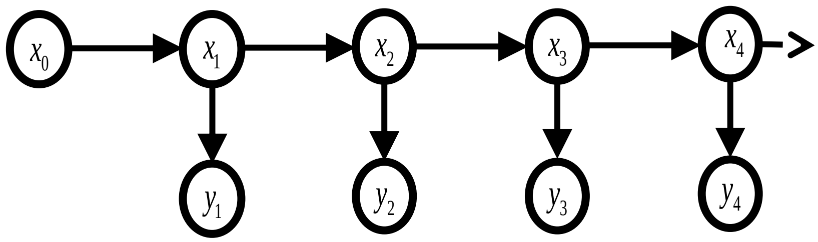
Lecture 13: HMMs



Transition model: $P(x_{t+1} | x_t)$

Observation (emission) model: $P(y_t | x_t)$ *for discrete* y_t; $p(y_t | x_t)$ *for continuous* y_t

Initial state distribution: $P(x_0 = i) = \pi_i$

Doubly embedded stochastic process: hidden process, x_t; uncertain observation process, y_t

$$P(y_{(1:T)}, x_{1:T}) = p(x_0)\prod_{t=1}^{T} P(x_t | x_{t-1})P(y_t | x_t)$$

Includes many models: any distribution, Naïve Bayes conditional independence assumption, Markov Chains, Hidden Markov models,….

$$p(y): x_{t+1} = x_t = 1 \; or \; anything; \; p(y | x) = p(y)$$
$$Multi-class \;\; problem: x_{t+1} = x_t = x \in \{1, 2, .., C\}; \; p(y | x = i) = p_i(y)$$
$$Markov\ Chain: P(x_{t+1} = j | x_t = i) = P_{ij}; \; y_t = x_t \Rightarrow P(y_t | x_t) = \begin{cases} \delta_{y_t x_t} ...discrete \\ \delta(y_t - x_t)...continuous \end{cases}$$

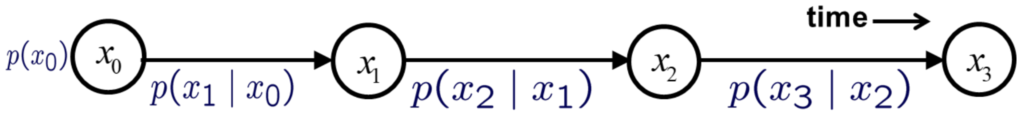
Perfectly observed HMM.... no need for y

- Automatic speech recognition: Here y_t represents features extracted from the speech signal, and x_t represents the word that is being spoken. The transition model $p(x_{t+1}|x_t)$ represents the language model, and the observation model $p(y_t |x_t)$ represents the acoustic model.
- Activity recognition: Here y_t represents features extracted from a video frame, and x_t is the class of activity the person is engaged in (e.g., running, walking, sitting, etc.).
- Part of speech tagging: Here y_t represents a word, and x_t represents its part of speech (noun, verb, adjective, etc.)
- Gene finding: Here y_t represents the DNA nucleotides (A,C,G,T), and x_t represents whether we are inside a gene-coding region or not.
- Protein sequence alignment: see (Durbin et al. 1998) for details on profile HMMs.
- Time series Prediction, as a black-box model of sequences, COVID-19 severity,…

You can think of HMM as a dynamic clustering/quantization scheme. It is unsupervised! Only data you have is observations.

Markov Chains:

- Sentence completion: A language model can predict the next word given the previous words in a sentence (reduce the amount of typing required, particularly for disabled users)
- Data compression
- Text classification
- Automatic essay writing



$$\underline{\alpha}_t = \begin{bmatrix} P(x_t = 1) \\ P(x_t = 2) \\ . \\ P(x_t = N) \end{bmatrix}; \underline{\beta}_t = \begin{bmatrix} P(x_T \in \Omega \mid x_t = 1) \\ P(x_T \in \Omega \mid x_t = 2) \\ . \\ P(x_T \in \Omega \mid x_t = N) \end{bmatrix}$$

$$Note: P(x_T \in \Omega) = \underline{\beta}_t^T \underline{\alpha}_t \forall t$$

