# Derivation of Baum-Welch Algorithm for Hidden Markov Models

Stephen Tu

## 1 Introduction

This short document goes through the derivation of the Baum-Welch algorithm for learning model parameters of a hidden markov model (HMM). For more generality, we treat the multiple observations case. Note that Baum-Welch is simply an instantiation of the more general Expectation-Maximization (EM) algorithm.

## 2 Setup

Let us consider discrete (categorical) HMMs of length $T$ (each observation sequence is $T$ observations long). Let the space of observations be $X = \{1, 2, ..., N\}$, and let the space of underlying states be $Z = \{1, 2, ..., M\}$. An HMM $\theta = (\pi, A, B)$ is parameterized by the initial state matrix $\pi$, the state transition matrix $A$, and the emission matrix $B$; $\pi_i = P(z_1 = i)$, $A_{ij} = P(z_{t+1} = j | z_t = 1)$, and $B_i(j) = P(x_t = j | z_t = i)$. See [1] for a more detailed treatment of HMMs.

We study the problem of learning the parameterization of $\theta$ from a dataset of $D$ observations. Let $\mathcal{X} = \left(X^{(1)}, ..., X^{(D)}\right)$, where each $X^{(i)} = (x_1^{(i)}, x_2^{(i)}, ..., x_T^{(i)})$. We assume each observation is drawn iid. The learning problem is non-trivial because we are not given the latent variables $Z^{(i)}$ for each $X^{(i)}$, otherwise we could directly compute $\theta^* = \operatorname{argmax}_\theta P(\mathcal{X}, \mathcal{Z}; \theta)$. Without $\mathcal{Z}$, the naive solution would be to directly compute $\theta^* = \operatorname{argmax}_\theta \sum_{z \in \mathcal{Z}} P(\mathcal{X}, z; \theta)$. This is not tractable, since there are $DT^M$ different values of $z$ to try.

## 3 Baum-Welch

Baum-Welch is an iterative procedure for estimating $\theta^*$ from only $\mathcal{X}$. It works by maximizing a proxy to the log-likelihood, and updating the current model to be closer to the optimal model. Each iteration of Baum-Welch is guaranteed to increase the log-likelihood of the data. But of course, convergence to the optimal solution is not guaranteed.

Baum-Welch can be described simply as repeating the following steps until convergence:

1. Compute $Q(\theta, \theta^s) = \sum\limits_{z \in \mathcal{Z}} \log \left[P(\mathcal{X}, z; \theta)\right] P(z | \mathcal{X}; \theta^s)$.

2. Set $\theta^{s+1} = \operatorname*{argmax}\limits_\theta Q(\theta, \theta^s)$.

Without justifying why this works, the rest of this document will focus on deriving the necessary update steps to run this algorithm. First, noting that $P(z, \mathcal{X}) = P(\mathcal{X}) P(z | \mathcal{X})$, we can write

$$\operatorname*{argmax}_\theta \sum_{z \in \mathcal{Z}} \log \left[P(\mathcal{X}, z; \theta)\right] P(z | \mathcal{X}; \theta^s) = \operatorname*{argmax}_\theta \sum_{z \in \mathcal{Z}} \log \left[P(\mathcal{X}, z; \theta)\right] P(z, \mathcal{X}; \theta^s) = \operatorname*{argmax}_\theta \hat{Q}(\theta, \theta^s)$$

since $P(\mathcal{X})$ is not affected by choice of $\theta$. Now $P(z, \mathcal{X}; \theta)$ is easy to write down

$$P(z, \mathcal{X}; \theta) = \prod_{d=1}^{D} \left( \pi_{z_1^{(d)}} B_{z_1^{(d)}}(x_1^{(d)}) \prod_{t=2}^{T} A_{z_{t-1}^{(d)} z_t^{(d)}} B_{z_t^{(d)}}(x_t^{(d)}) \right)$$

