

Introduction to Matlab

**Random Numbers,
Basic and Advanced Plotting**

Pouyan R. Fard

Dresden, 04.11.2016

Today's Plan

Date	Topics	Exercise/Project
14.10	Kick-off presentation	
21.10	Intro, basic operations Data Handling: vectors, matrices, variables	In-class exercise
4.11	Basic and advanced plotting	In-class exercise
	Scripts and functions	
18.11	Control Flow statements	In-class exercise
	Debugging and integration	
9.12	Data analysis and statistics	In-class exercise Project Distribution
	Sound, images and videos	
20.01	Experimental stimuli and GUI	In-class exercise Project Deadline
	Project Presentation	

Logical Variables

MATLAB operator	Description	Example
>	Greater than	<pre>>> c = a > 3 c = 0 0 1 1 1 >> d = a > b d = 0 0 0 1 1</pre>
>=	Greater than or equal to	<pre>>> e = a >= 3 e = 0 1 1 1 1 >> f = a >= b f = 0 0 1 1 1</pre>
<	Less than	<pre>>> g = b < 3 g = 0 0 0 0 0 >> h = a < b h = 1 1 0 0 0</pre>

<=	Less than or equal to	<pre>>> i = b <= 3 i = 1 0 0 0 0 >> j = a <= b j = 1 1 1 0 0</pre>
==	Equal to	<pre>>> k = a == 9 k = 0 0 0 1 0 >> l = a == b l = 0 0 1 0 0</pre>
~=	Not equal to	<pre>>> k = a ~= 9 k = 1 1 1 0 1 >> l = a ~= b l = 1 1 0 1 1</pre>

Logical Variables

```
>> A(:,1) = 23
```

```
A =
```

```
23 23 23 23
23 6 7 8
23 20 30 40
23 60 70 80
```

```
[1]
```

```
>> find(A(2,:)>=7)
```

```
ans =
```

```
1 3 4
```

```
[1]
```

```
>> A(2,find(A(2,:)>=7)) = 57
```

```
A =
```

```
23 23 23 23
57 6 57 57
23 20 30 40
23 60 70 80
```

```
[1]
```

Logical Variables

- Dealing with NaN values

```
>> S=sum(Meanex)
S =
    NaN

>> Meanex=[2 NaN 12 4 NaN 3 NaN]
Meanex =
    2     NaN     12     4     NaN     3     NaN
>> nanmean(Meanex)
ans =
    5.25

>> S=mean(Meanex (~ (isnan(Meanex) )) )
S =
    5.25
```



Strings

```
>> name ='Andrea';
>> surname = 'Palladio';
>> fullname=[name, ' ', surname]

fullname =
Andrea Palladio
>> whos name surname fullname
  Name      Size   Bytes  Class  Attributes
  fullname  1x15     30    char
  name      1x6      12    char
  surname   1x8      16    char
```

- Useful string functions

```
>> IntString=int2str(25);

>>
numString=num2str(23.4);
```

```
>> strcmp(str1,str2)
ans =
0
>> findstr(str1,'1')
ans =
3      4
```

Strings

- Useful string functions

```
>> RTmean=[0.431,0.321];
>> Pos=char('left', 'right');
>> sprintf('The RT for objects in %s position is %1.1f sec.',...
Pos(1,:),RTmean(1))
ans =
The RT for objects in left position is 0.4 sec.
```

```
>> disp(str3);
hello
help
>> disp([1 2; 3 4]);
  1   2
  3   4
>> input('How old are you? ')
How old are you? 35
ans =
 35
```

Cell Arrays

- Useful for making complex data variables

```
>> SoundInf={'sine','square','Sting'}
SoundInf =
    'sine' 'square' 'Sting'
>> whos SoundInf
  Name      Size   Bytes  Class     Attributes
  SoundInf  1x3     210    cell
>> SoundInf(2,3) = {[5, 6; 7, 8] }
SoundInf =
    'sine'  'ramp'  'sting'
    []      []      [2x2 double]
>> SoundInf{2,3}=5
SoundInf =
    'sine'  'ramp'  'sting'
    []      []      [      5]
>> SoundInf{3,3}=[2 3; 7 8]
SoundInf =
    'sine'  'ramp'  'sting'
    []      []      [      5]
    []      []      [2x2 double]
>> SoundInf{2,3}(2,2)
ans =
    8
>> cellplot SoundInf
```

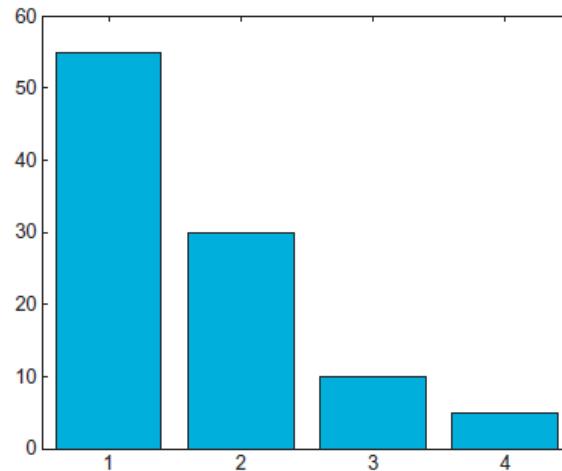
Import/Export

```
>> clear all;                                >> load test1
>> a=[4,2,1,43; 2,5,1,6];                  >> whos
>> b= sin(a)+5;                            Name    Size    Bytes  Class      Attributes
>> f='test';                               a        2x4       64  double
>> save test1                             b        2x4       64  double
>> clear all;                            f        1x4        8   char
.                                         >> save ('test1','RT1');
.                                         .
```

