Revised 1/10/07

LAB 5: Eigenvalues and Eigenvectors

In this lab you will use MATLAB to study these topics:

- The geometric meaning of eigenvalues and eigenvectors of a matrix
- Determination of eigenvalues and eigenvectors using the characteristic polynomial of a matrix
- Use of eigenvectors to transform a matrix to diagonal form.
- Steady-state eigenvector for a transition matrix
- Applications of eigenvalues and eigenvectors to study Markov chains.

Preliminaries

Reading from Textbook: In connection with this Lab, read through Sections 5.1, 5.2, 5.3, and pp. 291-294 of Section 5.5 of the text and work the suggested problems for each section.

Script Files and T-codes: For this lab you will need the the m-file `rmat.m` from Lab 2 and the Teaching Code `nulbasis.m`.

Lab Write-up: You should open a diary file at the beginning of each MATLAB session (see Lab 1 for details). Be sure to answer all the questions in the lab assignment.

Random Seed: When you start your MATLAB session, initialize the random number generator by typing

```
rand('seed', abcd)
```

where *abcd* are the last four digits of your Social Security number. This will ensure that you generate your own particular random vectors and matrices.

BE SURE TO INCLUDE THIS LINE IN YOUR LAB WRITE-UP

The lab report that you hand in must be your own work. The following problems all use randomly generated matrices and vectors, so the matrices and vectors in your lab report will not be the same as those of other students doing the lab. Sharing of lab report files is not allowed in this course.

Question 1. Graphic Demonstration of Eigenvectors and Eigenvalues

(a) Type `eigshow` at the MATLAB prompt. A graphics window should open. Underneath the graph the statement

```
Make A*x parallel to x
```

should appear (if it does not, then click on the `eig` button to get this statement).

Click on the pull-down bar above the graph and select the matrix `[1 3; 4 2]/4`. Move the cursor onto the vector $\mathbf{x}$, and make $\mathbf{x}$ go around a full circle. The transformed vector $A\mathbf{x}$ then moves around an ellipse. Search for the *special lines* through zero that contain both $A\mathbf{x}$ and $\mathbf{x}$. When $\mathbf{x}$ lies on such a line, it is an *eigenvector* of the matrix A (the word *eigen* means *special* in German). For any $\mathbf{x}$ lying on these *special lines*, $A\mathbf{x} = \lambda\mathbf{x}$, where λ is an *eigenvalue* of A. Since $\mathbf{x}$ is a unit vector, the length of $A\mathbf{x}$ is $|\lambda|$. If $A\mathbf{x}$ points in the *same* direction as $\mathbf{x}$, then $\lambda > 0$. If $A\mathbf{x}$ points in the *opposite* direction to $\mathbf{x}$, then $\lambda < 0$.

From your graphical experimentation answer the following questions (no algebraic calculations needed):

(i) How many different positive eigenvalues does A have?

(ii) How many different negative eigenvalues does A have?

(iii) What are the (approximate) numerical values of the eigenvalues?

Be careful in counting eigenvalues; if $\mathbf{x}$ is an eigenvector with eigenvalue λ, then $-\mathbf{x}$ is also an eigenvector with the *same* eigenvalue. (Don't try to print the eigshow window.)

(b) Click on pull-down matrix selection bar again and select `[3 1; -2 4]/4`. Move $\mathbf{x}$ around the circle with the cursor and observe what happens, as in part **(a)**. Use your graphical experimentation to answer the following questions (no algebraic calculations needed):

(i) Are there any lines through zero that contain both $A\mathbf{x}$ and $\mathbf{x}$?

(ii) Does A have any *real* eigenvectors or eigenvalues? Explain.

Question 2. Characteristic Polynomial

At the MATLAB prompt type `A = [1 3; 4 2]/4` (this is the matrix in part **(a)** of Question #1). The eigenvalues of A are the roots of the *characteristic polynomial* of A.

(a) Use the MATLAB to calculate its characteristic polynomial $p(t)$ by

```
syms t; I = eye(2); p = det(A - t*I)
```

(you must be running MATLAB in an environment such as the Rutgers computing labs where the symbolic toolbox is installed to do this–the MATLAB installations in the School of Engineering do not have this toolbox).

(i) Verify by hand calculation that the constant term in the polynomial $p(t)$ is $\det(A)$.

(b) Use the MATLAB command `solve(p)` to get the roots of $p(t)$ (the eigenvalues of A). Compare these values with your graphical estimates for the eigenvalues from Question #1(a).

(c) Now at the MATLAB prompt type `A = [3 1 ; -2 4 ]/4` (this is the matrix in part **(b)** of Question #1). Calculate the characteristic polynomial $p(t)$ of A as in part **(b)** (use the $\uparrow$ key) and find its roots (the eigenvalues).

(i) Are the eigenvalues real?

(ii) Use this information to explain what you observed Question #1(b).

Question 3. Eigenvectors and Diagonalization

For this question generate a random 3×3 integer matrix `A = rmat(3,3)`.

(a) Calculate the characteristic polynomial $p(t)$ of A by

```
syms t; I = eye(3); p = det(A - t*I)
```

(as in Question #2, this requires that your version of MATLAB has the symbolic toolbox installed).

(i) Show (by hand calculation) how $\det(A)$ occurs as a coefficient in $p(t)$.

