Department of Mathematics
Sun Yat-Sen University

Hidden Markov Model

Rockman Zheng

February 25, 2017

Contents

1 Overview

In this paper, we will discuss discrete time hidden Markov model (HMM). We will restrict our discusion on HMM which only has finite number of states and is stationary. We will first introduce discrete time Markov model (also restricted to has finite number of states, and is stationary) to give a basic structure of HMM, and some notations also needed when discussing HMM. In the following sections, we will formally introduce discrete time HMM as an extension of Markov model. And conversely, we will point out that discrete time Markov model can be viewed as a degenerated version of HMM. After setting up the basic model, we will cover 4 operations about HMM that frequently appear in applications, such as speech recognition. They are: filtering, smoothing, decoding, and learning. Both objectives and algorithms of these operation will be discussed in this paper.

1.1 Notations

There is some conventions of the notation in this paper.

Vector and matrix use bold font like: $\mathbf{A}, \mathbf{X}$. On the other hand, entries of vector use regular font like X_k.

We sometimes use short hand notations like $\mathbf{X}_0^k$ to represent $\mathbf{X}(0:k)$ and $(X_0, \dots, X_k)$.

In probability distribution function, we sometimes leave out the values of random variables if doing so has no ambiguity. For example, instead of writing:

$$p(\mathbf{X} = \mathbf{x}|Y_k = y_k), \quad p(X_k = i, X_{k+1} = j),$$

etc, we will use the short hand:

$$p(\mathbf{X}|Y_k), \quad p(X_k, X_{k+1}).$$

2 Markov Model

2.1 Mathematical Setup

A discrete time, stationary Markov model, is characterized by the following elements:

State Space

We use single number to represent a particular state. All possible states together form a set $S = \{0, 1, \dots, N, N+1\}$ which is called the state space. Note that there are only finite number of states. In particular, 0 is called **start state**, and $N+1$ is called **final state** or **end state**. We will say the size of S is N in this case.

Markov Process

A sequence of random variables $\mathbf{X} = (X_0, X_1, \dots)$, where $X_0 = 0$ always holds by letting

$$p(X_0 = 0) = 1.$$

$X_k = i, (i \in S)$ means that the process goes into state i at time k. Markov process has **Markov property**, that the probability of moving to the next state depends only on the present state and not on any previous states:

$$p(X_n = x|\mathbf{X}_{0:n-1} = \mathbf{x}_{0:n-1}) = p(X_n = x|X_{n-1} = x_{n-1}), \quad n > 0$$

where $x, x_{n-1} \in S$.

Remark. But the states the process will be in the future, that is, the value of $\mathbf{X}_{n+1:T+1}$, may affect the likelihood of observation $X_n = x$.

Markov process also has **stationary property**, that the probability for the process going into state j from state i is independent when it takes that transition:

$$p(X_k = j | X_{k-1} = i) = p(X_2 = j | X_1 = i),$$

for all $i, j \in S, k \geq 1$.

Transition Probabilities

Also referred to as **Transition matrix A**, which has dimension $(N+2) \times (N+2)$. The entry $\mathbf{A}(i, j)$ is the probability that state i transitions to state j:

$$\mathbf{A}(i, j) = p(X_2 = j | X_1 = i), \quad i, j \in S.$$

Transition matrix $\mathbf{A}$ has the following restrictions: $\mathbf{A}(:, 0) = 0$ so that it's impossible for any state to transition to start state; $\mathbf{A}(N+1, N+1) = 1$ so that end state always transitions back to end state.

To sum up, our Markov model can be represented as a tuple: $(S, \mathbf{X}, \mathbf{A})$.

2.2 Graphical Setup

2.3 Formula

Regularity I. The process must transition next time:

$$\sum_{k=0}^{N+1} \mathbf{A}(:, k) = [1, \ldots, 1]. \tag{2.1}$$

Proof. Take i th row for instance.

$$\begin{aligned}
\sum_{k=0}^{N+1} \mathbf{A}(i, k) &= \sum_{k=0}^{N+1} p(X_2 = k | X_1 = i) \\
&= \frac{\sum_{k=0}^{N+1} p(X_2 = k, X_1 = i)}{p(X_1 = i)} && \text{(conditional probability)} \\
&= \frac{p(X_1 = i)}{p(X_1 = i)} = 1 && \text{(marginal probability)}
\end{aligned}$$

Chain rule I. Let $\mathbf{X} = (X_0, \ldots, X_T), \mathbf{x} = (x_0, \ldots, x_T), x_0 = 0, x_k \in S, \forall 0 < k \leq T$

$$p(\mathbf{X} = \mathbf{x}) = \prod_{k=0}^{T} \mathbf{A}(x_k, x_{k+1}) \tag{2.2}$$

Proof.

$$
\begin{aligned}
& p(\mathbf{X} = \mathbf{x}) \\
&= p(X_{T+1} = x_{T+1} | \mathbf{X}_0^T = \mathbf{x}_0^T) p(\mathbf{X}_0^T = \mathbf{x}_0^T) && \text{(conditional probability)} \\
&= p(X_{T+1} = x_{T+1} | X_T = x_T) p(\mathbf{X}_0^T = \mathbf{x}_0^T) && \text{(Markov property)} \\
&= \cdots \\
&= \prod_{k=0}^{T} p(X_{k+1} = x_{k+1} | X_k = x_k) \\
&= \prod_{k=0}^{T} \mathbf{A}(x_k, x_{k+1}) && \text{(stationary property)}
\end{aligned}
$$

3 Hidden Markov Model

Based on Markov model introduced in the previous section, we now introduce hidden Markov model by adding more setup. Note that all Markov parts remain unchanged in HMM.

Each state of HMM would emit or generate a phenomenon randomly picked from the observation space. We are often only able to observe the sequence of phenomena themselves rather than directly measuring the underlying sequence of states that emit them. Our task is to guess the hidden sequence of states with our observations as evidences, so that leads to the name 'hidden Markov model'.

3.1 Mathematical Setup

A discrete time, stationary HMM, is characterized by the following elements:

State Space

We use single number to represent a particular state. All possible states together form a set $S = \{0, 1, \ldots, N, N+1\}$ which is called the state space. Note that there are only finite number of states. In particular, 0 is called **start state**, and $N+1$ is called **final state** or **end state**. We will say the size of S is N in this case.

Observation Space

We use single number to represent a particular observation. All possible observations together form a set $E = \{0, 1, \ldots, M\}$ which is called the observation space. Here observation 0 means no observation was made.

Observation Process

A finite sequence of random variables $\mathbf{Y} = (Y_0, Y_1, \ldots, Y_T, Y_{T+1})$ represents the process of observations. Here T indicates that we have made T observations.

$Y_k = i, (i \in E)$ means the observation at time k is i. At time 0 and $T+1$ we do not make any observation, thus $Y_0 = 0, Y_{T+1} = 0$ always hold by letting

$$p(Y_0 = 0) = 1, p(Y_{T+1} = 0) = 1$$

.

Markov Process

A sequence of random variables $\mathbf{X} = (X_0, X_1, \ldots, X_T, X_{T+1})$. Here T indicates that we have made T observations. $X_k = i$ means that the process goes into state i at time k. At time 0 the process is in start state, and in end state at time $T+1$, thus $X_0 = 0, X_{T+1} = N+1$ always holds by letting

$$p(X_0 = 0) = 1, p(X_{T+1} = N+1) = 1.$$

Markov process in HMM has an extended version of **Markov property**, that the probability for the process going into state x at time n only depends on the state it's in at time $n-1$ but not on which states it resided at any previous time, or the history observation sequence:

$$p(X_n = x|\mathbf{X}_0^{n-1} = \mathbf{x}_0^{n-1}, \mathbf{Y}_0^{n-1} = \mathbf{y}_0^{n-1}) = p(X_n = x|X_{n-1} = x_{n-1}), \tag{3.1}$$

where $x, x_{n-1} \in S$.

