A tutorial of Hidden Markov Model (HMM)

Nov. 2015

Qiuqiang Kong

q.kong@qmul.ac.uk

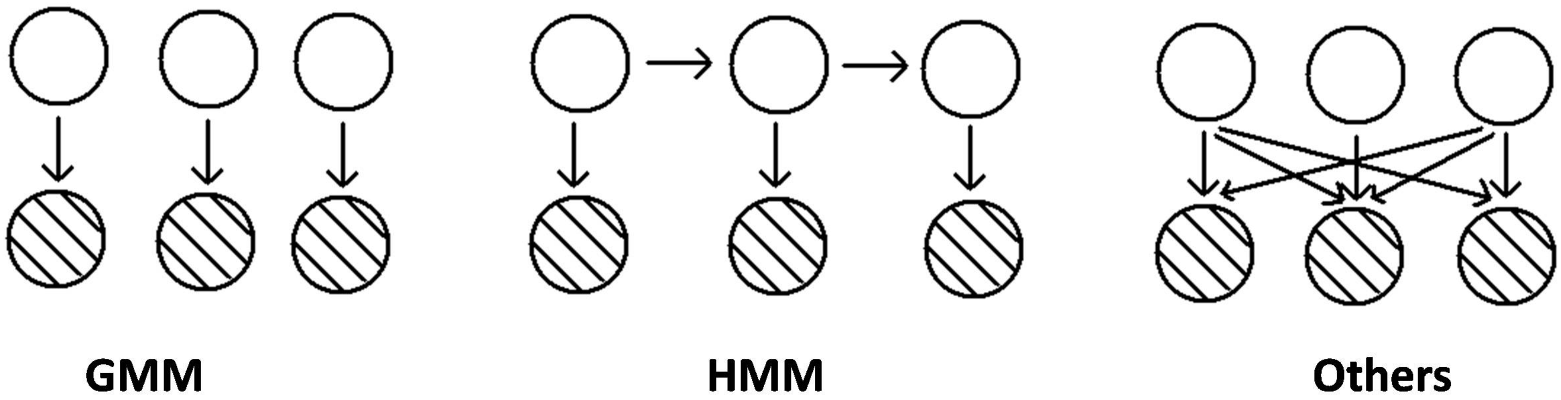
Application of HMM

- Stock price analysis
- Auto speech recognition
- Character recognition
- Sequence alignment
- etc.

Probabilistic Graphical Model (PGM)

PGM is a probabilistic model. The graph expresses the conditional dependence structure between random variables.

Example:



White circle represents latent variable.
Solid circle represents observed variable.
All of GMM, HMM, Others belongs to PGM.

EM algorithm for PGM

EM algorithm is used to estimate parameters in probabilistic graphic model (PGM) with latent variables*.

EM algorithm for PGM with latent variables

1. Init parameters

2. E step: q(**Z**)=p(**Z**|**X**θ^{old})

3. M step: $\hat{\theta} = \arg\max_{\theta} Q(\theta, \theta^{\text{old}})$

   where $Q(\theta, \theta^{\text{old}}) = \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}\theta^{\text{old}}) \ln p(\mathbf{X}\mathbf{Z}|\theta)$

4. If converge then stop, otherwise goto 2

* More details can be seen in GMM Tutorial.

Generally, estimation of p($\mathbf{Z}$|$\mathbf{X}\theta^{old}$) is difficult.
However, the independence of PGM will simplify the model.

D-separation property[1]
Let A, B, C be set of nodes. We can check whether A and B are conditional independent on C in belowing way.

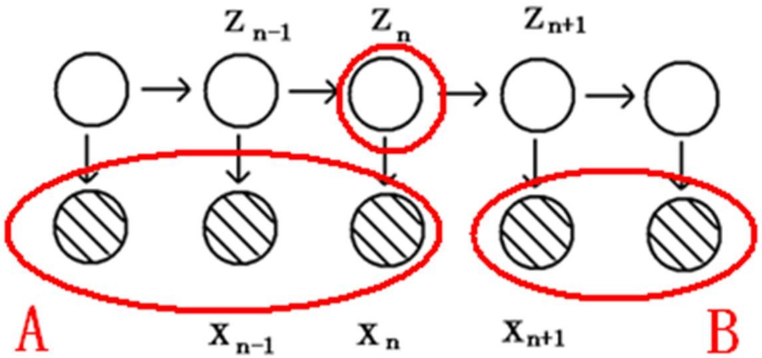
Check all the paths from node in A to node C. The path is said to be blocked if

(a) ∃ head-to-tail or tail-to-tail nodes on the path, and the node is in C.
(b) ∀ head-to-head nodes, neither the node, nor any of its descendants is in C

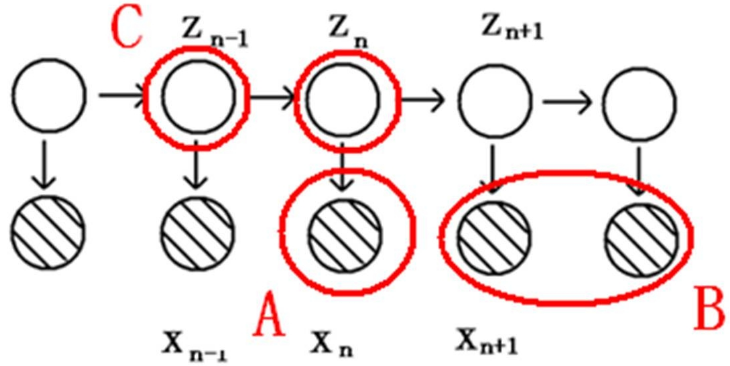
If all paths are blocked, then A is said to be d-separated from B by C.
That is P(AB|C) = P(A|C)P(B|C).

[1] Bishop, Christopher M. *Pattern recognition and machine learning*. springer, 2006. Chap. 8.

Exercises



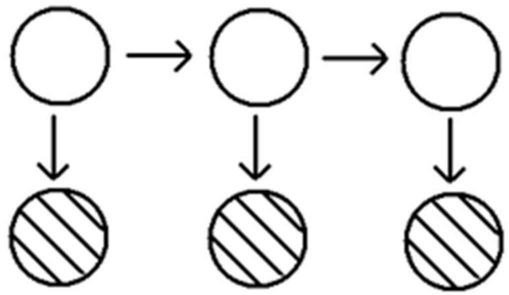
$$p(AB|\mathbf{z}_n) = p(A|\mathbf{z}_n)p(B|\mathbf{z}_n)$$



$$p(ABC|\mathbf{z}_n) = p(A|\mathbf{z}_n)p(B|\mathbf{z}_n)P(C|\mathbf{z}_n)$$

HMM model

Hidden Markov Model (HMM) is a kind of PGM with latent variables.



HMM

HMM's **joint probability distribution** over both latent and observed variables is

$$p(\mathbf{XZ}|\theta) = p(\mathbf{z}_1|\boldsymbol{\pi})\prod_{n=2}^{N} p(\mathbf{z}_n|\mathbf{z}_{n-1},\mathbf{A})\prod_{n=1}^{N} p(\mathbf{x}_n|\mathbf{z}_n,\phi) \qquad (1)$$

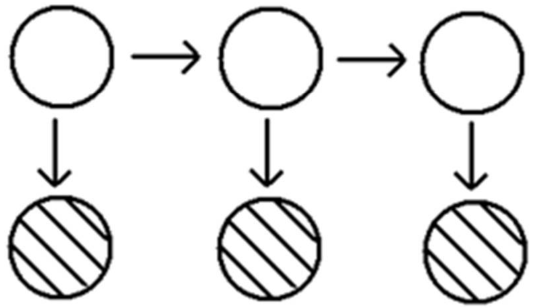
where the parameters are $\theta = \{\boldsymbol{\pi}, \mathbf{A}, \phi\}$

π is the start probability of each state.
A is the transition matrix.
ϕ is the parameter associated with emission distribution (can be multinominal, Gauss, GMM, etc.)

Three basic problem of HMM

- How to train a HMM model?
- Given a sequence X, how to get the likelihood p(X) from HMM model?
- How to find the best decoding path of HMM model?

How to train HMM model?



HMM is a kind of probabilistic graphic model (PGM) with latent variables.

