

Passage 24 A #5

■ Age structured population

Human being *Homo sapiens* has relatively long life span; they usually live up to several decades, or sometimes more than one hundred years. A population thus usually consists of people of various ages, from the young to the elder. Such a population is called “age structured” with a certain number of individuals of age 0, 1, 2, ..., etc.

Up to the previous lectures, we have ignored age structure. But its importance for demography is obvious.

An age structured population can be described by a vector with a certain number of elements; each elements represents the number of individuals of a certain age class.

For example, Japanese population by age class (5 years interval, 0-4 y old, 5-9, 10-14, ..., 85-90) in 1940 and 2014 are given as follows.

Data from <http://www.stat.go.jp/english/data/nenkan/1431-02.htm>

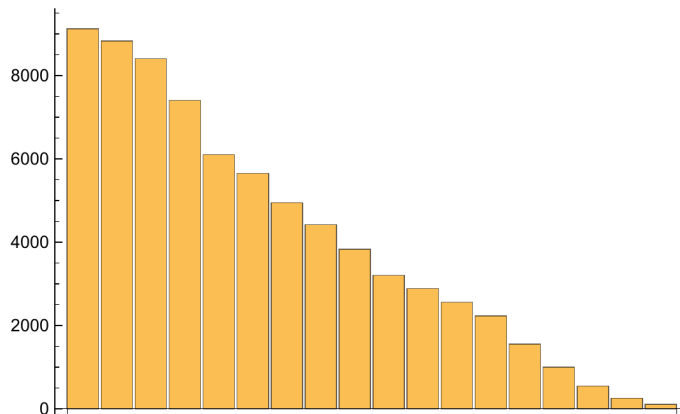
```
ageDist1940 = {9128, 8834, 8407, 7409, 6104, 5653, 4947,
               4423, 3834, 3207, 2887, 2560, 2227, 1555, 995, 546, 254, 103};
```

```
ageDist2014 = {5213, 5307, 5713, 6005, 6203, 6678, 7466,
               8670, 9793, 8608, 7791, 7654, 8980, 9154, 7928, 6269, 4869, 4779};
```

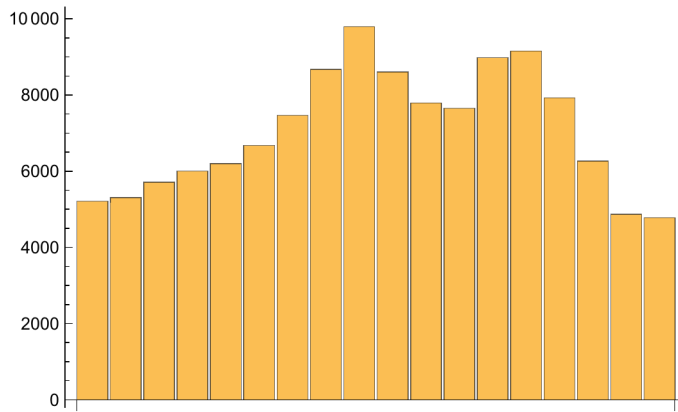
```
Length[ageDist1940]
```

```
18
```

```
BarChart[ageDist1940]
```



BarChart [ageDist2014]



We see a big difference in the age distribution in 1940 and 2014!; We had less elder people in 1940. But more elder people in 2014.

Age distribution is an important indicator of a population because various social activities such as economy, education, and social security, etc. depend on the percentage of children < 15 years old or elderly > 65 years old.

Strictly speaking, numbers of males and females in the real world can be different for each age class. But we simplify the complexity and hereafter we only focus on females by assuming that we always have the same number of males as females for all age classes.

■ How to describe an age structured population

Let $n_x(t)$ be the number of people aged x in the year t ($x = 1, 2, \dots$). Then an age structured population in year t can be represented by a vector whose elements are $n_1(t), n_2(t), n_3(t), \dots, n_w(t)$ where w is the terminal age that may be well set to be 100. Here we exclude babies aged 0 born in the year t .

We want to describe the age distribution at year $t + 1$ using that at year t . We use the following two rules to derive the equation - aging and birth.

