



یادگیری ماشین
(۰۱-۸۰۵-۱۱-۱۳)
فصل پانزدهم

مدل مخفی مارکوف



دانشگاه شهید بهشتی
دانشکده‌ی مهندسی برق و کامپیوتر
پاییز ۱۳۹۴
احمد محمودی ازناوه

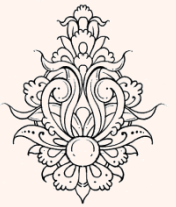
فهرست مطالب

- زنجیره‌ی مارکوف
- مدل مارکوف قابل مشاهده
- مدل مخفی مارکوف
- چند مثال
- مسائل سه‌گانه

– ارزیابی

– یافتن زنجیره‌ی حالات

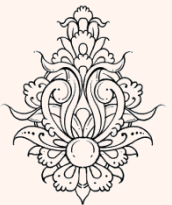
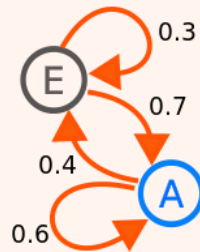
– آموزش



پیش‌گفتار

- تاکنون فرض می‌شد که نمونه‌ها «متغیرهای تصادفی مستقل با توزیع یکسان (iid)» هستند.
 - مزیت این فرض سادگی محاسبه‌ی درست‌نمایی است.
 - در عین حال، برای برخی کاربردها که نمونه‌های متوالی وابستگی دارند، این پیش‌فرض پذیرفتنی نیست.
- به عنوان مثال مروف یک کلمه وابستگی دارند، به عنوان مثال در زبان انگلیسی حرف h با احتمال یکسانی بعد از حرف‌های x و t ظاهر نمی‌شود.
- بازشناسی صدا نیز مربوط به شناسایی واچهایی است که به یکدیگر وابسته هستند و تنها توالی مشخصی از این واچه‌ها معتبر هستند، در سطحی بالاتر هر ترتیبی از کلمه‌ها نیز مجاز نیستند.
- یک «فرآیند تصادفی پارامتری» می‌تواند توالی نمونه‌ها را تولید کند.

Parametric random process



فرآیندهای گسسته‌ی مارکوف

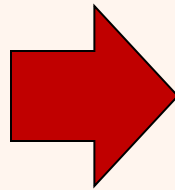
Discrete Markov Process

- سیستمی را در نظر بگیرید که در هر لحظه از زمان در یکی از N حالت مشخص شده باشد:

$$S_1, S_2, \dots, S_N$$

- حالت سیستم در زمان t با q_t نمایش داده می‌شود:

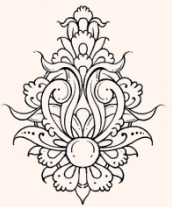
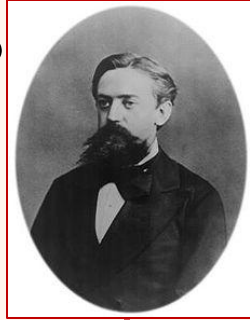
$$q_t = S_i$$



در زمان t سیستم در حالت S_i می‌باشد

- احتمال تغییر حالت سیستم به حالتی دیگر با توجه به حالت‌های قبلی سیستم تعیین می‌شود:

$$P(q_{t+1} = S_j \mid q_t = S_i, q_{t-1} = S_k, \dots)$$



فرآیندهای گسسته‌ی مارکوف (ادامه...)

- برای حالت خاصی از مدل مارکوف، حالت در زمان $t+1$ تنها به حالت در زمان t بستگی دارد، که به آن «مدل

مارکوف مرتبه‌ی اول» می‌گویند. **First-order Markov Model**

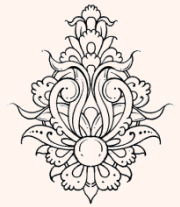
$$P(q_{t+1} = S_j \mid q_t = S_i, q_{t-1} = S_k, \dots) = P(q_{t+1} = S_j \mid q_t = S_i)$$

- با فرض این که «احتمال گذار» (transition) مستقل از زمان باشد:

$$a_{ij} \equiv P(q_{t+1} = S_j \mid q_t = S_i) \quad a_{ij} \geq 0 \text{ and } \sum_{j=1}^N a_{ij} = 1$$

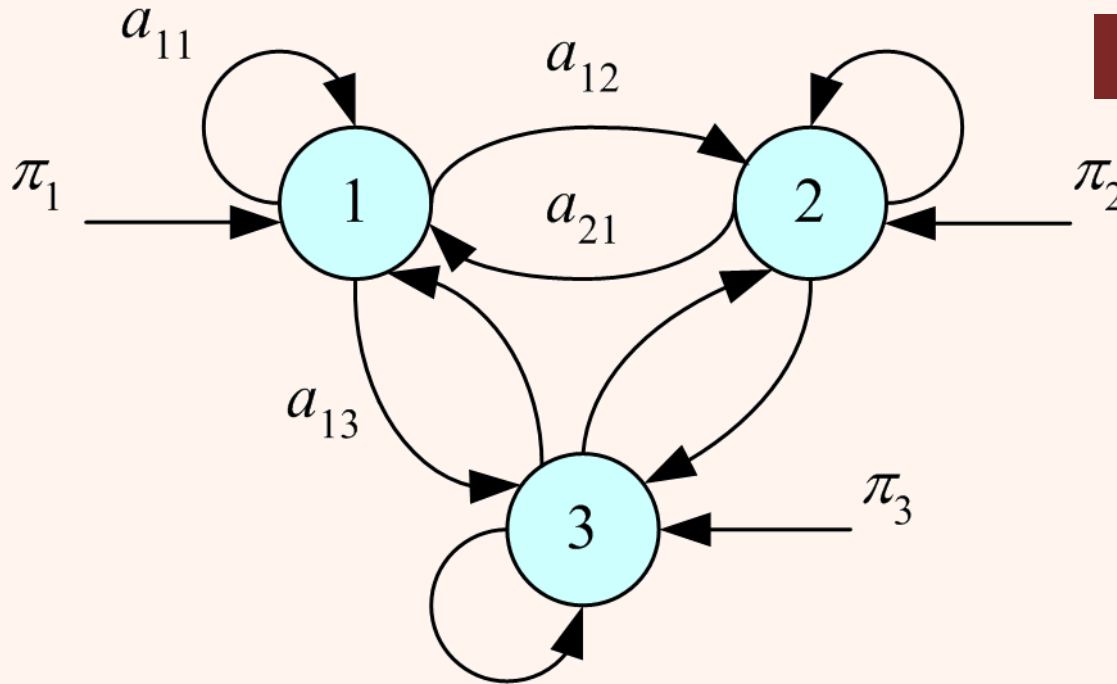
- احتمال اولیه، احتمال این است که اولین حالت S_i باشد:

$$\pi_i \equiv P(q_1 = S_i) \quad \sum_{i=1}^N \pi_i = 1$$

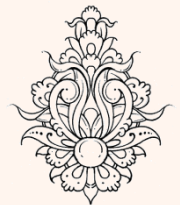


فرآیندهای گسسته‌ی مارکوف (ادامه...)

Stochastic automaton



- $A=[a_{ij}]$ یک ماتریس با ابعاد $N \times N$ است که جمع عناصر هر سطر آن برابر یک می‌شود.
- $\Pi=[\pi_i]$ برداری N -تایی است که حاصل جمع تمام عناصر آن برابر یک است.



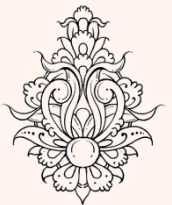
Observable Markov Model

- در یک «مدل مارکوف قابل مشاهده»، در زمان t می‌دانیم که q_t کدام حالت را نشان می‌دهد.
- فروجی فرآیند، برپسب حالت فعلی است؛ هر حالت متناظر با مشاهده‌ی یک رخداد فیزیکی می‌باشد.