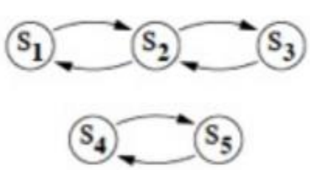
$$\alpha_{t+1}(j) = P(x_{t+1} = j) = \sum_{i=1}^{N} P(x_{t+1} = j, x_t = i) = \sum_{i=1}^{N} P(x_{t+1} = j \mid x_t = i) P(x_t = i) = \sum_{i=1}^{N} P_{ij} \alpha_t(i)$$

Rows of P sum to 1 $\Rightarrow$ have an eigen value of 1 with an eigen vector $\underline{e}$

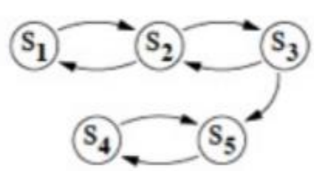
$\underline{\alpha}_{t+1} = P^T \underline{\alpha}_t = \left(P^T\right)^{t+1} \underline{\alpha}_0 = \left(P^T\right)^{t+1} \underline{\pi} \Rightarrow \underline{\alpha}_\infty = \left(P^T\right)^{\infty} \underline{\alpha}_0 = P^T \underline{\alpha}_\infty$....each column of $\left(P^T\right)^{\infty}$ is $\underline{\alpha}_\infty$!

P^T also has an eigen value of 1 with an eigen vector $\underline{\alpha}_\infty$...steady-state probability distribution

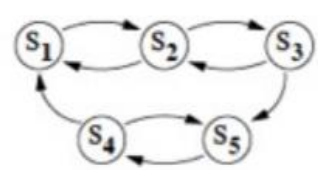
This happens if the chain is irreducible (there exists a path from every state to every other state) and aperiodic (a state is not visited periodically!).



Reducible

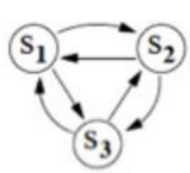


Reducible

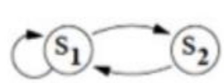


Irreducible

Periodic



Aperiodic



Aperiodic

$$\beta_t(i) = \sum_{j=1}^{N} P(x_T \in \Omega, x_{t+1} = j \mid x_t = i) = \sum_{j=1}^{N} P_{ij}\beta_{t+1}(j)$$

$$\underline{\beta}_t = P\underline{\beta}_{t+1}; \beta_T(i) = \begin{cases} 1 \, if \; i \in \Omega \\ 0 \; if \;\; i \notin \Omega \end{cases}$$

$\alpha\text{'}s \; and \; \beta\text{'}s$ are called forward and backward variables.

The equations are due to Andrey Kolomogorov.

$$P(x_T \in \Omega) = \sum_{i=1}^{N} P(x_T \in \Omega \mid x_t = i) P(x_t = i) = \sum_{i=1}^{N} \beta_t(i)\alpha_t(i) \forall t$$

$\Rightarrow$ in particular $P(x_T \in \Omega) = \sum_{i=1}^{N} \beta_0(i)\alpha_0(i) = \sum_{i=1}^{N} \beta_T(i)\alpha_T(i) = \sum_{i \in \Omega} \alpha_T(i)$

If want to evaluate for different initial conditions, use backward!

If want to evaluate for different goal states, use forward.

Similar forward-backward equations are valid for HMMs!

For online data analysis, we seek filtered state estimates given observations so far:

$$P(x_t \mid y_1, y_2, ..., y_t); t = 1, 2, ...,$$

In other cases, find smoothed estimates given earlier and later observations:

$$P(x_t \mid y_1, y_2, ..., y_T); t = 1, 2, ...,$$

Lots of other alternatives, including fixed-lag smoothing & fixed-lag prediction:

$$P(x_t \mid y_1, y_2, ..., y_{t+L}) \qquad P(x_t \mid y_1, y_2, ..., y_{t-L})$$

Smoothing:

$$\begin{aligned}
\gamma_t(x_t) = P(x_t \mid y_1, y_2, ..., y_t, y_{t+1}, ..., y_T) &= \frac{P(x_t, y_1, y_2, ..., y_t, y_{t+1}, ..., y_T)}{P(y_1, y_2, ..., y_t, y_{t+1}, ..., y_T)} \\
&= \frac{P(y_1, y_2, ..., y_t, x_t) P(y_{t+1}, ..., y_T \mid x_t)}{P(y_1, y_2, ..., y_t, y_{t+1}, ..., y_T)} \\
&= \frac{\alpha_t(x_t)\beta_t(x_t)}{\sum_{x_t} \alpha_t(x_t)\beta_t(x_t)}
\end{aligned}$$

