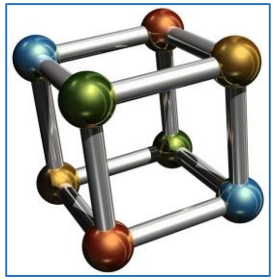
3차원 그래프


매트랩 이해 및 실습

최병조
임베디드시스템공학과

강의 주제

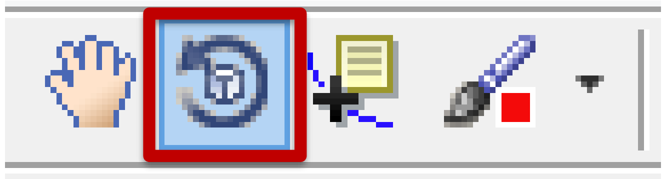
• Matlab의 역사와 간단한 사용법	• 다항식, 커브 피팅, 인터폴레이션
• 배열, 행렬 만들기와 소리 다루기	• **3차원 그래프 그리기**
• 행렬과 그림 다루기	• GUIDE로 GUI 만들기
• 라이브스크립트, 웹 게시, 엑셀 연동	• 애니메이션 GUI
• 2차원 그래프 그리기 기초	• 앱 디자이너로 GUI 만들기
• 다양한 2차원 그래프 그리기	• GUI 프로젝트 발표
• 함수 만들기	• MuPAD로 수학 문제 풀기
• 중간고사	• 기말고사

You will be able to

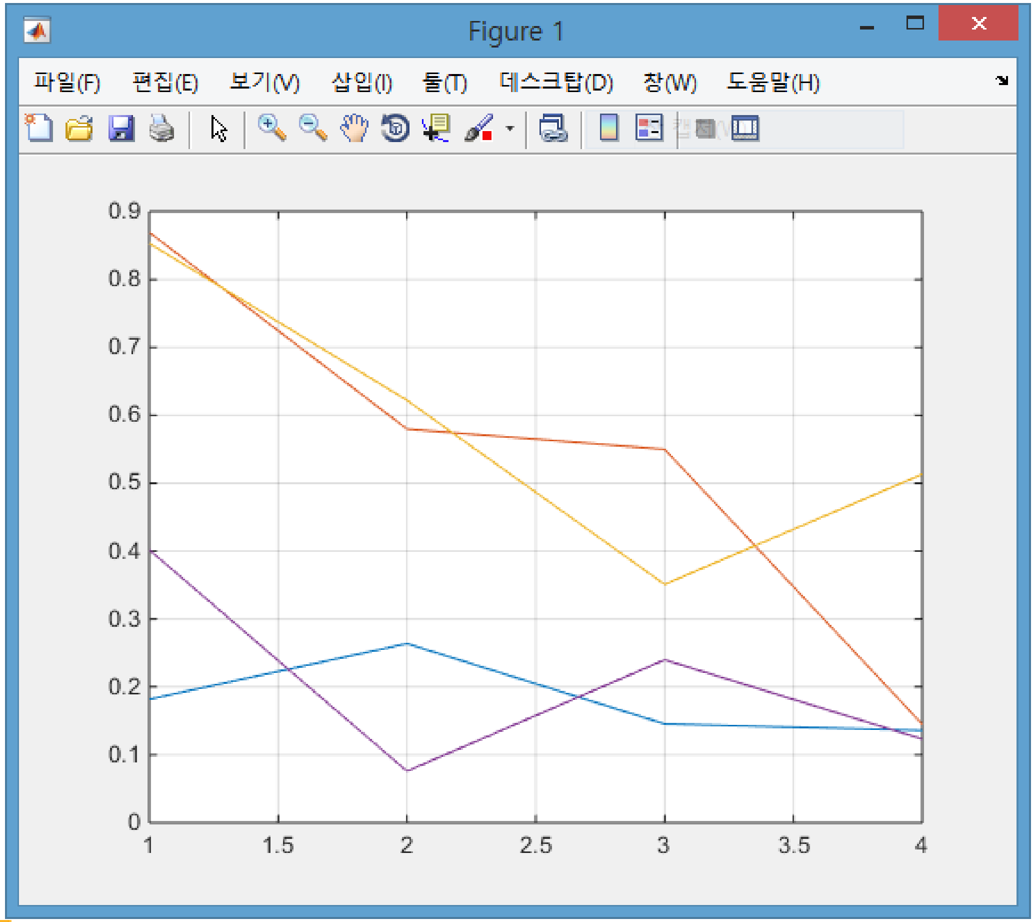
- Choose the right technique to visualize 3D data,
- Plot 3D surface plots with various properties, and
- Visualize volume data using iso-surfaces and slices.

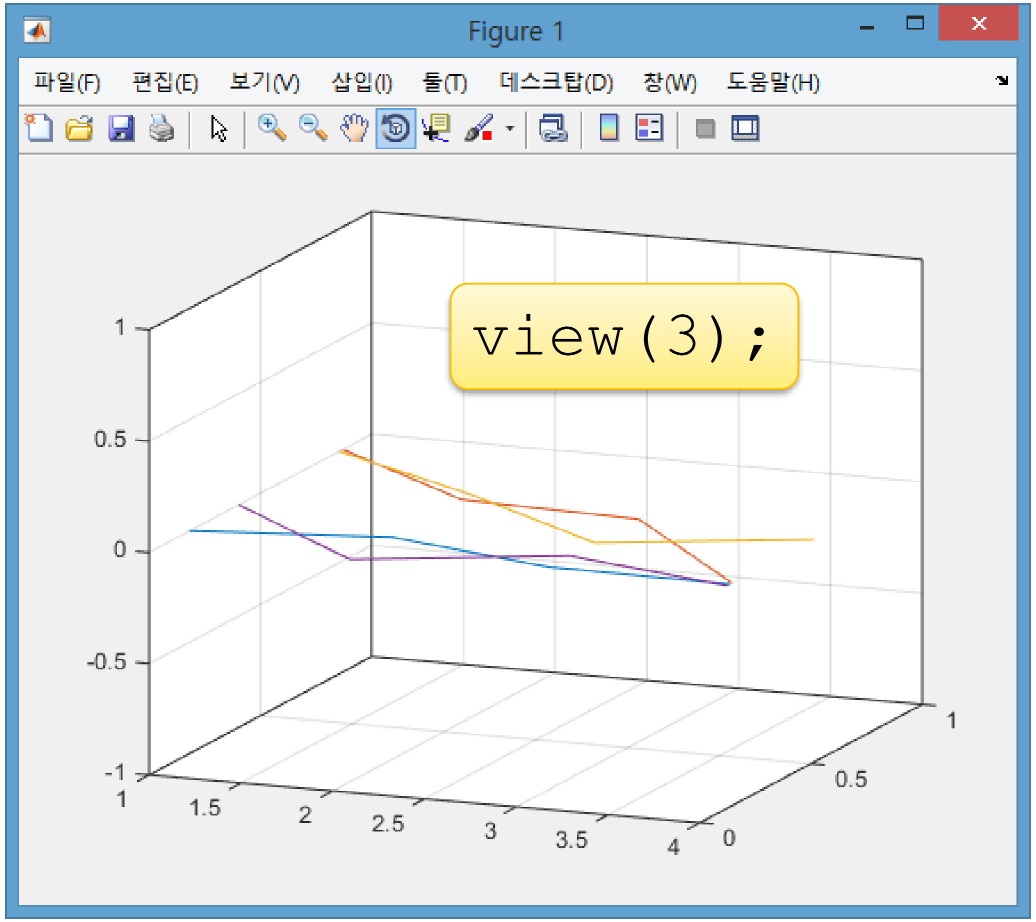
- All the scripts are available at
  - https://goo.gl/45vZGJ

2D Graph on 3D Space



- `figure(1), plot(rand(4)), grid on;`





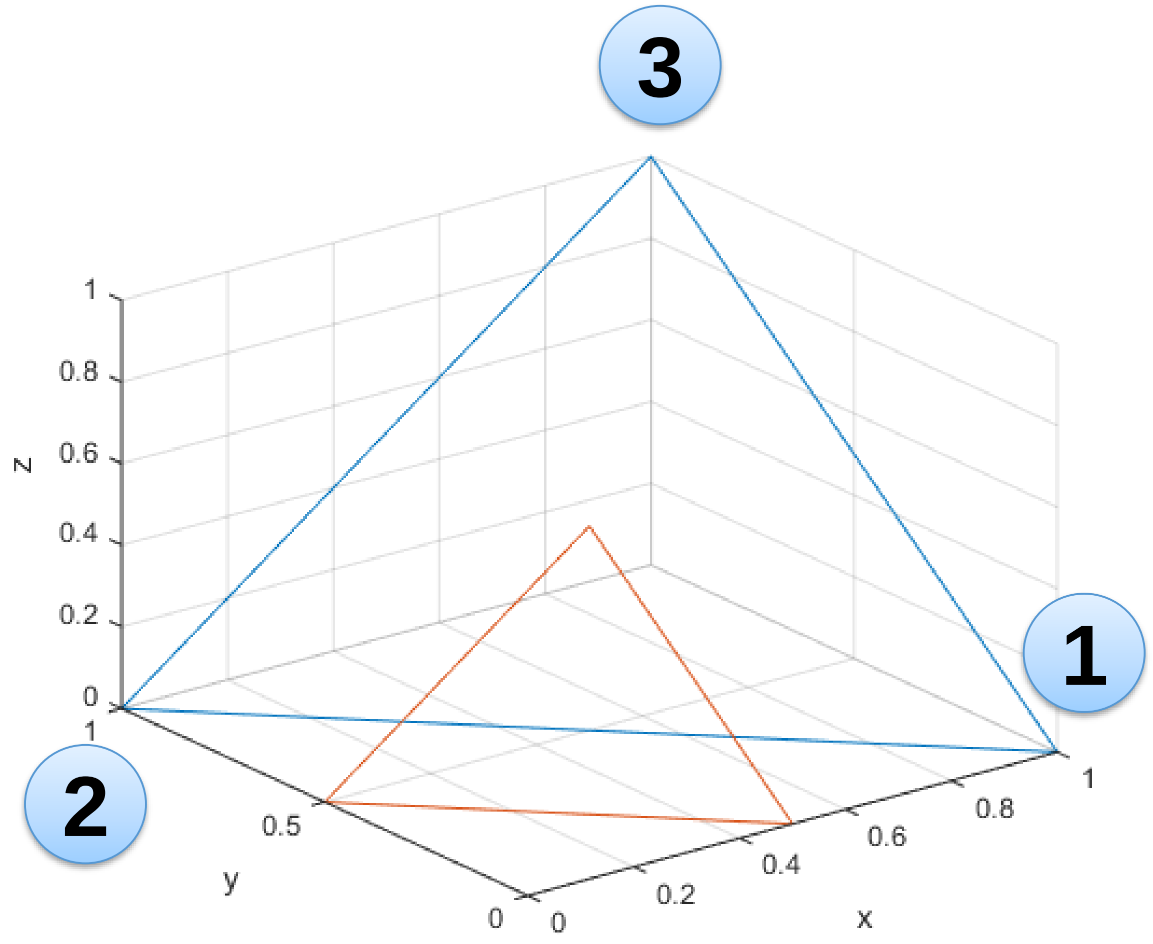
3D Line Plot

- **plot3(x, y, z)**

m10_plot3_triangle.m

```
%% 3차원 공간의 좌표
x = [1 0 1 1]';
y = [0 1 1 0]';
z = [0 0 1 0]';

%% 첫 번째 삼각형
figure(1);
plot3( x, y, z ); grid on;
xlabel('x'); ylabel('y'); zlabel('z');

%% 두번째 삼각형
hold on;
plot3( 0.5*x, 0.5*y, 0.5*z );
hold off;
```

