the Markov chain, only that it is finite.
HMM: As a set of independence properties

What the definition doesn’t require:

- does not limit the number of states \(|D_Q|\) in the Markov chain, only that it is finite.

- Does not require the observations \(X_{1:T}\) to be either discrete, continuous, scalar-, or vector-valued, does not designate the implementation of the dependencies (e.g., general regression, probability table, neural network, etc.),
HMM: As a set of independence properties

What the definition doesn’t require:

- does not limit the number of states $|D_Q|$ in the Markov chain, only that it is finite.
- Does not require the observations $X_{1:T}$ to be either discrete, continuous, scalar-, or vector-valued, does not designate the implementation of the dependencies (e.g., general regression, probability table, neural network, etc.),
- does not determine the model families for each of the variables (e.g., Gaussian, Laplace, etc.),
HMM: As a set of independence properties

What the definition doesn’t require:

- does not limit the number of states $|D_Q|$ in the Markov chain, only that it is finite.
- Does not require the observations $X_{1:T}$ to be either discrete, continuous, scalar-, or vector- valued, does not designate the implementation of the dependencies (e.g., general regression, probability table, neural network, etc.),
- does not determine the model families for each of the variables (e.g., Gaussian, Laplace, etc.),
- does not force the underlying Markov chain to be time-homogeneous,
HMM: As a set of independence properties

What the definition doesn’t require:

- does not limit the number of states $|D_Q|$ in the Markov chain, only that it is finite.
- Does not require the observations $X_{1:T}$ to be either discrete, continuous, scalar-, or vector-valued, does not designate the implementation of the dependencies (e.g., general regression, probability table, neural network, etc.),
- does not determine the model families for each of the variables (e.g., Gaussian, Laplace, etc.),
- does not force the underlying Markov chain to be time-homogeneous,
- does not fix the parameters or any tying mechanism.
HMM: As a set of independence properties

What the definition doesn’t require:

- does not limit the number of states $|D_Q|$ in the Markov chain, only that it is finite.
- Does not require the observations $X_{1:T}$ to be either discrete, continuous, scalar-, or vector-valued, does not designate the implementation of the dependencies (e.g., general regression, probability table, neural network, etc.),
- does not determine the model families for each of the variables (e.g., Gaussian, Laplace, etc.),
- does not force the underlying Markov chain to be time-homogeneous,
- does not fix the parameters or any tying mechanism
- HMM makes no marginal independence assumptions — nothing in an HMM is independent of anything else (no statements of form $A \perp \perp B$). only conditional independence statements exist.
Using the conditional independence statements mentioned above, we can derive the following factorization:

\[ p(x_{1:T}, q_{1:T}) = p(x_T, q_T | x_{1:T-1}, q_{1:T-1}) p(x_{1:T-1}, q_{1:T-1}) \]  
\[ = p(x_T | q_T, x_{1:T-1}, q_{1:T-1}) p(q_T | x_{1:T-1}, q_{1:T-1}) \]  
\[ p(x_{1:T-1}, q_{1:T-1}) \]  
\[ = p(x_T | q_T) p(q_T | q_{T-1}) p(x_{1:T-1}, q_{1:T-1}) \]  
\[ = \ldots \]  
\[ = p(q_1) \prod_{t=2}^{T} p(q_t | q_{t-1}) \prod_{t=1}^{T} p(x_t | q_t) \]  

This last equation is the classical factorization expression for an HMM joint distribution over \( x_{1:T}, q_{1:T} \).
HMMs

- HMMs are a sequential model simultaneously over both sequences $X_{1:T}$ and $Q_{1:T}$, i.e., $p(x_{1:T}, q_{1:T})$.
- HMMs represents a joint distribution, i.e., $p(x_{1:T}, q_{1:T})$, not a conditional distribution.
- Just like any graphical model, that represents a joint distribution over a set of random variables.
HMM parameters

- Parameters of HMM, depend on nature of underlying Markov chain.
- If time-homogeneous, we have an initial state distribution (typically $\pi$) with $p(Q_1 = i) = \pi_i$, and a state transition matrix $A$.
- We also have the set of observation distributions $b_j(x) = p(X_t = x|Q_t = j)$ in the time-homogeneous case. In time homogeneous case, we might have $b_{t,j}(x)$.
- Sampling from an HMM means: 1) first randomly choose an assignment to $Q_{1:T}$ and then 2) randomly choose an assignment to $X_{1:T}$.
- Each new $X$ sample requires a new $Q$ sample.
An HMM is a tree (in the GM sense) and so we know that computing inference (over cliques) has cost $O(r^2)$. Since there are $T$ such cliques, overall cost should be $O(Tr^2)$.

