Single population growth models

Objective

We are given a table with the population census at different time intervals between a date a and a date b, and want to get an expression for the population. This allows us to:

- compute a value for the population at any time between the date a and the date b (*interpolation*),
- predict a value for the population at a date before a or after b (*extrapolation*).

PROCEEDINGS
OF THE
NATIONAL ACADEMY OF SCIENCES

Volume 6 JUNE 15, 1920 Number 6

ON THE RATE OF GROWTH OF THE POPULATION OF THE UNITED STATES SINCE 1790 AND ITS MATHEMATICAL REPRESENTATION[1]

BY RAYMOND PEARL AND LOWELL J. REED

DEPARTMENT OF BIOMETRY AND VITAL STATISTICS, JOHNS HOPKINS UNIVERSITY

Read before the Academy, April 26, 1920

SHOWING THE DATES OF THE TAKING OF THE CENSUS AND THE RECORDED POPULATIONS FROM 1790 TO 1910

DATE OF CENSUS		RECORDED POPULATION (REVISED FIGURES FROM STATISTICAL ABST., 1918)
Year	Month and Day	
1790	First Monday in August	3,929,214
1800	First Monday in August	5,308,483
1810	First Monday in August	7,239,881
1820	First Monday in August	9,638,453
1830	June 1	12,866,020
1840	June 1	17,069,453
1850	June 1	23,191,876
1860	June 1	31,443,321
1870	June 1	38,558,371
1880	June 1	50,155,783
1890	June 1	62,947,714
1900	June 1	75,994,575
1910	April 15	91,972,266

The US population from 1790 to 1910

Year	Population (millions)
1790	3.929
1800	5.308
1810	7.240
1820	9.638
1830	12.866
1840	17.069
1850	23.192

Year	Population (millions)
1860	31.443
1870	38.558
1880	50.156
1890	62.948
1900	75.995
1910	91.972

PLOT THE DATA !!! (here, to 1910)

Using MatLab (or Octave), create two vectors using commands such as

`t=1790:10:1910;`
Format is

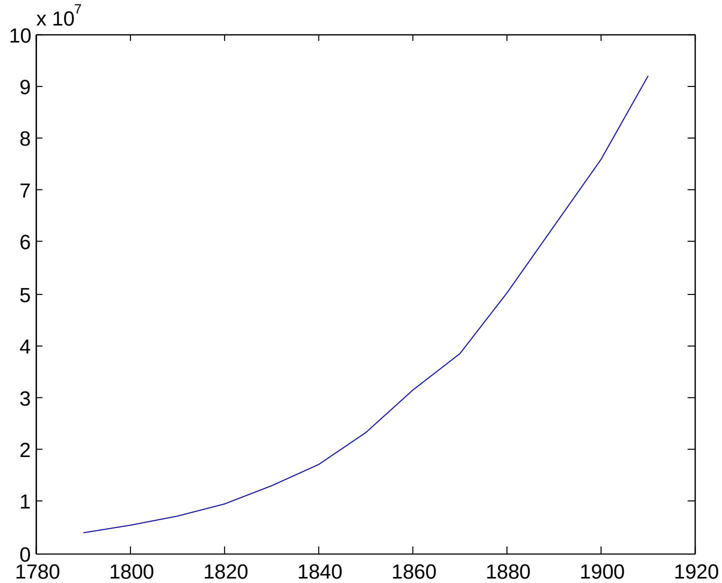
Vector=Initial value:Step:Final value

(semicolumn hides result of the command.)

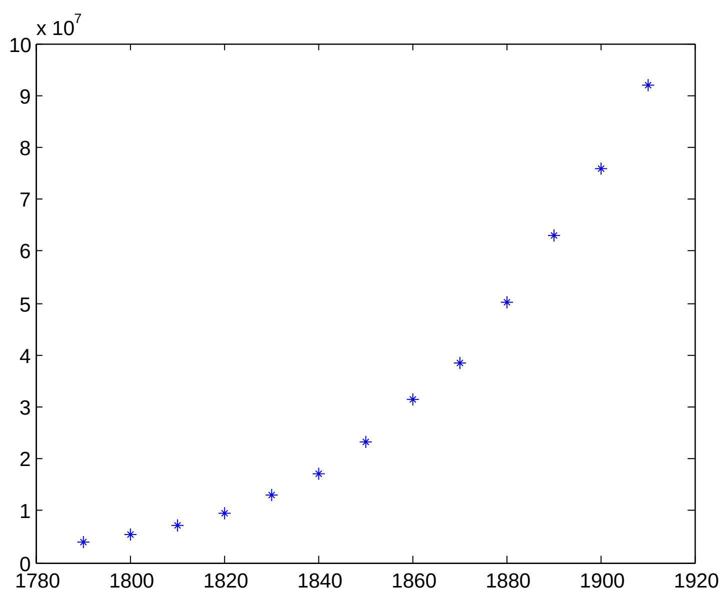
```
P=[3929214,5308483,7239881,9638453,12866020,...
17069453,23191876,31443321,38558371,50155783,...
62947714,75994575,91972266];
```