File format	File content	Extension	Functions
MATLAB formatted	Saved MATLAB workspace	.mat	load, save
Text	Text	.txt	textread
	Comma-separated numbers	.csv	Csvread csvwrite
Extended markup language	XML-formatted text	.xml	Xmlread xmlwrite
Spreadsheet	Excel worksheet	.xls	Xlsread xlswrite

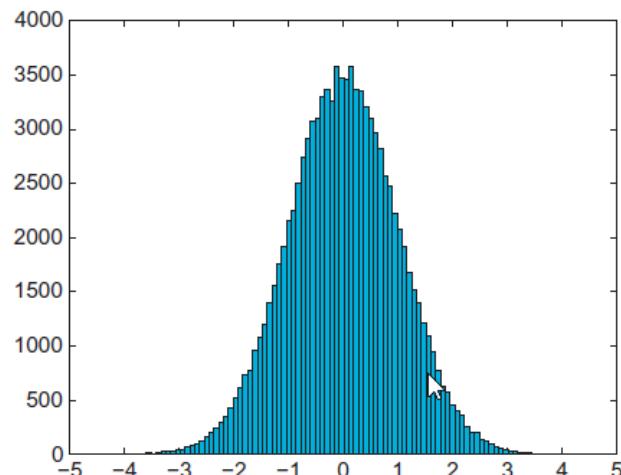
Bar Plots / Histograms

```
>> results = [55  30  10  5]  
results =  
    55  30  10  5  
>> bar(results)
```



```
>> suspicious = randn(100000,1);  
>> figure  
>> hist(suspicious, 100)
```

↳



Random Numbers

- Uniform Distribution

```

>> rand(2)

ans =

    0.1334    0.8875
    0.2043    0.3274

>> r = -5 + (5+5)*rand(1,6)

r =

    1.6665    1.0341   -1.2983    0.3298   -1.8825   -3.1169

>> r = -5 + (5+5)*rand(1,10000);
>> hist(r)
I

>> X=rand(size(r))

X =

    0.8594    0.4519    0.5600    0.6888    0.7874    0.7995
  
```

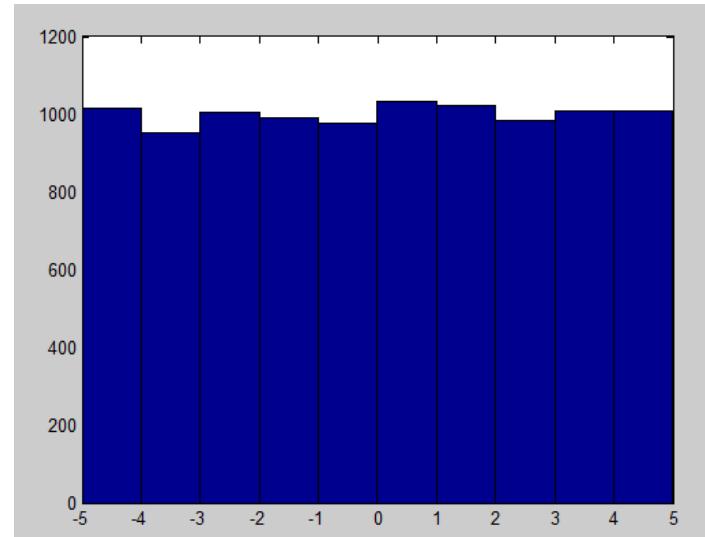
- Pseudo-random uniformly distributed integers

```

>> r1 = randi(10,1,5)

r1 =

    10     8     8     2     9
  
```



Random Numbers

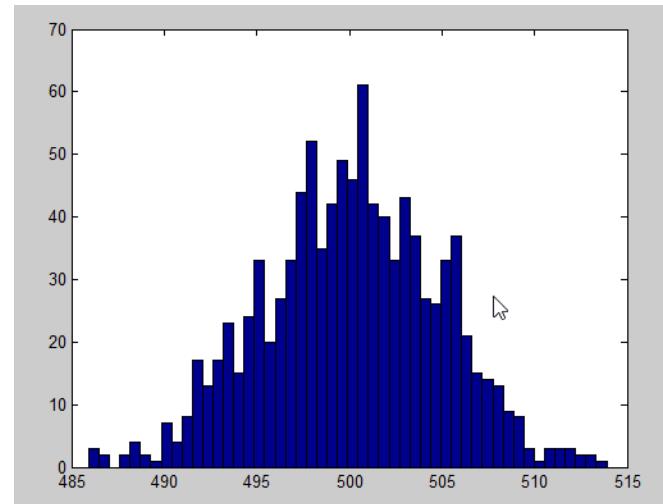
- Normal Distribution

```
>> randn(4)
```

ans =

-0.1431	0.6694	-1.3309	0.1598
0.7559	1.1854	1.0146	-0.2884
0.1325	0.4229	0.7457	-1.0838
-1.2572	-1.3795	-0.7116	1.3335

```
>> a = 5;  
b = 500;  
y = a.*randn(1000,1) + b;  
hist(y,50)
```



Random Numbers

- Normal Distribution

```
>> y=normrnd(5,10,[1,10])
```

```
y =
```

```
3.1124    7.1382    3.4948    3.4693    11.4685    3.2389    -0.4196    -0.0679   -15.7383    9.8145
```

```
>> y=normrnd(500,5,[1,1000]);  
hist(y,50)
```

```
>> mean(y)
```