Apply EM algorithm to HMM, $Q(\theta,\theta^{old})$ is

$$
\begin{aligned}
Q(\theta,\theta^{\text{old}}) &= \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}\theta^{\text{old}}) \ln p(\mathbf{X}\mathbf{Z}|\theta) \\
&= \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}\theta^{\text{old}}) \left[\ln p(\mathbf{z}_1) + \sum_{n=2}^{N} \ln p(\mathbf{z}_n|\mathbf{z}_{n-1}) + \sum_{n=1}^{N} \ln p(\mathbf{x}_n|\mathbf{z}_n) \right] \\
&= \sum_{\mathbf{z}_1} p(\mathbf{z}_1|\mathbf{X}\theta^{\text{old}}) \ln p(\mathbf{z}_1) + \sum_{n=2}^{N} \sum_{\mathbf{z}_{n-1}\mathbf{z}_n} p(\mathbf{z}_{n-1}\mathbf{z}_n|\mathbf{X}\theta^{\text{old}}) \ln p(\mathbf{z}_n|\mathbf{z}_{n-1}) + \sum_{n=1}^{N} \sum_{\mathbf{z}_n} p(\mathbf{z}_n|\mathbf{X}\theta^{\text{old}}) \ln p(\mathbf{x}_n|\mathbf{z}_n) \\
&= \underbrace{\sum_{k=1}^{K} \gamma(z_{nk}) \ln \pi_k}_{①} + \underbrace{\sum_{n=2}^{N} \sum_{j=1}^{K} \sum_{k=1}^{K} \xi(z_{n-1,j} z_{n,k}) \ln A_{jk}}_{②} + \underbrace{\sum_{n=1}^{N} \sum_{k=1}^{K} \gamma(z_{nk}) \ln p(\mathbf{x}_n|z_{nk}\phi)}_{③} \qquad (2)
\end{aligned}
$$

Where $\gamma(z_{nk}) = p(z_{nk}|\mathbf{X}\theta^{\text{old}})$ (3)

$$\xi(z_{n-1,j} z_{n,k}) = p(z_{n-1,j} z_{n,k}|\mathbf{X}\theta^{\text{old}}) \qquad (4)$$

π is only associated with ①

A is only associated with ②

ϕ is only associated with ③

E step

Evaluate $p(\mathbf{Z}|\mathbf{X}\theta^{old})$.
From the form of $Q(\theta, \theta^{old})$ of HMM, there is no need to calculate $p(\mathbf{Z}|\mathbf{X}\theta^{old})$ for all **Z**. Many terms vanish. So just need to calculate

$$\gamma(z_{nk}) = p(z_{nk} \,|\mathbf{X}\theta^{\text{old}})$$

$$\xi(z_{n-1,j} z_{n,k}) = p(z_{n-1,j} z_{n,k} \,|\mathbf{X}\theta^{\text{old}})$$

Generally, $p(\mathbf{Z}|\mathbf{X}\theta^{old})$ is difficult to estimate in PGM. While using dependency of HMM, we can simplify the computation.

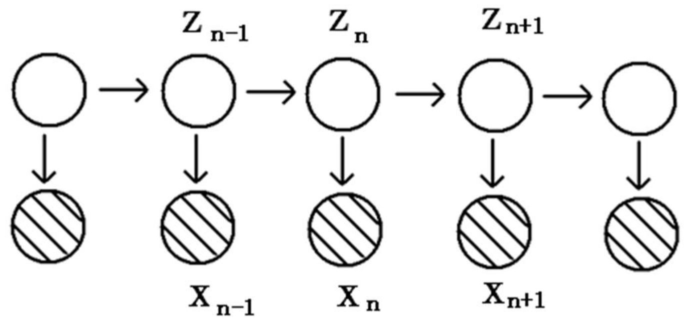
We will show $\gamma(z_{nk})$ can be decomposed to forward and backward term.

$$\gamma(\mathbf{z}_n) = \underbrace{\widehat{\alpha}(\mathbf{z}_n)}_{\text{forward}} \underbrace{\widehat{\beta}(\mathbf{z}_n)}_{\text{backward}} \tag{5}$$

Both of forward and backward term can be computed efficiently.

Forward Backward algorithm

Using independence of PGM, we can decompose $\gamma(\mathbf{z}_n|\mathbf{X})$ into forward and backward term.



HMM

$$
\begin{aligned}
\gamma(\mathbf{z}_n) &= p(\mathbf{z}_n|\mathbf{X}) \\
&= \frac{p(\mathbf{X}|\mathbf{z}_n)p(\mathbf{z}_n)}{p(\mathbf{X})} \\
&= \frac{p(\mathbf{x}_1,\ldots,\mathbf{x}_n|\mathbf{z}_n)p(\mathbf{x}_{n+1},\ldots,\mathbf{x}_N|\mathbf{z}_n)p(\mathbf{z}_n)}{p(\mathbf{x}_1,\ldots,\mathbf{x}_n)p(\mathbf{x}_{n+1},\ldots,\mathbf{x}_N|\mathbf{x}_1,\ldots,\mathbf{x}_n)} \\
&= \frac{p(\mathbf{x}_1,\ldots,\mathbf{x}_n,\mathbf{z}_n)p(\mathbf{x}_{n+1},\ldots,\mathbf{x}_N|\mathbf{z}_n)}{p(\mathbf{x}_1,\ldots,\mathbf{x}_n)p(\mathbf{x}_{n+1},\ldots,\mathbf{x}_N|\mathbf{x}_1,\ldots,\mathbf{x}_n)} \\
&= \widehat{\alpha}(\mathbf{z}_n)\widehat{\beta}(\mathbf{z}_n) \qquad (6)
\end{aligned}
$$

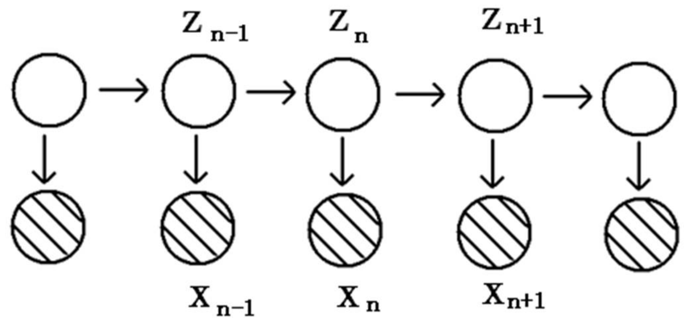
where $\widehat{\alpha}(\mathbf{z}_n) = \dfrac{p(\mathbf{x}_1,\ldots,\mathbf{x}_n,\mathbf{z}_n)}{p(\mathbf{x}_1,\ldots,\mathbf{x}_n)}$ (Forward) (7)

$\widehat{\beta}(\mathbf{z}_n) = \dfrac{p(\mathbf{x}_{n+1},\ldots,\mathbf{x}_N|\mathbf{z}_n)}{p(\mathbf{x}_{n+1},\ldots,\mathbf{x}_N|\mathbf{x}_1,\ldots,\mathbf{x}_n)}$ (Backward) (8)

Compute $\widehat{\alpha}(\mathbf{z}_n)$

$\widehat{\alpha}(\mathbf{z}_n)$ Can be computed efficiently using dependency of HMM

denote $c_n = p(\mathbf{x}_n | \mathbf{x}_1, \ldots, \mathbf{x}_{n-1})$ (9)



HMM

$$\widehat{\alpha}(\mathbf{z}_1) = p(\mathbf{z}_1 | \mathbf{x}_1) = \frac{p(\mathbf{x}_1 | \mathbf{z}_1) p(\mathbf{z}_1)}{c_1} \quad (10)$$

$$\begin{aligned}
\widehat{\alpha}(\mathbf{z}_n) &= p(\mathbf{x}_1, \ldots, \mathbf{x}_n, \mathbf{z}_n) / p(\mathbf{x}_1, \ldots, \mathbf{x}_n) \\
&= p(\mathbf{x}_1, \ldots, \mathbf{x}_n | \mathbf{z}_n) p(\mathbf{z}_n) / p(\mathbf{x}_1, \ldots, \mathbf{x}_n) \\
&= p(\mathbf{x}_1, \ldots, \mathbf{x}_{n-1} | \mathbf{z}_n) p(\mathbf{x}_n | \mathbf{z}_n) p(\mathbf{z}_n) / p(\mathbf{x}_1, \ldots, \mathbf{x}_n) \\
&= p(\mathbf{x}_1, \ldots, \mathbf{x}_{n-1}, \mathbf{z}_n) p(\mathbf{x}_n | \mathbf{z}_n) / p(\mathbf{x}_1, \ldots, \mathbf{x}_n) \\
&= \sum_{\mathbf{z}_{n-1}} p(\mathbf{x}_1, \ldots, \mathbf{x}_{n-1}, \mathbf{z}_{n-1}, \mathbf{z}_n) p(\mathbf{x}_n | \mathbf{z}_n) / p(\mathbf{x}_1, \ldots, \mathbf{x}_n) \\
&= \sum_{\mathbf{z}_{n-1}} p(\mathbf{x}_1, \ldots, \mathbf{x}_{n-1}, \mathbf{z}_n | \mathbf{z}_{n-1}) p(\mathbf{z}_{n-1}) p(\mathbf{x}_n | \mathbf{z}_n) / p(\mathbf{x}_1, \ldots, \mathbf{x}_n) \\
&= \sum_{\mathbf{z}_{n-1}} p(\mathbf{x}_1, \ldots, \mathbf{x}_{n-1} | \mathbf{z}_{n-1}) p(\mathbf{z}_n | \mathbf{z}_{n-1}) p(\mathbf{z}_{n-1}) p(\mathbf{x}_n | \mathbf{z}_n) / \left(p(\mathbf{x}_1, \ldots, \mathbf{x}_{n-1}) p(\mathbf{x}_n | \mathbf{x}_1, \ldots, \mathbf{x}_{n-1}) \right) \\
&= p(\mathbf{x}_n | \mathbf{z}_n) \sum_{\mathbf{z}_{n-1}} p(\mathbf{x}_1, \ldots, \mathbf{x}_{n-1} \mathbf{z}_{n-1}) p(\mathbf{z}_n | \mathbf{z}_{n-1}) / \left(p(\mathbf{x}_1, \ldots, \mathbf{x}_{n-1}) c_n \right) \\
&= \frac{1}{c_n} p(\mathbf{x}_n | \mathbf{z}_n) \sum_{\mathbf{z}_{n-1}} \widehat{\alpha}(\mathbf{z}_{n-1}) p(\mathbf{z}_n | \mathbf{z}_{n-1}) \quad (11)
\end{aligned}$$

where $c_1 = \sum_{\mathbf{z}_1} p(\mathbf{x}_1 | \mathbf{z}_1) p(\mathbf{z}_1)$ (12)

$$c_n = \sum_{\mathbf{z}_n} \left[p(\mathbf{x}_n | \mathbf{z}_n) \sum_{\mathbf{z}_{n-1}} \widehat{\alpha}(\mathbf{z}_{n-1}) p(\mathbf{z}_n | \mathbf{z}_{n-1}) \right]$$ (integrate both side of (11)) (13)

