National University of Singapore
MA2001 Linear Algebra

MATLAB Worksheet 1
Introduction to MATLAB and Solving Linear Systems

A. Installation of MATLAB

In this course, we use a highly acclaimed numerical computing software MATLAB, which stands for *Matrix Laboratory*. The National University of Singapore has a Total Academic Headcount Licence for MATLAB. Students may use it for academic, research, and learning. The license allows each student to install MATLAB on ONE personally-owned computer.

1. Creating MathWorks account

(i) Go to `https://www.mathworks.com/mwaccount/register`.

(ii) Create the account using your NUS email address (e.g., e0012345@u.nus.edu)

(iii) Fill in the rest of the information and follow the instructions to complete the registration process.

2. Connecting to nVPN

(i) If you are NOT using NUS network, you are required to connect to nVPN at the URL: `https://webvpn.nus.edu.sg/`

(ii) Sign in with your NUSNET ID in the format: **nusstu\nusnetid** and password.

(iii) If you are using NUS network, you may ignore this and proceed to download MATLAB.

3. Download MATLAB

(i) As you are allowed to download MATLAB on only one device, you are advised to do so on the PC that you are likely to use during the exam.

(ii) Go to `https://nusit.nus.edu.sg/services/software_and_os/software/software-student` and follow the instructions to download MATLAB.

(iii) It is required to sign in with your NUSNET ID in the format: **nusstu\nusnetid** and password.

(iv) Choose an installation folder in your PC.

(v) Select products to install, include MATLAB and Symbolic Math Toolbox.

(vi) If you encounter any installation problem, contact NUS IT Care at `itcare@nus.edu.sg` or 65162080.

B. MATLAB Commands

MATLAB environment behaves like a super-complex calculator. You can type the commands at the >> command prompt. The answer appears by pressing Enter.

(i) You can type any valid `expression` at the command prompt. The expression can be a number, a matrix, an operation and so on.

```
>>  5
ans =   5
>>  3 + 5
ans =   8
>>  [1 2 3; 4 5 6]
ans =   1   2   3
        4   5   6
```

representing the matrix $\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{pmatrix}$.

★ Note that, to input a matrix, use square brackets `[   ]` instead of round brackets. Within the square brackets, type the entries of the matrix row by row, separating each entry on the same row by a space, and separating every row by a semicolon `;`.

(ii) Usually we will assigns an `expression` to a `variable`:

```
>>  variable = expression
```

For example,

```
>>  a = 11
a =  11
```

Subsequently, the `variable` `a` will store the `expression` `11` (until it is overwritten by another `expression`).

```
>>  3*a
ans =   33
>>  a ^ 2
ans =   121
>>  a = a+4
a =  15
```

In the last command above, the `expression` `11` that was stored in the `variable` `a` is overwritten by the expression `a+4` which is equal to `15`.

★ Note that MATLAB is case sensitive. `a` and `A` represent different variables.

(iii) You may add a semicolon `;` at the end of a command. Then MATLAB will hide the output. For example,

```
>>  a = 4; b = 7;
>>  a + b
ans =  11
>>  c = a * b
c =  28
```

C. Output Formats

(i) By default, MATLAB displays four decimal digits to its answers.

```
>>  sqrt(2)
ans =  1.4142
>>  a=1/2
a =  0.5000
```

(ii) We can change the precision to 16 decimal digits:

```
>>  format long
>>  sqrt(2)
ans =  1.414213562373095
>>  a
a =  0.500000000000000
```

(iii) MATLAB can also display the answers as rational number (approximation):

```
>>  format rat
>>  sqrt(2)
ans =  1393/985
>>  a
a =  1/2
```

★ MATLAB will approximate irrational numbers with rational numbers when you use `format rat`. Sometimes, this may cause unexpected results. Occasionally, an asterisk $*$ may appear when you expect the quantity to be 0.

(iv) To change the display to four decimal digits, you can type `format short` or simply `format`:

```
>>  format short
>>  sqrt(2)
ans =  1.4142
>>  a
a =  0.5000
```

D. Solving Linear System

We input a linear system to MATLAB in terms of an augmented matrix $(\boldsymbol{A} \mid \boldsymbol{b})$. For example, the linear system

$$\begin{cases} 2x_1 - 3x_2 - 7x_3 + 5x_4 + 2x_5 = -2 \\ x_1 - 2x_2 - 4x_3 + 3x_4 + x_5 = -2 \\ 2x_1 - 4x_3 + 2x_4 + x_5 = 3 \\ x_1 - 5x_2 - 7x_3 + 6x_4 + 2x_5 = -7 \end{cases}$$

has augmented matrix

$$(\boldsymbol{A} \mid \boldsymbol{b}) = \left(\begin{array}{ccccc|c} 2 & -3 & -7 & 5 & 2 & -2 \\ 1 & -2 & -4 & 3 & 1 & -2 \\ 2 & 0 & -4 & 2 & 1 & 3 \\ 1 & -5 & -7 & 6 & 2 & -7 \end{array}\right)$$