Aging

It is obvious that an individual aged x in year t gets older by age one to be aged $x + 1$ in the next year $t + 1$. But some unlucky individuals may not survive to the next year because of an accident happened in year t . Let P_x be the probability of people aged x surviving to the next year. Then this rule of aging is given as.

$$n_x(t + 1) = P_{x-1} n_{x-1}(t) \quad \text{for } x = 2, 3, 4, \dots, w. \tag{1}$$

Birth

It is obvious that a baby is born aged 0 (age of all babies is 0). Let m_x be the number of babies born from a mother whose age is x . Then the total number of babies born in the year t is given as the sum of babies born from each age of mothers in year t .

$$m_1 n_1(t) + m_2 n_2(t) + m_3 n_3(t) + \dots + m_w n_w(t)$$

The number of age 1 in the next year t is these new babies born in the year t who have survived one year with the probability P_0 . Then, we obtain

$$n_1(t+1) = P_0 m_1 n_1(t) + P_0 m_2 n_2(t) + P_0 m_3 n_3(t) + \dots + P_0 m_w n_w(t)$$

Here, we replace $P_0 m_x$ with f_x to get

$$n_1(t+1) = f_1 n_1(t) + f_2 n_2(t) + f_3 n_3(t) + \dots + f_w n_w(t) \tag{2}$$

Leslie Model (Matrix model)

Equation (1) and (2) describe the age distribution at year $t+1$ in terms of that at year t . Although these equations look formidable, these can be readily represented by using matrix and vector notations as

$$\mathbf{n}(t+1) = \mathbf{A} \mathbf{n}(t) \tag{3}$$

where $\mathbf{n}(t)$ is the column vector $(n_1(t), n_2(t), n_3(t), \dots, n_w(t))$ and \mathbf{A} is a w by w matrix with the first row being $(f_1, f_2, f_3, \dots, f_w)$ and $(P_1, P_2, P_3, \dots, P_{w-1})$ in the subdiagonal. The matrix \mathbf{A} is called Leslie matrix after Patrick H Leslie who first used this matrix notation to describe age structured population.

Age-dependent fertility data f_x and annual survival probability P_x can be available from various sources.

Equation (3) can be easily solved if the matrix \mathbf{A} is a constant matrix, i.e., all elements are constant. The solution is

$$\mathbf{n}(t) = \mathbf{A}^t \mathbf{n}(0)$$

For human being, $w = 100$ and the matrix \mathbf{A} can be in huge dimension and calculation the power t may be tedious. But *Mathematica* can do it easily.

Age - dependent fertilities

Fertility f_x does depend on age. Females can give birth biologically between age 15 to 50. But actual age of having baby is critically affected by society they live, e.g., age of marriage, etc.

Age - dependent survival probabilities

Annual survival probability (or death probability) does depend on age. Basically it is high for infant and for elder people. Age-dependent death rates (Probability of death at each age) for Japanese women in 2014 for age 0, 1, 2, ..., 104, are available at the Ministry of Health (<http://www.mhlw.go.jp/toukei/saikin/hw/life/life14/index.html>)

```
SetDirectory["/Volumes/OSX/Users/takasu/home/情報科学科の仕事/講義/平成28年度/H28
           パッケージ/Presentation/data/"];

```

```

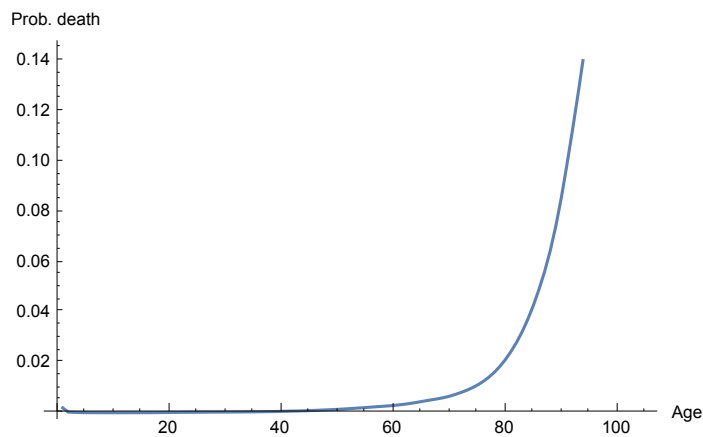
dataDeathRate = ReadList["death rate female Japan 2014.csv", Real]
{0.00198, 0.0003, 0.00021, 0.00014, 0.0001, 0.00009, 0.00008, 0.00006,
 0.00006, 0.00006, 0.00006, 0.00006, 0.00007, 0.00007, 0.00007, 0.00008,
 0.0001, 0.00012, 0.00015, 0.00017, 0.00019, 0.00021, 0.00023, 0.00024,
 0.00025, 0.00027, 0.00028, 0.00029, 0.0003, 0.00031, 0.00033, 0.00035,
 0.00036, 0.00038, 0.0004, 0.00043, 0.00046, 0.0005, 0.00054, 0.0006,
 0.00066, 0.00071, 0.00076, 0.00082, 0.0009, 0.00099, 0.00109, 0.00119,
 0.00129, 0.00139, 0.00152, 0.00166, 0.00181, 0.00196, 0.00212, 0.00228,
 0.00244, 0.0026, 0.00276, 0.00295, 0.00318, 0.00344, 0.00375, 0.0041,
 0.0045, 0.00488, 0.00525, 0.00563, 0.00606, 0.0066, 0.00727, 0.00805,
 0.00889, 0.00987, 0.01103, 0.01242, 0.01408, 0.01606, 0.01841, 0.02116,
 0.02438, 0.02802, 0.03225, 0.03715, 0.04274, 0.04906, 0.05612, 0.06436,
 0.07417, 0.08565, 0.0988, 0.11266, 0.12713, 0.14228, 0.15706, 0.17422,
 0.193, 0.2135, 0.23581, 0.26003, 0.2862, 0.31439, 0.3446, 0.37682, 0.41099}