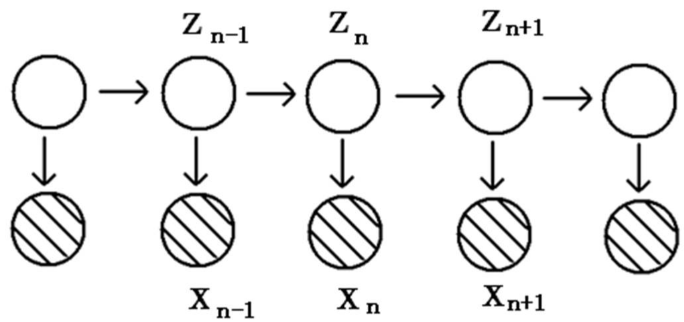
***Byproduct: the likelihood of a sequence X:** $p(\mathbf{X}) = \prod_{n=1}^{N} c_n$ (14)

Compute $\widehat{\beta}(\mathbf{z}_n)$

$$\begin{aligned}
\widehat{\beta}(\mathbf{z}_n) &= p(\mathbf{x}_{n+1}, \dots, \mathbf{x}_N | \mathbf{z}_n) / p(\mathbf{x}_{n+1}, \dots, \mathbf{x}_N | \mathbf{x}_1, \dots, \mathbf{x}_n) \\
&= \sum_{\mathbf{z}_{n+1}} p(\mathbf{x}_{n+1}, \dots, \mathbf{x}_N \mathbf{z}_{n+1} | \mathbf{z}_n) / \left[p(\mathbf{x}_{n+2}, \dots, \mathbf{x}_N | \mathbf{x}_1, \dots, \mathbf{x}_{n+1}) p(\mathbf{x}_{n+1} | \mathbf{x}_1, \dots, \mathbf{x}_n) \right] \\
&= \sum_{\mathbf{z}_{n+1}} p(\mathbf{x}_{n+1}, \dots, \mathbf{x}_N \mathbf{z}_n | \mathbf{z}_{n+1}) p(\mathbf{z}_{n+1}) / p(\mathbf{z}_n) / \left[p(\mathbf{x}_{n+2}, \dots, \mathbf{x}_N | \mathbf{x}_1, \dots, \mathbf{x}_{n+1}) c_{n+1} \right] \\
&= \sum_{\mathbf{z}_{n+1}} p(\mathbf{x}_{n+2}, \dots, \mathbf{x}_N | \mathbf{z}_{n+1}) p(\mathbf{x}_{n+1} | \mathbf{z}_{n+1}) p(\mathbf{z}_n | \mathbf{z}_{n+1}) p(\mathbf{z}_{n+1}) / p(\mathbf{z}_n) / \left[p(\mathbf{x}_{n+2}, \dots, \mathbf{x}_N | \mathbf{x}_1, \dots, \mathbf{x}_{n+1}) c_{n+1} \right] \\
&= \frac{1}{c_{n+1}} \sum_{\mathbf{z}_{n+1}} \widehat{\beta}(\mathbf{z}_{n+1}) p(\mathbf{x}_{n+1} | \mathbf{z}_{n+1}) p(\mathbf{z}_{n+1} | \mathbf{z}_n)
\end{aligned}$$ (15)

where $\widehat{\beta}(\mathbf{z}_n) = \dfrac{\gamma(\mathbf{z}_n)}{\widehat{\alpha}(\mathbf{z}_n)} = \dfrac{p(\mathbf{z}_n | \mathbf{X})}{p(\mathbf{z}_n | \mathbf{X})} = 1$ (16)

Compute $\xi(\mathbf{z}_{n-1}\mathbf{z}_n)$



HMM

Using conditional dependency of HMM

$$
\begin{aligned}
\xi(\mathbf{z}_{n-1}\mathbf{z}_n) &= p(\mathbf{z}_{n-1}\mathbf{z}_n|\mathbf{X}) \\
&= \frac{p(\mathbf{X}|\mathbf{z}_{n-1}\mathbf{z}_n)p(\mathbf{z}_{n-1}\mathbf{z}_n)}{p(\mathbf{X})} \\
&= \frac{p(\mathbf{x}_1,\ldots,\mathbf{x}_{n-1}|\mathbf{z}_n)p(\mathbf{x}_n|\mathbf{z}_n)p(\mathbf{x}_{n+1},\ldots,\mathbf{x}_N|\mathbf{z}_n)p(\mathbf{z}_{n-1}\mathbf{z}_n)}{p(\mathbf{X})} \\
&= \frac{p(\mathbf{x}_1,\ldots,\mathbf{x}_{n-1},\mathbf{z}_n)/p(\mathbf{z}_n)p(\mathbf{x}_n|\mathbf{z}_n)p(\mathbf{x}_{n+1},\ldots,\mathbf{x}_N|\mathbf{z}_n)p(\mathbf{z}_{n-1}\mathbf{z}_n)}{p(\mathbf{X})} \\
&= \widehat{\alpha}(\mathbf{z}_{n-1})p(\mathbf{x}_n|\mathbf{z}_n)p(\mathbf{z}_n|\mathbf{z}_{n-1})\widehat{\beta}(\mathbf{z}_n) \qquad (17)
\end{aligned}
$$

M step

In M step, need to optimize $Q(\theta,\theta^{\text{old}})$ with respect to θ = {**π**, **A**, ϕ}

$$Q(\theta,\theta^{\text{old}}) = \underbrace{\sum_{k=1}^{K}\gamma(z_{nk})\ln\pi_k}_{\text{①}} + \underbrace{\sum_{n=2}^{N}\sum_{j=1}^{K}\sum_{k=1}^{K}\xi(z_{n-1,j}z_{n,k})\ln A_{jk}}_{\text{②}} + \underbrace{\sum_{n=1}^{N}\sum_{k=1}^{K}\gamma(z_{nk})\ln p(\mathbf{x}_n \,|\, z_{nk}\phi)}_{\text{③}} \qquad (18)$$

π is only associated with ①
A is only associated with ②
ϕ is only associated with ③

π, **A**, ϕ are independent, so they can be optimized respectively.

① maximize π

Take constraint $\sum_{j=1}^{K} \pi_j = 1$ into consideration, introduce Lagrange function

$$R_1(\boldsymbol{\pi}, \lambda) = \sum_{k=1}^{K} \gamma(z_{nk}) \ln \pi_k + \lambda \left(\sum_{j=1}^{K} \pi_j - 1 \right) \tag{19}$$

Let $\frac{\partial R_1(\boldsymbol{\pi}, \lambda)}{\partial \pi_k} = 0 \quad k = 1, \ldots, K$ $\Rightarrow$ $\pi_k = \frac{\gamma_{1k}}{\sum_{j=1}^{K} \gamma_{1j}}, \quad k = 1, \ldots, K$ (20)

② maximize A

Take constraint $\sum_{l=1}^{K} A_{jl} = 1 \quad l = 1, \ldots, K$ into consideration, introduce Lagrange function

$$R_2(\mathbf{A}, \lambda_1, \ldots, \lambda_k) = \sum_{n=2}^{N} \sum_{j=1}^{K} \sum_{k=1}^{K} \xi(z_{n-1,j} z_{n,k}) \ln A_{jk} + \lambda_1 \left(\sum_{l=1}^{K} A_{1l} - 1 \right) + \ldots + \lambda_K \left(\sum_{l=1}^{K} A_{Kl} - 1 \right) \tag{21}$$

Let $\frac{\partial R_2(\mathbf{A}, \lambda_1, \ldots, \lambda_k)}{\partial A_{jk}} = 0 \quad k = 1, \ldots, K$ $\Rightarrow$ $A_{jk} = \frac{\sum_{n=2}^{N} \xi(z_{n-1,j} z_{n,k})}{\sum_{l=1}^{K} \sum_{n=2}^{N} \xi(z_{n-1,j} z_{n,l})} \quad j, k = 1, \ldots, K$ (22)

This shows the elements which are zero in A_{jk} will keep zero all the time.
So it you want to get left-right HMM just initialize A as upper diagonal matrix.

③ Maximize ϕ

the emission distribution can be multinominal, Gaussian, etc. We will discuss separately.

