Note:

1. The problems are from the textbook by G. H. Givens and J. A. Hoeting
2. There are total 4 questions:3.1, 3.8, 4.1, and 4.5.

The baseball data introduced in Section 3.3 are available from the website for this book. Problems 3.1–3.4 explore the implications of various algorithm configurations. Treat these problems in the spirit of experiments, trying to identify settings where interesting differences can be observed. Increase the run lengths from those used above to suit the speed of your computer, and limit the total number of objective function evaluations in every run (effectively the search effort) to a fixed number so that different algorithms and configurations can be compared fairly. Summarize your comparisons and conclusions. Supplement your comments with graphs to illustrate key points.

3.1. Implement a random starts local search algorithm for minimizing the AIC for the baseball salary regression problem. Model your algorithm after Example 3.3.

a. Change the move strategy from steepest descent to immediate adoption of the first randomly selected downhill neighbor.

b. Change the algorithm to employ 2-neighborhoods, and compare the results with those of previous runs.

3.8. Thirteen chemical measurements were carried out on each of 178 wines from three regions of Italy [53]. These data are available from the website for this book. Using one or more heuristic search methods from this chapter, partition the wines into three groups for which the total of the within-group sum of squares is minimal. Comment on your work and the results. This is a search problem of size 3^p where $p = 178$. If you have access to standard cluster analysis routines, check your results using a standard method like that of Hartigan and Wong [317].

4.1. Recall the peppered moth analysis introduced in Example 4.2. In the field, it is quite difficult to distinguish the *insularia* and *typica* phenotypes due to variations in wing color and mottle. In addition to the 622 moths mentioned in the example, suppose the sample collected by the researchers actually included $n_U = 578$ more moths that were known to be *insularia* or *typical* but whose exact phenotypes could not be determined.

a. Derive the EM algorithm for maximum likelihood estimation of p_C, p_I, and p_I for this modified problem having observed data n_C, n_I, n_T, and n_U as given above.

b. Apply the algorithm to find the MLEs.

c. Estimate the standard errors and pairwise correlations for $\hat{p}_C$, $\hat{p}_I$, and $\hat{p}_I$ using the SEM algorithm.

d. Estimate the standard errors and pairwise correlations for $\hat{p}_C$, $\hat{p}_I$, and $\hat{p}_I$ by bootstrapping.

e. Implement the EM gradient algorithm for these data. Experiment with step halving to ensure ascent and with other step scalings that may speed convergence.

f. Implement Aitken accelerated EM for these data. Use step halving.

g. Implement quasi-Newton EM for these data. Compare performance with and without step halving.

h. Compare the effectiveness and efficiency of the standard EM algorithm and the three variants in (e), (f), and (g). Use step halving to ensure ascent with the three variants. Base your comparison on a variety of starting points. Create a graph analogous to Figure 4.3.

4.5. A *hidden Markov model (HMM)* can be used to describe the joint probability of a sequence of unobserved (hidden) discrete-state variables, $\mathbf{H} = (H_0, \ldots, H_n)$, and a sequence of corresponding observed variables $\mathbf{O} = (O_0, \ldots, O_n)$ for which O_i is dependent on H_i for each i. We say that H_i emits O_i; consideration here is limited to discrete emission variables. Let the state spaces for elements of $\mathbf{H}$ and $\mathbf{O}$ be $\mathcal{H}$ and $\mathcal{E}$, respectively.

Let $\mathbf{O}_{\leq j}$ and $\mathbf{O}_{>j}$ denote the portions of $\mathbf{O}$ with indices not exceeding j and exceeding j, respectively, and define the analogous partial sequences for $\mathbf{H}$. Under an HMM, the H_i have the Markov property

$$P[H_i|\mathbf{H}_{\leq i-1}, O_0] = P[H_i|H_{i-1}] \tag{4.90}$$

and the emissions are conditionally independent, so

$$P[O_i|\mathbf{H}, \mathbf{O}_{\leq i-1}, \mathbf{O}_{>i}] = P[O_i|H_i]. \tag{4.91}$$

Time-homogeneous transitions of the hidden states are governed by transition probabilities $p(h, h^*) = P[H_{i+1} = h^*|H_i = h]$ for $h, h^* \in \mathcal{H}$. The distribution for H_0 is parameterized by $\pi(h) = P[H_0 = h]$ for $h \in \mathcal{H}$. Finally, define emission probabilities $e(h, o) = P[O_i = o|H_i = h]$ for $h \in \mathcal{H}$ and $o \in \mathcal{E}$. Then the parameter set $\boldsymbol{\theta} = (\boldsymbol{\pi}, \mathbf{P}, \mathbf{E})$ completely parameterizes the model, where $\boldsymbol{\pi}$ is a vector of initial-state probabilities, $\mathbf{P}$ is a matrix of transition probabilities, and $\mathbf{E}$ is a matrix of emission probabilities.

