Exam 2

Andrew O’Neill

Genetics

t_1 contains the probabilities of getting certain types of offspring based on the parent in the initial generation having a AA genotype. t_2 contains the probabilities of getting each type of offspring based on the parent from the next generation having Aa genotype.

In[61]:=
$$\mathbf{t1} = \begin{pmatrix} 1 & \frac{1}{2} & 0 \\ 0 & \frac{1}{2} & 1 \\ 0 & 0 & 0 \end{pmatrix};\ \mathbf{t2} = \begin{pmatrix} \frac{1}{2} & \frac{1}{4} & 0 \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \\ 0 & \frac{1}{4} & \frac{1}{2} \end{pmatrix};$$

b is the transition from generation 1 to generation 2. c is the transition from generation 2 to generation 1

In[62]:=
```
b = t1.t2;
c = t2.t1;
```

In[64]:=
```
eb = Eigenvalues[b]
ec = Eigenvalues[c]
```

Out[64]= $\left\{1, \frac{1}{4}, 0\right\}$

Out[65]= $\left\{1, \frac{1}{4}, 0\right\}$

We transpose the Eigenvectors since mathematica interprets them as row vectors.

In[66]:=
```
vb = Transpose[Eigenvectors[b]];
vc = Transpose[Eigenvectors[c]];
vb // MatrixForm
vc // MatrixForm
```

Out[68]//MatrixForm=
$$\begin{pmatrix} 2 & -1 & 1 \\ 1 & 1 & -2 \\ 0 & 0 & 1 \end{pmatrix}$$

Out[69]//MatrixForm=
$$\begin{pmatrix} 5 & -1 & 1 \\ 6 & 0 & -2 \\ 1 & 1 & 1 \end{pmatrix}$$

We create the d matrix by doing piecewise multiplication of the eigenvalues and the identity matrix

In[70]:=
```
db = IdentityMatrix[3] eb;
dc = IdentityMatrix[3] ec;
db // MatrixForm
dc // MatrixForm
```

Out[72]//MatrixForm=

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{4} & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

Out[73]//MatrixForm=

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{4} & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

Take the limit as n approaches infinity of each diagonalized matrix to find our equations.

In[74]:=
```
Limit[vb.db^n.Inverse[vb], n → ∞] // MatrixForm
Limit[vc.dc^n.Inverse[vc], n → ∞] // MatrixForm
```

Out[74]//MatrixForm=

$$\begin{pmatrix} \frac{2}{3} & \frac{2}{3} & \frac{2}{3} \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ 0 & 0 & 0 \end{pmatrix}$$

Out[75]//MatrixForm=

$$\begin{pmatrix} \frac{5}{12} & \frac{5}{12} & \frac{5}{12} \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \\ \frac{1}{12} & \frac{1}{12} & \frac{1}{12} \end{pmatrix}$$

For the even cases we have

$$a_{2n} \to \frac{2}{3} a_0 + \frac{2}{3} b_0 + \frac{2}{3} c_0$$

$$b_{2n} \to \frac{1}{3} a_0 + \frac{1}{3} b_0 + \frac{1}{3} c_0$$

$$c_{2n} \to 0$$

and in the odd cases we have

$$a_{2n+1} \to \frac{5}{12} a_1 + \frac{5}{12} b_1 + \frac{5}{12} c_1$$

$$b_{2n+1} \to \frac{1}{2} a_1 + \frac{1}{2} b_1 + \frac{1}{2} c_1$$

$$c_{2n+1} \to \frac{1}{12} a_1 + \frac{1}{12} b_1 + \frac{1}{12} c_1$$

Age - Specific Population Growth T2

- **A**

Pick random values for the elements of a, p, and b. Then contruct the L_8 matrix.

In[76]:=
```
a = RandomReal[{1, 5}];
p = RandomReal[];
b = RandomReal[];
```

$$L8 = \begin{pmatrix} a & a\,p & a\,p^2 & a\,p^3 & a\,p^4 & a\,p^5 & a\,p^6 & a\,p^7 \\ b & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & b & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & b & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & b & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & b & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & b & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & b & 0 \end{pmatrix};$$

In[80]:=
```
evals = Eigenvalues[L8]
```

Out[80]= {2.16241, 0.315209 + 0.329433 i, 0.315209 − 0.329433 i, −0.0113014 + 0.446337 i, −0.0113014 − 0.446337 i, −0.315829 + 0.306267 i, −0.315829 − 0.306267 i, −0.437723}

In[81]:=
```
a + b * p
```

Out[81]= 2.16241

In[82]:=
```
λ = evals[[1]]
```

Out[82]= 2.16241

B

In[83]:=
```
IN = IdentityMatrix[8] λ;
```

In[84]:=
```
Det[IN - L8]
```

Out[84]= 1.08477×10^{-12}

In[85]:= $\lambda^8 - a\left(\frac{\lambda^8 - (b\,p)^8}{\lambda - b\,p}\right)$

Out[85]= 1.08002×10^{-12}

In[86]:= $\lambda^9 - (a + b\,p)\,\lambda^8 + a\,(b\,p)^8$

Out[86]= 1.90972×10^{-12}

This is practically zero.

