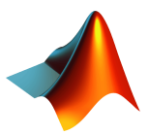



MATLAB 프로그래밍 및 실습

3강. 시각화 기초

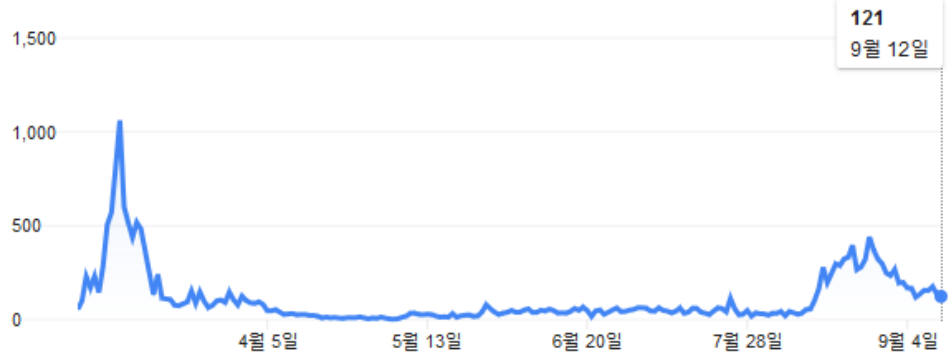


일일 변동 내역

신규 확진자 ▼

 대한민국 ▼

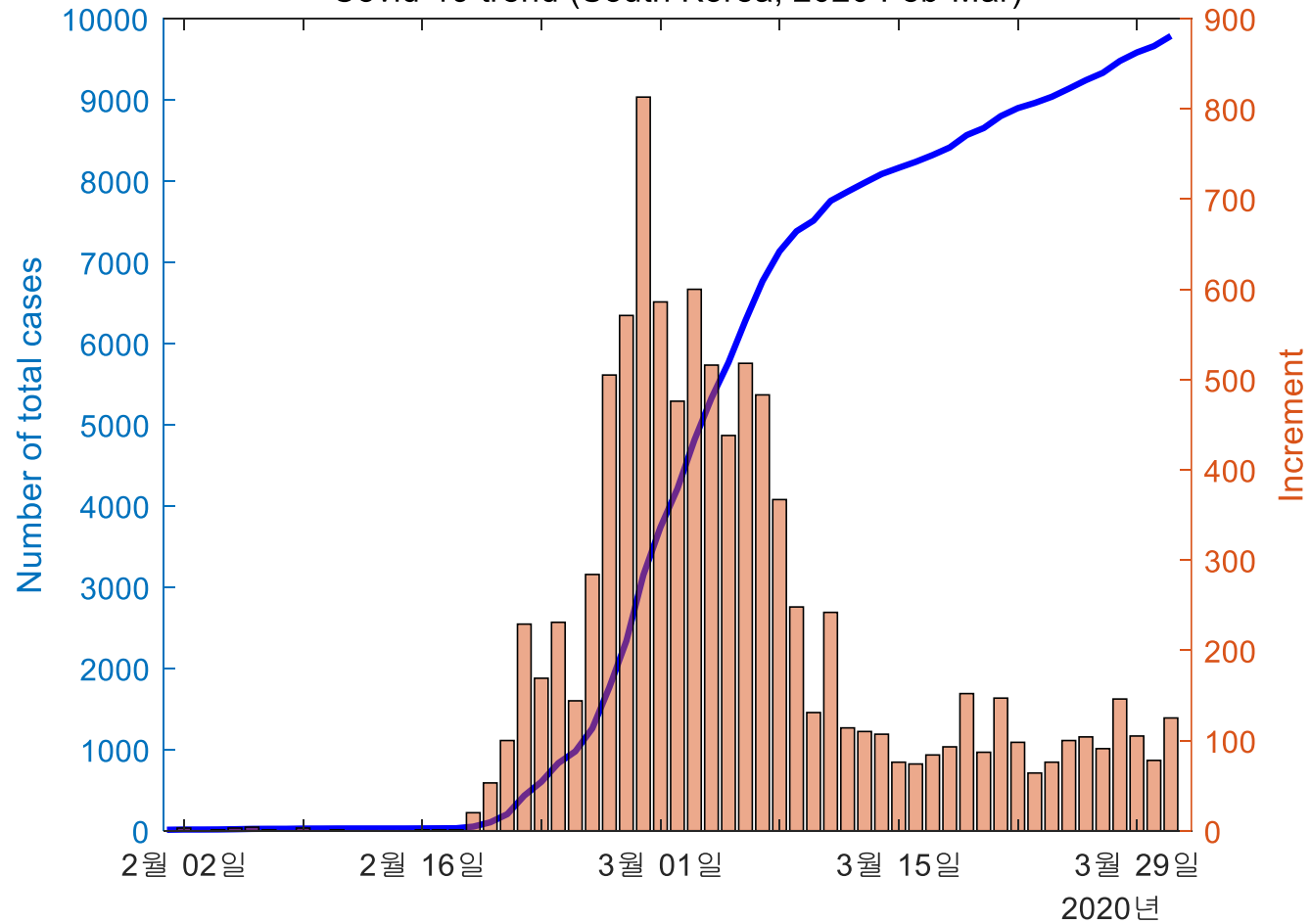
전체 ▼



각 날짜에는 전날 이후 보고된 신규 확진자가 표시됩니다. · 30분 전 업데이트됨 · 출처: [위키백과](#) · [데이터 정보](#)

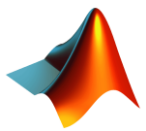
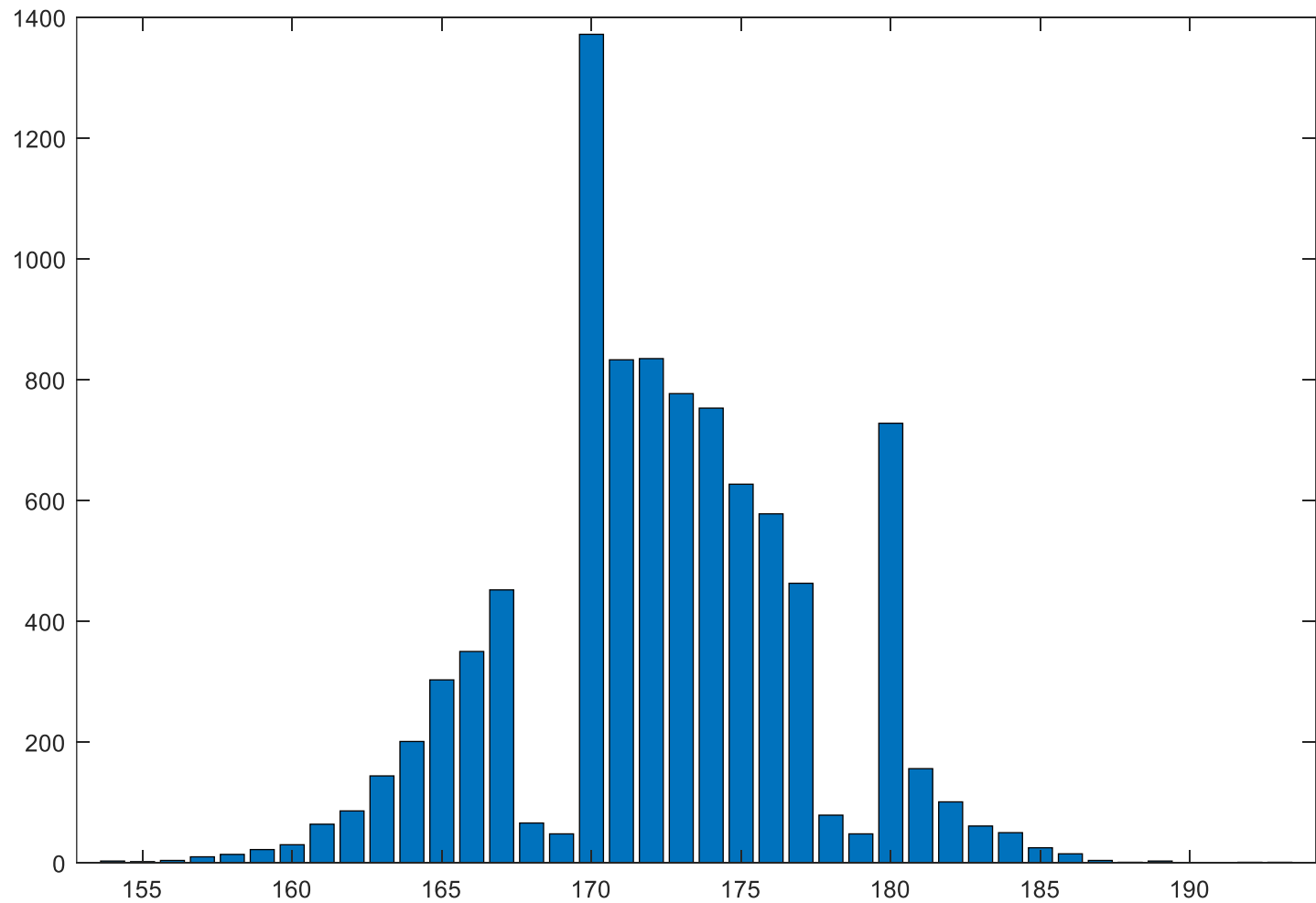
12	3736
15	4212
15	4812
16	5328
19	5766
23	6284
24	6767
24	7134
27	7387