1 → 2 → 3 → 1



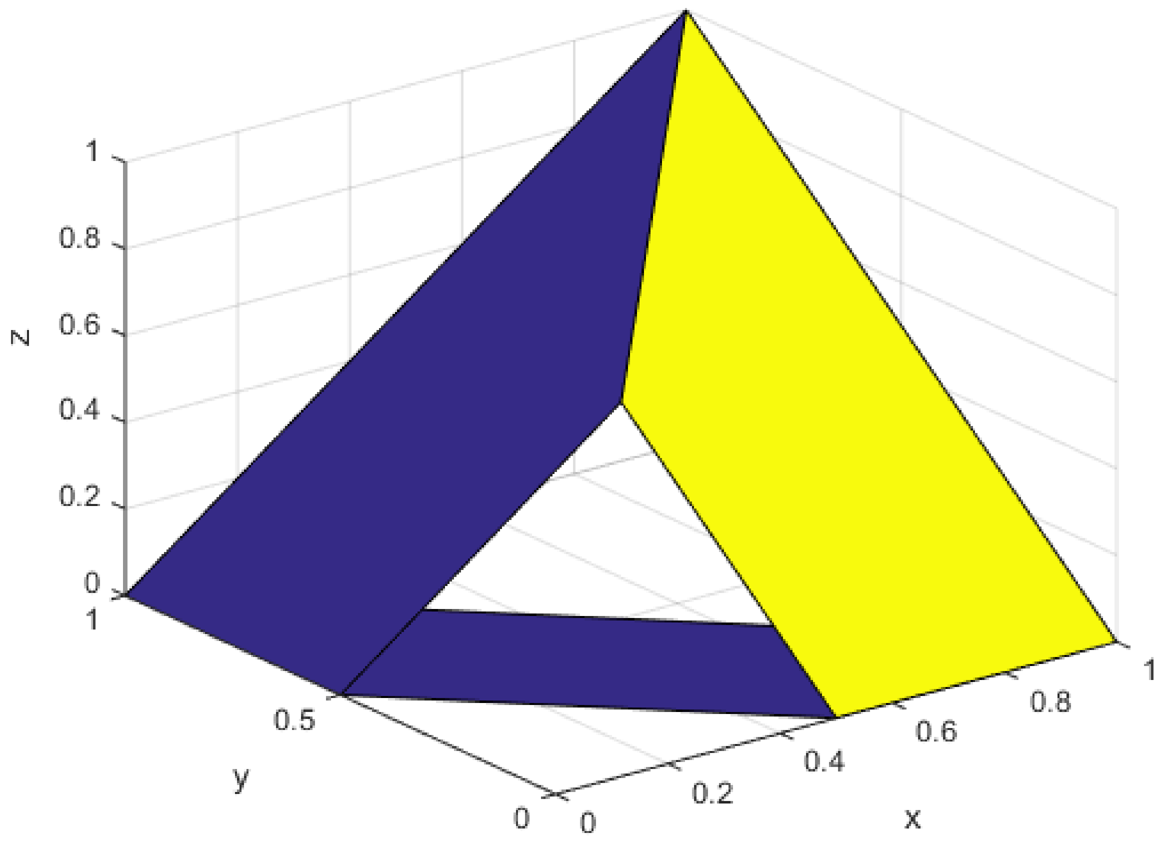
3D Surface Plot

- **surf(X, Y, Z)**

m10_surf_triangle.m

```
%% 첫 번째 삼각형의 좌표
x1 = [1 0 1 1]';
y1 = [0 1 1 0]';
z1 = [0 0 1 0]';

%% 면의 좌표
X = [x1, 0.5*x1];
Y = [y1, 0.5*y1];
Z = [z1, 0.5*z1];

%% 두 삼각형을 잇는 면
figure(1);
surf( X, Y, Z );
grid on;
xlabel('x'); ylabel('y'); zlabel('z');
```

2-column matrix

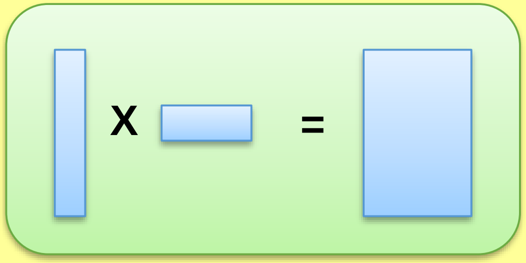


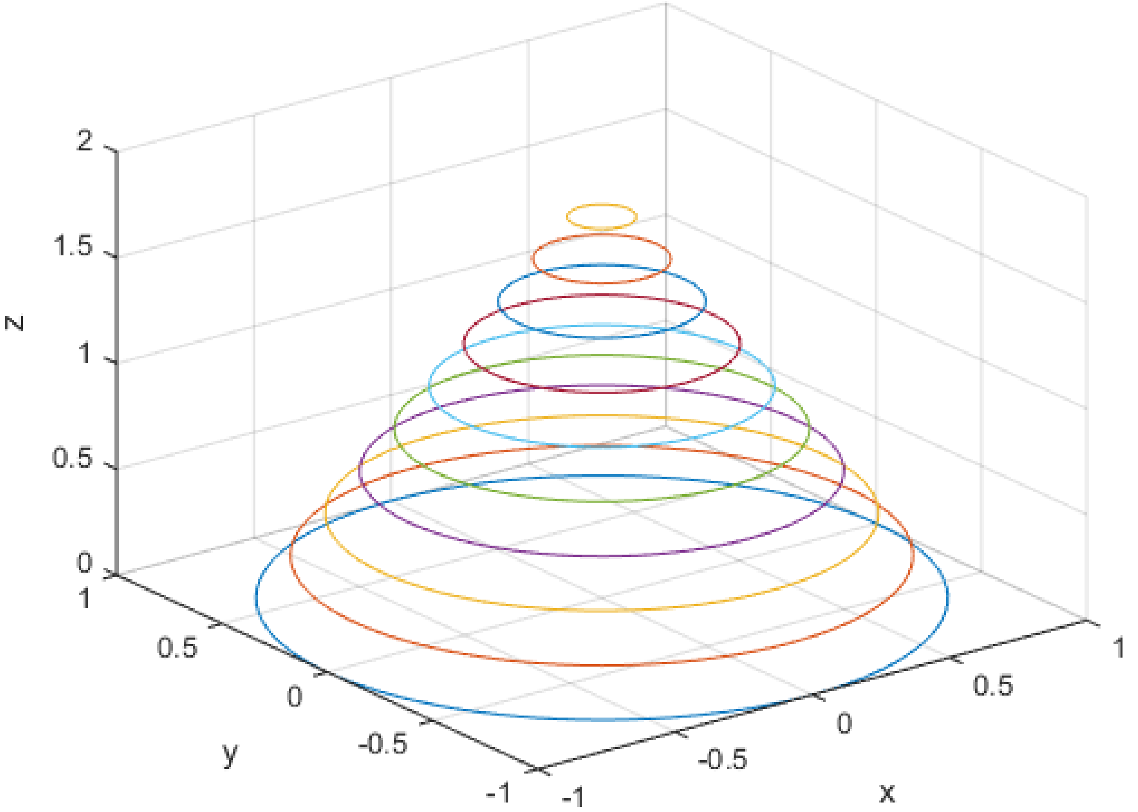
Circles on a Cone

- Each column in a Matrix represents each plot.

m10_plot3_cone.m

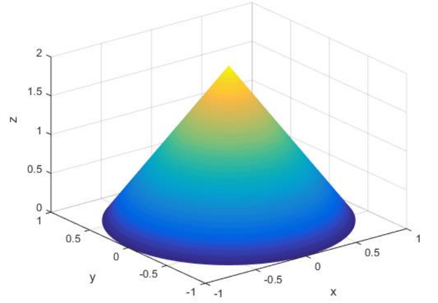
```
%% 3차원 공간에서 원의 좌표
t = linspace(0, 2*pi, 100)';
x = cos(t);
y = sin(t);
z = ones(size(x));
r = (1:-0.1:0);
h = 2*(0:0.1:1);
X = x * r;
Y = y * r;
Z = z * h;
%% 3차원 그래프
figure(1);
plot3( X, Y, Z );
grid on;
xlabel('x'); ylabel('y'); zlabel('z');
```





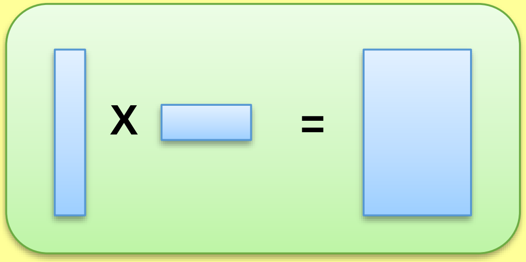
Cone

- Each column in a Matrix represents each plot.

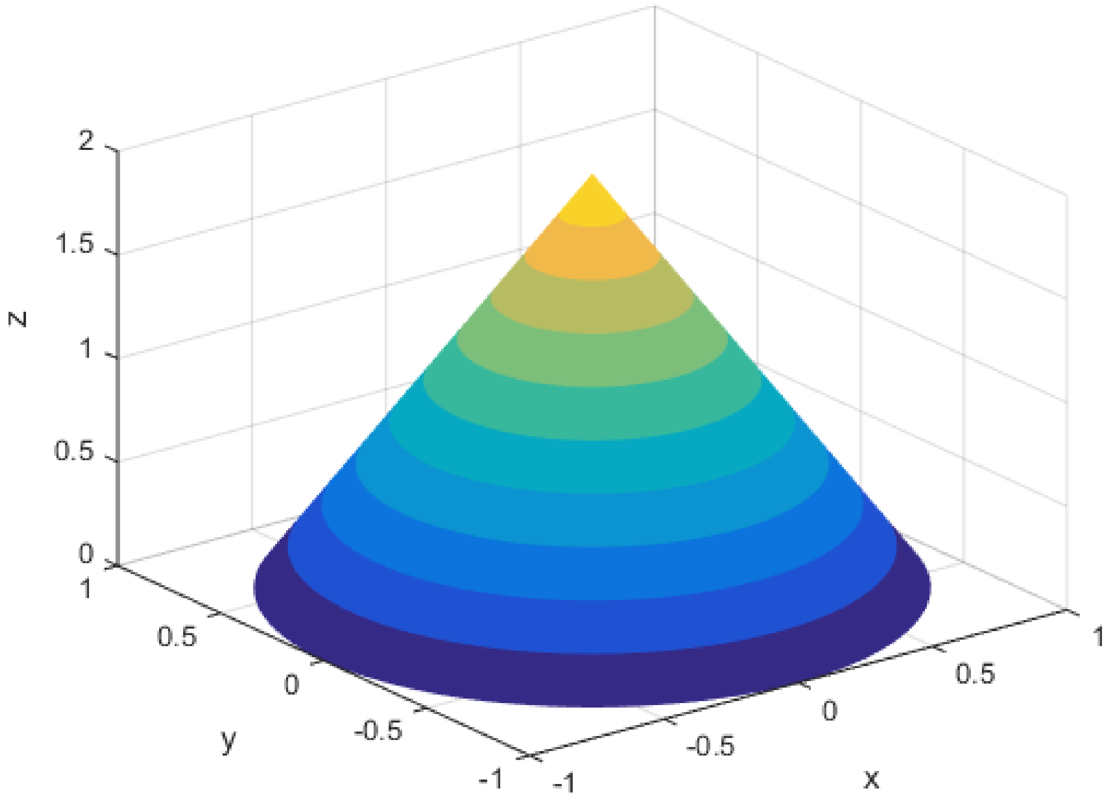


m10_surf_cone.m

```
%% 3차원 공간에서 원의 좌표
t = linspace(0, 2*pi, 100)';
x = cos(t);   y = sin(t);
z = ones(size(x));
r = (1:-0.1:0);
h = 2*(0:0.1:1);
X = x * r;
Y = y * r;
Z = z * h;
%% 3차원 그래프
figure(1);
surf( X, Y, Z, 'LineStyle', 'none');
grid on;
xlabel('x'); ylabel('y'); zlabel('z');
```