For an observed sequence $\mathbf{o}$, define the *forward variables* to be

$$\alpha(i, h) = P[\mathbf{O}_{\leq i} = \mathbf{o}_{\leq i},\ H_i = h] \tag{4.92}$$

and the *backward variables* to be

$$\beta(i, h) = P[\mathbf{O}_{>i} = \mathbf{o}_{>i}|H_i = h] \tag{4.93}$$

for $i = 1, \ldots, n$ and each $h \in \mathcal{H}$. Our notation suppresses the dependence of the forward and backward variables on $\boldsymbol{\theta}$. Note that

$$P[\mathbf{O} = \mathbf{o}|\boldsymbol{\theta}] = \sum_{h\in\mathcal{H}} \alpha(n, h) = \sum_{h\in\mathcal{H}} \pi(h)e(h, o_0)\beta(0, h). \tag{4.94}$$

The forward and backward variables are also useful for computing the probability that state h occurred at the ith position of the sequence given $\mathbf{O} = \mathbf{o}$ according to $P[H_i = h|\mathbf{O} = \mathbf{o}, \boldsymbol{\theta}] = \sum_{h\in H} \alpha(i, h)\beta(i, h)/P[\mathbf{O} = \mathbf{o}|\boldsymbol{\theta}]$, and expectations of functions of the states with respect to these probabilities.

a. Show that the following algorithms can be used to calculate $\alpha(i, h)$ and $\beta(i, h)$. The *forward algorithm* is

- Initialize $\alpha(0, h) = \pi(h)e(h, o_0)$.
- For $i = 0, \ldots, n-1$, let $\alpha(i+1, h) = \sum_{h^*\in\mathcal{H}} \alpha(i, h^*)p(h^*, h)e(h, o_{i+1})$.

The *backward algorithm* is

- Initialize $\beta(n, h) = 1$.
- For $i = n, \ldots, 1$, let $\beta(i-1, h) = \sum_{h^*\in\mathcal{H}} p(h, h^*)e(h^*, o_i)\beta(h, i)$.

These algorithms provide very efficient methods for finding $P[\mathbf{O} = \mathbf{o}|\boldsymbol{\theta}]$ and other useful probabilities, compared to naively summing over all possible sequences of states.

b. Let $N(h)$ denote the number of times $H_0 = h$, let $N(h, h^*)$ denote the number of transitions from h to h^*, and let $N(h, o)$ denote the number of emissions of o when the underlying state is h. Prove that these random variables have the following expectations:

$$E\{N(h)\} = \frac{\alpha(0, h)\beta(0, h)}{P[\mathbf{O} = \mathbf{o}|\boldsymbol{\theta}]}, \tag{4.95}$$

$$E\{N(h, h^*)\} = \sum_{i=0}^{n-1} \frac{\alpha(i, h)p(h, h^*)e(h^*, o_{i+1})\beta(i+1, h^*)}{P[\mathbf{O} = \mathbf{o}|\boldsymbol{\theta}]}, \tag{4.96}$$

$$E\{N(h, o)\} = \sum_{i:\ O_i=o} \frac{\alpha(i, h)\beta(i, h)}{P[\mathbf{O} = \mathbf{o}|\boldsymbol{\theta}]}. \tag{4.97}$$

c. The *Baum–Welch algorithm* efficiently estimates the parameters of an HMM [25]. Fitting these models has proven extremely useful in diverse applications including statistical genetics, signal processing and speech recognition, problems involving environmental time series, and Bayesian graphical networks [172, 236, 361, 392, 523]. Starting from some initial values $\boldsymbol{\theta}^{(0)}$, the Baum–Welch algorithm proceeds via iterative application of the following update formulas:

$$\pi(h)^{(t+1)} = \frac{E\{N(h)|\boldsymbol{\theta}^{(t)}\}}{\sum_{h^* \in \mathcal{H}} E\{N(h^*)|\boldsymbol{\theta}^{(t)}\}}, \tag{4.98}$$

$$p(h,h^*)^{(t+1)} = \frac{E\{N(h,h^*)|\boldsymbol{\theta}^{(t)}\}}{\sum_{h^{**} \in \mathcal{H}} E\{N(h,h^{**})|\boldsymbol{\theta}^{(t)}\}}, \tag{4.99}$$

$$e(h,o)^{(t+1)} = \frac{E\{N(h,o)|\boldsymbol{\theta}^{(t)}\}}{\sum_{o^* \in \mathcal{E}} E\{N(h,o^*)|\boldsymbol{\theta}^{(t)}\}}. \tag{4.100}$$

Prove that the Baum–Welch algorithm is an EM algorithm. It is useful to begin by noting that the complete data likelihood is given by

$$\prod_{h \in \mathcal{H}} \pi(h)^{N(h)} \prod_{h \in \mathcal{H}} \prod_{o \in \mathcal{E}} e(h,o)^{N(h,o)} \prod_{h \in \mathcal{H}} \prod_{h^* \in \mathcal{H}} p(h,h^*)^{N(h,h^*)}. \tag{4.101}$$

d. Consider the following scenario. In Flip's left pocket is a penny; in his right pocket is a dime. On a fair toss, the probability of showing a head is p for the penny and d for the dime. Flip randomly chooses a coin to begin, tosses it, and reports the outcome (heads or tails) without revealing which coin was tossed. Then, Flip decides whether to use the same coin for the next toss, or to switch to the other coin. He switches coins with probability s, and retains the same coin with probability $1 - s$. The outcome of the second toss is reported, again not revealing the coin used. This process is continued for a total of 200 coin tosses. The resulting sequence of heads and tails is available from the website for this book. Use the Baum–Welch algorithm to estimate p, d, and s.

e. Only for students seeking extra challenge: Derive the Baum–Welch algorithm for the case when the dataset consists of M independent observation sequences arising from a HMM. Simulate such data, following the coin example above. (You may wish to mimic the single-sequence data, which were simulated using $p = 0.25$, $d = 0.85$, and $s = 0.1$.) Code the Baum–Welch algorithm, and test it on your simulated data.

In addition to considering multiple sequences, HMMs and the Baum–Welch algorithm can be generalized for estimation based on more general emission variables and emission and transition probabilities that have more complex parameterizations, including time inhomogeneity.