Also, we view $X_{1:T}$ as the stochastic process with an underlying generative hidden chain $Q_{1:T}$. Thus, we might want to compute

$$p(\bar{x}_{1:T}) = \sum_{q_{1:T}} p(\bar{x}_{1:T}, q_{1:T})$$

again, naively, an exponential computation.

This is the probability of evidence computation, and again due to the query and the fact that it is a tree, it is relatively easy, at least from the perspective of clique size (there are many difficulties when $r$ gets extremely large, as we’ll see).
To compute $p(x_{1:T})$, we start out with $p(x_{1:t})$

\[
p(x_{1:t}, q_t, q_{t-1}) = p(x_{1:t-1}, q_{t-1}, x_t, q_t)
\]

\[(A) \quad \Rightarrow p(x_t, q_t | x_{1:t-1}, q_{t-1})p(x_{1:t-1}, q_{t-1})
\]

\[
= p(x_t | q_t, x_{1:t-1}, q_{t-1})p(q_t | x_{1:t-1}, q_{t-1})p(x_{1:t-1}, q_{t-1})
\]

\[(B) \quad \Rightarrow p(x_t | q_t)p(q_t | q_{t-1})p(x_{1:t-1}, q_{t-1})
\]

where (A) follows from the chain rule of probability, and (B) follows since $X_t \perp \perp \{X_{1:t-1}, Q_{1:t-1}\} | Q_t$ and $Q_t \perp \perp \{X_{1:t-1}, Q_{1:t-2}\} | Q_{t-1}$
HMM Forward Recursion

This yields,

\[ p(x_{1:t}, q_t) = \sum_{q_{t-1}} p(x_{1:t}, q_t, q_{t-1}) \]  \hspace{1cm} (2.42)

\[ = \sum_{q_{t-1}} p(x_t | q_t) p(q_t | q_{t-1}) p(x_{1:t-1}, q_{t-1}) \]  \hspace{1cm} (2.43)
HMM Forward Recursion

- This yields,

\[
p(x_{1:t}, q_t) = \sum_{q_{t-1}} p(x_{1:t}, q_t, q_{t-1})
\]

\[
= \sum_{q_{t-1}} p(x_t | q_t) p(q_t | q_{t-1}) p(x_{1:t-1}, q_{t-1})
\]

(2.42)

(2.43)

- If the following quantity is defined \( \alpha_q(t) \triangleq p(x_{1:t}, Q_t = q) \), then the preceding equations imply that

\[
\alpha_q(t) = p(x_t | Q_t = q) \sum_r p(Q_t = q | Q_{t-1} = r) \alpha_r(t - 1)
\]

(2.44)
HMM Forward Recursion

- This yields,

\[ p(x_{1:t}, q_t) = \sum_{q_{t-1}} p(x_{1:t}, q_t, q_{t-1}) \]  

\[ = \sum_{q_{t-1}} p(x_t | q_t) p(q_t | q_{t-1}) p(x_{1:t-1}, q_{t-1}) \]  

(2.42)

(2.43)

- If the following quantity is defined \( \alpha_q(t) \overset{\Delta}{=} p(x_{1:t}, Q_t = q) \), then the preceding equations imply that

\[ \alpha_q(t) = p(x_t | Q_t = q) \sum_r p(Q_t = q | Q_{t-1} = r) \alpha_r(t - 1) \]  

(2.44)

- This is called the \( \alpha \) (or forward) recursion for HMMs. It is also identical to message passing from left-to-right in the LBP algorithm.
Thus, \( p(x_{1:T}) = \sum_q \alpha_q(T) \), and the entire computation requires only \( O(|D_Q|^2 T) \) operations.
Thus, $p(x_{1:T}) = \sum_q \alpha_q(T)$, and the entire computation requires only $O(|D_Q|^2 T)$ operations.

To derive this recursion, it was necessary to use only the fact that $X_t$ was independent of its past given $Q_t$ (call this Assumption I) — in an HMM, $X_t$ is also independent of the future given $Q_t$, but this was not yet used (call this Assumption II).
Thus, \( p(x_{1:T}) = \sum_q \alpha_q(T) \), and the entire computation requires only \( O(|DQ|^2T) \) operations.

To derive this recursion, it was necessary to use only the fact that \( X_t \) was independent of its past given \( Q_t \) (call this Assumption I) — in an HMM, \( X_t \) is also independent of the future given \( Q_t \), but this was not yet used (call this Assumption II).

