Lab 1: Introduction to MATLAB and MATLAB Functions

Shido Nakajima

01/27/2023

01/27/2023

OBJECTIVE: The objective of this lab is to review some basics of MATLAB.

METHODS: For most of the steps, I did almost exactly how the exercises conducted the steps.

I made a 2x2 matrix "a", which I used to make a 4x4 matrix also named "a" by transposing the original matrix a and adding the transposed a onto a. For exercise 3, I took out second row by using the built-in function ":", which calls for all values of the specified dimension. For exercise 5, I output the text with the variable c simply by not ending the line with ";". For exercise 10, I initially made a 4x4 matrix using zeros function, which I then used for loop to change the values.

For the lab exercise assignment, I initially manually defined matrices A and B, and used eye(3) to define C. To define D, I simply multiplied A with 3; To define E, I transposed A using A'. To define F, I first replaced the 3rd row of B with values 3, which I then used to define F by B(:,end:-2:1) where I started from the last index of B. To define G, I started by taking the first 2x2 of C and then replaced the last index with the value -1, which I then used matrix multiplication with A to get G.

RESULTS: Through the exercises as well as the assignment, I was able to learn the basics of MATLAB, as well as array and matrix manipulation syntax used in MATLAB.

CONCLUSION: This exercise showed that MATLAB is a very strong tool in the manipulation of matrices, especially of those that are too large for calculation with hand. I believe that some of the exercises in mypractice.m can be viewed later in the semester to remind myself on the syntax used in MATLAB.

APPENDIX:

Contents


```
clear;clc;
```

2

```
a = [1 2; 3 4];
a = [a, a'; a',a]
```

```
a =

     1     2     1     3
     3     4     2     4
     1     3     1     2
     2     4     3     4
```

3

```
a(:,2)
```

```
ans =

     2
     4
     3
```

4

```
save myfile.mat;
load myfile.mat
```

5

```
f = 71;
c = (f-32)/1.8;
tempText = "Temperature is " + c + "C"
```

```
tempText =

    "Temperature is 21.6667C"
```

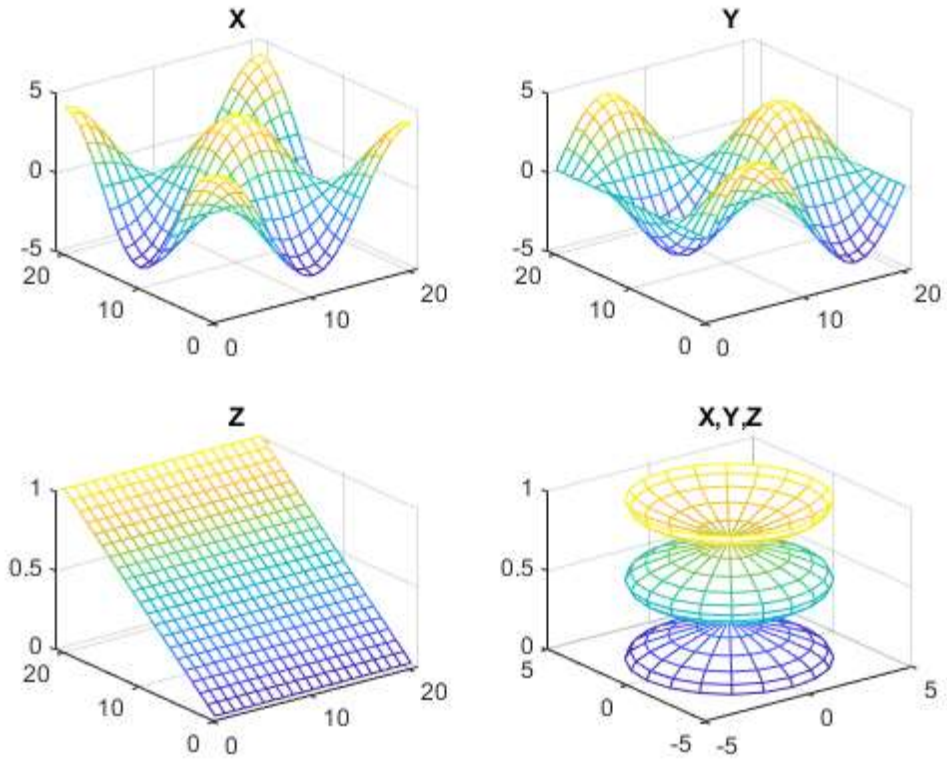
6

```
disp("hello world")
```

```
hello world
```