```

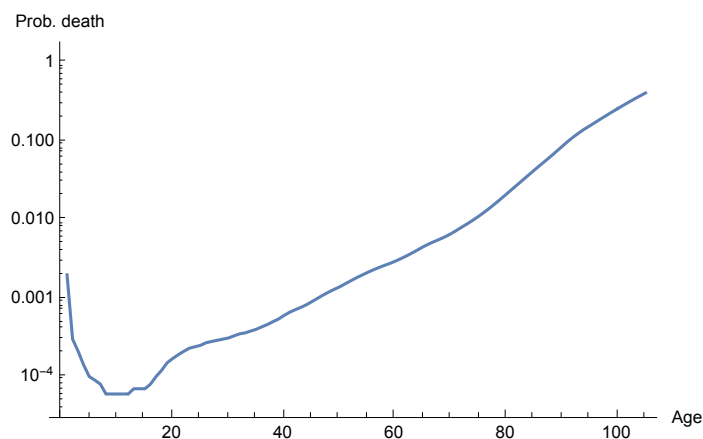
```
Length[dataDeathRate]
```

```
105
```

```
ListPlot[dataDeathRate, Joined → True, AxesLabel → {"Age", "Prob. death"}]
```



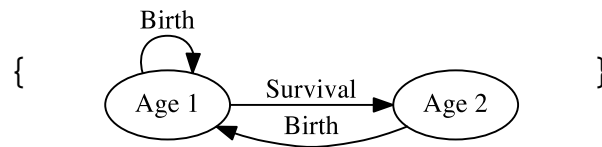
```
ListLogPlot[dataDeathRate, Joined → True, AxesLabel → {"Age", "Prob. death"}]
```



■ Example I: Two age classes

Assume a simple case that a biological organism who can live maximally two years (no more than two) gives birth and dies according to the below

model.



Then, the number of age 1 and 2, $n_1(t)$ and $n_2(t)$, follows the below equation.

```
mat = {{f1, f2}, {P1, 0}}
```

```
{{f1, f2}, {P1, 0}}
```

```
MatrixForm[mat]
```

```
( f1 f2 )
 ( P1 0 )
```

Let $f_1 = 0.5$, $f_2 = 1$, $P_1 = 0.75$. How does this organism grow in size when we start from $(n_1(0), n_2(0)) = (1, 0)$?

```
para = {f1 → 0.5, f2 → 1, P1 → 0.75}
```

```
{f1 → 0.5, f2 → 1, P1 → 0.75}
```

```
matN = mat /. para
```

```
{{0.5, 1}, {0.75, 0}}
```

```
MatrixPower[matN, 1].{1, 0}
```

```
{0.5, 0.75}
```

```
MatrixPower[matN, 2].{1, 0}
```

```
{1., 0.375}
```

```
MatrixPower[matN, 3].{1, 0}
```

```
{0.875, 0.75}
```

```
MatrixPower[matN, 4].{1, 0}
```

```
{1.1875, 0.65625}
```

