Recitation: HMM, GM and Learning Theory

Zeyu Jin

Outline

- Learning Theory
  - Uniform bound
  - |H| and VC(H)
  - Insights
- GM
  - Factorized probability
  - D-separation
  - Inference
- HMM (recap)
  - Basic questions
  - Algorithms
  - Insight

Learning Theory

1. The question
   - Want to know how good our classifier is

$$error_{true}(H) = ?$$

   - However, H is trained on some data; **the randomness of data** makes this "?" a distribution. Let's try

$$P(error_{true}(H) = p) = ?, \quad p \in [0,1]$$

   - It is non-trivial

Learning Theory

1. The question
   - With a family of models **H of certain complexity**, how many training **samples R** is needed in order to learn a model h with **reasonable training time** and **sufficient accuracy** on future data?

   - We want answer

$$error_{true}(H(X^m)) = ?$$

Computationally efficient in polynomial time

Learning Theory

1. The question
   – Distribution of error rate

$$P(error_{true}(H) = p)$$
$$= E[P(error_{true}(H(X)) = p \mid X = x)]$$
$$= \iiint_X P(error_{true}(H(X)) = p \mid X = x) P_{true}(X = x)\, dx$$

   – Maybe we can try to get a uniform bound for this question

$$P(\left| error_{true}(H) - E_X[error_{true}(H)] \right| < \varepsilon) = ?$$

Learning Theory

1. The question
   - Distribution of error rate

$$
\begin{aligned}
&P(error_{true}(H) = p) \\
&= E[P(error_{true}(H(X)) = p \mid X = x)] \\
&= \iiint_X P(error_{true}(H(X)) = p \mid X = x) P_{true}(X = x)\, dx
\end{aligned}
$$

   - Maybe we can try to get a uniform bound for this question

$$
P(\left| error_{true}(H) - E_X[error_{true}(H)] \right| < \varepsilon) = ?
$$

   - Still extremely hard. Maybe bound this probability

Learning Theory

1. Uniform Bound
   - Bound the probability of the bounded error

$$P(\left|error_{true}(H) - E_X[error_{true}(H)]\right| < \varepsilon) > 1 - \delta$$

   - Statisticians do have solution for this form!
   - Three basic questions
     - H is finite, $E_X[error_{true}(H)] = error_{train}(H)$ is 0 **PAC**
     - H is finite, $E_X[error_{true}(H)] = error_{train}(H)$ is non-zero
     - H is infinite

Learning Theory

2. Solutions

   1) H is finite, $E_X[error_{true}(H)] = error_{train}(H)$ is 0

$$P(\left|error_{true}(H) - 0\right| < \varepsilon) \geq 1 - \delta \qquad \varepsilon = \frac{\ln|H| + \ln(1/\delta)}{|X|}$$

   2) H is finite, $E_X[error_{true}(H)] = error_{train}(H)$ is non-zero

$$P(\left|error_{true}(H) - E_X[error_{true}(H)]\right| < \varepsilon) > 1 - \delta$$

$$\varepsilon = \sqrt{\frac{\left|H\right| + \ln(1/\delta)}{2|X|}}$$

Learning Theory

2. Solutions

   3) H is infinite

$$P(\left|error_{true}(H) - E_X[error_{true}(H)]\right| < \varepsilon) > 1 - \delta$$

$$\varepsilon = 8\sqrt{\frac{VC(H)\left(\ln\frac{m}{VC(H)} + 1\right) + \ln\frac{8}{\delta}}{2m}}$$

Learning Theory

3. Terms in this solutions
   - For solution 1 and 2: $|H| = ?$
   - For solution 3: $VC(H) = ?$

Learning Theory

3. Terms in this solutions
   - **For solution 1 and 2: $|H| = ?$**
   - For solution 3: $VC(H) = ?$

Instead of limiting the maximal depth of a decision tree, let's assume n binary attributes and binary class

What is $|H|$?

