Siyu Chen
schen4@andrew.cmu.edu

HMM Review

Contents

1 Fundamentals

1.1 Variable Definition

- $\pi_i = p(s_1 = i \mid \Theta)$

  initial transition probability, denoting the probability of each hidden state in the first time step

- $a_{i,j} = p(s_{t-1} = i, s_t = j \mid \Theta)$

  (official) transition probability, denoting the probability of to transit from state i to state j ($2 \leq t \leq T$)

  the compound matrix $A \in \mathbb{R}^{J \times J}$ satisfies the condition that $A.\text{sum}(\text{axis} = 1) = \sum A_{i,:} = 1$ (that all the outgoing probabilities from the same current state i sum up to 1)

- $B_t(s_t) = p(\mathbf{o}_t \mid s_t, \Theta) \xrightarrow{\text{simple diagonal Gaussian}} \mathcal{N}(\mathbf{o}_t \mid \mu_{s_t}, \mathbf{r}_{s_t})$

  emission probability, equivalent to $p(\mathbf{o}_t \mid s_t = j, \Theta) = \mathcal{N}(\mathbf{o}_t \mid \mu_j, \mathbf{r}_j)$, which is a function with parameters determined by specified mean values and variances, and a function output is generated for a specific input of observation state.

  [classical setting] in the most fundamental version, the emission matrix is shared among all time steps and hence $B_t(s_t)$ can be simplied to B_{s_t}. B_{s_t} is a vector same shaped as the number of observation states (eg. if $\mathbf{o}_t \in [L]$, then $B_{s_t} \in \mathbb{R}^L$ or $B \in \mathbb{R}^{J \times L}$.

- $\Theta = (A, B, \pi)$

  all the parameters above are together denoted as Θ

- $\gamma_t(i) = p(s_t = i \mid O, \Theta)$

  probability of hidden state at time step t being i given observation sequence O

- $\xi_{t-1}(i, j) = p(s_{t-1} = i, s_t = j \mid O, \Theta)$

  joint probability of 2 continuous hidden states given an output sequence

- $\alpha_t(i) = p(O_{1:t}, s_t = i \mid \Theta)$

  FORWARD joint probability, joint of both hidden states and observations

- $\beta_t(i) = p(O_{t+1:T} \mid s_t = i, \Theta)$

  BACKWARD conditional probability, joint of both hidden states and observations

1.2 Diagonal Gaussian Distribution

The general form of the probabilistic distribution function (pdf) of a multi-variate (say, D-dimensional) Gaussian distribution can be given by:

$$\mathcal{N}(\mathbf{o} \mid \mu, \mathbf{\Sigma}) = \frac{1}{(2\pi)^{\frac{D}{2}} |\mathbf{\Sigma}|^{\frac{1}{2}}} e^{-\frac{1}{2}(\mathbf{o}-\mu)^T \mathbf{\Sigma}^{-1} (\mathbf{o}-\mu)}$$

where $\mathbf{\Sigma}$ is the covariance matrix, $|\cdot|$ refers to the determinant, which in the case of diagonal matrix is the product of all diagonal elements, and $(\cdot)^{-1}$ takes the inverse.

1.2.1 Dimensionality Check

For the determinant of the first factor, $(2\pi)^{D/2}$ is a scalar, $|\mathbf{\Sigma}|$ is a scalar, resulting this entire factor $\frac{1}{(2\pi)^{D/2}|\mathbf{\Sigma}|^{1/2}}$ to be a scalar as well.

For the exponential power, $\mathbf{o}-\mu$ is a D-dimensional vector, and the inverse of $\mathbf{\Sigma}$, $\mathbf{\Sigma}^{-1}$, is a $D\times D$ matrix. Therefore, the matrix multiplication results in a scalar.

All packed together, pdf of D-dimensional multivariate diagonal Gaussian distribution can be defined on the following mapping of domains:

$$\mathbb{R}^D \to \mathbb{R}$$

1.2.2 Derivatives (Diagonal Gaussian Only)

To faster utilize the chain rule in derivative computation, we'd summarize the derivatives to μ and $\mathbf{r}$ here for future convenience:

$$\forall d \in [D]: \quad \frac{\partial \mathcal{N}(\mathbf{o} \mid \mu, \mathbf{\Sigma})}{\partial \mu_d} = \mathcal{N}(\mathbf{o} \mid \mu, \mathbf{\Sigma}) \cdot [-\frac{2(o_d - \mu_d)}{2\Sigma_{dd}}] = \mathcal{N}(\mathbf{o} \mid \mu, \mathbf{\Sigma}) \cdot \frac{o_d - \mu_d}{\Sigma_{dd}}$$

$$\forall d \in [D]: \quad \frac{\partial \mathcal{N}(\mathbf{o} \mid \mu, \mathbf{\Sigma})}{\partial \mathbf{r}_d} = \mathcal{N}(\mathbf{o} \mid \mu, \mathbf{\Sigma}) \cdot [-\frac{2(o_d - \mu_d)}{2\Sigma_{dd}}] = \frac{1}{2\Sigma_{dd}} \cdot \mathcal{N}(\mathbf{o} \mid \mu, \mathbf{\Sigma}) \left(\frac{(o_d - \mu_j)^2}{\Sigma_{dd}} - 1 \right)$$

1.2.3 Transformation onto Log-Domain

Since the probabilities usually end up being super hyper doper small, it's reasonable to transform them, along with all relevant multiplications in joint probability computation, to the log domain with simple summations.

Try taking log values to the pdf:

$$\begin{aligned}
\log\left(\mathcal{N}(\mathbf{o} \mid \mu, \mathbf{\Sigma})\right) &= \log\left(e^{-\frac{1}{2}(\mathbf{o}-\mu)^T \mathbf{\Sigma}^{-1}(\mathbf{o}-\mu)}\right) - \log\left((2\pi)^{\frac{D}{2}} |\mathbf{\Sigma}|^{\frac{1}{2}}\right) \\
&= -\frac{1}{2}(\mathbf{o}-\mu)^T \mathbf{\Sigma}^{-1}(\mathbf{o}-\mu) - \frac{D}{2}\log(2\pi) - \frac{1}{2}\log|\mathbf{\Sigma}| \\
&= -\frac{1}{2}\left(\sum_{i=1}^{D}\sum_{j=1}^{D} \mathbf{\Sigma}^{-1}_{ij} \cdot (\mathbf{o}-\mu)_i \cdot (\mathbf{o}-\mu)_j\right) - \frac{D}{2}\log(2\pi) - \frac{1}{2}\log|\mathbf{\Sigma}|
\end{aligned}$$

Then for simple diagonal Gaussian distributions, its inverse is obtained by purely replacing every diagonal element by its reciprocal and hence still diagonal, while the determinant is also easily obtained by multiplying diagonal elements all together, the log-pdf can be further simplified to:

$$\begin{aligned}
\log\left(\mathcal{N}(\mathbf{o} \mid \mu, \mathbf{\Sigma})\right) &= -\frac{1}{2}\left(\sum_{i=1}^{D}\sum_{j=1}^{D} \mathbf{\Sigma}^{-1}_{ij} \cdot (\mathbf{o}-\mu)_i \cdot (\mathbf{o}-\mu)_j\right) - \frac{D}{2}\log(2\pi) - \frac{1}{2}\log|\mathbf{\Sigma}| \\
&= -\frac{1}{2}\left(\sum_{i=1}^{D} \mathbf{\Sigma}^{-1}_{ii} \cdot (\mathbf{o}-\mu)_i^2\right) - \frac{D}{2}\log(2\pi) - \frac{1}{2}\log|\mathbf{\Sigma}| \\
&\xrightarrow{\text{simple diagonal Gaussian}} -\frac{1}{2}\left(\sum_{i=1}^{D} \frac{(\mathbf{o}-\mu)_i^2}{\mathbf{\Sigma}_{ii}}\right) - \frac{D}{2}\log(2\pi) - \frac{1}{2}\sum_{i=1}^{D}\log\mathbf{\Sigma}_{ii} \\
&= -\frac{1}{2}\left(\sum_{d=1}^{D} \frac{(\mathbf{o}-\mu)_d^2}{\mathbf{r}_d}\right) - \frac{D}{2}\log(2\pi) - \frac{1}{2}\sum_{i=1}^{D}\log\mathbf{r}_d
\end{aligned}$$

1.2.4 Optimization

To find the best μ and $\mathbf{\Sigma}$ for a specific output o, the objective is to maximize the conditional probability, which can be seen as the "likelihood". Modify the single-observation case to multiple observations:

$$\sum_{t=1}^{T} \log\left(\mathcal{N}(\mathbf{o}_t \mid \mu, \mathbf{\Sigma})\right) = -\frac{1}{2}\left(\sum_{t=1}^{T}\sum_{d=1}^{D} \frac{(\mathbf{o}_t - \mu)_d^2}{\mathbf{r}_d}\right) - \frac{TD}{2}\log(2\pi) - \frac{T}{2}\sum_{i=1}^{D}\log \mathbf{r}_d$$