$\alpha_t(x_t)$ = joint probability of observing all of the data upto time t and the value of x_t

$\beta_t(x_t)$ = conditional probability of all future data from time $(t+1)$ to T given the value of x_t

Forward Recursion:

$$\begin{aligned}
\alpha_t(x_t) &= P(y_1, y_2, ..., y_{t-1}, y_t, x_t) = P(y_t \mid x_t) P(y_1, y_2, ..., y_{t-1}, x_t) \\
&= P(y_t \mid x_t) \sum_{x_{t-1}} P(x_t \mid x_{t-1}) P(y_1, y_2, ..., y_{t-1}, x_{t-1}) \\
&= P(y_t \mid x_t) \sum_{x_{t-1}} P(x_t \mid x_{t-1}) \alpha_{t-1}(x_{t-1}); \alpha_0(x_0) = P(x_0) = \pi(x_0)
\end{aligned}$$

Filtering: $\delta_t(x_t) = P(x_t \mid y_1, y_2, ..., y_{t-1}, y_t) = \dfrac{P(y_1, y_2, ..., y_{t-1}, y_t, x_t)}{P(y_1, y_2, ..., y_{t-1}, y_t)} = \dfrac{\alpha_t(x_t)}{\sum_{\tilde{x}_t} \alpha_t(\tilde{x}_t)}$

$$p(\underline{x}_t \mid \underline{y}_1, \underline{y}_2, .., \underline{y}_t) = \frac{p(\underline{y}_t \mid \underline{x}_t)\, p(\underline{x}_t \mid \underline{y}_1, \underline{y}_2, .., \underline{y}_{t-1})}{p(\underline{y}_t \mid \underline{y}_1, \underline{y}_2, .., \underline{y}_{t-1})} = \frac{p(\underline{y}_t \mid \underline{x}_t) \int_{\underline{x}_{t-1}} p(\underline{x}_t \mid \underline{x}_{t-1})\, p(\underline{x}_{t-1} \mid \underline{y}_1, \underline{y}_2, ..., \underline{y}_{t-1})\, d\,\underline{x}_{t-1}}{p(\underline{y}_t \mid \underline{y}_1, \underline{y}_2, .., \underline{y}_{t-1})}$$

Backward Recursion:

$$\begin{aligned}
\beta_t(x_t) = P(y_{t+1}, y_{t+2}, ..., y_T \mid x_t) &= \sum_{x_{t+1}} P(x_{t+1}, y_{t+1}, y_{t+2}, ..., y_T \mid x_t) \\
&= \sum_{x_{t+1}} P(y_{t+2}, y_{t+3}, ..., y_T \mid x_{t+1}) P(y_{t+1} \mid x_{t+1}) P(x_{t+1} \mid x_t) \\
&= \sum_{x_{t+1}} \beta_{t+1}(x_{t+1}) P(y_{t+1} \mid x_{t+1}) P(x_{t+1} \mid x_t); \beta_T(x_T) = 1 \forall x_T
\end{aligned}$$

$$Why: \gamma_T(x_T) = \delta_T(x_T) = P(x_T \mid y_1, y_2, ..., y_T) = \frac{P(x_T, y_1, y_2, ..., y_T)}{P(y_1, y_2, ..., y_T)} = = \frac{\alpha_T(x_T)}{\sum_{\tilde{x}_T} \alpha_T(\tilde{x}_T)} \Rightarrow \beta_T(x_T) = 1 \forall x_T$$

