Forward-Backward

\[
F_k(i) = P(X_1, ..., X_i, s_i = k)
\]
\[
B_k(i) = P(X_{i+1}, ..., X_n | s_i = k)
\]

If we will look at \( F_k(i) \cdot B_k(i) = P(\vec{X}, s_i = k) \) \( \overset{\text{posterior on a state}}{= \overbrace{P(s_i = k | \vec{X})} \cdot P(\vec{X})} \)

and to find the likelihood of the sequence, \( P(\vec{X}) = \sum_k P(X_1, ..., X_n, s_n = k) = \sum_k F_k(n) \)

So, we can compute the posterior of being in a certain state \( k \) at time \( i \) if we have both forward and backward calculations.

Use of HMM - Motif Finding

We want to find \( \theta \) from the data - use HMM to find motifs.

For example, promoters hold motifs that bind transcription factors. A well-known transcription factor motif is TATA binding TBP, which in turn recruits the RNA polymerase (see figure 1).

Let’s find a known Transcription Factor motif: TATA that binds TBP (TATA Binding protein) which is a sub-unit of the transcription complex. The motif can be described by a PWM (Position Weight Matrix) as depicted below in Table 1.

We are also given the background, or non-motif, emission probability: \( B = \begin{bmatrix} A & C & G & T \end{bmatrix} \)

We want to suggest an HMM model for the motif-finding problem. First, the states will represent the position of the letter in the motif (\( T_1 \) for the first letter…), the BG states represent the non-motif part of the sequence. We look at the PWM columns as the emission values, now we need to find the transition probabilities.

We will see here 3 gHMM (generalized HMM), which provides the framework for describing the grammar of a legal parse of a DNA sequence (Haussler et al.)

1. OOPS (One Occurrence Per Sequence)
   (see figure 2) In this model, we have an assumption that the motif must appear exactly one time in the sequence. See the \( \tau \) (transition matrix) of this model at table 2.

2. ZOOPS (Zero or One Occurrences Per Sequence)
   this will be similar to OOPS, but the starting BG state will also accept.
Table 1: PWM for the sequence TATAAT

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>.89</td>
<td></td>
<td></td>
<td>.59</td>
<td>.49</td>
<td></td>
</tr>
<tr>
<td>C</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>G</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>T</td>
<td>.82</td>
<td>.52</td>
<td>.89</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 2: OOPS model scheme

Table 2: OOPS model transition matrix

<table>
<thead>
<tr>
<th></th>
<th>BG$_1$</th>
<th>T$_1$</th>
<th>T$_2$</th>
<th>T$_3$</th>
<th>...</th>
<th>T$_6$</th>
<th>BG$_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>BG$_1$</td>
<td>1-r</td>
<td>r</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>T$_1$</td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>T$_2$</td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td>...</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>T$_5$</td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>T$_6$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BG$_2$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>

Figure 3: TCM model scheme
Bayesian Parameter Estimation

**Frequentists**
- **Max Estimator:** \( \hat{\theta}_{MLE} = \arg\max_{\theta} L(\theta; D) \)
- **Coin Tossing Example:**
  Likelihood: \( L(\theta) = \theta^N_T \cdot (1 - \theta)^{N_H} \) thus the appropriate estimator is \( \hat{\theta}_{MLE} = \frac{N_H}{N_H + N_T} \)
- **Posterior Predictive Distribution:**
  Estimation of the probabilities based on the viewed dataset: \( P(x = \{H\}|D) \)

**Bayesians**
- **Max Estimator:**
  \[
  \hat{\theta}_{MAP} = \arg\max_{\theta} \frac{\pi(\theta) L(\theta; D)}{\pi(\theta)} = \arg\max_{\theta} \frac{L(\theta; D)}{L(\theta; D)}
  \]
- **Coin Tossing Example:**
  Here we use the \( \beta \) distribution as prior, as seen in scribe \#8.
  \[
  \hat{\theta}_{MAP} = \arg\max_{\theta} \theta^N_H (1 - \theta)^{N_T} \cdot \theta^{\alpha_H - 1} (1 - \theta)^{\alpha_T - 1} = \frac{N_H + \alpha_H - 1}{N_H + N_T + \alpha_H + \alpha_T - 2}
  \]
  We saw last week that with hyperparameters \( \alpha_H = 1, \alpha_T = 1 \) we get \( \hat{\theta}_{MAP} = \frac{N_H}{N_H + N_T} = \hat{\theta}_{MLE} \)
- **Prior Predictive Distribution:**
  Before seeing any data
  \[
  P(x = \{H\}) = \int_0^1 P(H|\theta)P(\theta)d\theta = \int_0^1 \theta P(\theta)d\theta = E[\theta]
  \]