Ⅰ. Emission distribtution is discrete multinominal distribution

probability density function

$$p(\mathbf{x}_n | z_{nk}\phi) = \prod_{m=1}^{M} B_{km}{}^{x_{nm}} \quad s.t. \sum_{l=1}^{M} B_{kl} = 1, \quad k = 1, \ldots, K \tag{23}$$

where $\phi = \{B_{km}\} \quad k = 1, \ldots, K, m = 1, \ldots, M$

B_{km} represents the probability of m-th event at k-th state.

To estimate B_{km}, introduce Lagarange function

$$R_3(\phi, \phi^{\text{old}}, \lambda_1, \ldots, \lambda_K) = \sum_{n=1}^{N} \sum_{k=1}^{K} \gamma(z_{nk}) \ln p(\mathbf{x}_n | z_{nk}\phi) + \lambda_1 \left(\sum_{l=1}^{M} B_{1l} - 1 \right) + \lambda_K \left(\sum_{l=1}^{M} B_{Kl} - 1 \right) \tag{24}$$

$$\frac{\partial R_3(\phi, \phi^{\text{old}}, \lambda_1, \ldots, \lambda_K)}{\partial B_{km}} = 0 \quad \Rightarrow \quad B_{km} = \frac{\sum_{n=1}^{N} \gamma_{nk} x_{nm}}{\sum_{n=1}^{N} \gamma_{nk} \sum_{m=1}^{M} x_{nm}} \quad k = 1, \ldots, K; m = 1, \ldots, M \tag{25}$$

EM algorithm for multinominal-HMM

1. Init parameters $\theta = \{\boldsymbol{\pi}, \mathbf{A}, \phi\}$

   where $\phi = \{B_{km}\} \quad k = 1, \ldots, K; m = 1, \ldots, M$

2. E step

   Calculate $\gamma(\mathbf{z}_n)$ using (6)

   Calculate $\xi(\mathbf{z}_{n-1}\mathbf{z}_n)$ using (17)

3. M step

$$\pi_k = \frac{\gamma_{1k}}{\sum_{j=1}^{K} \gamma_{1j}}, \quad k = 1, \ldots, K \qquad A_{jk} = \frac{\sum_{n=2}^{N} \xi(z_{n-1,j} z_{n,k})}{\sum_{l=1}^{K} \sum_{n=2}^{N} \xi(z_{n-1,j} z_{n,l})} \quad j, k = 1, \ldots, K$$

$$B_{km} = \frac{\sum_{n=1}^{N} \gamma_{nk} x_{nm}}{\sum_{n=1}^{N} \gamma_{nk} \sum_{m=1}^{M} x_{nm}} \quad k = 1, \ldots, K; m = 1, \ldots, M$$

4. If converge then stop, otherwise goto 2.

Example

Generate Multinominal-HMM for belowing sequences

Data{1} = [1 1 1 4 1 1 1 2 2 2 2 2 2 1 2 2 2 2 3 3 3 3 3 1 3 3 3]
Data{2} = [1 1 2 1 1 1 1 1 2 2 2 3 2 2 2 2 2 3 3 3 3 3 4 3 3 3]

State num: 3, multinominal num: 4

Output

$\boldsymbol{\pi}$ = [1, 0, 0]

A = [0.87, 0.13, 0,00
0.00, 0.90, 0.10
0.00, 0.00, 0.10]

B = [0.85, 0.06, 0.05
0.08, 0.89, 0.00
0.00, 0.05, 0.89
0.07, 0.00, 0.06]

Ⅱ. Emission distribution is Gaussian distribution

probability density function $p(\mathbf{x}_n | z_{nk}, \phi) = N(\mathbf{x}_n | \boldsymbol{\mu}_k \boldsymbol{\Sigma}_k)$ (26)

where $\phi = \{\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\} \quad k = 1, \ldots, K$

Define
$$R_3(\phi, \phi^{old}) = \sum_{n=1}^{N} \sum_{k=1}^{K} \gamma(z_{nk}) \ln p(\mathbf{x}_n | z_{nk} \phi)$$
$$= \sum_{n=1}^{N} \sum_{k=1}^{K} \gamma_{nk} \left[-\frac{D}{2} \ln(2\pi) - \frac{1}{2} \ln |\boldsymbol{\Sigma}_k| - \frac{1}{2} (\mathbf{x}_n - \boldsymbol{\mu}_k)^{\mathrm{T}} \boldsymbol{\Sigma}_k^{-1} (\mathbf{x}_n - \boldsymbol{\mu}_k) \right] \quad (27)$$

To estimate parameters, let derivative of (27) with respect to $\phi = \{\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\} \quad k = 1, \ldots, K$ be zero

$$\frac{\partial R_3(\phi, \phi^{old})}{\partial \boldsymbol{\mu}_k} = 0 \quad k = 1, \ldots, K$$

$$\frac{\partial R_3(\phi, \phi^{old})}{\partial \boldsymbol{\Sigma}_k} = 0 \quad k = 1, \ldots, K$$

$\Rightarrow$

$$\boldsymbol{\mu}_k = \frac{\sum_{n=1}^{N} \gamma_{nk} \mathbf{x}_n}{\sum_{n=1}^{N} \gamma_{nk}} \quad k = 1, \ldots, K$$

$$\boldsymbol{\Sigma}_k = \frac{\sum_{n=1}^{N} \gamma_{nk} (\mathbf{x}_n - \boldsymbol{\mu}_k)(\mathbf{x}_n - \boldsymbol{\mu}_k)^{\mathrm{T}}}{\sum_{n=1}^{N} \gamma_{nk}} \quad k = 1, \ldots, K \quad (28)$$

EM algorithm for Gaussian-HMM

1. Init parameters $\theta = \{\boldsymbol{\pi}, \mathbf{A}, \phi\}$

   where $\phi = \{\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_K, \boldsymbol{\Sigma}_1, \ldots, \boldsymbol{\Sigma}_K\} \quad k = 1, \ldots, K$

2. E step

   Calculate $\gamma(\mathbf{z}_n)$ using (6)

   Calculate $\xi(\mathbf{z}_{n-1}\mathbf{z}_n)$ using (17)

3. M step

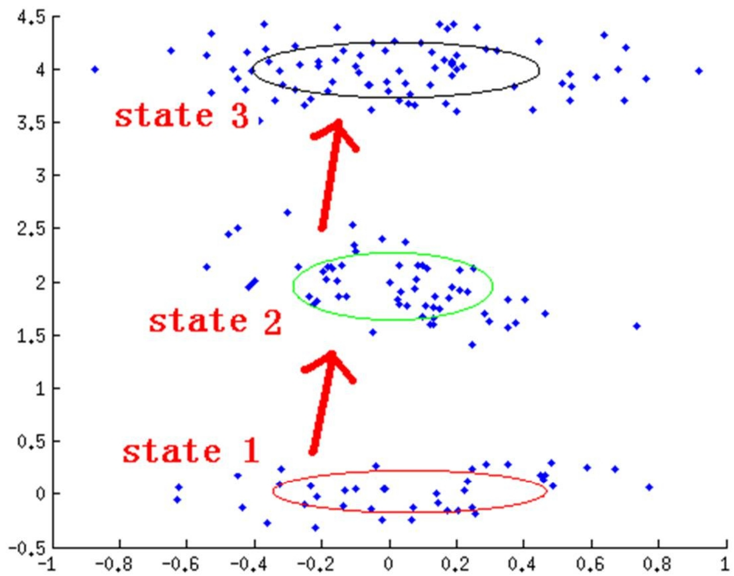
$$\pi_k = \frac{\gamma_{1k}}{\sum_{j=1}^{K} \gamma_{1j}}, \quad k = 1, \ldots, K \qquad A_{jk} = \frac{\sum_{n=2}^{N} \xi(z_{n-1,j} z_{n,k})}{\sum_{l=1}^{K} \sum_{n=2}^{N} \xi(z_{n-1,j} z_{n,l})} \quad j,k = 1, \ldots, K$$

$$\boldsymbol{\mu}_k = \frac{\sum_{n=1}^{N} \gamma_{nk} \mathbf{x}_n}{\sum_{n=1}^{N} \gamma_{nk}} \quad k = 1, \ldots, K \qquad \boldsymbol{\Sigma}_k = \frac{\sum_{n=1}^{N} \gamma_{nk} (\mathbf{x}_n - \boldsymbol{\mu}_k)(\mathbf{x}_n - \boldsymbol{\mu}_k)^{\mathrm{T}}}{\sum_{n=1}^{N} \gamma_{nk}} \quad k = 1, \ldots, K$$

4. If converge then stop, otherwise goto 2.

Example

The graph below shows the trained Gaussian-HMM model using created data.



State num: 3

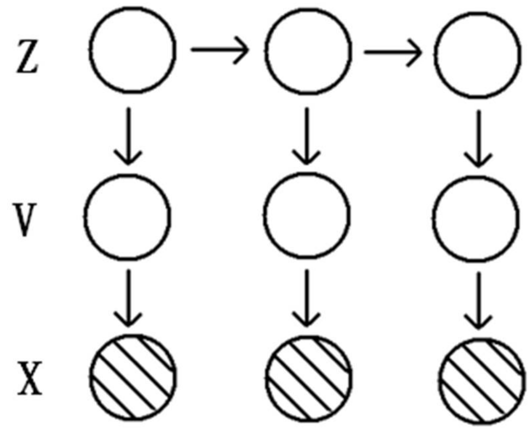
π = [1, 0, 0] **A** = [0.95, 0.05, 0.00
0.00, 0.97, 0.03
0.00, 0.00, 1.00]

$\phi = \{\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\} \quad k = 1, \ldots, K$ is shown in graph

GMM-HMM

Gaussian distribution is not able to capture distributions with many centers. Especially in ASR, where the timbre differentiates from person to person.