Then without detailed proof of convexity, the optimal solutions all have a 1st order derivative of 0.

$$\frac{\partial \sum_{t=1}^{T} \log\left(\mathcal{N}(\mathbf{o} \mid \mu, \mathbf{\Sigma})\right)}{\partial \mu_d} = -\frac{1}{2}\cdot\frac{\sum_{t=1}^{T} 2(o_{t,d} - \mu_d)}{\mathbf{r}_d} = 0 \quad \rightarrow \quad \hat{\mu}_d = \frac{\sum_{t=1}^{T} o_{t,d}}{T}$$

$$\frac{\partial \sum_{t=1}^{T} \log\left(\mathcal{N}(\mathbf{o} \mid \mu, \mathbf{\Sigma})\right)}{\partial \mathbf{r}_d} = \frac{1}{2}\cdot\frac{\sum_{t=1}^{T}(o_{t,d} - \mu_d)^2}{\mathbf{r}_d^2} - \frac{T}{2}\cdot\frac{1}{\mathbf{r}_d} = 0 \quad \rightarrow \quad \hat{\mathbf{r}}_d = \frac{\sum_{t=1}^{T}(o_{t,d} - \hat{\mu}_d)^2}{T}$$

which are just the mean value and variance of the observation sequence.

2 Derivations: Hidden States Given Full Observations

2.1 1-Step Hidden: $\gamma_t(i) = p(s_t = i \mid O, \Theta)$

By definition we'll have:

$$\gamma_t(i) = p(s_t = i \mid O, \Theta) = \frac{p(s_t = i, O \mid \Theta)}{p(O \mid \Theta)}$$

Given the fact that by multiplying *forward joint* and *backward conditional* probability, we'll have:

$$\begin{aligned}
\alpha_t(i) \cdot \beta_t(i) &= p(O_{1:t}, s_t = i \mid \Theta) \cdot p(O_{t+1:T} \mid s_t = i, \Theta) \\
&= p(O_{1:t}, s_t = i \mid \Theta) \cdot p(O_{t+1:T} \mid s_t = i, O_{1:t}, \Theta) && \text{(Conditional Independence Assumption)} \\
&= p(s_t = i, O \mid \Theta)
\end{aligned}$$

We could hence come to the conclusion that:

$$\gamma_t(i) = p(s_t = i \mid O, \Theta) = \frac{\alpha_t(i) \cdot \beta_t(i)}{p(O \mid \Theta)} = \frac{\alpha_t(i) \cdot \beta_t(i)}{\sum_{j=1}^{J} \alpha_T(j)}$$

2.2 2-Step Joint Hidden: $\xi_{t-1}(i, j) = p(s_{t-1} = i, s_t = j \mid O, \Theta)$

$$\begin{aligned}
\xi_{t-1}(i, j) &= p(s_{t-1} = i, s_t = j \mid O, \Theta) \\
&= \frac{p(O \mid s_{t-1} = i, s_t = j, \Theta) \cdot p(s_{t-1} = i, s_t = j \mid \Theta)}{p(O \mid \Theta)} && \text{(Bayesian Rule)} \\
&\propto p(O \mid s_{t-1} = i, s_t = j, \Theta) \cdot p(s_{t-1} = i, s_t = j \mid \Theta) \\
&= p(O_{1:t-1}, \mathbf{o}_t, O_{t+1:T} \mid s_{t-1} = i, s_t = j, \Theta) \cdot p(s_{t-1} = i, s_t = j \mid \Theta) && \text{(Sequence Decomposition)} \\
&= p(O_{1:t-1} \mid s_{t-1} = i, s_t = j, \Theta)) \cdot p(\mathbf{o}_t \mid s_{t-1} = i, s_t = j, \Theta) \\
&\quad \cdot p(O_{t+1:T} \mid s_{t-1} = i, s_t = j, \Theta) \cdot p(s_{t-1} = i, s_t = j \mid \Theta)
\end{aligned}$$

Think of the **1st order markov assumption**, where the current hidden state is dependent on only 1 previous hidden state, which is:

$$\forall t \in \{2, ..., T\}: \quad p(s_t \mid S_{1:t-1}) = p(s_t \mid s_{t-1})$$

And also the **output independence assumption**, where the observation at one time step is only dependent on its current hidden state:

$$p(\mathbf{o}_t \mid O, S) = p(\mathbf{o}_t \mid s_t)$$

Therefore we continue the transformation of $\xi_{t-1}(i, j)$:

$$\begin{aligned}
\xi_{t-1}(i, j) &\propto p(O_{1:t-1} \mid s_{t-1} = i, s_t = j, \Theta)) \cdot p(\mathbf{o}_t \mid s_{t-1} = i, s_t = j, \Theta) \\
&\quad \cdot p(O_{t+1:T} \mid s_{t-1} = i, s_t = j, \Theta) \cdot p(s_{t-1} = i, s_t = j \mid \Theta) \\
&= p(O_{1:t-1} \mid s_{t-1} = i, \Theta)) \cdot p(\mathbf{o}_t \mid s_t = j, \Theta) \cdot p(O_{t+1:T} \mid s_{t-1} = i, s_t = j, \Theta) \cdot p(s_{t-1} = i, s_t = j \mid \Theta)
\end{aligned}$$

Leveraging conditional independence based on Bayes Net, we know that a sequence of observations are conditionally independent from previous hidden states given the latest one, or mathematically speaking:

$$\forall 1 \leq \tau \leq t + 1: \quad p(O_{t+1:T} \mid s_{1:\tau}, \Theta) = p(O_{t+1:T} \mid s_\tau, \Theta)$$

(intuitively understandable using D-separation of Bayes Net)

Therefore we continue to obtain the following transformation:

$$
\begin{aligned}
\xi_{t-1}(i,j) &\propto p(O_{1:t-1} \mid s_{t-1}=i,\Theta)) \cdot p(\mathbf{o}_t \mid s_t=j,\Theta) \cdot p(O_{t+1:T} \mid s_{t-1}=i, s_t=j,\Theta) \cdot p(s_{t-1}=i, s_t=j \mid \Theta) \\
&= p(O_{1:t-1} \mid s_{t-1}=i,\Theta)) \cdot p(\mathbf{o}_t \mid s_t=j,\Theta) \cdot p(O_{t+1:T} \mid s_t=j,\Theta) \cdot p(s_{t-1}=i, s_t=j \mid \Theta) \\
&= p(O_{1:t-1} \mid s_{t-1}=i,\Theta)) \cdot p(\mathbf{o}_t \mid s_t=j,\Theta) \cdot p(O_{t+1:T} \mid s_t=j,\Theta) \cdot p(s_t=j \mid s_{t-1}=i,\Theta) \cdot p(s_{t-1}=i \mid \Theta) \\
&= p(O_{1:t-1} \mid s_{t-1}=i,\Theta)) \cdot p(s_{t-1}=i \mid \Theta) \cdot p(\mathbf{o}_t \mid s_t=j,\Theta) \cdot p(O_{t+1:T} \mid s_t=j,\Theta) \cdot p(s_t=j \mid s_{t-1}=i,\Theta) \\
&= p(O_{1:t-1}, s_{t-1}=i \mid \Theta) \cdot p(\mathbf{o}_t \mid s_t=j,\Theta) \cdot p(O_{t+1:T} \mid s_t=j,\Theta) \cdot p(s_t=j \mid s_{t-1}=i,\Theta)
\end{aligned}
$$

Further breaking the transformation down we'll have:

$$
\begin{aligned}
\xi_{t-1}(i,j) &= p(s_{t-1}=i, s_t=j \mid O,\Theta) \\
&\propto p(O_{1:t-1}, s_{t-1}=i \mid \Theta) && \alpha_{t-1}(i)\text{: joint prob of hidden \& prev [obs]} \\
&\quad \cdot p(s_t=j \mid s_{t-1}=i,\Theta) && a_{i,j}\text{: transition probability from state } i \text{ to } j \\
&\quad \cdot p(\mathbf{o}_t \mid s_t=j,\Theta) && B_t(j)\text{: emission probability for time step } t \text{ and hidden state } s_t=j \\
&\quad \cdot p(O_{t+1:T} \mid s_t=j,\Theta) && \beta_t(j)\text{: conditional prob of future [obs] given current hidden } s_t=j
\end{aligned}
$$

