This project is a combination of the following programming exercises, newly added to Chapter 2. You may have to experiment with some of the parameters to get good results.

**Exercise 2.4 (programming)**  Design and conduct an experiment to demonstrate the difficulties that sample-average methods have for nonstationary problems. Use a modified version of the 10-armed testbed in which all the $q_*(a)$ start out equal and then take independent random walks (say by adding a normally distributed increment with mean zero and standard deviation 0.01 to all the $q_*(a)$ on each step). Prepare plots like Figure 2.2 for an action-value method using sample averages, incrementally computed, and another action-value method using a constant step-size parameter, $\alpha = 0.1$. Use $\varepsilon = 0.1$ and longer runs, say of 10,000 steps.

**Exercise 2.7 (programming)**  Make a figure analogous to Figure 2.6 for the nonstationary case outlined in Exercise 2.4. Create a single long sequence of true action values $q_t^*(a)$ over 200,000 steps. As a performance measure for each algorithm (and parameter setting) use its average reward over the second 100,000 steps.