Observation sequence

- «دنباله‌ی مشاهدات»، O ، در اینجا معادل ترتیب حالت‌های مشاهده شده است، که احتمال رخداد آن به صورت زیر محاسبه می‌شود:

$$P(O = Q | A, \Pi) = p(q_1) \prod_{t=2}^T p(q_t | q_{t-1}) = \pi_{q_1} a_{q_1 q_2} \cdots a_{q_{T-1} q_T}$$



مثال ۱

- هر حالت بیانگر وضعیت جوی در یک زمان مشخص در روز (مثلاً ظهر) می‌باشد:

- حالت ۱: وجود بارندگی

- حالت ۲: هوای ابری

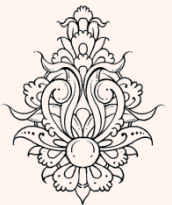
- حالت ۳: هوای آفتابی

- ماتریس انتقال:

$$A = \{a_{ij}\} = \begin{bmatrix} 0.4 & 0.3 & 0.3 \\ 0.2 & 0.6 & 0.2 \\ 0.1 & 0.1 & 0.8 \end{bmatrix}$$

- با فرض این که در روز اول هوا آفتابی باشد، احتمال این که هفت روز بعد، آفتابی-آفتابی-بارانی-بارانی-آفتابی-آفتابی باشد:

$$O = \{S_3, S_3, S_3, S_1, S_1, S_3, S_2, S_3\}$$



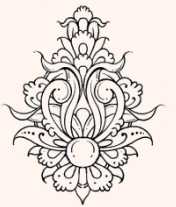
مثال (ادامه...)

$$\begin{aligned}P(O | Model) &= P(S_3, S_3, S_3, S_1, S_1, S_3, S_2, S_3) \\&= P(S_3)P(S_3 | S_3)P(S_3 | S_3)P(S_1 | S_3) \dots \\&\quad \dots P(S_1 | S_1)P(S_3 | S_1)P(S_2 | S_3)P(S_3 | S_2) \\&= \pi_3 a_{33} a_{33} a_{31} a_{11} a_{13} a_{32} a_{23} \\&= 1.(0.8).(0.8).(0.1).(0.4).(0.3).(0.1).(0.2) \\&= 1.536 \times 10^{-4}\end{aligned}$$

- احتمال باقی ماندن مدل در یک حالت به اندازهی زمان d :

$$O = \left\{ S_{i_1}, S_{i_2}, S_{i_3}, \dots, S_{i_d}, S_{i_{d+1}} \neq S_{i_d} \right\}$$

$$p_i(d) \equiv P(O | Model, q_1 = S_i) = (a_{ii})^{d-1} (1 - a_i)$$



مثال (ادامه...)

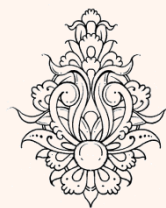
$$p_i(d) \equiv P(O \mid Model, q_1 = S_i) = (a_{ii})^{d-1} (1 - a_{ii})$$

به طور متوسط چند روز پیاپی هوا آفتابی است؟

$$\begin{aligned} E[d_i] &= \sum_{d=1}^{\infty} d p_i(d) = \sum_{d=1}^{\infty} d (a_{ii})^{d-1} (1 - a_{ii}) \\ &= (1 - a_{ii}) \sum_{d=1}^{\infty} d (a_{ii})^{d-1} = \frac{1}{1 - a_{ii}} \end{aligned}$$

به عنوان نمونه در مثال فوق انتظار می‌رود به طور متوسط پنج روز پیاپی هوا آفتابی، ۲.۵ روز ابری و تنها ۱.۶۷ روز متوالی هوا بارانی باشد.

$$A = \{a_{ij}\} = \begin{bmatrix} 0.4 & 0.3 & 0.3 \\ 0.2 & 0.6 & 0.2 \\ 0.1 & 0.1 & 0.8 \end{bmatrix}$$



مثال ۲

- فرض کنید N گلدان در اختیار داریم که در هر یک توپ‌هایی هم‌رنگ موجود است.
 - قرمز (S_1) ، آبی (S_2) و سبز (S_3)
- q_t رنگ توپی که در زمان t برداشته شده است، را نمایش می‌دهد.

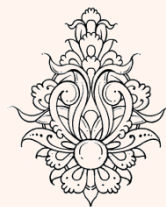
$$\Pi = [0.5, 0.2, 0.3]^T \quad \mathbf{A} = \begin{bmatrix} 0.4 & 0.3 & 0.3 \\ 0.2 & 0.6 & 0.2 \\ 0.1 & 0.1 & 0.8 \end{bmatrix}$$

$$O = \{S_1, S_1, S_3, S_3\}$$

$$P(O|\mathbf{A}, \Pi) = P(S_1) \cdot P(S_1|S_1) \cdot P(S_3|S_1) \cdot P(S_3|S_3)$$

$$= \pi_1 \cdot a_{11} \cdot a_{13} \cdot a_{33}$$

$$= 0.5 \cdot 0.4 \cdot 0.3 \cdot 0.8 = 0.048$$



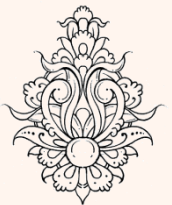
مثال ۲ (ادامه...)

- فرض کنید در این مثال یک سری (K) مشاهده با طول T موجود است، سیستم می‌تواند پارامترهای Π و A را یاد بگیرد. اگر حالت سیستم در زمان t در دنباله k -ام باشد، احتمال حالت اولیه را می‌توان به صورت زیر تقریب زد:

$$\hat{\pi}_i = \frac{\#\{\text{sequences starting with } S_i\}}{\#\{\text{sequences}\}} = \frac{\sum_k 1(q_1^k = S_i)}{K}$$

- و احتمال گذار

$$\hat{a}_{ij} = \frac{\#\{\text{transitions from } S_i \text{ to } S_j\}}{\#\{\text{transitions from } S_i\}} = \frac{\sum_k \sum_{t=1}^{T-1} 1(q_t^k = S_i \text{ and } q_{t+1}^k = S_j)}{\sum_k \sum_{t=1}^{T-1} 1(q_t^k = S_i)}$$



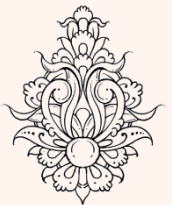
- مدل مارکوف قابل مشاهده برای استفاده‌ی عملی بسیار محدود می‌باشد.
- در «مدل مارکوف پنهان (HMM)»، حالت‌های سیستم را نمی‌توان مشاهده نمود بلکه در هر حالت، خروجی مشاهده شده، احتمال حضور سیستم در یک حالت خاص را با تابعی احتمالاتی بیان می‌کند.
- با فرض این که در حالت‌های مختلف خروجی سیستم از مجموعه‌ی زیر باشد:

$$\{v_1, v_2, \dots, v_M\}$$

- «احتمال مشاهده» به صورت زیر به دست می‌آید:

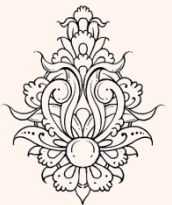
$$b_j(m) \equiv P(O_t = v_m \mid q_t = S_j)$$

Observation (emission) probability



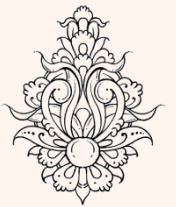
مدل پنهان مارکوف (ادامه...)

- دنباله‌ی حالت‌های سیستم قابل مشاهده نیست.
 - این همان نکته‌ای است که باعث شده است چنین سیستمی **پنهان** نامیده شود.
 - ولی با توجه به دنباله‌ی مشاهدات، می‌توان آن را حدس زد و یا به بیان بهتر احتمال آن را محاسبه نمود.
 - باید توجه داشت که به ازای هر «دنباله‌ی مشاهده» تعداد زیادی دنباله‌ی حالت موجود است که می‌تواند همان دنباله‌ی مشاهده را تولید نماید ولی با احتمال‌های متفاوت.



مدل پنهان مارکوف (ادامه...)