Taking the log gives us

$$\log P(z, \mathcal{X}; \theta) = \sum_{d=1}^{D} \left[ \log \pi_{z_1^{(d)}} + \sum_{t=2}^{T} \log A_{z_{t-1}^{(d)} z_t^{(d)}} + \sum_{t=1}^{T} \log B_{z_t^{(d)}}(x_t^{(d)}) \right]$$

Plugging this into $\hat{Q}(\theta, \theta^s)$, we get

$$\hat{Q}(\theta, \theta^s) = \sum_{z \in \mathcal{Z}} \sum_{d=1}^{D} \log \pi_{z_1^{(d)}} P(z, \mathcal{X}; \theta^s) + \sum_{z \in \mathcal{Z}} \sum_{d=1}^{D} \sum_{t=2}^{T} \log A_{z_{t-1}^{(d)} z_t^{(d)}} P(z, \mathcal{X}; \theta^s) + \sum_{z \in \mathcal{Z}} \sum_{d=1}^{D} \sum_{t=1}^{T} \log B_{z_t^{(d)}}(x_t^{(d)}) P(z, \mathcal{X}; \theta^s)$$

This is a nice form which we can optimize analytically with Lagrange multipliers. We need Lagrange multipliers because we have equality constraints which come from requiring that $\pi$, $A_{i\cdot}$ and $B_i(\cdot)$ form valid probability distributions. Let $\hat{L}(\theta, \theta^s)$ be the Lagrangian

$$\hat{L}(\theta, \theta^s) = \hat{Q}(\theta, \theta^s) - \lambda_\pi \left( \sum_{i=1}^{M} \pi_i - 1 \right) - \sum_{i=1}^{M} \lambda_{A_i} \left( \sum_{j=1}^{M} A_{ij} - 1 \right) - \sum_{i=1}^{M} \lambda_{B_i} \left( \sum_{j=1}^{N} B_i(j) - 1 \right)$$

First let us focus on the $\pi_i$'s

$$\begin{aligned}
\frac{\partial \hat{L}(\theta, \theta^s)}{\partial \pi_i} &= \frac{\partial}{\partial \pi_i} \left( \sum_{z \in \mathcal{Z}} \sum_{d=1}^{D} \log \pi_{z_1^{(d)}} P(z, X; \theta^s) \right) - \lambda_\pi = 0 \\
&= \frac{\partial}{\partial \pi_i} \left( \sum_{j=1}^{M} \sum_{d=1}^{D} \log \pi_j P(z_1^{(d)} = j, \mathcal{X}; \theta^s) \right) - \lambda_\pi = 0 \\
&= \sum_{d=1}^{D} \frac{P(z_1^{(d)} = i, \mathcal{X}; \theta^s)}{\pi_i} - \lambda_\pi = 0 \\
\frac{\partial \hat{L}(\theta, \theta^s)}{\partial \lambda_\pi} &= -\left( \sum_{i=1}^{M} \pi_i - 1 \right) = 0
\end{aligned}$$

The second step is simply the result of marginalizing out, for each $d$, all $z_{t \neq 1}^{(d)}$ and $z_t^{(d' \neq d)}$ for all $t$. We use this style of trick extensive throughout the remainder of the document. Some algebra yields