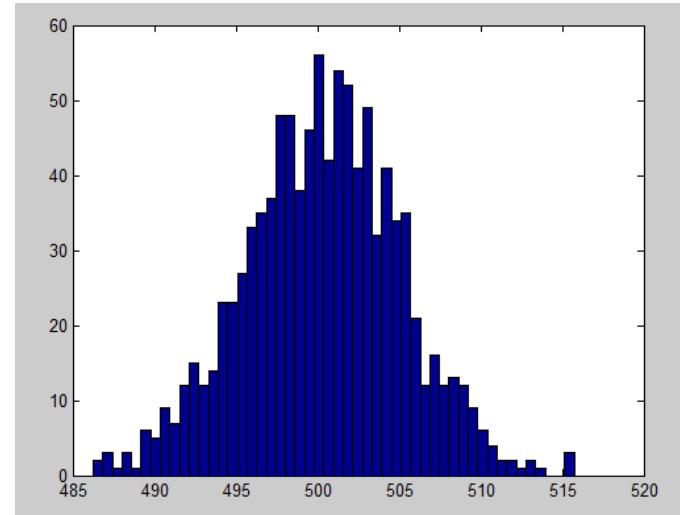
```
ans =
```

```
500.2999
```

```
>> std(y)
```

```
ans =
```

```
4.7836
```



Random Numbers

```
>> x1 = random('Normal', 0, 1, 2, 4)

x1 =

-1.2482    -0.7538     0.9198     0.5396
 0.1622    -0.4963    -0.9311    -0.1406
```

```
>> x2 = random('Poisson', 1:6, 1, 6)
```

```
x2 =

 2      1      1      3      6      8
```

- See MATLAB help for more details.

Random Numbers

```
>> x1 = random('Normal', 0, 1, 2, 4)

x1 =

-1.2482    -0.7538     0.9198     0.5396
 0.1622    -0.4963    -0.9311    -0.1406
```

```
>> x2 = random('Poisson', 1:6, 1, 6)
```

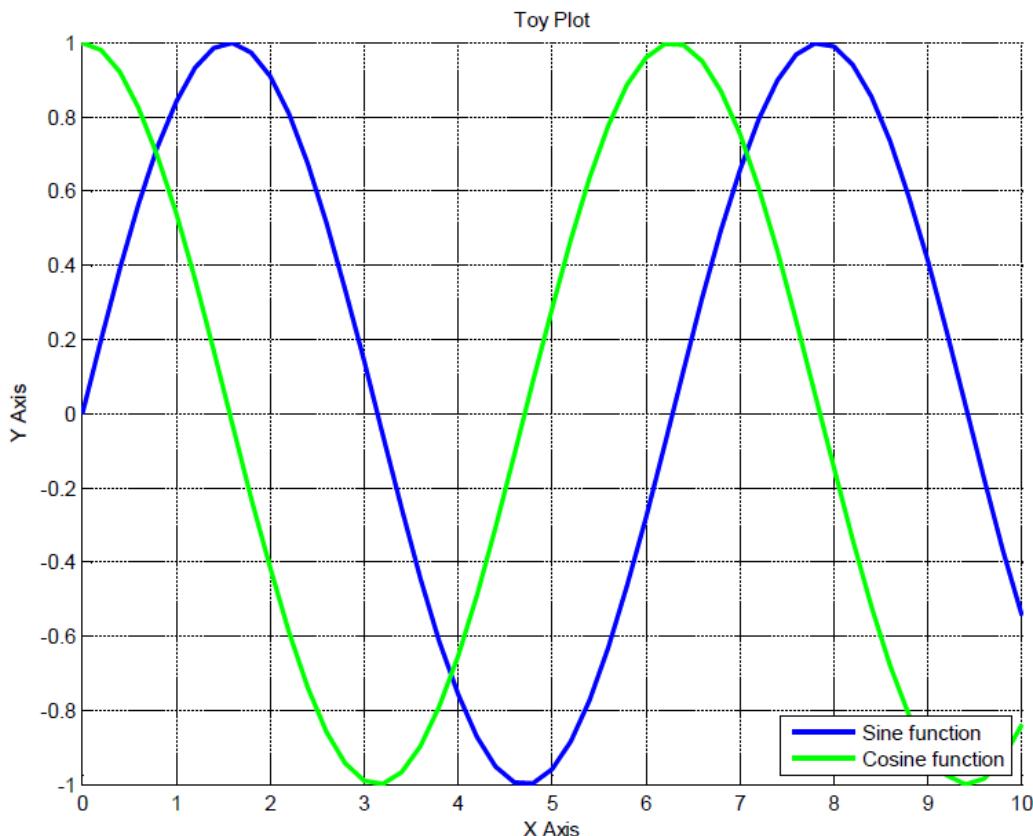
```
x2 =

 2      1      1      3      6      8
```

- See MATLAB help for more details.

Basics of Plotting

```
>> x=0:0.2:10;
y1=sin(x);
y2=cos(x);
plot(x,y1,'b','LineWidth',2);
hold on;
plot(x,y2,'g','LineWidth',2);
xlabel('X Axis');
ylabel('Y Axis');
title('Toy Plot');
legend('Sine function','Cosine function','Location','Southeast');
grid on
box off
```