Here, elements were just listed (`...` indicates that the line continues below).

Then plot using
```
plot(t,P);
```



To get points instead of a line
```
plot(t,P,'*');
```



First idea

The curve looks like a piece of a parabola. So let us fit a curve of the form

$$P(t) = a + bt + ct^2.$$

To do this, we want to minimize

$$S = \sum_{k=1}^{13} \left(P(t_k) - P_k\right)^2,$$

where t_k are the known dates, P_k are the known populations, and $P(t_k) = a + bt_k + ct_k^2$.

We proceed as in the notes (but note that the role of a, b, c is reversed):

$$S = S(a, b, c) = \sum_{k=1}^{13} \left(a + bt_k + ct_k^2 - P_k\right)^2$$

is maximal if (necessary condition) $\partial S/\partial a = \partial S/\partial b = \partial S/\partial c = 0$, with

$$\frac{\partial S}{\partial a} = 2\sum_{k=1}^{13}(a + bt_k + ct_k^2 - P_k)$$

$$\frac{\partial S}{\partial b} = 2\sum_{k=1}^{13}(a + bt_k + ct_k^2 - P_k)t_k$$

$$\frac{\partial S}{\partial c} = 2\sum_{k=1}^{13}(a + bt_k + ct_k^2 - P_k)t_k^2$$

So we want

$$2\sum_{k=1}^{13}(a + bt_k + ct_k^2 - P_k) = 0$$
$$2\sum_{k=1}^{13}(a + bt_k + ct_k^2 - P_k)t_k = 0$$
$$2\sum_{k=1}^{13}(a + bt_k + ct_k^2 - P_k)t_k^2 = 0,$$

that is

$$\sum_{k=1}^{13}(a + bt_k + ct_k^2 - P_k) = 0$$
$$\sum_{k=1}^{13}(a + bt_k + ct_k^2 - P_k)t_k = 0$$
$$\sum_{k=1}^{13}(a + bt_k + ct_k^2 - P_k)t_k^2 = 0.$$

Rearranging the system

$$\sum_{k=1}^{13}(a + bt_k + ct_k^2 - P_k) = 0$$
$$\sum_{k=1}^{13}(a + bt_k + ct_k^2 - P_k)t_k = 0$$
$$\sum_{k=1}^{13}(a + bt_k + ct_k^2 - P_k)t_k^2 = 0,$$

we get

$$\sum_{k=1}^{13}(a + bt_k + ct_k^2) = \sum_{k=1}^{13} P_k$$
$$\sum_{k=1}^{13}(at_k + bt_k^2 + ct_k^3) = \sum_{k=1}^{13} P_k t_k$$
$$\sum_{k=1}^{13}(at_k^2 + bt_k^3 + ct_k^4) = \sum_{k=1}^{13} P_k t_k^2.$$

$$\begin{aligned}\sum_{k=1}^{13}(a+bt_k+ct_k^2) &= \sum_{k=1}^{13}P_k\\ \sum_{k=1}^{13}(at_k+bt_k^2+ct_k^3) &= \sum_{k=1}^{13}P_k t_k\\ \sum_{k=1}^{13}(at_k^2+bt_k^3+ct_k^4) &= \sum_{k=1}^{13}P_k t_k^2,\end{aligned}$$

after a bit of tidying up, takes the form

$$\begin{aligned}\left(\sum_{k=1}^{13}1\right)a+\left(\sum_{k=1}^{13}t_k\right)b+\left(\sum_{k=1}^{13}t_k^2\right)c &= \sum_{k=1}^{13}P_k\\ \left(\sum_{k=1}^{13}t_k\right)a+\left(\sum_{k=1}^{13}t_k^2\right)b+\left(\sum_{k=1}^{13}t_k^3\right)c &= \sum_{k=1}^{13}P_k t_k\\ \left(\sum_{k=1}^{13}t_k^2\right)a+\left(\sum_{k=1}^{13}t_k^3\right)b+\left(\sum_{k=1}^{13}t_k^4\right)c &= \sum_{k=1}^{13}P_k t_k^2.\end{aligned}$$

