

Intro Math Problem Solving

November 14

Unequal Probability Simulation

A Transition Model

Going to California

Hawaiian Population Movement

Matrix-Vector Multiplication

Reference

Chapter 7, Section 1 of our textbook “Insight Through Computing” discusses the idea of a **transition matrix**, that represents how a vector of data changes from one step to the next, under a linear transformation.

.

Unequal Probability Simulation



Random, Equally Likely

Often we are faced with the necessity of simulating a random event which involves choosing one of several possible outcomes:

Flip a coin, roll a die, or select a card from a deck.

We usually assume that the outcomes are equally likely, and so we have gotten used to simulating this process, using the MATLAB functions `rand()` or `randi()`.

Flipping a Coin

To simulate flipping a coin, we can call `rand()`:

```
r = rand ();  
if ( r < 0.5 )  
  toss = 0;  <- for “tails”  
else  
  toss = 1;  <- for “heads”  
end
```

or we could call `randi()`, with 0 = ‘T’ and 1 = ‘H’:

```
toss = randi ( [ 0, 1 ] );
```

Simulating the roll of a die

When simulating the roll of a die, we would much prefer to use `randi()` because a single command gives us a random value between 1 and 6:

```
r = randi ( [ 1, 6 ] );
```

However, we can also use `rand()`. To do so, we will need many more commands.

We write two versions. The second version is simplified because we are checking the results in ascending order.

Rolling a die with rand()

```
r = rand ( );
if ( 0 <= r && r < 1/6 )
    d = 1;
elseif ( 1/6 <= r && r < 2/6 )
    d = 2;
elseif ( 2/6 <= r && r < 3/6 )
    d = 3;
elseif ( 3/6 <= r && r < 4/6 )
    d = 4;
elseif ( 4/6 <= r && r < 5/6 )
    d = 5;
elseif ( 5/6 <= r && r <= 1.0 )
    d = 6;
end
```

```
r = rand ( );
if ( r < 1/6 )
    d = 1;
elseif ( r < 2/6 )
    d = 2;
elseif ( r < 3/6 )
    d = 3;
elseif ( r < 4/6 )
    d = 4;
elseif ( r < 5/6 )
    d = 5;
else
    d = 6;
end
```

Unequal Likelihoods

If our coin is biased, or our die is loaded, or the deck includes 3 copies of the ace of spades, then some outcomes are more likely than others, and we need to change our model.

If a coin has been tampered with so that it comes up heads 60% of the time, our simulation needs to return a value of “H” 60% of the time.

We can no longer use `randi()`, but we can adjust our `rand()` approach to take care of the fact that the outcomes are no longer equally likely:

40% Tails, 60% Heads

```
r = rand ( );  
if ( r < 0.40 )  
    toss = 0;  
else  
    toss = 1;  
end
```

A Loaded Die

Suppose that we observe a die rolled $M = 1000$ times, and tabulate the frequencies F of the outcomes:

1	2	3	4	5	6
F: 120	130	140	160	200	250

Our estimated probabilities P are computed by dividing each frequency by the total number of observations $M = \text{sum}(F)$:

1	2	3	4	5	6
P: 0.120	0.130	0.140	0.160	0.200	0.250

Since the probabilities are unequal, how can we simulate this behavior?

Cumulative Probability

	1	2	3	4	5	6
P:	0.120	0.130	0.140	0.160	0.200	0.250

If we pick a random value r , it makes sense that if r is less than 0.120, we have outcome “1”.

But how do we get outcome 2? We need to give it a likelihood of 0.130. Now if r falls between 0.120 and $0.120 + 0.130 = 0.250$, the range $[0.120, 0.250]$ has width 0.130, which will give us the correct behavior.

Cumulative Probability

	1	2	3	4	5	6
P:	0.120	0.130	0.140	0.160	0.200	0.250

We need to divide the interval $[0.0, 1.0]$ into segments whose lengths are the probabilities of the outcomes. The easiest way to do this is to add the probabilities up from left to right to get cumulative probabilities C:

Outcome 1: $0.000 \leq r < 0.120$

Outcome 2: $0.120 \leq r < 0.120 + 0.130 = 0.250$

Outcome 3: $0.250 \leq r < 0.120 + 0.130 + 0.140 = 0.390$

Outcome 4: $0.390 \leq r < 0.120 + 0.130 + 0.140 + 0.160 = 0.550$

Outcome 5: $0.550 \leq r < 0.120 + 0.130 + 0.140 + 0.160 + 0.200 = 0.750$

Outcome 6: $0.750 \leq r < 0.120 + 0.130 + 0.140 + 0.160 + 0.200 + 0.250 = 1.000$

	1	2	3	4	5	6
C:	0.120	0.250	0.390	0.550	0.750	1.000

Computing Cumulative Probabilities

Can we do this for any problem?

Suppose we are given P , the vector of probabilities of each outcome, and we need to compute C , the cumulative probabilities. One way to do that is with a for loop:

```
c = zeros ( 1, n );  
s = 0.0;  
for i = 1 : n  
    s = s + p(i);  
    c(i) = s;  
end
```

MATLAB can also do this for you in a single command:

```
c = cumsum ( p );
```

Simulating the Loaded Die

```
r = rand ( );
if ( 0.0 <= r && r < C(1) )
    d = 1;
elseif ( C(1) <= r && r < C(2) )
    d = 2;
elseif ( C(2) <= r && r < C(3) )
    d = 3;
elseif ( C(3) <= r && r < C(4) )
    d = 4;
elseif ( C(4) <= r && r < C(5) )
    d = 5;
elseif ( C(5) <= r && r <= C(6) )
    d = 6;
end
```

```
r = rand ( );
if ( r < C(1) )
    d = 1;
elseif ( r < C(2) )
    d = 2;
elseif ( r < C(3) )
    d = 3;
elseif ( r < C(4) )
    d = 4;
elseif ( r < C(5) )
    d = 5;
else
    d = 6;
end
```

Simulating the Loaded Die

Since we have the cumulative probabilities in a vector, we can use a FOR loop to do our simulation, which is simpler.

```
r = rand ( );  
for i = 1 : n  
    if ( r < c(i) )  
        outcome = i;  
        break;           % <- Why do we need to break?  
    end  
end  
end
```

Example: A Loaded Die

I observe the rolling of a die $M=700$ times, getting the following frequencies of outcomes:

	1	2	3	4	5	6
F:	150	75	10	75	180	210

Probabilities, $P = F/M$:

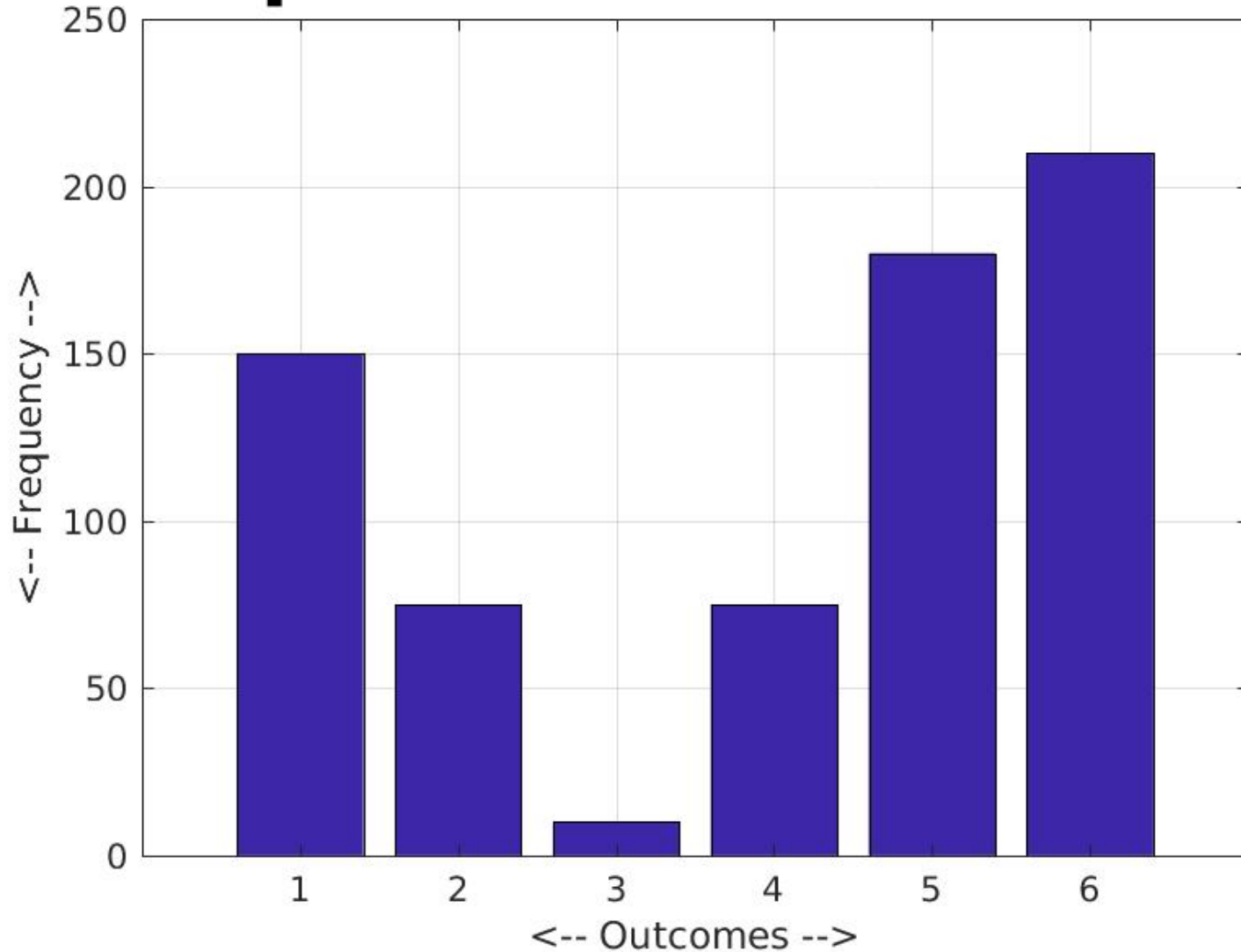
P:	0.21	0.11	0.01	0.11	0.26	0.30
----	------	------	------	------	------	------

Cumulative probabilities, $C = \text{cumsum}(P)$:

C:	0.21	0.32	0.33	0.44	0.70	1.00
----	------	------	------	------	------	------

700 Observations

Frequencies for loaded die



Simulating the Loaded Die

```
m = 700;
n = 6;
bins = zeros ( 1, n );

for j = 1 : m
    r = rand ( );

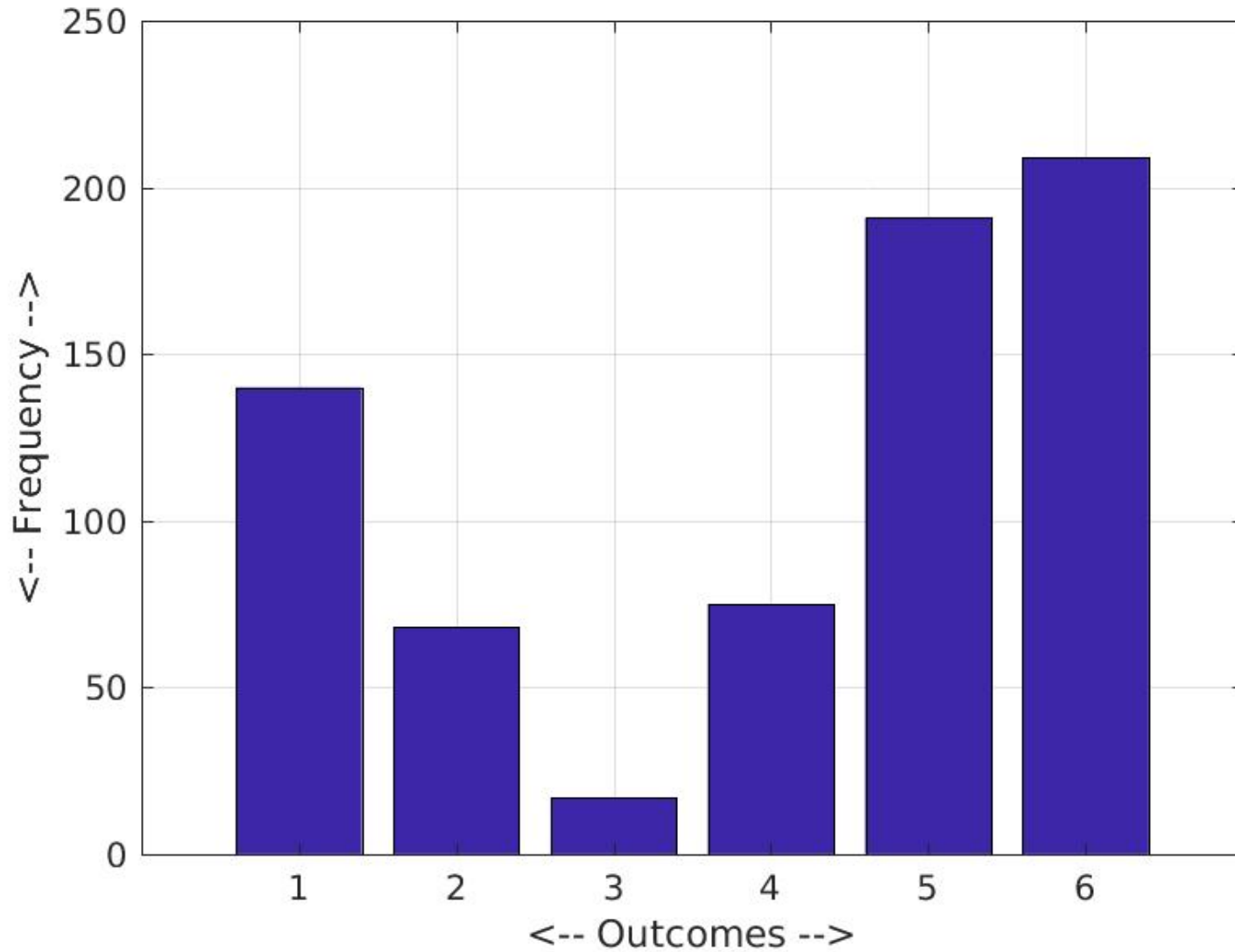
    for i = 1 : n
        if ( r < c(i) )
            bins(i) = bins(i) + 1;
            break;
        end
    end

end

bar ( 1:n, bins )
```