Color gradation control



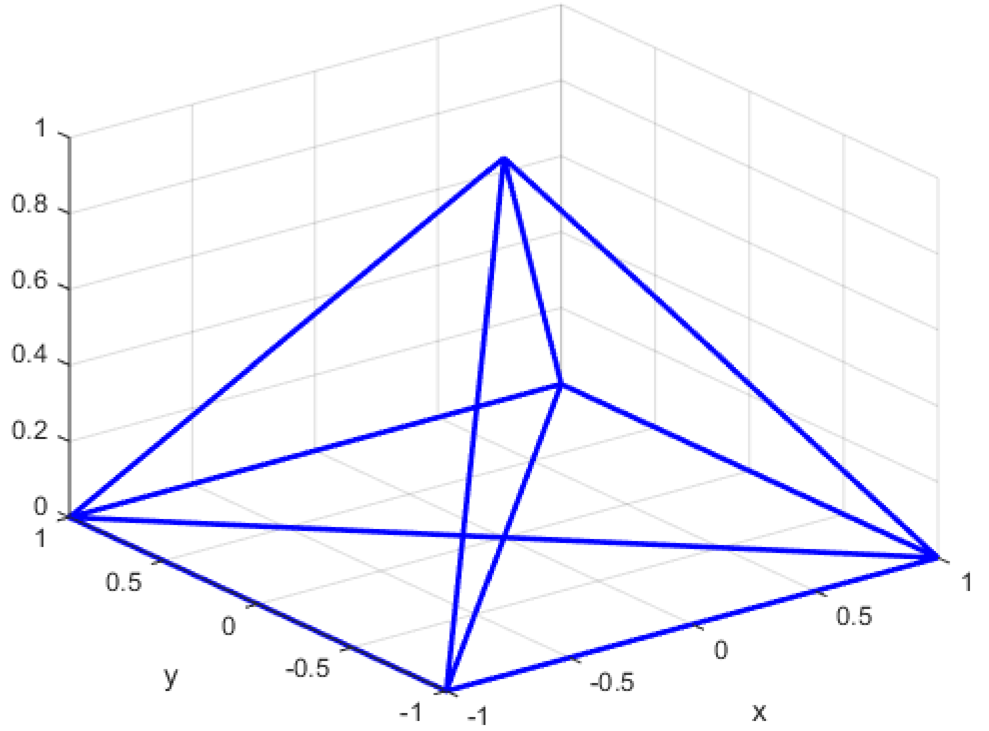
인천대학교
INCHEON NATIONAL UNIVERSITY

Eulerian Trail: Pyramid

- Find an Eulerian trail traversing each edge on the graph below exactly once.
  - Represent the trail with a set of vectors, x, y, and z.
  - Use plot3 to draw the graph.

m10_wired_pyramid.m

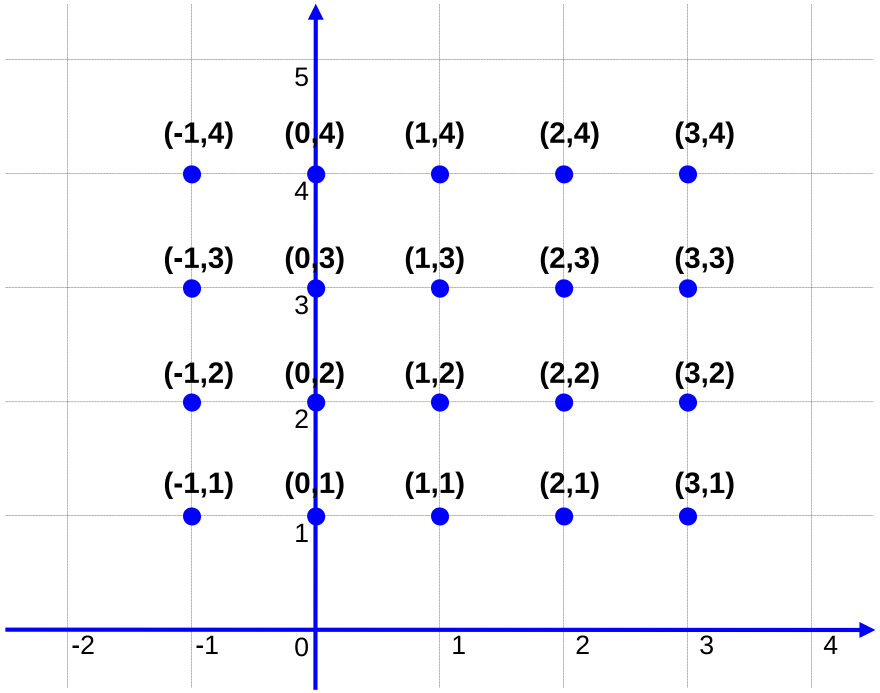
```
% Pyramid
x = [ -1 -1  1  1 -1  1  0  1 -1  0 -1 ];
y = [  1 -1 -1  1  1 -1  0  1 -1  0  1 ];
z = [  0  0  0  0  0  0  1  0  0  1  0 ];
figure(1);
plot3(x, y, z, 'b-', 'LineWidth', 2);
grid on; xlabel('x'); ylabel('y');
```



인천대학교
INCHEON NATIONAL UNIVERSITY

Mesh and Surface Plots

- 3D plot for $z = f(x, y)$
  - Step 1: Create a grid in the x-y plane.
  - Step 2: Calculate the values of z .
  - Step 3: Plot the 3D graph.



X=

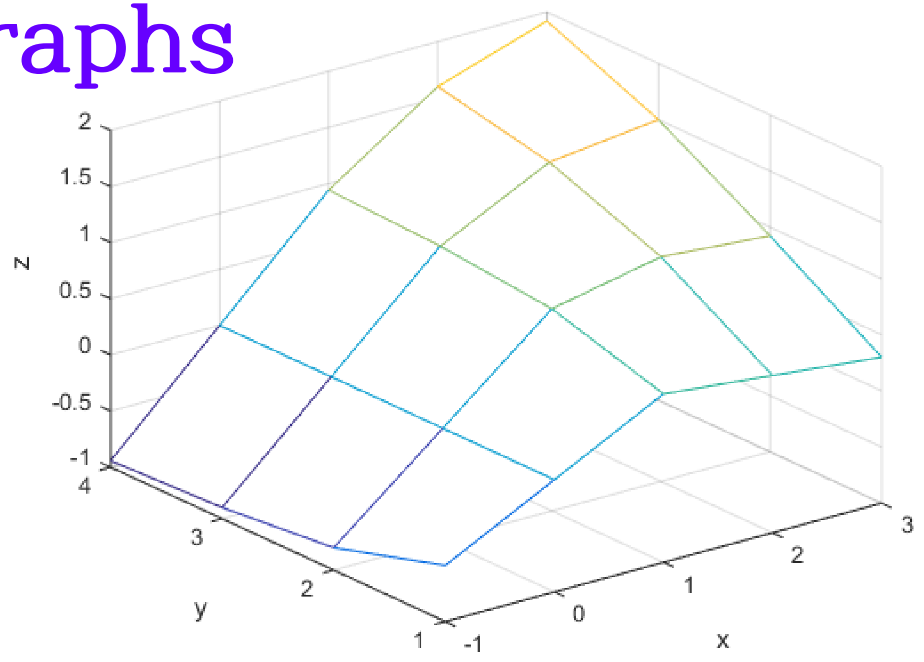
Y=

Mesh Grid

- Step 1: Create a grid in *x*-*y* plane

```
>> x = -1:3;
>> y =  1:4;
>> [ X Y ] = meshgrid(x,y)
X =
    -1      0      1      2      3
    -1      0      1      2      3
    -1      0      1      2      3
    -1      0      1      2      3
Y =
     1      1      1      1      1
     2      2      2      2      2
     3      3      3      3      3
     4      4      4      4      4
```

Plotting Mesh and Surface Graphs



- Step 2: Calculate the values of z

$$z = \frac{xy^2}{x^2 + y^2}$$

```
Z = X .* Y.^2 ./ (X.^2 + Y.^2);
```

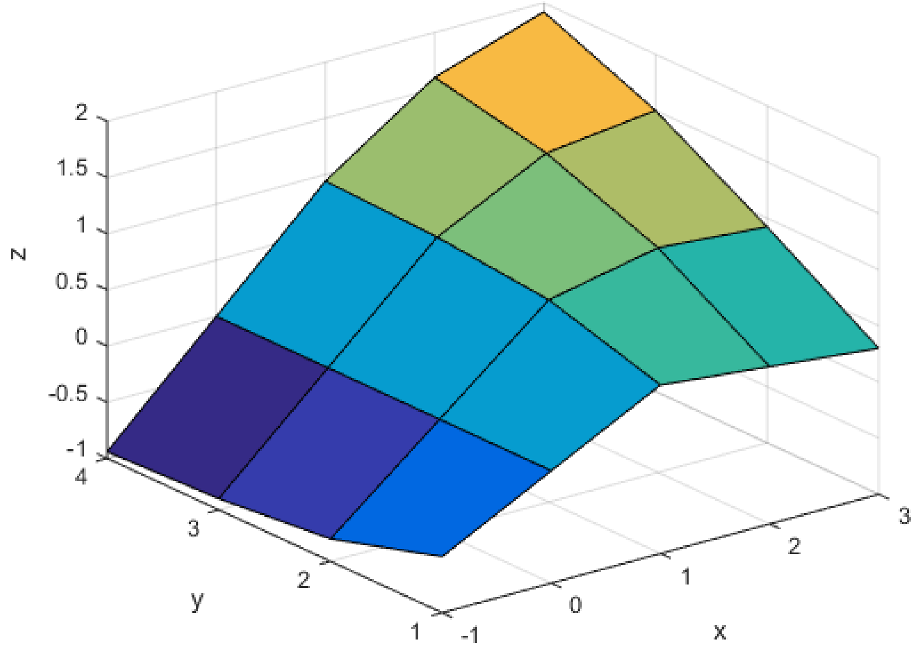
- Step 3: Plot the graph

```
mesh(X, Y, Z)
```

```
surf(X, Y, Z)
```

m10_mesh_and_surf.m

```
figure(1);
mesh(X,Y,Z);
xlabel('x'); ylabel('y'); zlabel('z');
```