In GMM-HMM, every single state is a GMM model. Define **v** as 1-of-K*M random variable, where K is the number of state. M is the mixture number of GMM. v_{km}=1 represents the k-th state, m-th mixture occurs.



GMM-HMM

Parameters are $\theta = \{\boldsymbol{\pi}, \mathbf{A}, \phi\}$

where $\phi = \{B_{km}, \boldsymbol{\mu}_{km}, \boldsymbol{\Sigma}_{km}\} \quad k = 1, \ldots, K; m = 1, \ldots, M$

$B_{km} = p(v_{km} | z_k)$ is the probability of m-th mixture under k-th state

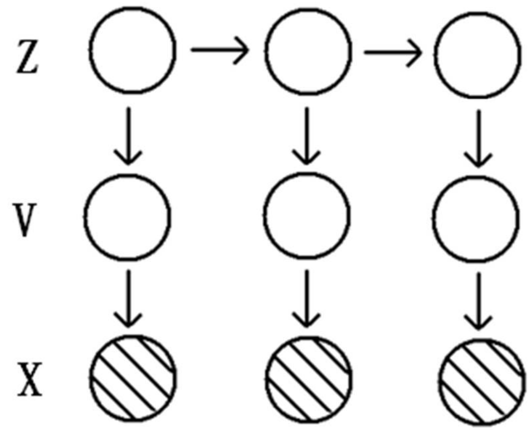
$\boldsymbol{\mu}_{km}, \boldsymbol{\Sigma}_{km}$ is mean and covariance of the k-th state, m-mixture, respectively.

Probability density function

$$p(\mathbf{x} | z_k \phi) = \sum_{m=1}^{M} p(v_{km} | z_k \phi) p(\mathbf{x} | v_{km} \phi) = \sum_{m=1}^{M} B_{km} N(\mathbf{x} | \boldsymbol{\mu}_{km}, \boldsymbol{\Sigma}_{km}) \qquad (29)$$

Apply EM algorithm to GMM-HMM

Compared with GMM-HMM, there are other latent variables $\mathbf{v}_n$, which dominates the emission GMM



GMM-HMM

denote
$$\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$$
$$\mathbf{V} = \{\mathbf{v}_1, \ldots, \mathbf{v}_N\} \tag{30}$$
$$\mathbf{Z} = \{\mathbf{z}_1, \ldots, \mathbf{z}_N\}$$

Using independence of GMM-HMM, calculate $Q(\theta,\theta^{old})$

$$
\begin{aligned}
Q(\theta,\theta^{\text{old}}) &= \sum_{\mathbf{VZ}} p(\mathbf{VZ}|\mathbf{X}\theta^{\text{old}}) \ln p(\mathbf{VXZ}|\theta) \\
&= \sum_{\mathbf{VZ}} p(\mathbf{VZ}|\mathbf{X}\theta^{\text{old}}) \left[\ln p(\mathbf{z}_1) + \sum_{n=2}^{N} \ln p(\mathbf{z}_n|\mathbf{z}_{n-1}) + \sum_{n=1}^{N} \ln p(\mathbf{v}_n|\mathbf{z}_n) + \sum_{n=1}^{N} \ln p(\mathbf{x}_n|\mathbf{v}_n) \right] \\
&= \sum_{\mathbf{z}_1} p(\mathbf{z}_1|\mathbf{X}\theta^{\text{old}}) \ln p(\mathbf{z}_1) + \sum_{n=2}^{N} \sum_{\mathbf{z}_{n-1}\mathbf{z}_n} p(\mathbf{z}_{n-1}\mathbf{z}_n|\mathbf{X}\theta^{\text{old}}) \ln p(\mathbf{z}_n|\mathbf{z}_{n-1}) + \sum_{n=1}^{N} \sum_{\mathbf{v}_n\mathbf{z}_n} p(\mathbf{v}_n\mathbf{z}_n|\mathbf{X}\theta^{\text{old}}) \left[\ln p(\mathbf{v}_n|\mathbf{z}_n) + \ln p(\mathbf{x}_n|\mathbf{v}_n) \right] \\
&= \underbrace{\sum_{k=1}^{K} \gamma(z_{1k}) \ln \pi_k}_{①} + \underbrace{\sum_{n=1}^{N}\sum_{j=1}^{K}\sum_{k=1}^{K} \xi(z_{n-1,j} z_{n,k}) \ln A_{jk}}_{②} + \underbrace{\sum_{n=1}^{N}\sum_{k=1}^{K}\sum_{m=1}^{M} \eta_{nkm} \left[\ln B_{km} - \frac{D}{2}\ln(2\pi) - \frac{1}{2}\ln|\boldsymbol{\Sigma}_{km}| - \frac{1}{2}(\mathbf{x}_n - \boldsymbol{\mu}_k)^{\mathrm{T}} \boldsymbol{\Sigma}_{km}^{-1} (\mathbf{x}_n - \boldsymbol{\mu}_k) \right]}_{③}
\end{aligned} \tag{31}
$$

E step

Calculate $p(\mathbf{VZ}|\mathbf{X}\theta^{\text{old}})$

From (31), no need to calculate all $p(\mathbf{VZ}|\mathbf{X}\theta^{\text{old}})$, only the terms below is needed

$$\gamma(z_{nk}) = p(z_{nk}|\mathbf{X}\theta^{\text{old}}) \tag{32}$$

$$\xi(z_{n-1,j}z_{n,k}) = p(z_{n-1,j}z_{n,k}|\mathbf{X}\theta^{\text{old}}) \tag{33}$$

$$\eta_{nkm} = p(v_{nkm}z_{nk}|\mathbf{X}\theta^{\text{old}}) \tag{34}$$

(32), (33) can be calculated in same way as (6), (17)

To calculate (34), just use forward-backward algorithm similar to (6)

$$\eta(\mathbf{v}_n\mathbf{z}_n) = p(\mathbf{v}_n\mathbf{z}_n|\mathbf{X}) = \underbrace{\widehat{\alpha}(\mathbf{v}_n\mathbf{z}_n)}_{\text{forward}}\underbrace{\widehat{\beta}(\mathbf{v}_n\mathbf{z}_n)}_{\text{backward}} \tag{35}$$

where $$\widehat{\alpha}(\mathbf{v}_n\mathbf{z}_n) = \frac{p(\mathbf{x}_1,\ldots,\mathbf{x}_n,\mathbf{v}_n\mathbf{z}_n)}{p(\mathbf{x}_1,\ldots,\mathbf{x}_1)} \tag{36}$$

$$\widehat{\beta}(\mathbf{v}_n\mathbf{z}_n) = \frac{p(\mathbf{x}_{n+1},\ldots,\mathbf{x}_N|\mathbf{v}_n\mathbf{z}_n)}{p(\mathbf{x}_{n+1},\ldots,\mathbf{x}_N|\mathbf{x}_1,\ldots,\mathbf{x}_n)} \tag{37}$$

denote $c_n = p(\mathbf{x}_n | \mathbf{x}_1, \ldots, \mathbf{x}_{n-1})$

$\widehat{\alpha}(\mathbf{v}_n \mathbf{z}_n), \widehat{\beta}(\mathbf{v}_n \mathbf{z}_n), c_n$ can be computed similar to (10) – (16)

Results are given without detailed deduction

$$\widehat{\alpha}(\mathbf{v}_1 \mathbf{z}_1) = p(\mathbf{v}_1 \mathbf{z}_1 | \mathbf{x}_1) = \frac{p(\mathbf{x}_1 | \mathbf{v}_1 \mathbf{z}_1) p(\mathbf{v}_1 \mathbf{z}_1)}{p(\mathbf{x}_1)} = \frac{p(\mathbf{x}_1 | \mathbf{v}_1) p(\mathbf{v}_1 | \mathbf{z}_1) p(\mathbf{z}_1)}{c_1} \tag{38}$$

$$\widehat{\alpha}(\mathbf{v}_n \mathbf{z}_n) = \frac{1}{c_n} p(\mathbf{x}_n | \mathbf{v}_n) p(\mathbf{v}_n | \mathbf{z}_n) \sum_{\mathbf{v}_{n-1} \mathbf{z}_{n-1}} \widehat{\alpha}(\mathbf{v}_{n-1} \mathbf{z}_{n-1}) p(\mathbf{z}_n | \mathbf{z}_{n-1}) \tag{39}$$

$$c_1 = p(\mathbf{x}_1) = \sum_{\mathbf{v}_1 \mathbf{z}_1} p(\mathbf{v}_1 \mathbf{z}_1 \mathbf{x}_1) = \sum_{\mathbf{v}_1 \mathbf{z}_1} p(\mathbf{z}_1) p(\mathbf{v}_1 | \mathbf{z}_1) p(\mathbf{x}_1 | \mathbf{v}_1) \tag{40}$$

$$c_n = \sum_{\mathbf{v}_n \mathbf{z}_n} \left[p(\mathbf{x}_n | \mathbf{v}_n) p(\mathbf{v}_n | \mathbf{z}_n) \sum_{\mathbf{v}_{n-1} \mathbf{z}_{n-1}} \widehat{\alpha}(\mathbf{v}_{n-1} \mathbf{z}_{n-1}) p(\mathbf{z}_n | \mathbf{z}_{n-1}) \right] \tag{41}$$

$$\widehat{\beta}(\mathbf{v}_N \mathbf{z}_N) = 1 \tag{42}$$

$$\widehat{\beta}(\mathbf{v}_n \mathbf{z}_n) = \frac{1}{c_{n+1}} \sum_{\mathbf{v}_{n+1} \mathbf{z}_{n+1}} \widehat{\beta}(\mathbf{v}_{n+1} \mathbf{z}_{n+1}) p(\mathbf{x}_{n+1} | \mathbf{v}_{n+1}) p(\mathbf{z}_{n+1} | \mathbf{z}_n) p(\mathbf{v}_{n+1} | \mathbf{z}_{n+1}) \tag{43}$$