**Exercise:** Are there models where \( X_t \) is independent of its past but not its future given \( Q_t \) where such a forward recursion can be defined?
HMM backwards recursion

- This later assumption (Assumption II), however, is obligatory for the beta or backward recursion in HMMs as we will now see.

\[
p(x_{t+1}, T | q_t) = \sum_{q_{t+1}} p(x_{t+1}, x_{t+2:T} | q_t)
\]

\[
(A) \implies \sum_{q_{t+1}} p(x_{t+2:T} | q_{t+1}, x_{t+1}, q_t) p(x_{t+1} | q_{t+1}, q_t) p(q_{t+1} | q_t)
\]

\[
(B) \implies \sum_{q_{t+1}} p(x_{t+2:T} | q_{t+1}) p(x_{t+1} | q_{t+1}) p(q_{t+1} | q_t)
\]

where (A) follows from the chain rule probability, and (B) follows since \( X_{t+2:T} \perp \perp \{X_{t+1}, Q_t\} | Q_{t+1} \) and \( X_{t+1} \perp \perp Q_t | Q_{t+1} \)
HMM backwards recursion

- Using the definition $\beta_q(t) \triangleq p(x_{t+1:T} | Q_t = q)$, the above equations imply the beta-recursion

$$\beta_q(t) = \sum_r \beta_r(t+1) p(x_{t+1} | Q_{t+1} = r) p(Q_{t+1} = r | Q_t = q) \quad (2.45)$$

, and another expression for the full probability

$$p(x_{1:T}) = \sum_q \beta_q(1) p(q) p(x_1 | q). \quad (2.46)$$
Recall, goal is to compute $p(x_{1:T})$, the probability of “evidence”, efficiently.
Recall, goal is to compute $p(x_{1:T})$, the probability of “evidence”, efficiently.

This probability may be computed using a combination of the alpha and beta values at any $t$ since

$$p(x_{1:T}) = \sum_{q_t} p(q_t, x_{1:t}, x_{t+1:T}) = \sum_{q_t} p(x_{t+1:T}|q_t, x_{1:t}) p(q_t, x_{1:t})$$

$$\quad \equiv \sum_{q_t} p(x_{t+1:T}|q_t) p(q_t, x_{1:t}) = \sum_{q_t} \beta_{q_t}(t) \alpha_{q_t}(t)$$

where (A) follows since $X_{t+1:T} \perp \perp X_{1:t}|Q_t$ in an HMM.
For any $p \in \mathcal{F}(\text{HMM}, R)$, from the directed local Markov property, we can immediately write down the joint as

\[
p(x_{1:T}, q_{1:T}) = p(q_1)p(x_1|q_1) \prod_{t=2}^{T} p(x_t|q_t)p(q_t|q_{t-1})
\]

(2.47)

\[
= \prod_{t} p(x_t|q_t)p(q_t|q_{t-1})
\]

(2.48)
HMM Queries
Queries associated with HMMs

The quantities we typically wish to compute for an HMM include:

- Compute $p(q_t|x_{1:t})$, or the **filtering** problem.
- Compute $p(q_t|x_{1:s})$, with $t > s$, or the **prediction** problem.
- Compute $p(q_t|x_{1:u})$, with $t < u$, or the **smoothing** problem.

Above three named from linear systems literature in EE (e.g., Kalman filters).

- Note: above includes $p(q_t|x_{1:T})$.
- Also needed query is $p(q_t, q_{t+1}|x_r:s)$ (often $r = 1$ and $s = T$).
- In all above cases, we need to sum out hidden variables from joint distributions. E.g., $p(q_t|x_{1:T}) = p(q_t, x_{1:T})/p(x_{1:T})$, so also need $p(x_{1:T})$. I.e., we compute both the numerator and denominator in each of the above queries. How does this relate to clique potentials in the graph?
Recall parameter names, time-homogeneous case.

1. \( P(Q_t = i | Q_{t-1} = j) = a_{ij} \) or \([A]_{ij}\) is a first-order time-homogeneous transition matrix.

2. \( P(Q_1 = i) = \pi_i \) is the initial state distribution.

3. \( P(X_t = x | Q_t = i) = b_i(x) \) is the observation distribution for the current state being in configuration \( i \).

Notice that there are a fixed number of parameters regardless of the length \( T \). In other words, parameters are shared across all time. This is a property of all dynamic graphical models.

What probabilistic queries would we need to learn these parameters?
To decide which queries to compute, should know which ones we want. If learning HMM parameters with EM, what queries do we need?