Remark. But both present and future information, that is, the value of $\mathbf{Y}_n$ and $\mathbf{X}_{n+1}^{T+1}, \mathbf{Y}_{n+1}^{T+1}$, may affect the likelihood of observation $X_n = x$.

And **stationary property**, that the probability for the process going into state j from state i is independent when it takes that transition:

$$p(X_k = j|X_{k-1} = i) = p(X_2 = j|X_1 = i), \tag{3.2}$$

for all $i, j \in S, k \geq 1$.

Observation sequence in HMM has **Output independence assumption**, that the likelihood of observation at time k depends only on the state the hidden Markov process was in at the same time:

$$p(Y_k = y|\mathbf{X}_{0:k} = \mathbf{x}_{0:k}, \mathbf{Y}_{0:k-1} = \mathbf{y}_{0:k-1}) = p(Y_k = y|X_k = x_k), \tag{3.3}$$

where $0 \leq k \leq T+1, x_k \in S, y \in E$.

Remark. But future information, that is, the value of $\mathbf{X}_{k+1:T+1}, \mathbf{Y}_{k+1:T+1}$, may affect the likelihood of observation $Y_k = y$.

Observation sequence also shares some kind of **stationary property**, that the probability for a state i emits an observation j is independent when this observation is made:

$$p(Y_s = j|X_s = i) = p(Y_1 = j|X_1 = i), \tag{3.4}$$

for all $i \in S, j \in E, 0 \leq s \leq T+1$.

Transition Probabilities

Also referred to as **Transition matrix A**, which has dimension $(N+2) \times (N+2)$. $\mathbf{A}(i, j)$ stores the probability that state $i \in S$ transitions to state $j \in S$:

$$\mathbf{A}(i, j) = p(X_k = j|X_{k-1} = i).$$

Transition matrix $\mathbf{A}$ has the following restrictions: $\mathbf{A}(:, 0) = 0$ so that it's impossible for any state to transition to start state; $\mathbf{A}(N+1, N+1) = 1$ so that end state always transitions back to end state.

Emission Probabilities

Also represented by **Emission matrix B**, which has dimension $(N+2) \times (M+1)$. $\mathbf{B}(i, j)$ stores the probability that state $i \in S$ emits observation $j \in E$:

$$\mathbf{B}(i, j) = p(Y_1 = j|X_1 = i).$$

There are some restrictions about emission matrix B that start state and end state are non-emitting. That is to say:

$$\mathbf{B}(0,0) = 1, \quad \mathbf{B}(N+1,0) = 1.$$

To sum up, our HMM can be represented as a tuple: $(S, E, \mathbf{X}, \mathbf{Y}, \mathbf{A}, \mathbf{B})$.
We now list some notations and assumptions before going into further discussion.

- $\mathbf{X} = (X_0, \ldots, X_{T+1})$ is the Markov process;
- $\mathbf{Y} = (Y_0, \ldots, Y_{T+1})$ is the observation process;
- $\mathbf{x} = (x_0, \ldots, x_{T+1})$ is the sequence of states, where $x_0 = 0, x_{T+1} = N+1, 0 < x_k \leq N, \forall 0 < k \leq T$;
- $\mathbf{y} = (y_0, \ldots, y_{T+1})$ is the sequence of observations, where $y_0 = 0, y_{T+1} = 0, 0 < y_k \leq M, \forall 0 < k \leq T$.

3.2 Graphical Setup

3.3 Formula

Regularity about transition matrix A and chain rule I are the same as discussed in Markov model section. Here are some properties and formula only related to HMM.

Regularity II. We now point out the regularity about the emission matrix B:

$$\sum_{k=0}^{M} \mathbf{B}(:,k) = [1, \ldots, 1]. \tag{3.5}$$

In words, every state must output some observation.

Proof. Take i-th row for example.

$$\begin{aligned}
\sum_{k=0}^{M} \mathbf{B}(i,k) &= \sum_{k=0}^{M} p(Y_1 = k | X_1 = i) \\
&= \frac{\sum_{k=0}^{M} p(Y_1 = k, X_1 = i)}{p(X_1 = i)} && \text{(conditional probability)} \\
&= \frac{p(X_1 = i)}{p(X_1 = i)} = 1 && \text{(marginal probability)}
\end{aligned}$$

Chain rule II

$$p(\mathbf{Y} = \mathbf{y} | \mathbf{X} = \mathbf{x}) = \prod_{k=1}^{T} \mathbf{B}(x_k, y_k)$$

Proof.

$$
\begin{aligned}
& p(\mathbf{Y}=\mathbf{y}|\mathbf{X}=\mathbf{x}) \\
&= \frac{p(\mathbf{Y}=\mathbf{y},\mathbf{X}=\mathbf{x})}{p(\mathbf{X}=\mathbf{x})} && \text{(conditional probability)} \\
&= \frac{p(Y_{T+1}|\mathbf{Y}_0^T,\mathbf{X})p(\mathbf{Y}_0^T,\mathbf{X})}{p(\mathbf{X})} && \text{(conditional probability)} \\
&= \frac{p(Y_{T+1}|X_{T+1})p(\mathbf{Y}_0^T,\mathbf{X})}{p(\mathbf{X})} && \text{(output independent assumption)} \\
&= \frac{\mathbf{B}(x_{T+1},y_{T+1})p(X_{T+1}|\mathbf{Y}_0^T,\mathbf{X}_0^T)p(\mathbf{Y}_0^T,\mathbf{X}_0^T)}{p(\mathbf{X})} && \text{(conditional probability)} \\
&= \frac{\mathbf{B}(x_{T+1},y_{T+1})p(X_{T+1}|X_T)p(\mathbf{Y}_0^T,\mathbf{X}_0^T)}{p(\mathbf{X})} && \text{(Markov property)} \\
&= \frac{\mathbf{B}(x_{T+1},y_{T+1})\mathbf{A}(x_T,x_{T+1})p(\mathbf{Y}_0^T,\mathbf{X}_0^T)}{p(\mathbf{X})} \\
&= \cdots \\
&= \frac{\prod_{k=0}^{T+1}\mathbf{B}(x_k,y_k)\prod_{k=0}^{T}\mathbf{A}(x_k,x_{k+1})}{p(\mathbf{X})} \\
&= \prod_{k=0}^{T+1}\mathbf{B}(x_k,y_k), && \text{(Chain rule I)}
\end{aligned}
$$

remember we have made the assumptions that $x_0 = 0, y_0 = 0, x_{T+1} = N+1, y_{T+1} = 0$, and recall that $\mathbf{B}(0,0) = 1, \mathbf{B}(N+1,0) = 1$. Substituting these facts into the equation above completes the proof.

Joint Probability: The joint probability of observations and its corresponding states:

$$p(\mathbf{Y}=\mathbf{y},\mathbf{X}=\mathbf{x}) = \prod_{k=1}^{T}\mathbf{B}(x_k,y_k)\prod_{k=1}^{T+1}\mathbf{A}(x_{k-1},x_k), \tag{3.6}$$

where $x_0 = 0, x_{T+1} = N+1$.

Proof. We only need to note that

$$p(\mathbf{Y},\mathbf{X}) = p(\mathbf{Y}|\mathbf{X})p(\mathbf{X}),$$

and then apply the 2 chain rules shown previously.

Observation Likelihood: The likelihood of a particular sequence of observation Y given the HMM:

$$p(\mathbf{Y}) = \sum_{\mathbf{x}}\prod_{k=1}^{T}\mathbf{B}(x_k,y_k)\prod_{k=1}^{T+1}\mathbf{A}(x_{k-1},x_k), \tag{3.7}$$

where $x_0 = 0, x_{T+1} = N+1$.

Proof. We only need to note that

$$p(\mathbf{Y}) = \sum_{x} p(\mathbf{Y},\mathbf{X}),$$

and then apply the joint probability formula given above.

Short Memory: The likelihood of observations only depends on the latest history:

$$p\left(\mathbf{Y}_k^{T+1}\left|\mathbf{X}_0^{k-t},\mathbf{Y}_0^{k-t}\right.\right)=p\left(\mathbf{Y}_k^{T+1}\right|\left.X_{k-t}\right). \tag{3.8}$$

where $t>0$.