- در مدل پنهان مارکوف علاوه بر حرکت تصادفی بین حالت‌ها، خروجی مشاهده شده هم تصادفی است.
- مدل مارکوف پنهان در واقع نوعی مدل مارکوف تو در تو است.
- بدین ترتیب که مدل مارکوف اصلی انتقال بین حالات را نشان می‌دهد و در هر حالت، مشاهده با توجه به یک مدل مارکوف وابسته به آن حالت انجام می‌شود.
- اولین مشکل تعیین تعداد حالات و تخصیص آن به دنباله‌ی مشاهدات است.

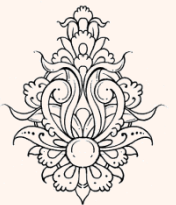


مثال ۳

- فرض کنید شخصی در پس یک مانع یک (یا چند) سکه را پرتاب می‌کند و بدون این که نموده‌ی عملکردش معین باشد، تنها نتیجه‌ی پرتاب را نمایش می‌دهد:

$$\begin{aligned}O &= o_1 o_2 o_3 \dots o_1 \\ &= HHT \dots TTH\end{aligned}$$

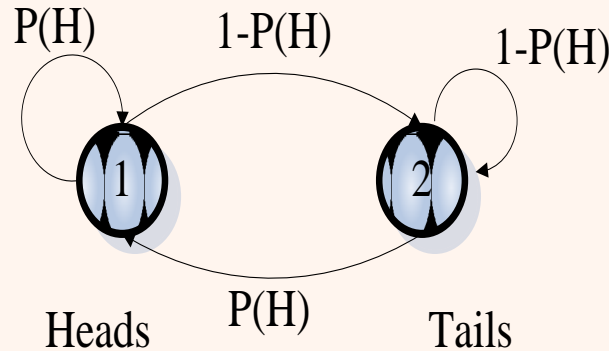
- چگونه می‌توان این فرآیند را با زنجیره‌ی مارکوف مدل کرد؟



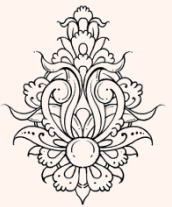
مثال ۳ (ادامه...)



O=HHTTHTHHTTH...
S=1 1 2 2 1 2 1 1 2 2 1 ...



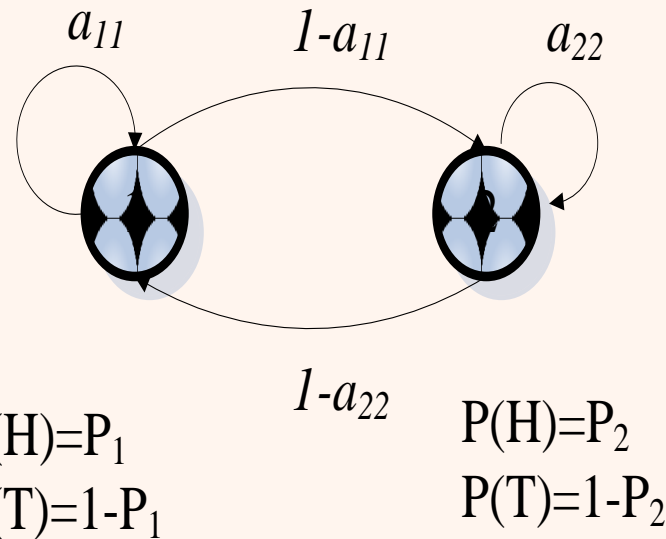
- دو حالت در نظر گرفته و هر حالت بیانگر یک روی سکه باشد.
- تنها پارامتر مجهول $p(H)$ است (البته به جز احتمال اولیه).
- «مدل مارکوف قابل مشاهده» است!



مثال ۳ (ادامه...)

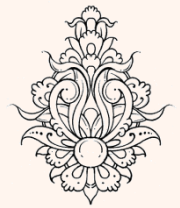
μ

O=HHTTHTHHTTH...
S=2 1 1 2 2 2 1 2 2 1 2 ...



• دو حالت در نظر گرفته و هر حالت بیانگر خروجی یک روی سکه است.

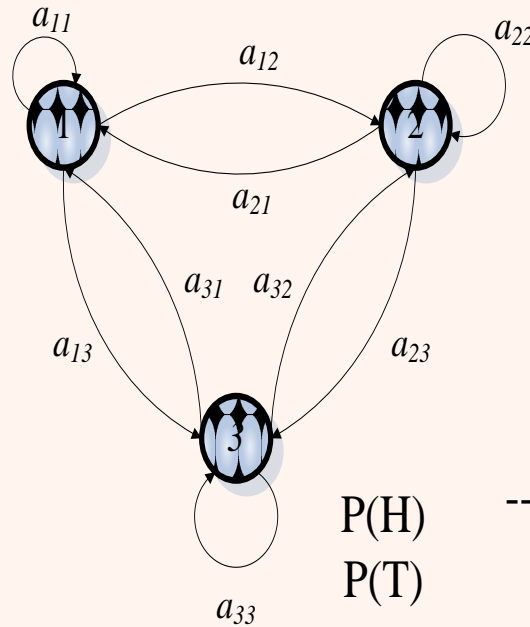
• چهار پارامتر مجهول وجود دارد(البته به جز احتمال اولیه).
• «مدل مارکوف پنهان» است!



مثال ۳ (ادامه...)

۳

O=HHTTHTHHTTH...
S=3 1 2 3 3 1 1 2 3 1 3 ...

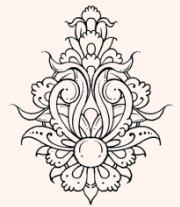


P(H)
P(T)

	State		
	1	2	3
P(H)	P_1	P_2	P_3
P(T)	$1-P_1$	$1-P_2$	$1-P_3$

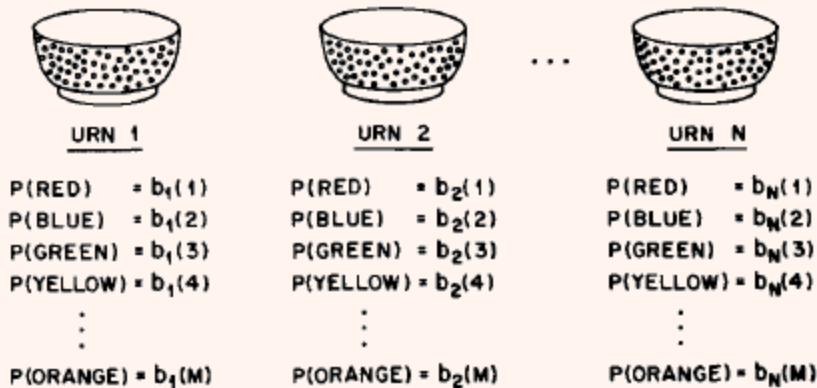
نه پارامتر مجهول وجود دارد(البته به جز احتمال اولیه).

هر چه مدل بزرگتر شود قابلیت مدل کردن آن بهتر خواهد بود.
اما در عین حال ممکن است مشکل **overfitting** رخ دهد.

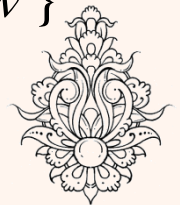


مثال ۴

- در مثال توپ و گلدان، مدل مارکوف پنهان معادل حالتی است که در هر گلدان توپ‌هایی با رنگ‌های متفاوت داشته باشیم.
- در اینجا $b_j(m)$ معادل فارچ کردن توپی با رنگ m از گلدان j می‌باشد.
- این بار نیز دنباله‌ای از رنگ‌ها موجود است با این تفاوت که نمی‌دانیم که توپ‌ها متعلق به کدام گلدان هستند.