Our long omitted denominator can also be reformed by summing up all potential hidden states s_{t-1} and s_t, which actually sums up all potential possibilities of the numerator:

$$
p(O \mid \Theta) = \sum_{s_T} p(O_{1:T}, s_T \mid \Theta) = \sum_{j=1}^{J} \alpha_T(j)
$$

Finally by taking the omitted denominator back, we'll obtain:

$$
\begin{aligned}
\xi_{t-1}(i,j) &= \frac{\alpha_{t-1}(i) \cdot a_{i,j} \cdot B_t(j) \cdot \beta_t(j)}{p(O \mid \Theta)} \\
&= \frac{\alpha_{t-1}(i) \cdot a_{i,j} \cdot B_t(j) \cdot \beta_t(j)}{\sum_{j=1}^{J} \alpha_T(j)} \\
&\xrightarrow{\text{simple diagonal Gaussian}} \frac{\alpha_{t-1}(i) \cdot a_{i,j} \cdot \mathcal{N}(\mathbf{o}_t \mid \mu_j, \mathbf{r}_j) \cdot \beta_t(j)}{\sum_{j=1}^{J} \alpha_T(j)}
\end{aligned}
$$

3 Forward Propagation

Subject: forward joint probability $\alpha_t(i)$

3.1 Initial Time Step

When $t = 1$ we have:

$$\begin{aligned}\alpha_1(i) &= p(\mathbf{o}_1, s_1 = i \mid \Theta) = p(s_1 = i \mid \Theta) \cdot p(\mathbf{o}_1 \mid s_1 = i, \Theta) \\ &= \pi_i \cdot B_{1,i} \\ &\xrightarrow{\text{simple diagonal Gaussian}} \pi_i \cdot \mathcal{N}(\mathbf{o}_1 \mid \mu_i, \mathbf{r}_i)\end{aligned}$$

3.2 Subsequent Time Steps

Hence when $t \geq 2$ and t increments, let's find an iterative computation rule given all the defined variables.

$$\begin{aligned}\alpha_t(i) &= p(O_{1:t}, s_t \mid \Theta) \\ &= \sum_{j=1}^{J} p(O_{1:t-1}, \mathbf{o}_t, s_{t-1} = j, s_t = i \mid \Theta) && \text{(Marginal Probability)} \\ &= \sum_{j=1}^{J} p(O_{1:t-1} \mid \mathbf{o}_t, s_{t-1} = j, s_t = i, \Theta) \cdot p(\mathbf{o}_t, s_{t-1} = j, s_t = i \mid \Theta \\ &= \sum_{j=1}^{J} p(O_{1:t-1} \mid s_{t-1} = j, \Theta) \cdot p(\mathbf{o}_t \mid s_{t-1} = j, s_t = i, \Theta) \cdot p(s_t = i \mid s_{t-1} = j, \Theta) \cdot p(s_t = i \mid \Theta) \\ &= \sum_{j=1}^{J} p(O_{1:t-1} \mid s_{t-1} = j, \Theta) \cdot p(s_t = i \mid \Theta) \cdot p(\mathbf{o}_t \mid s_t = i, \Theta) \cdot p(s_t = i \mid s_{t-1} = j, \Theta) \\ &= p(\mathbf{o}_t \mid s_t = i, \Theta) \cdot \sum_{j=1}^{J} p(O_{1:t-1}, s_{t-1} = j \mid \Theta) \cdot p(s_t = i \mid s_{t-1} = j, \Theta) \\ &= B_t(i) \cdot \sum_{j=1}^{J} \alpha_{t-1}(j) \cdot a_{j,i} \\ &\xrightarrow{\text{simple diagonal Gaussian}} \mathcal{N}(\mathbf{o}_t \mid \mu_i, \mathbf{r}_i) \cdot \sum_{j=1}^{J} \alpha_{t-1}(j) \cdot a_{j,i}\end{aligned}$$

Intuitively, we can consider this derivation as *all possibilities of previous hidden states* (why summed up along $j \in \{1..J\}$), *given the necessary transition probabilities as the bridge* (from state j to state i: $a_{j,i}$), with *the affiliated emission at time step t* (simple Gaussian for $\mathbf{o}_t$).

3.3 Final Derivation: Observation Sequence Likelihood

This computation leads to a very cool conclusion, is that given:

$$\forall i \in 1, ..., J: \quad \alpha_T(i) = p(O_{1:T}, s_T = i \mid \Theta)$$

By summing them up together utilizing the marginal probability again, we'll have the *joint probability of observations from start to end*, or to put it in human-understandable English, it is the **probability of a specific observation sequence**:

$$p(O \mid \Theta) = \sum_{i=1}^{J} p(O_{1:T}, s_T = i \mid \Theta) = \sum_{i=1}^{J} \alpha_T(i)$$

meaning that we have the following assertion to validate:

$$\forall t \in [T]: \quad p(O \mid \Theta) = (\vec{\alpha_t} \odot \vec{\beta_t}).\text{sum}()$$

3.4 Matrix Demonstration

Previously $\alpha_t(i)$ has been isolated by a specific state i. To concatenate all potential states in one vector, we'll have:

$$\begin{aligned}\vec{\alpha_t} &= \begin{bmatrix}\alpha_t(1)\\ \alpha_t(2)\\ \dots\\ \alpha_t(J)\end{bmatrix} = \begin{bmatrix}\mathcal{N}(\mathbf{o}_t \mid \mu_1, \mathbf{r}_1) \cdot \sum_{j=1}^{J} \alpha_{t-1}(j) \cdot a_{j,1}\\ \mathcal{N}(\mathbf{o}_t \mid \mu_2, \mathbf{r}_2) \cdot \sum_{j=1}^{J} \alpha_{t-1}(j) \cdot a_{j,2}\\ \dots\\ \mathcal{N}(\mathbf{o}_t \mid \mu_J, \mathbf{r}_J) \cdot \sum_{j=1}^{J} \alpha_{t-1}(j) \cdot a_{j,J}\end{bmatrix}\\ &= \begin{bmatrix}\mathcal{N}(\mathbf{o}_t \mid \mu_1, \mathbf{r}_1)\\ \mathcal{N}(\mathbf{o}_t \mid \mu_2, \mathbf{r}_2)\\ \dots\\ \mathcal{N}(\mathbf{o}_t \mid \mu_J, \mathbf{r}_J)\end{bmatrix} \odot \left(A^T \cdot \vec{\alpha_{t-1}}\right)\end{aligned}$$

Denote the first vector as an emission probability vector depending on both mean & variances and observation at a certain time step, B_t, we'll hence obtain the vector/matrix form of our previous recursion formula:

$$\vec{\alpha_t} = B_t \odot \left(A^T \cdot \vec{\alpha_{t-1}}\right)$$

Intuition (again): first accumulate **transition** along all possibilities, then apply **emission** element-wise.

3.5 Log-Likelihood

$$\log \alpha_t(i) = \log B_t(i) + \log \sum_{j=1}^{J} \alpha_{t-1}(j) \cdot a_{j,i} = \log B_t(i) + \underset{j}{\text{logsumexp}}\left(\log \alpha_{t-1}(j) + \log a_{j,i}\right)$$

which in vector/matrix format becomes:

$$\log \vec{\alpha_t} = \log \vec{B_t} + \text{logsumexp}\left(\log \vec{\alpha}_{t-1} + \log A, \text{axis=0}\right)^T$$

4 Backward Propagation

Subject: conditional probability $\beta_t(i) = p(O_{t+1:T} \mid s_t = i)$

4.1 "Initial Step": When $t = T$

For backward probabilities, the story starts from the last time step, and by default we let them all be 1s:

$$\text{when } t = T, \forall i \in [J]: \quad \beta_T(i) = p(O_{T+1:T} \mid s_T = i, \Theta) = 1$$

Compared to having some certain meanings, this is more like some placeholders, since $O_{T+1:T}$ does not have practical meanings.