Proof. We would expand both side to show the equality. For the left hand side:

$$p\left(\mathbf{Y}_k^{T+1}\left|\mathbf{X}_0^{k-t},\mathbf{Y}_0^{k-t}\right.\right)=\sum_{\mathbf{j}_{k-t+1}^{T+1}} p\left(\mathbf{Y}_k^{T+1},\mathbf{X}_{k-t+1}^{T+1}=\mathbf{j}_{k-t+1}^{T+1}\left|\mathbf{X}_0^{k-t},\mathbf{Y}_0^{k-t}\right.\right)$$

Now let's focus on the term under summation.

$$\begin{aligned}
&p\left(\mathbf{X}_{k-t+1}^{T+1},\mathbf{Y}_k^{T+1}\left|\mathbf{X}_0^{k-t},\mathbf{Y}_0^{k-t}\right.\right)\\
&=p\left(Y_{T+1}\left|\mathbf{X}_0^{T+1},\mathbf{Y}_0^{k-t},\mathbf{Y}_k^{T}\right.\right)p\left(\mathbf{X}_{k-t+1}^{T+1},\mathbf{Y}_k^{T}\left|\mathbf{X}_0^{k-t},\mathbf{Y}_0^{k-t}\right.\right)\\
&=p(Y_{T+1}|X_{T+1})p\left(X_{T+1}\left|\mathbf{X}_0^{T},\mathbf{Y}_0^{k-t},\mathbf{Y}_k^{T}\right.\right)p\left(\mathbf{X}_{k-t+1}^{T},\mathbf{Y}_k^{T}\left|\mathbf{X}_0^{k-t},\mathbf{Y}_0^{k-t}\right.\right)\\
&=\cdots\\
&=p\left(\mathbf{X}_{k-t+1}^{k}\left|\mathbf{X}_0^{k-t},\mathbf{Y}_0^{k-t}\right.\right)\prod_{t=k}^{T}p(X_{t+1}|X_t)\prod_{t=k}^{T+1}p(Y_t|X_t).
\end{aligned}$$

Now turn to the right hand side, we can similarly get:

$$p\left(\mathbf{Y}_k^{T+1}\right|\left.X_{k-t}\right)=\sum_{\mathbf{j}_{k-t+1}^{T+1}} p\left(\mathbf{X}_{k-t+1}^{k}\left|X_{k-t}\right.\right)\prod_{t=k}^{T}p(X_{t+1}|X_t)\prod_{t=k}^{T+1}p(Y_t|X_t).$$

So we only need to prove the identity:

$$p\left(\mathbf{X}_{k-t+1}^{k}\left|\mathbf{X}_0^{k-t},\mathbf{Y}_0^{k-t}\right.\right)=p\left(\mathbf{X}_{k-t+1}^{k}\left|X_{k-t}\right.\right).$$

But it is easy to use Markov property to show that both of them are just:

$$\prod_{s=1}^{t}p(X_{k-s+1}|X_{k-s}),$$

and that completes the proof. We also obtain the equation below which will be useful later when we discuss backward algorithm.

$$p\left(\mathbf{Y}_k^{T+1}\right|\left.X_{k-t}\right)=\prod_{s=1}^{t}p(X_{k-s+1}|X_{k-s})\prod_{t=k}^{T}p(X_{t+1}|X_t)\prod_{t=k}^{T+1}p(Y_t|X_t). \tag{3.9}$$

4 Operations and their Algorithms

There are 3 fundamental tasks about HMM, which are:

1. Given a HMM and an observation sequence $\mathbf{y}$, compute the observation likelihood $p(\mathbf{Y}=\mathbf{y})$.

2. Given a HMM and an observation sequence $\mathbf{y}$, determine the most probable hidden sequence of states $\mathbf{x}$ that emit them:
$$\arg\max_{\mathbf{x}} p(\mathbf{X} = \mathbf{x}|\mathbf{Y} = \mathbf{y}).$$

3. Given the state space S and observation space E, an observation sequence $\mathbf{y}$, learn the HMM parameters $\mathbf{A}$ and $\mathbf{B}$.

4.1 Computing Observation Likelihood: Forward Algorithm

In this subsection, we would resolve the first problem associated with HMM. To refresh, we now state the task here again:

- Given a HMM and an observation sequence $\mathbf{y}$, compute the observation likelihood $p(\mathbf{Y} = \mathbf{y})$.

Although we have the formula (3.7), to sum up all N^T possible state sequence x is too heavy a task. In fact, we could exploit the optimal structure within the formula above to develop an efficient algorithm using **dynamic programming** idea. We strongly recommend the readers to refer to further details about dynamic programming in the appendix.

We need to seek a recursive relation within the formula above.

4.1.1 Forward Algorithm

So here is the idea let
$$\boldsymbol{\alpha}(t, j) = p(\mathbf{Y}_0^t, X_t = j), \quad (0 \leq t \leq T + 1, 0 \leq j \leq N + 1),$$
we point out the following recurrence relationship:
$$\boldsymbol{\alpha}(t, j) = \sum_{i=0}^{N+1} \boldsymbol{\alpha}(t - 1, i)\mathbf{A}(i, j)\mathbf{B}(j, y_t),$$
In particular, we have initialization condition:
$$\boldsymbol{\alpha}(1, k) = \mathbf{A}(0, k)\mathbf{B}(k, y_1).$$
recursion condition:
$$\boldsymbol{\alpha}(t, j) = \sum_{i=1}^{N} \boldsymbol{\alpha}(t - 1, i)\mathbf{A}(i, j)\mathbf{B}(j, y_t), \quad 1 < t \leq T, 0 < j \leq N,$$
and the final answer is given by (termination condition):
$$p(\mathbf{Y}) = \boldsymbol{\alpha}(T + 1, N + 1) = \sum_{i=1}^{N} \boldsymbol{\alpha}(T, i)\mathbf{A}(i, N + 1)$$

Proof.

$$\begin{aligned}
\boldsymbol{\alpha}(t,j) &= p(\mathbf{Y}_0^t, X_t = j) \\
&= \sum_{i=0}^{N+1} p(\mathbf{Y}_0^t, X_{t-1} = i, X_t = j) \\
&= \sum_{i=0}^{N+1} p(Y_t = y_t | \mathbf{Y}_0^{t-1}, X_{t-1} = i, X_t = j) p(\mathbf{Y}_0^{t-1}, X_{t-1} = i, X_t = j) \\
&= \sum_{i=0}^{N+1} p(Y_t = y_t | X_t = j) p(\mathbf{Y}_0^{t-1}, X_{t-1} = i, X_t = j) \\
&= \sum_{i=0}^{N+1} p(Y_t = y_t | X_t = j) p(X_t = j | \mathbf{Y}_0^{t-1}, X_{t-1} = i) p(\mathbf{Y}_0^{t-1}, X_{t-1} = i) \\
&= \sum_{i=0}^{N+1} p(Y_t = y_t | X_t = j) p(X_t = j | (X_{t-1} = i) p(\mathbf{Y}_0^{t-1}, X_{t-1} = i) \\
&= \sum_{i=0}^{N+1} \mathbf{B}(j, y_t) \mathbf{A}(i,j) \boldsymbol{\alpha}(t-1, i).
\end{aligned}$$

In particular,

$$\boldsymbol{\alpha}(1,k) = \sum_{i=0}^{N+1} \boldsymbol{\alpha}(0,i) \mathbf{A}(i,k) \mathbf{B}(k, y_1)$$

notice that using assumptions given previously, that

$$\boldsymbol{\alpha}(0,i) = p(Y_0 = y_0, X_0 = i) = \delta_{0i},$$

thus we have the initialization shown above.

And for $1 < t \leq T, 0 < j \leq N$, we have

$$\boldsymbol{\alpha}(t,0) = \sum_{i=0}^{N+1} \boldsymbol{\alpha}(t-1,i) A(i,0) B(0, y_t),$$

and since $\mathbf{B}(0, y_t) = 0$, $\boldsymbol{\alpha}(t,0)$ is also 0. Second, we have $\mathbf{A}(N+1, j) = 0$. These 2 facts lead us to the recursion condition shown above.