M step

Maximize π, A

The maximization of **π**, **A** is same as (20), (22).

$$\pi_k = \frac{\gamma_{1k}}{\sum_{j=1}^{K} \gamma_{1j}}, \quad k = 1, \ldots, K \tag{44}$$

$$A_{jk} = \frac{\sum_{n=2}^{N} \xi(z_{n-1,j} z_{n,k})}{\sum_{l=1}^{K} \sum_{n=2}^{N} \xi(z_{n-1,j} z_{n,l})} \quad j,k = 1, \ldots, K \tag{45}$$

Maximize ϕ

To maximize $\phi = \{B_{km}, \boldsymbol{\mu}_{km}, \boldsymbol{\Sigma}_{km}\} \quad k = 1, \ldots, K; m = 1, \ldots, M$

Taking constraint $\sum_{l=1}^{M} B_{kl} = 1, \quad k = 1, \ldots, K$ into account

Introduce Lagrange function

$$R_3(\phi, \phi^{old}, \lambda_1, \ldots, \lambda_K) = \sum_{n=1}^{N} \sum_{k=1}^{K} \sum_{m=1}^{M} \eta_{nkm} \left[\ln B_{km} - \frac{D}{2} \ln(2\pi) - \frac{1}{2} \ln |\boldsymbol{\Sigma}_{km}| - \frac{1}{2} (\mathbf{x}_n - \boldsymbol{\mu}_k)^{\mathrm{T}} \boldsymbol{\Sigma}_{km}^{-1} (\mathbf{x}_n - \boldsymbol{\mu}_k) \right] + \lambda_1 \left(\sum_{l=1}^{M} B_{1l} - 1 \right) + \lambda_K \left(\sum_{l=1}^{M} B_{Kl} - 1 \right) \tag{46}$$

Let the derivative of $R_3(\phi, \phi^{\text{old}}, \lambda_1, \ldots, \lambda_K)$ With respect to $\phi = \{B_{km}, \boldsymbol{\mu}_{km}, \boldsymbol{\Sigma}_{km}\} \quad k = 1, \ldots, K; m = 1, \ldots, M$ be zero

$$\frac{\partial R_3(\phi, \phi^{\text{old}}, \lambda_1, \ldots, \lambda_K)}{\partial B_{km}} = 0$$

$$\frac{\partial R_3(\phi, \phi^{\text{old}}, \lambda_1, \ldots, \lambda_K)}{\partial \boldsymbol{\mu}_{km}} = 0$$

$$\frac{\partial R_3(\phi, \phi^{\text{old}}, \lambda_1, \ldots, \lambda_K)}{\partial \boldsymbol{\Sigma}_{km}} = 0$$

$$B_{km} = \frac{\sum_{n=1}^{N} \eta_{nkm}}{\sum_{n=1}^{N} \sum_{j=1}^{M} \eta_{nkj}}$$

$$\boldsymbol{\mu}_{km} = \frac{\sum_{n=1}^{N} \eta_{nkm} \mathbf{x}_n}{\sum_{n=1}^{N} \eta_{nkm}} \qquad k = 1, \ldots, K; m = 1, \ldots, M \qquad (47)$$

$$\boldsymbol{\Sigma}_{km} = \frac{\sum_{n=1}^{N} \eta_{nkm} (\mathbf{x}_n - \boldsymbol{\mu}_{km})(\mathbf{x}_n - \boldsymbol{\mu}_{km})^{\mathrm{T}}}{\sum_{n=1}^{N} \eta_{nkm}}$$

EM algorithm for GMM-HMM

1. Init parameters $\theta = \{\boldsymbol{\pi}, \mathbf{A}, \phi\}$

   where $\phi = \{B_{km}, \boldsymbol{\mu}_{km}, \boldsymbol{\Sigma}_{km}\} \quad k = 1, \ldots, K; m = 1, \ldots, M$

2. E step

   Calculate $\gamma(\mathbf{z}_n)$, $\xi(\mathbf{z}_{n-1}\mathbf{z}_n)$, $\eta(\mathbf{v}_n\mathbf{z}_n)$ from (32), (33), (34)

3. M step

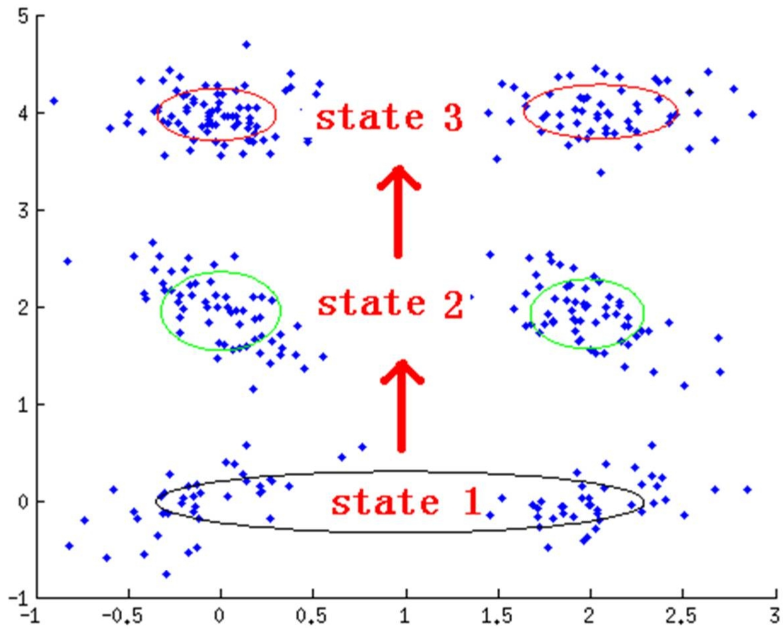
$$\pi_k = \frac{\gamma_{1k}}{\sum_{j=1}^{K} \gamma_{1j}}, \quad k = 1, \ldots, K \qquad A_{jk} = \frac{\sum_{n=2}^{N} \xi(z_{n-1,j} z_{n,k})}{\sum_{l=1}^{K}\sum_{n=2}^{N} \xi(z_{n-1,j} z_{n,l})} \quad j,k = 1, \ldots, K$$

$$B_{km} = \frac{\sum_{n=1}^{N} \eta_{nkm}}{\sum_{n=1}^{N}\sum_{j=1}^{M} \eta_{nkj}} \qquad \boldsymbol{\mu}_{km} = \frac{\sum_{n=1}^{N} \eta_{nkm} \mathbf{x}_n}{\sum_{n=1}^{N} \eta_{nkm}} \qquad \boldsymbol{\Sigma}_{km} = \frac{\sum_{n=1}^{N} \eta_{nkm} (\mathbf{x}_n - \boldsymbol{\mu}_{km})(\mathbf{x}_n - \boldsymbol{\mu}_{km})^{\mathrm{T}}}{\sum_{n=1}^{N} \eta_{nkm}}$$

4. If converge then stop, otherwise goto 2.

Example

A GMM-HMM with state num:3, mix num:2



$\boldsymbol{\pi}$: [0, 0, 1]

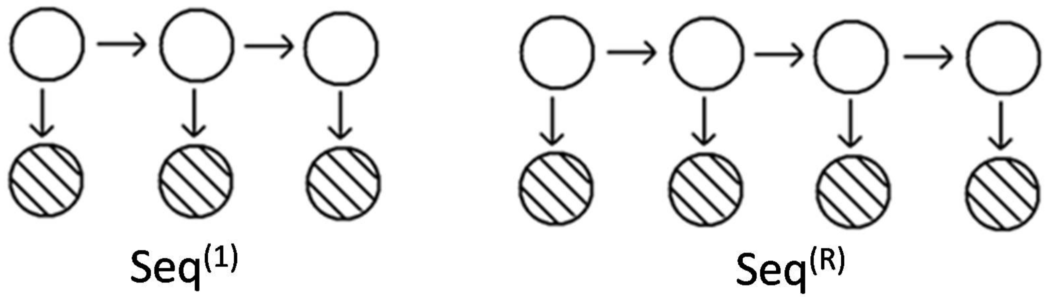
A: [1.00, 0.00, 0.00,
0.03, 0.97, 0.00,
0.00, 0.05, 0.95]

$\phi = \{B_{km}, \boldsymbol{\mu}_{km}, \boldsymbol{\Sigma}_{km}\} \quad k = 1, \ldots, K; m = 1, \ldots, M$

is shown on the graph

Multi Sequence Training

Till now, we train HMM only use one sequence.
In ASR, we have many utterance to train one HMM model for each phoneme / word.
This part will show how to train HMM using multi sequences.