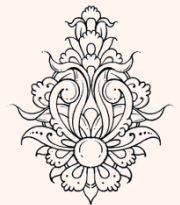
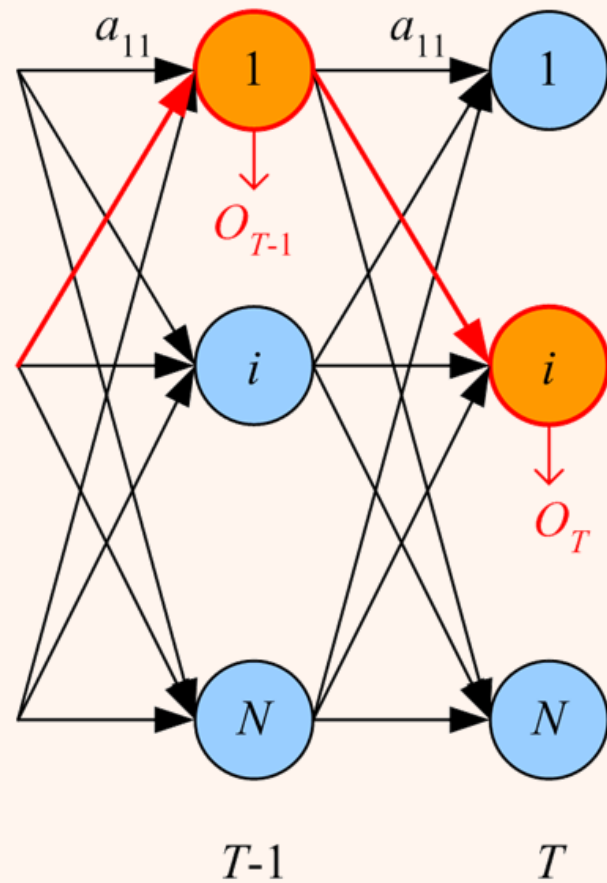
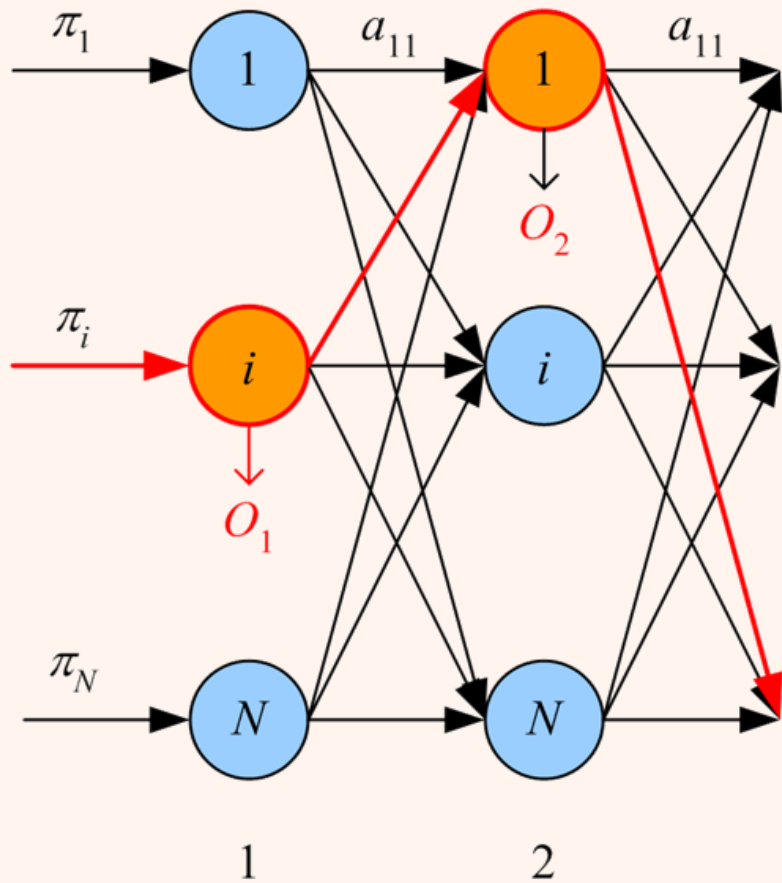
$$o = \{red, red, green, blue, yellow\}$$



$$o = \{GREEN, GREEN, BLUE, RED, YELLOW, RED, \dots, BLUE\}$$

Fig. 3. An N -state urn and ball model which illustrates the general case of a discrete symbol HMM.

HMM Unfolded in Time



مؤلفه‌های مدل پنهان مارکوف

$$S = \{S_1, S_2, \dots, S_N\}$$

• تعداد حالت‌ها: (N)

$$V = \{v_1, v_2, \dots, v_M\}$$

• تعداد نمادهای قابل مشاهده: (M)

$$A = [a_{ij}] \text{ where } a_{ij} \equiv P(q_{t+1} = S_j | q_t = S_i)$$

• احتمال گذار

• احتمال مشاهده

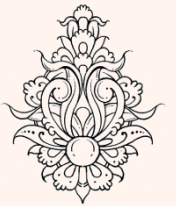
$$B = [b_j(m)] \text{ where } b_j(m) = P(O_t = v_m | q_t = S_j)$$

• احتمالات حالات اولیه:

$$\Pi = [\pi_i] \text{ where } \pi_i \equiv P(q_1 = S_i)$$

بنابراین یک مدل مارکوف پنهان را می‌توان به صورت
زیر نشان داد:

$$\lambda = (A, B, \Pi)$$



سه مسأله‌ی پایه مرتب با HMM

ارزیابی: تعیین احتمال رخداد یک دنباله از مشاهدات

$$P(O | \lambda) = ?$$

Evaluation

μ

State sequence

تعیین محتمل‌ترین حالت یک دنباله از مشاهدات

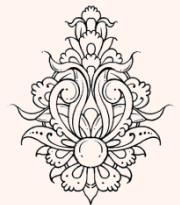
$$P(Q^* | O, \lambda) = \max_Q P(Q | O, \lambda)$$

Learning

یادگیری مدل

ν

Given $X = \{O^k\}_k$, find λ^* such that
$$P(X | \lambda^*) = \max_{\lambda} P(X | \lambda)$$



$$P(O | \lambda) = ?$$

مسئله ارزیابی

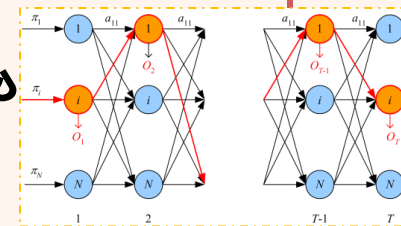
• با این فرض که دنباله‌ی زیر مشاهده شده است:

$$O = \{o_1 o_2 \dots o_T\}$$

– اگر دنباله‌ی حالت‌ها، مشخص باشد:

$$Q = \{q_1 q_2 \dots q_T\}$$

– احتمال رخداد این دنباله از مشاهدات در حالت‌های مشخص شده به ترتیب زیر به دست می‌آید:



$$P(O | Q, \lambda) = \prod_{t=1}^T P(o_t | q_t, \lambda) = b_{q_1}(o_1) b_{q_2}(o_2) \dots b_{q_T}(o_T)$$



تنها اطلاعاتی که در این حالت داریم، خروجی سیستم است و هیچ اطلاعاتی از حالات سیستم نداریم



$$P(O | \lambda) = ?$$

مسئله ارزیابی

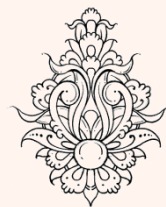
$$P(O | Q, \lambda) = \prod_{t=1}^T P(o_t | q_t, \lambda) = b_{q_1}(o_1) b_{q_2}(o_2) \cdots b_{q_T}(o_T)$$

• احتمال رخداد دنباله حالات سیستم:

$$P(Q | \lambda) = p(q_1) \prod_{t=2}^T P(q_t | q_{t-1}) = \pi_{q_1} a_{q_1 q_2} \cdots a_{q_{T-1} q_T}$$