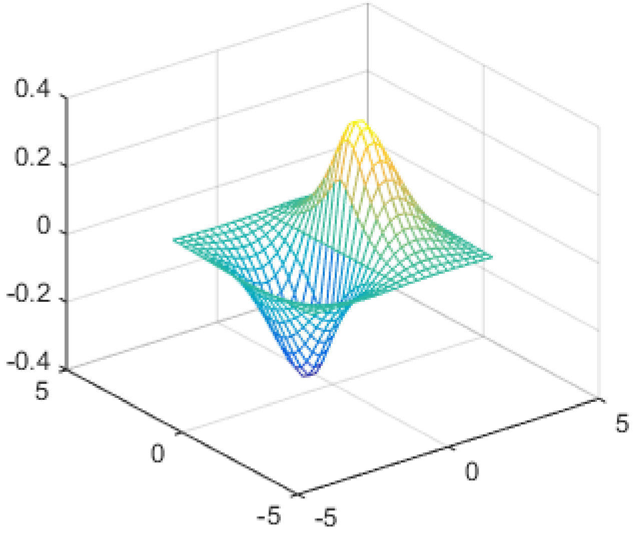


Various 3-D Graphs

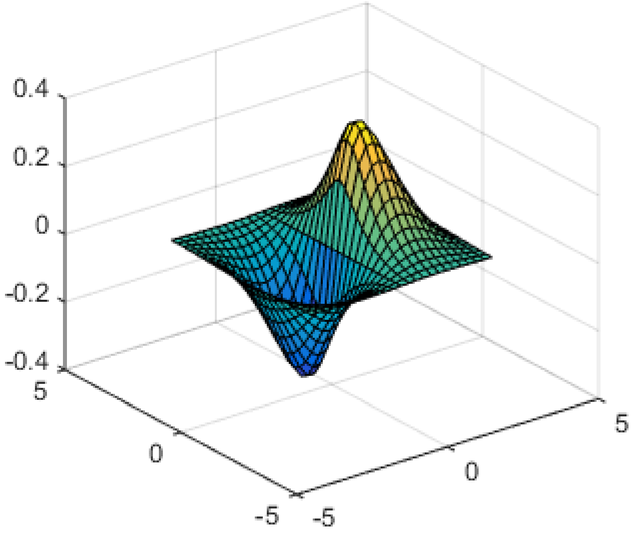
$$z = 1.8^{-1.5\sqrt{x^2+y^2}} \sin(x)\cos(y/2)$$

```
x = -3:0.25:3;
[ X Y ] = meshgrid(x,x);
Z = 1.8.^(-1.5*sqrt(X.^2 + Y.^2)) ...
    .* sin(X) .* cos(Y/2);
```

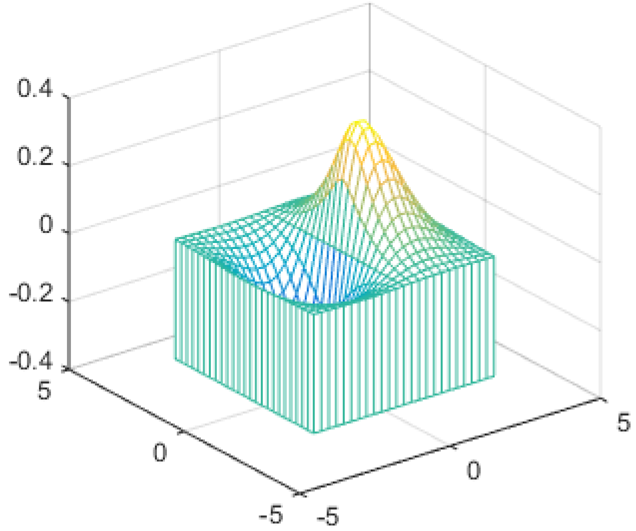
```
mesh( X, Y, Z );     surf( X, Y, Z );
meshz( X, Y, Z );    meshc( X, Y, Z );
surfc( X, Y, Z );    surfl( X, Y, Z );
waterfall( X, Y, Z );
trisurf( delaunay(X,Y), X, Y, Z);
contour( X, Y, Z, 15);
contour3( X, Y, Z, 15);
```