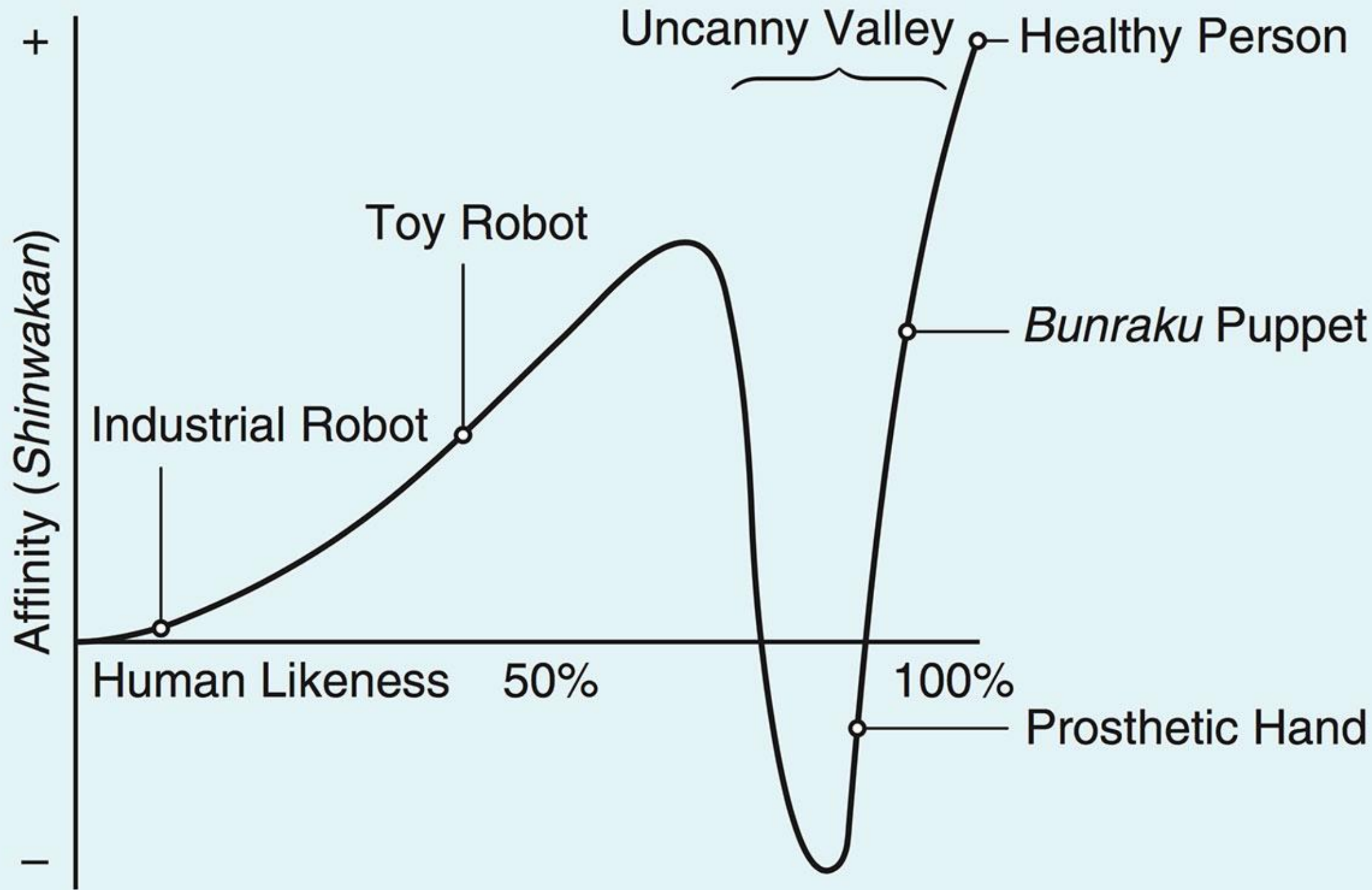
Covid-19 trend (South Korea, 2020 Feb-Mar)



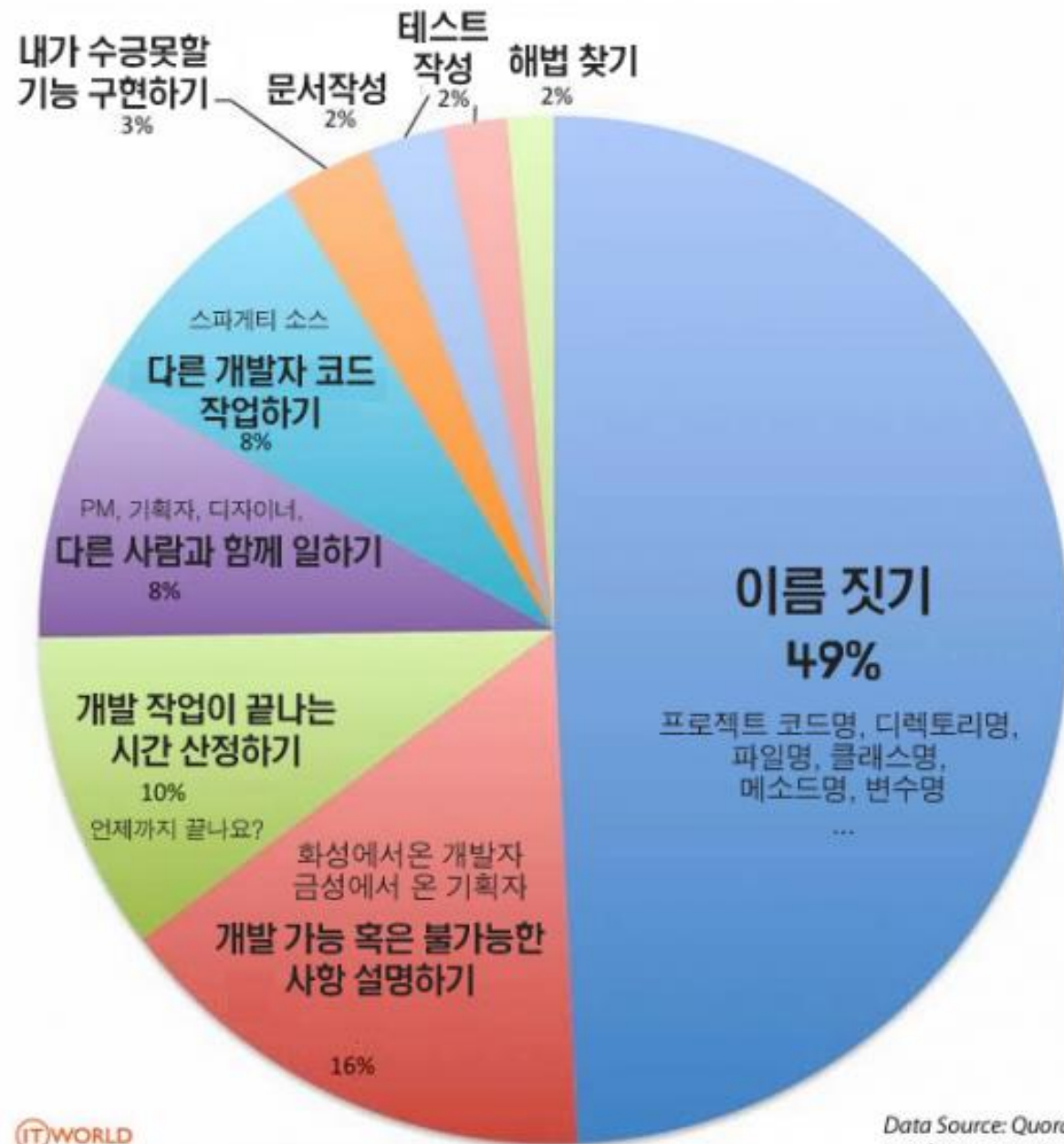
키	사람수
154	3
155	2
156	4
157	10
158	14
159	22
160	30
161	64
162	86
163	144
164	201
165	303
166	350
167	452
168	66
169	48
170	1372
171	833
172	835
173	777
174	753
175	627
176	578
177	463
178	79
179	48
180	728
181	156
182	101
183	61
184	50
185	25
186	15
187	4
188	1
189	3
190	2

