ME 406
Using Eigenvector Methods with *Mathematica* to Solve Linear Autonomous Systems of First Order Differential Equations

1. Introduction

In this notebook, we use the methods of linear algebra -- specifically eigenvector and eigenvalue analysis -- to solve systems of linear autonomous ordinary differential equations. Although the *Mathematica* routines DSolve and NDSolve could be used to attack these problems directly, we do not use them here. Our purpose is to make clear the underlying linear algebra, and to use *Mathematica* to do all of the calculations. The canonical problem under consideration is the following:

$$\mathbf{X} = \mathbf{A}\mathbf{X} + \mathbf{b} \;, \qquad (1)$$

$$\mathbf{X}(0) = \mathbf{X}_0 \;, \qquad (2)$$

where $\mathbf{A}$ is a constant n x n matrix, $\mathbf{b}$ is a constant n x 1 vector, and $\mathbf{X}_0$, the initial vector, is a given n x 1 vector.

The general approach to this problem is the following. We first find a particular solution, which is defined to be any solution of equation (1). We call the particular solution $\mathbf{X}_p$. The general solution $\mathbf{X}_g$ of the equation is then

$$\mathbf{X}_g = \mathbf{X}_h + \mathbf{X}_p \;, \qquad (3)$$

where $\mathbf{X}_h$ is the most general solution of the homogeneous equation:

$$\mathbf{X}_h = \mathbf{A}\mathbf{X}_h \;. \qquad (4)$$

The homogenous equation (4) has n linearly independent solutions. We call them $\mathbf{X}^{(1)}$, $\mathbf{X}^{(2)}$, . . . $\mathbf{X}^{(n)}$. The most general $\mathbf{X}_h$ is a linear combination of these. To solve the initial-value problem, we form the general solution $\mathbf{X}_g$ in terms of n constants a_1 , a_2 , . . . , a_n :

$$\mathbf{X}_g = \mathbf{X}_p + a_1 \mathbf{X}^{(1)} + a_2 \mathbf{X}^{(2)} + \ldots + a_n \mathbf{X}^{(n)} . \qquad (5)$$

We now impose the initial condition (2) on the solution (5). This gives the following set of linear equations to solve for the coefficients a_i :

$$a_1 \mathbf{X}^{(1)} + a_2 \mathbf{X}^{(2)} + \ldots + a_n \mathbf{X}^{(n)} = \mathbf{X}_0 - \mathbf{X}_p \text{ at } t = 0. \qquad (6)$$

The linear independence of the vectors $\mathbf{X}^{(i)}$ guarantees that the matrix in the above equations is nonsingular and hence the solution for the coefficients a_i is unique.

The rest of this notebook provides the details in carrying this out, and shows how to use *Mathematica* to advantage at each step. We will begin with a brief review of matrix manipulations in *Mathematica*. Then we will consider the problem of finding the particular solution. After that, we will solve the homogeneous equation, in the following sequence of cases of increasing difficulty: distinct real eigenvalues; distinct complex eigenvalues; repeated eigenvalues. Detailed examples will be done at each step.

2. Basic Matrix Manipulations in Mathematica

In *Mathematica*, a matrix is a list of lists. Each component list is a row of the matrix. As an example, we define a matrix named A for *Mathematica*, and then use MatrixForm to print it out in traditional form.

```
A = {{1, -2, 3}, {-2, -1, 4}, {3, 4, 5}};
```

```
MatrixForm[A]
```

$$\begin{pmatrix} 1 & -2 & 3 \\ -2 & -1 & 4 \\ 3 & 4 & 5 \end{pmatrix}$$

One way to find out whether A is singular is to compute the determinant.

```
Det[A]
```

```
-80
```

Because the determinant is not zero, A is not singular, and thus has an inverse.

```
B = Inverse[A]
```

$$\left\{\left\{\frac{21}{80}, -\frac{11}{40}, \frac{1}{16}\right\}, \left\{-\frac{11}{40}, \frac{1}{20}, \frac{1}{8}\right\}, \left\{\frac{1}{16}, \frac{1}{8}, \frac{1}{16}\right\}\right\}$$

We check the inverse by forming the product of A and the inverse of A. The product should be the identity matrix. Matrix products are indicated by a period.

```
A.B
```

```
{{1, 0, 0}, {0, 1, 0}, {0, 0, 1}}
```

```
B.A
```

```
{{1, 0, 0}, {0, 1, 0}, {0, 0, 1}}
```

We can use this inverse to solve a set of linear algebraic equations in which A is the coefficient matrix. Consider the equations

$x - 2y + 3z = 1$,

$-2x - y + 4z = 0$, (7)

$3x + 4y + 5z = -1.$

The coefficient matrix is the matrix A, and the right-hand side is the vector

```
b = {1, 0, -1};
```

We can use the inverse we have already calculated to solve these equations. We call the solution sol.

```
sol = B.b
```

$\{\frac{1}{5}, -\frac{2}{5}, 0\}$

We check this:

```
A.sol - b
```

```
{0, 0, 0}
```

An alternative method of solution is to use *Mathematica*'s LinearSolve.

```
sol2 = LinearSolve[A, b]
```

$\{\frac{1}{5}, -\frac{2}{5}, 0\}$

We get the same result.

In the above calculations, all the elements of A were integers, so *Mathematica* did an exact calculation. If one or more elements of A are written as real numbers, *Mathematica* will do a numerical rather than exact calculation.

```
Amod = {{1., -2, 3}, {-2, -1, 4}, {3, 4, 5}};
```

```
Inverse[Amod]
```

```
{{0.2625, -0.275, 0.0625},
 {-0.275, 0.05, 0.125}, {0.0625, 0.125, 0.0625}}
```

For the present calculations, it doesn't matter much which way we do them. For larger matrices and more difficult calculations, such as eigenanalysis, it is crucial to do the calculations numerically. We shall see an example of this shortly.

The determination of eigenvalues and eigenvectors is the central linear algebra calculation for solving systems of first-order linear autonomous differential equations. Given a square matrix **A**, we say that a non-zero vector **c** is an eigenvector of **A** with eigenvalue λ if $\mathbf{Ac} = \lambda\mathbf{c}$. *Mathematica* has a lot of built-in power to find eigenvectors and eigenvalues. We go back to our matrix **A** and use *Mathematica* to find its eigenvalue.