700 Simulations

Simulate a loaded die



Unequal Probability Summary

Given M observations, with N outcomes, create a vector $F(1:N)$ which counts the frequency of each outcome.

Compute probabilities: $P = F / M$;

Compute cumulative probabilities: $C = \text{cumsum}(P)$;

To simulate an outcome, find the first $C(I)$ such that $\text{rand}() < C(I)$, and return outcome I .

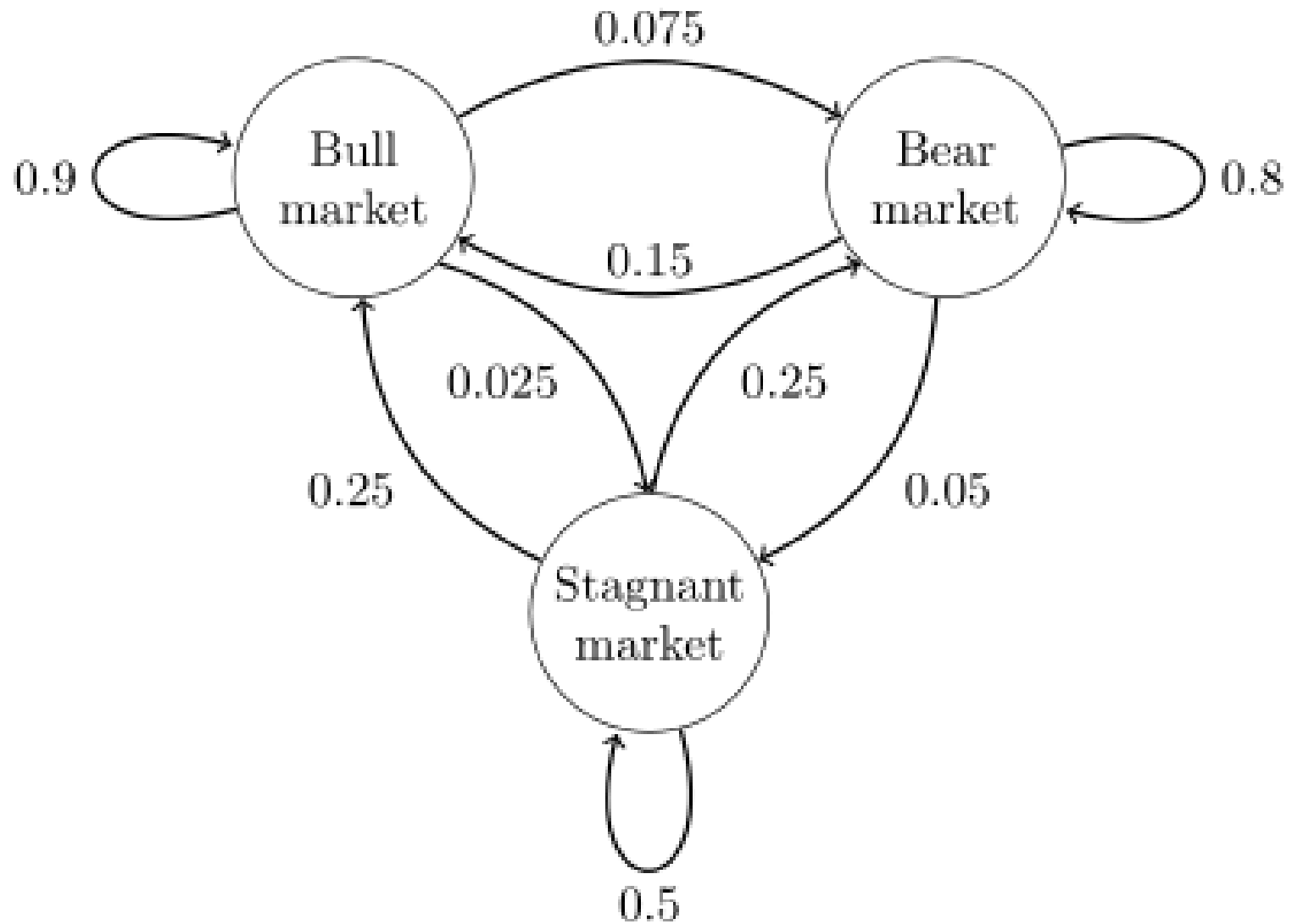
To test, create a $\text{BIN}(1:N)$ vector, simulate M outcomes, and store results in BIN . Then compare two bar plots:

```
bar ( 1:n, f )      <- original observed data
```

```
bar ( 1:n, bin )   <- simulated data
```

You expect the two plots to be similar (but not identical!)