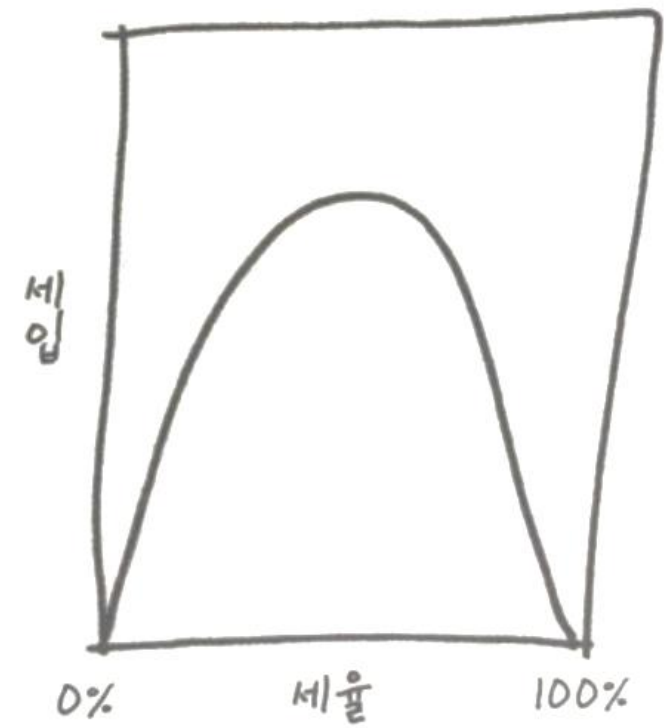
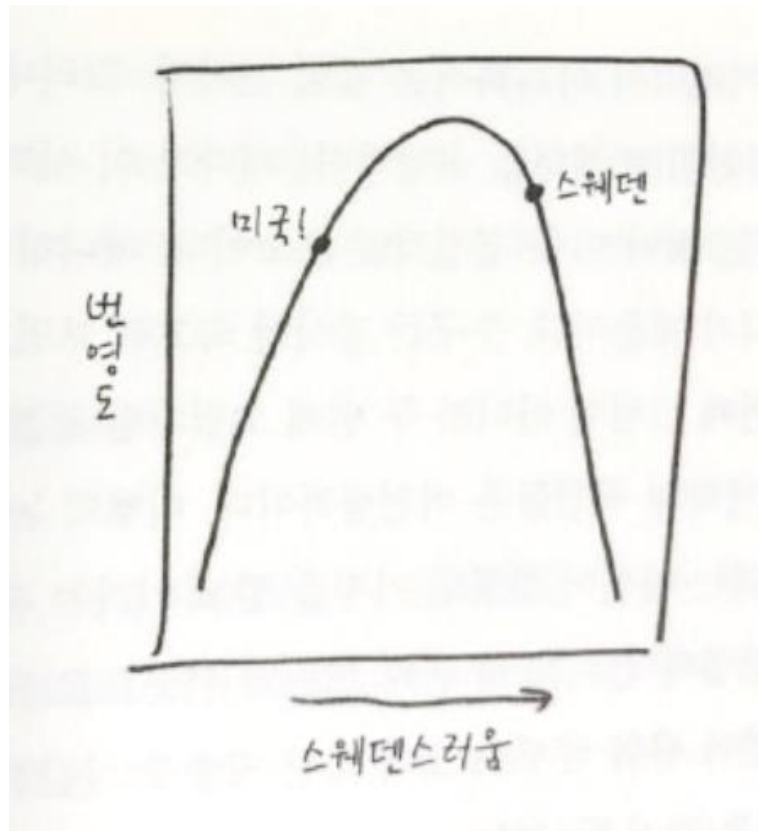
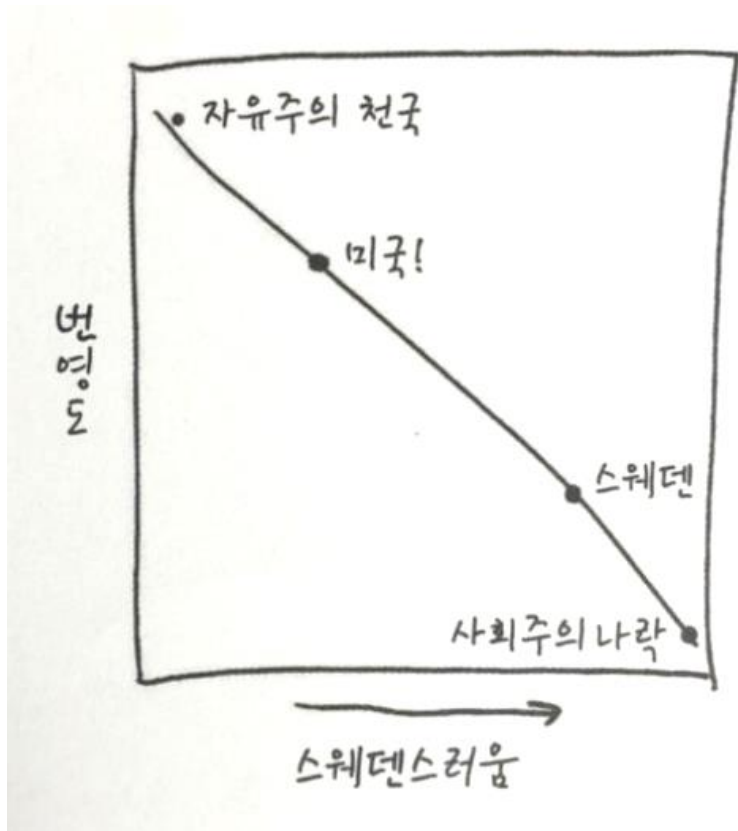
한국 남성 키 분포





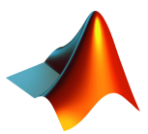
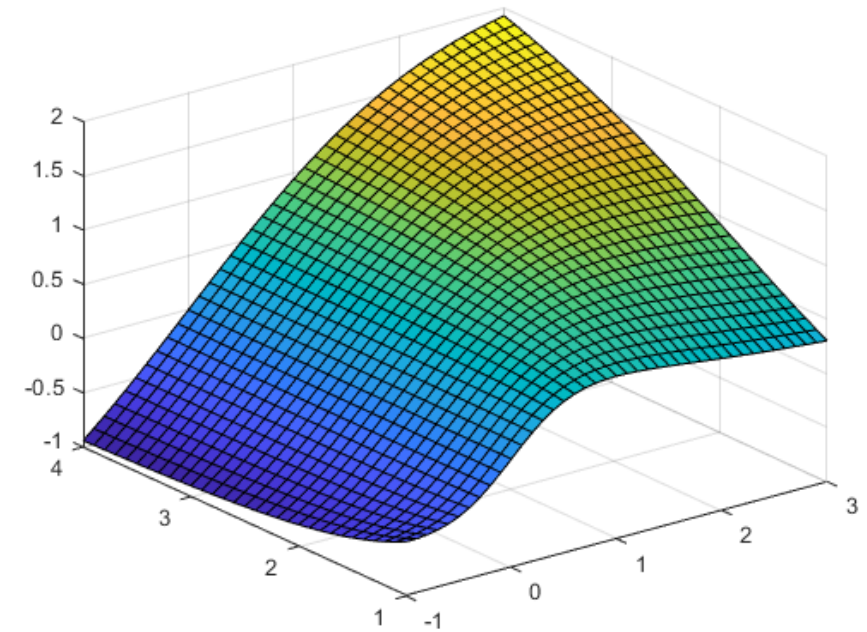
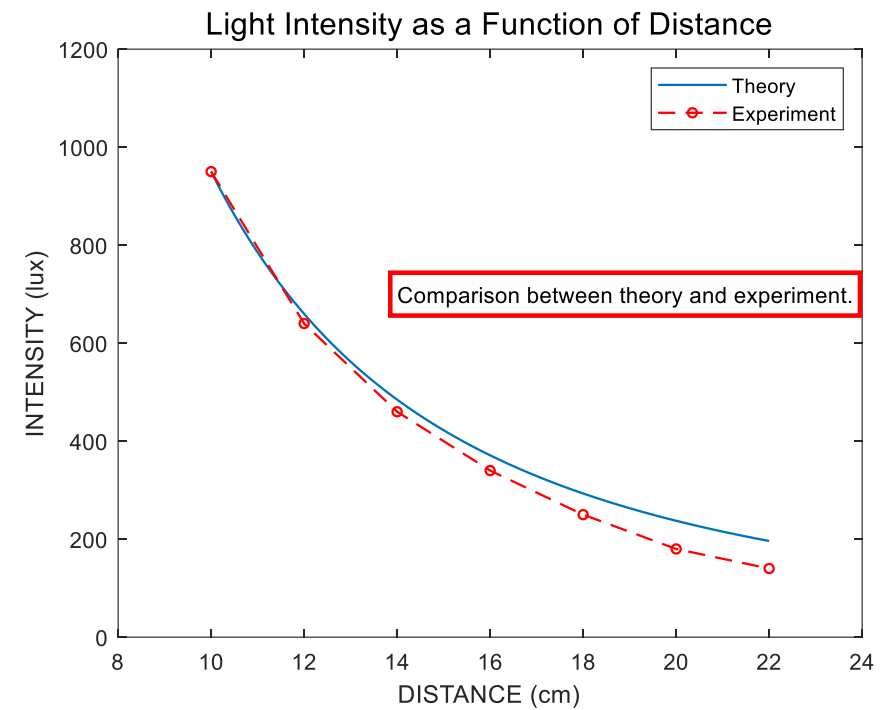
프로그래머가 가장 힘들어하는 일은?