C

From the last part we know that
$\lambda^{n+1} - (a + \text{bp})\,\lambda^n + a(\text{bp})^n = 0.$
As $n \to \infty$ we know that $(\text{bp})^n \to 0$ since both b and p are less than 1. That means that
$\lambda^{n+1} - (a + \text{bp})\,\lambda^n = 0.$
We can factor out a λ^n. This gives us
$\lambda - (a + \text{bp}) = 0.$
Therefore
$\lambda = (a + \text{bp})$

Age - Specific Population Growth T3

In[87]:=
$$L = \begin{pmatrix} 0 & 0 & \frac{1}{2} & \frac{4}{5} & \frac{3}{10} & 0 \\ \frac{4}{5} & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{9}{10} & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{9}{10} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{4}{5} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{3}{10} & 0 \end{pmatrix};$$

```
x0 = {50, 40, 30, 20, 10, 5};
```

- **A**

In[89]:=
$$R = 0 + 0 * \frac{4}{5} + \frac{1}{2}\,\frac{4}{5}\,\frac{9}{10} + \frac{4}{5}\,\frac{4}{5}\,\frac{9}{10}\,\frac{9}{10} + \frac{3}{10}\,\frac{4}{5}\,\frac{9}{10}\,\frac{9}{10}\,\frac{4}{5} + 0\;\frac{4}{5}\,\frac{9}{10}\,\frac{9}{10}\,\frac{4}{5}\,\frac{3}{10}$$

Out[89]=
$$\frac{3231}{3125}$$

- **B**

```
In[90]:= l100 = MatrixPower[L, 100].x0 // N
         l101 = MatrixPower[L, 101].x0 // N
         l101 / l100
Out[90]= {105.385, 83.5712, 74.5567, 66.5146, 52.7467, 15.6857}
Out[91]= {106.314, 84.308, 75.2141, 67.1011, 53.2117, 15.824}
Out[92]= {1.00882, 1.00882, 1.00882, 1.00882, 1.00882, 1.00882}
```

- **C**

The eigenvalue in our result is equal to the proportion of consecutive values. So the proportion of L101 = L100 * λ. The Eigenvector is a scalar of the $L^n\,x(0)$ vector. $L^n\,x(0) = a\,v$ where a is a constant and v is the eigenvector.

```
In[93]:= y = Part[Eigenvalues[L], 1] // N
Out[93]= 1.00882
In[94]:= P = Part[Eigenvectors[L], 1] // N
Out[94]= {6.71853, 5.32785, 4.75316, 4.24046, 3.36272, 1.}
In[95]:= P * 15.824
Out[95]= {106.314, 84.308, 75.214, 67.101, 53.2117, 15.824}
```

- **D**

Solve net reproduction rate scaled by r is equal to one. This gives when the reproduction rate is constant. Anything less that that will be negative.

```
In[96]:= Solve[R * r == 1, r]
```