A Transition Model



System / States / Transitions

Some systems can be thought of as having a set of possible states. The system is always in some particular state.

At regular time intervals, the system can move (or transition) from one state to another, or stay where it is.

Each possible change has a known likelihood, known as the **transition probability**.

The Transition Idea

An old proverb suggests that the best guess for tomorrow's weather is that it will be the same as today's.

Let us simplify each day's weather into the three states **sunny**, **cloudy** or **rainy**.

Even if we don't understand weather, we can make a simple transition model by recording the weather every day, and noticing the likelihood of each possible transition.

Transition Records

Here is a calendar of 25 days of weather, recorded as sunny, cloudy, or rainy:

C-S-S-C-R-R-C-C-R-C-R-C-R-S-S-S-C-R-R-S-S-C-C-S-R

This gives us 24 daily transitions, and we can tabulate their frequencies:

S->S = 4 C->S = 2 R->S = 2

S->C = 3 C->C = 2 R->C = 3

S->R = 1 C->R = 5 R->R = 2

Transition Probabilities

From these daily transition frequencies:

$$S \rightarrow S = 4 \quad C \rightarrow S = 2 \quad R \rightarrow S = 2$$

$$S \rightarrow C = 3 \quad C \rightarrow C = 2 \quad R \rightarrow C = 3$$

$$S \rightarrow R = 1 \quad C \rightarrow R = 5 \quad R \rightarrow R = 2$$

$$8 \text{ total} \quad 9 \text{ total} \quad 7 \text{ total}$$

we can give us a 3x3 matrix of transition **probabilities**:

$$S \rightarrow S = 4/8 \quad C \rightarrow S = 2/9 \quad R \rightarrow S = 2/7$$

$$S \rightarrow C = 3/8 \quad C \rightarrow C = 2/9 \quad R \rightarrow C = 3/7$$

$$S \rightarrow R = 1/8 \quad C \rightarrow R = 5/9 \quad R \rightarrow R = 2/7$$

Simulation with rand()

If yesterday was sunny, can we simulate what will happen today?

$$S \rightarrow S = 4/8$$

$$S \rightarrow C = 3/8$$

$$S \rightarrow R = 1/8$$

Let R be the value of $\text{rand}()$, and work with the cumulative probabilities:

If yesterday was sunny then:

if	$0 < R \leq 4/8:$	sunny today
elseif	$4/8 < R \leq 4/8 + 3/8:$	cloudy today
elseif	$4/8 + 3/8 < R < 4/8 + 3/8 + 1/8:$	rainy today

Similarly, we can simulate the followup to a cloudy or rainy day.

Weather Model

```
function today = weather_today ( yesterday )
r = rand ( )
if ( yesterday == 'S' )
    if ( r < 4.0 / 8.0 )
        today = 'S';
    elseif ( r < 7.0 / 8.0 )
        today = 'C';
    else
        today = 'R';
    end
elseif ( yesterday == 'C' )
    if ( r < 2.0 / 9.0 )
        today = 'S';
    elseif ( r < 4.0 / 9.0 )
        today = 'C';
    else
        today = 'R';
    end
elseif ( yesterday == 'R' )
    if ( r < 2.0 / 7.0 )
        today = 'S';
    elseif ( r < 5.0 / 7.0 )
        today = 'C';
    else
        today = 'R';
    end
end
end
```

Simulating More Weather

If we believe our transition probabilities are reasonable, we can simulate more weather, by noting yesterday's weather and choosing today's weather based on the probabilities;

```
for i = 0 : 50
    if ( i == 0 )
        today = 'S';
    else
        today = weather_today ( today );
    end
    fprintf ( '%c', today );
end
fprintf ( '\n' );
```

SSCCRSCCRRRCSSRSCRRSSSRSCRRSSSCRRSSSSCRCSCRCRCSSSSS

Transition Matrix

The table of probabilities is known as the transition probability matrix P .

Entry $P(i,j)$ of the matrix records the probability that we will have state i today, given that yesterday was state j .

Replacing fractions with decimals, our weather transition matrix is:

		Yesterday		
		S	C	R
Today	S:	$4/8$	$2/9$	$2/7$
	C:	$3/8$	$2/9$	$3/7$
	R:	$1/8$	$5/9$	$2/7$

Sum:		1	1	1

Every column sums to 1, because whatever happened yesterday, **something** must happen today!

Cumulative Probability Matrix

P stores the probabilities,

		Yesterday		
		S	C	R
Today	S:	4/8	2/9	2/7
	C:	3/8	2/9	3/7
	R:	1/8	5/9	2/7

The cumulative probability matrix C comes from the cumsum() command:

$C = \text{cumsum}(P)$;

Today	S:	4/8	2/9	2/7	=	0.50	0.22	0.29
	C:	7/8	4/9	5/7		0.87	0.44	0.57
	R:	8/8	9/9	7/7		1.00	1.00	1.00