At last,

$$\boldsymbol{\alpha}(T+1, N+1) = \sum_{i=0}^{N+1} \boldsymbol{\alpha}(T,i) \mathbf{A}(i, N+1) \mathbf{B}(N+1, y_{T+1}),$$

and we already know that $\alpha(T,0) = 0$. Besides,

$$\boldsymbol{\alpha}(T, N+1) = \sum_{i=0}^{N+1} \boldsymbol{\alpha}(T-1,i) \mathbf{A}(i, N+1) \mathbf{B}(N+1, y_T),$$

and since $\mathbf{B}(N+1, y_T) = 0$, we know $\boldsymbol{\alpha}(T, N+1) = 0$. Finally, $\mathbf{B}(N+1, y_{T+1}) = 1$. So this leads us to termination step, and that concludes the proof.

So in fact in constructing the matrix $\boldsymbol{\alpha}(t,j)$ we can compute the likelihood of observations. It is easy to know that this algorithm has complexity $O(N^2T)$, which is way faster that the original $O(T^2N^T)$.

4.1.2 Backward Algorithm

Here we need to compute a similar probability, which will be useful in training HMM. Define:

$$\boldsymbol{\beta}(i,k) = p(\mathbf{Y}_{k+1}^{T+1} = \mathbf{y}_{k+1}^{T+1} | X_k = i). \tag{4.1}$$

Here we point out the recursive relation within $\boldsymbol{\beta}$.

$$\boldsymbol{\beta}(i,T) = 1, \tag{4.2}$$

$$\boldsymbol{\beta}(i,k) = \sum_{j=1}^{N} \mathbf{A}(i,j)\mathbf{B}(j,y_{k+1})\boldsymbol{\beta}(j,k+1). \tag{4.3}$$

where in equation (4.3) we restrict $1 \le i \le N+1, 1 \le k < T$.

Proof. The idea behind the proof is a little bit different from that of $\boldsymbol{\alpha}$. We want to find a recursive structure in $\boldsymbol{\beta}$. Note that in equation (4.1) we are computing the likelihood of observations from time $k+1$ to $T+1$, by only knowing the process was in state i at time k. This information is too "old". On the other hand, the only equations that represent the structure of HMM, and that we can rely on to expand the expression above, are equation from (3.1) to (3.4), which require latest history. So we need to supplement missing information from latest history in equation (4.1) by:

$$\begin{aligned}
\boldsymbol{\beta}(i,k) = p\left(\mathbf{Y}_{k+1}^{T+1} \middle| X_k = i\right) &= \frac{p\left(\mathbf{Y}_{k+1}^{T+1}, X_k = i\right)}{p(X_k = i)} \\
&= \frac{1}{p(X_k = i)} \sum_{\mathbf{j}_{k+1}^{T+1}} p\left(\mathbf{Y}_{k+1}^{T+1}, \mathbf{X}_k^{T+1} = \left(i, \mathbf{j}_{k+1}^{T+1}\right)\right)
\end{aligned}$$

where we have use the short hand notation $\mathbf{j}_{k+1}^{T+1} = (j_{k+1}, \ldots, j_{T+1})$. Now let's focus on the term in summation.

$$\begin{aligned}
p\left(\mathbf{Y}_{k+1}^{T+1}, \mathbf{X}_k^{T+1}\right) &= p\left(Y_{T+1} \middle| \mathbf{Y}_{k+1}^{T}, \mathbf{X}_k^{T+1}\right) p\left(\mathbf{Y}_{k+1}^{T}, \mathbf{X}_k^{T+1}\right) \\
&= p(Y_{T+1}|X_{T+1}) p\left(X_{T+1} \middle| \mathbf{Y}_{k+1}^{T}, \mathbf{X}_k^{T}\right) p\left(\mathbf{Y}_{k+1}^{T}, \mathbf{X}_k^{T}\right) && (4.4) \\
&= p(Y_{T+1}|X_{T+1}) p(X_{T+1}|X_T) p\left(\mathbf{Y}_{k+1}^{T}, \mathbf{X}_k^{T}\right) && (4.5) \\
&= \cdots \\
&= p(X_k = i) \prod_{s=k}^{T} \mathbf{A}(j_s, j_{s+1}) \prod_{t=k+1}^{T+1} \mathbf{B}(j_t, y_t).
\end{aligned}$$

where $j_k = i$, equation (4.4) by output independence assumption (3.3), and equation (4.5) by (3.1).

Divide the expression above by $p(X_k = i)$ and notice the recursive structure we get:

$$\begin{aligned}\boldsymbol{\beta}(i,k) &= \sum_{\mathbf{j}_{k+1}^{T+1}} \prod_{s=k}^{T} \mathbf{A}(j_s, j_{s+1}) \prod_{t=k+1}^{T+1} \mathbf{B}(j_t, y_t) \\ &= \sum_{j_{k+1}} \mathbf{A}(i, j_{k+1}) \mathbf{B}(j_{k+1}, y_{k+1}) \sum_{\mathbf{j}_{k+2}^{T+1}} \prod_{s=k+1}^{T} \mathbf{A}(j_s, j_{s+1}) \prod_{t=k+2}^{T+1} \mathbf{B}(j_t, y_t) \\ &= \sum_{j} \mathbf{A}(i,j) \mathbf{B}(j, y_{k+1}) \boldsymbol{\beta}(j, k+1).\end{aligned}$$

And that completes the proof.

Remark. One can simply apply the equation derived from the short memory property of HMM, that is, equation (3.9) to directly prove this.

4.2 Decoding: Viterbi Algorithm

In this subsection, we are going to introduce the Viterbi algorithm to address decoding problem:

- Given a HMM and an observation sequence $\mathbf{y}$, determine the most probable hidden sequence of states $\mathbf{x}$ that emit them:
$$\arg\max_{\mathbf{x}} p(\mathbf{X} = \mathbf{x} | \mathbf{Y} = \mathbf{y}).$$

We can actually formulate the problem in a slightly different way:

$$\begin{aligned}\arg\max_{\mathbf{x}} p(\mathbf{X}|\mathbf{Y}) &= \arg\max_{\mathbf{x}} \frac{p(\mathbf{X}, \mathbf{Y})}{p(\mathbf{Y})} \\ &= \arg\max_{\mathbf{x}} p(\mathbf{X}, \mathbf{Y}) \\ &= \arg\max_{\mathbf{x}} \prod_{k=1}^{T} \mathbf{B}(x_k, y_k) \prod_{k=1}^{T+1} \mathbf{A}(x_{k-1}, x_k),\end{aligned}$$

which seems that we need to search over all N^T possible x, again which is inapplicable. So we, once again, need to exploit the recursive relationship within the above formula. Let $\mathbf{V}(t,k)$ be the probability of the most probable state sequence responsible for the first t observations that has j as its final state:

$$\mathbf{V}(t,j) = \max_{\mathbf{x}_0^{t-1}} p(\mathbf{X}_0^{t-1} = \mathbf{x}_0^{t-1}, X_t = j, \mathbf{Y}_0^t = \mathbf{y}_0^t).$$

We point out the following recurrence relationship:

$$\mathbf{V}(t,j) = \max_{i \in S} \mathbf{A}(i,j) \mathbf{B}(j, y_t) \mathbf{V}(t-1, i), \tag{4.6}$$

and in particular, we have initialization condition:

$$\mathbf{V}(1,j) = \mathbf{A}(0,j) \mathbf{B}(j, y_1),$$

and termination condition, the probability of the most probable state sequence:

$$\mathbf{V}(T+1, N+1) = \max_{\mathbf{x}} p(\mathbf{X} = \mathbf{x}|\mathbf{Y}) = \max_{i \in S} \mathbf{A}(i, N+1) \mathbf{B}(N+1, y_{T+1}) \mathbf{V}(T, i).$$