Subplots

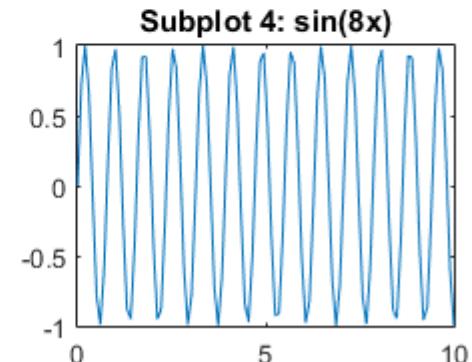
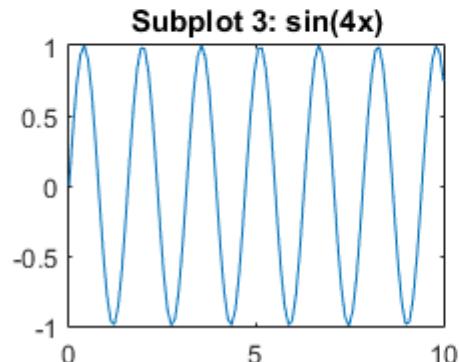
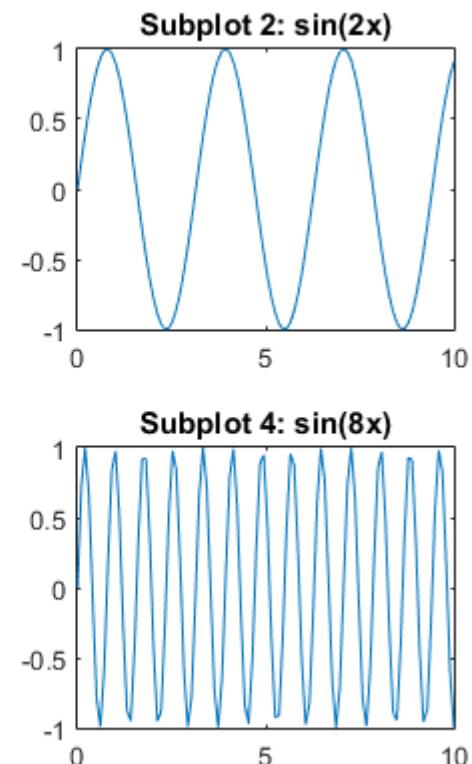
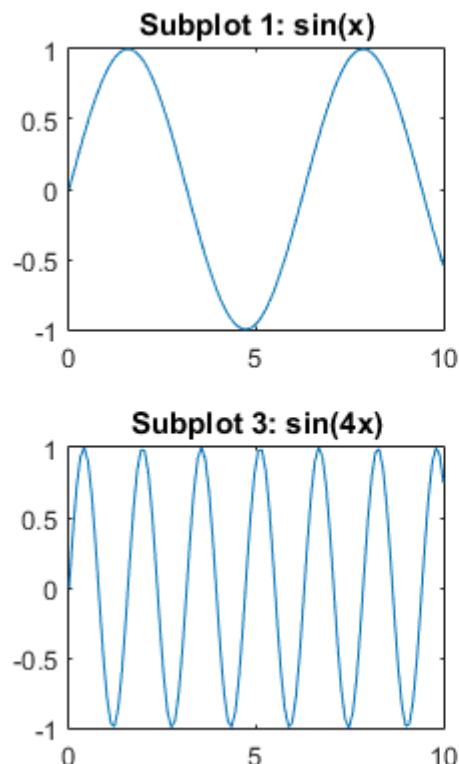
```
x = linspace(0,10);
y1 = sin(x);
y2 = sin(2*x);
y3 = sin(4*x);
y4 = sin(8*x);

figure
subplot(2,2,1)
plot(x,y1)
title('Subplot 1: sin(x)')

subplot(2,2,2)
plot(x,y2)
title('Subplot 2: sin(2x)')

subplot(2,2,3)
plot(x,y3)
title('Subplot 3: sin(4x)')

subplot(2,2,4)
plot(x,y4)
title('Subplot 4: sin(8x)')
```