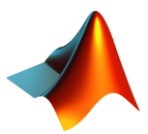


오늘 배울 내용

- 2차원 그래프
 - 선 스타일 지정
 - 여러 plot 겹쳐 그리기
 - subplot
- 3차원 그래프
 - meshgrid
 - surf, mesh
 - colormap



2차원 그래프

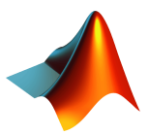
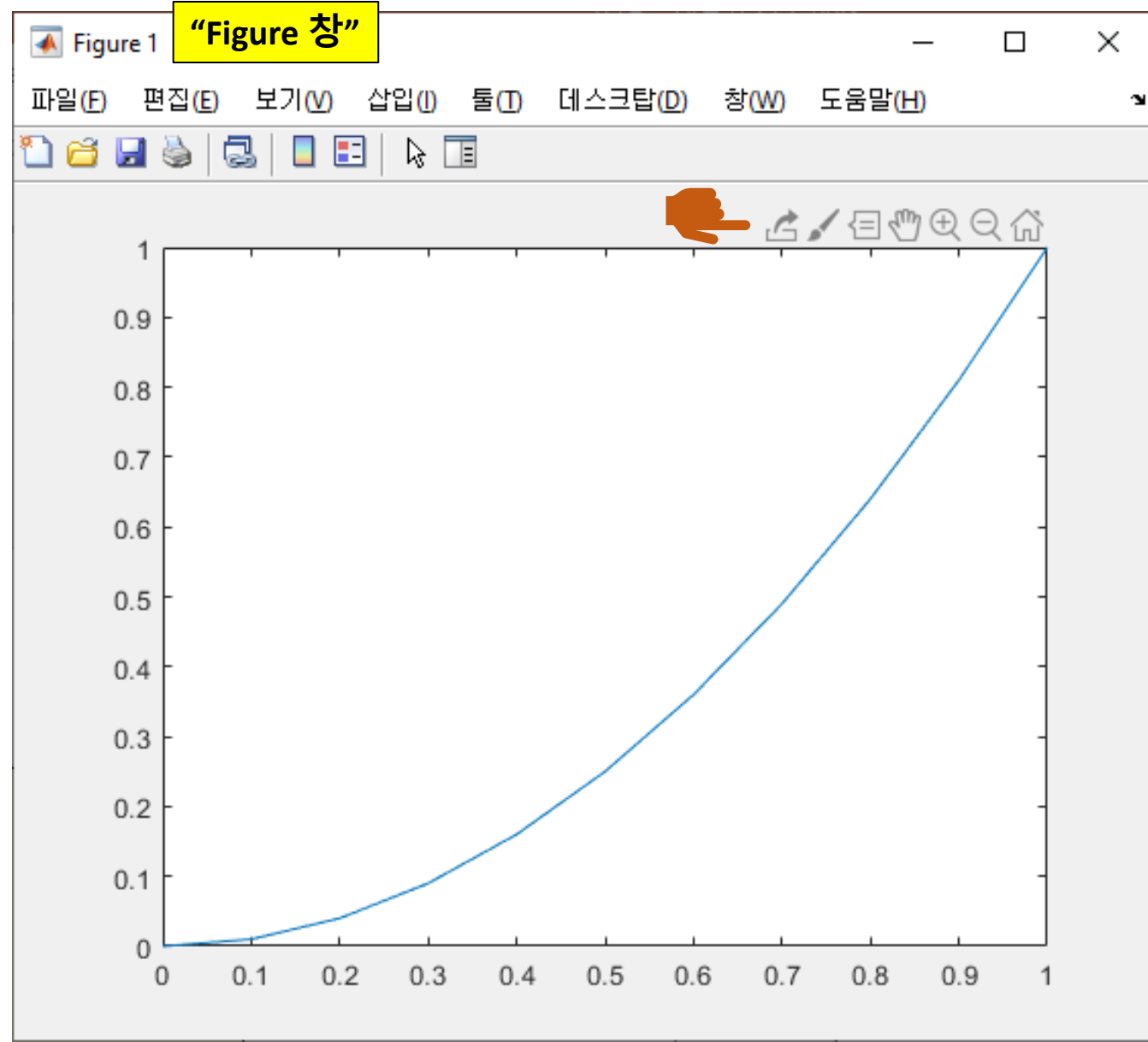


시작은 간단히 `plot(x, y)`로

* 조작패널을 간단히 이용해보자.

```
close all  
clear  
  
x = 0:0.1:1;  
y = x.^2;  
  
plot(x, y)
```

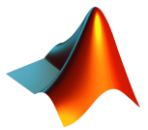
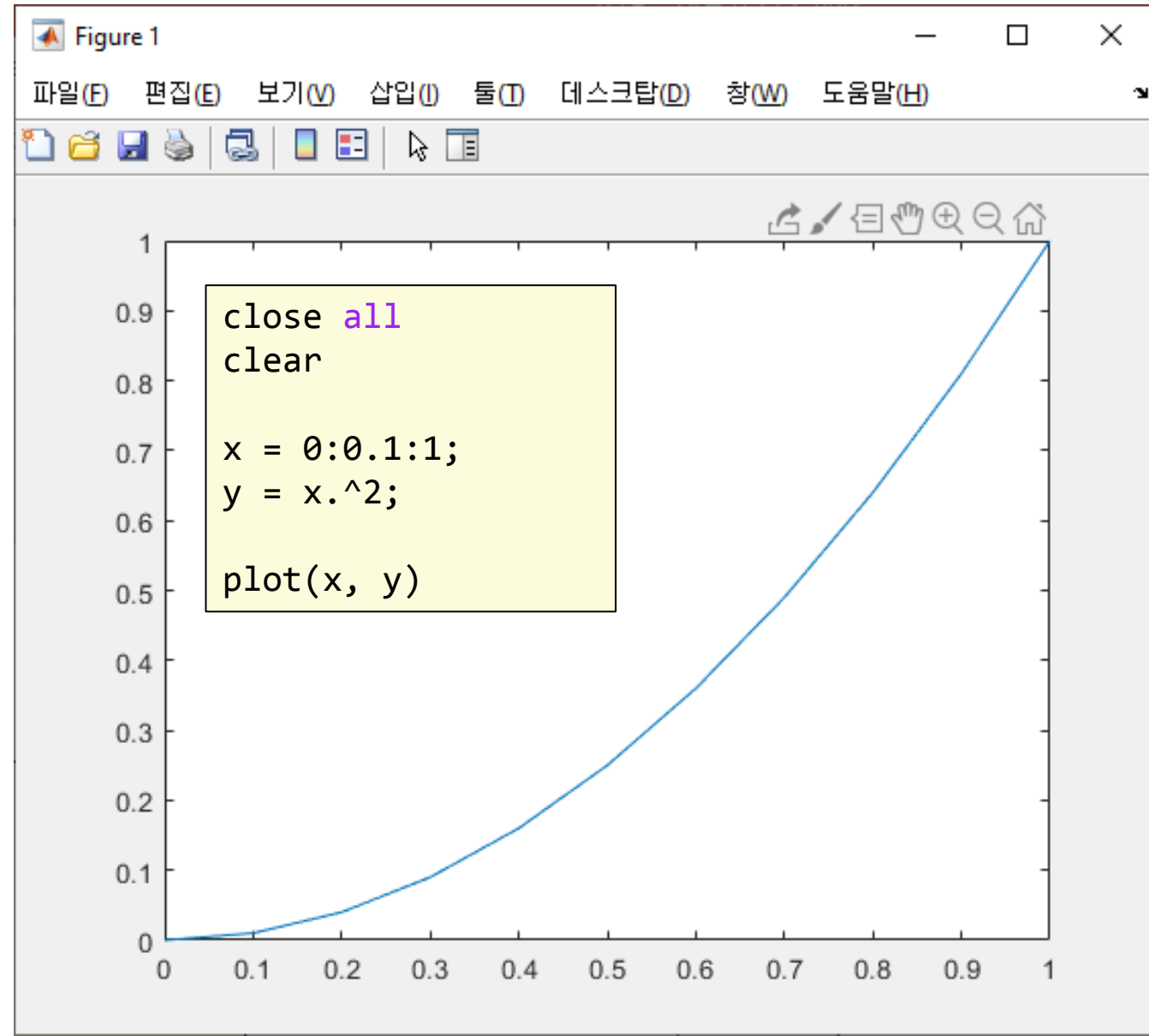
- x 와 y 의 값을 확인해보자.
 - workspace에서 확인
 - 그래프에서 점을 찍어 확인
 - 2차원 그래프는 각 점을 잇는 선분의 모음



plot(x, y)

※ 도움말을 읽자.