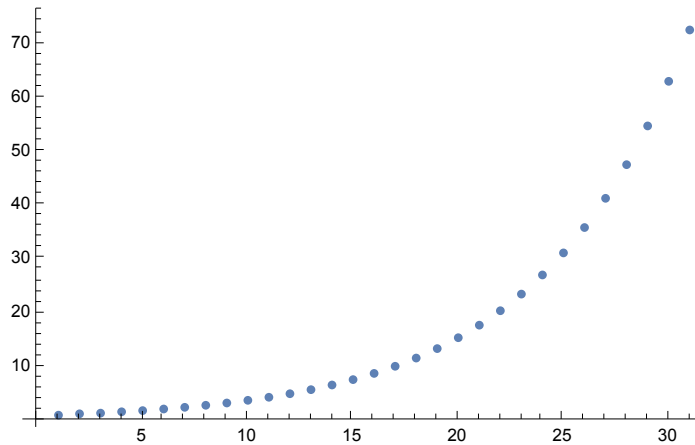
```
seqData = Table[MatrixPower[matN, t].{1, 0}, {t, 0, 30}]
```

```
{{1., 0.}, {0.5, 0.75}, {1., 0.375}, {0.875, 0.75}, {1.1875, 0.65625},
 {1.25, 0.890625}, {1.51563, 0.9375}, {1.69531, 1.13672}, {1.98438, 1.27148},
 {2.26367, 1.48828}, {2.62012, 1.69775}, {3.00781, 1.96509},
 {3.46899, 2.25586}, {3.99036, 2.60175}, {4.59692, 2.99277},
 {5.29123, 3.44769}, {6.09331, 3.96842}, {7.01508, 4.56998}, {8.07752, 5.26131},
 {9.30007, 6.05814}, {10.7082, 6.97505}, {12.3291, 8.03113}, {14.1957, 9.24685},
 {16.3447, 10.6468}, {18.8191, 12.2585}, {21.6681, 14.1143}, {24.9484, 16.2511},
 {28.7253, 18.7113}, {33.0739, 21.5439}, {38.0809, 24.8054}, {43.8459, 28.5607}}
```

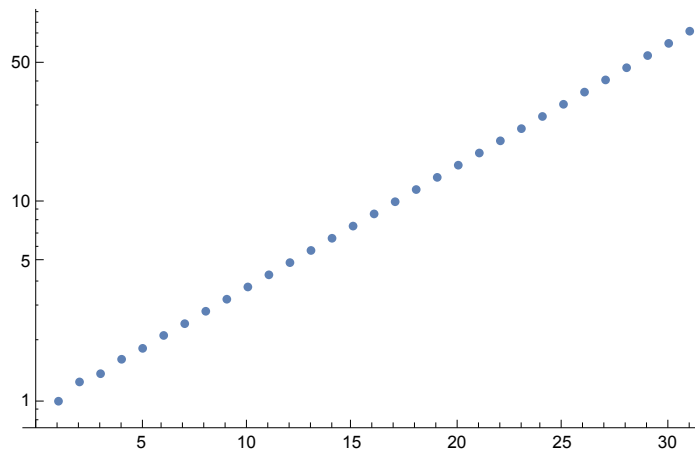
```
seqPopSize = Apply[Plus, Transpose[seqData]]
```

```
{1., 1.25, 1.375, 1.625, 1.84375, 2.14063, 2.45313, 2.83203, 3.25586,
 3.75195, 4.31787, 4.9729, 5.72485, 6.5921, 7.58969, 8.73892, 10.0617,
 11.5851, 13.3388, 15.3582, 17.6832, 20.3603, 23.4425, 26.9915,
 31.0776, 35.7824, 41.1994, 47.4365, 54.6179, 62.8863, 72.4066}
```

```
ListPlot[seqPopSize]
```



```
ListLogPlot[seqPopSize]
```



```
Eigensystem[matN]
```

```
{{1.15139, -0.651388}, {{0.837912, 0.545806}, {-0.655728, 0.754997}}}
```

Eventually, the total population increases by a factor 1.15139 (15.139% increase) and the age distribution is (0.8379, 0.5458). In other words, the population increases annually by the principal eigenvalue and the age distribution is given by the eigenvector corresponding to the principal eigenvalue.

■ Example 2 : A hypothetical scenario for human being

Assume that women start to give birth with a constant rate f during 21 to 35 years old every year. Let $f = 0.08$. A woman has roughly $0.08 (35 - 20) = 1.2$ children on average in her life.

