Scilab

Manjusha Joshi

July 4, 2009

1. Introduction

- scilab is a free software.
- It can work on windows as well as linux.
- scilab understands many mathematical data types like vectors, matrix, polynomial etc.
- scilab has inbuilt functions.
- It also allow us to do programming, in which we can use inbuilt commands e.g. rank, inv etc.
- scilab is case sensitive. So V and v is different in scilab.

In scilab basic data type is vector.

```
-->V=[2,-4,5]

-->V
 V  =


    2.  - 4.    5.

-->V'
 ans  =


    2.
  - 4.
    5.
```

Define two vector U and V and try $U + V$.
$U * V$
$U * V'$ What is your observation?
Try with other inbuilt functions of scilab like norm(V) size(U) etc.

Product of vectors:
Dot product and component wise product:

```
-->u
 u  =
    1.     2.     3.
 v  =
    2.     3.     4.

-->u*v
    !--error 10
inconsistent multiplication

-->u.*v
 ans  =
    2.     6.     12.
```

```
-->u*v'
 ans  =
    20.

-->u'*v
 ans  =
    2.     3.     4.
    4.     6.     8.
    6.     9.     12.
```

Other interesting data type is Matrix.
Here is an example of writing matrix in scilab.

```
B=[1,2;3,4]
 B  =


    1.     2.
    3.     4.
```

Now try to write matrix A. such that $A = \begin{bmatrix} 1 & -3 & 2.4 \\ .5 & 0 & 3 \end{bmatrix}$

With Matrix try

- A+B
- 3*A
- A*B
- B*A
- A-B

Introduction
Special Symbols in . . .
User define functions
Rows and Columns
Home Page
Title Page
Page 8 of 27
Go Back
Full Screen
Close
Quit

With matrix, linear algebra can be taught to students. For matrices there are in built commands like

- det(A)
- inv(A)
- size(A)
- length(A)
- spec(A)
- trace(A)
- diag(A)

2. Special Symbols in scilab

```
-->%pi
 %pi  =

    3.1415927
-->%i
 %i  =

    i

-->%e
 %e  =

    2.7182818
-->%inf
 %inf  =

   Inf

-->%eps
 %eps  =
```

```
-->format(5)

 -->%e
 %e   =

    2.72

-->format(20)

-->%e
 %e   =

    2.7182818284590451
```

Other function with matrices are:

- eye(3,3)
```
-->eye(3,3)
 ans  =

    1.    0.    0.
    0.    1.    0.
    0.    0.    1.
```
- zeros(3,2)
- ones(3,2)
- clean(inv(A))
-
- `-->int(10*rand(3,3))`
```
 ans  =

    8.    9.    3.
    6.    2.    2.
    3.    3.    5.
```

3. Rows and Columns

Consider matrix A

```
A  =

    2.    3.    4.
    4.    6.    8.
    6.    9.    12.
```

Now to obtain 1st row:

```
-->A(1,:)
 ans  =

    2.    3.    4.
```

To change $R_1 \longrightarrow 2R_1$

```
-->A(1,:)=2*A(1,:)
 A  =


    4.      6.      8.
    4.      6.      8.
    6.      9.      12.
```

To change $C_1 \longrightarrow 5C_1$

```
-->A(:,1)=5*A(:,1)
 A  =
    20.     6.      8.
    20.     6.      8.
    30.     9.      12.
```

To perform operation as $R_1 \longrightarrow R_1 - R_2$

```
-->A(1,:)=A(1,:)-A(2,:)
 A  =

    0.     0.    0.
    20.    6.    8.
    30.    9.    12.
```

To perform operation as $R_2 \longrightarrow R_2 - \frac{2}{3}R_3$

```
-->A(2,:)=A(2,:)-(2/3)*A(3,:)
 A  =

    0.     0.    0.
    0.     0.    0.
    30.    9.    12.

-->
```

4. Row Reduced Echelon Form

```
-->A=[4 -2 3; 12 2 -7; 20 -2 -1]
 A  =


    4.    - 2.     3.
    12.     2.   - 7.
    20.   - 2.   - 1.

-->B=[3; -2; 4]
 B  =


    3.
  - 2.
    4.
```

```
-->rank(A)
 ans  =

    2.

-->rank([A B])
 ans  =

    2.
```

The system of equations is consistant and it will have infinitely many solutions which can be given in terms of a parameter t.

```
-->rref([A B])
 ans  =

    1.    0.  - 0.25    0.0625
    0.    1.  - 2.    - 1.375
    0.    0.    0.      0.
```

Hence the solution to the system of equations $AX = B$ is given by

$$\begin{aligned} x &= 0.25t + 0.0625 \\ y &= 2t - 1.375 \\ z &= t \end{aligned}$$

For symbolic computation

```
-->x=poly(0,'x')
 x  =

    x

-->A=[x,2*x;x^2,x+3]
 A  =

    x      2x

     2
    x      3 + x

-->det(A)
 ans  =

          2    3
    3x + x - 2x
```

It is also possible to find inverse of symbolic matrix.

```
-->inv(A)
 ans  =

      3 + x            - 2
   -----------      ---------
         2    3               2
   3x + x - 2x      3 + x - 2x

     - x               1
   ---------        ---------
            2                 2
   3 + x - 2x       3 + x - 2x
```

Another way to define polynomial

```
-->poly(V,'x','coeff')
 ans  =

                  2
    2 - 4x + 5x

-->poly([1,-2,3],'y','coeff')
 ans  =

                  2
    1 - 2y + 3y
```

5. Polynomials and their roots

For odd degree polynomial with real coefficient there exists at least one real root. Also for a polynomial with real coefficient, complex roots, if any, occur in conjugate pairs.

Such results can be verified.
This property does not hold if some of the coefficients are complex.

```
-->f=(3+2*%i)*x^3+(7-%i)*x+1
 f  =

real part
                    3
    1 + 7x + 3x
imaginary part

               3
  - x + 2x

-->roots(f)
 ans  =

    0.5663902 + 1.3218189i
  - 0.4269950 - 1.3031021i
  - 0.1393951 - 0.0187168i
```

Define Polynomial by coefficient way:

```
  -->x=poly(0,'x')
 x  =
 
    x
 
-->v=[1,-2,3,4,-7,23]
 v  =
 
    1.  - 2.    3.    4.  - 7.    23.
 
-->f=poly(v,'x','coeff')
 f  =
 
               2    3    4     5
    1 - 2x + 3x + 4x - 7x + 23x
```

```
-->roots(f)
 ans  =

    0.0993290 + 0.5705546i
    0.0993290 - 0.5705546i
  - 0.5850125
    0.3453512 + 0.3198753i
    0.3453512 - 0.3198753i
```

For odd degree polynomial with real coefficients there exists at least one real root.

Thanks!