Simulating with a Matrix

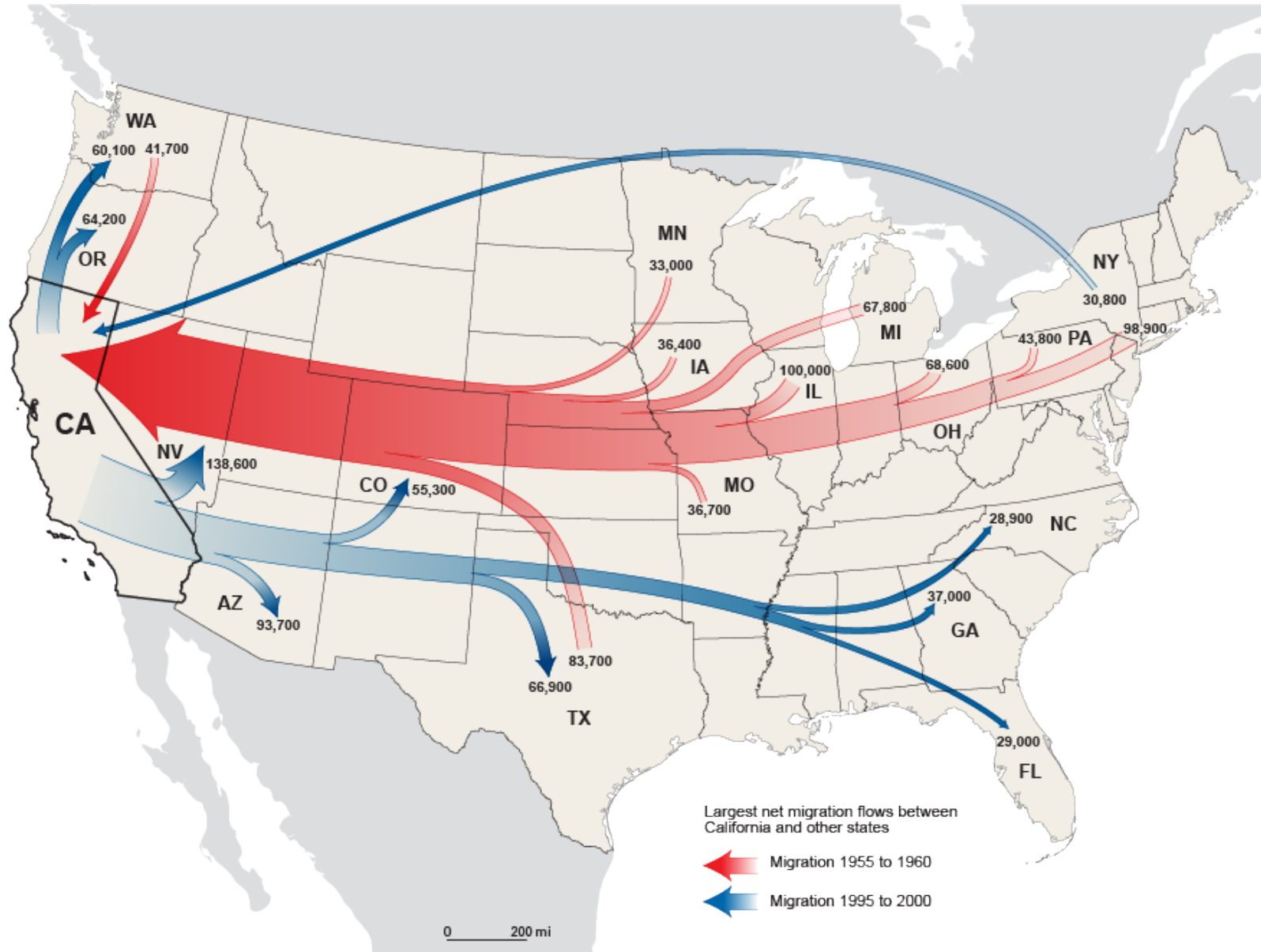
```
P = [ 4/8, 2/9, 2/7; ...  
      3/8, 2/9, 3/7; ...  
      1/8, 5/9, 2/7 ];
```

```
C = cumsum ( P );
```

```
r = rand ( );
```

```
for i = 1 : 3  
    if ( r < C(i,yesterday) )  
        today = i;  
        break;  
    end  
end
```

Going to California



Modeling Population Changes

For a while in the 1960's, the following statement was approximately true:

Every year, 30% of the population of California leaves the state, and every year, 10% of the population of the other states moves to California.

- 1) The statement sounds nonsensical. Can we write down some equations that give us numbers we can think about?
- 2) If 30% move out, and 10% move in, does this mean California is gradually going to have no population at all?
- 3) If this behavior lasts long enough, does the population curve of California look chaotic, go towards infinity, become negative, or oscillatory, or does it settle down?

One Person's Behavior

Suppose we model one person's behavior during this time, and assume that in 1960 they are living in California. Then there's a 30% chance they move out in 1961.

In 1962, if they are still in California, there's a 30% chance they move out then; but if they are outside of California, there's a 10% chance they move in.

We could simulate the location of such a person for 20 years if we wish.

Model EVERYBODY

Suppose that in 1960, California's population was 16 million, and the remaining US population was 164 million.

Then in 1961, our transition data suggests:

- * 30% of 16 million people moved OUT of CA.
- * 10% of 164 million people moved INTO CA.

In fact, we can track the CA and US populations from year to year, if we believe our model.

california.m

```
m = 21;
ca = zeros ( 1, m );
us = zeros ( 1, m );

for i = 1 : m
    year = 1959 + i;

    if ( year == 1960 )
        ca(i) = 160000000;
        us(i) = 1640000000;
    else
        [ ca(i), us(i) ] = california_update ( ca(i-1), us(i-1) );
    end
end

end
```

california_update.m

```
function [ ca, us ] = california_update ( ca, us )
```

```
    ca_old = ca;
```

```
    us_old = us;
```

```
    us = 0.90 * us_old + 0.30 * ca_old;
```

```
    ca = 0.10 * us_old + 0.70 * ca_old;
```