Using independency of PGM, Q(θ,θ^{old})

$$Q(\theta,\theta^{\text{old}}) = \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}\theta^{\text{old}}) \ln p(\mathbf{X}\mathbf{Z}|\theta)$$

$$= \sum_{r=1}^{R} \left(\sum_{\mathbf{Z}^{(r)}} p(\mathbf{Z}^{(r)}|\mathbf{X}^{(r)}\theta^{\text{old}}) \ln p(\mathbf{X}^{(r)}\mathbf{Z}^{(r)}|\theta) \right)$$

$$= \sum_{r=1}^{R}\sum_{k=1}^{K} \gamma_{1k}^{(r)} \ln \pi_k + \sum_{r=1}^{R}\sum_{n=2}^{N}\sum_{j=1}^{K}\sum_{k=1}^{K} \xi^{(r)}(z_{n-1,j} z_{n,k}) \ln \mathrm{A}_{jk} + \sum_{l=r}^{R}\sum_{n=1}^{N}\sum_{k=1}^{K} \gamma_{nk}^{(r)} \ln p(\mathbf{x}_n|z_{nk}) \qquad (48)$$

Where X$^{(r)}$ is the observation variables of r-th seq.
Z$^{(r)}$ is the latent variables of r-th seq.

E STEP

Estimate γ, ξ, η for each sequences separately.

$$\gamma_{nk}^{(r)} = p(z_{nk}^{(r)} \mid \mathbf{X}^{(r)} \theta^{old}) \tag{49}$$

$$\xi^{(r)}(z_{n-1,j} z_{n,k}) = p(z_{n-1,j}^{(r)} z_{n,k}^{(r)} \mid \mathbf{X}^{(r)} \theta^{old}) \tag{50}$$

$$\eta_{nkm}^{(r)} = p(v_{nkm}^{(r)} z_{nk}^{(r)} \mid \mathbf{X} \theta^{\text{old}}) \quad \text{(for GMM only)} \tag{51}$$

M STEP

Optimize (48) with respect to θ = {**π**, **A**, ϕ}

$$\pi_k = \frac{\sum_{r=1}^{R} \gamma_{1k}^{(r)}}{\sum_{r=1}^{R} \sum_{j=1}^{K} \gamma_{1j}^{(r)}} \tag{52}$$

$$A_{jk} = \frac{\sum_{r=1}^{R} \sum_{n=2}^{N} \xi^{(r)}(z_{n-1,j} z_{n,k})}{\sum_{r=1}^{R} \sum_{l=1}^{K} \sum_{n=2}^{N} \xi^{(r)}(z_{n-1,j} z_{n,l})} \tag{53}$$

Ⅰ. Multinominal Distribution

$$B_{km} = \frac{\sum_{r=1}^{R}\sum_{n=1}^{N^{(r)}} \gamma_{nk}^{(r)} x_{nm}^{(r)}}{\sum_{r=1}^{R}\sum_{n=1}^{N^{(r)}} \gamma_{nk}^{(r)} \sum_{m=1}^{M} x_{nm}^{(r)}} \tag{54}$$

Ⅱ. Gauss Distribution

$$\boldsymbol{\mu}_k = \frac{\sum_{r=1}^{R}\sum_{n=1}^{N^{(r)}} \gamma_{nk}^{(r)} \mathbf{x}_n^{(r)}}{\sum_{r=1}^{R}\sum_{n=1}^{N^{(r)}} \gamma_{nk}^{(r)}} \qquad \boldsymbol{\Sigma}_k = \frac{\sum_{r=1}^{R}\sum_{n=1}^{N^{(r)}} \gamma_{nk}^{(r)} (\mathbf{x}_n^{(r)} - \boldsymbol{\mu}_k)(\mathbf{x}_n^{(r)} - \boldsymbol{\mu}_k)^{\mathrm{T}}}{\sum_{r=1}^{R}\sum_{n=1}^{N^{(r)}} \gamma_{nk}^{(r)}} \tag{55}$$

Ⅲ. GMM

$$B_{km} = \frac{\sum_{r=1}^{R}\sum_{n=1}^{N^{(r)}} \eta_{nkm}^{(r)}}{\sum_{r=1}^{R}\sum_{n=1}^{N^{(r)}}\sum_{j=1}^{M} \eta_{nkj}^{(r)}} \qquad \boldsymbol{\mu}_{km} = \frac{\sum_{r=1}^{R}\sum_{n=1}^{N^{(r)}} \eta_{nkm}^{(r)} \mathbf{x}_n}{\sum_{r=1}^{R}\sum_{n=1}^{N^{(r)}} \eta_{nkm}^{(r)}} \qquad \boldsymbol{\Sigma}_{km} = \frac{\sum_{r=1}^{R}\sum_{n=1}^{N^{(r)}} \eta_{nkm}^{(r)} (\mathbf{x}_n - \boldsymbol{\mu}_{km})(\mathbf{x}_n - \boldsymbol{\mu}_{km})^{\mathrm{T}}}{\sum_{r=1}^{R}\sum_{n=1}^{N^{(r)}} \eta_{nkm}^{(r)}} \tag{56}$$

Decoding of HMM

Q: How to find the best decoding path?

A:
$$\mathbf{Z}_{opt} = \max_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}) = \max_{\mathbf{Z}} p(\mathbf{Z}\mathbf{X})$$
$$= \max_{\mathbf{Z}} p(\mathbf{z}_1)\prod_{n=2}^{N} p(\mathbf{z}_n|\mathbf{z}_{n-1})\prod_{n=1}^{N} p(\mathbf{x}_n|\mathbf{z}_n) \quad (57)$$

However, we need to evaluate all possible **Z** which is K^N times to get accurate solution. This infeasible.

Viterbi Algorithm

Use greedy algorithm to estimate optimized path step by step.
By discarding paths with low probability and storing previous step,
the computation complexity decreased to K*N

Viterbi Algorithm

$$V_{1k} = p(\mathbf{x}_1 | z_{1k}) \times p(z_{1k})$$

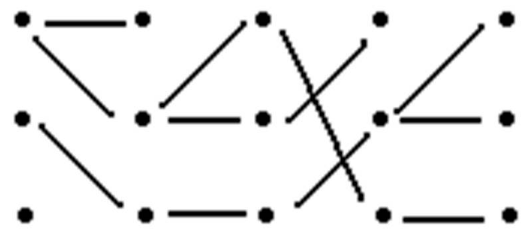
$$V_{1k} = p(\mathbf{x}_1 | z_{1k}) \times p(z_{1k})$$

$$for\ n = 2 : N$$

$$V_{nk} = \max_j \left(V_{n-1,j} \times p(z_{nk} | z_{n-1,j}) \times p(\mathbf{x}_n | z_{nj}) \right)$$

$$\text{path}(n-1) = j$$

$$\text{path}(n) = \arg\max_j V_{nj}$$

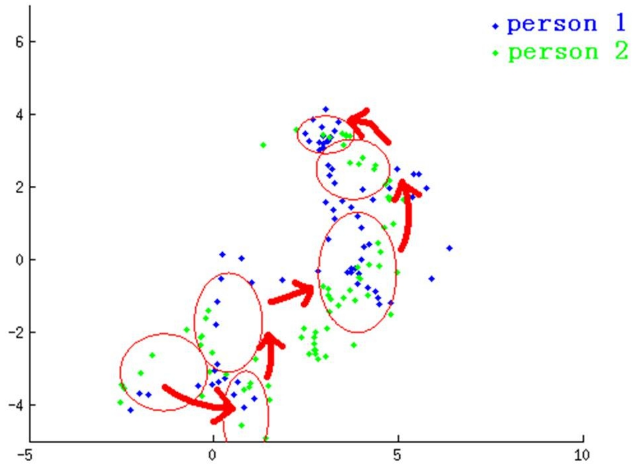


Path stored

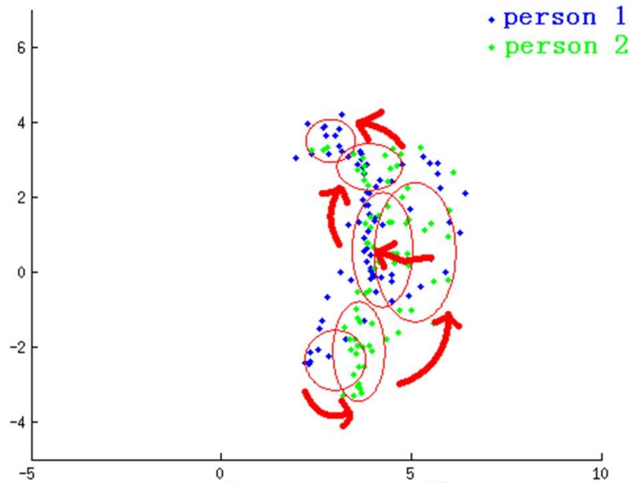
Experiments in small vocabulary ASR

12 Mfcc feature (only choose the 1st and 2nd dimension to plot)

1. **Gaussian-HMM**, state num: 6



“kaimen”