So the aim is to solve the linear system

$$\begin{pmatrix} 13 & \sum_{k=1}^{13} t_k & \sum_{k=1}^{13} t_k^2\\ \sum_{k=1}^{13} t_k & \sum_{k=1}^{13} t_k^2 & \sum_{k=1}^{13} t_k^3\\ \sum_{k=1}^{13} t_k^2 & \sum_{k=1}^{13} t_k^3 & \sum_{k=1}^{13} t_k^4\end{pmatrix}\begin{pmatrix}a\\ b\\ c\end{pmatrix}=\begin{pmatrix}\sum_{k=1}^{13} P_k\\ \sum_{k=1}^{13} P_k t_k\\ \sum_{k=1}^{13} P_k t_k^2\end{pmatrix}$$

With MatLab (or Octave), getting the values is easy.

- To apply an operation to every element in a vector or matrix, prefix the operation with a dot, hence

  ```
  t.^2;
  ```

  gives, for example, the vector with every element t_k squared.
- Also, the function `sum` gives the sum of the entries of a vector or matrix.
- When entering a matrix or vector, separate entries on the same row by `,` and create a new row by using `;`.

Thus, to set up the problem in the form of solving $Ax = b$, we need to do the following:

```
format long g;
A=[13,sum(t),sum(t.^2);sum(t),sum(t.^2),sum(t.^3);...
sum(t.^2),sum(t.^3),sum(t.^4)];
b=[sum(P);sum(P.*t);sum(P.*(t.^2))];
```

The `format long g` command is used to force the display of digits (normally, what is shown is in "scientific" notation, not very informative here).

Then, solve the system using

```
A\b
```

We get the following output:

```
>> A\b
Warning: Matrix is close to singular or badly scaled.
         Results may be inaccurate. RCOND = 1.118391e-020.

ans =

        22233186177.8195
        -24720291.325476
        6872.99686313725
```

(note that here, Octave gives a solution that is not as good as this one, provided by MatLab).

Thus

$$P(t) = 22233186177.8195 - 24720291.325476t + 6872.99686313725t^2$$

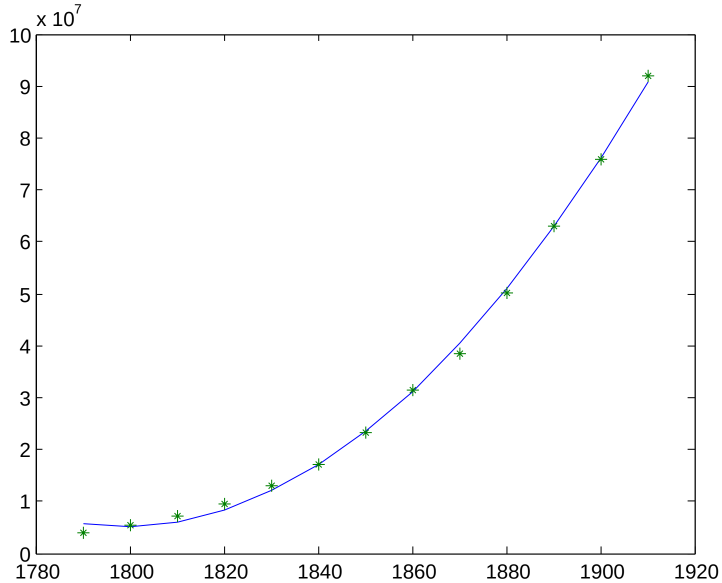
To see what this looks like,

```
plot(t,22233186177.8195-24720291.325476.*t...
+6872.99686313725.*t.^2);
```

(note the dots before multiplication and power, since we apply this function to every entry of t). In fact, to compare with original data:

```
plot(t,22233186177.8195-24720291.325476.*t...
+6872.99686313725.*t.^2,t,P,'*');
```

Our first guess, in pictures



Now we want to generate the table of values, to compare with the true values and thus compute the error. To do this, we can proceed directly:

```
computedP=22233186177.8195-24720291.325476.*t...
+6872.99686313725.*t.^2;
```

We get

```
computedP =

 Columns 1 through 4:

     5633954.39552689       5171628.52739334       6083902.03188705       8370774.90901184

 Columns 5 through 8:

     12032247.1587601       17068318.7811356       23478989.7761383       31264260.1437798

 Columns 9 through 12:

      40424129.884037       50958598.9969215       62867667.4824371       76151335.3405762

 Column 13:

     90809602.5713463
```

We can also create an *inline* function

```
f=inline('22233186177.8195-24720291.325476.*t+6872.99686313725.*t.^2')
f =

     Inline function:
     f(t) = 22233186177.8195-24720291.325476.*t+6872.99686313725.*t.^2
```

This function can then easily be used for a single value

```
octave:24> f(1880)
ans =      50958598.9969215
```

as well as for vectors..