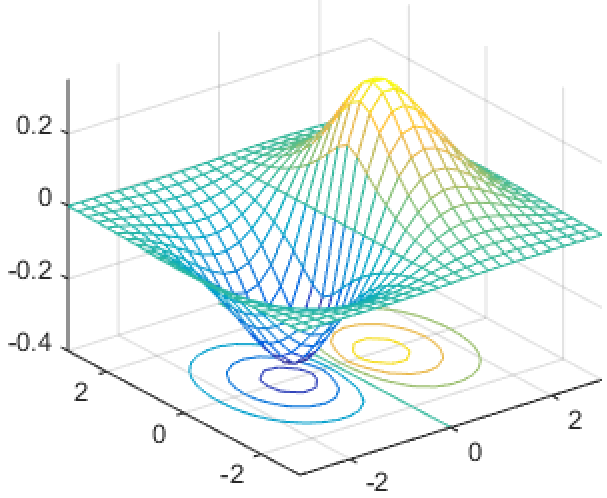
`mesh(X,Y,Z);`



`surf(X,Y,Z);`



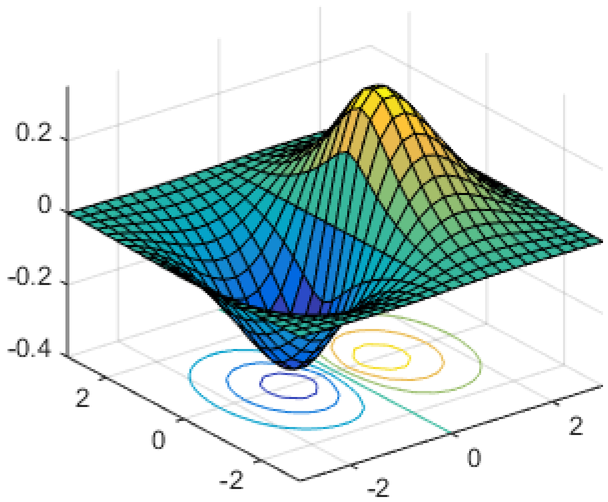
`meshz(X,Y,Z);`



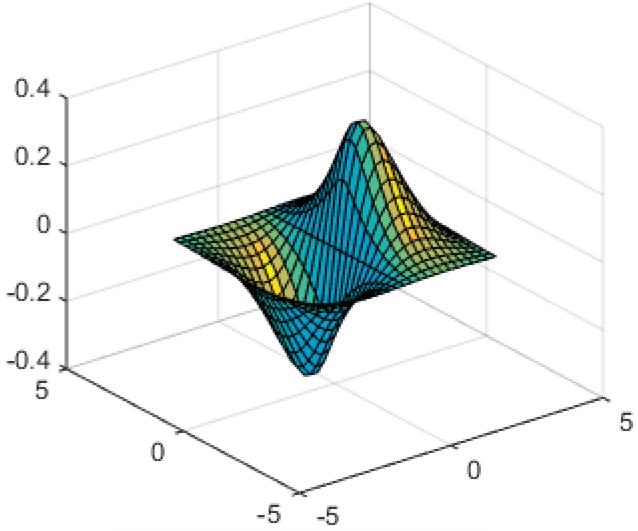
`meshc(X,Y,Z);`

인천대학교 INCHEON NATIONAL UNIVERSITY

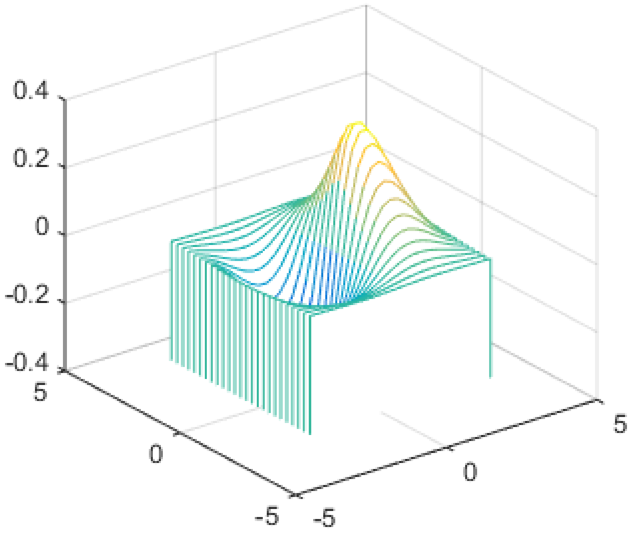
m10_various_3d_graphs.m



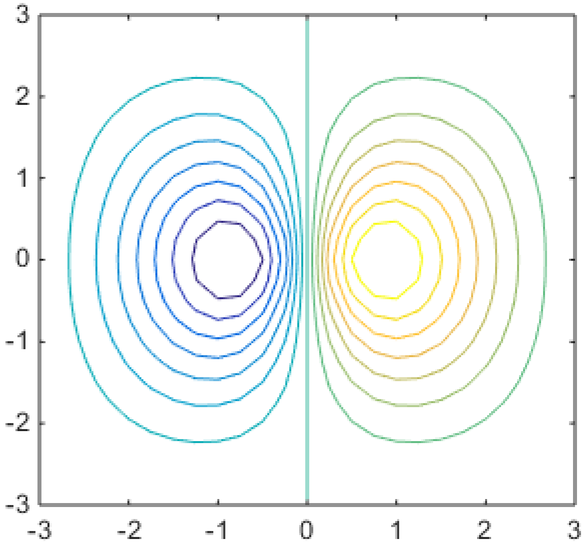
`surfc(X,Y,Z);`



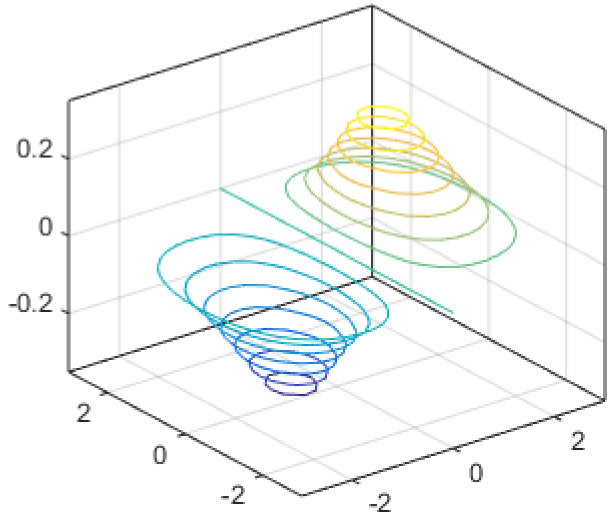
`surfl(X,Y,Z);`



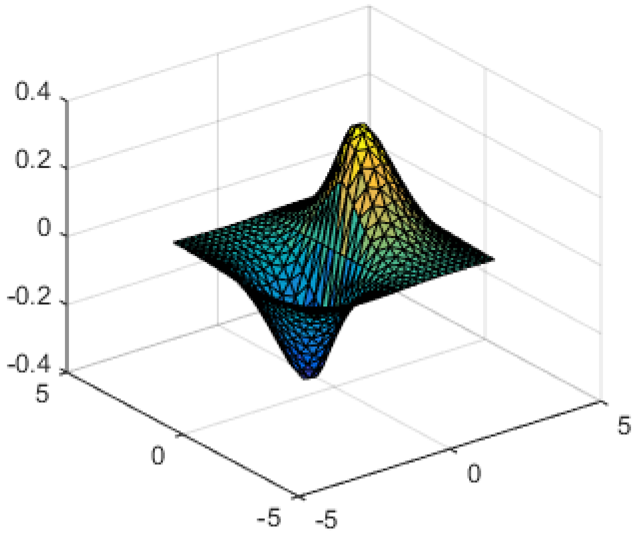
`waterfall(X,Y,Z);`



`contour(X,Y,Z,15);`

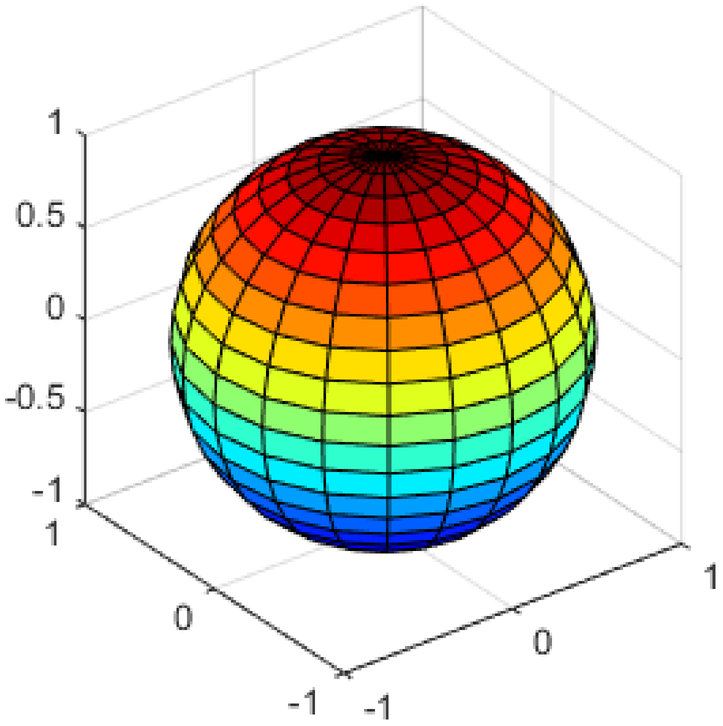


`contour3(X,Y,Z,15);`

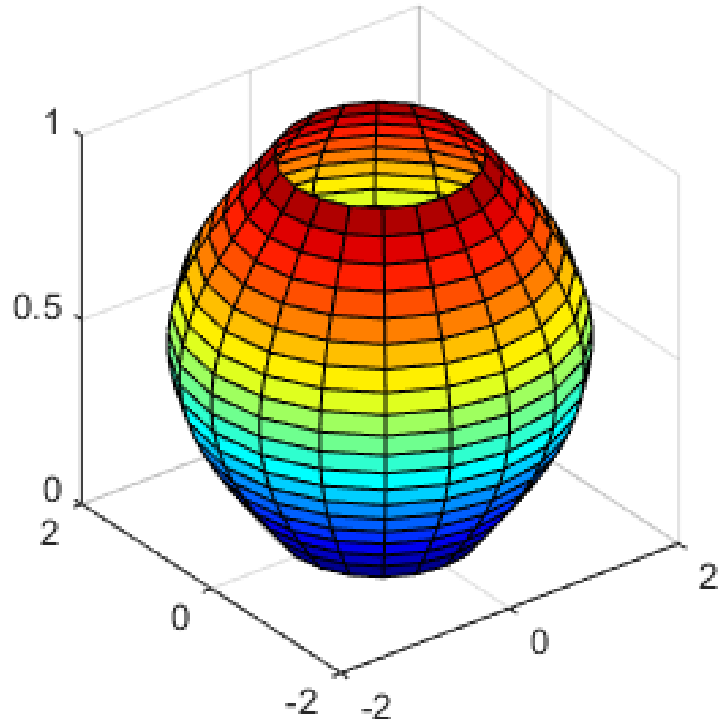


`trisurf(delaunay(X,Y),X,Y,Z);`

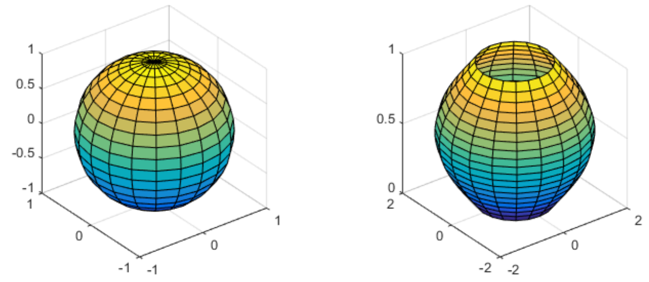
Sphere and Cylinder



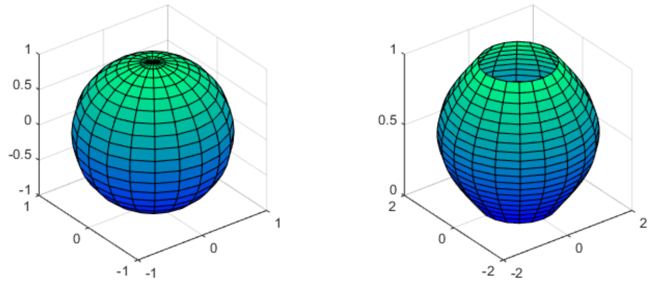

```
[ X Y Z ] = sphere(20);
figure(1), surf(X,Y,Z),
axis square;
colormap('jet');
```



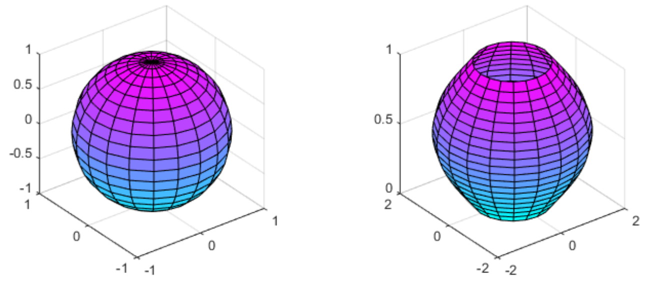

```
t = linspace(0,pi,20);
r = 1 + sin(t);
[X Y Z] = cylinder(r);
figure(2), surf(X,Y,Z),
axis square;
```



`colormap('parula');`



`colormap('winter');`



`colormap('cool');`

인천대학교
INCHEON NATIONAL UNIVERSITY

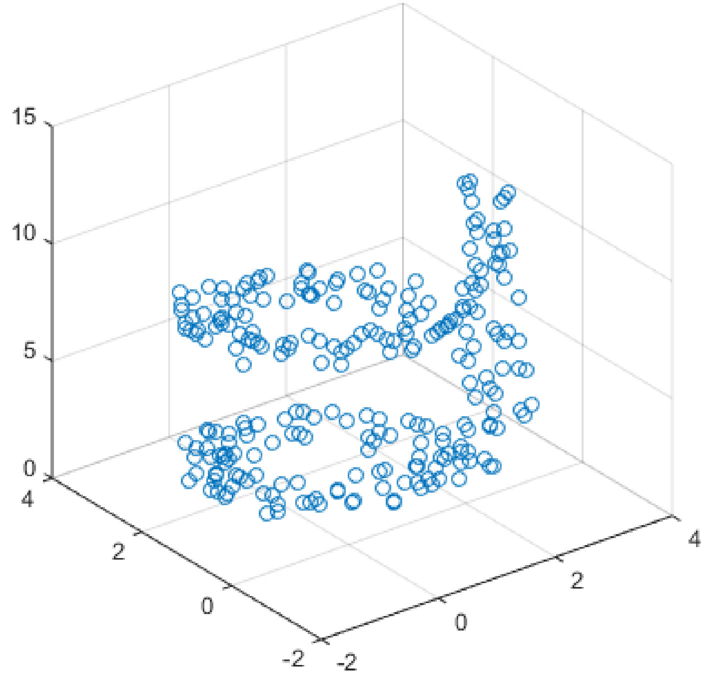
m10_sphere_cylinder.m

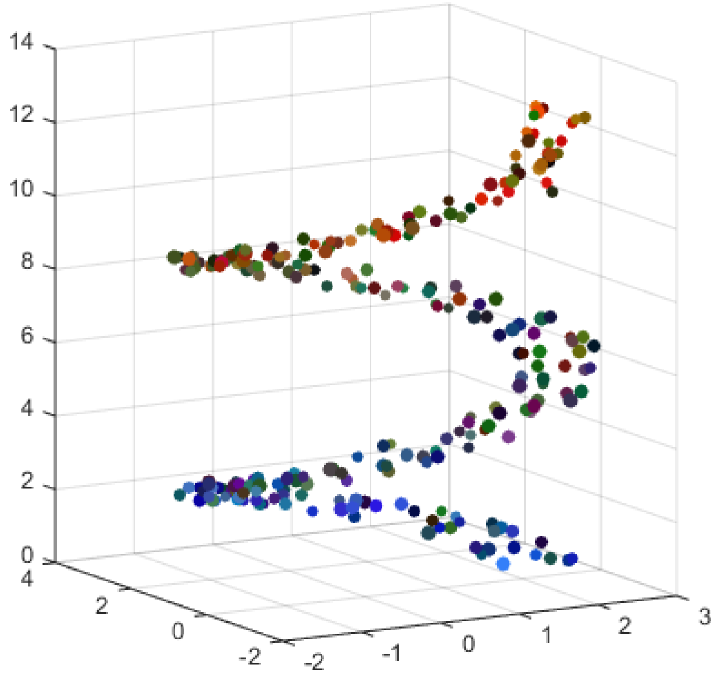
3D Scatter Plot

- **scatter(x, y, z, s, c)**

m10_scatter3.m

```
N = 250;
z = linspace(0,4*pi,N)';
x = 2*cos(z) + rand(N,1);
y = 2*sin(z) + rand(N,1);
s = 20 + 20*rand(N,1); % scale
c = rand(N,3);          % random color
g = [linspace(0.2,1,N); 0.5*ones(1,N);
linspace(1, 0, N) ]'; % color weight

figure('Position', ...
 [200 200 800 300], 'color', 'w');
subplot(1,2,1), scatter3( x, y, z );
subplot(1,2,2);
scatter3( x, y, z, s, c.*g, 'fill');
view(-30,10);
```





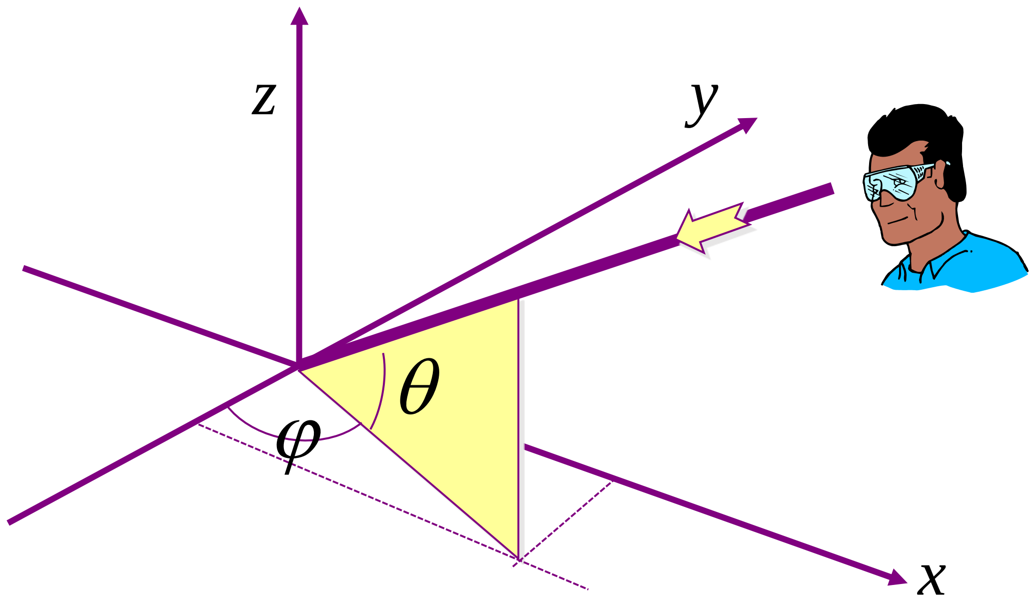
인천대학교
INCHEON NATIONAL UNIVERSITY

View Command

- The view command controls the direction from which the plot is viewed.

```
view(az, el) or view([az, el])
```