• در نتیجه:

$$\begin{aligned} P(O, Q | \lambda) &= p(q_1) \prod_{t=2}^T P(q_t | q_{t-1}) \prod_{t=1}^T P(o_t | q_t, \lambda) \\ &= \pi_{q_1} b_{q_1}(o_1) a_{q_1 q_2} b_{q_2}(o_2) a_{q_2 q_3} \cdots a_{q_{T-1} q_T} b_{q_T}(o_T) \end{aligned}$$



$$P(O | \lambda) = ?$$

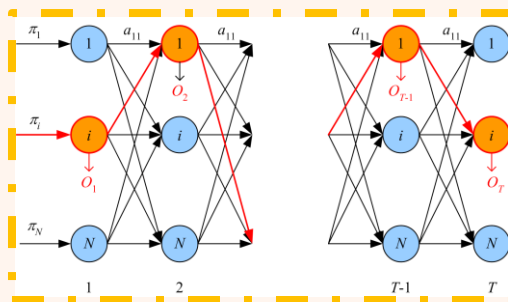
مسئله ارزیابی

$$P(O, Q | \lambda) = \pi_{q_1} b_{q_1}(o_1) a_{q_1 q_2} b_{q_2}(o_2) a_{q_2 q_3} \dots a_{q_{T-1} q_T} b_{q_T}(o_T)$$

- با در اختیار داشتن رابطه‌ی فوق و محاسبه‌ی تابع احتمال ماشیه‌ای O خواهیم داشت:

$$P(O | \lambda) = \sum_{\text{all possible } Q} P(O, Q | \lambda) \quad \mathbf{O(2TN^T)}$$

- که البته این شیوه **عملی** نیست!
- چرا که N^T شیوه‌ی مختلف برای حالت‌ها وجود دارد.



مسئله ارزیابی

$$P(O|\lambda)=?$$

Forward backward Procedure

Forward variable

$$\alpha_t(i) \equiv P(O_1 \cdots O_t, q_t = S_i | \lambda)$$

$$\beta_t(i) \equiv P(O_{t+1} \cdots O_T | q_t = S_i, \lambda)$$

Backward variable

1) Initialization:

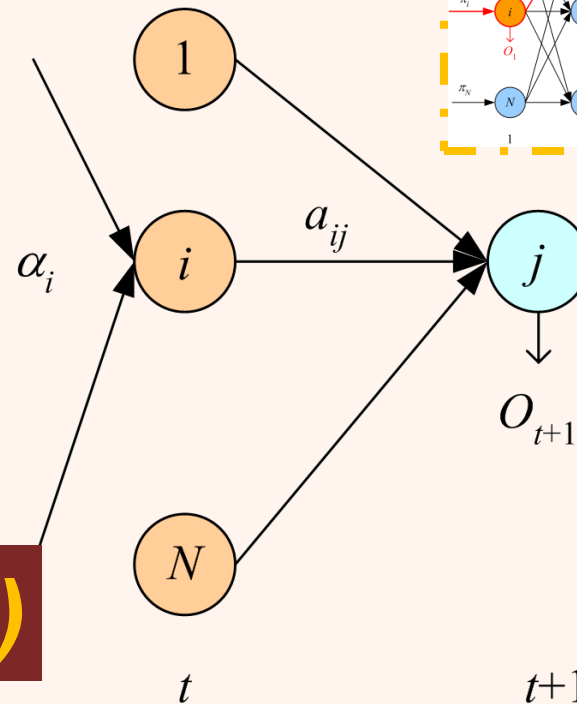
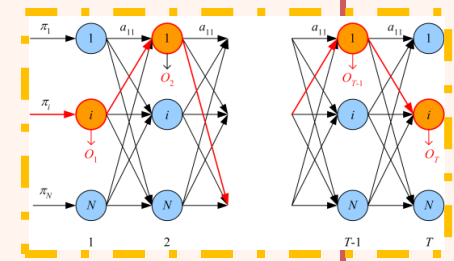
$$\alpha_1(i) = \pi_i b_i(O_1)$$

2) Induction (Recursion):

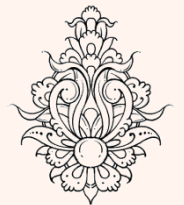
$$\alpha_{t+1}(j) = \left[\sum_{i=1}^N \alpha_t(i) a_{ij} \right] b_j(O_{t+1})$$

3) Termination:

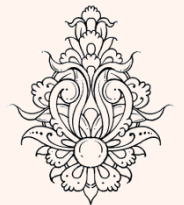
$$P(O|\lambda) = \sum_{i=1}^N \alpha_T(i)$$



$O(N^2T)$

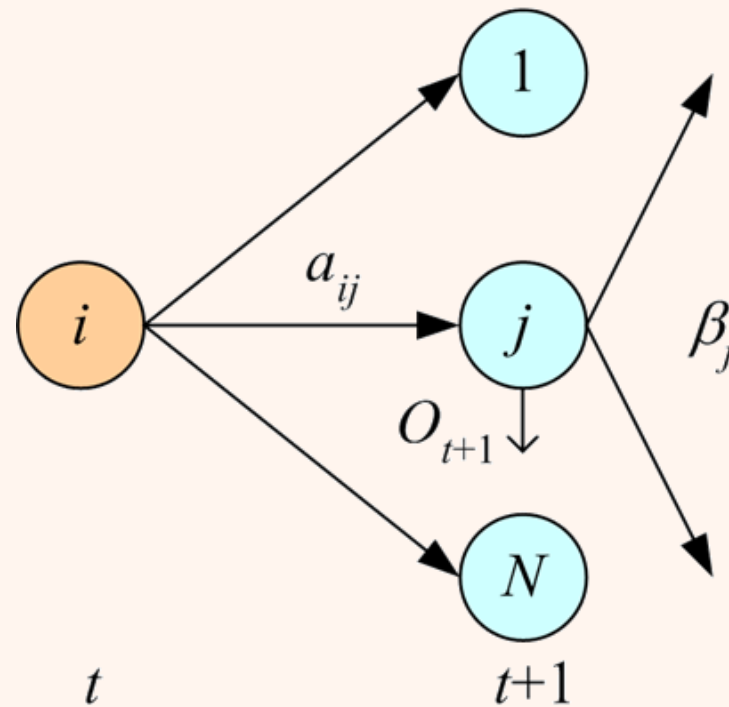


$$\begin{aligned}
 \alpha_{t+1}(j) &= P(o_1 \dots o_{t+1}, q_{t+1} = S_j | \lambda) \\
 &= P(o_1 \dots o_{t+1} | q_{t+1} = S_j, \lambda) P(q_{t+1} = S_j | \lambda) \\
 &= P(o_1 \dots o_t | q_{t+1} = S_j, \lambda) P(o_{t+1} | q_{t+1} = S_j, \lambda) P(q_{t+1} = S_j | \lambda) \\
 &= P(o_1 \dots o_t, q_{t+1} = S_j | \lambda) P(o_{t+1} | q_{t+1} = S_j, \lambda) \\
 &= P(o_{t+1} | q_{t+1} = S_j, \lambda) \sum_i P(o_1 \dots o_t, q_t = S_i, q_{t+1} = S_j | \lambda) \\
 &= P(o_{t+1} | q_{t+1} = S_j, \lambda) \\
 &\quad \sum_i P(o_1 \dots o_t, q_{t+1} = S_j | q_t = S_i, \lambda) P(q_t = S_i | \lambda) \\
 &= P(o_{t+1} | q_{t+1} = S_j, \lambda) \\
 &\quad \sum_i P(o_1 \dots o_t | q_t = S_i, \lambda) P(q_{t+1} = S_j | q_t = S_i, \lambda) P(q_t = S_i | \lambda) \\
 &= P(o_{t+1} | q_{t+1} = S_j, \lambda) \\
 &\quad \sum_i P(o_1 \dots o_t, q_t = S_i | \lambda) P(q_{t+1} = S_j | q_t = S_i, \lambda) \\
 &= P(o_{t+1} | q_{t+1} = S_j, \lambda) \sum_i \alpha_t(i) P(q_{t+1} = S_j | q_t = S_i, \lambda) \\
 &= \left[\sum_{i=1}^N \alpha_t(i) a_{ij} \right] b_j(O_{t+1})
 \end{aligned}$$