(Recall that t has the dates; t in the definition of the function is a dummy variable, we could have used another letter-.)

```
octave:25> f(t)

ans =

 Columns 1 through 4:

      5633954.39552689      5171628.52739334      6083902.03188705      8370774.90901184

 Columns 5 through 8:

      12032247.1587601      17068318.7811356      23478989.7761383      31264260.1437798

 Columns 9 through 12:

       40424129.884037      50958598.9969215      62867667.4824371      76151335.3405762
12186176863781.4
 Column 13:

      90809602.5713463
```

Form the vector of errors, and compute sum of errors squared:

```
octave:26> E=f(t)-P;
octave:27> sum(E.^2)
ans =      12186176863781.4
```

Quite a large error (12,186,176,863,781.4), which is normal since we have used actual numbers, not thousands or millions of individuals, and we are taking the square of the error.

To present things legibly, one way is to put everything in a matrix..

```
M=[P;f(t);E;E./P];
```

This matrix will have each type of information as a row, so to display it in the form of a table, show its transpose, which is achieved using the function `transpose` or the operator $'$.

```
M'
ans =
 3929214      5633954.39552689      1704740.39552689     0.433862954658
 5308483      5171628.52739334     -136854.472606659   -0.0257803354756
 7239881      6083902.03188705     -1155978.96811295    -0.159668227711
 9638453      8370774.90901184     -1267678.09098816    -0.131522983095
12866020      12032247.1587601     -833772.841239929   -0.0648042550252
17069453      17068318.7811356     -1134.21886444092  -6.644728828e-05
23191876      23478989.7761383      287113.776138306    0.0123799289086
31443321      31264260.1437798     -179060.856220245  -0.00569471832254
38558371       40424129.884037      1865758.88403702    0.0483879073635
50155783      50958598.9969215      802815.996921539    0.0160064492846
62947714      62867667.4824371     -80046.5175628662  -0.00127163502018
75994575      76151335.3405762      156760.340576172   0.00206278330494
91972266      90809602.5713463     -1162663.42865372    -0.012641456813
```


Now for the big question...

How does our formula do for present times?

```
f(2006)
ans =     301468584.066013
```

Actually, quite well: 301,468,584, compared to the 298,444,215 July 2006 estimate, overestimates the population by 3,024,369, a relative error of approximately 1%.

The US population from 1790 to 2000 (revised numbers)

Year	Population (millions)	Year	Population (millions)
1790	3.929	1900	76.212
1800	5.308	1910	92.228
1810	7.240	1920	106.021
1820	9.638	1930	123.202
1830	12.866	1940	132.164
1840	17.069	1950	151.325
1850	23.192	1960	179.323
1860	31.443	1970	203.302
1870	38.558	1980	226.542
1880	50.156	1990	248.709
1890	62.948	2000	281.421

Other similar approaches

Pritchett, 1891:

$$P = a + bt + ct^2 + dt^3.$$

(we have done this one, and found it to be quite good too).
Pearl, 1907:

$$P(t) = a + bt + ct^2 + d \ln t.$$

Finds

$$P(t) = 9,064,900 - 6,281,430t + 842,377t^2 + 19,829,500 \ln t.$$

SHOWING (*a*) THE ACTUAL POPULATION[1] ON CENSUS DATES, (*b*) ESTIMATED POPULATION FROM PRITCHETT'S THIRD-ORDER PARABOLA, (*c*) ESTIMATED POPULATION FROM LOGARITHMIC PARABOLA, AND (*d*) (*e*) ROOT-MEAN SQUARE ERRORS OF BOTH METHODS