$Also,\ P(y_1, y_2, ..., y_T) = \sum_{x_t} \alpha_t(x_t) \beta_t(x_t)\ \forall t$. Note that $P(y_1, y_2, ..., y_T) = \sum_{x_T} \alpha_T(x_T) = \sum_{x_0} \alpha_0(x_0) \beta_0(x_0)$

Joint Conditional probability of (x_t, x_{t-1})

$$\begin{aligned}
\xi_t(x_{t-1}, x_t) &= P(x_{t-1}, x_t \mid y_1, y_2, ..., y_{t-1}, y_t, ..., y_T) \\
&= \frac{P(y_1, y_2, ..., y_{t-1}, y_t, ..., y_T, x_{t-1}, x_t)}{P(y_1, y_2, ..., y_{t-1}, y_t, ..., y_T)} \\
&= \frac{P(y_1, y_2, ..., y_{t-1}, x_{t-1}) P(x_t, y_t, y_{t+1}, ..., y_T \mid x_{t-1})}{P(y_1, y_2, ..., y_{t-1}, y_t, ..., y_T)} \\
&= \frac{P(y_1, y_2, ..., y_{t-1}, x_{t-1}) P(y_t, y_{t+1}, ..., y_T \mid x_t, x_{t-1}) P(x_t \mid x_{t-1})}{P(y_1, y_2, ..., y_{t-1}, y_t, ..., y_T)} \\
&= \frac{P(y_1, y_2, ..., y_{t-1}, x_{t-1}) P(y_t, y_{t+1}, ..., y_T \mid x_t) P(x_t \mid x_{t-1})}{P(y_1, y_2, ..., y_{t-1}, y_t, ..., y_T)} \\
&= \frac{P(y_1, y_2, ..., y_{t-1}, x_{t-1}) P(x_t \mid x_{t-1}) P(y_t \mid x_t) P(y_{t+1}, ..., y_T \mid x_t)}{P(y_1, y_2, ..., y_{t-1}, y_t, ..., y_T)} \\
&= \frac{\alpha_{t-1}(x_{t-1}) P_{x_{t-1}x_t} b_{y_t x_t} \beta_t(x_t)}{\sum_{x_t} \alpha_t(x_t) \beta_t(x_t)} = \frac{\alpha_{t-1}(x_{t-1}) P_{x_{t-1}x_t} b_{y_t x_t} \beta_t(x_t)}{\sum_{x_{t-1}} \sum_{x_t} \alpha_{t-1}(x_{t-1}) b_{y_t x_t} P_{x_{t-1}x_t} \beta_t(x_t)}
\end{aligned}$$

$Evidently,$

$$\gamma_t(x_t) = P(x_t \mid y_1, y_2, ..., y_t, y_{t+1}, ..., y_T) = \sum_{x_{t-1}} \xi_t(x_{t-1}, x_t)$$

Discuss the four problems: Once a HMM is specified, it can be used to (1) generate an observation sequence; (2) compute the likelihood of observing a given sequence, given model parameters; (3) learn the parameters from observed data; and (4) determine the most likely evolution of the state sequence over time.

Baum-Welch (EM) Algorithm for learning HMM parameters:

- Computing α,β,ξ is the E-step
- M-step is learning the parameters: transition probabilities, emission probabilities and initial state probabilities

  $\hat{\pi}_i = \text{expected freq. in state } i \text{ at time } (t=0) = \gamma_0(i)$

  $$\hat{P}_{ij} = \frac{\text{Expected no. of transitions from } i \text{ to } j}{\text{Expected no. of transitions from } i} = \frac{\sum_{t=1}^{T} \xi_t(i,j)}{\sum_{t=0}^{T-1} \gamma_t(i)}$$

  $$\boxed{\gamma_t(x_t) = \frac{\alpha_t(x_t)\beta_t(x_t)}{\sum_{x_t} \alpha_t(x_t)\beta_t(x_t)}}$$

  $$\boxed{\xi_t(x_{t-1}, x_t) = \frac{\alpha_{t-1}(x_{t-1}) b_{y_t x_t} P_{x_{t-1}x_t} \beta_t(x_t)}{\sum_{x_t} \alpha_t(x_t)\beta_t(x_t)}}$$

  $$\hat{b}_{kj} = \frac{\text{exptd. no. of times in state } j \text{ and observing symbol } k}{\text{expected number of times in state } j} = \frac{\sum_{\substack{t=1 \\ s.t.\ y_t = k}}^{T} \gamma_t(j)}{\sum_{t=1}^{T} \gamma_t(j)}$$