And let $\mathbf{K}(t,j)$ be the state at time $t-1$ in the most probable state sequence responsible for the first t observations that has j as its final state:

$$\mathbf{K}(t,j) = \arg_{x_{t-1}} \max_{\mathbf{x}_0^{t-1}} p(\mathbf{X}_0^{t-1} = \mathbf{x}_0^{t-1}, X_t = j, \mathbf{Y}_0^t = \mathbf{y}_0^t).$$

We point out the recursion condition:

$$\mathbf{K}(t,j) = \arg\max_{i \in S} \mathbf{A}(i,j)\mathbf{V}(t-1,i).$$

And in particular, we have initialization condition, which indicates that process always begins from start state:

$$\mathbf{K}(1,j) = 0,$$

And termination condition, which is the start of back tracing the most probable state sequence.

$$\mathbf{K}(T+1, N+1) = \arg\max_{i \in S} \mathbf{A}(i, N+1)\mathbf{V}(T,i)$$

Proof. We only prove for recurrence relationship (4.6).

$$\begin{aligned}
\mathbf{V}(t,j) &= \max_{\mathbf{x}_0^{t-1}} p\left(Y_t \left| \mathbf{X}_0^t, \mathbf{Y}_0^{t-1} \right.\right) p\left(\mathbf{X}_0^t, \mathbf{Y}_0^{t-1}\right) \\
&= \max_{\mathbf{x}_0^{t-1}} p(Y_t | X_t = j) p\left(X_t \left| \mathbf{X}_0^{t-1}, \mathbf{Y}_0^{t-1} \right.\right) p\left(\mathbf{X}_0^{t-1}, \mathbf{Y}_0^{t-1}\right) \\
&= \max_{\mathbf{x}_0^{t-2}} \max_{x_{t-1}} \mathbf{B}(j, y_t) p(X_t | X_{t-1}) p\left(\mathbf{X}_0^{t-1}, \mathbf{Y}_0^{t-1}\right) \\
&= \max_{x_{t-1}} \mathbf{B}(j, y_t)\mathbf{A}(x_{t-1}, j) \max_{\mathbf{x}_0^{t-2}} p\left(\mathbf{X}_0^{t-1}, \mathbf{Y}_0^{t-1}\right) \\
&= \max_{i \in S} \mathbf{A}(i,j)\mathbf{B}(j, y_t)\mathbf{V}(t-1, i).
\end{aligned}$$

The construction procedure of $\mathbf{K}$ follows immediately.

4.3 Training/Learning: Baum-Welch Algorithm

In this subsection, we would use EM algorithm idea to derive the Baum-Welch algorithm [1]. See [1] for the original discussion brought by Baum in 1966. And use this algorithm to train the HMM. The derivation of Baum-Welch Algorithm using EM algorithm has referred to this article [3].

To refresh, let us state the goal again:

- Given the state space S and observation space E, an observation sequence $\mathbf{y}$, learn the HMM parameters $\mathbf{A}$ and $\mathbf{B}$.

If we replace here matrix $\mathbf{A}$ and $\mathbf{B}$ with $\boldsymbol{\theta}$, that is $\boldsymbol{\theta} = (\mathbf{A}, \mathbf{B})$, to represent their parametric nature, we can rewrite our objective as below:

- Given the state space S and observation space E, an observation sequence $\mathbf{y}$, learn the HMM parameter $\boldsymbol{\theta}$

  $$\arg\max_{\boldsymbol{\theta}} p(\mathbf{Y} = \mathbf{y}; \boldsymbol{\theta}),$$

  to maximize the likelihood of observation $\mathbf{y}$.

[1] We will see in this subsection that there are very natural motivations and clear logic about how the training algorithm is developed.

We recognize in this problem, we have known data $\mathbf{Y} = \mathbf{y}$ and latent variables $\mathbf{X}$, and we want to find the parameter $\boldsymbol{\theta}$ to maximize the likelihood of known data. This is exactly what EM algorithm does [2]. For further discussion on EM algorithm, please refer to the appendix.

So now it is natural for us to derive an algorithm to train the HMM using EM algorithm idea. Let's first write down the expectation function at t iteration:

$$\begin{aligned} Q\left(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}\right) &= \sum_{\mathbf{x}} p(\mathbf{X}|\mathbf{Y};\boldsymbol{\theta}^{(t)}) \log p\left(\mathbf{X},\mathbf{Y};\boldsymbol{\theta}\right) \\ &= \sum_{\mathbf{x}} p(\mathbf{X}|\mathbf{Y};\boldsymbol{\theta}^{(t)}) \left(\sum_{k=1}^{T+1} \log \mathbf{A}(x_{k-1},x_k) + \sum_{l=1}^{T} \log \mathbf{B}(x_l,y_l) \right). \end{aligned}$$

Note that we have used equation (3.6).

We need to maximize the above expression in M-step of the EM algorithm. But first recall that there are constraints in parameter $\boldsymbol{\theta}$, namely equation (2.1) and (3.5). Lagrange multiplier method seems to be perfect for this. So we introduce Lagrange multipliers $\lambda_0, \ldots, \lambda_{N+1}$ and $\mu_1, \ldots, \mu_N$ to enforce those constraints, and construct our new objective function:

$$R(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) = Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) + \sum_{i=0}^{N+1} \lambda_i \left(1 - \sum_{j=0}^{N+1} \mathbf{A}(i,j)\right) + \sum_{i=1}^{N} \mu_i \left(1 - \sum_{j=1}^{M} \mathbf{B}(i,j)\right). \tag{4.7}$$

Taking derivative and equate it to 0:

$$\frac{\partial R(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)})}{\partial \mathbf{A}(i,j)} = 0.$$

Or write it as:

$$\begin{aligned} \frac{\partial}{\partial \mathbf{A}(i,j)} \sum_{\mathbf{x}} \sum_{k=0}^{T} p(\mathbf{X}|\mathbf{Y};\boldsymbol{\theta}^{(t)}) \log \mathbf{A}(x_k,x_{k+1}) &= \lambda_i, \\ \sum_{\mathbf{x}} \sum_{k=0}^{T} p(\mathbf{X}|\mathbf{Y};\boldsymbol{\theta}^{(t)}) \frac{\partial}{\partial \mathbf{A}(i,j)} \log \mathbf{A}(x_k,x_{k+1}) &= \lambda_i, \\ \sum_{\mathbf{x}} \sum_{k=0}^{T} p(\mathbf{X}|\mathbf{Y};\boldsymbol{\theta}^{(t)}) \frac{[x_k = i, x_{k+1} = j]}{\mathbf{A}(i,j)} &= \lambda_i, \\ \sum_{k=0}^{T} \sum_{\mathbf{x}} p(\mathbf{X}|\mathbf{Y};\boldsymbol{\theta}^{(t)}) [x_k = i, x_{k+1} = j] &= \lambda_i \mathbf{A}(i,j), \\ \sum_{k=0}^{T} p(X_k = i, X_{k+1} = j|\mathbf{Y};\boldsymbol{\theta}^{(t)}) &= \lambda_i \mathbf{A}(i,j), \end{aligned}$$

Sum up both sides of the equation above over j from 0 to $N+1$ and use regularity condition of $\mathbf{A}$ to solve for λ_i, we obtain the updated value of $\mathbf{A}(i,j)$:

$$\mathbf{A}(i,j) = \frac{\sum_{k=0}^{T} p(X_k = i, X_{k+1} = j|\mathbf{Y};\boldsymbol{\theta}^{(t)})}{\sum_{k=0}^{T} p(X_k = i|\mathbf{Y};\boldsymbol{\theta}^{(t)})}.$$

[2] It is very important for one to recognize a problem as fit in resolved by EM algorithm. So then he/she can derive a particular algorithm for the problem using EM algorithm idea.