CENSUS YEAR	(*a*) OBSERVED POPULATION	(*b*) PRITCHETT ESTIMATE	(*c*) LOGARITHMIC PARABOLA ESTIMATE	(*d*) ERROR OF (*b*)	(*e*) ERROR OF (*c*)
1790	3,929,000	4,012,000	3,693,000	+ 83,000	− 236,000
1800	5,308,000	5,267,000	5,865,000	− 41,000	+ 557,000
1810	7,240,000	7,059,000	7,293,000	− 181,000	+ 53,000
1820	9,638,000	9,571,000	9,404,000	− 67,000	− 234,000
1830	12,866,000	12,985,000	12,577,000	+ 119,000	− 289,000
1840	17,069,000	17,484,000	17,132,000	+ 415,000	+ 63,000
1850	23,192,000	23,250,000	23,129,000	+ 58,000	− 63,000
1860	31,443,000	30,465,000	30,633,000	− 978,000	− 810,000
1870	38,558,000	39,313,000	39,687,000	+ 755,000	+1,129,000
1880	50,156,000	49,975,000	50,318,000	− 181,000	+ 162,000
1890	62,948,000	62,634,000	62,547,000	− 314,000	− 401,000
1900	75,995,000	77,472,000	76,389,000	+1,477,000	+ 394,000
1910	91,972,000	94,673,000	91,647,000	+2,701,000	− 325,000
				935,000[2]	472,000[2]
1920		114,416,000	108,214,000		

[1] To the nearest thousand.
[2] Root-mean square error.

The logistic curve

Pearl and Reed try

$$P(t) = \frac{be^{at}}{1 + ce^{at}}$$

or

$$P(t) = \frac{b}{e^{-at} + c}.$$

The logistic equation

The logistic curve is the solution to the ordinary differential equation

$$N' = rN\left(1 - \frac{N}{K}\right),$$

which is called the *logistic equation*. r is the *intrinsic growth rate*, K is the *carrying capacity*.

This equation was introduced by Pierre-François Verhulst (1804-1849), in 1844.

Deriving the logistic equation

The idea is to represent a population with the following components:

- birth, at the *per capita* rate b,
- death, at the *per capita* rate d,
- competition of individuals with other individuals reduces their ability to survive, resulting in death.

This gives

$$N' = bN - dN - \text{competition}.$$

Accounting for competition

Competition describes the mortality that occurs when two individuals meet.

- In chemistry, if there is a concentration X of one product and Y of another product, then XY, called *mass action*, describes the number of interactions of molecules of the two products.
- Here, we assume that X and Y are of the same type (individuals). So there are N^2 contacts.
- These N^2 contacts lead to death of one of the individuals at the rate c.

Therefore, the *logistic* equation is

$$N' = bN - dN - cN^2.$$

Reinterpreting the logistic equation

The equation

$$N' = bN - dN - cN^2$$

is rewritten as

$$N' = (b - d)N - cN^2.$$

- $b - d$ represents the rate at which the population increases (or decreases) in the absence of competition. It is called the *intrinsic growth rate* of the population.
- c is the rate of *intraspecific* competition. The prefix *intra* refers to the fact that the competition is occurring between members of the same species, that is, within the species. [We will see later examples of *interspecific* competition, that is, between different species.]

Another (..) interpretation of the logistic equation

We have

$$N' = (b-d)N - cN^2.$$

Factor out an N:

$$N' = \big((b-d) - cN\big)N.$$

This gives us another interpretation of the logistic equation. Writing

$$\frac{N'}{N} = (b-d) - cN,$$

we have N'/N, the *per capita growth rate* of N, given by a constant, $b-d$, minus a *density dependent inhibition* factor, cN.

Equivalent equations

$$\begin{aligned} N' &= (b-d)N - cN^2 \\ &= \big((b-d) - cN\big)N \\ &= \left(r - \frac{r}{r}cN\right)N, \quad \text{with } r = b-d \\ &= rN\left(1 - \frac{c}{r}N\right) \\ &= rN\left(1 - \frac{N}{K}\right), \end{aligned}$$

with

$$\frac{c}{r} = \frac{1}{K},$$

that is, $K = r/c$.

3 ways to tackle this equation

1. The equation is separable. [explicit method]
2. The equation is a Bernoulli equation. [explicit method]
3. Use qualitative analysis.

Studying the logistic equation qualitatively

We study

$$N' = rN\left(1-\frac{N}{K}\right). \tag{ODE1}$$

For this, write

$$f(N) = rN\left(1-\frac{N}{K}\right).$$

Consider the initial value problem (IVP)

$$N' = f(N), \quad N(0) = N_0 > 0. \tag{IVP1}$$

- f is C^1 (differentiable with continuous derivative) so solutions to (IVP1) exist and are unique.