4.2 Reversed Recursion

For all $1 \leq t \leq T-1$ we have:

$$\begin{aligned}
\beta_t(i) &= p(O_{t+1:T} \mid s_t = i, \Theta) = p(\mathbf{o}_{t+1}, O_{t+2:T} \mid s_t = i, \Theta) \\
&= \sum_{j=1}^{J} p(\mathbf{o}_{t+1}, O_{t+2:T}, s_{t+1} = j \mid s_t = i, \Theta) && \text{(Marginal Probability)} \\
&= \sum_{j=1}^{J} p(\mathbf{o}_{t+1} \mid O_{t+1:T}, s_{t+1} = j, s_t = i, \Theta) \cdot p(O_{t+2:T} \mid s_{t+1} = j, s_t = i, \Theta) \cdot p(s_{t+1} = j \mid s_t = i, \Theta) \\
&= \sum_{j=1}^{J} p(\mathbf{o}_{t+1} \mid s_{t+1} = j, \Theta) \cdot p(O_{t+2:T} \mid s_{t+1} = j, \Theta) \cdot p(s_{t+1} = j \mid s_t = i, \Theta) \\
&= \sum_{j=1}^{J} p(\mathbf{o}_{t+1} \mid s_{t+1} = j, \Theta) \cdot p(O_{t+2:T} \mid s_{t+1} = j, \Theta) \cdot p(s_{t+1} = j \mid s_t = i, \Theta) \\
&= \sum_{j=1}^{J} B_{t+1}(j) \cdot \beta_{t+1}(j) \cdot a_{i,j} \\
&\xrightarrow{\text{simple diagonal Gaussian}} \sum_{j=1}^{J} \mathcal{N}(\mathbf{o}_{t+1} \mid \mu_j, \mathbf{r}_j) \cdot \beta_{t+1}(j) \cdot a_{i,j}
\end{aligned}$$

4.3 Matrix Demonstration

Similarly if we utilize a vector to show $\beta_t(i)$ at time step t for all possible hidden states $i \in \{1..J\}$, we will then have:

$$\begin{aligned}
\vec{\beta_t} &= \begin{bmatrix} \beta_t(1) \\ \beta_t(2) \\ \dots \\ \beta_t(J) \end{bmatrix} = \begin{bmatrix} \sum_{j=1}^{J} B_{t+1}(j) \cdot \beta_{t+1}(j) \cdot a_{1,j} \\ \sum_{j=1}^{J} B_{t+1}(j) \cdot \beta_{t+1}(j) \cdot a_{2,j} \\ \dots \\ \sum_{j=1}^{J} B_{t+1}(j) \cdot \beta_{t+1}(j) \cdot a_{J,j} \end{bmatrix} = \sum_{j=1}^{J} B_{t+1}(j) \cdot \beta_{t+1}(j) \cdot \begin{bmatrix} a_{1,j} \\ a_{2,j} \\ \dots \\ a_{J,j} \end{bmatrix} \\
&= A \cdot \left(\vec{\beta_{t+1}} \odot B_{t+1} \right)
\end{aligned}$$

When specifying "hidden-state-to-observation" emission matrix B to be a series of simple Gaussian distributions, the above recursive computation is transformed to:

$$\vec{\beta}_t = A \cdot \left(\vec{\beta}_{t+1} \odot \begin{bmatrix} \mathcal{N}(\mathbf{o}_{t+1} \mid \mu_1, \mathbf{r}_1) \\ \mathcal{N}(\mathbf{o}_{t+1} \mid \mu_2, \mathbf{r}_2) \\ \dots \\ \mathcal{N}(\mathbf{o}_{t+1} \mid \mu_J, \mathbf{r}_J) \end{bmatrix} \right)$$

Intuition: the column of **emission**, through element-wise multiplication, add current step observation ($\mathbf{o}_t$) to the next-step $\vec{\beta}_{t+1}$, and then by summing up **transition** to all potential "outflowing" hidden states at time step $t + 1$, there's the probability of current hidden state.

4.4 Log-Likelihood

$$\log \beta_t(i) = \underset{j}{\text{logsumexp}} \left(\log B_{t+1}(j) + \log \beta_{t+1}(j) + \log a_{i,j} \right)$$

which in vector/matrix form becomes:

$$\log \vec{\beta}_t = \text{logsumexp} \left(\log \vec{B}_{t+1} + \log \vec{\beta}_{t+1} + \log A, \text{axis=1} \right)$$

5 Viterbi

Focused property: the Viterbi algorithm focuses on finding the most likely **hidden sequence** given the observation sequence:

$$\arg\max_{S_{1:t}} p(S_{1:t} \mid O_{1:t}, \Theta)$$

Consider that the observation sequence stays unchanged for one specific input, finding the maximum of this conditional probability is equivalent to maximizing the following joint probability:

$$\delta_t(j) = \max_{S_{1:t-1}} p(S_{1:t-1}, s_t = j, \mathbf{o}_{1:t} \mid \Theta)$$

Note that this is only the **most likely hidden sequence till time step** $t-1$, since the current hidden state is specified to $s_t = j$. To maximum the entire sequence, remember to acquire $\max_j \delta_t(j)$.

5.1 Initial State

When $t = 0$:

$$\begin{aligned}\delta_1(i) &= p(\mathbf{o}_1, s_1 = i \mid \Theta) = p(\mathbf{o}_1 \mid s_1 = i, \Theta) \cdot p(s_1 = i \mid \Theta) \\ &= \mathcal{N}(\mathbf{o}_1 \mid \mu_i, \mathbf{r}_i) \cdot \pi_i\end{aligned}$$

And the matrix form:

$$\begin{aligned}\delta_1 &= B \odot \pi \\ &\xrightarrow{\text{simple Gaussian}} \begin{bmatrix} \mathcal{N}(\mathbf{o}_1 \mid \mu_1, \mathbf{r}_1) \\ \mathcal{N}(\mathbf{o}_1 \mid \mu_2, \mathbf{r}_2) \\ \dots \\ \mathcal{N}(\mathbf{o}_1 \mid \mu_J, \mathbf{r}_J) \end{bmatrix} \odot \pi\end{aligned}$$

5.2 Subsequent Computation

As $1 \le t \le T$, we'll try to form the Viterbi computation as recursion as well:

$$\begin{aligned}\delta_t(i) &= \max_{S_{1:t-1}} p(S_{1:t-1}, s_t = i, O_{1:t} \mid \Theta) \\ &= \max_{S_{1:t-1}} p(S_{1:t-2}, s_{t-1}, s_t = i, O_{1:t-1}, \mathbf{o}_t \mid \Theta) && \text{(sequence decomposition)} \\ &= \max_{S_{1:t-1}} p(\mathbf{o}_t \mid s_t = i, s_{t-1}, S_{1:t-2}, O_{1:t-1}, \Theta) \cdot p(s_t = i \mid s_{t-1}, S_{1:t-2}, O_{1:t-1}, \Theta) \cdot p(s_{t-1}, S_{1:t-2}, O_{1:t-1} \mid \Theta) \\ &= \max_{S_{1:t-1}} p(\mathbf{o}_t \mid s_t = i, \Theta) \cdot p(s_t = i \mid s_{t-1}, \Theta) \cdot p(s_{t-1}, S_{1:t-2}, O_{1:t-1} \mid \Theta) \\ &= p(\mathbf{o}_t \mid s_t = i, \Theta) \cdot \max_{S_{1:t-2}, s_{t-1}} p(s_t = i \mid s_{t-1}, \Theta) \cdot p(s_{t-1}, S_{1:t-2}, O_{1:t-1} \mid \Theta) \\ &= p(\mathbf{o}_t \mid s_t = i, \Theta) \cdot \max_{S_{1:t-2}, j} p(s_t = i \mid s_{t-1} = j, \Theta) \cdot p(s_{t-1} = j, S_{1:t-2}, O_{1:t-1} \mid \Theta) \\ &= p(\mathbf{o}_t \mid s_t = i, \Theta) \cdot \max_j \left[p(s_t = i \mid s_{t-1} = j, \Theta) \cdot \max_{S_{1:t-2}} p(s_{t-1} = j, S_{1:t-2}, O_{1:t-1} \mid \Theta) \right] \\ &= B_t(i) \cdot \max_j \left(a_{ji} \cdot \delta_{t-1}(j) \right) \\ &\xrightarrow{\text{simple Gaussian}} \mathcal{N}(\mathbf{o}_t \mid \mu_i, \mathbf{r}_i) \cdot \max_j \left(a_{ji} \cdot \delta_{t-1}(j) \right)\end{aligned}$$

5.3 Matrix Demonstration

Similarly we can stack all $\delta_t(i)$ together to obtain a vector:

$$\vec{\delta}_t = \begin{bmatrix} B_t(1) \cdot \max\limits_j \left(a_{j,1} \cdot \delta_{t-1}(j)\right) \\ B_t(2) \cdot \max\limits_j \left(a_{j,2} \cdot \delta_{t-1}(j)\right) \\ \cdots \\ B_t(j) \cdot \max\limits_j \left(a_{j,J} \cdot \delta_{t-1}(j)\right) \end{bmatrix} = B_t \odot \left(A \times \vec{\delta}_{t-1}^{T}\right).\text{max(axis=1)}$$

where the symbol $\times$ denotes broadcasting, of the row vector $\vec{\delta}_{t-1}^{T}$ toward matrix A.

Again the **intuition** behind: $\vec{\delta}_{t-1}$ is all the most likely preceding sequence ending with s_{t-1}, then multiplying every transition probability $a_{:,i}$ will add the new hidden state $s_t = i$, hence paired with the emission probability of B_i, the probability transcends to its next time step.