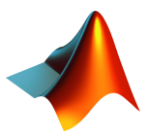
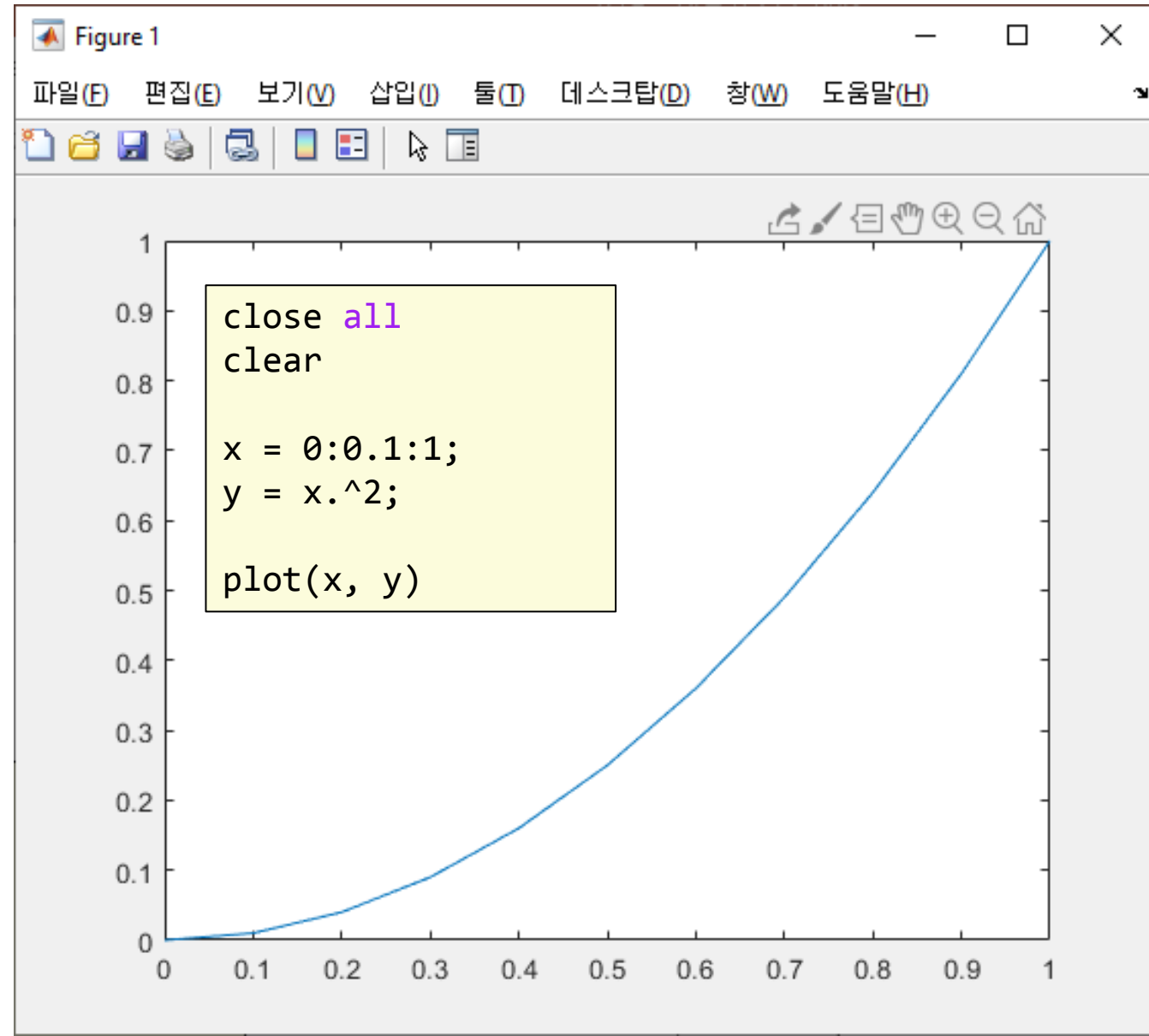
- x와 y는 길이가 같은 벡터
 - 길이가 다르면 에러
- 실습
 - $y = \sin(x)$
 - $y = \exp(-x)$
 - 연대측정 (^{14}C)
 - $N(t) = N_0 e^{-\lambda t}$
 - 감쇠진동
 - $A(t) = 3.5^{-0.5t} \sin(2\pi t)$



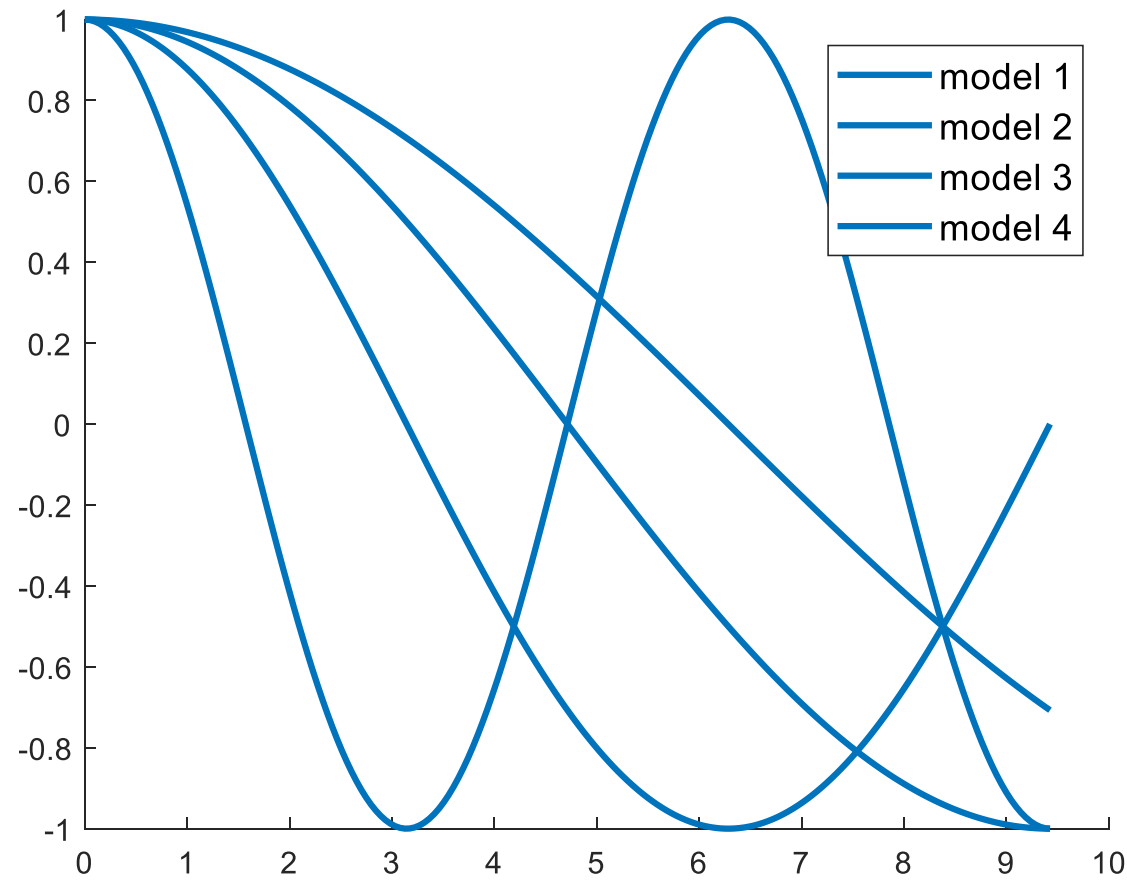
plot(x, y)

※ 도움말을 읽자.