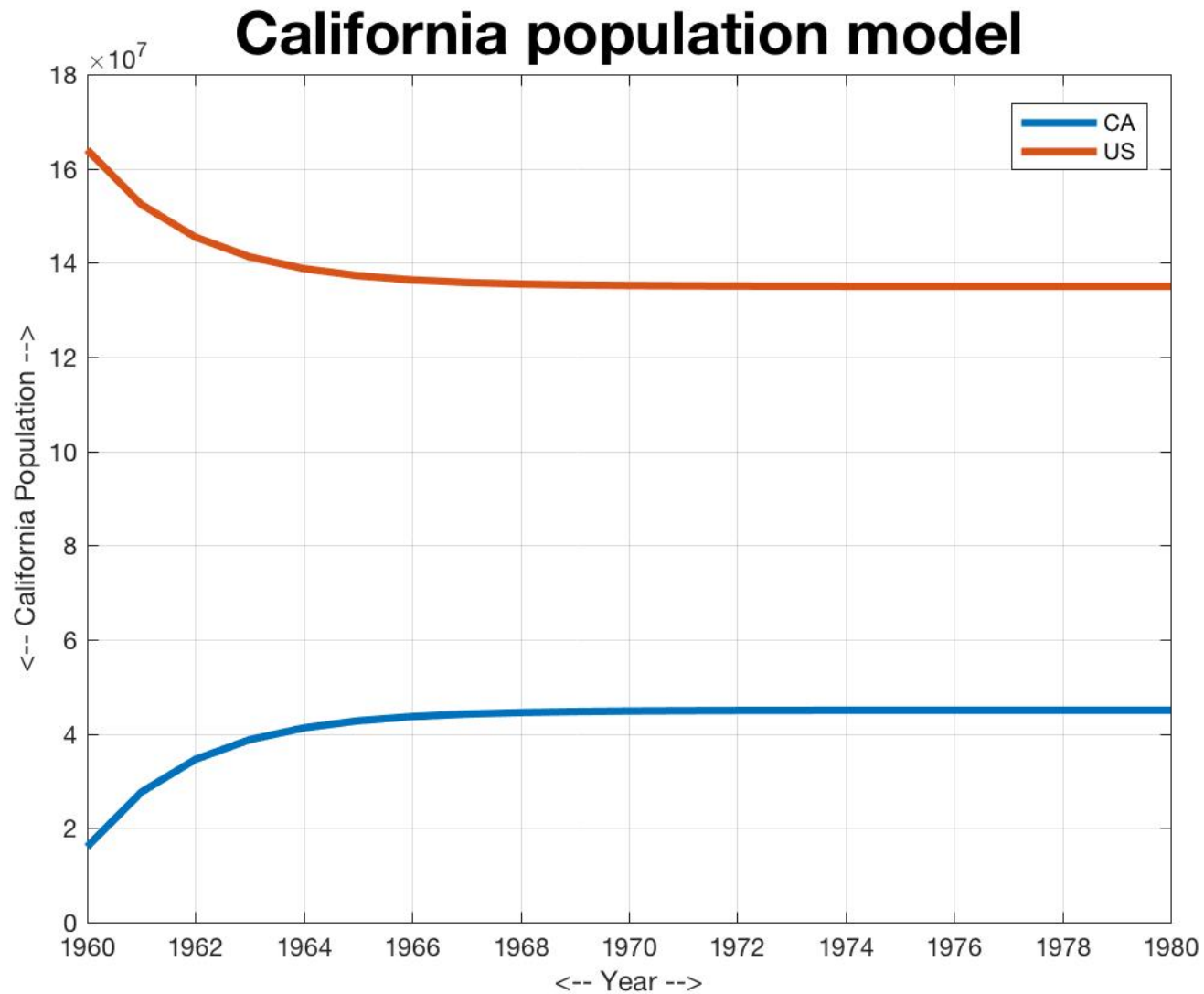
```
    return
```

```
end
```

1960 to 1970

Year	CA Pop	US pop	Total
1960	16000000	164000000	180000000
1961	27600000	152400000	180000000
1962	34560000	145440000	180000000
1963	38736000	141264000	180000000
1964	41241600	138758400	180000000
1965	42744960	137255040	180000000
1966	43646976	136353024	180000000
1967	44188186	135811814	180000000
1968	44512911	135487089	180000000
1969	44707747	135292253	180000000
1970	44824648	135175352	180000000

Reaching an Equilibrium?



There is a Natural Balance Point

The population data seems to be driving towards values of 45 million for California and 135 million for the rest of the US.

If we plug these values into our formula:

$$us = 0.90 * us_old + 0.30 * ca_old;$$

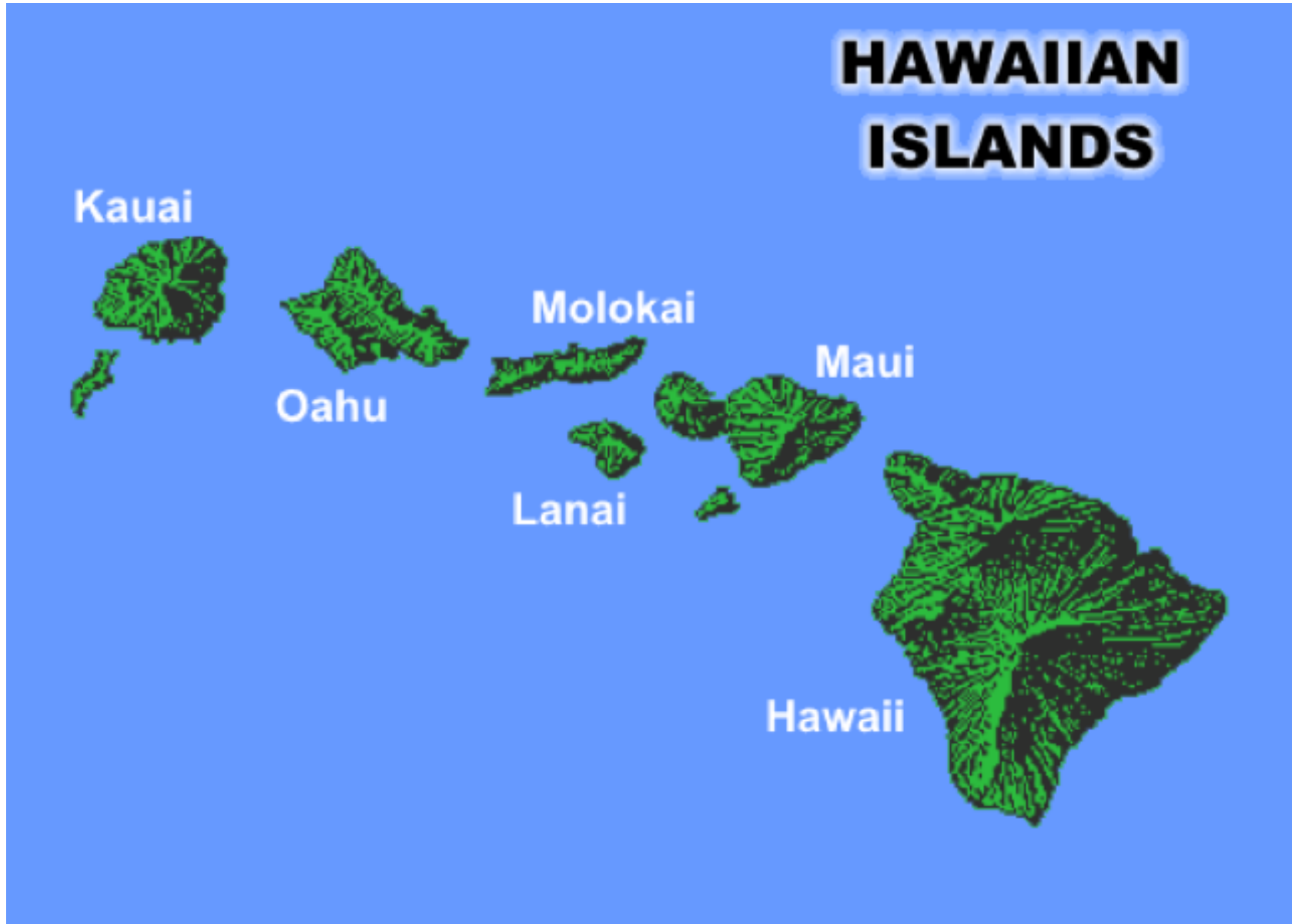
$$ca = 0.10 * us_old + 0.70 * ca_old;$$

the new values are the same as the old:

$$135m = 0.90 * 135m + 0.30 * 45m$$

$$45m = 0.10 * 135m + 0.70 * 45m$$

Hawaiian Population Movement



Hawaii Migration

Our textbook considers a more complicated example involving four Hawaiian islands, Oahu, Kauai, Maui and Lanai.

Suppose in the year 2000 each island had one million inhabitants, but we had the following transition data for moving from one island to another:

		Last Year:			
		OA	KA	MA	LA
This Year:	OA	0.32	0.17	0.11	0.46
	KA	0.18	0.43	0.32	0.33
	MA	0.27	0.22	0.39	0.14
	LA	0.23	0.18	0.18	0.07
Sum		1.00	1.00	1.00	1.00

What the Matrix Means

Row I describes all the places a person moving to island I can come from, with a probability. Thus, next year's population of Lanai will be 23% of Oahu's population, 18% of Kauai's population, and so on.

Column J describes all the places a person on island J can go to, with a probability. Thus, 32% of the Oahu residents stay on Oahu, 18% move to Kauai, and so on.

hawaii_update.m

```
function [ oa, ka, ma, la ] = hawaii_update ( oa, ka, ma, la )
```

```
    oa_old = oa;
```

```
    ka_old = ka;
```

```
    ma_old = ma;
```

```
    la_old = la;
```

```
    oa = 0.32 * oa_old + 0.17 * ka_old + 0.11 * ma_old + 0.46 * la_old;
```

```
    ka = 0.18 * oa_old + 0.43 * ka_old + 0.32 * ma_old + 0.33 * la_old;
```

```
    ma = 0.27 * oa_old + 0.22 * ka_old + 0.39 * ma_old + 0.14 * la_old;
```

```
    la = 0.23 * oa_old + 0.18 * ka_old + 0.18 * ma_old + 0.07 * la_old;
```

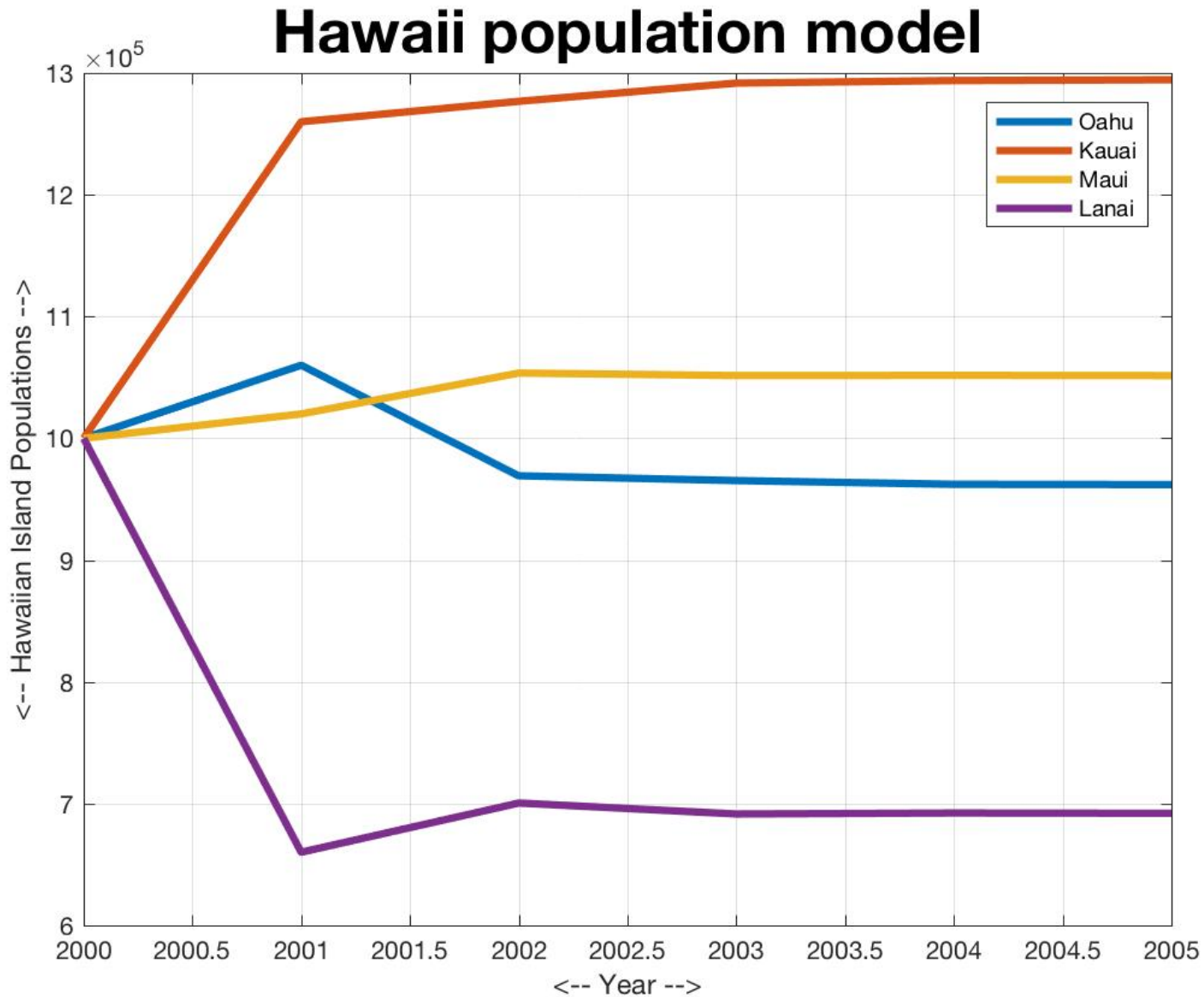
```
    return
```

```
end
```

Quick Changes, Then Settling

Year	Oahu	Kauai	Maui	Lanai	Total
2000	1000000	1000000	1000000	1000000	4000000
2001	1060000	1260000	1020000	660000	4000000
2002	969200	1276800	1053600	700400	4000000
2003	965280	1291764	1051540	691416	4000000
2004	962210	1293869	1051713	692208	4000000
2005	961969	1294538	1051525	691968	4000000

