```
In[65]:= CobwebPlot[f_, {x_, a_, b_}, x0_, n_Integer] :=
          CobwebPlot[f, {x, a, b}, x0, {0, n}]

In[66]:= CobwebPlot[f_, {x_, a_, b_}, x0_, {n0_, n_}] :=
          Module[{data, fcn},
           fcn = Compile[x, f];
           data = NestList[fcn, s = Nest[fcn, x0, n0], n - 1];
           y0 = If[n0 == 0, 0, s];
           Show[Block[{$DisplayFunction = Identity},
             {Plot[f, {x, a, b}],
              Graphics[{{Thickness[0.0005], Line[Drop[Prepend[
                    Flatten[
                     data /. z_?NumericQ :> {{z, ff = fcn[z]}, {ff, ff}}, 1], {s, y0}], -1]]},
                {PointSize[0.025], Point[{s, y0}]}, {GrayLevel[0.4], Line[{{a, a}, {b, b}}]}}]}],
            PlotRange → {{a, b}, All}, AxesLabel → {"N(t)", "N(t+1)"}]]

In[67]:= Needs["Utilities`FilterOptions`"];

In[68]:= elimDups[v_] :=
           Module[{d = Sort[v]}, Union[FoldList[If[#2 < 0.001 + #1, #1, #2] &, First[d], Rest[d]]]];

In[69]:= Options[BifurcationPlot] = {PlotPoints → 100};

In[70]:= BifurcationPlot[f_, {r_, a_, b_}, {x_, x0_}, {iter0_, iterShow_}, opts___] :=
           Module[{n}, n = PlotPoints /. {opts} /. Options[BifurcationPlot];
            makePts[{s_, v_}] := Map[Point[{s, #}] &, v];
            cf = Compile[{{r, _Real, 1}, {x, _Real, 1}}, Evaluate[f]];
            rVals = Min[#, b] & /@ Range[N[a], b, N[(b - a) / (n - 1)]];
            data =
             Transpose[
              {rVals,
               elimDups /@ Transpose[NestList[cf[rVals, #] &,
                  Nest[cf[rVals, #] &, Array[x0 &, n], iter0], iterShow]]}];
            Show[Graphics[{AbsolutePointSize[0.4], makePts /@ data}],
             FilterOptions[Graphics, opts], Frame → True, PlotRange → {{a, b}, All}]];
```

"Cell[TextData[{Cell[TextData[{ValueBox["FileName"]}], "Header"], Cell[" ", "Header", CellFrame -> {{0, 0.5}, {0, 0}}, CellFrameMargins -> 4], " ", Cell[TextData[{CounterBox["Page"]}], "PageNumber"]}], CellMargins -> {{Inherited, 0}, {Inher

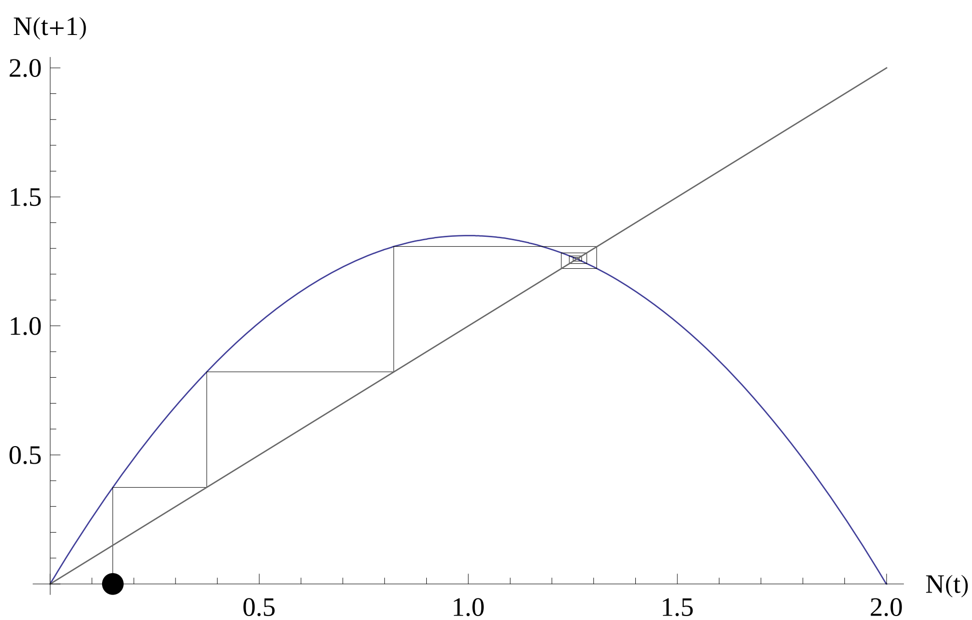
Say we have a population model where the population after a year (one time-step) is N(t+1) = 1.35 N(t) (2-N(t)). An easy way to visualize the long term behavior of the population is to create a cobweb plot, as below.

The population function is plotted along with the line N(t+1) = N(t). An initial population value is chosen.
A vertical line extends from N_0 to N(N_0). This is the new population level.
A horizontal line is extended from the point (N_0, N(N_0)) to the line, defining the new initial population level.

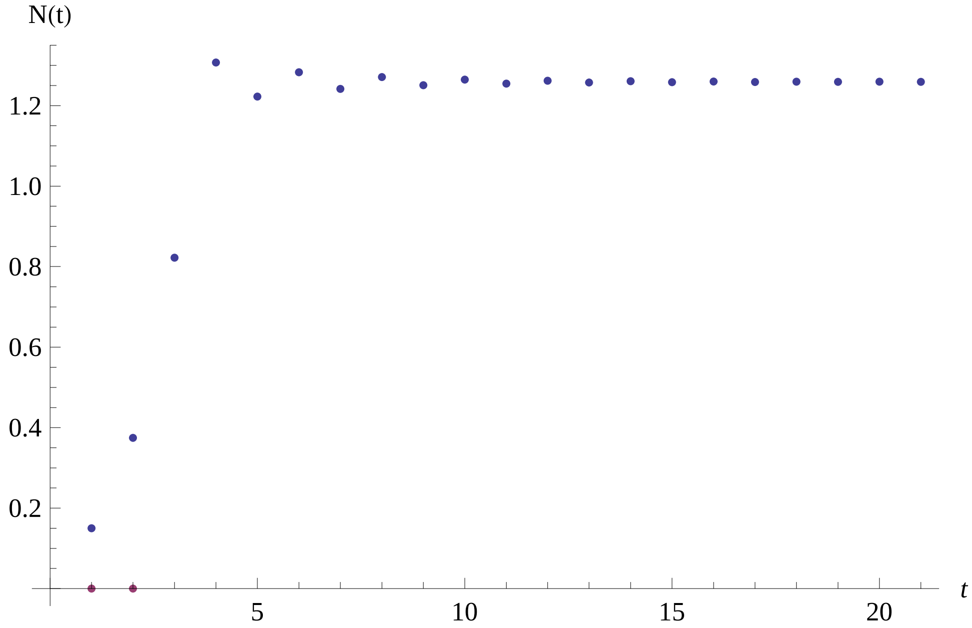
The second plot shows the population level after τ time steps. Notice the similarity to the familiar logarithmic population growth model. The parabola as the function defining the population change is known as a logistic map.

