```
double x = 0.5:
      for(int i = 0;i<ntransient;i++) { // x values not plotted
         x = map(x, r);
      // plot half the points in dataset zero
      for(int i = 0; i < nplot/2; i++) {
         x = map(x, r):
         // shows different x values for given value of r
         plotFrame.append(0, r, x);
      // plot remaining points in dataset one
      for (int i = nplot/2+1; i < nplot; i++) {
         x = map(x, r):
         // dataset one has a different color
         plotFrame.append(1, r, x);
         j++;
      r += dr;
public void reset() {
   control.setValue("initial r", 0.2);
   control.setValue("dr", 0.005);
   control.setValue("ntransient", 200);
   control.setValue("nplot", 50):
double map(double x, double r) {
   return 4*r*x*(1-x):
public static void main(String[] args) {
   SimulationControl.createApp(new BifurcateApp());
```

Problem 6.2 Qualitative features of the logistic map

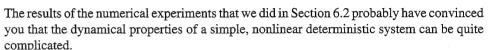
- (a) Use BifurcateApp to identify period 2, period 4, and period 8 behavior as can be seen in Figure 6.2. Choose ntransient ≥ 1000. It might be necessary to "zoom in" on a portion of the plot. How many period doublings can you find?
- (b) Change the scale so that you can follow the iterations of x from period 4 to period 16 behavior. How does the plot look on this scale in comparison to the original scale?
- (c) Describe the shape of the trajectory near the bifurcations from period 2 to period 4, period 4 to period 8, etc. These bifurcations are frequently called *pitchfork bifurcations*.

The bifurcation diagram in Figure 6.2 indicates that the period doubling behavior ends at $r \approx 0.892$. This value of r is known very precisely and is given by $r = r_{\infty} = 0.892486417967...$ At $r = r_{\infty}$, the sequence of period doublings accumulates to a trajectory of infinite period. In Problem 6.3 we explore the behavior of the trajectories for $r > r_{\infty}$.

Problem 6.3 Chaotic behavior

- (a) For $r > r_{\infty}$, two initial conditions that are very close to one another can yield very different trajectories after a few iterations. As an example, choose r = 0.91 and consider $x_0 = 0.5$ and 0.5001. How many iterations are necessary for the iterated values of x to differ by more than ten percent? What happens for r = 0.88 for the same choice of seeds?
- (b) The accuracy of floating point numbers retained on a digital computer is finite. To test the effect of the finite accuracy of your computer, choose r=0.91 and $x_0=0.5$ and compute the trajectory for 200 iterations. Then modify your program so that after each iteration, the operation x=x/10 is followed by x=10*x. This combination of operations truncates the last digit that your computer retains. Compute the trajectory again and compare your results. Do you find the same discrepancy for $r < r_{\infty}$?
- (c) What are the dynamical properties for r = 0.958? Can you find other windows of periodic behavior in the interval $r_{\infty} < r < 1$?

6.3 ■ PERIOD DOUBLING



To gain more insight into how the dynamical behavior depends on r, we introduce a simple graphical method for iterating (6.5). In Figure 6.3 we show a graph of f(x) versus x for r=0.7. A diagonal line corresponding to y=x intersects the curve y=f(x) at the two fixed points $x^*=0$ and $x^*=9/14\approx 0.642857$ (see (6.6b)). If x_0 is not a fixed point, we can find the trajectory in the following way. Draw a vertical line from $(x=x_0,y=0)$ to the intersection with the curve y=f(x) at $(x_0,y_0=f(x_0))$. Next draw a horizontal line from (x_0,y_0) to the intersection with the diagonal line at (y_0,y_0) . On this diagonal line y=x, and hence the value of x at this intersection is the first iteration $x_1=y_0$. The second iteration x_2 can be found in the same way. From the point (x_1,y_0) , draw a vertical line to the intersection with the curve y=f(x). Keep y fixed at $y=y_1=f(x_1)$, and draw a horizontal line until it intersects the diagonal line; the value of x at this intersection is x_2 . Further iterations can be found by repeating this process.

This graphical method is illustrated in Figure 6.3 for r = 0.7 and $x_0 = 0.9$. If we begin with any x_0 (except $x_0 = 0$ and $x_0 = 1$), the iterations will converge to the fixed point $x^* \approx 0.643$. It would be a good idea to repeat the procedure shown in Figure 6.3 by hand. For r = 0.7, the fixed point is stable (an attractor of period 1). In contrast, no matter how close x_0 is to the fixed point at x = 0, the iterates diverge away from it, and this fixed point is unstable.

How can we explain the qualitative difference between the fixed point at x=0 and at $x^*=0.642857$ for r=0.7? The local slope of the curve y=f(x) determines the distance moved horizontally each time f is iterated. A slope steeper than 45° leads to a value of x further away from its initial value. Hence, the criterion for the stability of a fixed point is that the magnitude of the slope at the fixed point must be less than 45°. That is, if $|df(x)/dx|_{x=x^*}$ is less than unity, then x^* is stable; conversely, if $|df(x)/dx|_{x=x^*}$ is greater than unity, then x^* is unstable.