Similarly, from equation

$$\frac{\partial R(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)})}{\partial \mathbf{B}(i,j)} = 0,$$

we have:

$$\begin{aligned}
\frac{\partial}{\partial \mathbf{B}(i,j)} \sum_{\mathbf{x}} \sum_{k=1}^{T} p\left(\mathbf{X}|\mathbf{Y};\boldsymbol{\theta}^{(t)}\right) \log \mathbf{B}(x_k, y_k) &= \mu_i, \\
\sum_{\mathbf{x}} \sum_{k=1}^{T} p\left(\mathbf{X}|\mathbf{Y};\boldsymbol{\theta}^{(t)}\right) \frac{\partial}{\partial \mathbf{B}(i,j)} \log \mathbf{B}(x_k, y_k) &= \mu_i, \\
\sum_{\mathbf{x}} \sum_{k=1}^{T} p\left(\mathbf{X}\left|\mathbf{Y};\boldsymbol{\theta}^{(t)}\right.\right) \frac{[x_k = i, y_k = j]}{\mathbf{B}(i,j)} &= \mu_i, \\
\sum_{k=1}^{T} \sum_{\mathbf{x}} p\left(\mathbf{X}\left|\mathbf{Y};\boldsymbol{\theta}^{(t)}\right.\right) [x_k = i, y_k = j] &= \mu_i \mathbf{B}(i,j), \\
\sum_{k=1}^{T} p\left(X_k = i\left|\mathbf{Y};\boldsymbol{\theta}^{(t)}\right.\right) [y_k = j] &= \mu_i \mathbf{B}(i,j),
\end{aligned}$$

Sum up both sides of the equation above over j from 0 to M and use regularity condition of $\mathbf{B}$ to solve for μ_i, we obtain the updated value of $\mathbf{B}(i,j)$:

$$\mathbf{B}(i,j) = \frac{\sum_{k=1}^{T} p\left(X_k = i\left|\mathbf{Y};\boldsymbol{\theta}^{(t)}\right.\right) [y_k = j]}{\sum_{k=1}^{T} p\left(X_k = i\left|\mathbf{Y};\boldsymbol{\theta}^{(t)}\right.\right)}.$$

Next, we only need to apply the forward-backward algorithm to compute the probabilities appear in M-step above. To simplify, let

$$\begin{aligned}
\boldsymbol{\alpha}^{(t)}(i,k) &= p\left(\mathbf{Y}_0^k = \mathbf{y}_0^k, X_k = i; \boldsymbol{\theta}^{(t)}\right), \\
\boldsymbol{\beta}^{(t)}(i,k) &= p\left(\mathbf{Y}_{k+1}^{T+1} = \mathbf{y}_{k+1}^{T+1}\left|X_k = i; \boldsymbol{\theta}^{(t)}\right.\right), \\
\boldsymbol{\gamma}^{(t)}(i,k) &= p\left(X_k = i\left|\mathbf{Y};\boldsymbol{\theta}^{(t)}\right.\right), \\
\boldsymbol{\xi}^{(t)}(i,j,k) &= p\left(X_k = i, X_{k+1} = j\left|\mathbf{Y};\boldsymbol{\theta}^{(t)}\right.\right).
\end{aligned}$$

Since all operations we discuss later happen in the same iteration, the supper script (t) which indicates iteration time is ignored.

It is already proved the following forward and backward recursive procedures:

Forward Procedure

$$\begin{aligned}
\boldsymbol{\alpha}(i,1) &= \mathbf{A}(0,i)\mathbf{B}(i,y_1), \\
\boldsymbol{\alpha}(i,k) &= \mathbf{B}(i,y_k) \sum_{j=1}^{N} \alpha(j,k-1)\mathbf{A}(j,i),
\end{aligned}$$

where $1 \leq i \leq N, 1 < k \leq T$.

Backward Procedure

$$\begin{aligned}\beta(i,T) &= 1,\\ \beta(i,k) &= \sum_{j=1}^{N} \beta(j,k+1)\mathbf{A}(i,j)\mathbf{B}(j,y_{k+1}),\end{aligned}$$

where $1 \leq i \leq N, 1 \leq k < T$.

We can now calculate the temporary variables, according to Bayes' theorem:

$$\begin{aligned}\boldsymbol{\gamma}(i,k) &= \frac{\boldsymbol{\alpha}(i,k)\boldsymbol{\beta}(i,k)}{\sum_{j=1}^{N} \boldsymbol{\alpha}(j,k)\boldsymbol{\beta}(j,k)},\\ \boldsymbol{\xi}(i,j,k) &= \frac{\boldsymbol{\alpha}(i,k)\mathbf{A}(i,j)\boldsymbol{\beta}(j,k+1)\mathbf{B}(j,y_{k+1})}{\sum_{i,j=1}^{N} \boldsymbol{\alpha}(i,k)\mathbf{A}(i,j)\boldsymbol{\beta}(j,k+1)\mathbf{B}(j,y_{k+1})}.\end{aligned}$$

And update our parameters $\mathbf{A}, \mathbf{B}$ by:

$$\mathbf{A}^{(t+1)}(i,j) = \frac{\sum_{k=0}^{T} \boldsymbol{\xi}^{(t)}(i,j,k)}{\sum_{k=0}^{T} \boldsymbol{\gamma}^{(t)}(i,k)},$$

where $0 \leq i \leq N, 1 \leq j \leq N+1$.

$$\mathbf{B}^{(t+1)}(i,j) = \frac{\sum_{k=1}^{T} \boldsymbol{\gamma}^{(t)}(i,k)[y_k = j]}{\sum_{k=1}^{T} \boldsymbol{\gamma}^{(t)}(i,k)},$$

where $1 \leq i \leq N, 1 \leq j \leq M$.

Finally, iterate until convergence. And this is the derivation and procedure of Baum-Welch algorithm.

We now give a brief proof about the computation procedures of $\boldsymbol{\gamma}(i,k)$ and $\boldsymbol{\xi}(i,j,k)$.

Proof. By Bayes' rule:

$$\boldsymbol{\gamma}(i,k) = \frac{p\left(X_k, \mathbf{Y}\right)}{\sum_{j=0}^{N+1} p\left(X_k = j | \mathbf{Y}\right)},$$

And since the numerator can be written as:

$$p\left(X_k, \mathbf{Y}_0^k, \mathbf{Y}_{k+1}^{T+1}\right) = p\left(\mathbf{Y}_{k+1}^{T+1} \left| X_k, \mathbf{Y}_0^k \right.\right) p\left(X_k, \mathbf{Y}_0^k\right),$$

which by short memory property of HMM (3.8), is equal to:

$$\boldsymbol{\alpha}(i,k)\boldsymbol{\beta}(i,k).$$

we can get the desired recursion.

And similarly, by Bayes' rule, we have:

$$\boldsymbol{\xi}(i,j,k) = \frac{p\left(X_k, X_{k+1}, \mathbf{Y}\right)}{\sum_{s,t=0}^{N+1} p\left(X_k = s, X_{k+1} = t, \mathbf{Y}\right)}.$$

And since the numerator can be written as:

$$\begin{aligned}
&p\left(X_k, X_{k+1}, \mathbf{Y}_0^k, Y_{k+1}, \mathbf{Y}_{k+2}^{T+1}\right) \\
&= p\left(\mathbf{Y}_{k+2}^{T+1} \left| \mathbf{Y}_0^{k+1}, \mathbf{X}_k^{k+1} \right.\right) p\left(\mathbf{X}_k^{k+1}, \mathbf{Y}_0^{k+1}\right) \\
&= p\left(\mathbf{Y}_{k+2}^{T+1} \left| X_{k+1} \right.\right) p\left(Y_{k+1} \left| \mathbf{X}_k^{k+1}, \mathbf{Y}_0^k \right.\right) p\left(\mathbf{X}_k^{k+1}, \mathbf{Y}_0^k\right) \\
&= \boldsymbol{\beta}(j, k+1) p(Y_{k+1}|X_{k+1}) p\left(X_{k+1} \left| X_k, \mathbf{Y}_0^k \right.\right) p\left(X_k, \mathbf{Y}_0^k\right) \\
&= \boldsymbol{\beta}(j, k+1) \mathbf{B}(j, y_{k+1}) p(X_{k+1}|X_k) \boldsymbol{\alpha}(i, k),
\end{aligned}$$

and this obviously completes the proof. Note that we have used the short memory property of HMM.