In[71]:=
```
N0 = .15;
test11[x_] := 1.35 x (-x + 2)
CobwebPlot[test11[x], {x, 0, 2}, N0, 10]
ListPlot[{NestList[test11, N0, 20], {0, 0}}, AxesLabel → {t, "N(t)"}]
```

Out[73]=



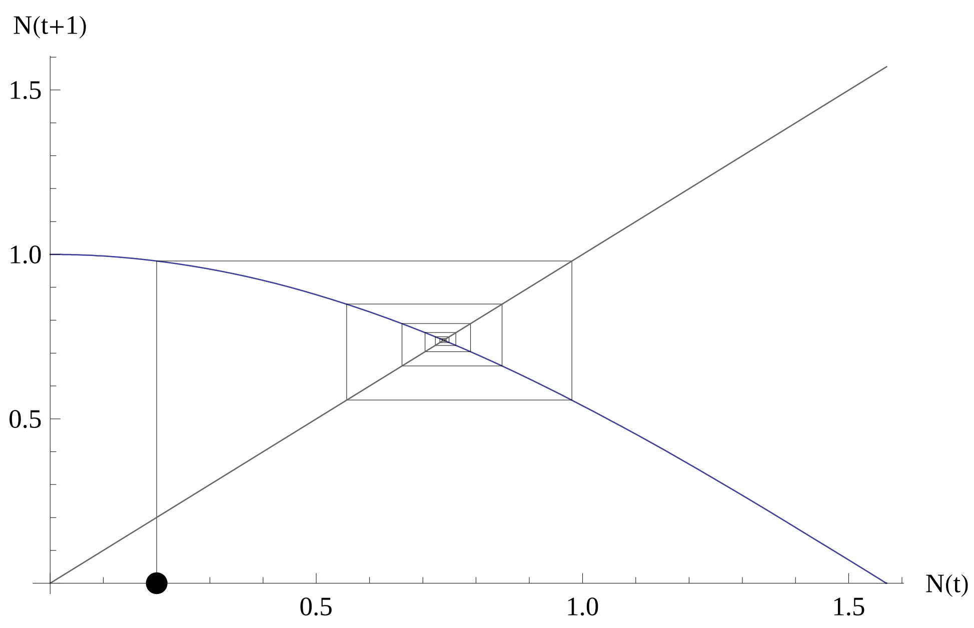
Out[74]=



Of course, other maps are possible:

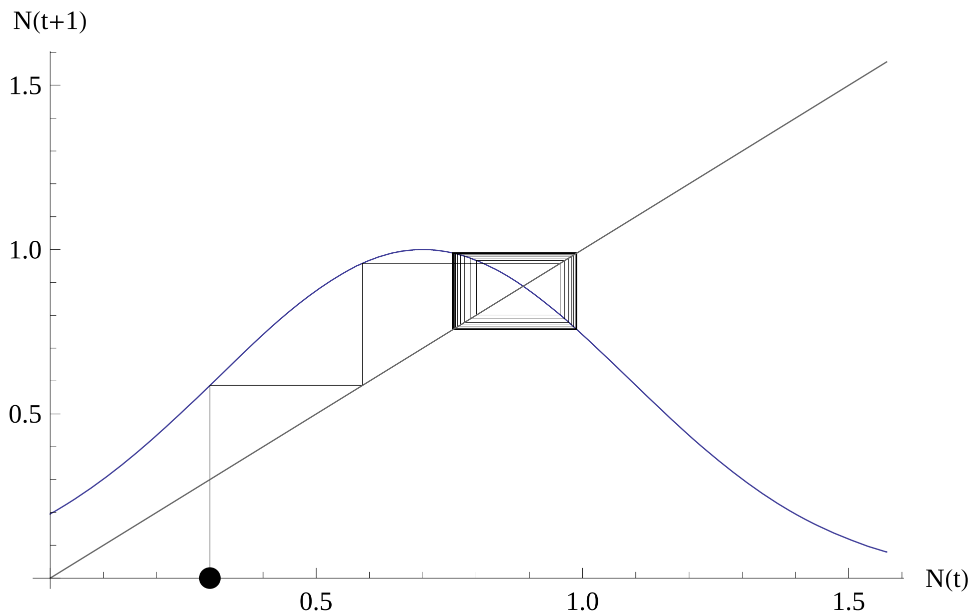
In[75]:=
```
test1[x_] := Cos[x]
CobwebPlot[test1[x], {x, 0, π / 2}, 0.2, 15]
```

Out[76]=

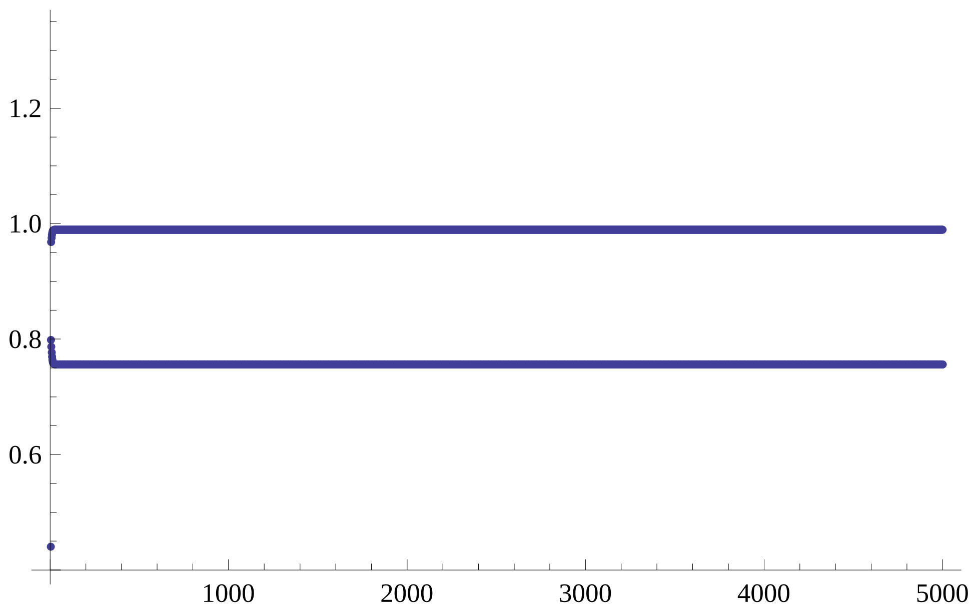


In[77]:=
```
test2[x_] := Exp[-(x - .7)^2 / .3]
CobwebPlot[test2[x], {x, 0, π / 2}, .3, 300]
ListPlot[NestList[test2, 1.885, 5000]]
```

Out[78]=



Out[79]=



But here we see the population bounces between two different values or fixed points. It has periodicity 2. Let's examine this in greater detail with the logistic equation.

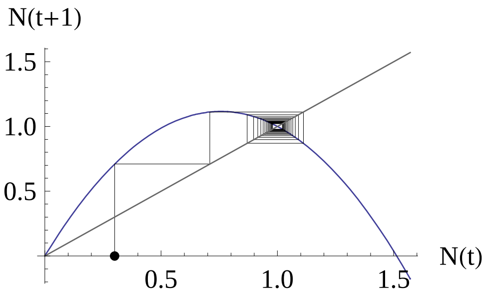
Through the '70s, ecologists were split into two camps. One thought populations were deterministic, periodic and predictable while the other thought environmental effects created too much noise to ever accurately predict populations. Chaos shows both are right some of the time.

```
In[80]:= logisticmap[r_][x_] := x + r x (1 - x)
```

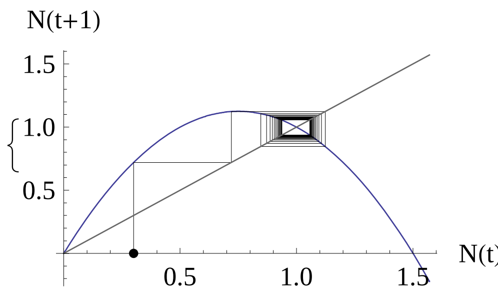
Now we start at the same point, 0.3, and vary the value of r, from 1.95 to 3.00.

```
In[81]:= Table[{CobwebPlot[logisticmap[i][x], {x, 0, π / 2}, .3, 30], i}, {i, 1.95, 3, .05}]
```