$X_{1:T} = \bar{x}_{1:T}$ observed, $Q_{1:T}$ hidden variables, $\lambda$ are parameters to learn, and $\lambda^p$ are the previous iteration parameters. EM then repeatedly optimizes the following objective:
HMM - learning with EM (cont. II)

\[ f(\lambda) = Q(\lambda, \lambda^p) = E_p(x_{1:T}, q_{1:T}|\lambda^p)[\log p(x_{1:T}, q_{1:T}|\lambda)] \]  
\[ = E_p[\log \prod_t p(q_t|q_{t-1}, \lambda)p(x_t|q_t, \lambda)] \]  
\[ = E_p[\sum_t \log p(q_t|q_{t-1}, \lambda) + \sum_t \log p(x_t|q_t, \lambda)] \]  
\[ = \sum_t \sum_{ij} p(Q_t = j, Q_{t-1} = i|x_{1:T}, \lambda^p) \log p(Q_t = j|Q_{t-1} = i, \lambda) \]  
\[ + \sum_t \sum_i p(Q_t = i|x_{1:T}, \lambda^p) \log p(x_t|Q_t = i, \lambda) \]  

So this means that for EM learning, we need for all \( t \), the queries \( p(Q_t = i|x_{1:T}) \) and \( p(Q_t = j, Q_{t-1} = i|x_{1:T}) \) in an HMM. Note again that these are clique posteriors.
EM isn’t the only way to learn parameters.

Suppose we wanted to use a gradient descent like algorithm on
\[ f(\lambda) = \log p(x_{1:T} | \lambda), \]
as in

\[
\frac{\partial}{\partial \lambda} f(\lambda) = \frac{\partial}{\partial \lambda} \log p(x_{1:T} | \lambda) = \frac{\partial}{\partial \lambda} \log \sum_{q_{1:T}} p(x_{1:T}, q_{1:T} | \lambda) \quad (2.53)
\]

\[
= \frac{\partial}{\partial \lambda} \sum_{q_{1:T}} p(x_{1:T}, q_{1:T} | \lambda) = \frac{\partial}{\partial \lambda} \sum_{q_{1:T}} p(x_{1:T}, q_{1:T} | \lambda) \frac{p(x_{1:T} | \lambda)}{\sum_{q_{1:T}} p(x_{1:T}, q_{1:T} | \lambda)} \quad (2.54)
\]
HMM - learning with gradient descent (cont. II)

Say we’re interested in $\partial/\partial a_{ij}$. Let’s expand the numerator above:

$$\text{numerator} = \frac{\partial}{\partial a_{ij}} \sum_{q_1:T} p(x_{1:T}, q_{1:T} | \lambda) = \frac{\partial}{\partial a_{ij}} \sum_{q_1:T} \prod_{t} p(x_t | q_t) p(q_t | q_{t-1})$$

(2.55)

Define $\mathcal{T}_{ij}(q_{1:T}) \triangleq \{ t : q_{t-1} = i, q_t = j \}$ in the following:

$$\text{numerator} = \frac{\partial}{\partial a_{ij}} \sum_{q_1:T} \prod_{t \in \mathcal{T}_{ij}(q_{1:T})} p(x_t | q_t) \prod_{t \in \mathcal{T}_{ij}(q_{1:T})} a_{ij} \prod_{t \notin \mathcal{T}_{ij}(q_{1:T})} p(q_t | q_{t-1})$$

(2.56)
HMM - learning with gradient descent

We get

\[
\text{num} = \sum_{q_1:T} \prod_t p(x_t | q_t) \frac{\partial}{\partial a_{ij}} a_{ij}^{|T_{ij}(q_1:T)|} \prod_{t \notin T_{ij}(q_1:T)} p(q_t | q_{t-1})
\]

\[
= \sum_{q_1:T} \prod_t p(x_t | q_t) T_{ij}(q_1:T) a_{ij}^{T_{ij}(q_1:T)-1} \prod_{t \notin T_{ij}(q_1:T)} p(q_t | q_{t-1})
\]

\[
= \sum_{q_1:T} \prod_t p(x_t | q_t) p(q_t | q_{t-1}) \frac{T_{ij}(q_1:T)}{a_{ij}} = \sum_{q_1:T} p(x_1:T, q_1:T) \frac{T_{ij}(q_1:T)}{a_{ij}}
\]

\[
= \frac{1}{a_{ij}} \sum_{q_1:T} p(x_1:T, q_1:T) \sum_t 1\{q_{t-1} = i, q_t = j\}
\]