Subplots

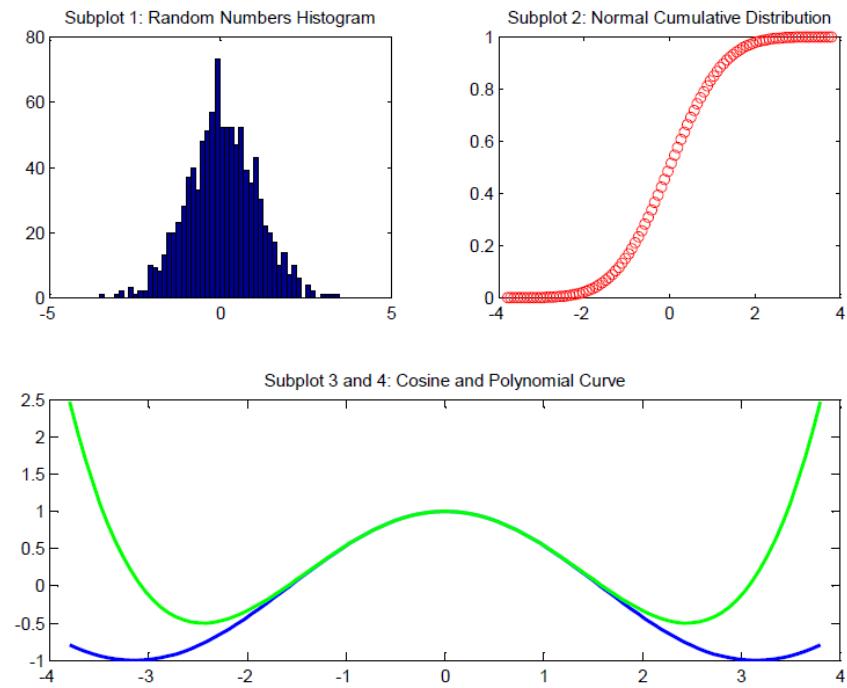
```

x = linspace(-3.8,3.8);
figure
subplot(2,2,1);
hist(normrnd(0,1,[1,1000]),50);
title('Subplot 1: Random Numbers Histogram','LineWidth',2)
xlim([-5 5])

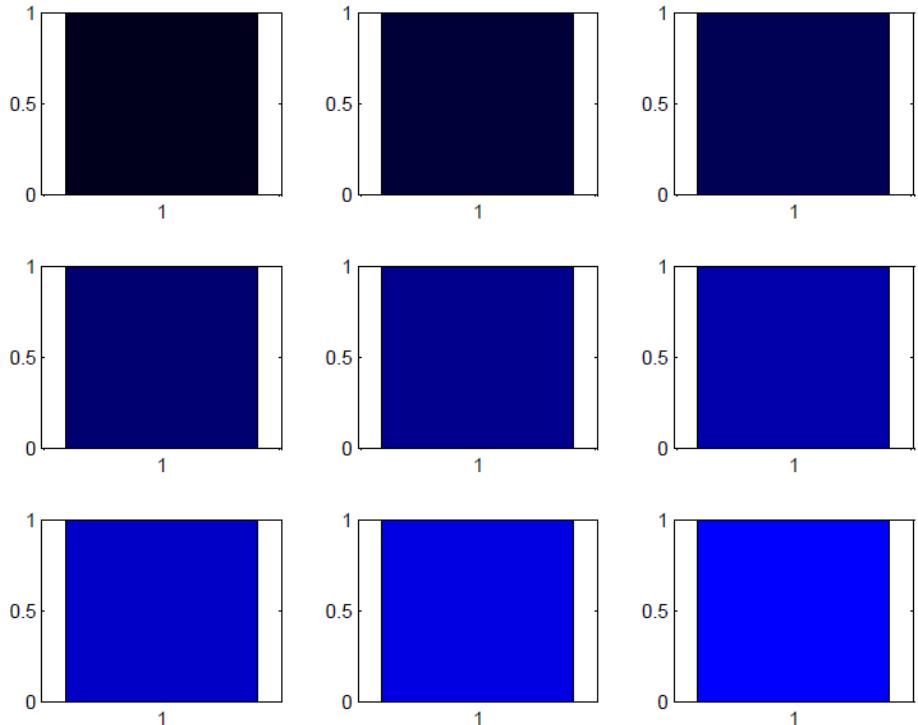
subplot(2,2,2);
plot(x,normcdf(x,0,1),'ro');
title('Subplot 2: Normal Cumulative Distribution','LineWidth',1.5)

y_cos = cos(x);
y_poly = 1 - x.^2./2 + x.^4./24;
subplot(2,2,[3,4]);
plot(x,y_cos,'b',x,y_poly,'g','LineWidth',2);
title('Subplot 3 and 4: Cosine and Polynomial Curve')

```



Subplots

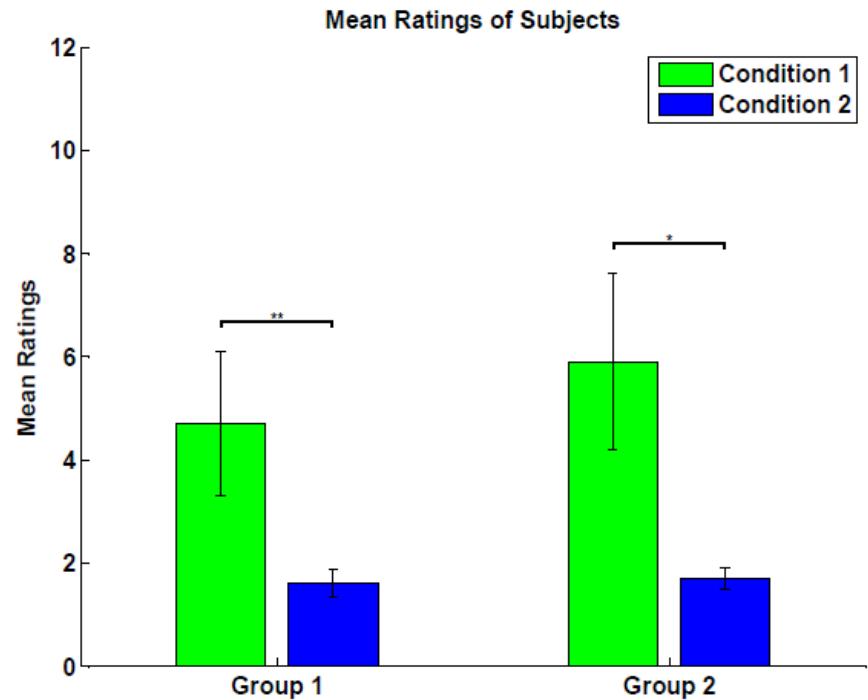


```
>> figure %Open a new figure
for ii=1:9 %Start loop, have counter ii run from 1 to 9
subplot(3,3,ii) %Draw into the subplot ii, arranged in 3 rows, 3 columns
h=bar(1,1); %This is just going to fill the plot with a uniform color
set(h,'FaceColor',[0 0 ii/9]); %Draw each in a slightly different color
end %End loop
```

Exercise: Multi-subject Data Analysis

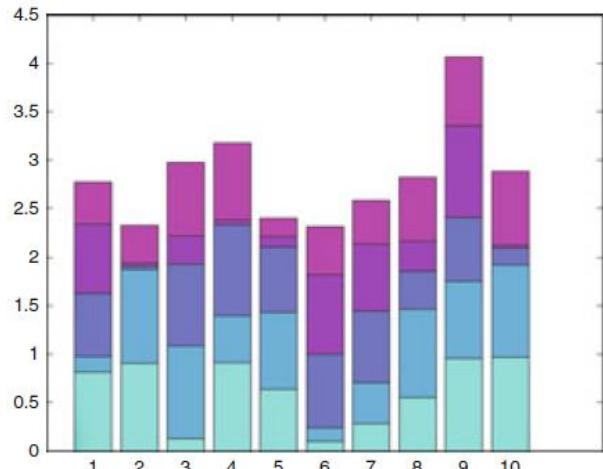
- Generate the results of a psychophysical experiment with 5 subjects giving ratings in a specific task for 80 trials using uniform random distribution.
- Reproduce Plots like below. Error bars should indicate standard error.
- Use barwitherr and sigstar functions provided to you.

```
>> barwitherr(errors, means)  
>> sigstar(groups_x_axis,p_values);
```