5.3.1 Index-Of-The-Previous Matrix

In detailed computation, it's always reasonable to build a **index-of-previous-max** matrix for keeping track of the maximization flowing paths, for the sake of final backtracking of most likely hidden sequences. This is reasonable since

$$\arg\max_{S_{1:T}} p(S_{1:T} \mid O_{1:T}, \Theta)$$

is not obtained simply combining the calls of argmax() at every time step. Given below is a naive toy instance to strike some resonance:

$$\begin{bmatrix} 0.11 & 0.06 & 0.008 & 0.00095 \\ 0.10 & 0.21 & 0.012 & 0.00046 \\ 0.23 & 0.12 & 0.024 & 0.00003 \\ 0.78 & 0.32 & 0.107 & 0.00229 \end{bmatrix}$$

Suppose the best path obtained when backtracked is $\{s_4, s_1, s_3, s_4\}$, and clearly, step-wise argmax() leads to $\{s_4, s_4, s_4, s_4\}$.

It's more than obvious that **the most likely hidden sequence does not guarantee the largest likelihood at each time step, which means it is not obtainable by greedy search**.

This is similar to the text sequence scenario that "I live in San" does not make more sense compared to "I live in Vatican", but it's definitely more frequent to have people say "I live in San Francisco" than "I live in Vatican City", if purely considering the co-occurrence counts.

Therefore, it's reasonable to construct the following matrix of dimension $PI \in \mathbb{R}^{J \times T}$, where $PI_{j,t}$ refers to the index in time step $t-1$ that leads to $\max\, p(S_{1:t-1}, s_t = j, O_{1:t} \mid \Theta)$:

$$\begin{bmatrix} N/A & \mathbf{4} & 3 & 2 \\ N/A & 2 & 4 & 4 \\ N/A & 2 & \mathbf{1s} & 2 \\ \mathbf{N/A} & 1 & 1 & \mathbf{3} \end{bmatrix}$$

6 Expectation Maximization and Baum-Welch Algorithm

6.1 Joint Probability

6.1.1 For a specific hidden state $s_t = i$

With previous variables we've defined, we can obtain the following joint probability that specifies the hidden state at time step t being i:

$$\begin{aligned} p(O_{1:T}, s_t = i \mid \Theta) &= p(O_{1:t}, s_t = i \mid \Theta) \cdot p(O_{t+1:T} \mid s_t = i, O_{1:t}, \Theta) \\ &= p(O_{1:t}, s_t = i \mid \Theta) \cdot p(O_{t+1:T} \mid s_t = i, \Theta) \\ &= \alpha_t(i) \cdot \beta_t(i) \end{aligned}$$

Then in the vector form, where we stack all potential hidden states together, we'll then have:

$$p(O_{1:T}, s_t \mid \Theta) = \alpha_t \odot \beta_t$$

6.1.2 For a hidden state sequence

It is also possible to compute the joint probability of a specific hidden state sequence given the learned parameters and observation sequence:

$$p(O_{1:T}, S_{1:T} \mid \Theta) = p(s_1 \mid \Theta) \prod_{t=2}^{T} p(s_t \mid s_{t-1}, \Theta) \cdot p(\mathbf{o}_t \mid s_t, \Theta) = \pi_{s_1} \prod_{t=2}^{T} a_{s_{t-1}, s_t} \cdot B_{s_t}$$

6.2 Cross Entropy & Maximum Parameter Likelihood

Note: the following induction from KL-divergence to the final auxiliary function is purely my **personal induction**. Feel free to correct me wherever I'm wrong.

Think of one crucial loss used in various ML models for classification tasks, **cross entropy**:

$$X_{\text{ent}}(P(x), Q(x)) = -\sum_{x \in \mathcal{X}} P(x) \log Q(x)$$

In the specific context of HMM, parameters are by nature probabilities, hence with fixed point iteration, the same parameter in 2 adjacent iterations can be considered as 2 different distributions as well:

$$X_{\text{ent}}(\omega^{(t)}, \omega^{(t-1)}) = -\omega^{(t)} \log \omega^{(t-1)}$$

Maximization of parameter log-likelihood is equivalent to:

$$\log \mathcal{L}(\hat{\omega}) = \log \left(\prod_i \hat{\omega}_i^{\text{true count of } i} \right) \propto \sum_i \frac{\text{true count of } i}{N} \log \hat{\omega}_i = \sum_i \omega_i \log \hat{\omega}_i$$

So we can conclude that when flipping the sign of cross entropy, it is equivalent to parameter likelihood and hence can be used as an objective function.

6.3 Objective: Auxiliary Function

Denote previous parameters as Θ' and current ones Θ. In order to improve output results gradually through more iterations, think of the newer parameters as the (closer approximation to) groundtruths. Therefore, we declare to use the following auxiliary function as our optimization target:

$$Q(\Theta, \Theta') = \text{some\_}P(\Theta) \cdot \log \text{same\_}P(\Theta')$$

Then using the joint probability of **observations** and **hidden states**, we have:

$$\begin{aligned} Q(\Theta, \Theta') &= \sum_{S} P(O, S \mid \Theta') \cdot \log\left[P(O, S \mid \Theta)\right] \\ &= \sum_{S} P(S \mid O, \Theta') \cdot P(O \mid \Theta') \cdot \log\left[P(O, S \mid \Theta)\right] \end{aligned}$$

Since in the current iteration, previous parameters Θ' is fixed and hence the observation sequence probability $P(O \mid \Theta')$ stays unchanged for all potential new Θs.

Thus we come up with the concise format of our final auxiliary objective function:

$$Q(\Theta, \Theta') = \sum_{S} P(S \mid O, \Theta') \cdot \log\left[P(O, S \mid \Theta)\right]$$

As for the joint probability of observations and hidden states given current parameters:

$$\begin{aligned} P(O, S \mid \Theta) &= p(\mathbf{o}_1 \mid s_1, S_{2:T}, O_{2:T}, \Theta) \cdot p(S_{2:T}, O_{2:T} \mid s_1, \Theta) \cdot p(s_1 \mid \Theta) && \text{(Product Rule)} \\ &= p(s_1 \mid \Theta) \cdot p(\mathbf{o}_1 \mid s_1, \Theta) \cdot p(O_{2:T}, S_{2:T} \mid s_1, \Theta) && \text{(Conditional Independence)} \\ &= \pi_{s_1} B_1(s_1) \cdot p(S_{3:T}, O_{2:T} \mid s_2, s_1, \Theta) \cdot p(s_2 \mid s_1, \Theta) && \text{(Product Rule)} \\ &= \pi_{s_1} B_1(s_1) \cdot a_{s_1, s_2}\, p(\mathbf{o}_2 \mid s_2, \Theta) \cdot p(S_{3:T}, O_{3:T} \mid s_2, \Theta) && \text{(2 Rules Combined)} \\ &= \pi_{s_1} B_1(s_1) \prod_{t=2}^{T} a_{s_{t-1}, s_t} \cdot B_t(s_t) && \text{(\textit{applied recursively})} \\ &\xrightarrow{\text{simple diagonal Gaussian}} \pi_{s_1} \mathcal{N}(\mathbf{o}_1 \mid \mu_{s_1}, \mathbf{r}_{s_1}) \prod_{t=2}^{T} a_{s_{t-1}, s_t} \cdot \mathcal{N}(\mathbf{o}_t \mid \mu_{s_t}, \mathbf{r}_{s_t}) \end{aligned}$$

Then using log-probability in the full maximum likelihood objective of $Q(\Theta, \Theta')$, we'll have:

$$\begin{aligned} Q(\Theta, \Theta') &= \sum_{S} P(S \mid O, \Theta') \cdot \log\left[P(O, S \mid \Theta)\right] \\ &= \sum_{S_{1:T}} P(S \mid O, \Theta') \cdot \log\left[\pi_{s_1} B_1(s_1) \prod_{t=2}^{T} a_{s_{t-1}, s_t} \cdot B_t(s_t)\right] \\ &= \sum_{S_{1:T}} P(S \mid O, \Theta') \cdot \log\left[\pi_{s_1} B_1(s_1)\right] && \text{(Initial States)} \quad (1) \\ &\quad + \sum_{S_{1:T}} P(S \mid O, \Theta') \cdot \sum_{t=2}^{T} \log a_{s_{t-1}, s_t} && \text{(Transitional Probabilities)} \quad (2) \\ &\quad + \sum_{S_{1:T}} P(S \mid O, \Theta') \cdot \sum_{t=2}^{T} \log B_t(s_t) && \text{(Emission Probabilities)} \quad (3) \end{aligned}$$

Then let's simplify the three items respectively. My personal understanding of how to apply the tricks, is to **firstly isolate time steps present in the log items**, where we can have the log moved to be a factor applied to the rest of items. **Then** we'll find out that whatever else has been specified in sequence composition would **end up summing to vanish** thanks to the incredible concept of marginal probability.

