Math 283, Winter 2009, Prof. Tesler – March 2, 2009
Example of powers of a diagonalizeable matrix

Sample matrix: (not a transition matrix, just easy numbers)

$$P = \begin{bmatrix} 8 & -1 \\ 6 & 3 \end{bmatrix}$$

Diagonalize: $P = VDV^{-1}$ where

$$V = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \qquad D = \begin{bmatrix} 5 & 0 \\ 0 & 6 \end{bmatrix} \qquad V^{-1} = \begin{bmatrix} -2 & 1 \\ 3/2 & -1/2 \end{bmatrix}$$

Eigenvalues of P: From the diagonal of D, $\lambda_1 = 5$, $\lambda_2 = 6$.

Right eigenvectors of P: The columns of $V = \left[\ \vec{r}_1 \mid \vec{r}_2\ \right]$:

$$\begin{bmatrix} 8 & -1 \\ 6 & 3 \end{bmatrix} \begin{bmatrix} 1 \\ 3 \end{bmatrix} = \begin{bmatrix} 8(1) - 1(3) \\ 6(1) + 3(3) \end{bmatrix} = \begin{bmatrix} 5 \\ 15 \end{bmatrix} = 5 \begin{bmatrix} 1 \\ 3 \end{bmatrix} \qquad \begin{bmatrix} 8 & -1 \\ 6 & 3 \end{bmatrix} \begin{bmatrix} 2 \\ 4 \end{bmatrix} = \begin{bmatrix} 8(2) - 1(4) \\ 6(2) + 3(4) \end{bmatrix} = \begin{bmatrix} 12 \\ 24 \end{bmatrix} = 6 \begin{bmatrix} 2 \\ 4 \end{bmatrix}$$

Left eigenvectors of P: The rows of $V^{-1} = \begin{bmatrix} \vec{\ell}'_1 \\ \hline \vec{\ell}'_2 \end{bmatrix}$:

$$\begin{bmatrix} -2 & 1 \end{bmatrix} \begin{bmatrix} 8 & -1 \\ 6 & 3 \end{bmatrix} = \begin{bmatrix} -2(8) + 1(6) & -2(-1) + 1(3) \end{bmatrix} = \begin{bmatrix} -10 & 5 \end{bmatrix} = 5 \begin{bmatrix} -2 & 1 \end{bmatrix}$$

$$\begin{bmatrix} 1.5 & -.5 \end{bmatrix} \begin{bmatrix} 8 & -1 \\ 6 & 3 \end{bmatrix} = \begin{bmatrix} 1.5(8) - .5(6) & 1.5(-1) - .5(3) \end{bmatrix} = \begin{bmatrix} 9 & -3 \end{bmatrix} = 6 \begin{bmatrix} 1.5 & -.5 \end{bmatrix}$$

Powers of P: An expansion of P^n is $P = (VDV^{-1})(VDV^{-1}) \cdots (VDV^{-1}) = VD^nV^{-1}$:

$$VD^nV^{-1} = V \begin{bmatrix} 5^n & 0 \\ 0 & 6^n \end{bmatrix} V^{-1} = V \begin{bmatrix} 5^n & 0 \\ 0 & 0 \end{bmatrix} V^{-1} + V \begin{bmatrix} 0 & 0 \\ 0 & 6^n \end{bmatrix} V^{-1}$$

$$V \begin{bmatrix} 5^n & 0 \\ 0 & 0 \end{bmatrix} V^{-1} = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \begin{bmatrix} 5^n & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} -2 & 1 \\ 1.5 & -.5 \end{bmatrix} = \begin{bmatrix} (1)(5^n)(-2) & (1)(5^n)(1) \\ (3)(5^n)(-2) & (3)(5^n)(1) \end{bmatrix} = 5^n \begin{bmatrix} 1 \\ 3 \end{bmatrix} \begin{bmatrix} -2 & 1 \end{bmatrix} = {\lambda_1}^n \vec{r}_1 \vec{\ell}'_1$$