We input the coefficient matrix $\boldsymbol{A}$ into MATLAB as follow:

```
>>  A = [2 -3 -7 5 2; 1 -2 -4 3 1; 2 0 -4 2 1; 1 -5 -7 6 2]
A =  2  -3  -7  5  2
     1  -2  -4  3  1
     2  0  -4  2  1
     1  -5  -7  6  2
```

and input the constant matrix $\boldsymbol{b}$ as:

```
>>  b = [-2; -2; 3; -7]
b =  -2
     -2
     3
     -7
```

★ To input the column $\boldsymbol{b}$, remember to separate the entries with semicolon `;`.
Then the augmented matrix $(\boldsymbol{A} \mid \boldsymbol{b})$ can be obtained by typing the command `[A b]`.
★ Note that the separator `|` should not be included in the MATLAB command.

```
>>  [A b]
ans =  2  -3  -7  5  2   -2
       1  -2  -4  3  1   -2
       2  0  -4  2  1   3
       1  -5  -7  6  2   -7
```

We shall find the reduced row-echelon form of $(\boldsymbol{A} \mid \boldsymbol{b})$ using the MATLAB command `rref`:

```
>>  rref([A b])
ans =  1  0  -2  1  0  1
       0  1  1  -1  0  2
       0  0  0  0  1  1
       0  0  0  0  0  0
```

★ Note that the pair of round brackets and square brackets must both be included in the above MATLAB command.

Once we obtain the reduced row-echelon form, we can proceed to find the general solution of the linear system by hand. Observe that the 1^{st}, 2^{nd} and 5^{th} columns of the reduced row-echelon form are pivot columns. Set $x_3 = s$ and $x_4 = t$ to be arbitrary parameters, and solve the other variables:

$$x_1 = 2s - t + 1, \quad x_2 = -s + t + 2, \quad x_5 = 1.$$

This gives us the general solution $\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{pmatrix} = \begin{pmatrix} 2s - t + 1 \\ -s + t + 2 \\ s \\ t \\ 1 \end{pmatrix}.$

E. Elementary Row Operations

We seldom need to perform elementary row operations in MATLAB. Nevertheless, for completeness, we shall illustrate how such operations can be performed in MATLAB.

For example,

$$\boldsymbol{A} = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 2 & 3 & 4 & 5 & 6 \\ 3 & 4 & 5 & 6 & 7 \\ 4 & 5 & 6 & 7 & 8 \end{pmatrix}.$$

```
>>  A = [1 2 3 4 5; 2 3 4 5 6; 3 4 5 6 7; 4 5 6 7 8]
ans =  1  2  3  4  5
       2  3  4  5  6
       3  4  5  6  7
       4  5  6  7  8
```

The i^{th} row of $\boldsymbol{A}$ can be extracted using the command `A(i,:)`. For example, to get the 4^{th} row of $\boldsymbol{A}$:

```
>>  A(4,:)
ans =  4  5  6  7  8
```

We can perform the three types of elementary row operations as follows:

(i) Multiplying the i^{th} row by a nonzero constant c: `A(i,:)  = c*A(i,:)`.

```
>>  A(1,:)  = -2*A(1,:)
A =  -2  -4  -6  -8  -10
      2   3   4   5   6
      3   4   5   6   7
      4   5   6   7   8
```

Note that MATLAB returns directly the matrix $\boldsymbol{A}$ with its 1^{st} row changed under this elementary row operation.

(ii) Interchanging the i^{th} and j^{th} rows: `A([i,j],:)  = A([j,i],:)`.

```
>>  A([2,3],:)  = A([3,2],:)
A =  -2  -4  -6  -8  -10
```

```
        3   4   5   6   7
        2   3   4   5   6
        4   5   6   7   8
```

Note that MATLAB returns directly the matrix $\boldsymbol{A}$ with its 2^{nd} and 3^{rd} rows interchanged under this elementary row operation.

(iii) Adding c times of the j^{th} row to the i^{th} row: `A(i,:)  = A(i,:)  + c*A(j,:)`.

```
>>  A(4,:)  = A(4,:)  + 2*A(1,:)
A =   -2   -4   -6   -8   -10
       3    4    5    6    7
       2    3    4    5    6
       40   -3   -6   -9   -12
```

Note that MATLAB returns directly the matrix $\boldsymbol{A}$ with its 4^{th} row changed under this elementary row operation.

F. Practices

1. Enter the following commands in MATLAB window and observe the outputs. Describe what MATLAB has done.

```
>>  x = [1 2 3]
>>  y = [1; 2; 3]
>>  A = [1 2 pi; 0.1 5 6; 7 8 1/4]
>>  format rat
>>  A
>>  format short
>>  A
>>  format long
>>  A
>>  format
>>  A
>>  [A x]
>>  [A y]
```

2. For each of the linear systems in Question 1.16 in the textbook Exercise 1, Use `rref` to find its general solution (if any).