Learning Theory

3. Terms in this solutions
   - For solution 1 and 2: $|H| = ?$
   - **For solution 3: VC(H) = ?**
     - Find the **maximal N**
     - Where there **EXIST** N points in the problem's space
     - s.t. **ALL** element of the superset of these points (2^N)
     - can be picked out by S

Learning Theory

3. Terms in this solutions
   - For solution 1 and 2: |H| = ?
   - **For solution 3: VC(H) = ?**

What is the VC dimension of a 2D-circle?

$$\{(x, y) \mid x^2 + y^2 \leq R^2\}$$

Learning Theory

3. Terms in this solutions
   - For solution 1 and 2: |H| = ?
   - **For solution 3: VC(H) = ?**

What if the a circle plus a point?

$$\{(x,y) \mid x^2 + y^2 \le R^2 \setminus (0,0)\}$$

Learning Theory

4. Insights
   – VC Dimension
     - Find the **maximal N**
     - Where there **EXIST** N points in the problem's space
     - s.t. **ALL** element of the superset of these points (2^N)
     - can be picked out by S

$S_N(H)$ = The number of elements of the superset of these N points can be picked out by H

$$\mathbb{P}\left(\sup_{A\in\mathcal{A}} |P_n(A) - P(A)| > \epsilon\right) \leq 8\,(n+1)^d\, e^{-n\epsilon^2/32}.$$

$S_N(H)$ is not easy to obtain, but it can be shown that $s(\mathcal{A}, n) \leq (n+1)^d.$

Where d is VC dimension

Learning Theory

4. Insights
   - Obtain error bound by simulation
     - Known: marginal distribution of data D, true model p(Y|X)

Repeat the following N times

- Draw m data from D for training; draw k >> m from D for test
- Draw y_i for each x_i from p(Y|X)
- Learn h based on your hypothesis space H
- Evaluate $error_{true}$(h)-$error_{train(h)}$ on test data (or do it mathematically)

Then you will get a histogram of error which approximates P($error_{true}$(h)). Solve

$$P(error_{true}(H) < \varepsilon) \geq 1 - \delta$$

Learning Theory

4. Insights
   - Connection

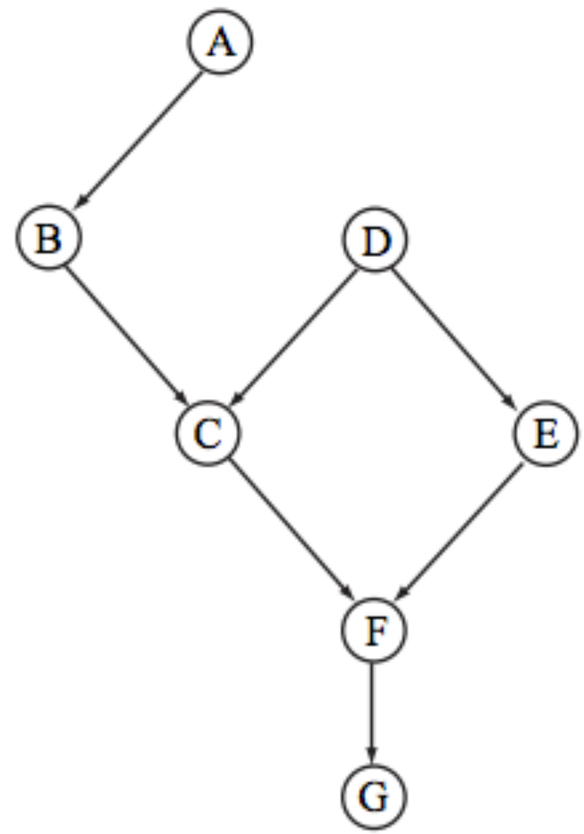
$$P(error_{true}(H) < \varepsilon) \geq 1 - \delta$$