Backward variable

$$\beta_t(i) \equiv P(O_{t+1} \cdots O_T | q_t = S_i, \lambda)$$

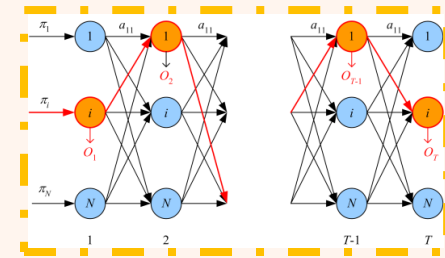


1) Initialization:

$$\beta_T(i) = 1$$

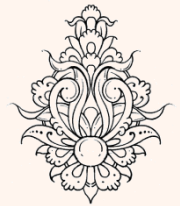
2) Recursion:

$$\beta_t(i) = \sum_{j=1}^N a_{ij} b_j(O_{t+1}) \beta_{t+1}(j)$$

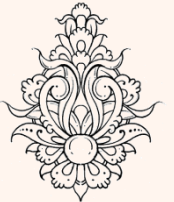


نکته: هر دو متغیر مرسوم ضرب اعداد بسیار کوچکی هستند و امکان پاریز آن‌ها وجود دارد، برای پرهیز از این مشکل توصیه می‌شود در هر مرحله نتایج نرمال شوند.

$$c_t = \sum_j \alpha_t(j)$$



$$\begin{aligned}\beta_t(i) &\equiv P(o_{t+1} \dots o_T \mid q_t = S_i, \lambda) \\ &= \sum_j P(o_{t+1} \dots o_T, q_{t+1} = S_j \mid q_t = S_i, \lambda) \\ &= \sum_j P(o_{t+1} \dots o_T \mid q_{t+1} = S_j, q_t = S_i, \lambda) P(q_{t+1} = S_j \mid q_t = S_i, \lambda) \\ &= \sum_j P(o_{t+1} \mid q_{t+1} = S_j, q_t = S_i, \lambda) \\ &\quad P(o_{t+2} \dots o_T \mid q_{t+1} = S_j, q_t = S_i, \lambda) P(q_{t+1} = S_j \mid q_t = S_i, \lambda) \\ &= \sum_j P(o_{t+1} \mid q_{t+1} = S_j, \lambda) \\ &\quad P(o_{t+2} \dots o_T \mid q_{t+1} = S_j, \lambda) P(q_{t+1} = S_j \mid q_t = S_i, \lambda) \\ &= \sum_{j=1}^N a_{ij} b_j(o_{t+1}) \beta_{t+1}(j)\end{aligned}$$



$$P(Q^* | O, \lambda) = \max_Q P(Q | O, \lambda)$$

یافتن دنباله‌ی حالات

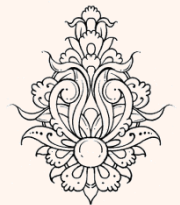
- با در اختیار داشتن خصوصیات یک مدل مارکوف پنهان λ و یک دنباله از مشاهدات،

$$O = \{o_1 o_2 \dots o_T\}$$

- در پی دنباله‌ای از حالت‌ها هستیم که با بیشترین احتمال دنباله‌ی مشاهدات مورد نظر را تولید کند:

$$Q = \{q_1 q_2 \dots q_T\}$$

- یک راه محاسبه‌ی تمام حالات ممکن و انتخاب مسیر با بیشترین احتمال است!



$$P(Q^* | O, \lambda) = \max_Q P(Q | O, \lambda)$$

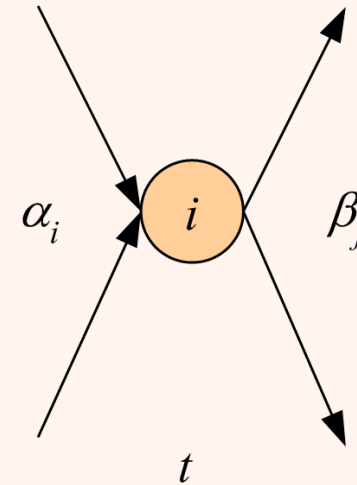
یافتن دنباله‌ی حالات

$$\gamma_t(i) \equiv P(q_t = S_i | O, \lambda)$$

$$= \frac{P(O | q_t = S_i, \lambda) P(q_t = S_i | \lambda)}{P(O | \lambda)}$$

⋮

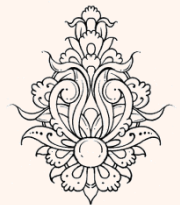
$$= \frac{\alpha_t(i) \beta_t(i)}{\sum_{j=1}^N \alpha_t(j) \beta_t(j)}$$



در هر گام (t) حالتی انتخاب می‌شود که بیشترین احتمال را داشته باشد



$$q_t^* = \arg \max_i \gamma_t(i)$$



$$P(Q^* | O, \lambda) = \max_Q P(Q | O, \lambda)$$

ریز محاسبات

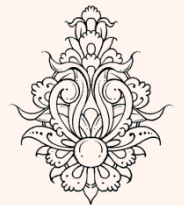
$$\gamma_t(i) \equiv P(q_t = S_i | O, \lambda) \quad \text{Bayse theorem}$$

$$= \frac{P(O | q_t = S_i, \lambda)P(q_t = S_i | \lambda)}{P(O | \lambda)}$$

$$= \frac{P(o_1 \dots o_t | q_t = S_i, \lambda)P(o_{t+1} \dots o_T | q_t = S_i, \lambda)P(q_t = S_i | \lambda)}{\sum_{j=1}^N P(O, q_t = S_j | \lambda)}$$

$$= \frac{P(o_1 \dots o_t, q_t = S_i | \lambda)P(o_{t+1} \dots o_T | q_t = S_i, \lambda)}{\sum_{j=1}^N P(O | q_t = S_j, \lambda)P(q_t = S_j)}$$

$$= \frac{\alpha_t(i)\beta_t(i)}{\sum_{j=1}^N \alpha_t(j)\beta_t(j)}$$



Viterbi's Algorithm

$$P(Q^* | O, \lambda) = \max_Q P(Q | O, \lambda)$$

$$\delta_t(i) \equiv \max_{q_1 q_2 \dots q_{t-1}} P(q_1 q_2 \dots q_{t-1}, q_t = S_i, o_1 \dots o_t | \lambda)$$

1) initialization :

مقداردهی اولیه

$$\delta_1(i) = \pi_i b_i(o_1)$$

$$\psi_1(i) = 0$$

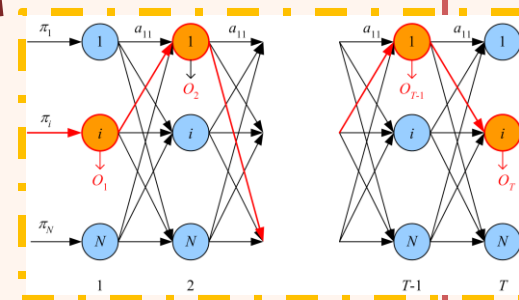
فایده

2) induction :

بازگشتی

$$\delta_t(j) = \max_i [\delta_{t-1}(i) a_{ij}] \cdot b_j(o_t)$$

$$\psi_t(i) = \arg \max_i [\delta_{t-1}(i) a_{ij}]$$



3) Termination :

$$p^* = \max_i \delta_T(i)$$

$$q_T^* = \arg \max_i \delta_T(i)$$

برگشت مسیر: (دنباله‌ی حالت‌ها)

4) Path backtracking :

$$q_t^* = \psi_{t+1}(q_{t+1}^*), \quad t = T-1, T-2, \dots, 1$$



تأسیسات
بهبودی

Given $X = \{O^k\}_k$ find λ^* such that
 $P(X | \lambda^*) = \max_{\lambda} P(X | \lambda)$

یادگیری

Learning

• هدف این است که با در اختیار داشتن یک

مجموعه‌ای آموزشی از مشاهدات $X = \{O^k\}_{k=1}^K$

پارامترهای مدل $\lambda^* = (A, B, \Pi)$ به گونه‌ای برآورد

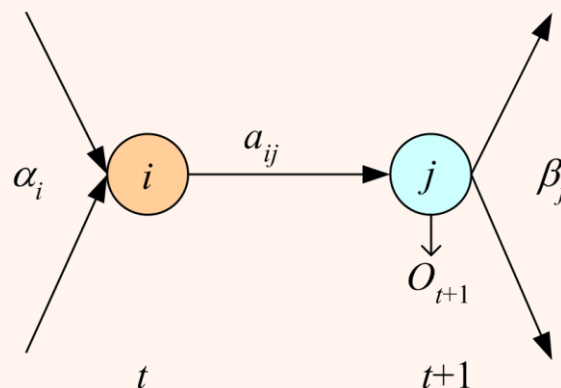
شوند که تابع درست‌نمایی $P(X | \lambda^*)$ بیشینه شود.