$$\begin{aligned}
\pi_i &= \frac{\sum_{d=1}^{D} P(z_1^{(d)} = i, \mathcal{X}; \theta^s)}{\sum_{j=1}^{M} \sum_{d=1}^{D} P(z_1^{(d)} = j, \mathcal{X}; \theta^s)} = \frac{\sum_{d=1}^{D} P(z_1^{(d)} = i, \mathcal{X}; \theta^s)}{\sum_{d=1}^{D} \sum_{j=1}^{M} P(z_1^{(d)} = j, \mathcal{X}; \theta^s)} \\
&= \frac{\sum_{d=1}^{D} P(z_1^{(d)} = i, \mathcal{X}; \theta^s)}{\sum_{d=1}^{D} P(\mathcal{X}; \theta^s)} = \frac{\sum_{d=1}^{D} P(z_1^{(d)} = i, \mathcal{X}; \theta^s)}{D P(\mathcal{X}; \theta^s)} \\
&= \frac{\sum_{d=1}^{D} P(\mathcal{X}; \theta^s) P(z_1^{(d)} = i | \mathcal{X}; \theta^s)}{D P(\mathcal{X}; \theta^s)} = \frac{1}{D} \sum_{d=1}^{D} P(z_1^{(d)} = i | \mathcal{X}; \theta^s) \\
&= \frac{1}{D} \sum_{d=1}^{D} P(z_1^{(d)} = i | X^{(d)}; \theta^s)
\end{aligned}$$

We now follow a similar process for the $A_{ij}$'s.

$$\begin{aligned}
\frac{\partial \hat{L}(\theta, \theta^s)}{\partial A_{ij}} &= \frac{\partial}{\partial A_{ij}} \left( \sum_{z \in \mathcal{Z}} \sum_{d=1}^{D} \sum_{t=2}^{T} \log A_{z_{t-1}^{(d)} z_t^{(d)}} P(z, \mathcal{X}; \theta^s) \right) - \lambda_{A_i} = 0 \\
&= \frac{\partial}{\partial A_{ij}} \left( \sum_{j=1}^{M} \sum_{k=1}^{M} \sum_{d=1}^{D} \sum_{t=2}^{T} \log A_{jk} P(z_{t-1}^{(d)} = j, z_t^{(d)} = k, \mathcal{X}; \theta^s) \right) - \lambda_{A_i} = 0 \\
&= \sum_{d=1}^{D} \sum_{t=2}^{T} \frac{P(z_{t-1}^{(d)} = i, z_t^{(d)} = j, \mathcal{X}; \theta^s)}{A_{ij}} - \lambda_{A_i} = 0 \\
\frac{\partial \hat{L}(\theta, \theta^s)}{\partial \lambda_{A_i}} &= - \left( \sum_{j=1}^{M} A_{ij} - 1 \right) = 0
\end{aligned}$$

This yields

$$\begin{aligned}
A_{ij} &= \frac{\sum_{d=1}^{D} \sum_{t=2}^{T} P(z_{t-1}^{(d)} = i, z_t^{(d)} = j, \mathcal{X}; \theta^s)}{\sum_{j=1}^{M} \sum_{d=1}^{D} \sum_{t=2}^{T} P(z_{t-1}^{(d)} = i, z_t^{(d)} = j, \mathcal{X}; \theta^s)} \\
&= \frac{\sum_{d=1}^{D} \sum_{t=2}^{T} P(z_{t-1}^{(d)} = i, z_t^{(d)} = j, \mathcal{X}; \theta^s)}{\sum_{d=1}^{D} \sum_{t=2}^{T} P(z_{t-1}^{(d)} = i, \mathcal{X}; \theta^s)} \\
&= \frac{\sum_{d=1}^{D} \sum_{t=2}^{T} P(\mathcal{X}; \theta^s) P(z_{t-1}^{(d)} = i, z_t^{(d)} = j | \mathcal{X}; \theta^s)}{\sum_{d=1}^{D} \sum_{t=2}^{T} P(\mathcal{X}; \theta^s) P(z_{t-1}^{(d)} = i | \mathcal{X}; \theta^s)} \\
&= \frac{\sum_{d=1}^{D} \sum_{t=2}^{T} P(z_{t-1}^{(d)} = i, z_t^{(d)} = j | X^{(d)}; \theta^s)}{\sum_{d=1}^{D} \sum_{t=2}^{T} P(z_{t-1}^{(d)} = i | X^{(d)}; \theta^s)}
\end{aligned}$$

The final thing is the $B_i(j)$'s, which are slightly trickier. Let $I(x)$ denote an indicator function which is 1 if $x$ is true, 0 otherwise.