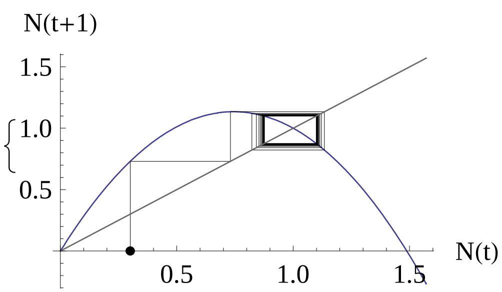
Out[81]= {{



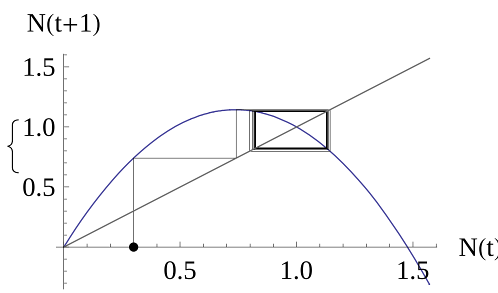
, 1.95}, {



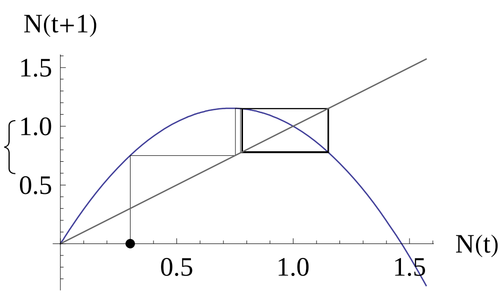
, 2.}, {



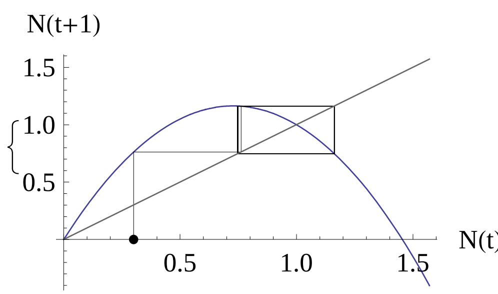
, 2.05}, {



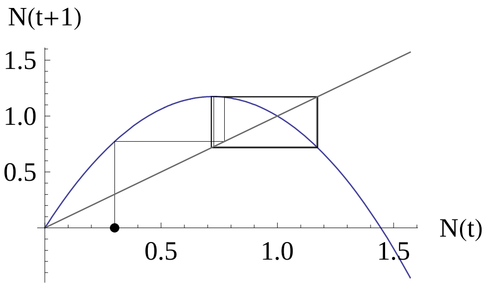
, 2.1}, {

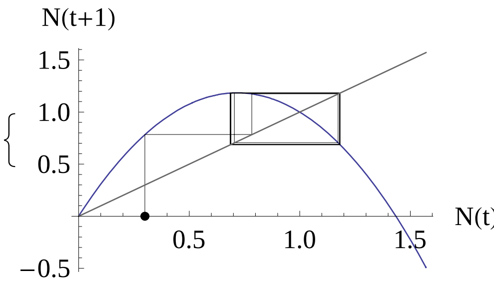


, 2.15}, {

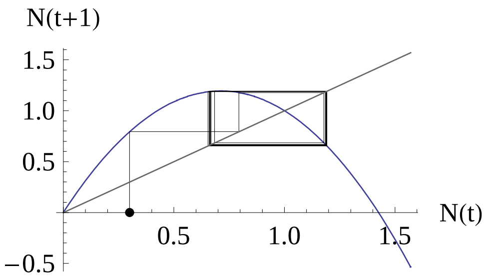


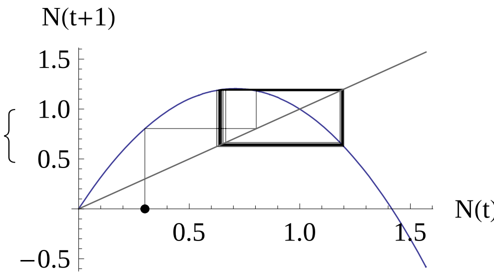
, 2.2},

{ 

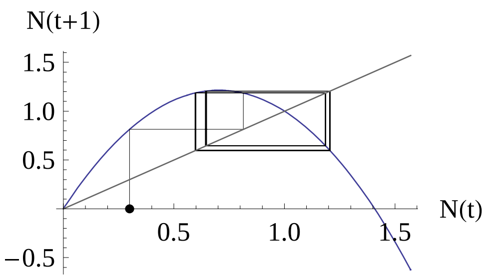
, 2.25}, { 

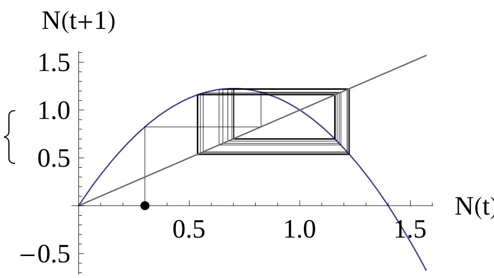
, 2.3},

{ 

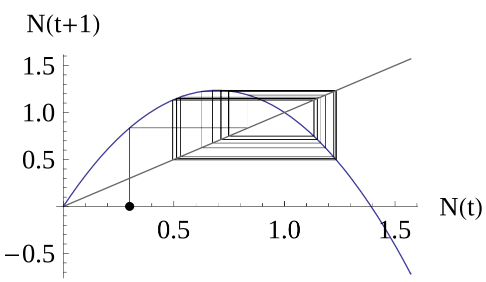
, 2.35}, { 

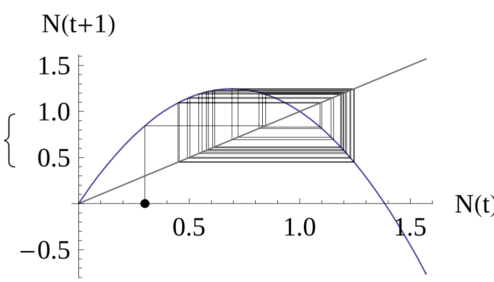
, 2.4},

{ 

, 2.45}, { 

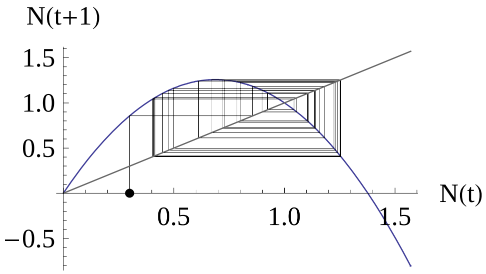
, 2.5},

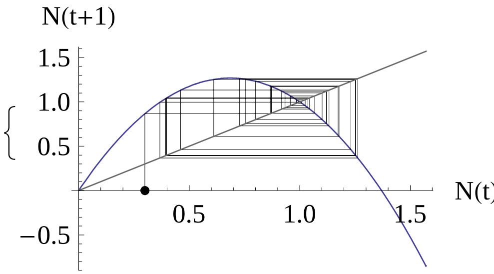
{ 

, 2.55}, { 

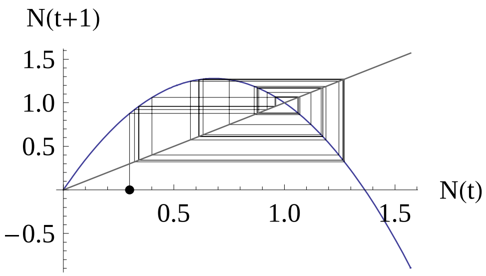
, 2.6},

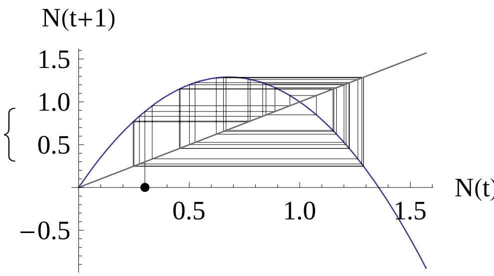
"Cell[TextData[{Cell[TextData[{ValueBox["FileName"]}], "Header"], Cell[" ", "Header", CellFrame -> {{0, 0.5}, {0, 0}}, CellFrameMargins -> 4], " ", Cell[TextData[{CounterBox["Page"]}], "PageNumber"]}], CellMargins -> {{Inherited, 0}, {Inher

