Mathematics 244: Lab 4

Linear Systems

Spring 2005

0. Introduction and Setup In this lab we use Maple to find **eigenvalues** and **eigenvectors** of matrices, and solve linear systems of algebraic equations and systems of first order linear differential equations. We also obtain pictures of the slope fields of these equations in the **phase plane**.

The first step is to obtain the **seed file** from the web page and save it in your directory on eden. You will be asked to execute the Maple commands that have been placed in this worksheet and to add some of your own. Please turn in only the printout of your main Maple worksheet. Include explicit answers to all questions asked, using the **text** feature of Maple to insert them in the worksheet. Remove any extraneous material from the worksheet. There is also a **supplementary worksheet** that suggests additional experiments that may lead to better answers to the questions considered in this project.

In addition to the `DEtools` and `plots` libraries used in other labs, we require a Linear Algebra package. There are now two such packages in Maple, not completely compatible. The newer package, called `LinearAlagebra`, introduced in Maple6 is used here since it has more respect for its user. In **Section 0** of the worksheet, the libraries are loaded and two matrices are defined.

```
with(LinearAlgebra):with(DEtools):with(plots):
A:= <<9,-3>|<0,6>|<3,32>>;
B:= <<8|-12>,<42|-121>,<-6|30>>;
```

1. Matrix Operations We first try some operations. Since some will give errors, and some may give unexpected results, these will be investigated in the **supplementary worksheet**. A few examples are already in the supplementary worksheet, but you should add others to allow a full discussion of these operations. Some of these examples will lead to errors. The errors will find their place in the **discussion**, but the Maple commands leading to them **should not appear in the main worksheet**. Consult the help pages to find properties of the operations used in these statements and **write a brief description** of how + and . behave as operators on matrices (you may need to use a **Search** for "dot" to find all the help pages for that operator). Here are the questions for discussion:

(1) How is `M1+M2` computed when `M1` and `M2` are matrices? Is it **always defined**? If not, how does Maple indicate that it is not defined?

(2) How is `M+c` computed when `M` is a matrix and `c` is a scalar constant? Is it **always defined**? If not, how does Maple indicate that it is not defined?

(3) How is `M1.M2` computed when `M1` and `M2` are matrices? Is it **always defined**? If not, how does Maple indicate that it is not defined?

(4) How is `M.c` computed when `M` is a matrix and `c` is a scalar constant? Is it **always defined**? If not, how does Maple indicate that it is not defined?

(5) How is `M^c` computed when `M` is a matrix and `c` is an integer constant (possibly negative)? Is it **always defined**? If not, how does Maple indicate that it is not defined?

2. Eigenvalues and eigenvectors

The following lines appear in the seed file — the first three in the preceding section. They use the `Eigenvectors` command to obtain eigenvectors **and** eigenvalues of three related matrices. The `Map` command is then used to modify the entries in the eigenvectors so that their entries take a standard appearance (`Vecs2` was particularly stubborn, but the two operations used here produce a reasonable result — other commands may be investigated in the supplementary worksheet). These instructions may be replaced if you find a better way to get the same result.

```
M1:=A.B;
M2:=B.A;
M3:=M1^(-1);
(Vals1,Vecs1):=Eigenvectors(M1);
(Vals2,Vecs2):=Eigenvectors(M2);
(Vals3,Vecs3):=Eigenvectors(M3);
Map(rationalize,Vecs1);
Map(rationalize,Vecs2);
Map(rationalize,Vecs3);
Map(expand,Vecs2);
C:=<<1|1|0>,<0|1|0>,<0|0|2>>;
Eigenvectors(C);
```

The relation found here between the eigenvalues of `M1` and `M2` is quite general, although the usual proof would be a distraction in this course. In special cases, though, it follows from the **easy** observation that left multiplication by `B` takes an eigenvector of `AB` to an eigenvector of `BA` and left multiplication by `A` takes an eigenvector of `BA` to an eigenvector of `AB`. This is illustrated in the **supplementary worksheet**.

The relation between the eigenvalues of `M1` and `M3` is also quite general, and a reason for it can be found by looking at the eigenvectors for this pair of matrices. The `rationalize` operation was applied to each entry of these matrices by the `Map` instruction to allow us to compare the entries of matrices constructed by different computations. If the results are not yet in simplest form, it may be necessary to use either `simplify` or `expand` as the first argument of a `Map` instruction to get results that can be compared. (Maple often sees some expressions as simple that are not considered completely simplified in Elementary Algebra, but you can force Maple to multiply expressions with the `expand` instruction. After that, a `simplify` instruction will generally only collect similar terms.)