Confidence interval: with confidence δ

we conclude that error of estimating $error_{true}$ is less than ε

Bayesian Networks

- Bayes Net ⇔ Factorized probability
  - Write Factorized probability

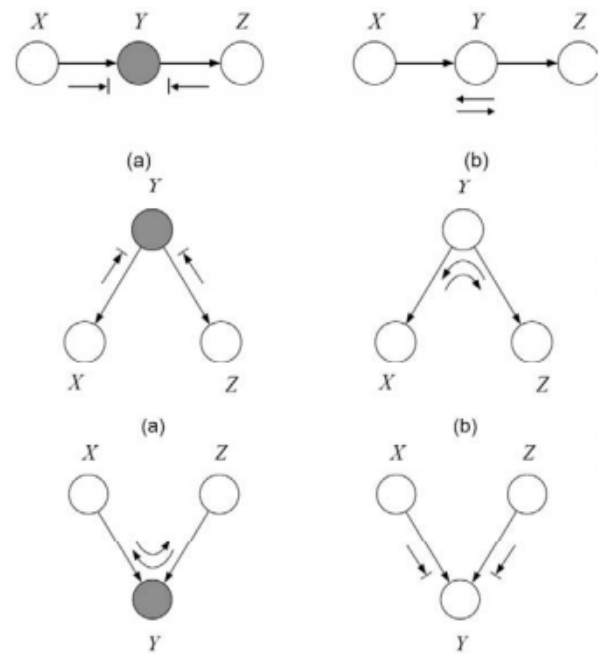


Bayesian Networks

- Bayes Net ⇔ Factorized probability
  - What's the Bayes net for
    - Naïve Bayes?
    - Full Bayes?
    - k-th order Markov Model
    - Hidden Markov model

Bayesian Networks

- Understand dependency in BN – D-separation



X and Y are D-separated by Z

If all the path from X to Y are blocked

Maybe we lost one case in class

Bayesian Networks

- Understand dependency in BN – D-separation

Original Definition

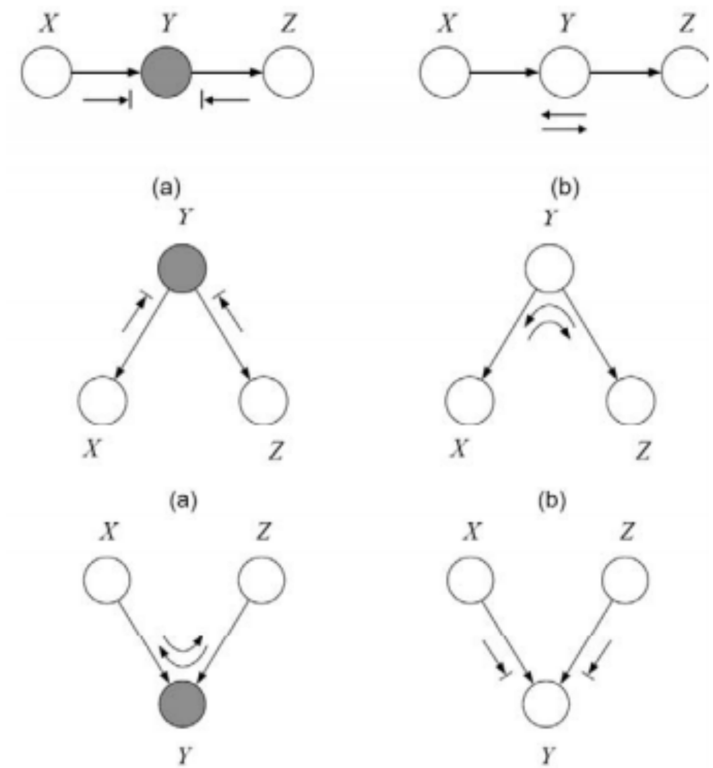
(a) the arrows on the path meet either head-to-tail or tail-to-tail at the node, and the node is in the set C, or

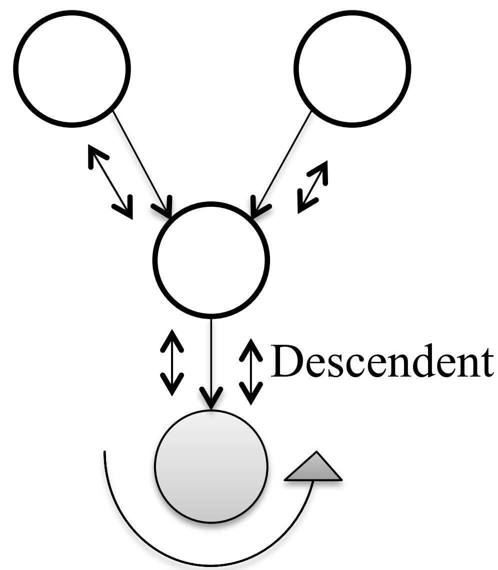
(b) the arrows meet head-to-head at the node, and neither the node, nor any of its descendants, is in the set C.