{ 

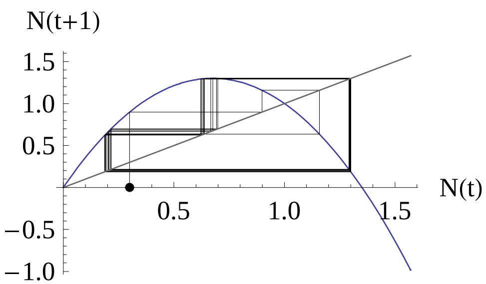
, 2.65}, { 

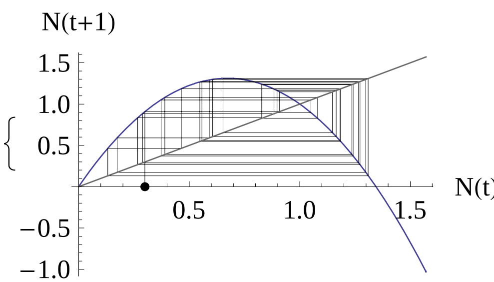
, 2.7},

{ 

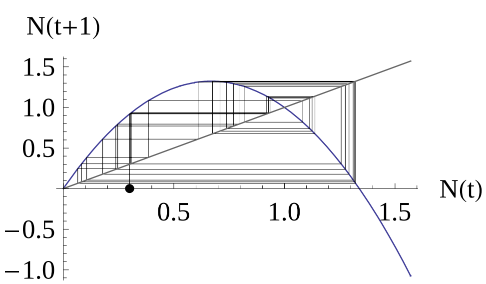
, 2.75}, { 

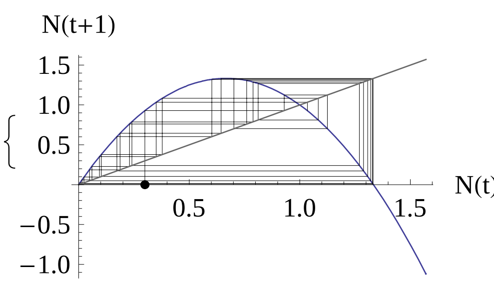
, 2.8},

{ 

, 2.85}, { 

, 2.9},

{ 

, 2.95}, { 

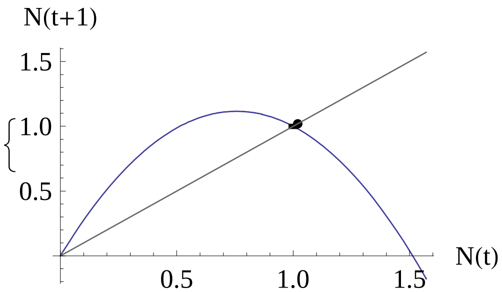
, 3.}}

These cobweb plots seem to be showing more and more complexity! How much of this is real and how much is just transient effects? We'll investigate by applying 300 recursions and ignoring the first 30.

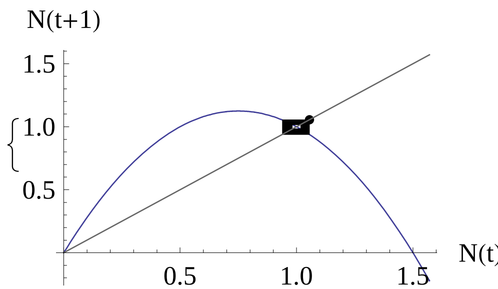
"Cell[TextData[{Cell[TextData[{ValueBox["FileName"]}], "Header"], Cell[" ", "Header", CellFrame -> {{0, 0.5}, {0, 0}}, CellFrameMargins -> 4], " ", Cell[TextData[{CounterBox["Page"]}], "PageNumber"]}], CellMargins -> {{Inherited, 0}, {Inher

```
In[91]:= Table[{CobwebPlot[logisticmap[i][x], {x, 0, π / 2}, .3, {30, 300}], i},
           {i, 1.95, 3, .05}]
```