Population Settles Down



Matrix-Vector Multiplication

$$\begin{bmatrix} 1 & 0 & 2 & 0 \\ 0 & 3 & 0 & 4 \\ 0 & 0 & 5 & 0 \\ 6 & 0 & 0 & 7 \end{bmatrix} \cdot \begin{bmatrix} 2 \\ 5 \\ 1 \\ 8 \end{bmatrix} = \begin{bmatrix} 4 \\ 47 \\ 5 \\ 68 \end{bmatrix}$$

CA Transition Method #1

Let's look at the California population problem again. If we had the California and US populations from last year, we estimated the populations next year by this procedure:

```
ca_old = ca;
```

```
us_old = us;
```

```
us = 0.90 * us_old + 0.30 * ca_old;
```

```
ca = 0.10 * us_old + 0.70 * ca_old;
```

CA Transition Method #2

It's a little neater to keep the probabilities in a matrix P . We could keep the populations in a vector "pop". Then our calculation becomes:

$$P = \begin{bmatrix} 0.90 & 0.30 \\ 0.10 & 0.70 \end{bmatrix};$$

$$\text{pop_old} = \text{pop};$$

$$\text{pop}(1) = P(1,1) * \text{pop_old}(1) + P(1,2) * \text{pop_old}(2);$$

$$\text{pop}(2) = P(2,1) * \text{pop_old}(1) + P(2,2) * \text{pop_old}(2);$$

CA Transition Method #3

We can use a FOR loop to rewrite

```
pop(1) = P(1,1) * pop_old(1) + P(1,2) * pop_old(2);  
pop(2) = P(2,1) * pop_old(1) + P(2,2) * pop_old(2);
```

As:

```
for i = 1 : 2  
    pop(i) = P(i,1) * pop_old(1) + P(i,2) * pop_old(2);  
end
```

CA Transition Method #4

But we can actually write the whole operation in a simple form using the rules of matrix-vector multiplication in linear algebra:

$$\text{pop}(1) = P(1,1) * \text{pop_old}(1) + P(1,2) * \text{pop_old}(2);$$

$$\text{pop}(2) = P(2,1) * \text{pop_old}(1) + P(2,2) * \text{pop_old}(2);$$

is rewritten as:

$$\text{pop} = P * \text{pop_old};$$

Note that pop and pop_old must be **column vectors**. Also note that $P * \text{pop_old}$ does not use the "dot" notation for multiplication!

Matrix-Vector Multiplication

We have just seen a linear algebra operation called **matrix-vector multiplication**.

In this case, the rules for "pop = P * pop_old" are

To get item **1** of pop, combine row **1** of P with the entries of pop_old:

$$\text{pop}(\mathbf{1}) = P(\mathbf{1},1) * \text{pop_old}(1) + P(\mathbf{1},2) * \text{pop_old}(2)$$

To get item **2** of pop, combine row **2** of P with the entries of pop_old:

$$\text{pop}(\mathbf{2}) = P(\mathbf{2},1) * \text{pop_old}(1) + P(\mathbf{2},2) * \text{pop_old}(2)$$

Example #1

Matrix-vector multiplication gives us a short way to represent and compute the operations we were doing for transitions.

Let's look at some examples:

$$A = \begin{bmatrix} 0.90 & 0.30 \\ 0.10 & 0.70 \end{bmatrix};$$

$$v = \begin{bmatrix} 200 \\ 100 \end{bmatrix}; \quad \leftarrow \text{Column vector}$$

$$w = A * v = \begin{bmatrix} 0.90 * 200 + 0.30 * 100 \\ 0.10 * 200 + 0.70 * 100 \end{bmatrix} = \begin{bmatrix} 180 + 30 \\ 20 + 70 \end{bmatrix} = \begin{bmatrix} 210 \\ 90 \end{bmatrix};$$

Example #2

$$A = \begin{bmatrix} 1, & 2; \\ 3, & 4 \end{bmatrix};$$

$$v = \begin{bmatrix} 50; \\ 60 \end{bmatrix}; \quad \leftarrow \text{Column vector}$$

$$w = A * v = \begin{bmatrix} 1 * 50 + 2 * 60; \\ 3 * 50 + 4 * 60 \end{bmatrix} = \begin{bmatrix} 50 + 120; \\ 150 + 240 \end{bmatrix} = \begin{bmatrix} 170; \\ 390 \end{bmatrix};$$

Example #3

$$A = \begin{bmatrix} 2 & 6 & 5 \\ 3 & 2 & 0 \\ 4 & 1 & 7 \end{bmatrix};$$

$$v1 = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}; \quad v2 = \begin{bmatrix} 10 \\ 20 \\ 30 \end{bmatrix}; \quad \leftarrow \text{Column vectors}$$

$$w1 = A * v1 = \begin{bmatrix} 2*1 + 6*0 + 5*1 \\ 3*1 + 2*0 + 0*1 \\ 4*1 + 1*0 + 7*1 \end{bmatrix} = \begin{bmatrix} 2 + 0 + 5 \\ 3 + 0 + 0 \\ 4 + 0 + 7 \end{bmatrix} = \begin{bmatrix} 7 \\ 3 \\ 11 \end{bmatrix};$$

$$w2 = A * v2 = \begin{bmatrix} 2 * 10 + 6 * 20 + 5 * 30 \\ 3 * 10 + 2 * 20 + 0 * 30 \\ 4 * 10 + 1 * 20 + 7 * 30 \end{bmatrix} = \begin{bmatrix} 20 + 120 + 150 \\ 30 + 40 + 0 \\ 40 + 20 + 210 \end{bmatrix} = \begin{bmatrix} 290 \\ 70 \\ 270 \end{bmatrix};$$

Conclusion

The transition model is a simple system that simulates repeated change that can be described by probabilities.

The probabilities can be stored in a matrix P .

We can simulate changes to one individual case, or the expected behavior of big populations.

Transitions can be described in terms of matrix-vector multiplication.