“guanmen”

π = [1, 0, 0, 0, 0, 0]

A =

0.86	0.14	0	0	0	0
0	0.90	0.10	0	0	0
0	0	0.87	0.13	0	0
0	0	0	0.97	0.03	0
0	0	0	0	0.91	0.09
0	0	0	0	0	1.00

π = [1, 0, 0, 0, 0, 0]

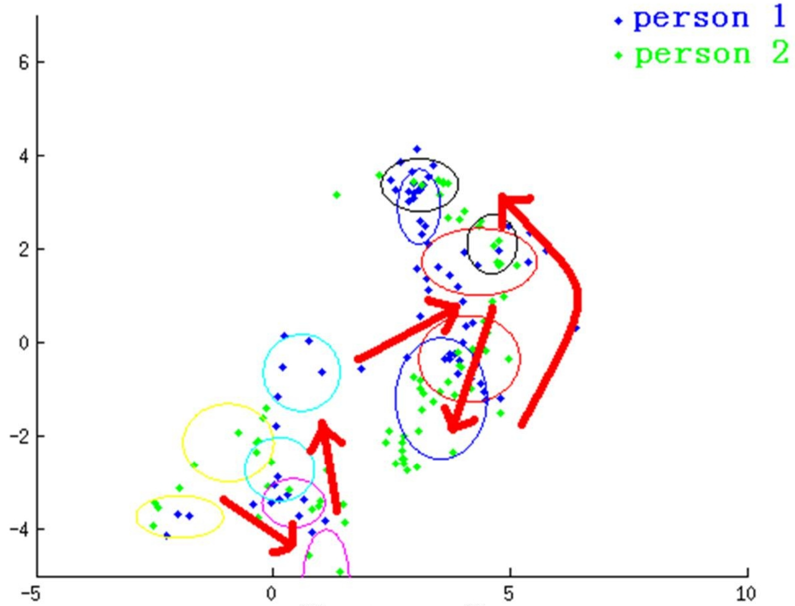
A =

0.89	0.11	0	0	0	0
0	0.87	0.13	0	0	0
0	0	0.77	0.23	0	0
0	0	0	0.97	0.03	0
0	0	0	0	0.95	0.05
0	0	0	0	0	1.00

2. **GMM-HMM** (6 states, 2 mixutre)

With the increase of dataset, Gaussian-HMM is not able to capture the timbre of different people, gender, age.

We can use GMM-HMM model instead.



"kaimen"

However, GMM-HMM is sensitive to the initial parameters. The tricks we are using in this example is:

1. Use Gaussian-HMM to train different people separately.
2. Combine the data point which are in the same state. Use GMM to initialize the parameters.
3. Run GMM-HMM to fine-tune the model.

Results

Dataset: 20 Isolated Chinese words. 11 male + 9male. Altogether 800 pronunciations.
10 male and 9 female for training. 10 male and 10 female for testing.

Feature: 12 dimension MFCC

Model: Gaussian-HMM, GMM-HMM

Model	Accuracy
Gaussian-HMM	82.25%
GMM-HMM	84.00%

Weakness of HMM

1. Markov assumption

The next state is only dependent upon the current state. So is poor at capturing long-range correlations between the observed variables.

$$p(\mathbf{z}_{n+1} | \mathbf{z}_1, \ldots, \mathbf{z}_n) = p(\mathbf{z}_{n+1} | \mathbf{z}_n) \tag{58}$$

2. Stationary assumption

$$p(\mathbf{z}_{n+1} | \mathbf{z}_n) = p(\mathbf{z}_n | \mathbf{z}_{n-1}) \tag{59}$$

3. Output independence assumption

The current output is conditionally independent of the previous output.

$$p(\mathbf{X} | \mathbf{Z}) = \prod_{n=1}^{N} p(\mathbf{x}_n | \mathbf{Z}) \tag{60}$$

Tricks of HMM

1. HMM is more sensitive to the initial parameters than GMM. So it is easy to get into local minimum.

Solve: Use GMM or other methods to initialize parameters.
Initialize parameters randomly and run HMM separately for several times.

2. For Gaussian-HMM & GMM-HMM, if eig($\mathbf{\Sigma}$) is too small. Then
   $\mathbf{\Sigma}^{-1}$ will be unstable.

Solve: if eig($\mathbf{\Sigma}$) < ε then $\mathbf{\Sigma} = \mathbf{\Sigma} + \sigma\mathbf{I}$

3. If p(**x**|**z**) is Gaussian or GMM pdf, underflow may occurs.

$$p(\mathbf{x} \mid \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{D/2} |\boldsymbol{\Sigma}|^{1/2}} \exp\left(\underline{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^{\mathrm{T}} \boldsymbol{\Sigma}^{-1} (\mathbf{x}-\boldsymbol{\mu})} \right)$$

if this term is too small,
after exp it will underflow

Solve: use ln p(**x**|**z**) to instead p(**x**|**z**) in code implementation.

For Gaussian

$$\ln p(\mathbf{x}|\mathbf{z}) = \ln \pi - \frac{D}{2}\ln(2\pi) - \frac{1}{2}\ln|\boldsymbol{\Sigma}| - \frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^{\mathrm{T}} \boldsymbol{\Sigma}^{-1} (\mathbf{x}-\boldsymbol{\mu})$$

For GMM

$$\begin{aligned}
\ln p(\mathbf{x}|\mathbf{z}) &= \ln \sum_{m=1}^{m} \pi_m N(\mathbf{x}|\boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m) \\
&= \ln \sum_{m=1}^{m} \exp\left(\underline{\ln \pi_m - \frac{D}{2}\ln(2\pi) - \frac{1}{2}\ln|\boldsymbol{\Sigma}_m| - \frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_m)^{\mathrm{T}} \boldsymbol{\Sigma}_m^{-1} (\mathbf{x}-\boldsymbol{\mu}_m)} \right) \\
&= \left[\ln \sum_{m=1}^{m} \exp\left(\ln \pi_m - \frac{D}{2}\ln(2\pi) - \frac{1}{2}\ln|\boldsymbol{\Sigma}_m| - \frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_m)^{\mathrm{T}} \boldsymbol{\Sigma}_m^{-1} (\mathbf{x}-\boldsymbol{\mu}_m) - U \right) \right] + U
\end{aligned}$$

if this term is too small,
after exp it will underflow

Normalization factor,
To avoid underflow

4. According to (11), (15), $\widehat{\alpha}(\mathbf{z}_n), \widehat{\beta}(\mathbf{z}_n)$ may underflow, and $\gamma(\mathbf{z}_n)$ will be unstable

Solve: Use $\ln\gamma(\mathbf{z}_n), \ln c_n, \ln\widehat{\alpha}(\mathbf{z}_n), \ln\widehat{\beta}(\mathbf{z}_n)$ replace $\gamma(\mathbf{z}_n), c_n, \widehat{\alpha}(\mathbf{z}_n), \widehat{\beta}(\mathbf{z}_n)$

$$\ln\widehat{\alpha}(\mathbf{z}_n) = \underline{-\ln c_n + \ln p(\mathbf{x}_n|\mathbf{z}_n) + \ln\sum_{\mathbf{z}_{n-1}}\exp\Big(\ln\widehat{\alpha}(\mathbf{z}_{n-1}) + \ln p(\mathbf{z}_n|\mathbf{z}_{n-1})\Big)}$$

if this term is too small, after exp it will underflow

$$= -\ln c_n + \ln p(\mathbf{x}_n|\mathbf{z}_n) + \left[\ln\sum_{\mathbf{z}_{n-1}}\exp\Big(\ln\widehat{\alpha}(\mathbf{z}_{n-1}) + \ln p(\mathbf{z}_n|\mathbf{z}_{n-1}) - U\Big)\right] + U$$

Normalization factor, To avoid underflow

The same strategy can be applied to $\ln c_n, \ln\widehat{\beta}(\mathbf{z}_n)$

$\gamma(\mathbf{z}_n) = \widehat{\alpha}(\mathbf{z}_n)\widehat{\beta}(\mathbf{z}_n)$ will turn to

$$\ln\gamma(\mathbf{z}_n) = \ln\widehat{\alpha}(\mathbf{z}_n) + \ln\widehat{\beta}(\mathbf{z}_n)$$

Furthermore, for parameter estimation, such as (28) will turn to

$$\boldsymbol{\mu}_k = \frac{\sum_{n=1}^{N} \gamma_{nk} \mathbf{x}_n}{\sum_{n=1}^{N} \gamma_{nk}} = \frac{\sum_{n=1}^{N} \exp[\ln \gamma_{nk}] \mathbf{x}_n}{\sum_{n=1}^{N} \exp[\underline{\ln \gamma_{nk}}]} = \frac{\sum_{n=1}^{N} \exp[(\ln \gamma_{nk}) - U] \mathbf{x}_n}{\sum_{n=1}^{N} \exp[(\ln \gamma_{nk}) - U]}$$

if this term is too small, after exp it will underflow

Normalization factor, To avoid underflow

Matlab Code

https://github.com/qiuqiangkong/matlab-hmm

THANK YOU!