\[
= \frac{1}{a_{ij}} \sum_t \sum_{q_1:T} p(x_1:T, q_1:T) 1\{q_{t-1} = i, q_t = j\}
\]

\[
= \frac{1}{a_{ij}} \sum_t p(x_1:T, q_t-1 = i, q_t = j)
\]
HMM - learning with gradient descent

\[
\frac{\partial}{\partial \lambda} f(\lambda) = \frac{\partial}{\partial \lambda} \sum_{q_1:T} p(x_{1:T}, q_{1:T}|\lambda) \frac{p(x_{1:T}|\lambda)}{p(x_{1:T}|\lambda)} = \frac{1}{a_{ij}} \sum_t p(x_{1:T}, q_{t-1} = i, q_t = j) p(x_{1:T}|\lambda)
\]

(2.57)

\[
= \frac{1}{a_{ij}} \sum_t p(q_{t-1} = i, q_t = j|x_{1:T})
\]

(2.58)

This means that, like in EM, for gradient descent learning, we also need for all \(t\) the queries \(p(Q_t = j, Q_{t-1} = i|x_{1:T})\) from the HMM. A similar analysis shows that we also need \(\forall t p(Q_t = i|x_{1:T})\). These are also needed when performing discriminative training. So clique posteriors are fundamental, we must have a procedure that produces them quickly.
HMMs and elimination - forward recursion

- HMM forward recursion is just the elimination algorithm on the graph

- Choose elimination order:
  \[ X_1, X_2, Q_1, X_3, Q_2, X_4, Q_3, X_5, \ldots, X_T, Q_{T-1}, Q_T. \]

- Evidence from delta functions, i.e., \( x_1 = \bar{x}_1 \Rightarrow \delta(x_1, \bar{x}_1), \)
  \( x_2 = \bar{x}_2 \Rightarrow \delta(x_2, \bar{x}_2), \) and so on.

- We get:

\[
\cdots \sum_{x_4} \sum_{q_2} \sum_{x_3} \sum_{q_1} \sum_{x_2} \sum_{x_1} \prod_{t=1}^{T} p(x_t|q_t)p(q_t|q_{t-1})\delta(x_t, \bar{x}_t)
\]

\[
= \cdots \sum_{x_3} \sum_{q_1} \sum_{x_2} \left( \prod_{t=2}^{T} p(x_t|q_t)p(q_t|q_{t-1})\delta(x_t, \bar{x}_t) \right) \sum_{x_1} p(x_1|q_1)\delta(x_1, \bar{x}_1)p(q_1)
\]

\[
= p(\bar{x}_1|q_1)p(q_1) \Delta \alpha_1(q_1)
\]
HMMs and elimination - forward recursion

\[ \ldots \sum_{x_3} \sum_{q_1} \left( \prod_{t=3}^{T} p(x_t|q_t)p(q_t|q_{t-1})\delta(x_t, \bar{x}_t) \right) \sum_{x_2} p(x_2|q_2)p(q_2|q_1)\delta(x_2, \bar{x}_2) \alpha(q_1) \]

\[ = \ldots \sum_{x_4} \sum_{q_2} \sum_{x_3} \left( \prod_{t=3}^{T} p(x_t|q_t)p(q_t|q_{t-1})\delta(x_t, \bar{x}_t) \right) \sum_{q_1} p(q_2|q_1)p(\bar{x}_2|q_2) \alpha(q_1) \]

\[ = \ldots \sum_{x_{r+2}} \sum_{q_r} \sum_{x_{r+1}} \left( \prod_{t=r+1}^{T} p(x_t|q_t)p(q_t|q_{t-1})\delta(x_t, \bar{x}_t) \right) \sum_{q_{r-1}} p(q_r|q_{r-1})p(\bar{x}_r|q_r) \alpha(q_{r-1}) \]

\[ \alpha(q_r) \]

\[ \bullet \alpha\text{-recursion becomes} \]

\[ \alpha_{t+1}(j) = \sum_i \alpha_t(i)p(Q_{t+1} = j|Q_t = i)p(x_{t+1}|Q_{t+1} = j) \quad (2.59) \]

\[ \alpha_1(j) = p(Q_1 = j)p(\bar{x}_1|Q_1 = j) \quad (2.60) \]
HMMs and elimination - forward recursion