Annual survival probabilities are assumed to be given those of Japanese women in 2014.

0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0

Eigenvalues[mat] // Chop

```
{1.0064, 0.963879 + 0.217819 i, 0.963879 - 0.217819 i, 0.806593 + 0.510536 i,
0.806593 - 0.510536 i, 0.868205 + 0.366656 i, 0.868205 - 0.366656 i,
0.678349 + 0.645733 i, 0.678349 - 0.645733 i, 0.414617 + 0.838603 i,
0.414617 - 0.838603 i, 0.566957 + 0.740127 i, 0.566957 - 0.740127 i,
0.104635 + 0.925441 i, 0.104635 - 0.925441 i, -0.215423 + 0.897477 i,
-0.215423 - 0.897477 i, -0.646252 + 0.655892 i, -0.646252 - 0.655892 i,
-0.830269 + 0.396158 i, -0.830269 - 0.396158 i, 0.264981 + 0.879556 i,
0.264981 - 0.879556 i, -0.380366 + 0.835115 i, -0.380366 - 0.835115 i,
-0.910695 + 0.087263 i, -0.910695 - 0.087263 i, -0.0639602 + 0.912583 i,
-0.0639602 - 0.912583 i, -0.507887 + 0.758962 i, -0.507887 - 0.758962 i,
-0.878363 + 0.233759 i, -0.878363 - 0.233759 i, -0.738202 + 0.527974 i,
-0.738202 - 0.527974 i, 0.7138, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0}
```

The population will eventually increases by 0.64%


```
tmp = Eigensystem[mat] // Chop
```

```
{ { { 1.0064, 0.963879 + 0.217819 i, 0.963879 - 0.217819 i,
      0.806593 + 0.510536 i, 0.806593 - 0.510536 i, 0.868205 + 0.366656 i,
      0.868205 - 0.366656 i, ... 86 ..., 0, 0, 0, 0, 0, 0, 0, 0 }, { ... 1 ... } }
```

大きい出力

表示を少なく

もっと表示

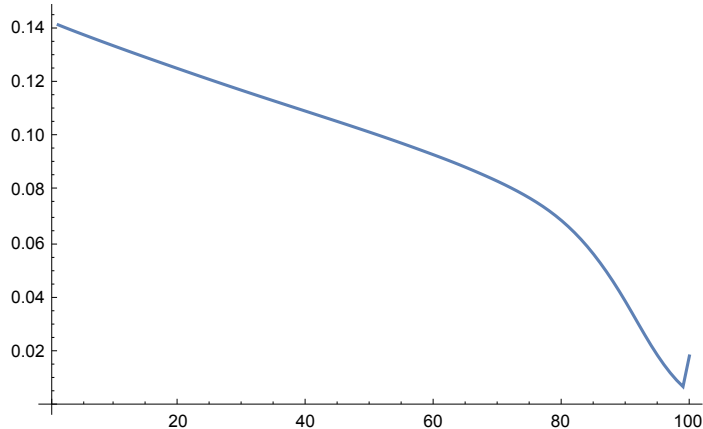
すべて表示

大きさ制限の設定...

```
ageDistStable = tmp[[2]][[1]]
```

```
{ 0.141295, 0.140355, 0.139433, 0.138527, 0.137632, 0.136745, 0.135865, 0.134993,
  0.134126, 0.133265, 0.13241, 0.13156, 0.130715, 0.129874, 0.129039, 0.128209,
  0.127381, 0.126556, 0.125732, 0.124911, 0.124093, 0.123279, 0.122467,
  0.121659, 0.120855, 0.120054, 0.119257, 0.118464, 0.117676, 0.116891,
  0.11611, 0.115331, 0.114557, 0.113785, 0.113016, 0.112249, 0.111484, 0.11072,
  0.109957, 0.109192, 0.108426, 0.10766, 0.106894, 0.106127, 0.105358,
  0.104584, 0.103806, 0.103023, 0.102236, 0.101445, 0.100646, 0.0998404,
  0.099026, 0.0982035, 0.0973722, 0.0965324, 0.0956846, 0.094829, 0.0939659,
  0.093093, 0.0922069, 0.0913054, 0.0903847, 0.0894417, 0.0884731, 0.0874815,
  0.0864689, 0.0854354, 0.0843777, 0.0832878, 0.0821566, 0.080977, 0.0797468,
  0.0784576, 0.0770989, 0.0756572, 0.0741176, 0.0724636, 0.0706772, 0.0687418,
  0.0666395, 0.0643604, 0.0618887, 0.0592106, 0.0563196, 0.053216, 0.0499101,
  0.046401, 0.0426863, 0.038782, 0.0347281, 0.0306197, 0.0265571, 0.0226337,
  0.0189575, 0.0155552, 0.0124732, 0.00974782, 0.00740182, 0.0187189 }
```

```
ListPlot[ageDistStable, Joined -> True]
```



The stable age distribution is given as the above graph.