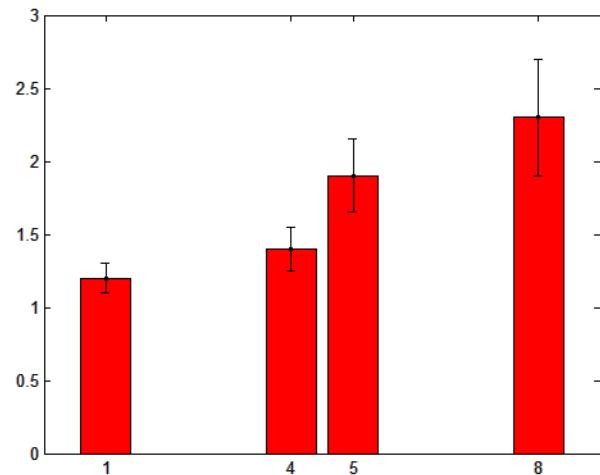


Bar Plots

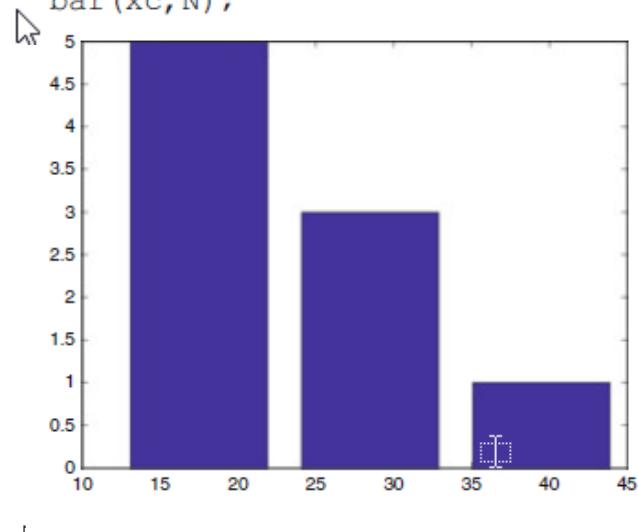
```
>> bar(rand(10,5), 'stacked');
>> colormap(cool);
```



```
x=[1,4,5,8];
RT=[1.2,1.4,1.9,2.3];
SD=[0.1,0.15,0.25,0.4];
bar(x,RT,'w'); hold on;
errorbar(x,RT,SD,'.k');
```

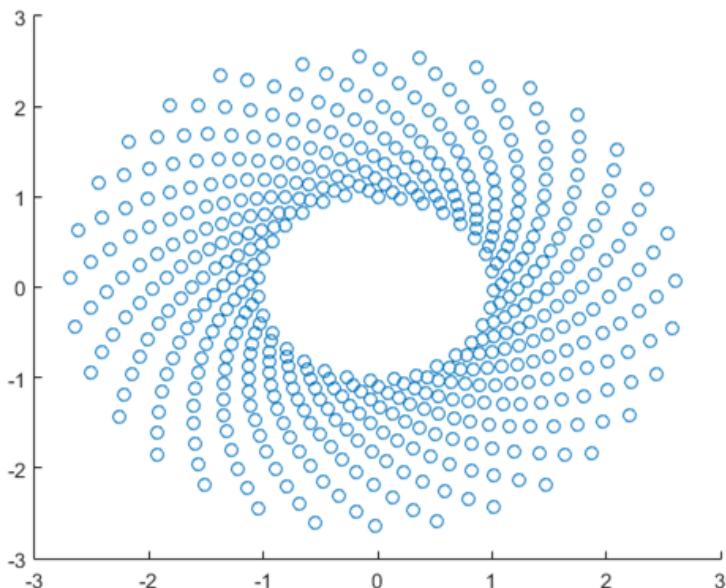


```
Ages=[22,25,23,22,45,12,34,33,21];
[N, xc]=hist(Ages,3);
bar(xc,N);
```

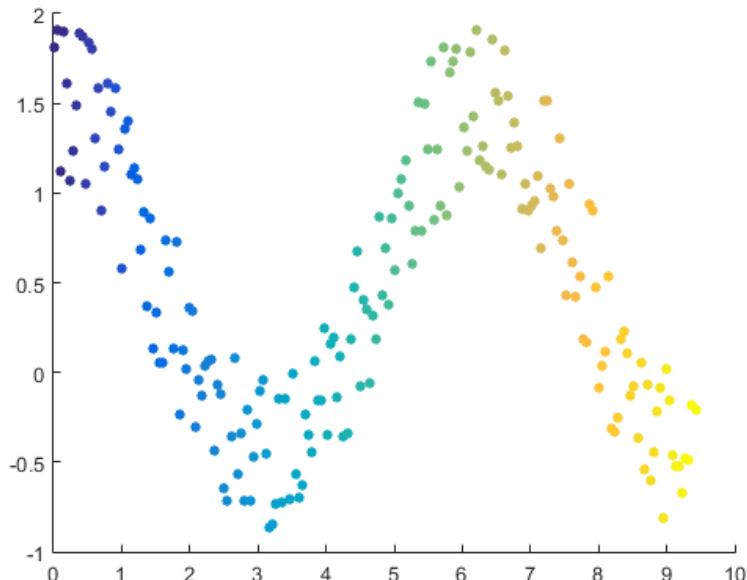


Scatter Plots

```
theta = linspace(0,1,500);
x = exp(theta).*sin(100*theta);
y = exp(theta).*cos(100*theta);
s = scatter(x,y);
```