Use MATLAB to check this fact about $\det(A)$.

Plot the characteristic polynomial of A in a graphics window by

```
figure; ezplot(p, [-10, 10]), grid
```

Adjust the horizontal range of the plot (change $[-10, 10]$ as needed) until you can determine whether $p(t)$ has three real roots (zoom in as necessary, using the magnifying glass button on the top of the graph) . If there are not three real roots, generate a new matrix A and repeat the graphing until you get a characteristic polynomial with three real roots. Print the graph with a range that shows all three real roots, and include the graph in your lab report.

(ii) Use the graph to obtain approximate values for the three real roots of $p(t)$.

(b) Use the MATLAB command

```
[P  D] = eig(A)
```

to generate a matrix P and a diagonal matrix D.

(i) Compare the diagonal entries of D with your graphical estimates for the eigenvalues of A in part **(a)**.

Use MATLAB to define

```
p1 = P(:,1),   p2 = P(:,2),   p3 = P(:,3)
```

(the columns of P). Calculate

```
A*p1 - D(1,1)*p1,  A*p2 - D(2,2)*p2,  A*p3 - D(3,3)*p3
```

(ii) What does this calculation tell you about the eigenvalues and eigenvectors of A? (See Theorem 5.2, page 272.)

(c) Let A, P, D be as in part **(b)**. Verify by MATLAB that `A = P*D*inv(P)`.

(i) Show how to use this formula to express A^5 and A^{10} in terms of D and P.

Verify your answer to this question using MATLAB.

Question 4. Steady-State Eigenvector for a Transition Matrix

(a) A matrix is called a *transition matrix* if its entries are nonnegative and the sum of the entries in each column is one. For this question generate a random 2×2 transition matrix A by

```
A = eye(2); B = rand(2);
A(:,1) = B(:,1)/sum(B(:,1)); A(:,2) = B(:,2)/sum(B(:,2))
```

Calculate `[1  1]*A`.

(i) Show (by a hand calculation) why you can use the vector $[1\ 1]A$ to show that A is a transition matrix.

(b) Transition matrices such as A with all entries positive are called *regular*. For a regular transition matrix, 1 is the largest eigenvalue, and the corresponding eigenspace is one-dimensional. Use the T-code `nulbasis` to calculate a normalized eigenvector for the matrix A you generated in part **(a)**.

```
u = nulbasis(A - eye(2)), v = u/sum(u)
```

The vector $\mathbf{v}$ should have components that are positive and sum to 1. Verify by MATLAB that $A\mathbf{v} = \mathbf{v}$. Thus $\mathbf{v}$ is an eigenvector for A with eigenvalue 1, called the *steady-state vector* for A. Plot this vector (as a solid line) by

```
figure; plot([0,v(1)], [0, v(2)]); hold on
```

Leave the graphic window open for the next part.

(c) A general result about regular transition matrices (Theorem 5.4 on page 293) asserts that if $\mathbf{p}$ is any initial choice of a probability vector in $\mathbf{R}^2$, then the sequence of vectors $A^k\mathbf{p}$ converges to the steady-state vector $\mathbf{v}$ as $k \to \infty$. To demonstrate this graphically for your matrix A, generate a random initial probability vector

```
w = rand(2,1), p = w/sum(w)
```

Now graph the vector $A\mathbf{p}$ (as a dotted line) in the same window from part **(b)**:

```
p = A*p, plot([0,p(1)], [0, p(2)], ':'); hold on
```

To plot the sequence of vectors $A^2\mathbf{p}$, $A^3\mathbf{p}, A^4\mathbf{p}, \ldots$ in the graphics window, just use the up-arrow key $\uparrow$ to repeat this last command. Do this as many times as needed until the vector **p** has converged numerically (to three decimal places in each component) to the steady-state vector **v** that you plotted in part **(b)**. Print the graphics window and include it in your lab report.

Question 5. Markov Chains

Read the section *Markov Chains* in Section 5.5 and look at Practice Problem 1 (page 302) and its solution (page 307). The following questions refer to Exercise #14 on page 303 of the text. Enumerate the states as 1 = city, 2 = suburbs, 3 = country.

(a) Determine the transition matrix A in Exercise #14 (page 303) and enter it into your MATLAB workspace. Let **u** be the row vector [1, 1, 1]. Verify that each column of A sums to 1 by calculating that $\mathbf{u} * A = \mathbf{u}$.

(b) Determine the initial probability vector **p** from the description given in Exercise #14. Verify that the entries of **p** sum to 1 by calculating that $\mathbf{u} * \mathbf{p} = 1$. Now use powers of the matrix A and the vector **p** to find the percentage of people living in the city, suburbs, and country after 1, 2, 3, 5, and 8 years (insert comments to identify your results).

(c) The steady-state probability vector **v** is an eigenvector for A with eigenvalue 1. Use MATLAB to find **v** by the same method as Question #4(b).

(i) What is the relation between the vector **v** and the results in part **(b)**?

Final Editing of Lab Write-up: After you have worked through all the parts of the lab assignment, you will need to edit your diary file. Remove all errors and other material that is not directly related to the questions. Your write-up should only contain the required MATLAB calculations and the answers to the questions. Preview the document before printing and remove unnecessary page breaks and blank space.