$$= 5^n \begin{bmatrix} -2 & 1 \\ -6 & 3 \end{bmatrix}$$

$$V \begin{bmatrix} 0 & 0 \\ 0 & 6^n \end{bmatrix} V^{-1} = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \begin{bmatrix} 0 & 0 \\ 0 & 6^n \end{bmatrix} \begin{bmatrix} -2 & 1 \\ 1.5 & -.5 \end{bmatrix} = \begin{bmatrix} 2(6^n)(1.5) & 2(6^n)(-.5) \\ 4(6^n)(1.5) & 4(6^n)(-.5) \end{bmatrix} = 6^n \begin{bmatrix} 2 \\ 4 \end{bmatrix} \begin{bmatrix} 1.5 & -.5 \end{bmatrix} = {\lambda_2}^n \vec{r}_2 \vec{\ell}'_2$$

$$= 6^n \begin{bmatrix} 3 & -1 \\ 6 & -2 \end{bmatrix}$$

Spectral decomposition of P^n:

$$VD^nV^{-1} = {\lambda_1}^n \vec{r}_1 \vec{\ell}'_1 + {\lambda_2}^n \vec{r}_2 \vec{\ell}'_2 = 5^n \begin{bmatrix} -2 & 1 \\ -6 & 3 \end{bmatrix} + 6^n \begin{bmatrix} 3 & -1 \\ 6 & -2 \end{bmatrix}$$

What if the matrix is not diagonalizeable? A similar trick can be done with the Jordan Canonical Form, which gives a block diagonalization of P. On the Markov Chain handout, several of the examples require this: the matrix $P1$ (for overlapping occurrences of `GAGA`), the matrix $P2$ (for non-overlapping occurrences of `GAGA`), and the matrix $P4$ (for the complicated 10 state machine) all required using the Jordan Canonical Form.

Matlab:

```
>> P=[8,-1;6,3]
P =
     8    -1
     6     3
```

```
>> [V,D] = eig(P)
V =
    0.4472    0.3162
    0.8944    0.9487

D =
     6     0
     0     5
```

```
>> Vi = inv(V)
Vi =
    6.7082   -2.2361
   -6.3246    3.1623
```

Right eigenvectors

```
>> r1 = V(:,1)
    0.4472
    0.8944

>> P * r1
    2.6833
    5.3666
>> 6 * r1
    2.6833
    5.3666

>> r2 = V(:,2)
    0.3162
    0.9487

>> P * r2
    1.5811
    4.7434
>> 5 * r2
    1.5811
    4.7434
```

Left eigenvectors

```
>> l1 = Vi(1,:)
    6.7082   -2.2361

>> l1 * P
   40.2492  -13.4164

>> l1 * 6
   40.2492  -13.4164

>> l2 = Vi(2,:)
   -6.3246    3.1623

>> l2 * P
  -31.6228   15.8114

>> l2 * 5
  -31.6228   15.8114
```

Transpose (actually adjoint = complex conj. of transpose)

```
>> P'
     8     6
    -1     3

>> r1'
    0.4472    0.8944

>> l1'
    6.7082
   -2.2361

>> C = [[1+2i,3+4i];[5+6i,7+8i]]
 1.0000+2.0000i 3.0000+4.0000i
 5.0000+6.0000i 7.0000+8.0000i

>> C'
 1.0000-2.0000i 5.0000-6.0000i
 3.0000-4.0000i 7.0000-8.0000i

>> real(C)
     1     3
     5     7

>> imag(C)
     2     4
     6     8
```

Spectral decomposition

```
>> S1 = r1 * l1
    3.0000   -1.0000
    6.0000   -2.0000

>> S2 = r2 * l2
   -2.0000    1.0000
   -6.0000    3.0000

>> lambda1 = D(1,1)
     6
>> lambda2 = D(2,2)
     5
```

```
>> P^2
    58   -11
    66     3

>> V * D^2 * Vi
   58.0000  -11.0000
   66.0000    3.0000

>> lambda1^2 * S1 + lambda2^2 * S2
   58.0000  -11.0000
   66.0000    3.0000
```