φ θ




```
[az el] = view;
```

```
view(3);
```

az = -37.5, el = 30

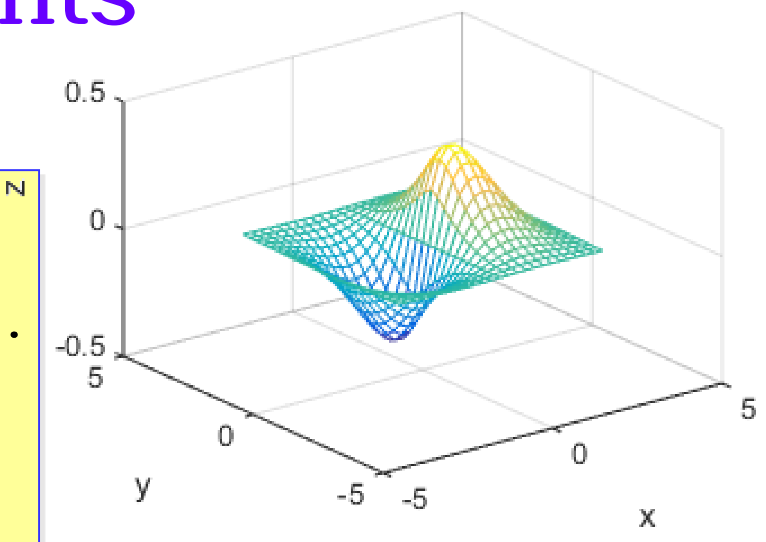
인천대학교
INCHEON NATIONAL UNIVERSITY

Different View Points

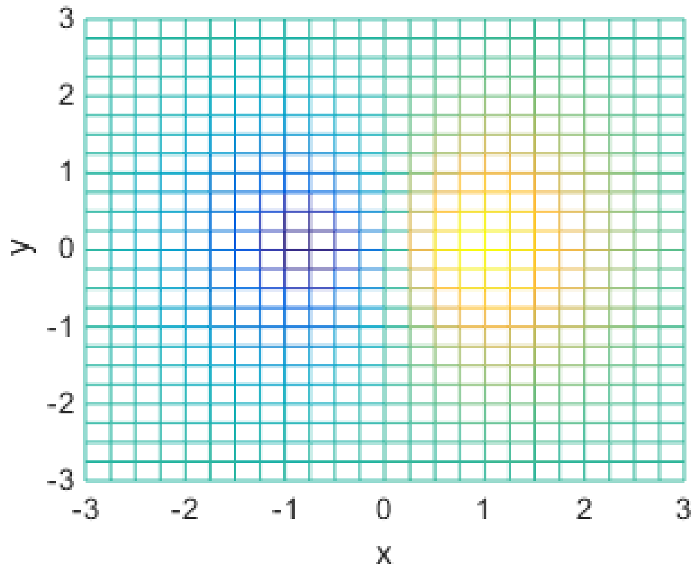
m10_view.m

```
[X,Y]=meshgrid([-3:0.25:3],...
               [-3:0.25:3]);
Z=1.8.^(-1.5*sqrt(X.^2+Y.^2))...
  .*sin(X) .*cos(Y/2);

subplot(2,2,1), mesh(X,Y,Z);
view(3);

subplot(2,2,2), mesh(X,Y,Z);
view(2); % view( 0, 90 );

subplot(2,2,3), mesh(X,Y,Z);
view(90, 0);

subplot(2,2,4), mesh(X,Y,Z);
view( 0, 0);
```

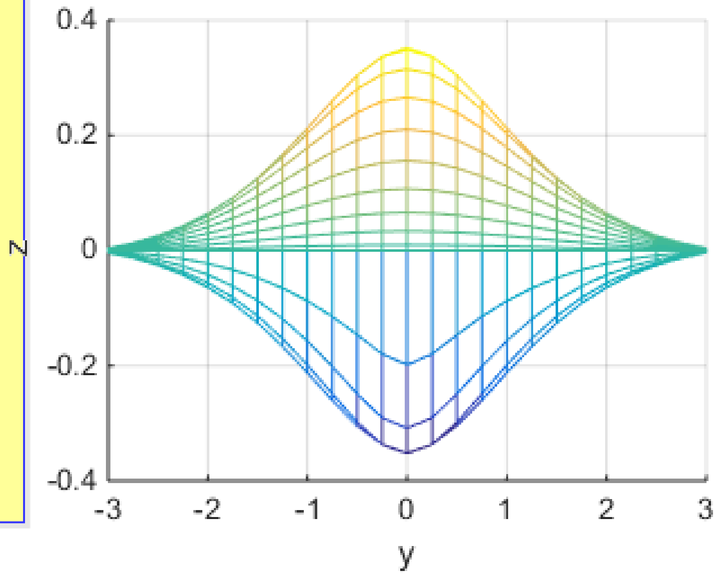
default or view(3)



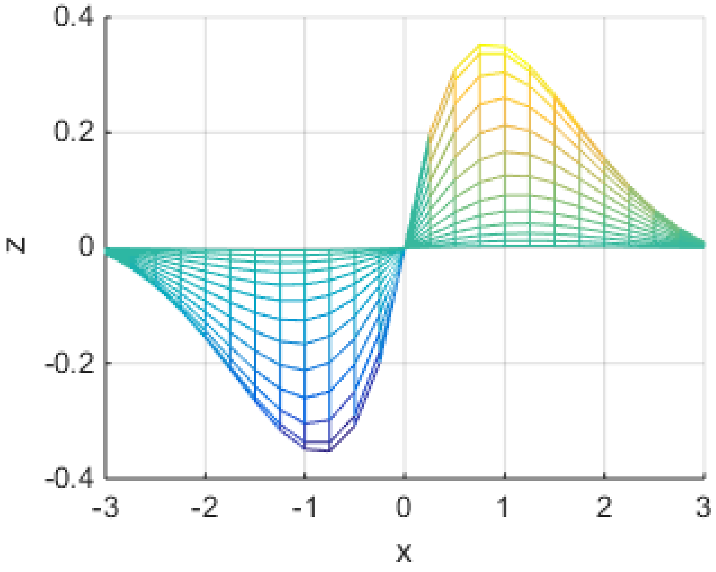
view(0,90) or view(2)



view(90,0)



view(0,0)