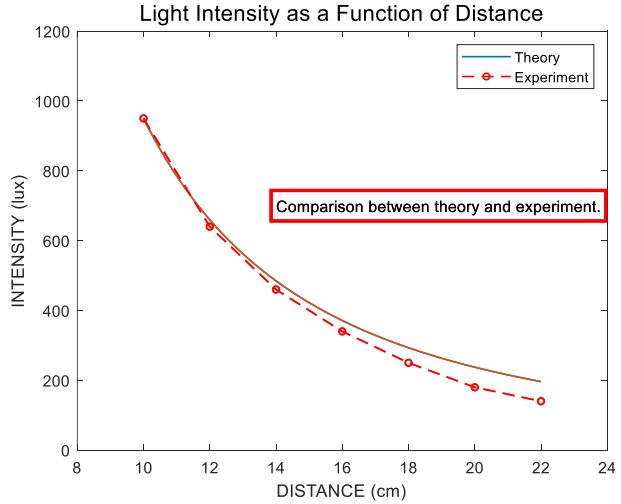
- y가 $M \times N$ 행렬이라면?
 - 뒤에 나옴
- x, y 중 하나 이상이 스칼라이면?
 - 아무것도 안 나옴
 - 왜? → 도움말을 읽어보자.
- x, y는 길이가 같으면 한다.
 - 행벡터, 열벡터 조합도 된다.
 - 그래도 가급적 맞추자.



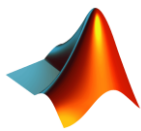
이것만으로는 부족하다...



지금까지 배운 것...



우리가 그리고 싶은 것



선 스타일을 바꾸고 싶다.

- I want to draw...

- red line



→

```
plot(x, y, 'r')
```

- blue line



→

```
plot(x, y, 'b')
```

- dashed line



→

```
plot(x, y, '--')
```

- red dotted line



→

```
plot(x, y, ':r')
```

- magenta dash-dot line



→

```
plot(x, y, 'm-.')
```

- black solid line



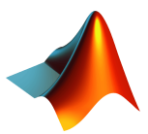
→

```
plot(x, y, 'k-')
```

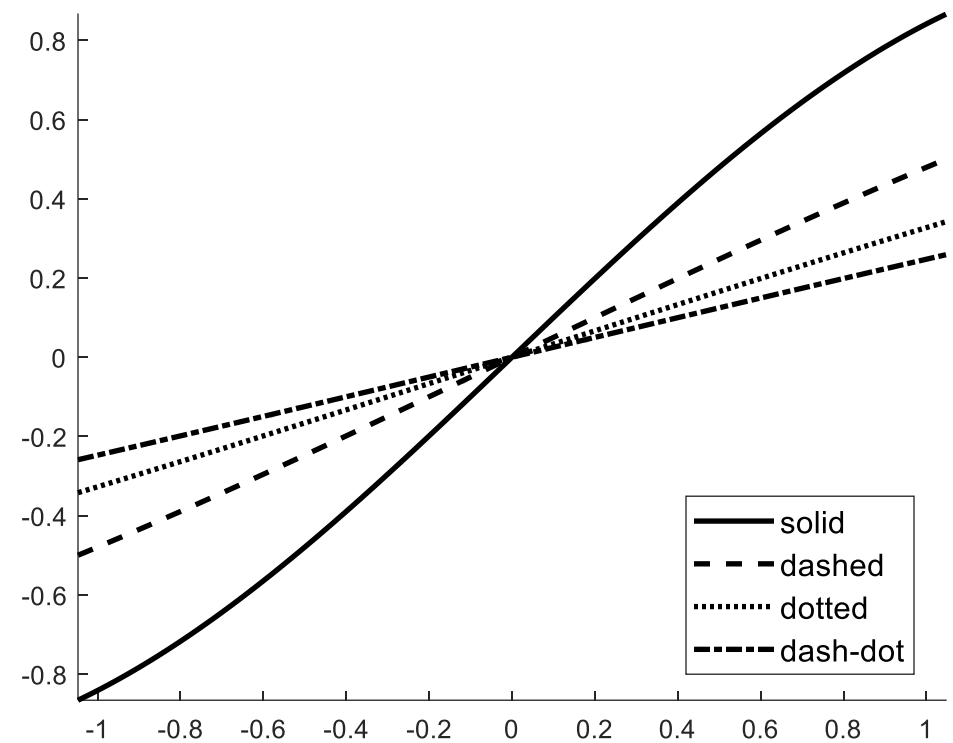
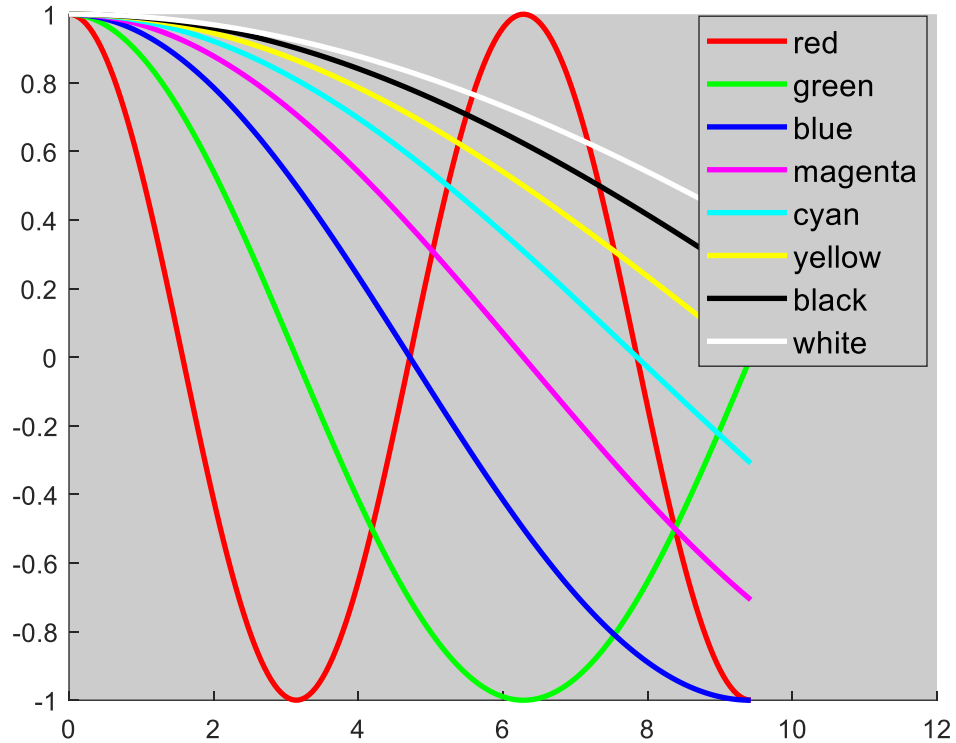
```
plot(x, y, 'k')
```

색, 스타일은
순서 바뀌어도 됨

solid line이 기본값



선 스타일 총정리



Line Color	Specifier	Line Color	Specifier
red	r	magenta	m
green	g	yellow	y
blue	b	black	k
cyan	c	white	w

Line Style	Specifier	Line Style	Specifier
solid (default)	-	dotted	:
dashed	--	dash-dot	-.

데이터 점에 마커를 붙이고 싶다.

• I want to draw...