R:

```
> P = rbind(c(8,-1),
            c(6,3))
> P
     [,1] [,2]
[1,]    8   -1
[2,]    6    3
```

```
> eigP = eigen(P)
> eigen(P)
$values
[1] 6 5

$vectors
          [,1]      [,2]
[1,] 0.4472136 0.3162278
[2,] 0.8944272 0.9486833

> eigP = eigen(P)
> V = eigP$vectors
```

```
> D = diag(eigP$values)
> D
     [,1] [,2]
[1,]    6    0
[2,]    0    5

> Vi = solve(V)
> V
          [,1]      [,2]
[1,] 0.4472136 0.3162278
[2,] 0.8944272 0.9486833
```

- R's data structure called "vector" is not a mathematical vector; it's essentially a one-dimensional list.
- R prints it horizontally, but it's not intrinsically a mathematical "row vector."
- Extracting a row or column of a "matrix" data structure results in an R "vector." For true matrix operations, sometimes R correctly guesses whether to treat it as a row or column vector, but not reliably.
- Thus, we must coerce it into a one column or one row matrix data structure.

Right eigenvectors

```
> r1 = matrix(V[,1],
+              ncol=1)
> r1
          [,1]
[1,] 0.4472136
[2,] 0.8944272
> P %*% r1
         [,1]
[1,] 2.683282
[2,] 5.366563
> 6 * r1
         [,1]
[1,] 2.683282
[2,] 5.366563
> r2 = matrix(V[,2],
+              ncol=1)
> r2
          [,1]
[1,] 0.3162278
[2,] 0.9486833
> P %*% r2
         [,1]
[1,] 1.581139
[2,] 4.743416
> 5 * r2
         [,1]
[1,] 1.581139
[2,] 4.743416
```

Left eigenvectors

```
> l1 = matrix(Vi[1,],
+              nrow=1)
> l1
         [,1]      [,2]
[1,] 6.708204 -2.236068
> l1 %*% P
         [,1]      [,2]
[1,] 40.24922 -13.41641
> l1 * 6
         [,1]      [,2]
[1,] 40.24922 -13.41641
> l2 = matrix(Vi[2,],
+              nrow=1)
> l2 %*% P
          [,1]     [,2]
[1,] -31.62278 15.81139
> l2 * 5
          [,1]     [,2]
[1,] -31.62278 15.81139
```

Transpose

```
> t(P)
     [,1] [,2]
[1,]    8    6
[2,]   -1    3
> t(r1)
          [,1]      [,2]
[1,] 0.4472136 0.8944272
> t(l1)
          [,1]
[1,]  6.708204
[2,] -2.236068
```

Spectral decomposition

```
> S1 = r1 %*% l1
> S1
     [,1] [,2]
[1,]    3   -1
[2,]    6   -2
> S2 = r2 %*% l2
> S2

> lambda1 = eigP$values[1]
> lambda1
[1] 6
> lambda2 = eigP$values[2]
> lambda2
[1] 5
```

```
> P %*% P
     [,1] [,2]
[1,]   58  -11
[2,]   66    3
> V %*% diag(eigP$values ^ 2) %*% Vi
     [,1] [,2]
[1,]   58  -11
[2,]   66    3
> lambda1^2 * S1 + lambda2^2 * S2
     [,1] [,2]
[1,]   58  -11
[2,]   66    3
```

Complex numbers: In R, `t()` is just transpose, not adjoint.

```
> C = rbind(c(1+2i,3+4i),c(5+6i,7+8i))
> C
     [,1] [,2]
[1,] 1+2i 3+4i
[2,] 5+6i 7+8i


> t(C)
     [,1] [,2]
[1,] 1+2i 5+6i
[2,] 3+4i 7+8i

> Conj(C)
     [,1] [,2]
[1,] 1-2i 3-4i
[2,] 5-6i 7-8i
> Re(C)
     [,1] [,2]
[1,]    1    3
[2,]    5    7
> Im(C)
     [,1] [,2]
[1,]    2    4
[2,]    6    8
```