• راه حل تحلیلی برای این مساله وجود ندارد.

– از یک فرآیند تکرار شونده استفاده می‌شود:

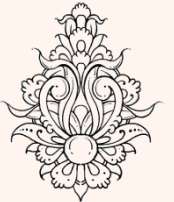
Baum-Welch algorithm

$$\xi_t(i, j) \equiv P(q_t = S_i, q_{t+1} = S_j | O, \lambda)$$



ریز محاسبات

$$\begin{aligned}\xi_t(i, j) &\equiv P(q_t = S_i, q_{t+1} = S_j | O, \lambda) \\ &= \frac{P(O | q_t = S_i, q_{t+1} = S_j, \lambda) P(q_t = S_i, q_{t+1} = S_j | \lambda)}{P(O | \lambda)} \\ &= \frac{P(O | q_t = S_i, q_{t+1} = S_j, \lambda) P(q_{t+1} = S_j | q_t = S_i, \lambda) P(q_t = S_i | \lambda)}{P(O | \lambda)} \\ &= \frac{1}{P(O | \lambda)} P(o_1 \dots o_t | q_t = S_i, \lambda) P(o_{t+1} | q_{t+1} = S_j, \lambda) \\ &\quad P(o_{t+2} \dots o_T | q_{t+1} = S_j, \lambda) a_{ij} P(q_t = S_i | \lambda) \\ &= \frac{1}{P(O | \lambda)} P(o_1 \dots o_t, q_t = S_i | \lambda) P(o_{t+1} | q_{t+1} = S_j, \lambda) \\ &\quad P(o_{t+2} \dots o_T | q_{t+1} = S_j, \lambda) a_{ij} \\ &= \frac{\alpha_t(i) b_j(o_{t+1}) \beta_{t+1}(j) a_{ij}}{\sum_k \sum_l \alpha_t(k) a_{kl} b_l(o_{t+1}) \beta_{t+1}(l)} \\ &= \frac{\alpha_t(i) a_{ij} b_j(o_{t+1}) \beta_{t+1}(j)}{\sum_k \sum_l \alpha_t(k) a_{kl} b_l(o_{t+1}) \beta_{t+1}(l)}\end{aligned}$$



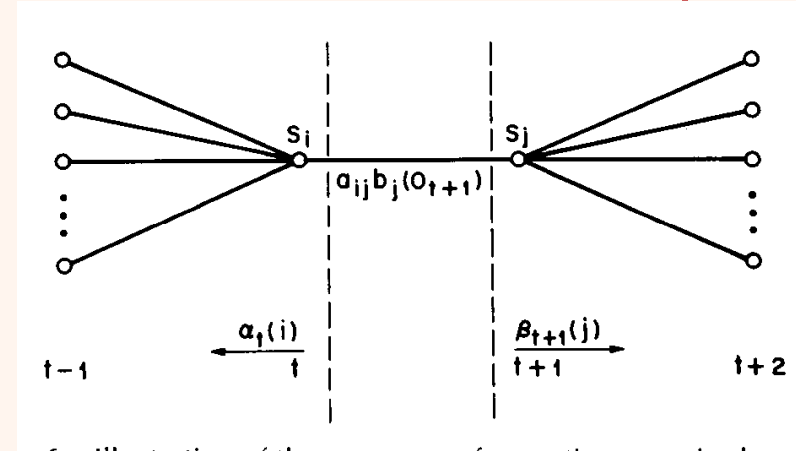
Given $X=\{O^k\}_k$ find λ^* such that
 $P(X|\lambda^*)=\max_{\lambda} P(X|\lambda)$

یادگیری (ادامه...)

Baum-Welch algorithm

$$\xi_t(i, j) \equiv P(q_t = S_i, q_{t+1} = S_j | O, \lambda)$$

$$= \frac{\alpha_t(i) a_{ij} b_j(O_{t+1}) \beta_{t+1}(j)}{\sum_k \sum_l \alpha_t(k) a_{kl} b_l(O_{t+1}) \beta_{t+1}(l)}$$



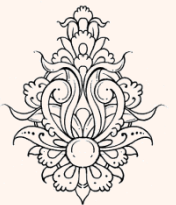
• می‌توان احتمال حضور در یک حالت را محاسبه

کرد:

$$\gamma_t(i) \equiv P(q_t = S_i | O, \lambda) \quad \gamma_t(i) = \sum_{j=1}^N \xi_t(i, j)$$

• در صورتی که مدل مارکوف قابل مشاهده باشد،

- هر کدام از مقادیر γ و ξ صفر و یک خواهند بود.



Given $X=\{O^k\}_K$ find λ^* such that
 $P(X|\lambda^*)=\max_{\lambda} P(X|\lambda)$

یادگیری (ادامه...)

$$\hat{\pi}_i = \frac{\#\{\text{sequences starting with } S_i\}}{\#\{\text{sequences}\}} = \frac{\sum_k 1(q_1^k = S_i)}{K}$$

$$\gamma_t(i) \equiv P(q_t = S_i | O, \lambda)$$

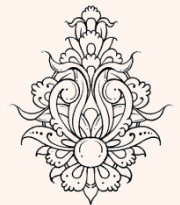
$$\begin{aligned} \hat{a}_{ij} &= \frac{\#\{\text{transitions from } S_i \text{ to } S_j\}}{\#\{\text{transitions from } S_i\}} \\ &= \frac{\sum_k \sum_{t=1}^{T-1} 1(q_t^k = S_i \text{ and } q_{t+1}^k = S_j)}{\sum_k \sum_{t=1}^{T-1} 1(q_t^k = S_i)} \end{aligned}$$

$$\xi_t(i, j) \equiv P(q_t = S_i, q_{t+1} = S_j | O, \lambda)$$

Baum-Welch algorithm(EM)

$$z_i^t = \begin{cases} 1 & \text{if } q_t = S_i \\ 0 & \text{otherwise} \end{cases} \quad z_{ij}^t = \begin{cases} 1 & \text{if } q_t = S_i \text{ and } q_{t+1} = S_j \\ 0 & \text{otherwise} \end{cases}$$

$$E[z_i^t] = \gamma_t(i) \quad E[z_{ij}^t] = \xi_t(i, j)$$



Baum-Welch algorithm

E-Step

• در گام ۱، با پارامترهای با مقدار فعلی پارامترهای

M-Step

مدل مقادیر μ و ξ تخمین زده می‌شوند.