7

```
t = 0:pi/10:2*pi;
[X,Y,Z] = cylinder(4*cos(t));
subplot(2,2,1); mesh(X); title('X');
subplot(2,2,2); mesh(Y); title('Y');
subplot(2,2,3); mesh(Z); title('Z');
subplot(2,2,4); mesh(X,Y,Z); title('X,Y,Z');
```



8

```
num = randi(100);
if num < 34
    sz = 'low'
elseif num < 67
    sz = 'medium'
else
    sz = 'high'
end
```

```
sz =

    'high'
```

9

```
doc mean
```

10

```
B = zeros(4,4);
for i = 1:16
    B(i) = i;
end
B'
```

```
ans =

     1     2     3     4
     5     6     7     8
     9    10    11    12
    13    14    15    16
```

11

```
mean(B)
```

```
ans =

    2.5000    6.5000   10.5000   14.5000
```

12

```
B(1,4)
```

```
ans =

    13
```

13

```
B(1:2:15)
```

```
ans =

     1     3     5     7     9    11    13    15
```

14 and 15 are summaries

16

```
A = [1,2;3,4];
[A;A(end:-1:1,:)]
```

```
ans =

     1     2
     3     4
     3     4
     1     2
```

17

```
B(3,:) = []
```

```
B =

     1     5     9    13
     2     6    10    14
     4     8    12    16
```

18

```
B*a
```

```
ans =

    51   101    59    93
    58   114    66   106
    72   140    80   132
```