A Dynamic Programming

There are two key attributes that a problem must have in order for dynamic programming to be applicable [4]: **optimal substructure** and **overlapping sub-problems**. If a problem can be solved by combining optimal solutions to non-overlapping sub-problems, the strategy is called "divide and conquer" instead.

Optimal substructure means that the solution to a given optimization problem can be obtained by the combination of optimal solutions to its sub-problems. Such optimal substructures are usually described by means of recursion.

Overlapping sub-problems means that the space of sub-problems must be small, that is, any recursive algorithm solving the problem should solve the same sub-problems over and over, rather than generating new sub-problems.

B Probability and Statistics

B.1 Gibbs' inequality

Suppose that

$$P = p_1, \ldots, p_n, \quad Q = q_1, \ldots, q_n$$

are any probability distributions, so that we have:

$$\sum_{i=1}^{n} p_i = \sum_{i=1}^{n} q_i = 1.$$

The Gibbs' inequality states that the information entropy of a distribution P is less than or equal to its cross entropy with any other distribution Q:

$$-\sum_{i=1}^{n} p_i \ln p_i \leq -\sum_{i=1}^{n} p_i \ln q_i.$$

Proof. First we apply Jessen's inequality to the right hand side. Note that function $f(x) = -\ln x$ is convex, so we have:

$$-\ln\left(\sum_{i=1}^{n} p_i q_i\right) \leq -\sum_{i=1}^{n} p_i \ln q_i.$$

On the other hand, we want to apply Jessen's inequality to the left hand side, and to prove it is less than or equal to the right hand side, we do the following modification:

$$-\sum_{i=1}^{n} p_i \ln p_i \leq -\sum_{i=1}^{n} q_i p_i \ln p_i. \tag{B.1}$$

Now apply Jessen's inequality we get (noticing function $g(x) = -x \ln x$ is concave):

$$\text{(B.1)} \leq -\left(\sum_{i=1}^{n} p_i q_i\right) \ln \left(\sum_{i=1}^{n} p_i q_i\right). \tag{B.2}$$

Note that we have inequality:

$$\sum_{i=1}^{n} p_i q_i \leq \left(\sum_{i=1}^{n} p_i\right) \left(\sum_{i=1}^{n} q_i\right) = 1.$$

So thus we get

$$\text{(B.2)} \leq -\ln \left(\sum_{i=1}^{n} p_i q_i\right),$$

and that completes the proof.

B.2 EM Algorithm

In statistics, an expectation-maximization (EM) algorithm is an iterative method to find maximum likelihood or maximum a posteriori (MAP) estimates of parameters in statistical models, where the model depends on unobserved latent variables [5].

B.2.1 Derivation [3]

Given the statistical model which generates a set of observed data $\mathbf{X}$, a set of unobserved latent data or missing values $\mathbf{Z}$, and a vector of unknown parameters $\boldsymbol{\theta}$, along with a likelihood function $L(\boldsymbol{\theta}; \mathbf{X}, \mathbf{Z}) = p(\mathbf{X}, \mathbf{Z}|\boldsymbol{\theta})$, the maximum likelihood estimate (MLE) of the unknown parameters is determined by the marginal likelihood of the observed data

$$L(\boldsymbol{\theta}; \mathbf{X}) = p(\mathbf{X}|\boldsymbol{\theta}) = \sum_{\mathbf{Z}} p(\mathbf{X}, \mathbf{Z}|\boldsymbol{\theta})$$

However, this quantity is often intractable (e.g. if $\mathbf{Z}$ is a sequence of events, so that the number of values grows exponentially with the sequence length, making the exact calculation of the sum extremely difficult).

We now derive the EM algorithm. Consider the following inequality (using Jessen's inequality):

$$\begin{aligned} l(\boldsymbol{\theta}) &= \log p(\mathbf{X}; \boldsymbol{\theta}) = \log \sum_{\mathbf{Z}} p(\mathbf{X}, \mathbf{Z}; \boldsymbol{\theta}) \\ &= \log \sum_{\mathbf{Z}} q(\mathbf{Z}|\mathbf{X}; \boldsymbol{\theta}) \frac{p(\mathbf{X}, \mathbf{Z}; \boldsymbol{\theta})}{q(\mathbf{Z}|\mathbf{X}; \boldsymbol{\theta})} \\ &\geq \sum_{\mathbf{Z}} q(\mathbf{Z}|\mathbf{X}; \boldsymbol{\theta}) \log \left(\frac{p(\mathbf{X}, \mathbf{Z}; \boldsymbol{\theta})}{q(\mathbf{Z}|\mathbf{X}; \boldsymbol{\theta})}\right) := F(q, \boldsymbol{\theta}) \end{aligned} \tag{B.3}$$

where $q(\mathbf{Z}|\mathbf{X};\boldsymbol{\theta})$ is an arbitrary conditional probability distribution of $\mathbf{Z}$ given $\mathbf{X}$, and $F(q,\boldsymbol{\theta})$ is called negative variational free energy in information theory.

Instead of maximizing $l(\boldsymbol{\theta})$ directly, we would seek to maximize the lower bound $F(q,\boldsymbol{\theta})$ via coordinate ascent:

$$q^{(t+1)} = \arg\max_{q} F(q, \boldsymbol{\theta}^{(t)}) \tag{B.4}$$

$$\boldsymbol{\theta}^{(t+1)} = \arg\max_{\boldsymbol{\theta}} F(q^{(t+1)}, \boldsymbol{\theta}) \tag{B.5}$$

So by iterating between these 2 steps, one can improve $l(\boldsymbol{\theta})$. And this is the original idea of EM algorithm, which also guarantees convergence.

However, computation in the first step involves optimizing $F(q,\boldsymbol{\theta}^{(t)})$ over the space of all possible distribution $q(\mathbf{Z}|\mathbf{X};\boldsymbol{\theta}^{(t)})$, which is inapplicable. So we need to find its closed form analytically.

Recall we want to find the (discrete) distribution q that maximizes the following:

$$F(q,\boldsymbol{\theta}^{(t)}) = \sum_{\mathbf{Z}} q(\mathbf{Z}|\mathbf{X};\boldsymbol{\theta}^{(t)}) \log\left(\frac{p(\mathbf{X},\mathbf{Z};\boldsymbol{\theta}^{(t)})}{q(\mathbf{Z}|\mathbf{X};\boldsymbol{\theta}^{(t)})}\right)$$

Since in this case the parameter $\boldsymbol{\theta}^{(t)}$ is assumed to be correct, we would ignore it in our following equations, for example, rewrite our objective function as:

$$F(q) = \sum_{\mathbf{Z}} q(\mathbf{Z}|\mathbf{X}) \log\left(\frac{p(\mathbf{X},\mathbf{Z})}{q(\mathbf{Z}|\mathbf{X})}\right).$$

Now we apply the Lagrange multiplier method to perform maximization [2]. Introducing a Lagrange multiplier λ to enforce the constraint $\sum_{\mathbf{Z}} q(\mathbf{Z}|\mathbf{X}) = 1$, our objective function becomes:

$$G(q) = \lambda\left(1 - \sum_{\mathbf{Z}} q(\mathbf{Z}|\mathbf{X})\right) + \sum_{\mathbf{Z}} q(\mathbf{Z}|\mathbf{X}) \log p(\mathbf{X},\mathbf{Z}) - \sum_{\mathbf{Z}} q(\mathbf{Z}|\mathbf{X}) \log q(\mathbf{Z}|\mathbf{X}). \tag{B.6}$$

Taking the derivative:

$$\frac{\partial G}{\partial q(\mathbf{Z}|\mathbf{X})} = -\lambda + \log p(\mathbf{X},\mathbf{Z}) - \log q(\mathbf{Z}|\mathbf{X}) - 1,$$

and taking exponential to both sides, we can solve for $q(\mathbf{Z}|\mathbf{X})$ we obtain:

$$q(\mathbf{Z}|\mathbf{X}) = \frac{p(\mathbf{X},\mathbf{Z})}{\sum_{\mathbf{Z}} p(\mathbf{X},\mathbf{Z})} = p(\mathbf{Z}|\mathbf{X})$$

Finally since:

$$F(p) = \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}) \log\left(\frac{p(\mathbf{X},\mathbf{Z})}{p(\mathbf{Z}|\mathbf{X})}\right) = \log p(\mathbf{X}),$$

and recall from equation (B.3) we find that $q^{(t+1)} = p(\mathbf{Z}|\mathbf{X};\boldsymbol{\theta}^{(t)})$ indeed reaches the maximum of the initial objective function $F(q)$.