  To avoid small $\{\alpha,\ \beta\}$, normalize *both* using the <u>same</u> normalization factor $Z(\mathrm{t}) = \sum_{i=1}^{N} \alpha_i(t)$

Viterbi algorithm: Problem: Find the most probable state sequence (MAP estimate) given data

$$\begin{aligned}
\underline{x}_{(0:T)} &= \arg \max_{x_0, x_1, .., x_T} P(x_0, x_1, ..., x_t, ..., x_T \mid y_1, y_2, ..., y_t, .., y_T) \\
&= \arg \max_{x_0, x_1, .., x_T} P(y_1, y_2, ..., y_t, .., y_T, x_0, x_1, ..., x_t, ..., x_T) \\
&= \arg \max_{x_0, x_1, .., x_T} P(x_0) \prod_{t=1}^{T} \left[P(y_t \mid x_t) P(x_t \mid x_{t-1}) \right] \\
&= \arg \max_{x_0, x_1, .., x_T} \left[\ln P(x_0) + \sum_{t=1}^{T} \left\{ \ln P(y_t \mid x_t) + \ln P(x_t \mid x_{t-1}) \right\} \right]
\end{aligned}$$

Use forward dynamic programming (DP) to recursively find the probability of the most likely state sequence

$$\omega(x_0) = \ln P(x_0)$$

$$\omega(x_1) = \ln P(y_1 \mid x_1) + \max_{x_0} \left[\ln P(x_1 \mid x_0) + \underbrace{\ln P(x_0)}_{\omega(x_0)} \right]$$

In general,

$$\omega(x_t) = \max_{x_0,x_1,..,x_{t-1}} P(y_1, y_2, ..., y_t, x_0, x_1, ..., x_t)$$

$$= \max_{x_0,x_1,..,x_{t-1}} \left[\ln P(x_0) + \sum_{n=1}^{t} \{ \ln P(y_n \mid x_n) + \ln P(x_n \mid x_{n-1}) \} \right]$$

$$= \ln P(y_t \mid x_t) + \max_{x_0,x_1,..,x_{t-1}} \left[\ln P(x_0) + \sum_{n=1}^{t-1} \ln P(y_n \mid x_n) + \sum_{n=1}^{t} \ln P(x_n \mid x_{n-1}) \right]$$

$$= \ln P(y_t \mid x_t) + \max_{x_{t-1}} \left[\ln P(x_t \mid x_{t-1}) + \omega(x_{t-1}) \right]; \omega(x_0) = \ln P(x_0); t = 1, 2, .., T$$

Keep a record of the values of x_{t-1} that correspond to the maxima for each of the N values of x_t.
Store this in a list function $\psi_t(x_t) = \arg\max_{x_{t-1}} \left[\ln P(x_t \mid x_{t-1}) + \omega(x_{t-1}) \right]$

Backtrack to get the best sequence

$$x_T^* = \arg\max_{x_T} \omega(x_T)$$

$$x_t^* = \psi_{t+1}(x_{t+1}^*); t = T-1, T-2, .., 1, 0$$

Poor Man's Viterbi:

$$x_t^* = \arg\max_{x_t} \delta_t(x_t)$$

$$\delta_t(x_t) \triangleq P(x_t \mid y_1, y_2, .., y_{t-1}, y_t)$$

Generalizations of HMMs:

- Higher order HMMs
- Factorial HMMs. See Tbishirani and some of my papers on diagnosis.
- Coupled HMMs. See some of my papers on diagnosis and refs there.
- Semi-Markov HMMs
- Hierarchical HMMs
- I/O HMMs
- Auto-regressive HMMs
- Buried HMMs
- **State space HMMs or State Space Models (SSMs)**
- Dynamic Bayesian networks (DBNs)

Discuss SS models using slides.