Bishop 8.2.2

Bayesian Networks

- Understand dependency in BN – D-separation





Bayesian Networks

- Understand dependency in BN – D-separation

Exam problem TRUE/FALSE

(a) $P(D, H) = P(D)P(H)$

(b) $P(A, I) = P(A)P(I)$

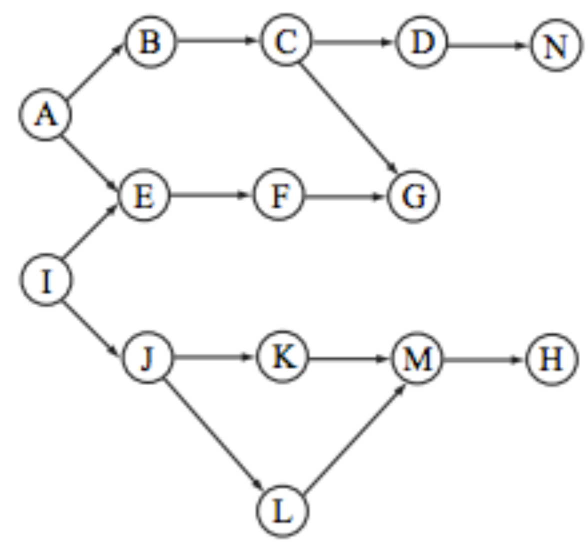
(c) $P(A, I|G) = P(A|G)P(I|G)$

(d) $P(J, G|F) = P(J|F)P(G|F)$

(e) $P(J, M|K, L) = P(J|K, L)P(M|K, L)$

(f) $P(E, C|A, G) = P(E|A, G)P(C|A, G)$

(g) $P(E, C|A) = P(E|A)P(C|A)$



Bayesian Networks

- Understand dependency in BN – D-separation

Exam problem TRUE/FALSE

(a) $P(D, H) = P(D)P(H)$

(b) $P(A, I) = P(A)P(I)$

(c) $P(A, I|G) = P(A|G)P(I|G)$

(d) $P(J, G|F) = P(J|F)P(G|F)$

(e) $P(J, M|K, L) = P(J|K, L)P(M|K, L)$

(f) $P(E, C|A, G) = P(E|A, G)P(C|A, G)$

(g) $P(E, C|A) = P(E|A)P(C|A)$

Key…

a) Yes, blocked by on E on one path and G on another path
b) Yes, blocked by E
c) No, G is a descendent of E
d) No, the path JIEABCG is unblocked
e) Yes, blocked on both paths
f) No, path EFGC unblocked
g) Yes, EABC blocked by A, and EFGC blocked by G

Bayesian Networks

- Inference
  - What is inference?
    the process of computing answers to queries about the distribution P defined by given BN
    - Likelihood
    - Conditional probability (we will see one example after this slide)
    - Most probable assignment (most likely states sequence in HMM)
  - Methods?
    - Variable elimination, belief propogation (do exact calculation)
    - Gibbs sampling (simulation)

Bayesian Networks

- Inference 1: variable elimination

$$P(G=T|A=T)=?$$

$$P(G=T|A=T)=\frac{P(G=T,A=T)}{P(A=T)}=\sum_{BCDEF}P(A=T,B,C,D,E,F,G=T)$$

$$=\sum_{BCDEF}P(B|A=T)P(D)P(C|B,D)P(E|C,D)P(F|C,E)P(G=T|F)$$

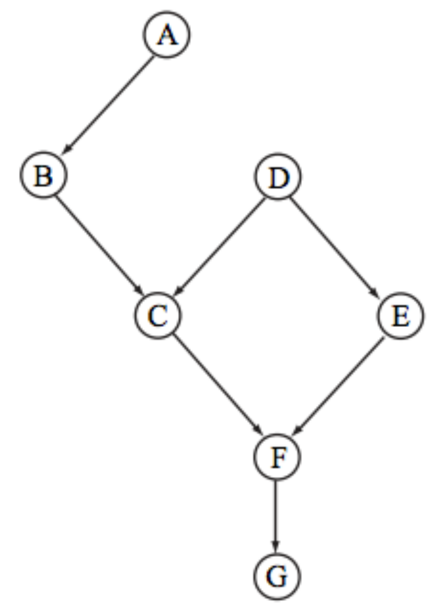
$$=\sum_{B}P(B|A=T)\sum_{D}P(D)\sum_{C}P(C|B,D)\sum_{E}P(E|D)\sum_{F}P(F|C,E)P(G=T|F)$$