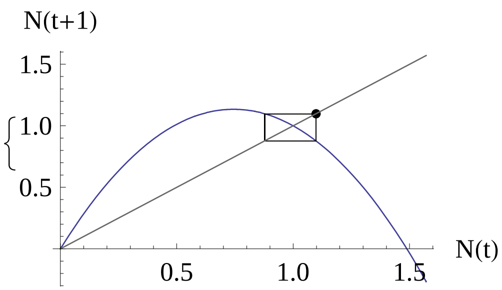
Out[91]= {{



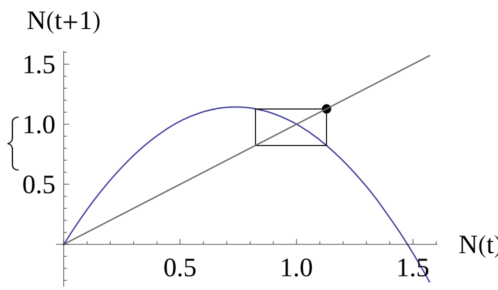
, 1.95}, {



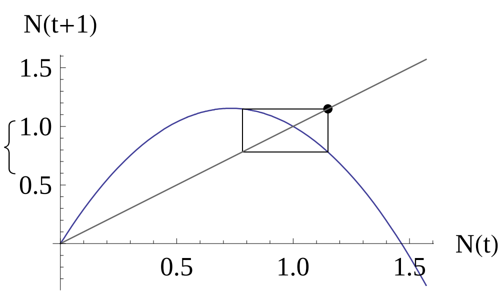
, 2.}, {



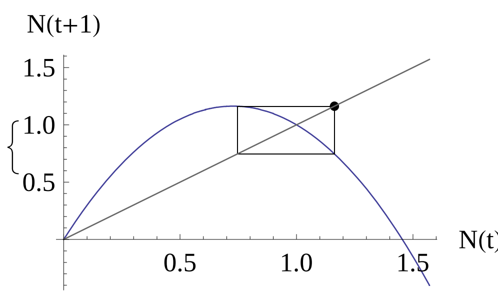
, 2.05}, {



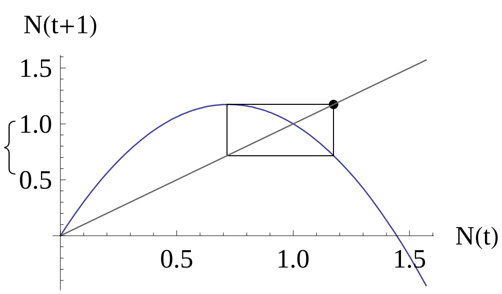
, 2.1}, {



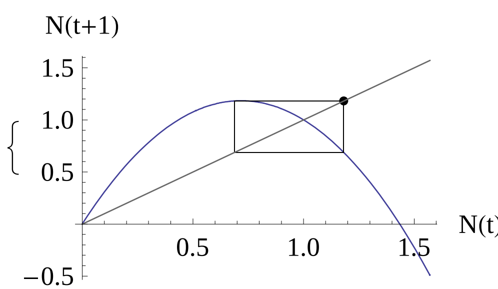
, 2.15}, {



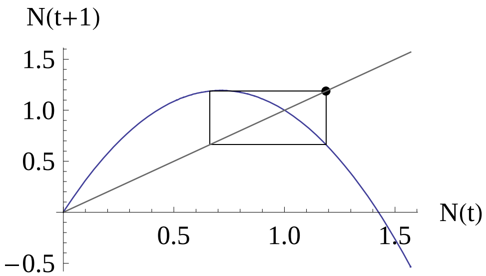
, 2.2}, {

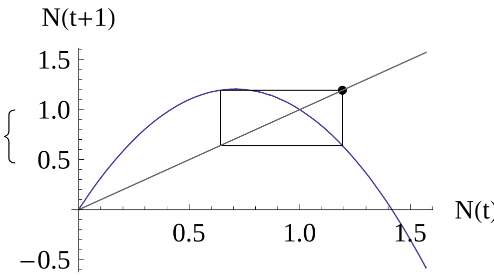


, 2.25}, {

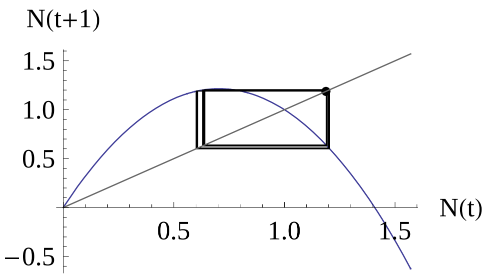


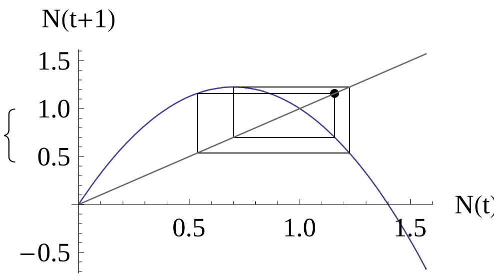
, 2.3},

{ 

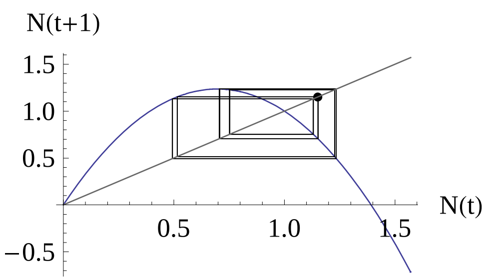
, 2.35}, { 

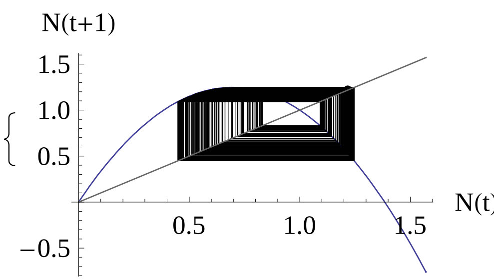
, 2.4},

{ 

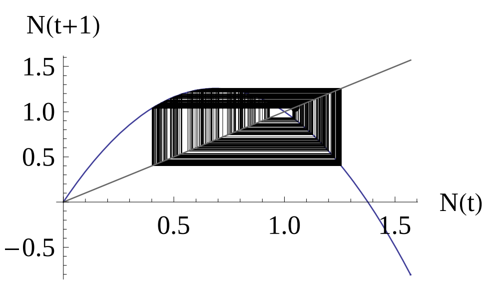
, 2.45}, { 

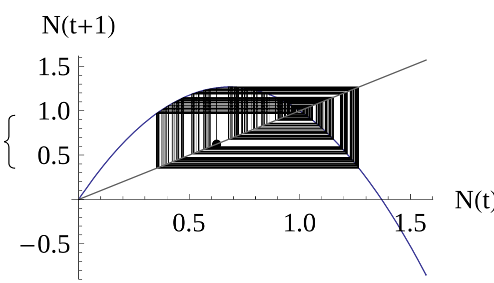
, 2.5},

{ 

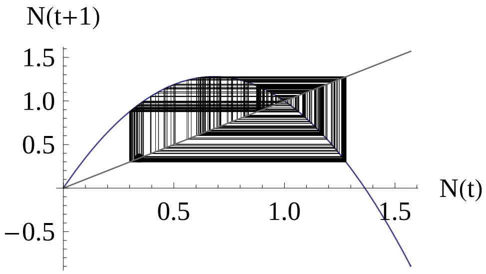
, 2.55}, { 

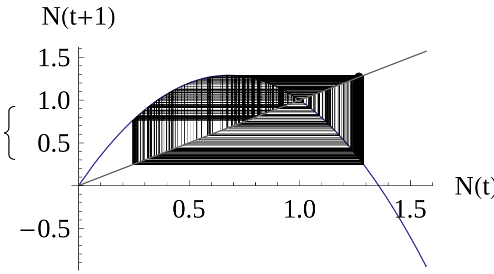
, 2.6},

{ 

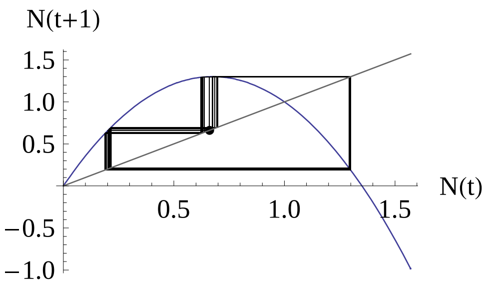
, 2.65}, { 

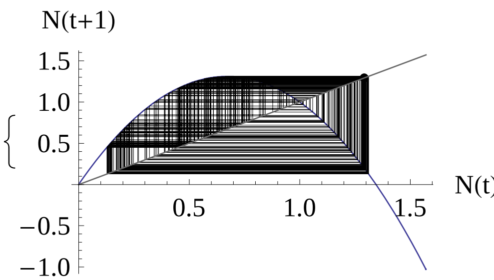
, 2.7},

{

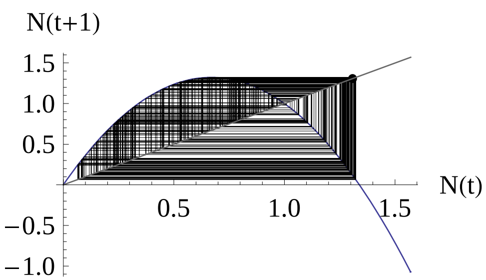
, 2.75}, {

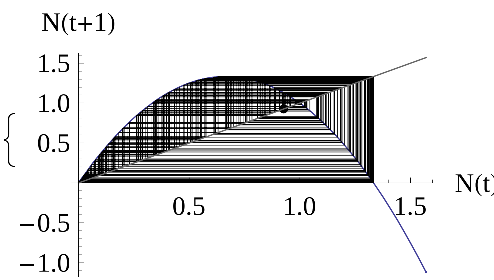
, 2.8},

{

, 2.85}, {

, 2.9},

{

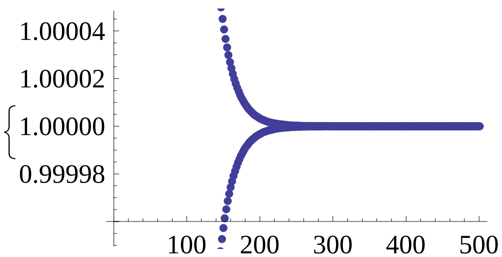
, 2.95}, {

, 3.}}

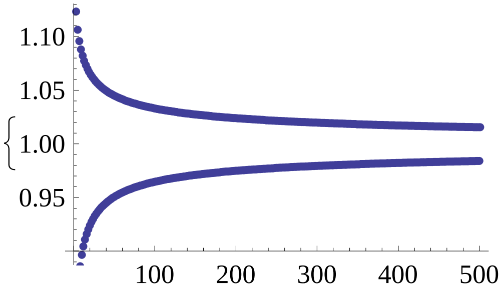
Some clearly look chaotic, but not neccessarily in a predictable way based on the r-value.
Looking at this from the perspective of fixed points:

In[92]:=
```
Table[{ListPlot[NestList[logisticmap[i], .3, 500]], i}, {i, 1.95, 3, .05}]
```