- From the last line of the elimination (when we sum out \( q_{r-1} \), we see \( \alpha \)-recursion, which is:

\[
\alpha_{t+1}(j) = \sum_{i} \alpha_{t}(i) p(Q_{t+1} = j|Q_{t} = i) p(x_{t+1}|Q_{t+1} = j) \quad (2.61)
\]

and

\[
\alpha_{1}(j) = p(Q_{1} = j)p(\bar{x}_{1}|Q_{1} = j) \quad (2.62)
\]

- We have that

\[
\alpha_{1}(Q_{1} = j) = p(Q_{1} = j)p(\bar{x}_{1}|Q_{1} = j) = p(\bar{x}_{1}, Q_{1} = j), \text{ and}
\]

\[
\alpha_{1}(q_{1}) = p(\bar{x}_{1}, q_{1}).
\]

- Also, \( \alpha_{2}(q_{2}) = \sum_{q_{1}} p(q_{2}|q_{1})p(\bar{x}_{2}|q_{2})\alpha(q_{1}) = \sum_{q_{1}} p(q_{2}|q_{1})p(\bar{x}_{2}|q_{2})p(\bar{x}_{1}, q_{1}) = \sum_{q_{1}} p(q_{1}, q_{2}, \bar{x}_{1}, \bar{x}_{2}) = p(\bar{x}_{1}, \bar{x}_{2}, q_{2})
\]

In general, the forward (\( \alpha \)) recursion has meaning

\[
\alpha_{t}(j) = p(x_{1:t}, Q_{t} = j)
\]

- So \( \alpha \) (forward) recursion is just an instance of the elimination algorithm run on the GM for the HMM graph.
HMMs and elimination - backward recursion

Next, consider elimination order $X_T, Q_T, X_{T-1}, Q_{T-1}, \ldots$

\[
\cdots \sum_{x_{T-2}} \sum_{q_{T-1}} \sum_{x_{T-1}} \sum_{q_T} \sum_{x_T} \prod_{t=1}^{T} p(x_t|q_t) p(q_t|q_{t-1}) \delta(x_t, \bar{x}_t) \\
= \cdots \sum_{x_{T-2}} \sum_{q_T} \left( \prod_{t=1}^{T-1} p(x_t|q_t) p(q_t|q_{t-1}) \delta(x_t, \bar{x}_t) \right) \sum_{x_T} p(x_T|q_T) p(q_T|q_{T-1}) \delta(x_T, \bar{x}_T) \\
= \cdots \sum_{x_{T-2}} \sum_{q_T} \left( \prod_{t=1}^{T-1} p(x_t|q_t) p(q_t|q_{t-1}) \delta(x_t, \bar{x}_t) \right) p(\bar{x}_T|q_T) p(q_T|q_{T-1})
\]
HMMs and elimination - backward recursion

\[ \sum_{x_{T-2}} \sum_{q_T} \left( \prod_{t=1}^{T-1} p(x_t | q_t) p(q_t | q_{t-1}) \delta(x_t, \bar{x}_t) \right) p(\bar{x}_T | q_T) p(q_T | q_{T-1}) \frac{1}{\beta_T(q_T)} \]

\[ = \sum_{q_{T-1}} \sum_{x_{T-2}} \sum_{q_T} \left( \prod_{t=1}^{T-1} p(x_t | q_t) p(q_t | q_{t-1}) \delta(x_t, \bar{x}_t) \right) \sum_{q_T} p(\bar{x}_T | q_T) p(q_T | q_{T-1}) \beta_T(q_T) \]

This corresponds to the \( \beta \) recursion

\[ \beta_t(q_t) = \sum_{q_{t+1}} \beta_{t+1}(q_{t+1}) p(q_{t+1} | q_t) p(x_{t+1} | q_{t+1}) \quad (2.63) \]

\[ \beta_T(q_T) = 1 \quad (2.64) \]
HMMs and elimination - backward recursion, meaning

\[ \beta_T(j) = 1 \]  \hspace{1cm} (2.65)

\[ \beta_{T-1}(i) = \sum_j p(\bar{x}_T | Q_T = j)p(Q_T = j | Q_{T-1} = i)\beta_T(j) \]  \hspace{1cm} (2.66)

\[ \sum_j p(\bar{x}_T, Q_T = j | Q_{T-1} = i) = p(\bar{x}_T | Q_{T-1} = i) \]  \hspace{1cm} (2.67)

In general,

\[ \beta_t(i) = p(x_{t+1:T} | Q_t = i) \]  \hspace{1cm} (2.68)
HMMs and elimination - \( p(x_{1:t}) \)