\[
E[\theta] = \frac{\alpha_H}{\alpha_H + \alpha_T}
\]

The calculation:
Denote \( \theta \sim \text{Beta}(\alpha_H, \alpha_T) \) this means \( P(\theta|\alpha_H, \alpha_T) = \frac{\theta^{\alpha_H-1}(1-\theta)^{\alpha_T-1}}{Z(\alpha_H, \alpha_T)} \) with the corresponding normalization factor \( Z(\alpha_H, \alpha_T) = \frac{\Gamma(\alpha_H+1)\Gamma(\alpha_T)}{\Gamma(\alpha_H+\alpha_T)} \) (using the \( \Gamma \) function defined in scribe \#8)

\[
E[\theta] = \int_0^1 \frac{\theta^{\alpha_H-1}(1-\theta)^{\alpha_T-1}}{Z(\alpha_H, \alpha_T)} d\theta = \frac{Z(\alpha_H+1, \alpha_T)}{Z(\alpha_H, \alpha_T)} \int_0^1 \frac{\theta^{\alpha_H}(1-\theta)^{\alpha_T-1}}{Z(\alpha_H+1, \alpha_T)} d\theta =
\]

\[
E[\theta] = \frac{Z(\alpha_H+1, \alpha_T)}{Z(\alpha_H, \alpha_T)} \frac{\Gamma(\alpha_H+1)\Gamma(\alpha_T+1)}{\Gamma(\alpha_H+\alpha_T+1)}
\]

\[
\approx \frac{\alpha_H!}{(\alpha_H-1)!\alpha_T!} \left( \frac{\alpha_H}{\alpha_H+\alpha_T} \right) = \frac{\alpha_H}{\alpha_H+\alpha_T}
\]
• **Posterior Predictive Distribution:**

\[
P(x = \{H\}|D) = \int_0^1 P(H|\theta) \cdot P(\theta|D) d\theta = \int_0^1 \theta P(\theta|D) d\theta =
\]

\[
E[\theta|D] = \frac{N_H + \alpha_H}{N + \alpha}
\]

Where \(N\) is the total number of coin tosses, and \(\alpha\) is the pseudocount, an imaginary set of samples taken before seeing the data.

The calculation:

\[
E[\theta|D] = \int_0^1 \theta P(\theta|D) d\theta = \int_0^1 \theta \frac{\theta^{N_H + \alpha_H - 1} \cdot (1 - \theta)^{N_T + \alpha_T - 1}}{Z(N_H + \alpha_H, N_T + \alpha_T)} d\theta = \frac{N_H + \alpha_H}{N + \alpha}
\]

With * being the same trick as in the prior calculation.

**The EM (Expectation Maximization)\ Baum-Welch Algorithm**

Note this is explained in more detail in scribe #12.

We have in HMM the emission \(e_{kx}\) (the probability of state \(k\) to emit \(x\) from the alphabet ), and transition \(\tau_{kl}\) (the probability the transition from state \(k\) to \(l\) )

Given the sequence of hidden states we can estimate the parameters, \(e, \tau\) and given \(e, \tau\) we can estimate the most probable hidden states sequence (Viterbi) and the sufficient statistics for the parameters. In this algorithm we iteratively switch between Expectation (evaluating the hidden parameters or the sufficient statistics for a lower bound of the parameters) and Maximization (maximizing the likelihood with the parameters):

1. Initiate parameters \(\theta\)
2. Iterate:
   (a) evaluation of the hidden variables
   (b) maximize a lower bound of the likelihood for \(\theta\)
3. conditioned halt