So now we no longer need to find $q^{(t+1)}$ computationally in equation (B.4) but only need to compute the objective function for optimization in equation (B.5):

$$\begin{aligned} F(q^{(t+1)}, \boldsymbol{\theta}) &= \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}; \boldsymbol{\theta}^{(t)}) \log \left(\frac{p(\mathbf{X}, \mathbf{Z}; \boldsymbol{\theta})}{p(\mathbf{Z}|\mathbf{X}; \boldsymbol{\theta}^{(t)})} \right) \\ &= \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}; \boldsymbol{\theta}^{(t)}) \log p(\mathbf{X}, \mathbf{Z}; \boldsymbol{\theta}) - \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}; \boldsymbol{\theta}^{(t)}) \log p(\mathbf{Z}|\mathbf{X}; \boldsymbol{\theta}^{(t)}) \\ &= \mathrm{E}_{\mathbf{Z}|\mathbf{X}, \boldsymbol{\theta}^{(t)}} \left[\log L(\boldsymbol{\theta}; \mathbf{X}, \mathbf{Z}) \right] + H(p(\mathbf{Z}|\mathbf{X}; \boldsymbol{\theta}^{(t)})) \end{aligned}$$

The second term $H(p(\mathbf{Z}|\mathbf{X}; \boldsymbol{\theta}^{(t)}))$ in the equation above is the entropy of conditional distribution $p(\mathbf{Z}|\mathbf{X}; \boldsymbol{\theta}^{(t)})$, which is independent of $\boldsymbol{\theta}$. So all we need to maximize is the first term, which we will denote as:

$$Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) = \mathrm{E}_{\mathbf{Z}|\mathbf{X}, \boldsymbol{\theta}^{(t)}} \left[\log L(\boldsymbol{\theta}; \mathbf{X}, \mathbf{Z}) \right]$$

or in words, it is the expected value of the log likelihood function, with respect to the conditional distribution of $\mathbf{Z}$ given $\mathbf{X}$ under the current estimate of the parameters $\boldsymbol{\theta}^{(t)}$.

Now we can restate the EM algorithm, which seeks to find the MLE of the marginal likelihood by iteratively applying these two steps:

Expectation step (E step): Calculate expected value:

$$Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) = \mathrm{E}_{\mathbf{Z}|\mathbf{X}, \boldsymbol{\theta}^{(t)}} \left[\log L(\boldsymbol{\theta}; \mathbf{X}, \mathbf{Z}) \right]$$

Maximization step (M step): Find the parameter that maximizes this quantity:

$$\boldsymbol{\theta}^{(t+1)} = \arg\max_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)})$$

So the overall effect of one iteration of EM algorithm is to choose a new parameter $\boldsymbol{\theta}^{(t+1)}$ to maximize the expected value of the log likelihood function.

Remark. In discrete case, we may compute the expected value like the following:

$$\mathrm{E}_{\mathbf{Z}|\mathbf{X}, \boldsymbol{\theta}^{(t)}} \left[\log L(\boldsymbol{\theta}; \mathbf{X}, \mathbf{Z}) \right] = \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}; \boldsymbol{\theta}^{(t)}) \log L(\boldsymbol{\theta}; \mathbf{X}, \mathbf{Z}).$$

B.2.2 Proof of Convergence [5]

Expectation-maximization works to improve $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)})$ rather than directly improving $\log p(\mathbf{X}|\boldsymbol{\theta})$. Here is shown that improvements to the former imply improvements to the latter. *Proof.* For any $\mathbf{Z}$ with non-zero probability $p(\mathbf{Z}|\mathbf{X}, \boldsymbol{\theta})$, we can write

$$\log p(\mathbf{X}|\boldsymbol{\theta}) = \log p(\mathbf{X}, \mathbf{Z}|\boldsymbol{\theta}) - \log p(\mathbf{Z}|\mathbf{X}, \boldsymbol{\theta}) \,.$$

We take the expectation over possible values of the unknown data $\mathbf{Z}$ under the current parameter estimate $\theta^{(t)}$ by multiplying both sides by $p(\mathbf{Z}|\mathbf{X}, \boldsymbol{\theta}^{(t)})$ and summing (or integrating) over $\mathbf{Z}$. The left-hand side is the expectation of a constant, so we get:

$$\log p(\mathbf{X}|\boldsymbol{\theta}) = \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}, \boldsymbol{\theta}^{(t)}) \log p(\mathbf{X}, \mathbf{Z}|\boldsymbol{\theta}) - \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{X}, \boldsymbol{\theta}^{(t)}) \log p(\mathbf{Z}|\mathbf{X}, \boldsymbol{\theta}) \tag{B.7}$$

$$= Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) + H(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) \,, \tag{B.8}$$

where $H(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)})$ is defined by the negated sum it is replacing. This last equation holds for any value of $\boldsymbol{\theta}$ including $\boldsymbol{\theta} = \boldsymbol{\theta}^{(t)}$

$$\log p(\mathbf{X}|\boldsymbol{\theta}^{(t)}) = Q(\boldsymbol{\theta}^{(t)}|\boldsymbol{\theta}^{(t)}) + H(\boldsymbol{\theta}^{(t)}|\boldsymbol{\theta}^{(t)})\,,$$

and subtracting this last equation from the previous equation gives

$$\log p(\mathbf{X}|\boldsymbol{\theta}) - \log p(\mathbf{X}|\boldsymbol{\theta}^{(t)}) = Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) - Q(\boldsymbol{\theta}^{(t)}|\boldsymbol{\theta}^{(t)}) + H(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) - H(\boldsymbol{\theta}^{(t)}|\boldsymbol{\theta}^{(t)})\,,$$

However, Gibbs' inequality tells us that

$$H(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) \geq H(\boldsymbol{\theta}^{(t)}|\boldsymbol{\theta}^{(t)}),$$

so we can conclude that

$$\log p(\mathbf{X}|\boldsymbol{\theta}) - \log p(\mathbf{X}|\boldsymbol{\theta}^{(t)}) \geq Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}) - Q(\boldsymbol{\theta}^{(t)}|\boldsymbol{\theta}^{(t)})\,.$$

In words, choosing $\boldsymbol{\theta}$ to improve $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)})$ beyond $Q(\boldsymbol{\theta}^{(t)}|\boldsymbol{\theta}^{(t)})$ can not cause $\log p(\mathbf{X}|\boldsymbol{\theta})$ to decrease below $\log p(\mathbf{X}|\boldsymbol{\theta}^{(t)})$, and so the marginal likelihood of the data is non-decreasing. Finally, we only need to note that $\log p(\mathbf{X}|\boldsymbol{\theta})$ is bounded by 0, the convergence is thus guaranteed.

References

[1] Leonard E. Baum and Ted Petrie. Statistical inference for probabilistic functions of finite state markov chains. *The Annals of Mathematical Statistics*, 37(6):1554–1563, Dec. 1966. Published by: Institute of Mathematical Statistics.

[2] Frank Dellaert. The expectation maximization algorithm. Technical Report GIT-GVU-02-20, College of Computing, Georgia Institute of Technology, February 2002.

[3] Ajit Singh. The em algorithm. November 2005.

[4] Wikipedia. Dynamic programming — wikipedia, the free encyclopedia, 2017. [Online; accessed 19-February-2017].

[5] Wikipedia. Expectation-maximization algorithm — wikipedia, the free encyclopedia, 2017. [Online; accessed 22-February-2017].