- — + * marker
- --- + ○ marker
- -.- + ◇ marker
- + □ marker

```
plot(x, y, 'b-*')
```

```
plot(x, y, '--oc')
```

```
plot(x, y, 'k-.d')
```

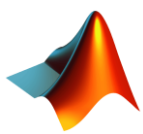
```
plot(x, y, 'rs:')
```

색, 스타일, 마커끼리
순서 바뀌어도 됨

Marker Type	Specifier		Marker Type	Specifier
plus sign	+		square	s
circle	o		diamond	d
asterisk	*		five-pointed star	p
point	.		six-pointed star	h
cross	x		triangle (pointed left)	<
triangle (pointed up)	^		triangle (pointed right)	>
triangle (pointed down)	v			


+
○
*
.
×
△
▽

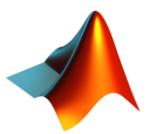
□
◇
☆
☆
△
▽



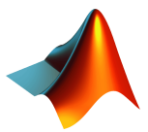
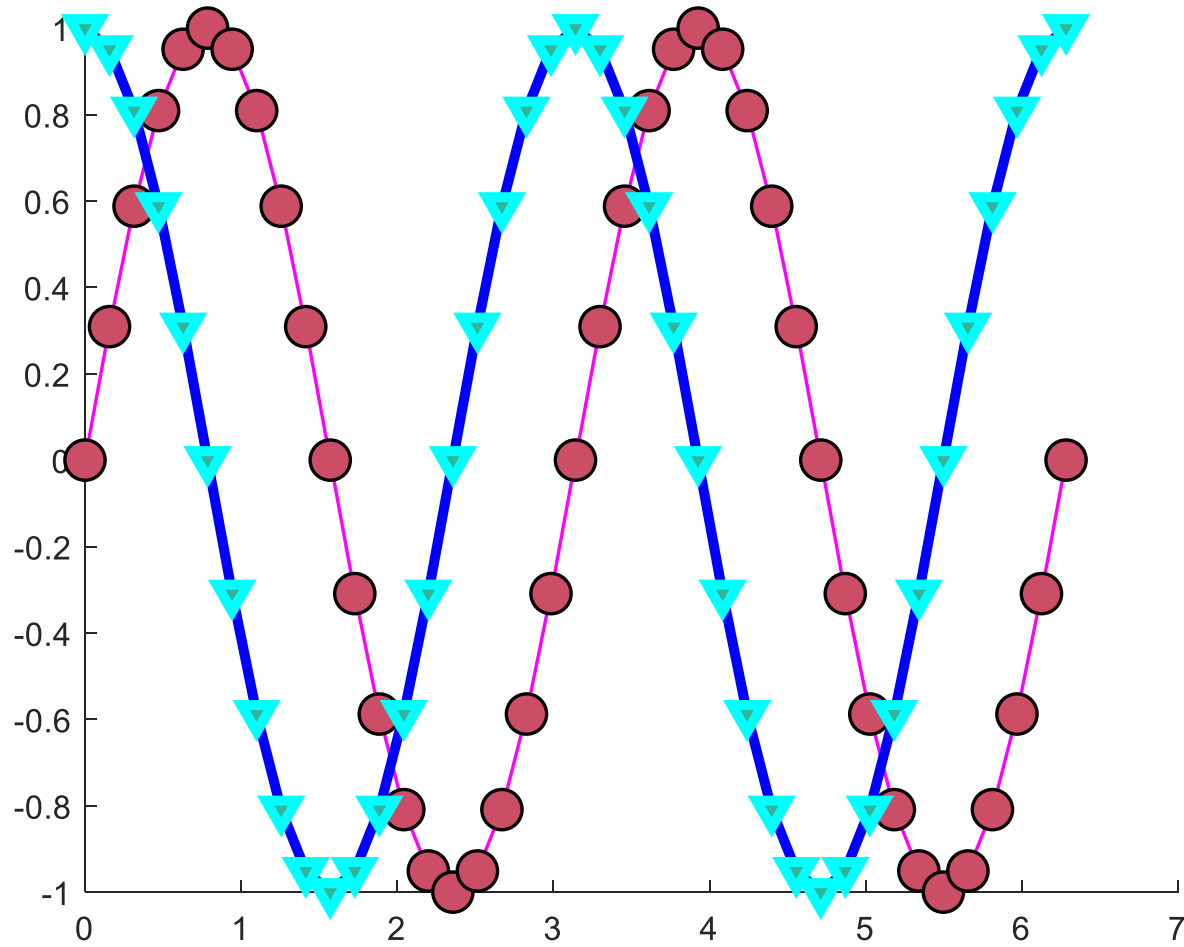
line specifier 정리

```
plot(x, y, 'lineSpec')
```

- line specifier에서 선 색, 스타일, 마커 종류 지정
 - 색, 스타일, 마커 간에는 순서 변경 가능
 - 색을 생략하면? → 기본값 = #0072BD 
 - 스타일을 생략하면? → 기본값 = solid line
 - 마커를 생략하면 → 기본값 = 마커 없음
- 주의
 - 마커를 지정하고 스타일을 지정하지 않을 경우
→ 선은 그리지 않고 마커만 그림

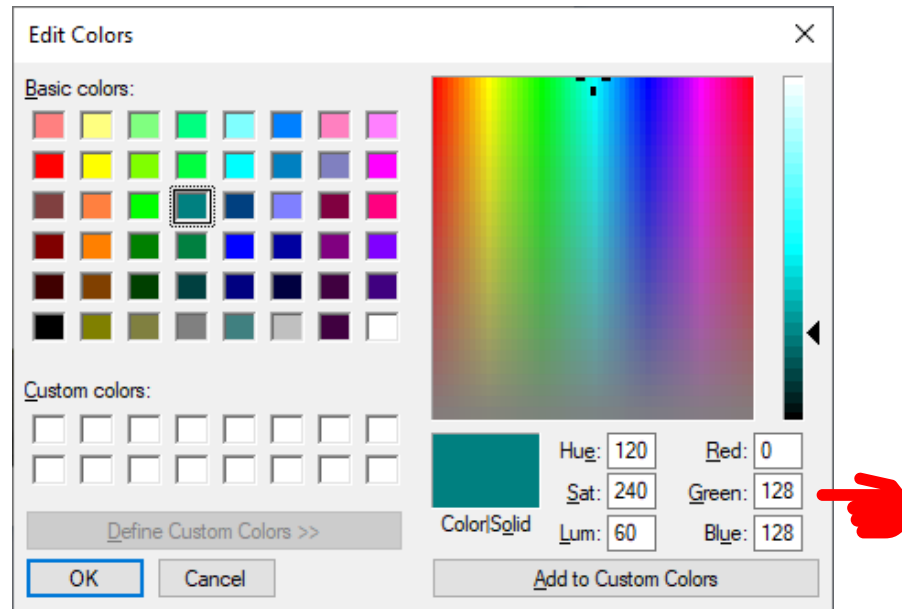


나는 아직 배고프다...

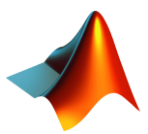


rgbmcykw가 아닌 색을 쓰고 싶다.

- 우선 컴퓨터가 색을 어떻게 만드는지 알아보자.
- 컴퓨터가 표현하는 모든 색은 RGB의 조합으로 이루어진다. (feat. 그림판)



- Recall: 컴퓨터는 모든 것을 0과 1로만 받아들인다.
 - 0 = 0b00000000 또는 0x00
 - 255 = 0b11111111 또는 0xFF



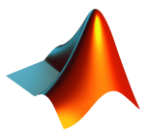
rgbmcyk가 아닌 색을 쓰고 싶다.

- 컴퓨터가 표현하는 모든 색 = 8비트 RGB의 조합

색	색 조합	8비트 조합	매트랩 표현	색 코드
RED	RED only	255, 0, 0	[1, 0, 0] 또는 'r' (#FF0000)	
GREEN	GREEN only	0, 255, 0	[0, 1, 0] 또는 'g' (#00FF00)	
BLUE	BLUE only	0, 0, 255	[0, 0, 1] 또는 'b' (#0000FF)	
MAGENTA	RED + BLUE	255, 0, 255	[1, 0, 1] 또는 'm' (#FF00FF)	
CYAN	GREEN + BLUE	0, 255, 255	[0, 1, 1] 또는 'c' (#00FFFF)	
YELLOW	RED + GREEN	255, 255, 0	[1, 1, 0] 또는 'y' (#FFFF00)	
BLACK	NO COLOR	0, 0, 0	[0, 0, 0] 또는 'k' (#000000)	
HITE	ALL COLORS	255, 255, 255	[1, 1, 1] 또는 'w' (#FFFFFF)	



tip. 색 미지정 시 plot 기본색:
0, 114, 189 = #0072BD



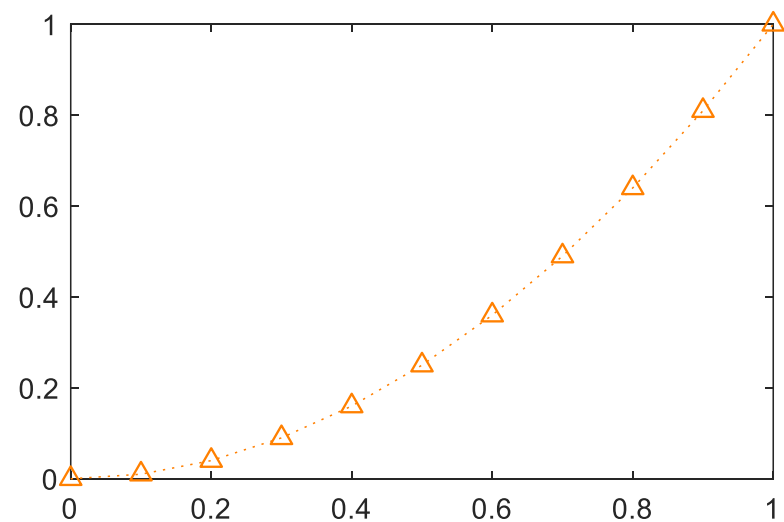
그럼 rgbmcykw에 속하지 않는 색은 어떻게 표시할까?