Equilibria of (ODE1) are points such that $f(N) = 0$ (so that $N' = f(N) = 0$, meaning N does not vary). So we solve $f(N) = 0$ for N. We find two points:

- $N = 0$
- $N = K$.

By uniqueness of solutions to (IVP1), solutions cannot cross the lines $N(t) = 0$ and $N(t) = K$.

There are several cases.

- $N = 0$ for some t, then $N(t) = 0$ for all $t \geq 0$, by uniqueness of solutions.
- $N \in (0, K)$, then $rN > 0$ and $N/K < 1$ so $1 - N/K > 0$, which implies that $f(N) > 0$. As a consequence, $N(t)$ increases if $N \in (0, K)$.
- $N = K$, then $rN > 0$ but $N/K = 1$ so $1 - N/K = 0$, which implies that $f(N) = 0$. As a consequence, $N(t) = K$ for all $t \geq 0$, by uniqueness of solutions.
- $N > K$, the $rN > 0$ and $N/K > 1$, implying that $1 - N/K < 0$ and in turn, $f(N) < 0$. As a consequence, $N(t)$ decreases if $N \in (K, +\infty)$.

Therefore,

Theorem

Suppose that $N_0 > 0$*. Then the solution* $N(t)$ *of* (IVP1) *is such that*

$$\lim_{t\to\infty} N(t) = K,$$

so that K *is the number of individuals that the environment can support, the carrying capacity of the environment.*

If $N_0 = 0$*, then* $N(t) = 0$ *for all* $t \geq 0$*.*

The delayed logistic equation

Consider the equation as

$$\frac{N'}{N} = (b-d) - cN,$$

that is, the per capita rate of growth of the population depends on the net growth rate $b-d$, and some density dependent inhibition cN (resulting of competition).

Suppose that instead of instantaneous inhibition, there is some delay τ between the time the inhibiting event takes place and the moment where it affects the growth rate. (For example, two individuals fight for food, and one later dies of the injuries sustained when fighting).

The delay logistic equation

In the of a time τ between inhibiting event and inhibition, the equation would be written as

$$\frac{N'}{N} = (b-d) - cN(t-\tau).$$

Using the change of variables introduced earlier, this is written

$$N'(t) = rN(t)\left(1 - \frac{N(t-\tau)}{K}\right). \tag{DDE1}$$

Such an equation is called a *delay* differential equation. It is much more complicated to study than (ODE1). In fact, some things remain unknown about (DDE1).

Delayed initial value problem

The IVP takes the form

$$\begin{aligned} N'(t) &= rN(t)\left(1 - \frac{N(t-\tau)}{K}\right), \\ N(t) &= \phi(t) \text{ for } t \in [-\tau, 0], \end{aligned} \tag{IVP2}$$

where $\phi(t)$ is some continuous function. Hence, initial conditions (called initial data in this case) must be specific on an interval, instead of being specified at a point, to guarantee existence and uniqueness of solutions.
We will not learn how to study this type of equation (this is graduate level mathematics). I will give a few results.

To find equilibria, remark that delay should not play a role, since N should be constant. Thus, equilibria are found by considering the equation with no delay, which is (ODE1).

Theorem

Suppose that $r\tau < 22/7$. Then all solutions of (IVP2) *with positive initial data $\phi(t)$ tend to K. If $r\tau > \pi/2$, then K is an unstable equilibrium and all solutions of* (IVP2) *with positive initial data $\phi(t)$ on $[-\tau, 0]$ are oscillatory.*

Note that there is a gray zone between $22/7$ and $\pi/2$.. The first part of the theorem was proved in 1945 by Wright. Although there is very strong numerical evidence that this is in fact true up to $\pi/2$, nobody has yet managed to prove it.

Discrete-time systems

So far, we have seen continuous-time models, where $t \in \mathbb{R}_+$. Another way to model natural phenomena is by using a discrete-time formalism, that is, to consider equations of the form

$$x_{t+1} = f(x_t),$$

where $t \in \mathbb{N}$ or $\mathbb{Z}$, that is, t takes values in a discrete valued (countable) set.

Time could for example be days, years, etc.

The logistic map

The logistic *map* is, for $t \geq 0$,

$$N_{t+1} = rN_t \left(1 - \frac{N_t}{K}\right). \tag{DT1}$$

To transform this into an initial value problem, we need to provide an initial condition $N_0 \geq$ for $t = 0$.