Initial population is (1, 0, 0, ...), i.e., the population starts a youngest age class.

```
initPop = Table[0, {100}];
```

```
initPop[[1]] = 1;
```

```
initPop
```

```
{ 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
  0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
  0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
  0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0 }
```

Population distribution at t - th year. is given as $\mathbf{n}(t) = A^t \mathbf{n}(0)$

```
MatrixPower[mat, 10].initPop
```

```
{0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.998841, 0., 0., 0., 0., 0., 0., 0., 0., 0.,
  0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
  0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
  0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
  0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.}
```

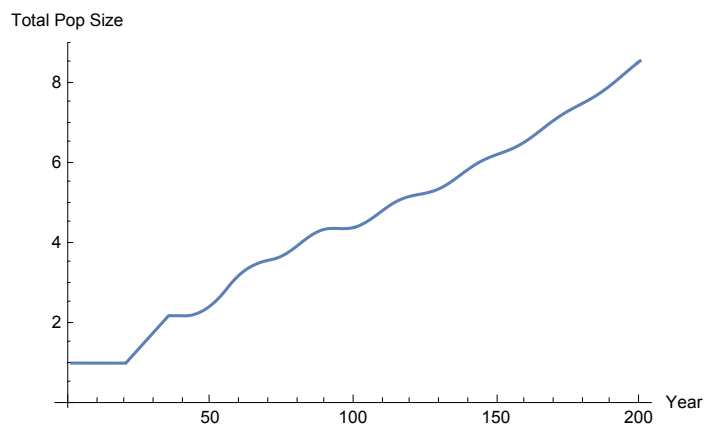
```
tEnd = 200;
```

```
seqData = Table[MatrixPower[mat, t].initPop, {t, 1, tEnd}];
```