$$\begin{aligned}
\frac{\partial \hat{L}(\theta, \theta^s)}{\partial B_i(j)} &= \frac{\partial}{\partial B_i(j)} \left( \sum_{z \in \mathcal{Z}} \sum_{d=1}^{D} \sum_{t=1}^{T} \log B_{z_t^{(d)}}(x_t^{(d)}) P(z, \mathcal{X}; \theta^s) \right) - \lambda_{B_i} = 0 \\
&= \frac{\partial}{\partial B_i(j)} \left( \sum_{i=1}^{N} \sum_{d=1}^{D} \sum_{t=1}^{T} \log B_i(x_t^{(d)}) P(z_t^{(d)} = i, \mathcal{X}; \theta^s) \right) - \lambda_{B_i} = 0 \\
&= \sum_{d=1}^{D} \sum_{t=1}^{T} \frac{P(z_t^{(d)} = i, \mathcal{X}; \theta^s) I(x_t^{(d)} = j)}{B_i(j)} - \lambda_{B_i} = 0 \\
\frac{\partial \hat{L}(\theta, \theta^s)}{\partial \lambda_{B_i}} &= - \left( \sum_{j=1}^{N} B_i(j) - 1 \right) = 0
\end{aligned}$$

This should come as no surprise by now

$$
\begin{aligned}
B_i(j) &= \frac{\sum\limits_{d=1}^{D}\sum\limits_{t=1}^{T} P(z_t^{(d)} = i, \mathcal{X}; \theta^s) I(x_t^{(d)} = j)}{\sum\limits_{j=1}^{N}\sum\limits_{d=1}^{D}\sum\limits_{t=1}^{T} P(z_t^{(d)} = i, \mathcal{X}; \theta^s) I(x_t^{(d)} = j)} \\
&= \frac{\sum\limits_{d=1}^{D}\sum\limits_{t=1}^{T} P(z_t^{(d)} = i, \mathcal{X}; \theta^s) I(x_t^{(d)} = j)}{\sum\limits_{d=1}^{D}\sum\limits_{t=1}^{T} P(z_t^{(d)} = i, \mathcal{X}; \theta^s)} \\
&= \frac{\sum\limits_{d=1}^{D}\sum\limits_{t=1}^{T} P(z_t^{(d)} = i | X^{(d)}; \theta^s) I(x_t^{(d)} = j)}{\sum\limits_{d=1}^{D}\sum\limits_{t=1}^{T} P(z_t^{(d)} = i | X^{(d)}; \theta^s)}
\end{aligned}
$$

To summarize, the update steps are

$$
\begin{aligned}
\pi_i^{(s+1)} &= \frac{1}{D}\sum_{d=1}^{D} P(z_1^{(d)} = i | X^{(d)}; \theta^s) \\
A_{ij}^{(s+1)} &= \frac{\sum\limits_{d=1}^{D}\sum\limits_{t=2}^{T} P(z_{t-1}^{(d)} = i, z_t^{(d)} = j | X^{(d)}; \theta^s)}{\sum\limits_{d=1}^{D}\sum\limits_{t=2}^{T} P(z_{t-1}^{(d)} = i | X^{(d)}; \theta^s)} \\
B_i^{(s+1)}(j) &= \frac{\sum\limits_{d=1}^{D}\sum\limits_{t=1}^{T} P(z_t^{(d)} = i | X^{(d)}; \theta^s) I(x_t^{(d)} = j)}{\sum\limits_{d=1}^{D}\sum\limits_{t=1}^{T} P(z_t^{(d)} = i | X^{(d)}; \theta^s)}
\end{aligned}
$$

Note that $P(z_t | X; \theta)$ and $P(z_{t-1}, z_t | X; \theta)$ are both quantities which can be computed efficiently for HMMs by the forward-backwards algorithm. Once again, see [1] for more details.

## References


[1] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. 2006.