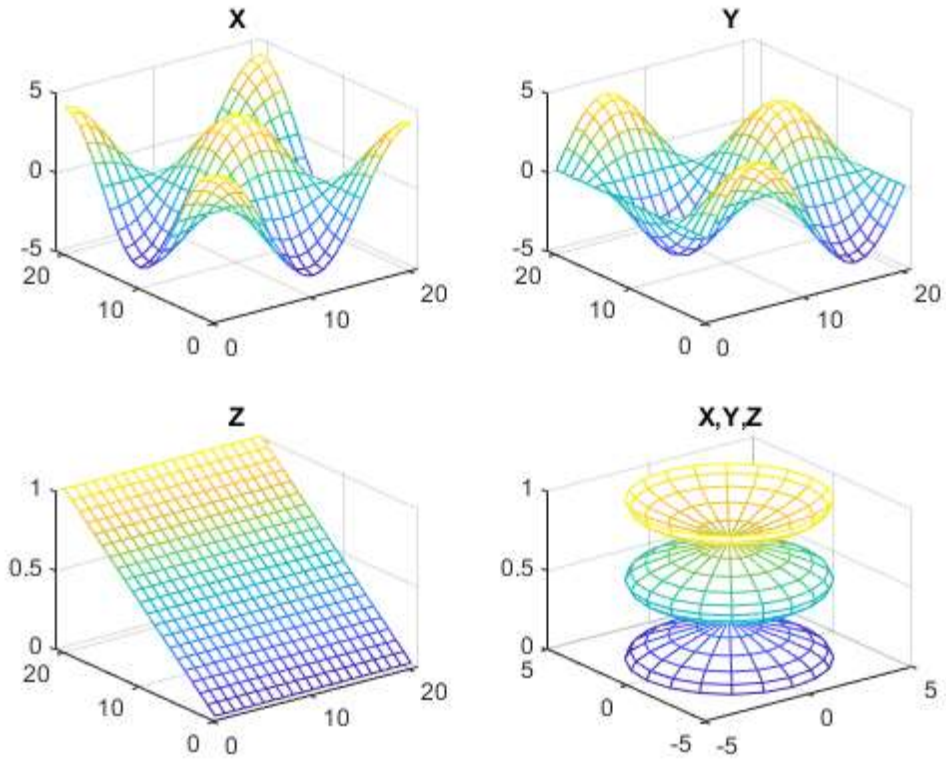
19 is also a summary

20

```
help elfun
```

```
  Elementary math functions.

  Trigonometric.
    sin         - Sine.
    sind        - Sine of argument in degrees.
    sinh        - Hyperbolic sine.
    asin        - Inverse sine.
    asind       - Inverse sine, result in degrees.
    asinh       - Inverse hyperbolic sine.
    cos         - Cosine.
    cosd        - Cosine of argument in degrees.
    cosh        - Hyperbolic cosine.
    acos        - Inverse cosine.
    acosd       - Inverse cosine, result in degrees.
    acosh       - Inverse hyperbolic cosine.
    tan         - Tangent.
    tand        - Tangent of argument in degrees.
    tanh        - Hyperbolic tangent.
    atan        - Inverse tangent.
    atand       - Inverse tangent, result in degrees.
    atan2       - Four quadrant inverse tangent.
    atan2d      - Four quadrant inverse tangent, result in degrees.
    atanh       - Inverse hyperbolic tangent.
```

```
  sec         - Secant.
  secd        - Secant of argument in degrees.
  sech        - Hyperbolic secant.
  asec        - Inverse secant.
  asecd       - Inverse secant, result in degrees.
  asech       - Inverse hyperbolic secant.
  csc         - Cosecant.
  cscd        - Cosecant of argument in degrees.
  csch        - Hyperbolic cosecant.
  acsc        - Inverse cosecant.
  acscd       - Inverse cosecant, result in degrees.
  acsch       - Inverse hyperbolic cosecant.
  cot         - Cotangent.
  cotd        - Cotangent of argument in degrees.
  coth        - Hyperbolic cotangent.
  acot        - Inverse cotangent.
  acotd       - Inverse cotangent, result in degrees.
  acoth       - Inverse hyperbolic cotangent.
  hypot       - Square root of sum of squares.
  deg2rad     - Convert angles from degrees to radians.
  rad2deg     - Convert angles from radians to degrees.

Exponential.
  exp         - Exponential.
  expm1       - Compute exp(x)-1 accurately.
  log         - Natural logarithm.
  log1p       - Compute log(1+x) accurately.
  log10       - Common (base 10) logarithm.
  log2        - Base 2 logarithm and dissect floating point number.
  pow2        - Base 2 power and scale floating point number.
  realpow     - Power that will error out on complex result.
  reallog     - Natural logarithm of real number.
  realsqrt    - Square root of number greater than or equal to zero.
  sqrt        - Square root.
  nthroot     - Real n-th root of real numbers.
  nextpow2    - Next higher power of 2.

Complex.
  abs         - Absolute value.
  angle       - Phase angle.
  complex     - Construct complex data from real and imaginary parts.
  conj        - Complex conjugate.
  imag        - Complex imaginary part.
  real        - Complex real part.
  unwrap      - Unwrap phase angle.
  isreal      - True for real array.
  cplxpair    - Sort numbers into complex conjugate pairs.

Rounding and remainder.
  fix         - Round towards zero.
  floor       - Round towards minus infinity.
  ceil        - Round towards plus infinity.
  round       - Round towards nearest integer.
  mod         - Modulus (signed remainder after division).
  rem         - Remainder after division.
  sign        - Signum.
```



Published with MATLAB® R2022a

Contents


```
clear;clc;
```

1

```
A = [22,5; 1,11; 13,2]
B = [17,9,2,1,3; 6,4,11,9,16]
C = eye(3)
```

```
A =

    22     5
     1    11
    13     2


B =

    17     9     2     1     3
     6     4    11     9    16


C =

     1     0     0
     0     1     0
     0     0     1
```

2

```
D = 3*A
E = A'
```

```
D =

    66    15
     3    33
    39     6


E =

    22     1    13
     5    11     2
```

3

```
B(:,3) = 3
F = B(:,end:-2:1)
```

```
B =

    17     9     3     1     3
     6     4     3     9    16


F =

     3     3    17
    16     3     6
```

4

```
C = C(1:2,1:2)
C(2,2) = -1
G = A*C
```

```
C =

     1     0
     0     1


C =

     1     0
     0    -1


G =

    22    -5
     1   -11
    13    -2
```

Published with MATLAB® R2022a