For **item (1)** of the **initial states**, both hidden and observations, it can be further transformed to:

$$\begin{aligned}
\sum_{S_{1:T}} p(S \mid O, \Theta') \cdot \log\left[\pi_{s_1} B_1(s_1)\right] &= \sum_{s_1, S_{2:T}} p(s_1, S_{2:T} \mid O, \Theta') \cdot \log\left[\pi_{s_1} B_1(s_1)\right] \\
&= \sum_{s_1} \log\left[\pi_{s_1} B_1(s_1)\right] \cdot \sum_{S_{2:T}} p(s_1, S_{2:T} \mid O, \Theta') \\
&= \sum_{s_1} \log\left[\pi_{s_1} B_1(s_1)\right] \cdot p(s_1 \mid O, \Theta') \\
&= \sum_{j=1}^{J} \log\left[\pi_{s_1} B_1(j)\right] \cdot \gamma_1(j)
\end{aligned}$$

For **item (2)** of the **transmission probabilities**, it can be further transformed to:

$$\begin{aligned}
\sum_{S_{1:T}} P(S \mid O, \Theta') \cdot \sum_{t=2}^{T} \log a_{s_{t-1}, s_t} &= \sum_{t=2}^{T} \left(\sum_{S_{1:t-2}, s_{t-1}, s_t, S_{t+1:T}} p(S_{1:t-2}, s_{t-1}, s_t, S_{t+1:T} \mid O, \Theta') \log a_{s_{t-1}, s_t} \right) \\
&= \sum_{t=2}^{T} \sum_{S_{1:t-2}, s_{t-1}, s_t, S_{t+1:T}} p(S_{1:t-2}, s_{t-1}, s_t, S_{t+1:T} \mid O, \Theta') \log a_{s_{t-1}, s_t} \\
&= \sum_{t=2}^{T} \sum_{s_{t-1}, s_t} \log a_{s_{t-1}, s_t} \sum_{S_{1:t-2}, S_{t+1:T}} p(S_{1:t-2}, s_{t-1}, s_t, S_{t+1:T} \mid O, \Theta') \\
&= \sum_{t=2}^{T} \sum_{s_{t-1}, s_t} \log a_{s_{t-1}, s_t} \cdot p(s_{t-1}, s_t \mid O, \Theta') \\
&= \sum_{t=2}^{T} \sum_{s_{t-1}} \sum_{s_t} p(s_{t-1}, s_t \mid O, \Theta') \cdot \log a_{s_{t-1}, s_t} \\
&= \sum_{t=2}^{T} \sum_{i=1}^{J} \sum_{j=1}^{J} p(s_{t-1} = i, s_t = j \mid O, \Theta') \cdot \log a_{i,j} \\
&= \sum_{t=2}^{T} \sum_{i=1}^{J} \sum_{j=1}^{J} \xi_{t-1}(i, j) \cdot \log a_{i,j}
\end{aligned}$$

For **item (3)** of the **emission probabilities**, it can be further transformed to:

$$\begin{aligned}
\sum_{S_{1:T}} P(S \mid O, \Theta') \cdot \sum_{t=2}^{T} \log B_t(s_t) &= \sum_{t=2}^{T} \left(\sum_{S_{1:t-1}, s_t, S_{t+1:T}} p(S_{1:t-1}, s_t, S_{t+1:T} \mid O, \Theta') \cdot \log B_t(s_t) \right) \\
&= \sum_{t=2}^{T} \sum_{s_t} \log B_t(s_t) \sum_{S_{1:t-1}, S_{t+1:T}} p(S_{1:t-1}, s_t, S_{t+1:T} \mid O, \Theta') \\
&= \sum_{t=2}^{T} \sum_{j=1}^{J} \log B_t(s_t) \cdot p(s_t = j \mid O, \Theta') \\
&= \sum_{t=2}^{T} \sum_{j=1}^{J} \gamma_t(j) \cdot \log B_t(j)
\end{aligned}$$

Finally replacing all the 3 items in the original auxiliary function objective, we come to the clearest form of what to optimize during the training of HMM:

$$Q(\Theta, \Theta') = \sum_{j=1}^{J} \gamma_1(j) \cdot \log\left[\pi_j \, B_1(j)\right] + \sum_{t=2}^{T} \sum_{i=1}^{J} \sum_{j=1}^{J} \xi_{t-1}(i,j) \cdot \log a_{i,j} \quad + \sum_{t=2}^{T} \sum_{j=1}^{J} \gamma_t(j) \cdot \log B_t(j)$$

...move $B_1(j)$ from item (1) to item (3)...

$$= \sum_{j=1}^{J} \gamma_1(j) \cdot \log \pi_j \qquad \text{(Initial States)}$$

$$+ \sum_{t=2}^{T} \sum_{i=1}^{J} \sum_{j=1}^{J} \xi_{t-1}(i,j) \cdot \log a_{i,j} \qquad \text{(Transition Probabilities)}$$

$$+ \sum_{t=1}^{T} \sum_{j=1}^{J} \gamma_t(j) \cdot \log B_t(j) \qquad \text{(Emission Probabilities)}$$

6.4 Optimal Solution for Θ

Without specific proof of concavity (or I may be able to do it after this section–when 1st order derivative is obtained and if it seems easy to derive the 2nd), the optimal solution to all components of Θ' is obtained by letting the 1st order derivative be 0.

6.4.1 Probability for Initial Hidden State: π

For **distribution of initial hidden states**, since it conforms to the condition that all sum up to one:

$$\sum_{j=1}^{J} \pi_j = 1$$

Then using Lagrange multiplier we'll have:

$$\frac{\partial\left[Q(\Theta, \Theta') + \lambda_\pi(1 - \sum_{j=1}^{J} \pi_j)\right]}{\partial \pi_j} = \frac{\gamma_1(j)}{\pi_j} - \lambda = 0$$

$$\rightarrow \forall j \in [J]: \quad \hat{\pi_j} = \frac{\gamma_1(j)}{\lambda_\pi}$$

Since the optimal solutions also follows the *sum-to-1* rule, summing them up to obtain a math expression that offsets λ:

$$\sum_{j=1}^{J} \hat{\pi_j} = \sum_{j=1}^{J} \frac{\gamma_1(j)}{\lambda_\pi} = \frac{1}{\lambda_\pi} \sum_{j=1}^{J} \gamma_1(j) = 1 \quad \rightarrow \quad \lambda_\pi = \sum_{j=1}^{J} \gamma_1(j)$$

Thus the optimal solution for initial hidden state probability is:

$$\hat{\pi}_j = \frac{\gamma_1(j)}{\sum_{j=1}^{J} \gamma_1(j)}$$

Vector/Matrix Form

$$\vec{\hat{\pi}}_j = \frac{\vec{\gamma}_1(j)}{\vec{\gamma}_1(j).\text{sum}()}$$

So compute the vector first, then grab the sum, and finally normalize the vector with the sum.

6.4.2 Transition Probability: A

For **transition probability**, we also have the constraint that for one source state i, probabilities to all potential destination state j shall sum up to 1:

$$\forall i \in [J]: \quad \sum_{j=1}^{J} a_{i,j} = 1$$

Or in a more concise form, $\forall i \in [J]: \sum a_{i,:} = 1$.

Then similarly for each element:

$$\frac{\partial \left[Q(\Theta, \Theta') + \lambda_A (1 - \sum_{j=1}^{J} a_{i,j}) \right]}{\partial a_{i,j}} = \sum_{t=2}^{T} \frac{\xi_{t-1}(i,j)}{a_{i,j}} - \lambda_A = 0$$

$$\rightarrow \forall i \in [J], j \in [J]: \quad \hat{a}_{i,j} = \frac{\sum_{t=2}^{T} \xi_{t-1}(i,j)}{\lambda_A}$$

Similarly we have:

$$\sum_{j'=1}^{J} \hat{a}_{i,j} = \sum_{j'=1}^{J} \frac{\sum_{t=2}^{T} \xi_{t-1}(i,j')}{\lambda_A} = 1 \quad \rightarrow \quad \lambda_A = \sum_{j'=1}^{J} \sum_{t=2}^{T} \xi_{t-1}(i,j')$$

Finally it is derived that:

$$\hat{a}_{i,j} = \frac{\sum_{t=2}^{T} \xi_{t-1}(i,j)}{\sum_{j'=1}^{J} \sum_{t=2}^{T} \xi_{t-1}(i,j')}$$

Vector/Matrix Form

$$\hat{A} = \frac{\sum_{t=2}^{T} \Xi_t}{\left(\sum_{t=2}^{T} \Xi_t \right) .\text{sum(axis=1)}}$$

And the division (matrix over vector) is broadcast row-wise.