The discussion section of this part should

(∗) explain the significance of a zero column in the matrix of eigenvectors of C, since we have claimed that **the zero vector is never an eigenvector**.

If a vector v is linearly independent of the eigenvectors of C, the lack of a third eigenvector suggests that Cv **never** simply a multiple of v.

(∗) Why should this be?

3. Real matrix Exponentials

The `LinearAlgebra` package provides a good interface to do numerical work with matrices, but it needs to be cajoled into doing symbolic work. This can be done with the `map` function. This is not the same as the `Map` function that was used in Section 2 to modify `Matrices` in place. This time we need to construct a new matrix without affecting the original one, so we will use the lower case variant. Be careful to distinguish these two instructions. If one is used where the other is needed, the results will be **very different from what you intend** and subsequent instructions will not work correctly. A simple example is given in the **supplementary worksheet**.

When eigenvalues of an n by n matrix M are real and distinct, the eigenvectors form a **basis** of the space $\mathbb{R}^n$. For each eigenvalue λ_i, the corresponding eigenvector v_i is the vector of coefficients of $e^{\lambda_i t}$ in a solution of $d\mathbf{y}/dt = M\mathbf{y}$. Since the exponential factor takes the value 1 when $t = 0$, the initial conditions give a system of equations whose matrix of coefficients Φ whose i^{th} column is the eigenvector v_i of M and whose right side is $\mathbf{y}(0)$. The solution $\Phi^{-1}\mathbf{y}(0)$ is the vector of coefficients of the special solutions $\mathbf{v}_i e^{\lambda_i t}$. This leads to the expression

$$\mathbf{y} = \Phi e^{\Lambda t}\Phi^{-1}\mathbf{y}(0),$$

where $e^{\Lambda t}$ is a diagonal matrix whose entries are the functions $e^{\lambda_i t}$.

The construction of this matrix when M is the 3×3 matrix `M2` from Section 1 uses the following Maple commands (requiring results found in Section 1) that are included in the seed file. The exact values may be used in the exponents, but the exact form of the coefficients would be awkward to work with. Before constructing e^{Mt} for this matrix, the `Map` instruction is used to replace `Vecs2` by a numerical approximation. When working with approximations, `DY2` and `MY2` are only **approximately equal**. The difference of these is small and `Y2` is **approximately equal** to an identity matrix, the reliability of the solution is verified. Here are the commands in the seed file.

```
Map(evalf,Vals2);
Map(evalf,Vecs2);
EL2:=DiagonalMatrix(map(c->exp(c*t),Vals2));
Y2:=Vecs2.EL2.Vecs2^(-1);
DY2:=map(diff,Y2,t);
MY2:=M2.Y2;
DY2-MY2;
subs(t=0,Y2);
```

Follow this computation with a **text comment** identifying the size of the coefficients in the expressions measuring the failure of `Y2` to be an exact solution.

4. Saddle points and nodes

Consider the matrices

$$M_{4A} = \begin{bmatrix} 4 & -1 \\ 2 & 1 \end{bmatrix} \quad \text{and} \quad M_{4B} = \begin{bmatrix} 8 & -1 \\ 2 & -7 \end{bmatrix}.$$

For each, we will use the method of Section 2 (which will be a little easier here because these systems have 2×2 matrices) to solve the equation $d\mathbf{y}/dt = M\mathbf{y}$ with initial conditions

$$(a) \quad \mathbf{y}(0) = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \qquad (b) \quad \mathbf{y}(0) = \begin{bmatrix} -1 \\ 1 \end{bmatrix}, \qquad (c) \quad \mathbf{y}(0) = \begin{bmatrix} -1 \\ -1 \end{bmatrix}.$$