$$f_{F,G}(c,e,G=T)=\sum_{F}P(F|C,E)P(G=T|F)$$

$$f_{C,D}(c,d,G=T)=\sum_{E}P(E|d)f_{F,G}(c,E,G=T)$$

$$f_{B,D}(b,d,G=T)=\sum_{C}Pf_{C}(c,B,D)f_{C,D}(c,d,G=T)$$

…

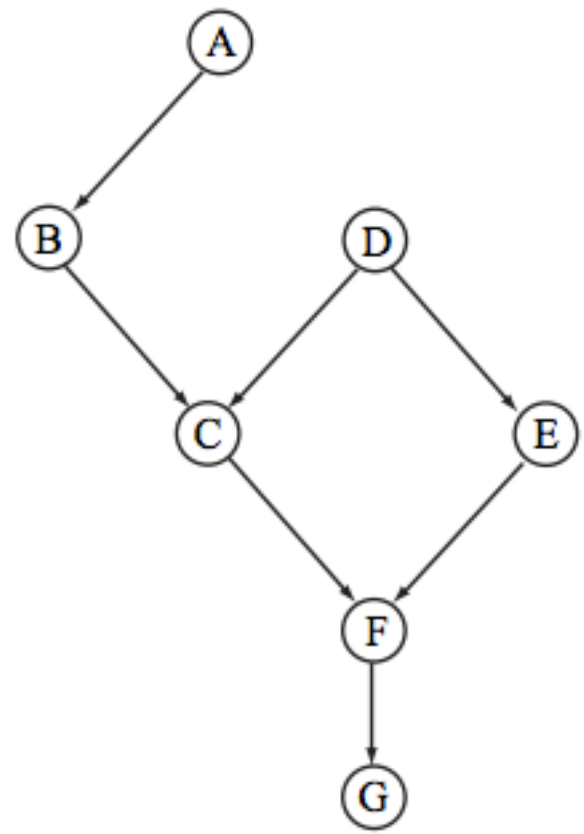


Binary for all RV

Bayesian Networks

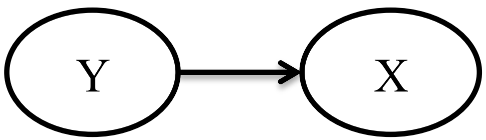
- Inference 2: sampling
  - Naïve sampling:
    - (A,B,C,D,E,F,G) each time from p(A,B,C,D,E,F,G)
    - Calculate P(G|A=T) by counting
    - Have problem with rare event
  - Weighted sampling
    - If P(A=T) is rare, just set A=T and sample (A=T,B,C,D,E,F,G)
    - When calculating P(G|A=T), the number of (A=T,b,c,d,e,f,g) is weighted by p(A=T)

$$P(G = T | A = T) = ?$$

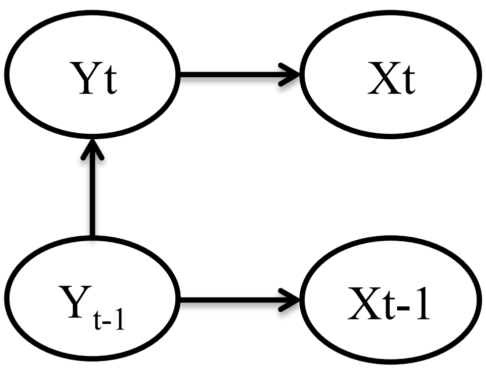


HMM (recap)

- Static vs. time series



	Discriminative	Generative
	$P(Y\|X)$	$P(X,Y)$
	$P(Yt, Yt\text{-}1 \ldots \|Xt\ Xt\text{-}1 \ldots)$	$P(Xt, Yt, X_{t-1}, Y_{t-1} \ldots$
	Conditional random field	