6.4.3 Emission Probability: B

For **emission probability**, there's no specific constraint but we have to work extra to obtain optimal solutions to parameters inside B.

In the specific case of using single diagonal Gaussian as the emission probability, which is:

$$B_t(j) = \mathcal{N}(\mathbf{o}_t \mid \mu_j, \mathbf{r}_j) = \frac{1}{\sqrt{2\pi \mathbf{r}_j}} \exp\left(-\frac{(\mathbf{o}_t - \mu_j)^2}{2\mathbf{r}_j} \right)$$

Let's get the derivatives for both means and variances:

$$\begin{aligned}
\frac{\partial B_t(j)}{\partial \mu_j} &= \frac{1}{\sqrt{2\pi \mathbf{r}_j}} \exp\left(-\frac{(\mathbf{o}_t - \mu_j)^2}{2\mathbf{r}_j} \right) \times (-\frac{2(\mathbf{o}_t - \mu_j)}{2\mathbf{r}_j}) = -\frac{\mathbf{o}_t - \mu_j}{\mathbf{r}_j} \cdot B_t(j) \\
\frac{\partial B_t(j)}{\partial \mathbf{r}_j} &= \frac{1}{\sqrt{2\pi}} \cdot (-\frac{1}{2\mathbf{r}_j \sqrt{\mathbf{r}_j}}) \cdot \exp\left(-\frac{(\mathbf{o}_t - \mu_j)^2}{2\mathbf{r}_j} \right) + \frac{1}{\sqrt{2\pi \mathbf{r}_j}} \exp\left(-\frac{(\mathbf{o}_t - \mu_j)^2}{2\mathbf{r}_j} \right) \cdot (-\frac{(\mathbf{o}_t - \mu_j)^2}{2}) \cdot (-\frac{1}{\mathbf{r}_j^2}) \\
&= -\frac{1}{2\mathbf{r}_j} \cdot B_t(j) + B_t(j) \cdot \frac{(\mathbf{o}_t - \mu_j)^2}{2\mathbf{r}_j^2} \\
&= \frac{1}{2\mathbf{r}_j} \cdot B_t(j) \left(\frac{(\mathbf{o}_t - \mu_j)^2}{\mathbf{r}_j} - 1 \right)
\end{aligned}$$

And hence get the derivative for $B_t(j)$ as an entity:

$$\frac{\partial Q(\Theta, \Theta')}{\partial B_t(j)} = \frac{\gamma_t(j)}{B_t(j)}$$

Since means and variances influence B_t at all time steps, the derivatives shall be **summed up for all potential** $t \in [T]$. Therefore, respectively for means and variances, using the chain rule in derivative computation:

$$\begin{aligned}
\frac{\partial Q(\Theta, \Theta')}{\partial \mu_j} &= \sum_{t=1}^{T} \frac{\partial Q(\Theta, \Theta')}{\partial B_t(j)} \cdot \frac{\partial B_t(j)}{\partial \mu_j} \\
&= \sum_{t=1}^{T} \frac{\gamma_t(j)}{B_t(j)} \cdot \left(-\frac{\mathbf{o}_t - \mu_j}{\mathbf{r}_j} \cdot B_t(j) \right) \\
&= -\frac{\sum_{t=1}^{T} \gamma_t(j) \cdot (\mathbf{o}_t - \mu_j)}{\mathbf{r}_j} = 0 \\
\rightarrow \hat{\mu}_j &= \frac{\sum_{t=1}^{T} \gamma_t(j) \cdot \mathbf{o}_t}{\sum_{t=1}^{T} \gamma_t(j)} \\
\frac{\partial Q(\Theta, \Theta')}{\partial \mathbf{r}_j} &= \sum_{t=1}^{T} \frac{\partial Q(\Theta, \Theta')}{\partial B_t(j)} \cdot \frac{\partial B_t(j)}{\partial \mathbf{r}_j} \\
&= \sum_{t=1}^{T} \frac{\gamma_t(j)}{B_t(j)} \cdot \frac{1}{2\mathbf{r}_j} \cdot B_t(j) \left(\frac{(\mathbf{o}_t - \mu_j)^2}{\mathbf{r}_j} - 1 \right) \\
&= \frac{1}{2\mathbf{r}_j} \sum_{t=1}^{T} \gamma_t(j) \cdot \left(\frac{(\mathbf{o}_t - \mu_j)^2}{\mathbf{r}_j} - 1 \right) = 0 \\
\rightarrow \hat{\mathbf{r}}_j &= \frac{\sum_{t=1}^{T} \gamma_t(j) \cdot (\mathbf{o}_t - \mu_j)^2}{\sum_{t=1}^{T} \gamma_t(j)}
\end{aligned}$$

Vector/Matrix Form

$$\vec{\hat{\mu}} = \frac{\sum_{t=2}^{T} \mathbf{o}_t \cdot \vec{\gamma}_t}{\sum_{t=2}^{T} \vec{\gamma}_t} \qquad \vec{\hat{\mathbf{r}}} = \frac{\sum_{t=2}^{T} \left[(\mathbf{o}_t - \vec{\hat{\mu}}) \odot \vec{\gamma}_t \right]}{\sum_{t=2}^{T} \vec{\gamma}_t}$$

6.5 Baum-Welch Algorithm

Algorithm 1 Baum-Welch Algorithm for HMM-DiagGaussian: Iterative Parameter Optimzation

Require: Observations: $O = (\mathbf{o}_t \in \mathbf{R}^D \mid t = 1, ..., T)$
Require: (initial) Parameters: $\Theta = \{\pi_j, \{a_{i,j}\}_{i=1}^{J}, \mu_j, \mathbf{r}_j\}_{j=1}^{J}$
  while Stopping criteria not satisfied **do**
    if N is even **then**
      Forward algorithm to update $\forall t \in [T], j \in [J] : \alpha_t(j)$ and joint output sequence likelihood $p(O \mid \Theta)$
      Backward algorithm to update $\forall t \in [T], j \in [J] : \beta_t(j)$
      Forward-Backward algorithm to update $\forall t \in [T], i \in [J], j \in [J] : \gamma_t(j), \xi_{t-1}(i, j)$
      Update HMM and DiagGaussian parameters $\Theta \leftarrow \hat{\Theta}$ following update rules
    end if
  end while

7 HMM-GMM: Introducing Gaussian Mixture Model

7.1 GMM & Observation Sequence

Given an observation sequence O with T time steps, if a Gaussian Mixture Model (GMM) with K components is applied to compute the joint likelihood, this problem can be formulated as follows:

7.1.1 Estimation-At-A-Single-Time-Step

$$p(\mathbf{o}_t \mid \Theta) = \sum_{k=1}^{K} p(\mathbf{o}_t, v_t = k \mid \Theta) \cdot p(v_t = k \mid \Theta) = \sum_{k=1}^{K} \omega_k \cdot \mathcal{N}(\mathbf{o}_t \mid \mu_k, \mathbf{r}_k)$$

In this scenario, we can consider $\forall k \in [K]: \ \omega_k, \mu_k, \mathbf{r}_k$ are new parameters added to the set Θ, and the observation is purely a weighted sum of probabilities from a series of Gaussian distributions. Note that the probability of variable v_t can be interpreted as "how much does the single Gaussian, identified as k, contribute to the observation at time step t".