We can get \( p(x_{1:t}) \) in many different ways:

\[
p(x_{1:T}) = \sum_{q_t} p(x_{1:T}, q_t) \\
= \sum_{q_t} p(x_{1:t}, x_{t+1:T}, q_t) \\
= \sum_{q_t} p(x_{t+1:T} | q_t, x_{1:t}) p(q_t, x_{1:t}) \\
= \sum_{q_t} p(x_{t+1:T} | q_t) p(q_t, x_{1:t}) \\
= \sum_{q_t} \beta_t(q_t) \alpha_t(q_t)
\]

So this means that for any \( t \), we can get \( p(x_{1:T}) \) by using the corresponding \( \alpha_t(i) \) and \( \beta_t(i) \) quantities, for any \( t \). Which one to choose?
HMMs and elimination - $p(x_{1:t})$

We can get $p(x_{1:t})$ in many different ways:

$$p(x_{1:T}) = \sum_{q_t} p(x_{1:T}, q_t)$$

$$= \sum_{q_t} p(x_{1:t}, x_{t+1:T}, q_t)$$

$$= \sum_{q_t} p(x_{t+1:T} | q_t, x_{1:t}) p(q_t, x_{1:t})$$

$$= \sum_{q_t} p(x_{t+1:T} | q_t) p(q_t, x_{1:t})$$

$$= \sum_{q_t} \beta_t(q_t) \alpha_t(q_t)$$

So this means that for any $t$, we can get $p(x_{1:T})$ by using the corresponding $\alpha_t(i)$ and $\beta_t(i)$ quantities, for any $t$. Which one to choose? primarily finite-precision arithmetic (i.e., numerical) reasons for choosing one or another.
HMMs and elimination

So we have the forward recursion as an elimination order and the backwards recursion as an elimination order.

Green order is $\alpha$-recursion, and blue order is $\beta$-recursion.

Since HMM is a tree, there are no additional fill-in edges via the elimination orders we have chosen.

If we just eliminated the hidden variables, we’d get:
HMMs and elimination

- Another elimination order will produce a different recursion for HMMs. For example, an *inside-outside* algorithm of sorts can be defined (normally this is used for computing marginals for stochastic context-free grammars) via the following elimination order.

![Diagram of elimination order](image)

- Challenge Problem/HW: Derive the equations and a recursion for this elimination order. How is it different than standard forward/backward?
But as mentioned before, we want more than just \( p(x_{1:T}) \).

We need clique posteriors \( \gamma_t(i) = p(Q_t = i | x) \) and 
\[ \xi_t(i, j) = p(Q_{t-1} = i, Q_t = j | x). \]

We can also get these from \( \alpha \) and \( \beta \).

\[
\gamma(q_t) = p(q_t | x_{1:T}) = \frac{p(x_{1:T} | q_t)p(q_t)}{p(x_{1:T})} \\
= p(x_{1:t}, x_{t+1:T} | q_t)p(q_t)/p(x_{1:T}) \\
= p(x_{t+1:T} | q_t, x_{1:t})p(x_{1:t} | q_t)p(q_t)/p(x_{1:T}) \\
= p(x_{1:t}, q_t)p(x_{t+1:T} | q_t)/p(x) \\
= \alpha(q_t)\beta(q_t)/p(x) \\
= \alpha(q_t)\beta(q_t) / \sum_{q_t'} \alpha(q_t')\beta(q_t')
\]
HMM, posteriors

How best to compute $\gamma_t(i)$ for all $t$?

- suboptimal way:

  1. for $t = 1 \ldots T$ do
  2. Compute $\alpha_t(j)$ starting at time 1 up to time $t$
  3. Compute $\beta_t(j)$ starting at time $T$ down to time 1
  4. Compute $\gamma_t(j)$
  5. Update parameters based on current learning procedure.

- But this is extremely wasteful. Once we have computed $\alpha_t(j)$, for time $t$, should hold on to it for the next time step, $\alpha_{t+1}(j)$. Similarly, once we have $\beta_t(j)$ save it for previous time $\beta_{t-1}(j)$.

- Dynamic programming: optimal substructure & common subproblems of $\alpha_t(j)$, exactly the complete computation of $\alpha_{t-1}(j)$.