Conditioning on no variable, all Xs and Ys are correlated

HMM (recap)

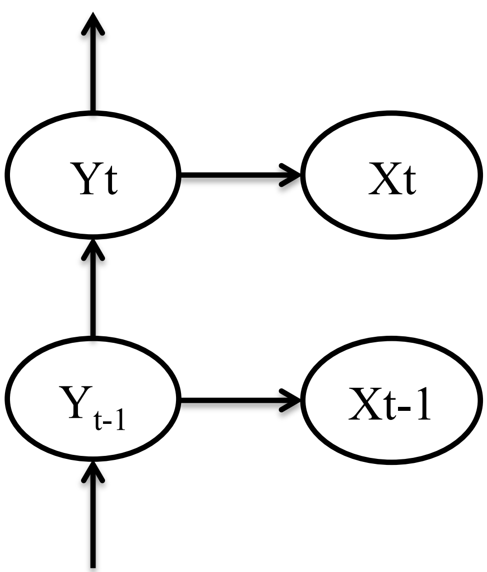
- Basic questions
  1. **Parameters**
  2. Factorization
  3. Inference
  4. Learning

- K: number of states
- M: number of observations

	#par.	shorthand
– Initial state: $P(y_1)$	K-1	$\pi_i = P(y_1=i)$
– Transitoin: $P(y_t \mid y_{t-1})$	K*(K-1)	$a_{ij} = P(y_{t+1}=j \mid y_t=i)$
– Emission: $P(x_t \mid y_t)$	K*(M-1)	$b_{ik} = P(x_t=k \mid y_t=i)$

HMM (recap)

- Basic questions
  1. Parameters
  2. **Factorization**
  3. Inference
  4. Learning



HMM is Generative

Complete likelihood based on given parameters is

$$P(x_1, \ldots, x_T, y_1, \ldots, y_T)$$

$$= P(y_1)p(x_1|y_1)p(y_2|y_1)\ldots P(y_2|y_1)P(x_T|y_T)$$

$$= P(y_1)\prod_t P(y_t|y_{t-1})P(x_t|y_t)$$

HMM (recap)

- Basic questions
  1. Parameters
  2. Factorization
  3. **Inference**
  4. Learning

$P(y_t|X_{1:t}) = ?$

$P(y_t|X_{1:T}) = ?$

$\mathrm{argmax}_y\ P(y_{1:T}|X_{1:T}) = ?$

Before that, we have the following tools

1. Forward probability

$$\alpha_t^k = P(x_1,\dots,x_{t-1},x_t,y_t=k) = P(x_t \mid y_t=k)\sum_i \alpha_{t-1}^i a_{i,k}$$

2. Backward probability

$$\beta_t^i = p(x_{t+1},\dots,x_T|y_t=i) = \sum_i a_{k,i}\, p(x_{t+1} \mid y_{t+1}=i)\beta_{t+1}^i$$

HMM (recap)

- Basic questions
  1. Parameters
  2. Factorization
  3. **Inference**
  4. Learning

$\underline{P(y_t|X_{1:t}) = ?}$

$P(y_t|X_{1:T}) = ?$

$\text{argmax}_y\ P(y_{1:T}|X_{1:T}) = ?$

$$p(y_t = i \mid X_{1:t}) = \frac{p(x_1,\ldots,x_t,y_t = i)}{p(x_1,\ldots,x_t)} = \frac{\alpha_t^i}{\boxed{p(x_1,\ldots,x_t) = }} \sum_{i=1}^{k} \alpha_T^i$$

HMM (recap)

- Basic questions
  1. Parameters
  2. Factorization
  3. **Inference**
  4. Learning

$P(y_t|X_{1:t}) = ?$

$\underline{\mathbf{P(y_t|X_{1:T}) = ?}}$

$\text{argmax}_y\, P(y_{1:T}|X_{1:T}) = ?$

$$
\begin{aligned}
p(y_t = i | x_1, \ldots, x_T) &= \frac{p(y_t = i, x_1, \ldots, x_T)}{p(x_1, \ldots, x_T)} \\
&= \frac{p(y_t = i, x_1, \ldots, x_t) p(x_{t+1}, \ldots, x_T | y_t = i, \boxed{x_1, \ldots, x_t})}{p(x_1, \ldots, x_T)} \\
&= \frac{\alpha_t^i \beta_t^i}{\boxed{p(x_1, \ldots, x_T)}} \quad = \sum_{i=1}^{k} \alpha_T^i
\end{aligned}
$$

?

