MATHEMATICS 201-NYC-05
Vectors and Matrices
Martin Huard
Fall 2007

Matrices with Maple

When working with matrices on Maple, the first thing to do is to load the *LinearAlgebra* package, which contains a lot of commands that we need.

```
> with(LinearAlgebra);
```

Definition of Matrices with Maple

Matrices can defined with the command `Matrix( )` by entering each row of the matrix. The command has the form

Matrix([[row 1], [row 2],…,[row n]]);

For example, let us define A as the matrix $\begin{bmatrix} 1 & 2 & 3 \\ 5 & 6 & 7 \end{bmatrix}$.

```
> A:=Matrix( [ [1,2,3], [5,6,7] ] );
```

For small matrices, you can use the matrix palette on your left. The numbers are filled in by typing them one at a time, and pressing the TAB key between two entries. This would give

```
> A:=<<1 | 2 | 3>, <5 | 6 | 7>>;
```

We can call elements in the matrix simply by specifying its location with the command

A[row #, column #]

For example, suppose we want the 5, which is in the second row and first column..

```
> A[2,1];
```

We can also call columns or rows in a matrix with the `Column( )` or `Row( )` commands from the *LinearAlgebra* package. For example, let us call the second row and the third column.

```
> Column(A,3);
> Row(A,2);
```

The identity and zero matrices are built into the *LinearAlgebra* package. For example, to have $0_{2\times 3}$ and $I_{3\times 3}$, we have the following

```
> ZeroMatrix(2,3);
> IdentityMatrix(3,3);
```

Algebra of Matrices

The algebra of matrices can be done using the `evalm( )` command, where we use ' &* ' as the symbol for matrix multiplication.

For example, let $A=\begin{bmatrix}1 & 2 & 3\\ 5 & 6 & 7\end{bmatrix}$, $B=\begin{bmatrix}3 & 5 & -1\\ 3 & 2 & 0\end{bmatrix}$ and $C=\begin{bmatrix}5 & -2 & 1\\ 1 & 3 & 0\\ -4 & 7 & 1\end{bmatrix}$.

```
> A:=Matrix( [ [1,2,3], [5,6,7] ] );
  B:=Matrix([ [3,5,-1], [3,2,0] ]);
  C:=Matrix([ [5,-2,1], [1,3,0], [-4,7,1] ]);
```

To find $A + B$, $4A$, AC and C^4, we have

```
> evalm(A+B);
> evalm(4*A);
> evalm(A&*C);
> evalm(C^4);
```

There are also specific commands in the *LinearAlgebra* package.

For addition, `MatrixAdd( , )`

```
> MatrixAdd(A,B);
```

For scalar multiplication, `Multiply( , )` or `ScalarMultiply( , )`

```
> Multiply(A,4);
> ScalarMultiply(A,4);
```

For matrix multiplication `Multiply( , )` or `MatrixMatrixMultiply( , )`

```
> Multiply(A,C);
> MatrixMatrixMultiply(A,C);
```

And for powers of *A*, `MatrixPower( , )`

```
> MatrixPower(C,4);
```

Other operations

There are other operations in the *LinearAlgebra* package that we can do with matrices, such as taking the *trace*, the *transpose*, the *adjoint*, the *determinant*, finding the *inverse*, the *characteristic matrix* $\lambda I - A$, the *characteristic polynomial*, the *eigenvalues* and the *eigenvectors*. Let us look at these operations using, as an example, the matrix *A* defined below.

```
> A:=Matrix([ [0,-3,5],[-4,4,-10],[0,0,4] ]);

> Trace(A);
> Transpose(A);
> Adjoint(A);
> Determinant(A);
> MatrixInverse(A);
> CharacteristicMatrix(A,lambda);
> CharacteristicPolynomial(A,lambda);
> Eigenvalues(A);
> Eigenvectors(A);
```