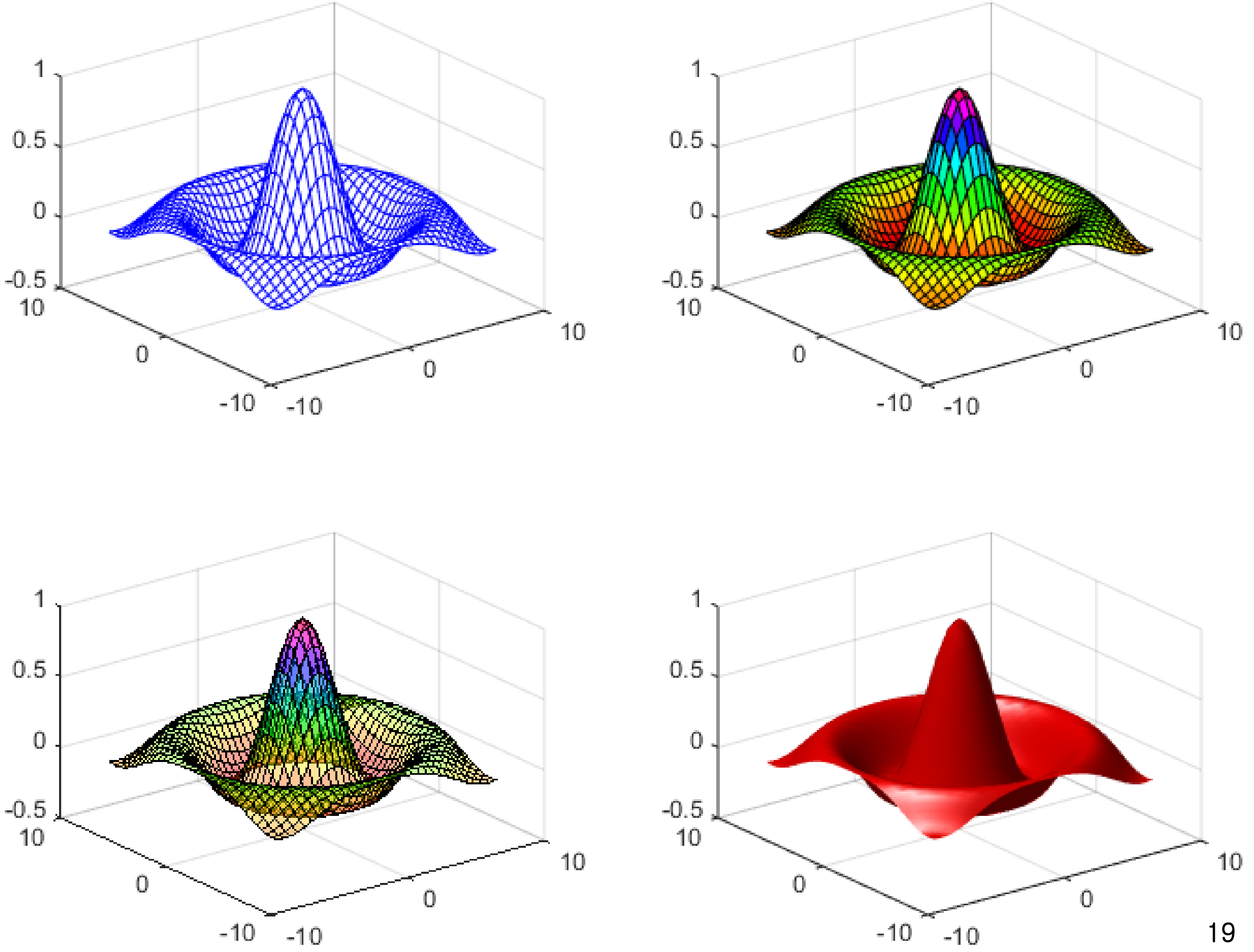


인천대학교
INCHEON NATIONAL UNIVERSITY

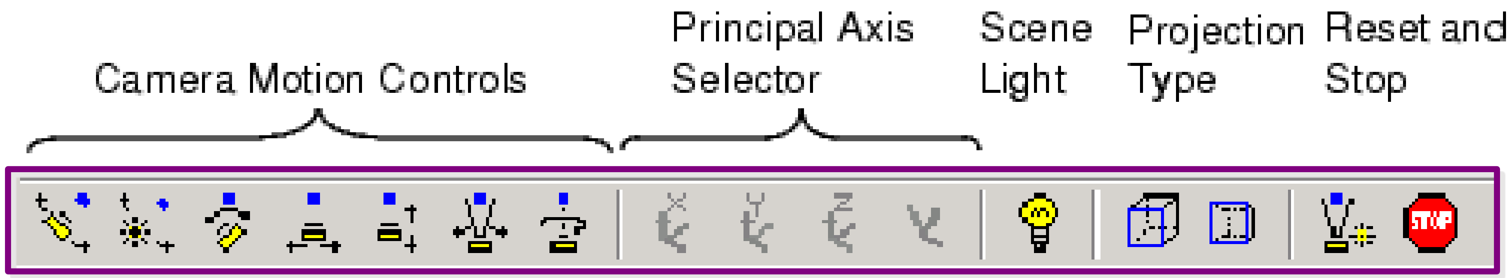
Colors

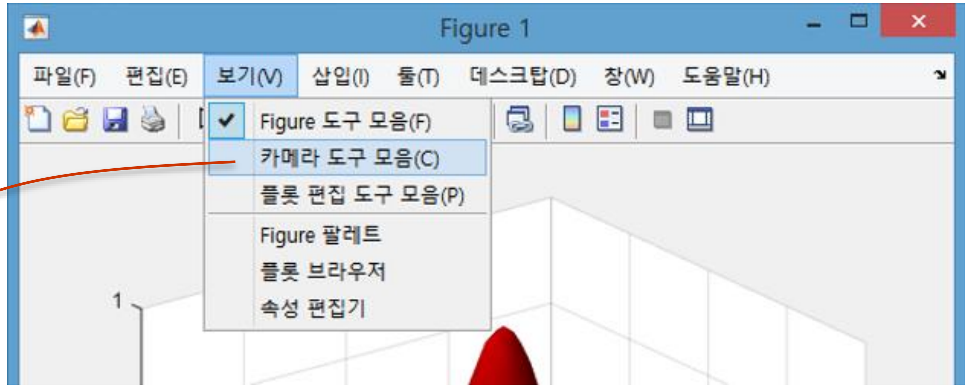
m10_surface_colors.m

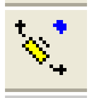
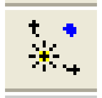
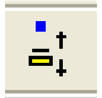
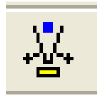
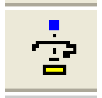
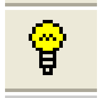
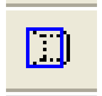
```
[X,Y]=meshgrid(-8:.5:8);
R = sqrt(X.^2 + Y.^2) + eps;
Z = sin(R) ./ R;

subplot(2,2,1);
mesh(X,Y,Z, 'EdgeColor', 'b');

subplot(2,2,2);
surf(X,Y,Z); colormap('hsv');

subplot(2,2,3);
surf(X,Y,Z); alpha(0.4);

subplot(2,2,4),
surf(X,Y,Z,'FaceColor', ...
  'red', 'EdgeColor','none');
camlight left; lighting phong;
```

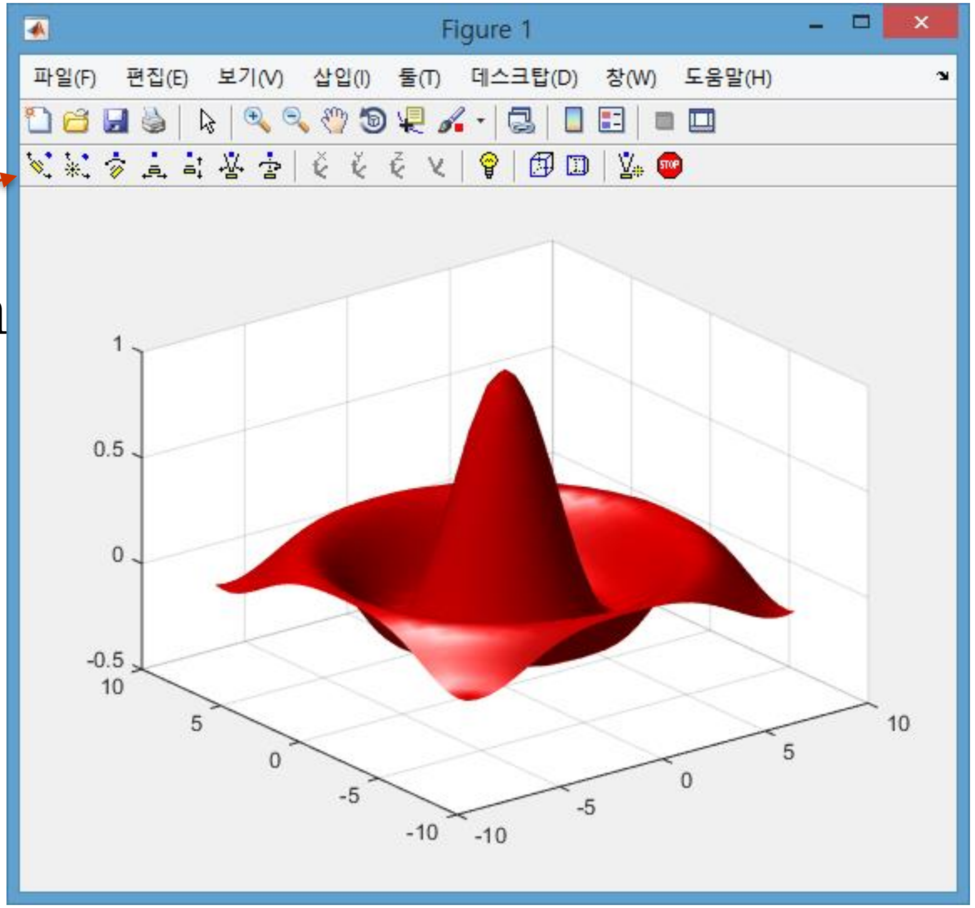


Camera Toolbar





-  Orbit Camera
-  Orbit Scene Light
-  Move Camera Forward and Backwa
-  Zoom In and Out
-  Camera Roll
-  Toggle Lighting Effect
-  3D Projection



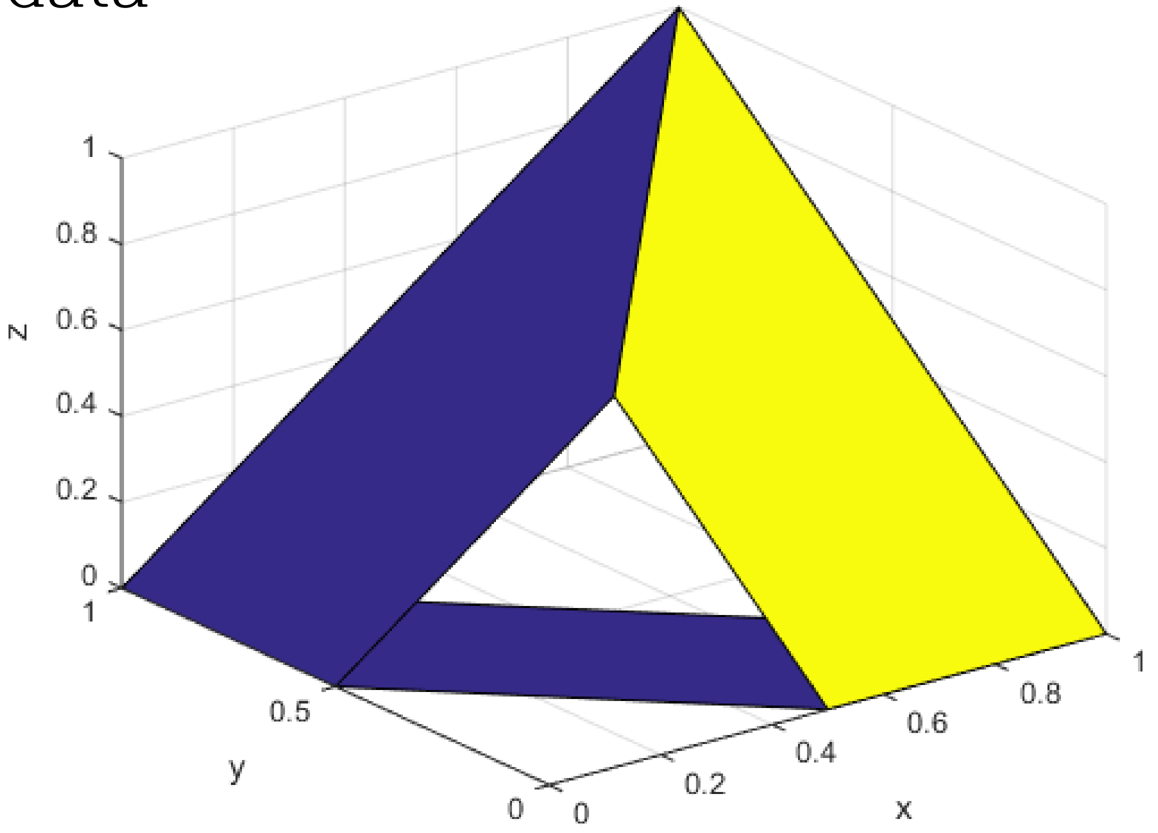
Patch

- Low level surface plot function
  - Suitable for machine generated data

m10_patch_triangle.m

```
X = [0.0 0.5 0.5;
     0.0 1.0 1.0;
     0.5 1.0 0.0;
     1.0 0.5 0.0];
Y = [1.0 0.0 0.0;
     0.5 0.0 0.0;
     0.5 1.0 1.0;
     1.0 0.5 0.5];
Z = [0.0 0.0 0.0;
     0.0 0.0 0.0;
     0.5 1.0 0.0;
     1.0 0.5 0.0];
```

```
C = [1 15 1;
     1 15 1;
     1 15 1;
     1 15 1];

patch( X, Y, Z, C);
view(3); grid on;
xlabel('x');
ylabel('y');
zlabel('z');
```