7.1.2 Estimation-For-Full-Sequence

As for the entire observation, just utilize marginal probability and sequence decomposition:

$$\begin{aligned} p(O \mid \Theta) &= \sum_{V} p(O, V \mid \Theta) = \sum_{V} p(O_{1:T}, V \mid \Theta) \\ &= \prod_{t=1}^{T} \left(\sum_{k=1}^{K} \omega_k \mathcal{N}(\mathbf{o}_t \mid \mu_k, \mathbf{r}_k) \right) \end{aligned}$$

Log-Likelihood

$$\log p(O \mid \Theta) = \sum_{t=1}^{T} \log \left(\sum_{k=1}^{K} \omega_k \mathcal{N}(\mathbf{o}_t \mid \mu_k, \mathbf{r}_k) \right)$$

Jensen's Inequality

$$\log(\sum_{i} a_i b_i) \geq \sum_{i} a_i \log b_i$$

Lower Bound of Log-Likelihood

$$\log p(O \mid \Theta) \geq \sum_{t=1}^{T} \sum_{k=1}^{K} \omega_k \log \mathcal{N}(\mathbf{o}_t \mid \mu_k, \mathbf{r}_k)$$

Using Individual Log-Likelihood and Log-Weight

$$\begin{aligned} \log p(O \mid \Theta) &= \sum_{t=1}^{T} \log \left(\sum_{k=1}^{K} \exp(\log \omega_k + \log \mathcal{N}(\mathbf{o}_t \mid \mu_k, \mathbf{r}_k)) \right) \\ &= \sum_{t=1}^{T} \operatorname*{logsumexp}_{k} \left[\log \omega_k + \log \mathcal{N}(\mathbf{o}_t \mid \mu_k, \mathbf{r}_k) \right] \end{aligned}$$

7.1.3 Optimization with Expectation Maximization

Similarly start with getting the 1st order derivative:

$$\frac{\partial \log p(O \mid \Theta)}{\partial \mathcal{N}(\mathbf{o}_t \mid \mu_k, \mathbf{r}_k)} = \sum_{t=1}^{T} \frac{\omega_k}{\sum_{k'=1}^{K} \omega_{k'} \mathcal{N}(\mathbf{o}_t \mid \mu_{k'}, \mathbf{r}_{k'})}$$

To simplify notations, define $\gamma_t(k)$, **contribution weight** of the k-th simple Gaussian in the GMM, as follows:

$$\gamma_t(k) = \frac{\omega_k \mathcal{N}(\mathbf{o}_t \mid \mu_k, \mathbf{r}_k)}{\sum_{k'=1}^{K} \omega_{k'} \mathcal{N}(\mathbf{o}_t \mid \mu_{k'}, \mathbf{r}_{k'})}$$

And it has the following property:

$$\sum_{k=1}^{K} \gamma_t(k) = \frac{\sum_{k=1}^{K} \omega_k \mathcal{N}(\mathbf{o}_t \mid \mu_k, \mathbf{r}_k)}{\sum_{k'=1}^{K} \omega_{k'} \mathcal{N}(\mathbf{o}_t \mid \mu_{k'}, \mathbf{r}_{k'})} = 1$$

***Note**: Compared to the derivative above, $\mathcal{N}(\mathbf{o}_t \mid \mu_k, \mathbf{r}_k)$ in the denominator is added, since the derivatives of both μ_k and $\mathbf{r}_k$ have an extra $\mathcal{N}(\mathbf{o}_t \mid \mu_k, \mathbf{r}_k)$.*

Log Computation

$$\begin{aligned}
\log \gamma_t(k) &= \log \left[\omega_k \mathcal{N}(\mathbf{o}_t \mid \mu_k, \mathbf{r}_k)\right] - \log \left(\sum_{k'=1}^{K} \omega_{k'} \mathcal{N}(\mathbf{o}_t \mid \mu_{k'}, \mathbf{r}_{k'}) \right) \\
&= \log \omega_k + \log \mathcal{N}(\mathbf{o}_t \mid \mu_k, \mathbf{r}_k) - \operatorname*{logsumexp}_{k'} \left[\log \omega_{k'} + \log \mathcal{N}(\mathbf{o}_t \mid \mu_{k'}, \mathbf{r}_{k'})\right]
\end{aligned}$$

Computation for the derivative of ω_k is under constraint, and we apply the Lagrange multiplier again:

$$\begin{aligned}
\frac{\partial \left[\log p(O \mid \Theta) + \lambda_\omega (1 - \sum_{k=1}^{K} \omega_k)\right]}{\partial \omega_k} &= \sum_{t=1}^{T} \frac{\mathcal{N}(\mathbf{o}_t \mid \mu_k, \mathbf{r}_k)}{\sum_{k'=1}^{K} \omega_{k'} \mathcal{N}(\mathbf{o}_t \mid \mu_{k'}, \mathbf{r}_{k'})} - \lambda_\omega = \sum_{t=1}^{T} \frac{\gamma_t(k)}{\omega_k} - \lambda_\omega = 0 \\
&\rightarrow \begin{cases} \hat{\omega}_k = \frac{\sum_{t=1}^{T} \gamma_t(k)}{\lambda_\omega} \\ \sum_{k'=1}^{K} \hat{\omega}_{k'} = \sum_{k'=1}^{K} \frac{\sum_{t=1}^{T} \gamma_t(k')}{\lambda_\omega} = 1 \end{cases} \\
&\rightarrow \lambda_\omega = \sum_{t=1}^{T} \sum_{k'=1}^{K} \gamma_t(k') \\
\hat{\omega}_k &= \frac{\sum_{t=1}^{T} \gamma_t(k)}{\sum_{t=1}^{T} \sum_{k'=1}^{K} \gamma_t(k')}
\end{aligned}$$

Vector/Matrix Log Computation

$$\begin{aligned}
\log \vec{\hat{\omega}} &= \log \left(\sum_{t=1}^{T} \vec{\gamma}_t \right) - \log \left(\sum_{t=1}^{T} \sum_{k'=1}^{K} \gamma_t(k') \right) \\
&= \operatorname*{logsumexp}_{t} \left[\log \vec{\gamma}_t\right] - \operatorname*{logsumexp}_{t,k'} \left[\log \gamma_t(k')\right]
\end{aligned}$$

Coming further to add μ_k and $\mathbf{r}_k$:

$$\frac{\partial \log p(O \mid \Theta)}{\partial \mu_k} = \sum_{t=1}^{T} \frac{\gamma_t(k)}{\mathcal{N}(\mathbf{o}_t \mid \mu_k, \mathbf{r}_k)} \cdot \frac{\partial \mathcal{N}(\mathbf{o}_t \mid \mu_k, \mathbf{r}_k)}{\partial \mu_k} = -\frac{1}{\mathbf{r}_k} \sum_{t=1}^{T} \gamma_t(k) \cdot (\mathbf{o}_t - \mu_k) = 0$$

$$\rightarrow \hat{\mu}_k = \frac{\sum_{t=1}^{T} \gamma_t(k) \cdot \mathbf{o}_t}{\sum_{t=1}^{T} \gamma_t(k)}$$

$$\frac{\partial \log p(O \mid \Theta)}{\partial \mathbf{r}_k} = \sum_{t=1}^{T} \frac{\gamma_t(k)}{\mathcal{N}(\mathbf{o}_t \mid \mu_k, \mathbf{r}_k)} \cdot \frac{\partial \mathcal{N}(\mathbf{o}_t \mid \mu_k, \mathbf{r}_k)}{\partial \mathbf{r}_k} = \frac{1}{2\mathbf{r}_k} \sum_{t=1}^{T} \gamma_t(k) \left(\frac{(\mathbf{o}_t - \mu_k)^2}{\mathbf{r}_k} - 1 \right) = 0$$

$$\rightarrow \hat{\mathbf{r}}_k = \frac{\sum_{t=1}^{T} \gamma_t(k) \cdot (\mathbf{o}_t - \hat{\mu}_k)^2}{\sum_{t=1}^{T} \gamma_t(k)}$$

Vector/Matrix Log Computation

$$\begin{aligned}
\log \vec{\hat{\mu}} &= \log \left[\sum_{t=1}^{T} \vec{\gamma}_t \cdot \mathbf{o}_t \right] - \log \left[\sum_{t=1}^{T} \vec{\gamma}_t \right] \\
&= \underset{t}{\text{logsumexp}} \left[\log \vec{\gamma}_t + \log \mathbf{o}_t \right] - \underset{t}{\text{logsumexp}} \left[\log \vec{\gamma}_t \right] \\
\log \vec{\hat{\mathbf{r}}} &= \log \left[\sum_{t=1}^{T} \vec{\gamma}_t \cdot (\mathbf{o}_t - \hat{\mu}_k)^2 \right] - \log \left[\sum_{t=1}^{T} \vec{\gamma}_t \right] \\
&= \underset{t}{\text{logsumexp}} \left[\log \vec{\gamma}_t + 2 \log(\mathbf{o}_t - \hat{\mu}_k) \right] - \underset{t}{\text{logsumexp}} \left[\log \vec{\gamma}_t \right]
\end{aligned}$$

7.2 With Dependence on Hidden States: HMM-GMM

Intuition: using multiple Gaussian distributions to map the emission from s_t to $\mathbf{o}_t$:

$$p(\mathbf{o}_t \mid s_t = j, \Theta) = \sum_{k=1}^{K} \omega_{jk} \cdot \mathcal{N}(\mathbf{o}_t \mid \mu_{jk}, \mathbf{r}_{jk})$$

$$\text{s.t.} \forall j \in [J], k \in [K]: \quad 0 \leq \omega_{jk} \leq 1 \text{ and } \sum_{k=1}^{K} \omega_{jk} = 1$$

The rest of forward / backward computation, as well as Viterbi and Baum-Welch algorithms are basically the same following the same idea in previous simple HMM-DiagGaussian case. For more detailed expansion, refer to slides of *Lecture 11: Hidden Markove Model, Part III* for more details.