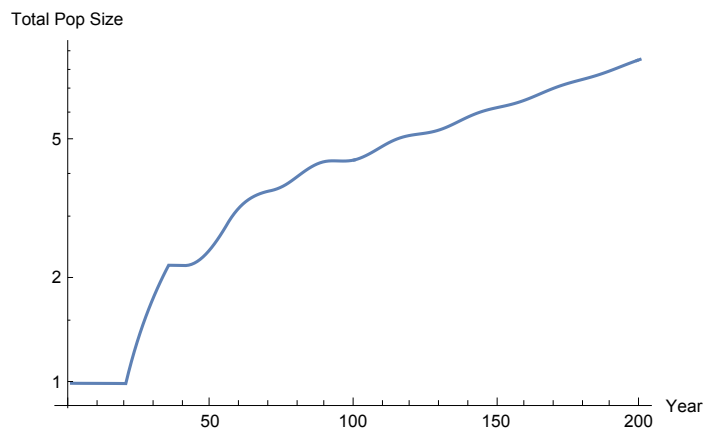
```
seqPopSize = Apply[Plus, Transpose[seqData]]
```

```
{0.9997, 0.99949, 0.99935, 0.99925, 0.99916, 0.99908, 0.99902, 0.99896, 0.998901,
  0.998841, 0.998781, 0.998711, 0.998641, 0.998571, 0.998491, 0.998391,
  0.998271, 0.998122, 0.997952, 0.997762, 1.07737, 1.15692, 1.23643, 1.3159,
  1.39531, 1.47469, 1.55403, 1.63334, 1.7126, 1.79182, 1.87098, 1.9501, 2.02917,
  2.10818, 2.18713, 2.18655, 2.18595, 2.18532, 2.18462, 2.18387, 2.18305,
  2.18855, 2.20034, 2.2184, 2.24272, 2.27328, 2.31007, 2.3531, 2.40235, 2.4578,
  2.51943, 2.58722, 2.66117, 2.74128, 2.82753, 2.91992, 3.00576, 3.08505,
  3.15778, 3.2239, 3.28338, 3.33619, 3.3828, 3.42366, 3.45932, 3.49028, 3.51705,
  3.54009, 3.55981, 3.5766, 3.59088, 3.60942, 3.6326, 3.66079, 3.69432, 3.73348,
  3.77854, 3.82826, 3.88137, 3.93661, 3.99279, 4.04866, 4.10296, 4.15459,
  4.20253, 4.24596, 4.28393, 4.31556, 4.34019, 4.35752, 4.368, 4.37219, 4.37216,
  4.37022, 4.3677, 4.36621, 4.36729, 4.37233, 4.38237, 4.39812, 4.42136,
  4.45086, 4.48569, 4.52518, 4.56875, 4.61589, 4.66614, 4.71849, 4.77191,
  4.82532, 4.87764, 4.92782, 4.97483, 5.01795, 5.05673, 5.09095, 5.1206,
  5.14591, 5.16731, 5.18541, 5.20095, 5.21491, 5.22819, 5.24175, 5.25649,
  5.27334, 5.29311, 5.31656, 5.34418, 5.37621, 5.4127, 5.45347, 5.49815,
  5.54618, 5.59683, 5.64931, 5.70266, 5.75601, 5.80861, 5.85975, 5.90887,
  5.95546, 5.99916, 6.03974, 6.07715, 6.11148, 6.14299, 6.17208, 6.19929,
  6.22527, 6.25073, 6.27636, 6.30289, 6.33097, 6.36117, 6.39396, 6.4297, 6.46859,
  6.51066, 6.55584, 6.60392, 6.65457, 6.70736, 6.76175, 6.81717, 6.87301,
  6.92868, 6.98361, 7.03729, 7.08931, 7.13936, 7.18726, 7.23295, 7.27651,
  7.31811, 7.35805, 7.39669, 7.43447, 7.47186, 7.50938, 7.54753, 7.58677,
  7.62754, 7.6702, 7.71502, 7.7622, 7.81179, 7.86378, 7.91803, 7.97431, 8.03233,
  8.09173, 8.15209, 8.21299, 8.274, 8.33473, 8.3948, 8.45392, 8.51184, 8.5684}
```

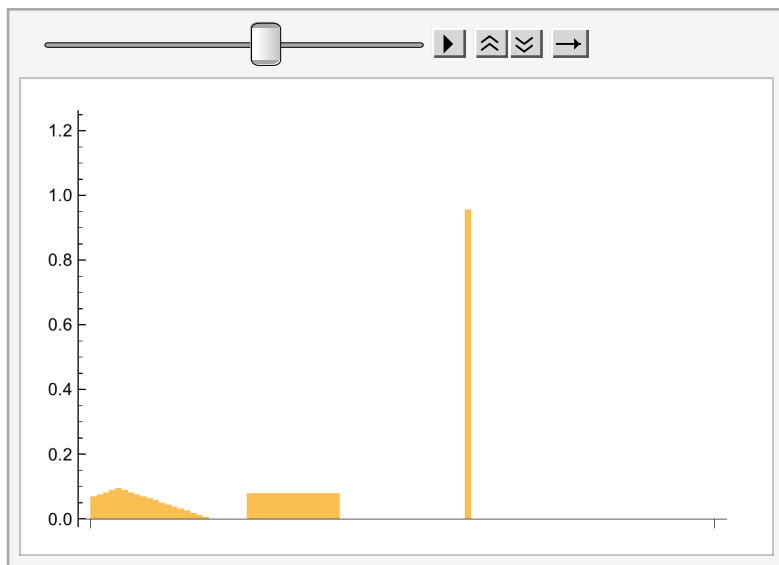
```
ListPlot[seqPopSize, Joined → True, AxesLabel → {"Year", "Total Pop Size"}]
```