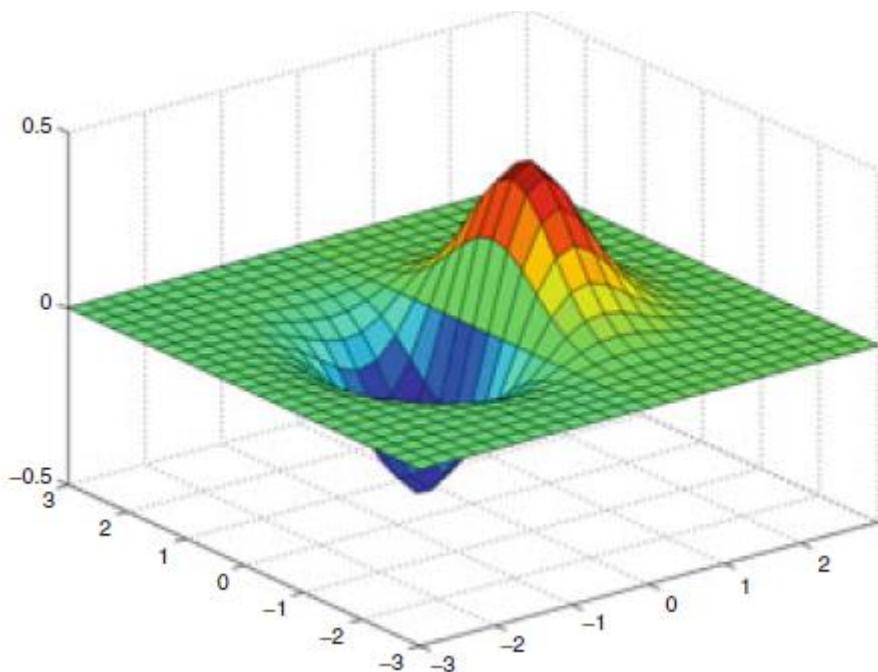


```
x = linspace(0,3*pi,200);
y = cos(x) + rand(1,200);
a = 25;
c = linspace(1,10,length(x));
scatter(x,y,a,c,'filled')
```

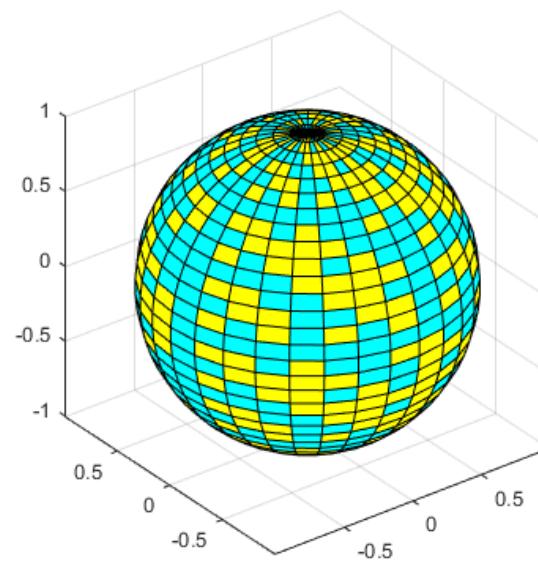


3D Plotting

```
>> a=[-3:0.25:3];  
>> b=[-3:0.25:3];  
>> [X, Y]=meshgrid(a,b);  
>> Z= X.*exp(-X.^2-Y.^2);  
>> surf(X, Y, Z);
```

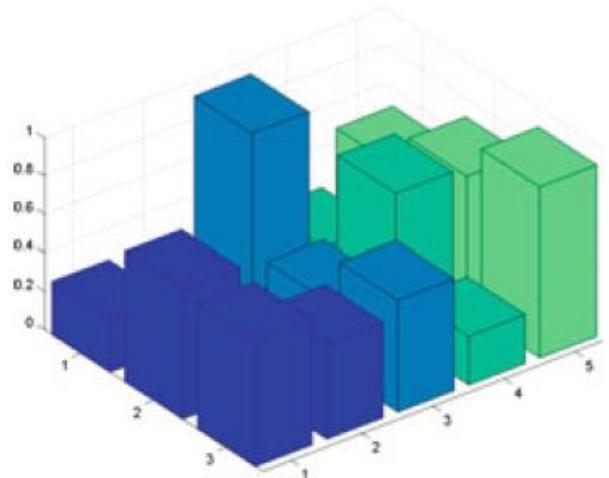


```
k = 5;  
n = 2^k-1;  
[x,y,z] = sphere(n);  
c = hadamard(2^k);  
  
figure  
surf(x,y,z,c);  
colormap([1 1 0; 0 1 1])  
axis equal
```

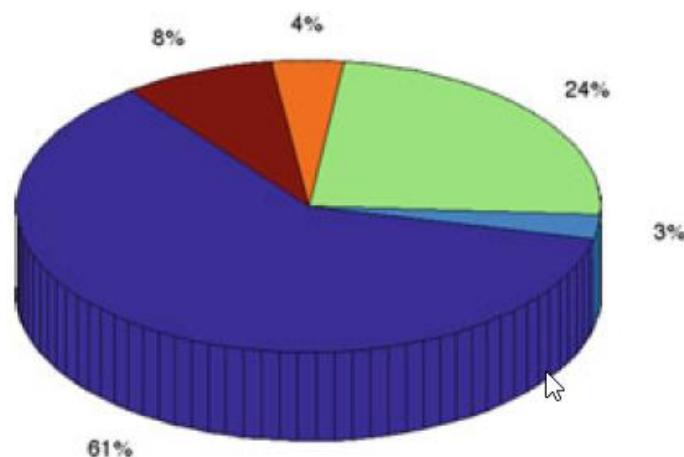


3D Plotting

```
>> y=rand(3,5);  
>> bar3(y);  
>> colormap(winter);
```



```
>> y=rand(5,1);  
>> pie3(y);  
>> axis square; grid off;
```