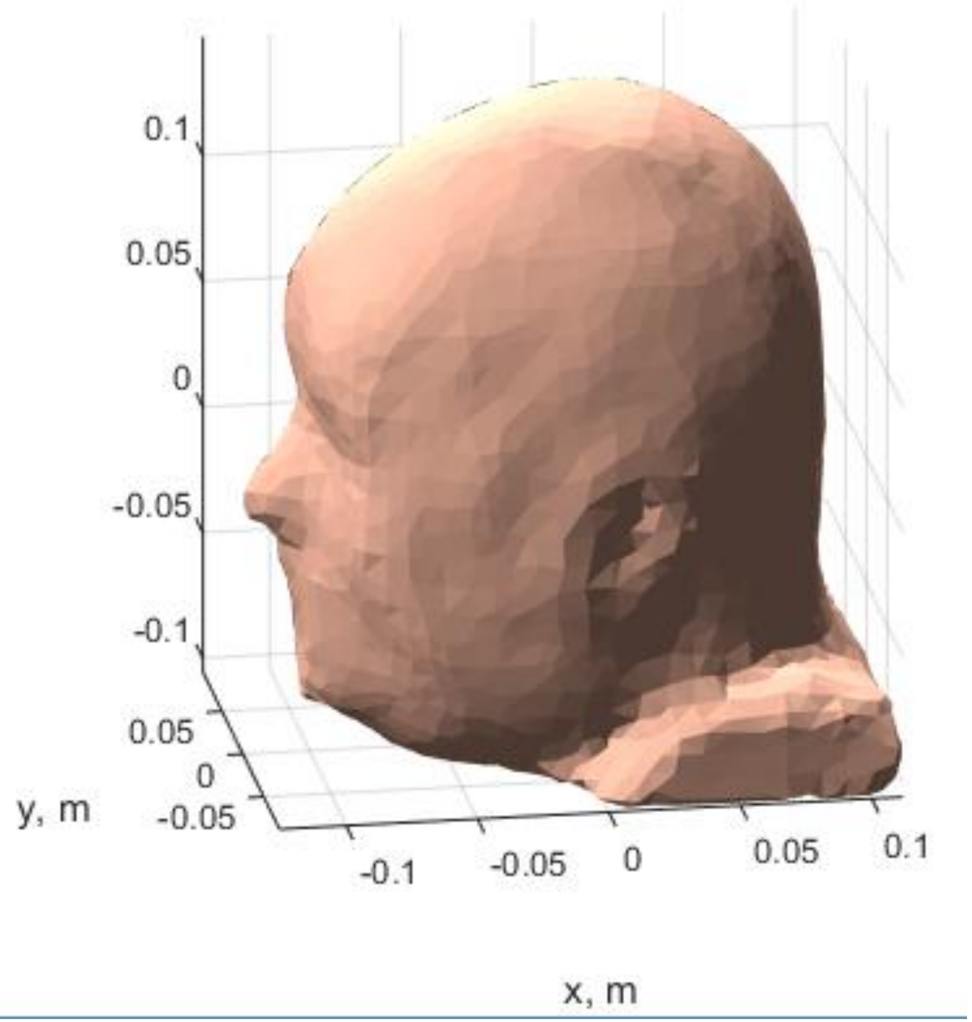


Human Head using Patch

- Patch

m10_patch_man_head.m

```
load  man_head X Y Z;

figure('Color', 'w');
patch(X, Y, Z, [1 0.75 0.65], ...
 'EdgeColor', 'none', 'FaceAlpha', 1.0);
axis 'equal';    axis 'tight';
xlabel('x, m'); ylabel('y, m');
view(-9, 19); grid on;
light('Position',[1 3 2]);
light('Position',[-3 -1 3]);
material dull;
```

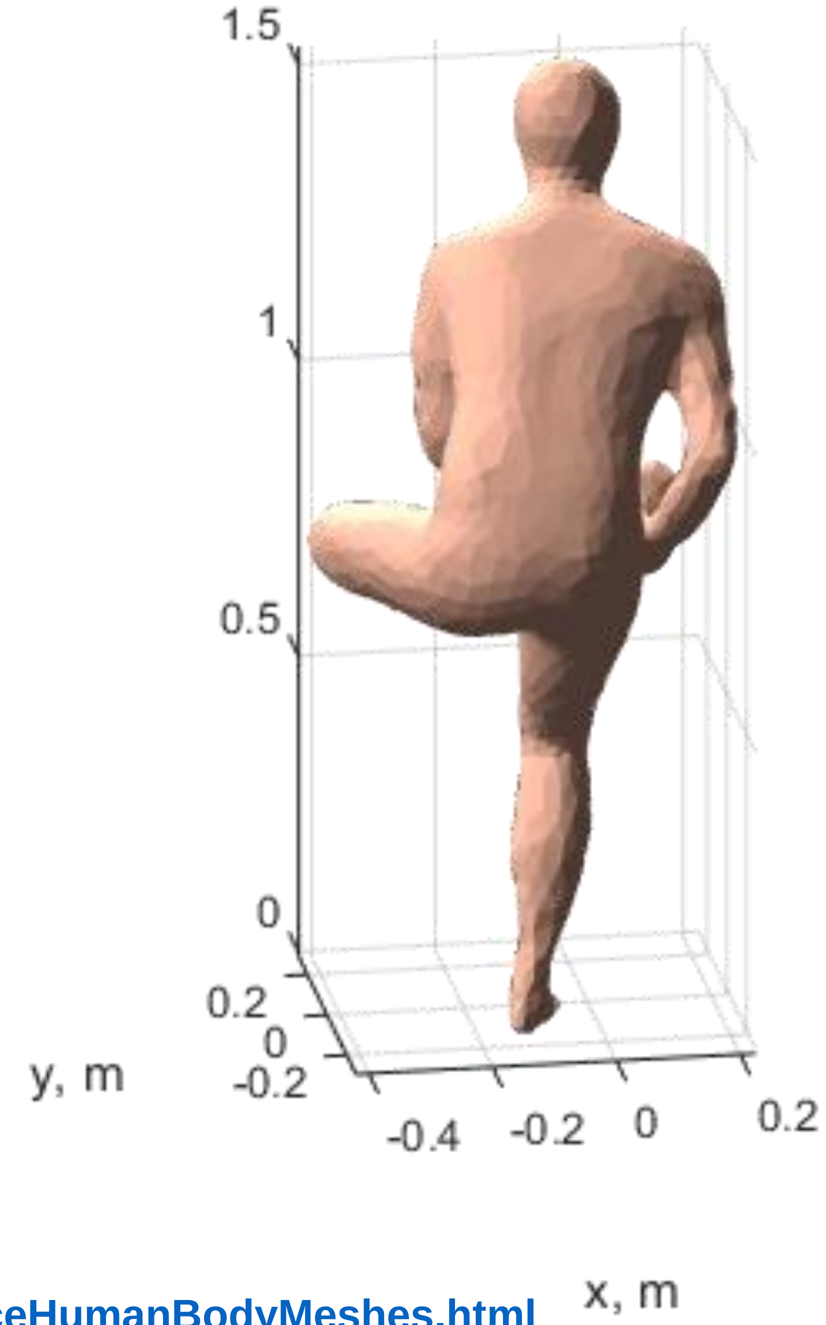


Human Body using Patch

- Patch

m10_patch_leg_across.m

```
load  man_leg_across X Y Z;
figure('Color', 'w');
patch(X, Y, Z, [1 0.75 0.65], ...
         'EdgeColor', 'none', ...
         'FaceAlpha', 1.0);
axis 'equal';    axis 'tight';
xlabel('x, m'); ylabel('y, m');
view(-9, 19); grid on;
light('Position',[1 3 2]);
light('Position',[-3 -1 3]);
material dull;
```



https://www.nevaelectromagnetics.com/SurfaceHumanBodyMeshes.html

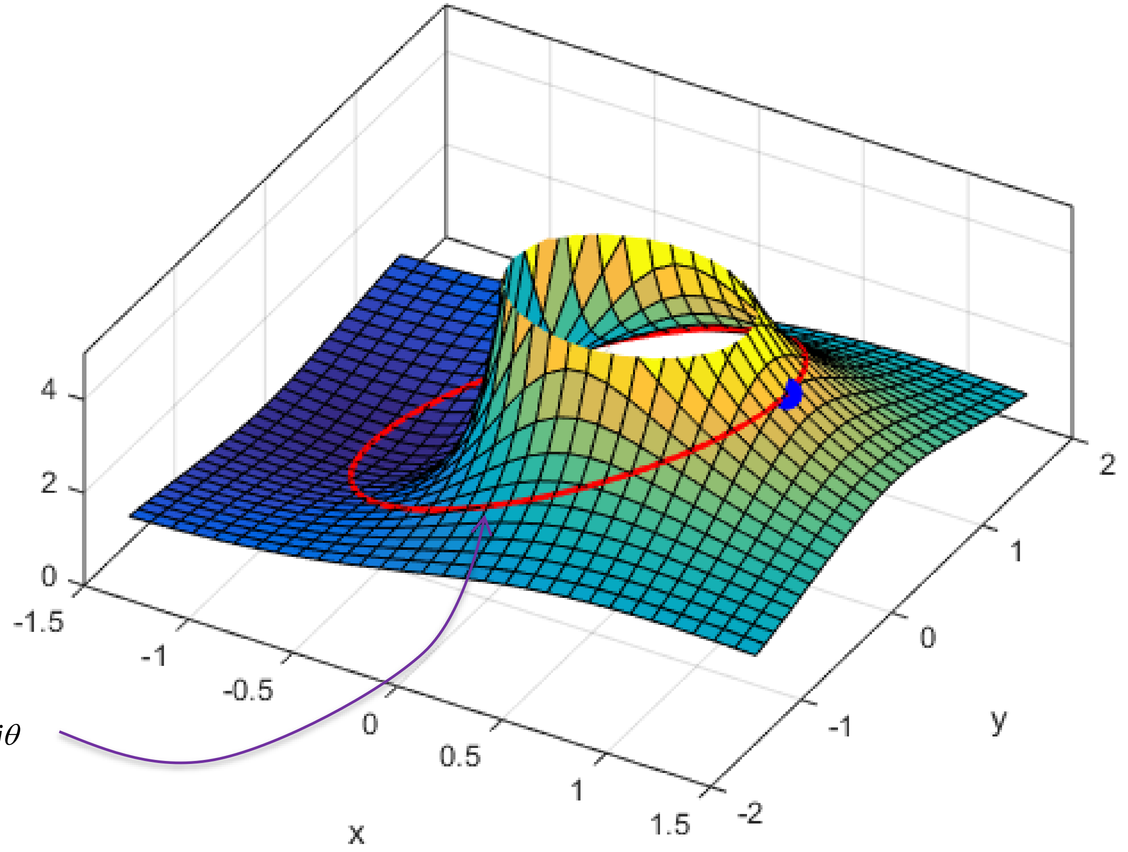
Application: Frequency Response of FIR Filter

- Plot the magnitude surface of the following FIR filter over the z-plane as well as over the unit circle.

$$H(z) = 1 + 2z^{-1} + z^{-2}$$

where $z = x + iy$

$|H(z)|$



$|H(z)|_{z=e^{j\theta}}$

Script for z-Domain Response

m10_freq_response_fir_filter.m

```
[X Y] = meshgrid([-1.5:0.1:1.5],[-1.5:0.1:1.5]);
Z = X + j*Y;
Hz = 1 + 2*Z.^(-1) + Z.^(-2);
figure(1);
surf(X,Y,abs(Hz), min(abs(Hz),5)); zlim([0 5]);
view(30,60);
hold on;
t = 0:0.1:2*pi;
x = cos(t); y = sin(t); z = x + j*y;
hz = 1 + 2*z.^(-1) + z.^(-2);
plot3(x,y,abs(hz),'r-','LineWidth',2);
plot3(1,0,4,'yo', ...
    'MarkerSize',10,'MarkerFaceColor','blue', ...
    'MarkerEdgeColor','blue');
hold off;
xlabel('x'); ylabel('y');
box on;
```

Summary

- Do you recognize the following commands?

`plot3(x,y,z,'r-o')`	`box on`	`box off`	`hold on`
`sind(x)`	`sin(x)`	`[X Y]=meshgrid(x,y)`	`grid`
`view(30,65)`	`mesh(X,Y,Z)`	`surf(X,Y,Z,C)`	`alpha(0.4)`
`surf(X,Y,Z,'FaceColor','interp','EdgeColor','none')`			
`colorbar`	`lighting phong`	`axis square`	`camlight left`
`colormap('Cool')`	`daspect([1 1 1])`	`view(3)`	`view(2)`
`camproj perspective`	`waterfall(X,Y,Z)`	`H(find(H>5))=5`	