38. Note that in this problem our densities need not be normal.

(a) Here we have the criterion function

\[ J_1(w) = \frac{(\mu_1 - \mu_2)^2}{\sigma_1^2 + \sigma_2^2}. \]

We make use of the following facts for \( i = 1, 2 \):

\[
\begin{align*}
y &= w^t x \\
\mu_i &= \frac{1}{n_i} \sum_{y \in Y_i} y = \frac{1}{n_i} \sum_{x \in D_i} w^t x = w^t \mu_i \\
\sigma_i^2 &= \sum_{y \in Y_i} (y - \mu_i)^2 = w^t \left[ \sum_{x \in D_i} (x - \mu_i)(x - \mu_i)^t \right] w \\
\Sigma_i &= \sum_{x \in D_i} (x - \mu_i)(x - \mu_i)^t.
\end{align*}
\]

We define the within- and between-scatter matrices to be

\[
\begin{align*}
S_W &= \Sigma_1 + \Sigma_2 \\
S_B &= (\mu_1 - \mu_2)(\mu_1 - \mu_2)^t.
\end{align*}
\]

Then we can write

\[
\begin{align*}
\sigma_1^2 + \sigma_2^2 &= w^t S_W w \\
(\mu_1 - \mu_2)^2 &= w^t S_B w.
\end{align*}
\]

The criterion function can be written as

\[ J_1(w) = \frac{w^t S_B w}{w^t S_W w}. \]

For the same reason Eq. 103 in the text is maximized, we have that \( J_1(w) \) is maximized at \( w = \mathbf{w}_{S_W}^{-1}(\mu_1 - \mu_2) \). In sum, that \( J_1(w) \) is maximized at \( w = (\Sigma_1 + \Sigma_2)^{-1}(\mu_1 - \mu_2) \).

(b) Consider the criterion function

\[ J_2(w) = \frac{(\mu_1 - \mu_2)^2}{P(\omega_1)\sigma_1^2 + P(\omega_2)\sigma_2^2}. \]

Except for letting \( S_W = \hat{P}(\omega_1)\Sigma_1 + \hat{P}(\omega_2)\Sigma_2 \), we retain all the notations in part (a). Then we write the criterion function as a Rayleigh quotient

\[ J_2(w) = \frac{w^t S_B w}{w^t S_W w}. \]

For the same reason Eq. 103 is maximized, we have that \( J_2(w) \) is maximized at

\[ w = (\hat{P}(\omega_1)\Sigma_1 + \hat{P}(\omega_2)\Sigma_2)^{-1}(\mu_1 - \mu_2). \]

(c) Equation 96 of the text is more closely related to the criterion function in part (a) above. In Eq. 96 in the text, we let \( \bar{m}_i = \mu_i \), and \( \bar{s}_i^2 = \sigma_i^2 \) and the statistical meanings are unchanged. Then we see the exact correspondence between \( J(w) \) and \( J_1(w) \).
2. Problem 50 (page 154)

50. The standard method for calculating the probability of a sequence in a given HMM is to use the forward probabilities $\alpha_i(t)$.

(a) In the forward algorithm, for $t = 0, 1, \ldots, T$, we have

$$
\alpha_j(t) = \begin{cases} 
0 & t = 0 \text{ and } j \neq \text{initial status} \\
1 & t = 0 \text{ and } j = \text{initial status} \\
\sum_{i=1}^{c} \alpha_i(t-1)a_{ij}b_{jk}v(t) & \text{otherwise}.
\end{cases}
$$

In the backward algorithm, we use for $t = T, T-1, \ldots, 0$,

$$
\beta_j(t) = \begin{cases} 
0 & t = T \text{ and } j \neq \text{final status} \\
1 & t = T \text{ and } j = \text{final status} \\
\sum_{i=1}^{c} \beta_i(t+1)a_{ij}b_{jk}v(t+1) & \text{otherwise}.
\end{cases}
$$

Thus in the forward algorithm, if we first reverse the observed sequence $V^T$ (that is, set $b_{jk}v(t) = b_{jk}(T+1-t)$ and then set $\beta_j(t) = \alpha_j(T-t)$, we can obtain the backward algorithm.

(b) Consider splitting the sequence $V^T$ into two parts — $V_1$ and $V_2$ — before, during, and after each time step $T'$ where $T' < T$. We know that $\alpha_i(T')$ represents the probability that the HMM is in hidden state $\omega_i$ at step $T'$, having generated the first $T'$ elements of $V^T$, that is $V_1$. Likewise, $\beta_i(T')$ represents the probability that the HMM given that it is in $\omega_i$ at step $T'$ generates the remaining elements of $V^T$, that is, $V_2$. Hence, for the complete sequence we have

$$
P(V^T) = P(V_1, V_2) = \sum_{i=1}^{c} P(V_1, V_2, \text{hidden state } \omega_i \text{ at step } T')
$$

$$
= \sum_{i=1}^{c} P(V_1, \text{hidden state } \omega_i \text{ at step } T')P(V_2|\text{hidden state } \omega_i \text{ at step } T')
$$

$$
= \sum_{i=1}^{c} \alpha_i(T')\beta_i(T').
$$

(c) At $T' = 0$, the above reduces to $P(V^T) = \sum_{i=1}^{c} \alpha_i(0)\beta_i(0) = \beta_j(0)$, where $j$ is the known initial state. This is the same as line 5 in Algorithm 3. Likewise, at $T' = T$, the above reduces to $P(V^T) = \sum_{i=1}^{c} \alpha_i(T)\beta_i(T) = \alpha_j(T)$, where $j$ is the known final state. This is the same as line 5 in Algorithm 2.
3. Computer Exercise 13 (page 159)

In this problem we want to train two Hidden Markov Models for a sequence of characters. We models on the given data samples, and the goal is to classify the test sequences as belonging to the models. First we initialize the transition matrices for each of the models randomly. Then HMM toolbox in matlab to train the transition matrices.

The results are shown below:

Part A:

Initial Transition Matrix for both model:

\[
\begin{align*}
\text{ans} &= \\
0.4828 & 0.3238 & 0.0244 & 0.1690 \\
0.2160 & 0.3052 & 0.2000 & 0.2789 \\
0.2088 & 0.1551 & 0.3125 & 0.3236 \\
0.1003 & 0.1675 & 0.0321 & 0.7000
\end{align*}
\]

Trained Transition Matrix for w1:

\[
\begin{align*}
\text{ans} &= \\
0.1891 & 0.3989 & 0.0746 & 0.3375 \\
0.1558 & 0.3322 & 0.1691 & 0.3429 \\
0.0847 & 0.0618 & 0.1471 & 0.7063 \\
0.0076 & 0.0211 & 0.0148 & 0.9585
\end{align*}
\]

Trained Transition Matrix for w2:

\[
\begin{align*}
\text{ans} &= \\
0.5058 & 0.2549 & 0.0180 & 0.2213 \\
0.0496 & 0.9034 & 0.0326 & 0.0145 \\
0.3892 & 0.0108 & 0.4571 & 0.1429 \\
0.0494 & 0.5549 & 0.0060 & 0.3898
\end{align*}
\]

Part B: Sequence Classification

Sequence ABBBCDDD belongs to w1
Sequence DADBCBAA belongs to w2
Sequence CDCBABA belongs to w2
Sequence ADEBBBCD belongs to w1

Part C: Prior Probability

\[
P_w1 = \\
0.3354
\]

\[
P_w2 = \\
0.6646
\]