HMM (recap)

- Basic questions
  1. Parameters
  2. Factorization
  3. **Inference**
  4. Learning

$P(y_t|X_{1:t}) = ?$

$P(y_t|X_{1:T}) = ?$

$\underline{\mathbf{argmax_y\ P(y_{1:T}|X_{1:T}) = ?}}$

$$V_{t+1}^{k} = \max_{\{y_1,\dots y_t\}} P(x_1,\dots,x_t,y_1,\dots,y_t,x_{t+1},y_{t+1}=k)$$

$$= \max_{\{y_1,\dots y_t\}} P(x_1,\dots,x_t,y_1,\dots,y_t)P(x_{t+1},y_{t+1}=k \mid x_1,\dots,x_t,y_1,\dots,y_t)$$

$$= \max_{\{y_1,\dots y_t\}} P(x_{t+1},y_{t+1}=k \mid y_t)P(x_1,\dots,x_{t-1},y_1,\dots,y_{t-1},x_t,y_t)$$

$$= \max_i P(x_{t+1},y_{t+1}=k \mid y_t=i)\max_{\{y_1,\dots y_{t-1}\}} P(x_1,\dots,x_{t-1},y_1,\dots,y_{t-1},x_t,y_t=i)$$

$$= \max_i P(x_{t+1}, \mid y_{t+1}=k)a_{i,k}V_t^i$$

$$= P(x_{t+1}, \mid y_{t+1}=k)\max_i a_{i,k}V_t^i$$

Viterbi

HMM (recap)

- Basic questions
  1. Parameters
  2. Factorization
  3. Inference
  4. **Learning**

$$\langle \ell_c(\theta; \mathbf{x}, \mathbf{y}) \rangle = \sum_n \left(\langle y_{n,1}^i \rangle_{p(y_{n,1}|\mathbf{x}_n)} \log \pi_i \right) + \sum_n \sum_{t=2}^{T} \left(\langle y_{n,t-1}^i y_{n,t}^j \rangle_{p(y_{n,t-1}, y_{n,t}|\mathbf{x}_n)} \log a_{i,j} \right) + \sum_n \sum_{t=1}^{T} \left(x_{n,t}^k \langle y_{n,t}^i \rangle_{p(y_{n,t}|\mathbf{x}_n)} \log b_{i,k} \right)$$

E-step

$$\gamma_{n,t}^i = \langle y_{n,t}^i \rangle = p(y_{n,t}^i = 1 \mid \mathbf{x}_n)$$

$$\xi_{n,t}^{i,j} = \langle y_{n,t-1}^i y_{n,t}^j \rangle = p(y_{n,t-1}^i = 1, y_{n,t}^j = 1 \mid \mathbf{x}_n)$$

M-step

$$\pi_i^{ML} = \frac{\sum_n \gamma_{n,1}^i}{N} \qquad a_{ij}^{ML} = \frac{\sum_n \sum_{t=2}^{T} \xi_{n,t}^{i,j}}{\sum_n \sum_{t=1}^{T-1} \gamma_{n,t}^i} \qquad b_{ik}^{ML} = \frac{\sum_n \sum_{t=1}^{T} \gamma_{n,t}^i x_{n,t}^k}{\sum_n \sum_{t=1}^{T-1} \gamma_{n,t}^i}$$

HMM (recap)

- Computational complexity
  - Forward: K states, N time points => $O(K^2N)$
  - Backward: $O(K^2N)$
  - $P(y_t|X_{1:t})$: Forward + sum of forward = $O(K^2N)$
  - $P(y_t|X_{1:T})$: Forward + backward + sum of forward = $O(K^2N)$
  - Viterbi: forward + $O(K^2N)$ updates of V = $O(K^2N)$

HMM (recap)

- Other mutation of HMM
  - IO-HMM
  - Kalman Filter
  - MEMM

  - Spectral HMM (algorithm)