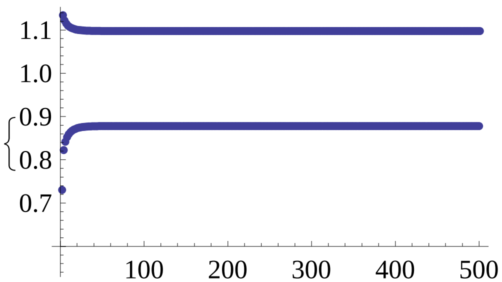
Out[92]= {{



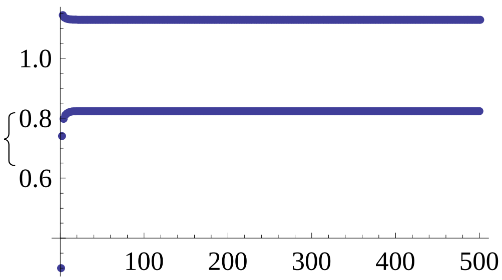
, 1.95}, {



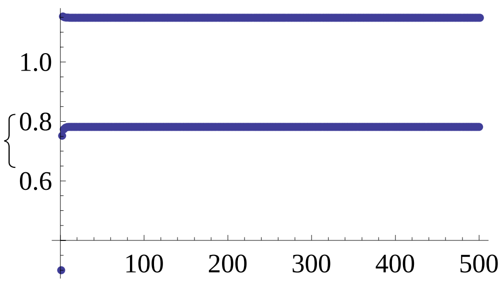
, 2.}, {



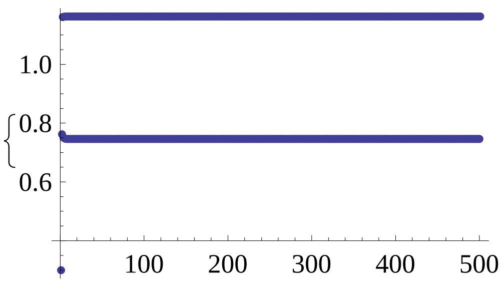
, 2.05}, {



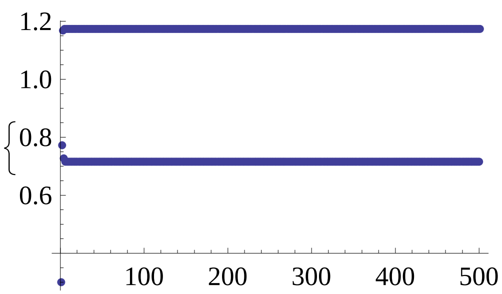
, 2.1}, {



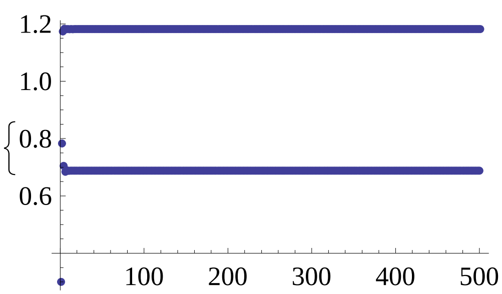
, 2.15}, {



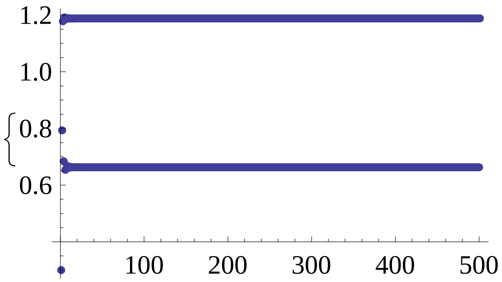
, 2.2}, {

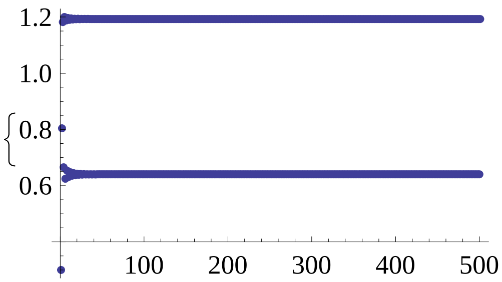


, 2.25}, {

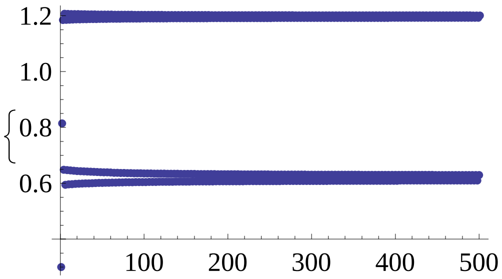


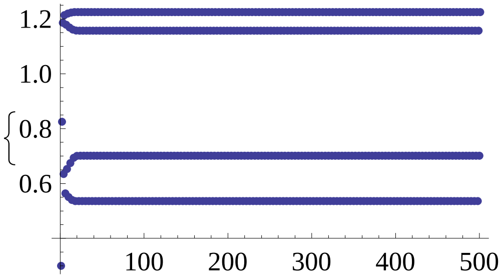
, 2.3},

{

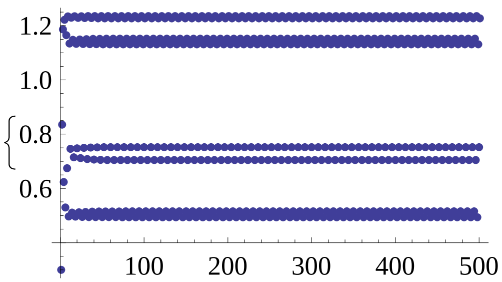
, 2.35}, {

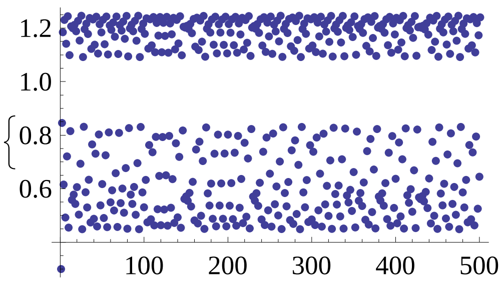
, 2.4},

{

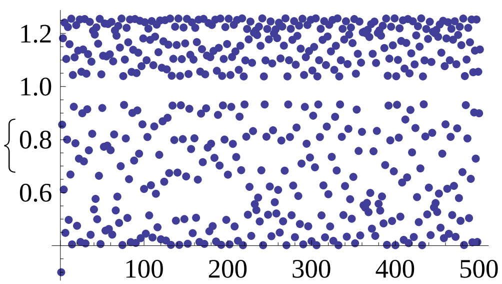
, 2.45}, {

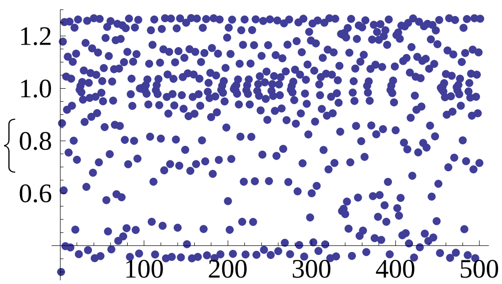
, 2.5},

{

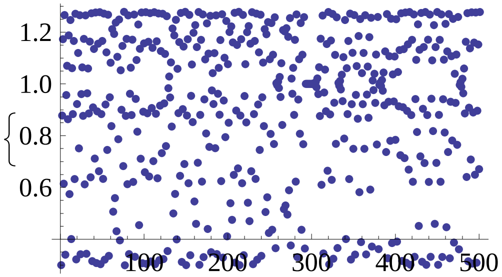
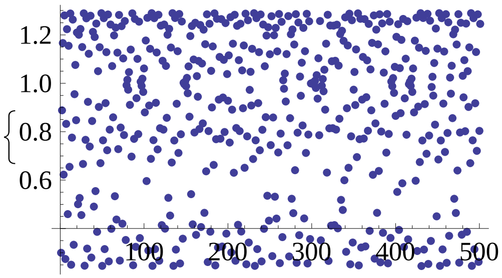
, 2.55}, {

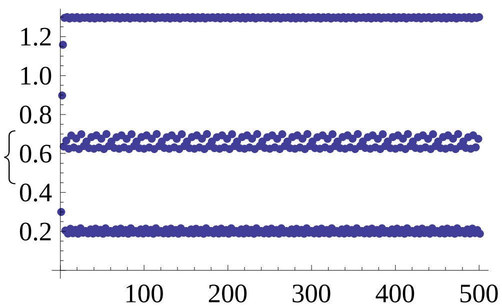
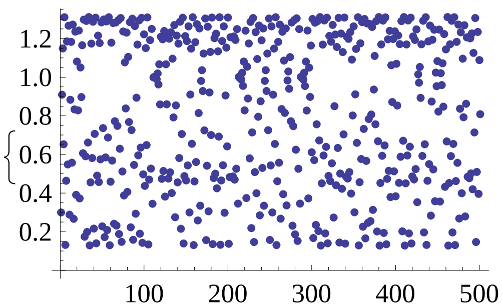
, 2.6},