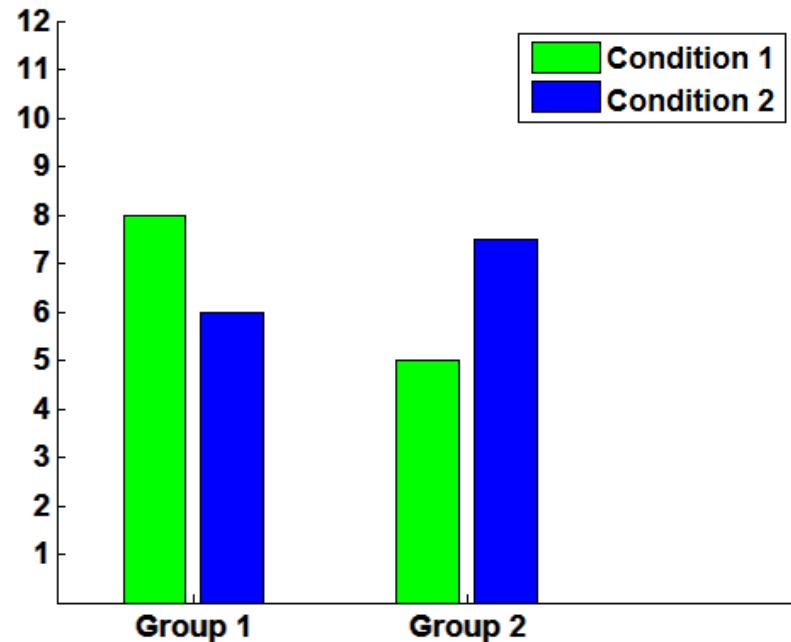
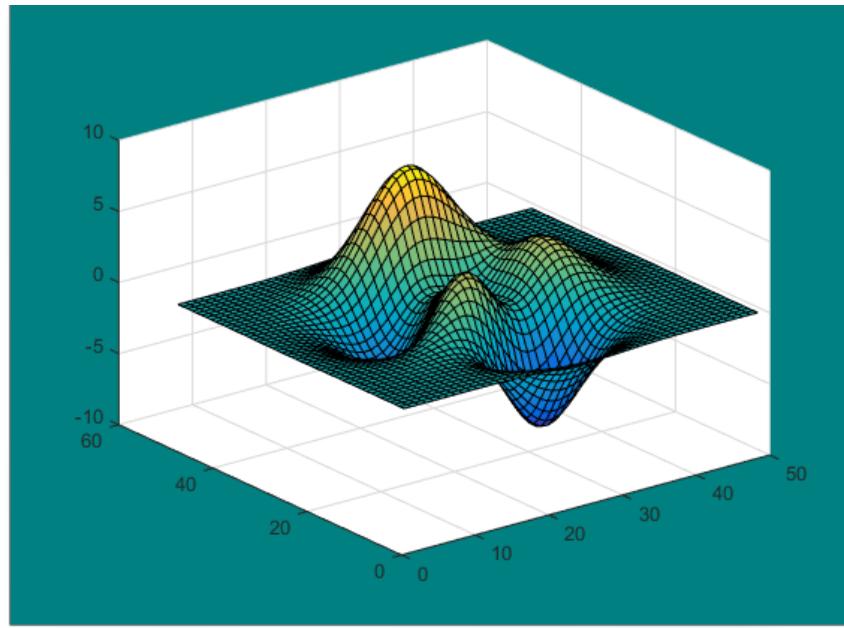


Graphics Handles

```
surf(peaks)
fig = gcf; % current figure handle
fig.Color = [0 0.5 0.5];
figToolBar = 'none';
```



```
h=bar(data);
set(gca, 'FontWeight', 'Bold', 'FontSize', 14);
set(gca, 'XTickLabel', {'Group 1', 'Group 2'});
set(gca, 'YTick', 1:12);
set(h(1), 'FaceColor', 'g', 'LineWidth', 1.2);
set(h(2), 'FaceColor', 'b', 'LineWidth', 1.2);
```



References

- **MATLAB for Psychologists (2012)**, Borgo, M., Soranzo, A., Grassi, M., Springer-Verlag, 2012, ISBN. 978-1-4614-2196-2.
 - Chapter 2-4. pp. 25-82.
- **MATLAB for Neuroscientists, 2nd Ed: An Introduction to Scientific Computing (2014)**, Wallisch, P., Lusignan, M.E., Benayoun, M.D., Baker, T.I., Dickey, A.S. and Hatsopoulos, N.G., Academic Press, ISBN. 978-0123838360.
 - Chapter 2-3. pp. 7-114.
- **MATLAB help:**
 - <http://www.mathworks.com/help/matlab/random-number-generation.html>
 - <http://www.mathworks.com/help/stats/random.html>
 - <http://www.mathworks.com/help/matlab/ref subplot.html>
 - <http://www.mathworks.com/help/matlab/ref subplot.html>
 - <http://www.mathworks.com/help/matlab/ref surf.html>
 - <http://www.mathworks.com/help/matlab/ref scatter.html>
 - <http://www.mathworks.com/help/matlab/ref gcf.html>