Out[96]=
$$\left\{\left\{r \to \frac{3125}{3231}\right\}\right\}$$

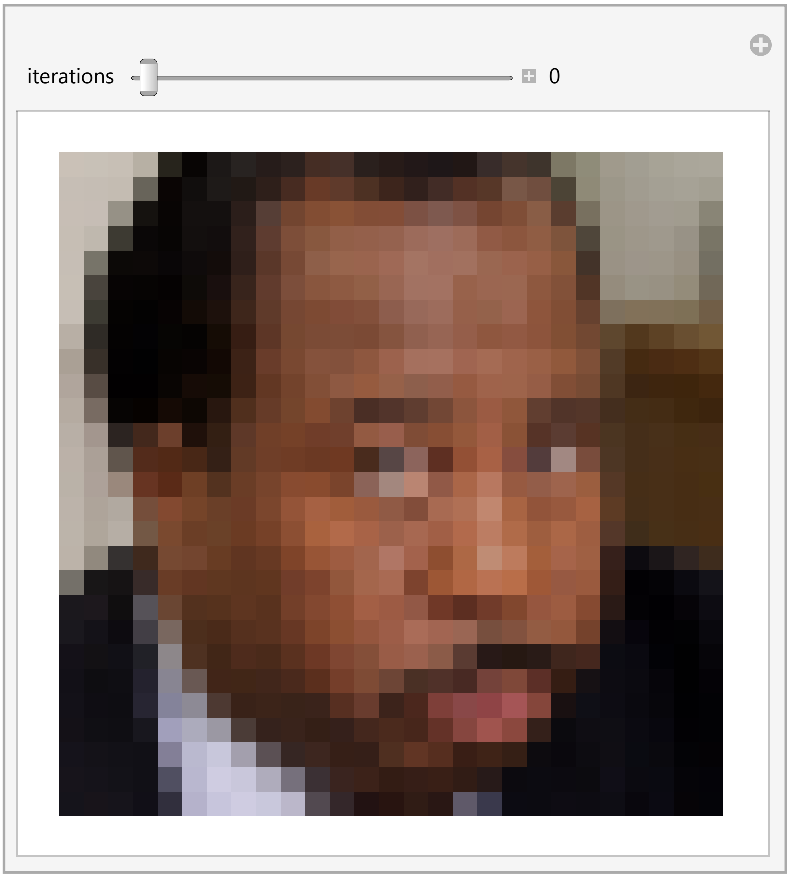
So the population will decrease if r is between 0 and .96 (3125/3231). r is the reciprocal of the net reproduction rate.

Arnold’s Cat Map

In[97]:=
```
data = Import["Downloads/stanley.bmp", "Data"];
(* This imports the picture from my desktop and
 saves the pixel color data as a numerical values. *)
catMap = Compile[{{pic, _Integer, 3}},
  Table[pic[[Mod[x + y, 27, 1], Mod[x + 2 y, 27, 1]]], {x, 27}, {y, 27}]];
stanley = 255 - data; (* The picture colors are inverted otherwise *)
```

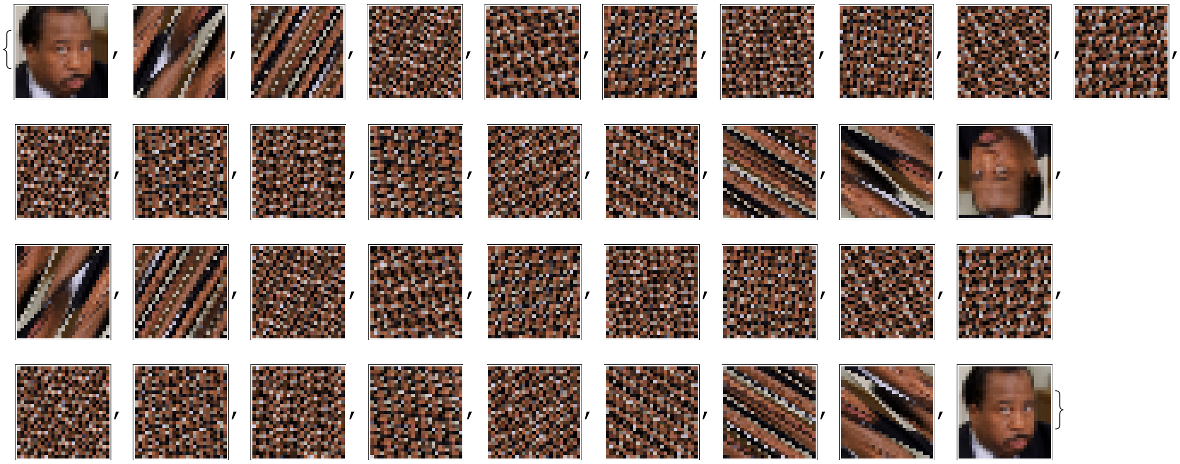
In[99]:=
```
Manipulate[ArrayPlot[Nest[catMap, stanley, iter], Frame → False],
 {{iter, 0, "iterations"}, 0, 36, 1, Appearance → "Labeled"}, SaveDefinitions → True]
```

Out[99]=



In[100]:=
```
plt[x_] := ArrayPlot[x, ImageSize → {50, 50}]
Table[plt[Nest[catMap, stanley, i]] // MatrixForm, {i, 0, 36}]
```

Out[101]=



At iteration 18 the picture is inverted and shifted up by 1. At iteration 36 the picture is back to its original state.