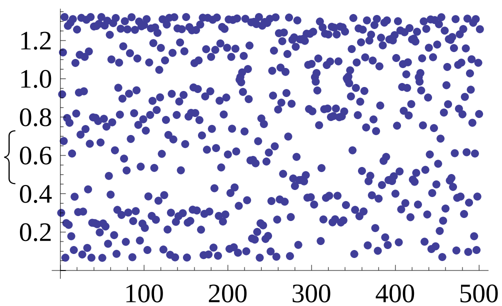
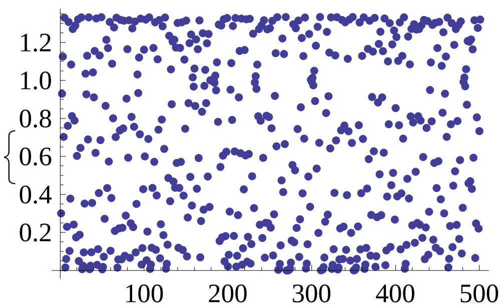
{

, 2.65}, {

, 2.7},

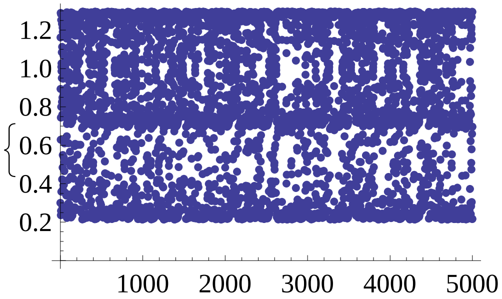
{, 2.75}, {, 2.8},

{, 2.85}, {, 2.9},

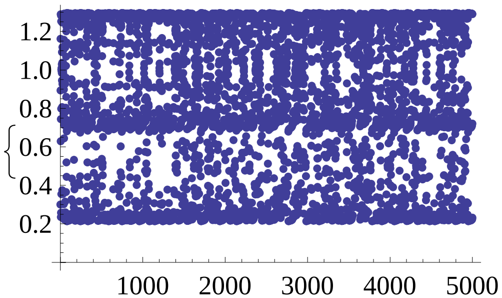
{, 2.95}, {, 3.}}

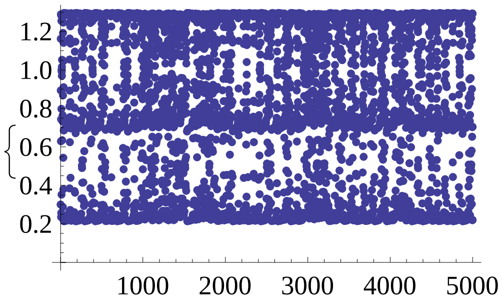
In some of these plots, the population never returns to the same level! It is chaotic. But how quickly does it change?
If we vary the thousandths' place, we get:

```
In[94]:= Table[{ListPlot[NestList[logisticmap[i], .3, 5000]], i}, {i, 2.828, 2.829, .0001}]
```