A graphical check of the solution involves plotting the slope field of the equation in a **phase plane** whose coordinates are the components of $\mathbf{y}$ and superimposing a **parametric plot** of the **trajectories** of the solutions. Because the equations are **autonomous**, all solutions starting at a point on one of these trajectories will follow the trajectory — the only difference being the value of t at which it visits a particular point. This uses features not explored previously in this course, so the seed file contains instructions to construct the individual plots for the first of these equations. You will need to combine these plots with a `display` command and repeat the process, with modifications, for the second equation. Here are the instructions that appear in the seed file (spread across several sections) for getting graphs for the first system. You will need to provide the instructions for the second system.

```
M4A:=<<4,2>|<-1,1>>;
(Vals4A,Vecs4A):=Eigenvectors(M4A);
EL4A:=DiagonalMatrix(map(c->exp(c*t),Vals4A));
Y4A:=Vecs4A.EL4A.Vecs4A^(-1);
Y4Aa:=Y4A.<1,0>;
Y4Ab:=Y4A.<-1,1>;
Y4Ac:=Y4A.<-1,-1>;
VecVar:=<y1(t),y2(t)>;
listVar:=convert(VecVar,'list');
tvals:=t=-2..2;
range4:=y1=-2..2,y2=-2..2;
eq4A:=[diff(VecVar[1],t)=(M4A.VecVar)[1],
       diff(VecVar[2],t)=(M4A.VecVar)[2]];
Field4A:=DEplot(eq4A,listVar,tvals,range4,color=GREEN):
Sol4A:=plot([[Y4Aa[1],Y4Aa[2],tvals],
             [Y4Ab[1],Y4Ab[2],tvals],
             [Y4Ac[1],Y4Ac[2],tvals]],
             range4,color=[BLACK,RED,BLUE]):
display({Field4A,Sol4A},title="Equation 4A");
```

We need $y_1(t)$ and $y_2(t)$ to be combined into a `Vector` in order to use the `LinearAlgebra` package, and into a `list` to serve as a argument of the `DEplot` function. The `convert` instruction assures that the related objects will have the **same contents** in **different formats**. The entries of the vector appear in the differential equation because the `diff` operation only applies to scalar functions. The expression `Sol4A` plots a **list** of objects, each of which is a parametric description of the trajectory of a solution in the phase plane. Colors are assigned to the plots in the same order that they appear in the list.

The **Discussion** portion of this section should investigate the role of these plots in checking this method of solving differential equations. In particular:

(1) do the claimed solution curves look like they follow the slope field of the equation?
(2) for which of these equations has is the origin a saddle point?
  (∗) does the shape of the solution agree with what you expect?
(3) for the other equation,
  (∗) is the origin **stable** (i.e., attracting) or **unstable** (i.e., repelling)?
(4) describe how both the eigenvalues of the coefficient matrix and the slope field illustrate your classification;
(5) for both equations, find the solutions whose trajectories lie along straight lines.

5. Spiral points

Consider the equation

$$\frac{dY}{dt} = \begin{bmatrix} 2 & 5 \\ -2 & 0 \end{bmatrix} Y. \tag{S}$$

It can be shown that, if M is a 2×2 real matrix with eigenvalues $a \pm bi$ and J is defined by $M = aI + bJ$, then

$$e^{Mt} = e^{at}\big((\cos bt)I + (\sin bt)J \big).$$

The seed file implements this solution in Maple for (S), leading to a matrix `Y5`. You should use methods explored in previous sections to **verify** that this matrix is e^{Mt}, and to **illustrate** the solutions with the same initial values (a), (b) and (c) used in Section 4 superimposed on a slope field of this equation.

Here are the instructions leading to `Y5`.

```
M5:=<<2|5>,<-2|0>>;
E5:=Eigenvalues(M5);
(a5,b5):=Re(E5[1]),Im(E5[1]);
J5:=(1/b5).(M5-a5);
Y5:=Multiply(exp(a5*t),cos(b5*t)+Multiply(sin(b5*t),J5));
```

6. Repeated Eigenvalues

A similar process to the one used in Section 5 can be applied in the case of a matrix like

$$M_6 = \begin{bmatrix} -3 & -2 \\ 18 & 9 \end{bmatrix}$$

that has a repeated eigenvalue. If M is a 2×2 matrix with a as a double eigenvalue, then $M = aI + N$ where N^2 is the zero matrix. General properties of matrix exponentials show that

$$e^{Mt} = e^{aIt}e^{Nt} = e^{at}(I + Nt). \qquad (N)$$

You don't need to **derive** this result to show that it gives the solution. Instead, you can modify the instructions of Section 5 to implement the expression for $e^{M_6 t}$ given by (N), obtaining a matrix `Y6`. **Verify** that the derivative of `Y6` is equal to the matrix obtained by multiplying on the left by M_6, and that `Y6` reduces to the identity matrix when $t = 0$.

Then, **illustrate** these solutions with a plot of the direction field of the equation in the phase plane with the solutions with the initial conditions (a), (b) and (c) introduced in Section 4.

End of Lab 4