- This is obvious when viewed as messages in the GM (messages don’t proceed until they have received appropriate incoming messages).
Generic message definition for arbitrary $p \in \mathcal{F}(G, R)$

$$
\mu_{i \rightarrow j}(x_j) = \sum_{x_i} \psi_{i,j}(x_i, x_j) \prod_{k \in \delta(i) \setminus \{j\}} \mu_{k \rightarrow i}(x_i)
$$

(2.69)

- If graph is a tree, and if we obey message-passing protocol order, then we will reach a point where we’ve got marginals. I.e.,

$$
p(x_i) \propto \prod_{j \in \delta(i)} \mu_{j \rightarrow i}(x_i)
$$

(2.70)

and

$$
p(x_i, x_j) \propto \psi_{i,j}(x_i, x_j) \prod_{k \in \delta(i) \setminus \{j\}} \mu_{k \rightarrow i}(x_i) \prod_{\ell \in \delta(j) \setminus \{i\}} \mu_{\ell \rightarrow j}(x_j)
$$

(2.71)
LBP messages on an HMM are straightforward, and again since an HMM is a tree, we can use any message order that obeys the message-passing protocol.
HMM, and junction tree

- Can view this using junction tree algorithm with cliques and separators. One solution has each node a separator, & each edge a 2-clique, but there are others.
HMM, and junction tree

- Can view this using junction tree algorithm with cliques and separators. One solution has each node a separator, & each edge a 2-clique, but there are others.
Using either of these junction trees, we can define a forward and/or backwards recursion, and we’ll again, in some cases, either recover exactly the $\alpha, \beta$ recursions, or computations that are quite similar.

For example, Hugin message passing computes the backward messages based on the forward (if we go right and then left), or computes the forward messages based on the backwards (if we go left and then right).
It is possible to derive a temporal recursion for quantities other than $\alpha$ and $\beta$. E.g., here is a $\gamma_t(j) = p(Q_t = j| x_{1:T})$ backwards recursion.

$$\gamma(q_t) = \sum_{q_{t+1}} p(q_t, q_{t+1}| x_{1:T}) = \sum_{q_{t+1}} p(q_t| q_{t+1}, x_{1:T}) p(q_{t+1}| x_{1:T})$$

$$= \sum_{q_{t+1}} p(q_t| q_{t+1}, x_{1:T}) \gamma(q_{t+1}) = \sum_{q_{t+1}} p(q_t| q_{t+1}, x_{1:t}) \gamma(q_{t+1})$$

$$= \sum_{q_{t+1}} \frac{p(q_t, q_{t+1}, x_{1:t})}{\sum_{q_t} p(q_t, q_{t+1}, x_{1:t})} \gamma(q_{t+1})$$

$$= \sum_{q_{t+1}} \frac{p(q_{t+1}| q_t) p(q_t, x_{1:t})}{\sum_{q_t} p(q_{t+1}| q_t) p(q_t, x_{1:t})} \gamma(q_{t+1})$$

$$= \sum_{q_{t+1}} \frac{p(q_{t+1}| q_t) \alpha_t(q_t)}{\sum_{q_t} p(q_{t+1}| q_t) \alpha_t(q_t)} \gamma(q_{t+1})$$

Therefore, there is a backwards pass recursion using just the $\alpha$’s without directly touching the observations again (better memory).
HMMs are more powerful than you might think. We’ll see that many DGMs can be represented by HMMs, so before we move on to DGMs, we should understand how flexible HMMs are (and then as we go through course, we’ll see what the penalties are for making such HMM representations).

We next visit a set of “facts” about HMMs that should be remembered.
Joint probability under an HMM:

\[ p(X_{t:t+h} = x_{t:t+h}) = \sum_{q_{t:t+h}} \prod_{j=t}^{t+h} p(X_j = x_j | Q_j = q_j) a_{q_j-1 q_j}. \]

Unless only one state is possible, observations do not factorize.

In an HMM, there are no statements of the form \( A \perp \perp B \).

HMMs have conditional independence properties (like all DGMs have).

\[ p(X_{t:t+h} = x_{t:t+h} | Q_{t:t+h} = q_{t:t+h}) = \prod_{\tau=t}^{t+h} p(X_{\tau} = x_{\tau} | Q_{\tau} = q_{\tau}). \]
Observations are not “Viterbi i.i.d.”

- The Viterbi path (most-probable explanation) of an HMM is defined as follows:

\[ q^*_1:T = \text{argmax}_{q_1:T} p(X_1:T = x_1:T, q_1:T) \]

- We may wish to max-marginalize the hidden states. Does that lead to i.i.d.?

- The “Viterbi” distribution of the HMM is:

\[
p_{\text{Vit}}(X_1:T = x_1:T) = c \prod_{t=1}^{T} p(X_t = x_t|Q_t = q_t)p(Q_t = q_t|Q_{t-1} = q_{t-1})
\]

where \(c\) is a positive normalizing constant.
Sources for Today’s Lecture

- “doc.pdf” sections 8.1 - 8.3