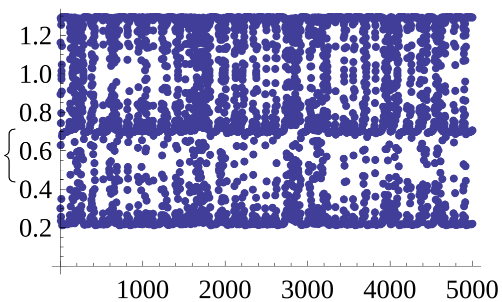
Out[94]= {{

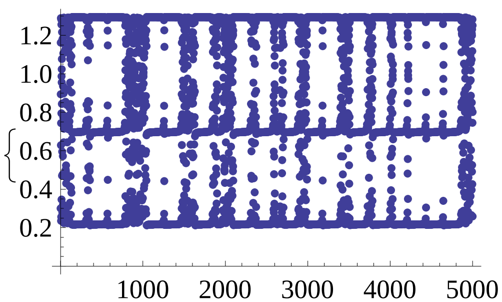
, 2.828},

{

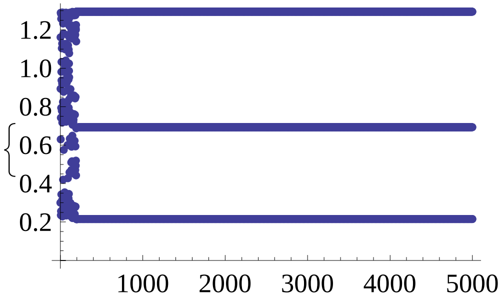
, 2.8281}, {

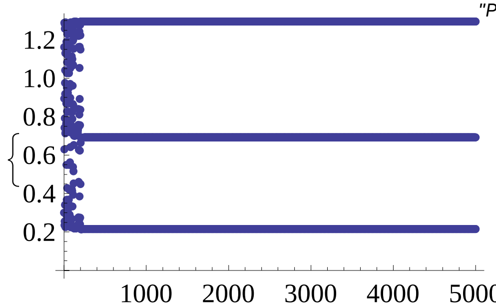
, 2.8282},

{

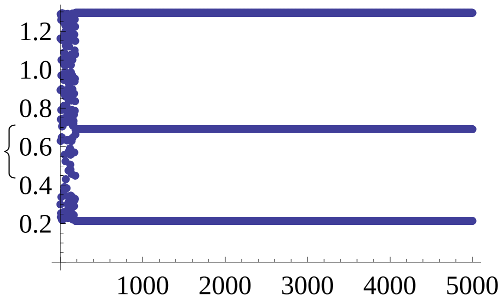
, 2.8283}, {

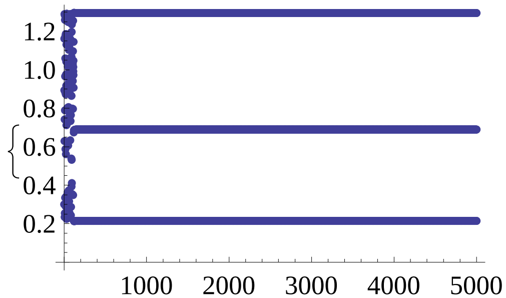
, 2.8284},

{

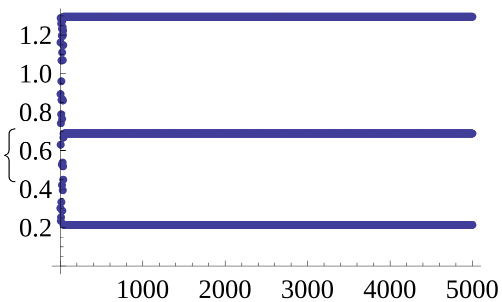
, 2.8285}, {

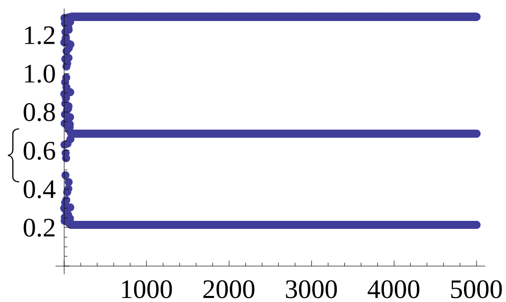
, 2.8286},

{

, 2.8287}, {

, 2.8288},

{

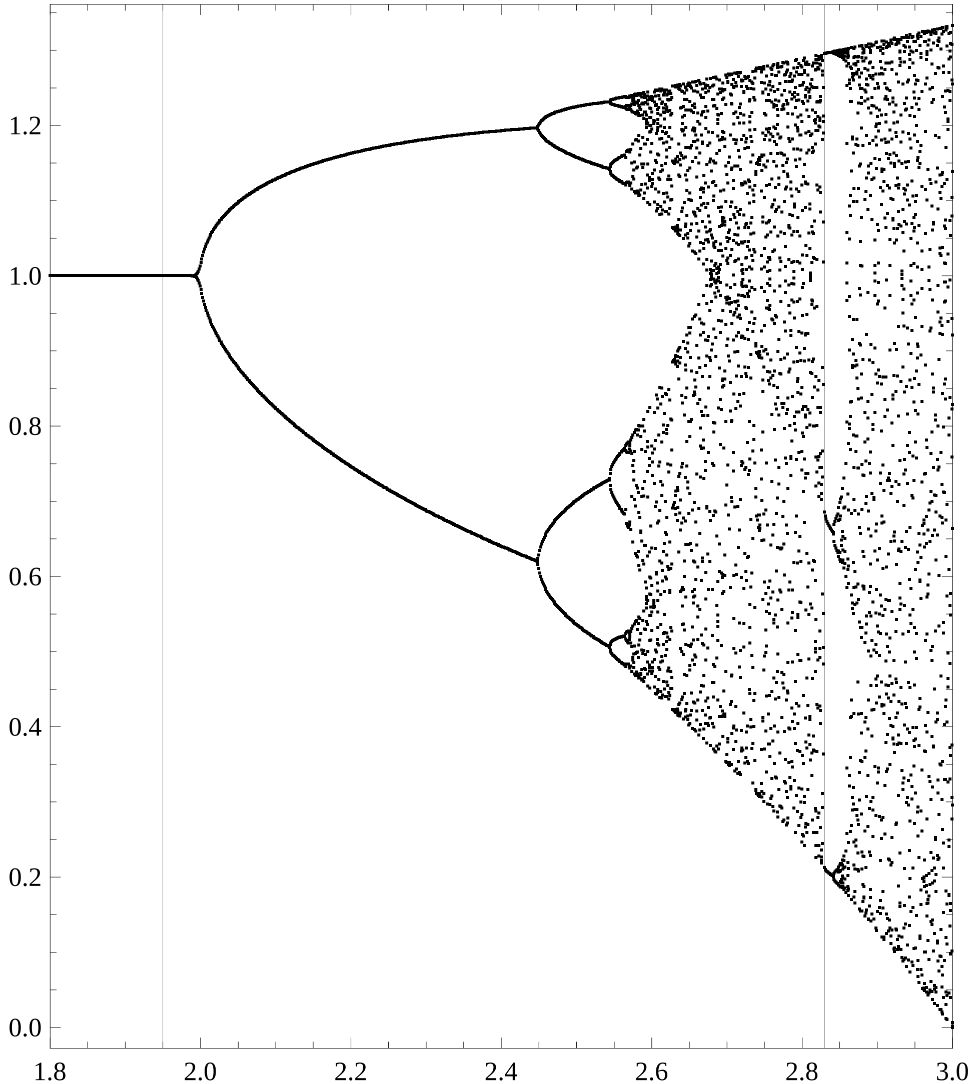
, 2.8289}, {

, 2.829}}

Happily we can summarize these results in a bifurcation plot.

In[96]:=
```
BifurcationPlot[logisticmap[r][x], {r, 1.8, 3}, {x, 0.1}, {500, 10},
 PlotPoints → 1000, GridLines → {{1.95, 2.83, 3}, None}]
```

Out[96]=

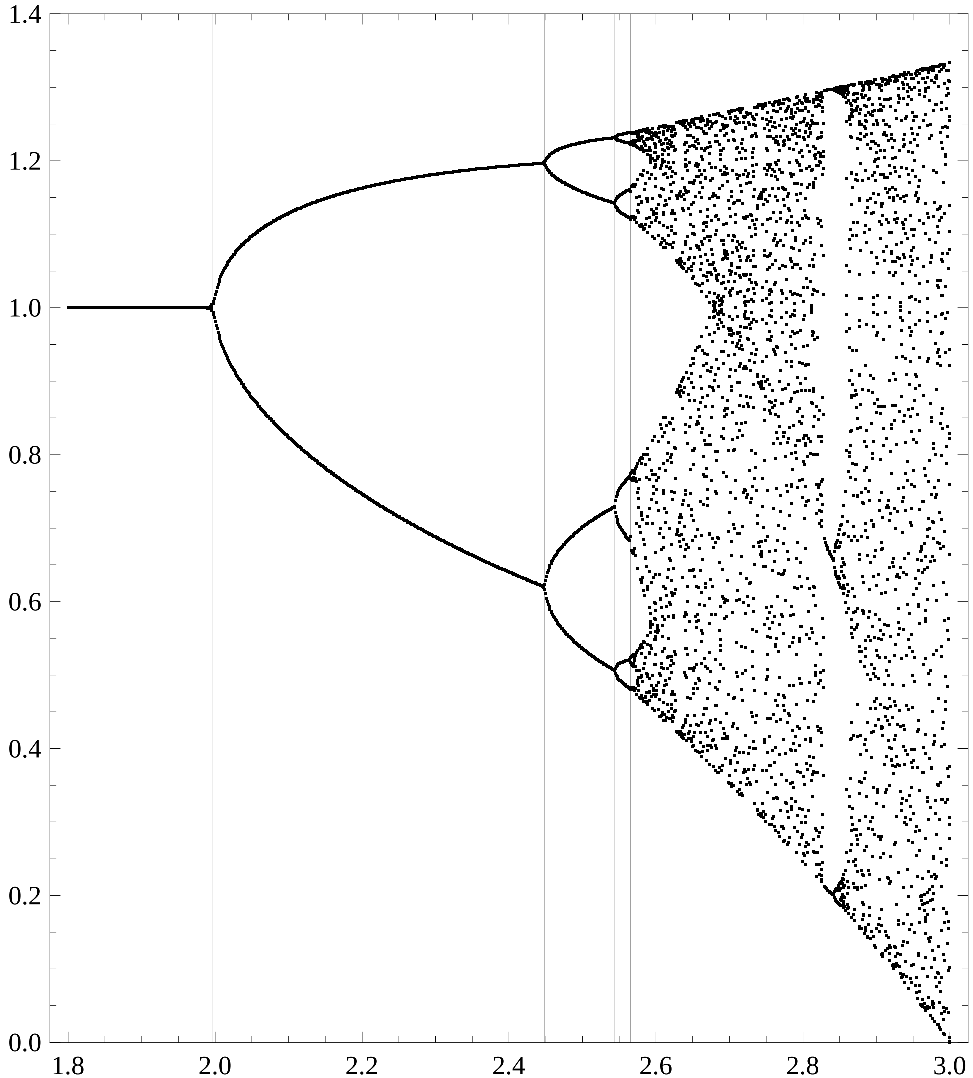


"Cell[TextData[{Cell[TextData[{ValueBox["FileName"]}], "Header"], Cell[" ", "Header", CellFrame -> {{0, 0.5}, {0, 0}}, CellFrameMargins -> 4], " ", Cell[TextData[{CounterBox["Page"]}], "PageNumber"]}], CellMargins -> {{Inherited, 0}, {Inher

We see initial bifurcations, or binary splittings from one fixed point to two at 1.997, then to 4 at 2.448, etc. Labeling these and looking at their ratio we find:

In[86]:=
```
BifurcationPlot[logisticmap[r][x], {r, 1.8, 3}, {x, 0.1}, {500, 10},
 PlotPoints → 1000,
 GridLines → {{1.997, 2.448, 2.544, 2.565}, None},
 PlotRange → {0, 1.4}]
```

Out[86]=



In[97]:= $\frac{2.448 - 1.997}{2.544 - 2.448}$ $\frac{2.544 - 2.448}{2.565 - 2.544}$

Out[97]= 4.69792

Out[98]= 4.57143

In fact, in the limit n→∞, the ratio $\frac{r_{2(n+1)} - r_{2n}}{r_{2(n+2)} - r_{2(n+1)}} \to 4.66920160910299067185320382...$, Feigenbaum's constant.

In[103]:= **N[π, 50]**

Out[103]= 3.1415926535897932384626433832795028841971693993751