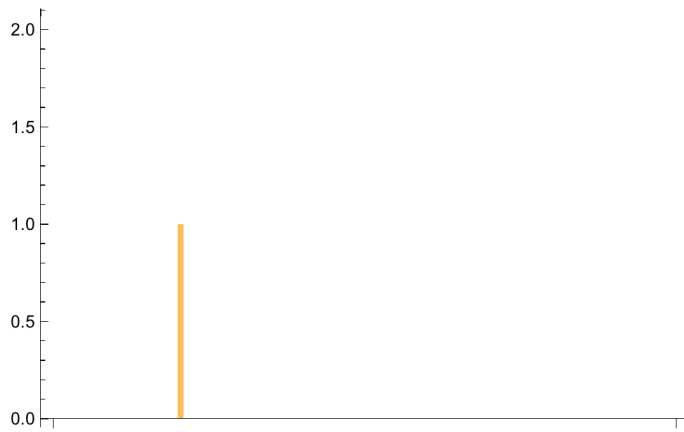
```
ListLogPlot[seqPopSize, Joined → True, AxesLabel → {"Year", "Total Pop Size"}]
```



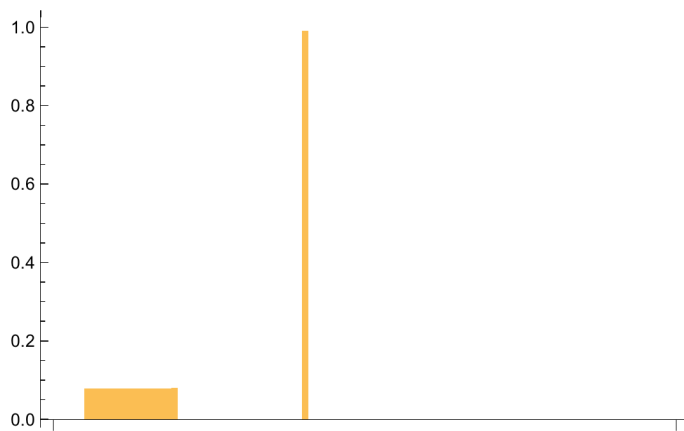
```
glist = {};
Do[
  g = BarChart[seqData[[t]], PlotRange → {0, 1.2}];
  PrintTemporary[t];
  PrintTemporary[g];
  AppendTo[glist, g], {t, 1, tEnd}
]
ListAnimate[glist]
```



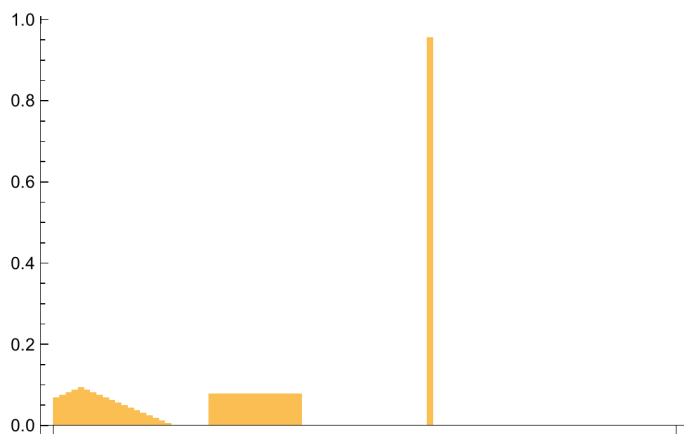
BarChart[seqData[[20]], PlotRange -> {0, 2}]



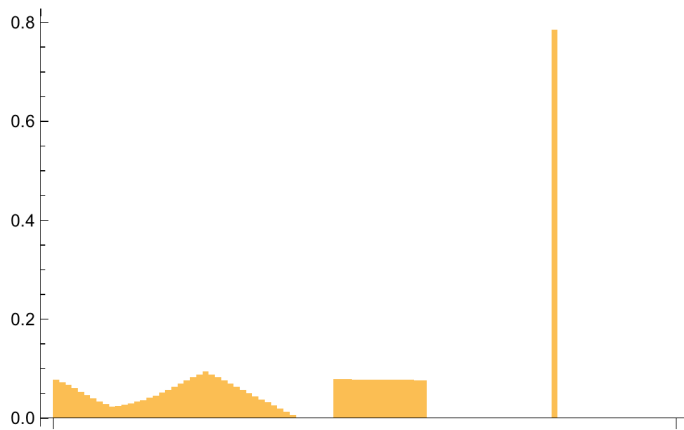
BarChart[seqData[[40]]]



BarChart[seqData[[60]]]



BarChart[seqData[[80]]]



BarChart[Last[seqData]]



Question

In the above Example 2, what happens if women start to have babies 10 years later in their lives, e.g., from 31 to 45, not from 21 to 35, while annual survival probabilities remain the same? The total number of children a woman has is the same. But the timing of start having babies is delayed.