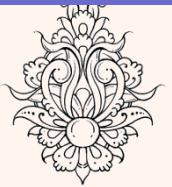
• بر اساس تخمین زده شده، پارامترهای مدل به روز می‌شوند.

$$\hat{\pi}_i = \frac{\sum_{k=1}^K \gamma_1^k(i)}{K}$$

$$\hat{a}_{ij} = \frac{\sum_{k=1}^K \sum_{t=1}^{T_k-1} \xi_t^k(i, j)}{\sum_{k=1}^K \sum_{t=1}^{T_k-1} \gamma_t^k(i)}$$

$$\hat{b}_j(m) = \frac{\sum_{k=1}^K \sum_{t=1}^{T_k} \gamma_t^k(j) 1(O_t^k = v_m)}{\sum_{k=1}^K \sum_{t=1}^{T_k} \gamma_t^k(i)}$$

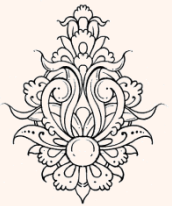
Soft count



این روند تا همگرایی ادامه خواهد یافت، ثابت شده است که $p(O|\lambda)$ نزولی خواهد بود.

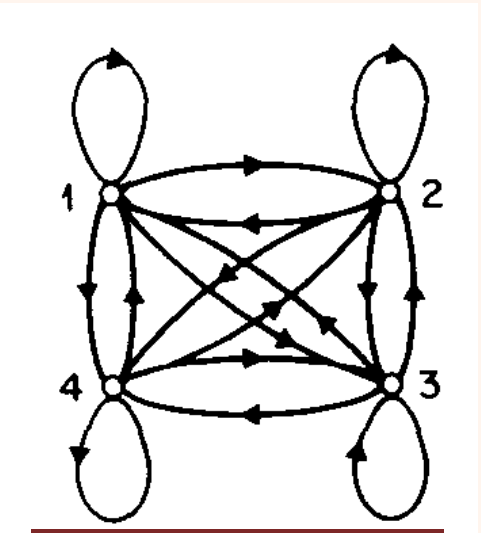
یادگیری - چند نکته

- این الگوریتم، ماکزیمم محلی را می‌یابد و در عمل رویه‌ی هدف (maximization surface) شکل پیچیده‌ای دارد و دارای تعداد زیادی ماکزیمم محلی است.
- نظر به این که کلیت مسأله‌ی آموزش به نوعی یک مسأله‌ی بهینه‌سازی است و از تکنیک‌های نظیر نزول گرادیان برای حل این مسأله می‌توان بهره جست.

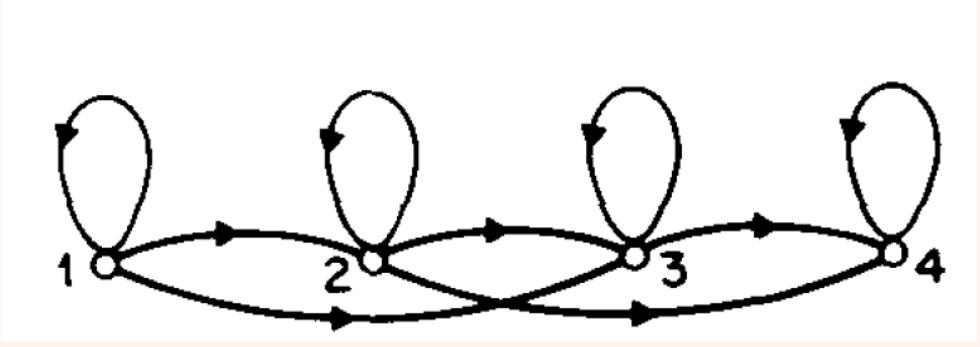


Model Selection in HMM

- در برخی کاربردها مانند تشخیص گفتار استفاده از مدل‌های خاصی توصیه می‌شود.



Ergodic model



Left to right HMMs(Bakis Model)

$$a_{ij} = 0 \quad i < j \quad \pi_i = \begin{cases} 0, & i \neq 1 \\ 1, & i = 1 \end{cases}$$

$$a_{ij} = 0 \quad j < i + \Delta$$

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} & 0 \\ 0 & a_{22} & a_{23} & a_{24} \\ 0 & 0 & a_{33} & a_{34} \\ 0 & 0 & 0 & a_{44} \end{bmatrix}$$

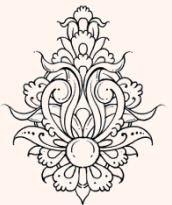


دسته بندی

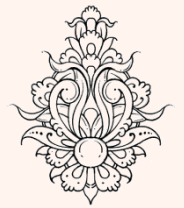
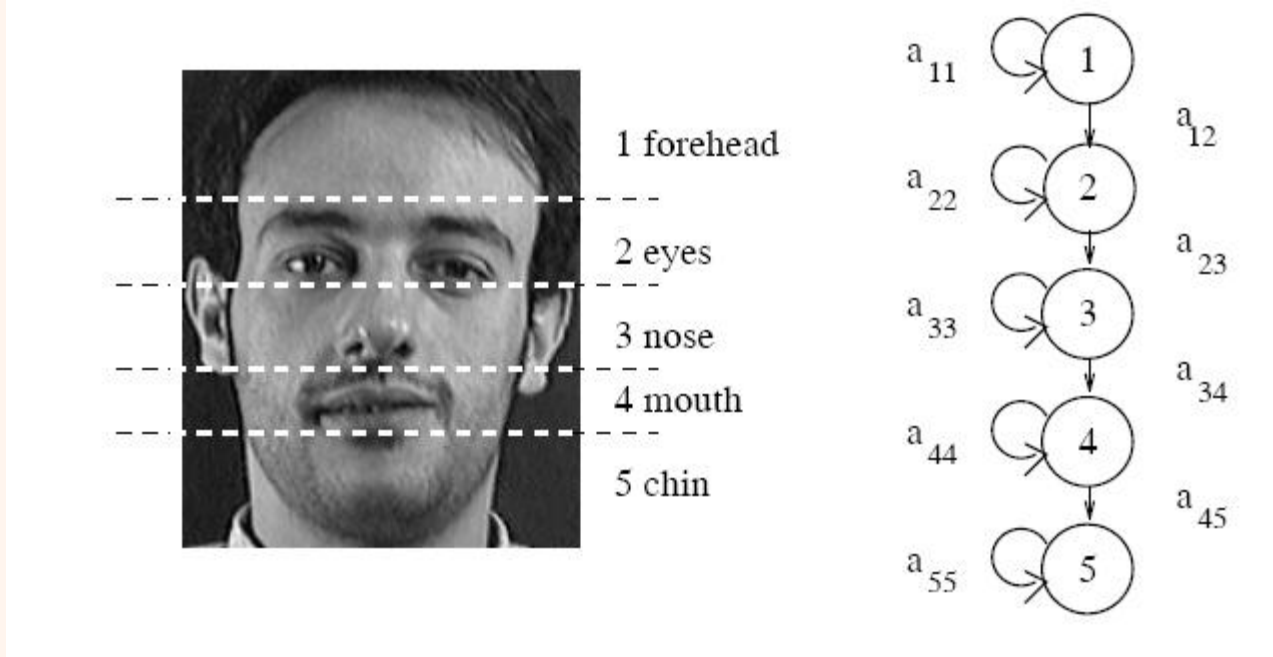
- یک مجموعه از HMM ها خواهیم داشت که هر یک، دنباله های مربوط به یک دسته را مدل می کنند.
- مثلاً در بازشناسی کلمات ادا شده به ازای هر کلمه، یک HMM جداگانه آموزش داده می شود.
- با ارائه ی یک کلمه ی جدید برای شناسایی، تمام مدل های موجود مورد ارزیابی قرار می گیرند و مقدار محاسبه می شود. سپس با استفاده از قانون بیز خواهیم داشت:

$$P(\lambda_i | O) = \frac{P(O | \lambda_i) P(\lambda_i)}{\sum_j P(O | \lambda_j) P(\lambda_j)}$$

- مدلی که برای بیشترین احتمال $P(\lambda_i | O)$ باشد به عنوان دسته ی شناسایی شده معرفی می گردد.



مثال - شناسایی چهره



Ferdinando Silvestro Samaria, "Face Recognition Using Hidden Markov Model", 1994

سایر منابع

- Rabiner, L. R. (1989). "A tutorial on hidden Markov models and selected applications in speech recognition." Proceedings of the IEEE 77(2): 257-286.