색	8비트 조합	매트랩 표현	색 코드
주황색	255, 128, 0	$[255, 128, 0]/255 = [1.0000, 0.5020, 0]$	#FF8000
고동색	73, 56, 49	$[73, 56, 49]/255 = [0.2863, 0.2196, 0.1922]$	#493831
국방색	83, 99, 73	$[83, 99, 73]/255 = [0.3255, 0.3882, 0.2863]$	#536349
용달블루	0, 52, 88	$[0, 52, 88]/255 = [0, 0.2039, 0.3451]$	#003458
다라이레드	168, 49, 43	$[168, 49, 43]/255 = [0.6588, 0.1922, 0.1686]$	#A8312B
마미손핑크	236, 92, 93	$[236, 92, 93]/255 = [0.9255, 0.3608, 0.3647]$	#EC5C5D

```
x = 0:0.1:1;
y = x.^2;
orange = [255, 128, 0]/255;
plot(x, y, ':^', 'color', orange)
```

property name

property value



property는 사실 color만 있는게 아니다.

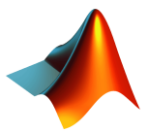
```
x = 0:0.1:1;  
y = x.^2;  
plot(x, y, 'LineStyle', 'propertyName', propertyValue)
```

※ 다시 한번 도움말을 보자.

	propertyName	의미	값	line specifier?	default
마커	marker	마커 종류	+o*.x^vsdph<>	Y	none
	markeredgecolor	마커 테두리 색	rgbmcykw 또는 1x3 벡터	N	auto
	markerfacecolor	마커 면 색		N	none
	markersize	마커 크기	양의 실수	N	6
선	color	선 색	rgbmcykw 또는 1x3 벡터	Y	[0 0.4470 0.7410]
	linestyle	선 스타일	- , -- , : , -.	Y	solid (-)
	linewidth	선 두께	양의 실수	N	0.5

※ marker, color, linestyle은 LineSpec과 propertyName 모두 가능

※ property는 여러 개 동시 설정 가능



맨 처음 나왔던 그래프를 그려보자.

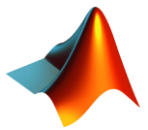
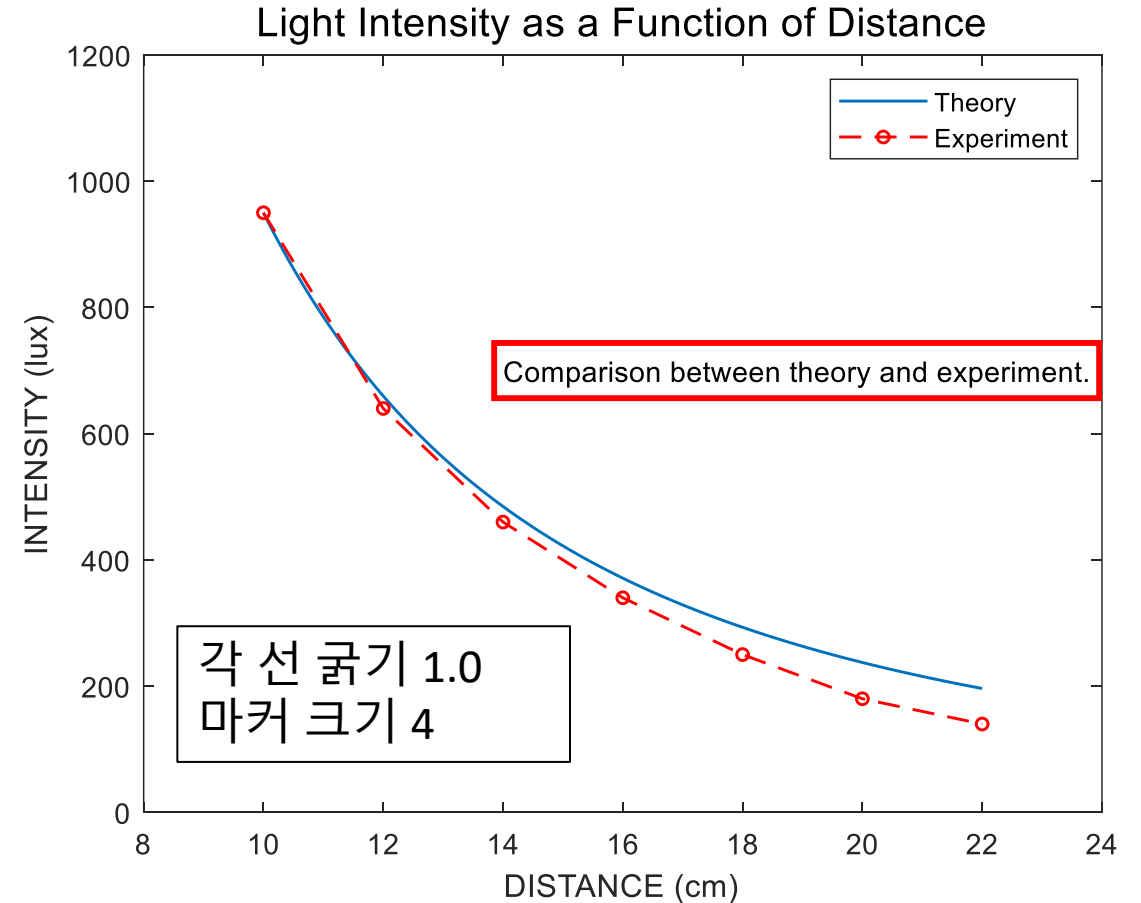
```
close all
clc

x = 10:0.1:22;
y = 95000./(x.^2);

xd = 10:2:22;
yd = [950 640 460 340 250 180 140];

plot(x,y,'-','linewidth',1.0)

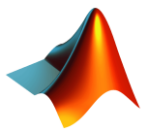
plot(xd,yd,'ro--', ...
      'linewidth',1.0,'markersize',4)
```



한 figure 창에 그래프 여러 개 그리는 방법 1

```
x = 0:0.1:1;  
y = x.^2;  
u = 0:0.2:1;  
v = exp(u)-1;  
t = linspace(0,1,20);  
h = tan(pi/4*t);  
  
plot(x, y, u, v, t, h)
```

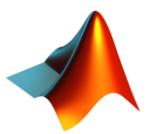
- 각 (x, y) pair 뒤에 line specifier를 적을 수 있음
- 장점
 - 여러 (x, y) pair를 한줄컷 가능
 - 각 pair의 길이가 달라도 됨
- 단점
 - (x, y) pair마다 property를 다르게 하지 못함 (동일한 property는 가능)



한 figure 창에 그래프 여러 개 그리는 방법 2

```
x = 0:0.1:1;  
y = x.^2;  
u = 0:0.2:1;  
v = exp(u)-1;  
t = linspace(0,1,20);  
h = tan(pi/4*t);  
  
plot(x, y, 'r*-','linewidth', 1)  
hold on,  
plot(u, v, 'g', 'marker', 'v')  
plot(t, h, 'b-.x')
```

- 장점
 - 겹침/안겹침을 하나씩 제어 가능
 - plot마다 line specifier와 property 설정 가능
- 단점
 - 여러 줄 써야 함



한 figure 창에 그래프 여러 개 그리는 방법 3

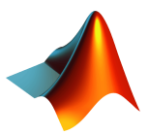
```
x = 0:0.1:1;  
y = x.^2;  
u = 0:0.2:1;  
v = exp(u)-1;  
t = linspace(0,1,20);  
h = tan(pi/4*t);  
  
line(x, y)  
line(u, v)  
line(t, h)
```

- 장점

- hold on 없이 간단하게 겹쳐 그릴 수 있음

- 단점

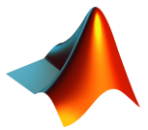
- line specifier를 쓰지 못하며, 모든 설정은 property로만 가능



한 figure 창에 그래프 여러 개 그리는 방법 4

```
x = 0:0.1:1;  
u = x.^2;  
v = exp(x)-1;  
w = tan(pi/4*x);  
  
y = [u; v; w];  
  
plot(x, y, 'rs-', 'linewidth', 2)
```

- x도 행렬일 경우 → x, y의 각 column을 pair로 plot
- 장점
 - 한줄컷 가능
- 단점
 - 선마다 line specifier 또는 property 지정 불가 (동일하게는 가능)
 - 모든 pair의 길이가 같아야 함



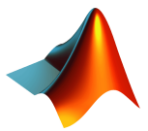
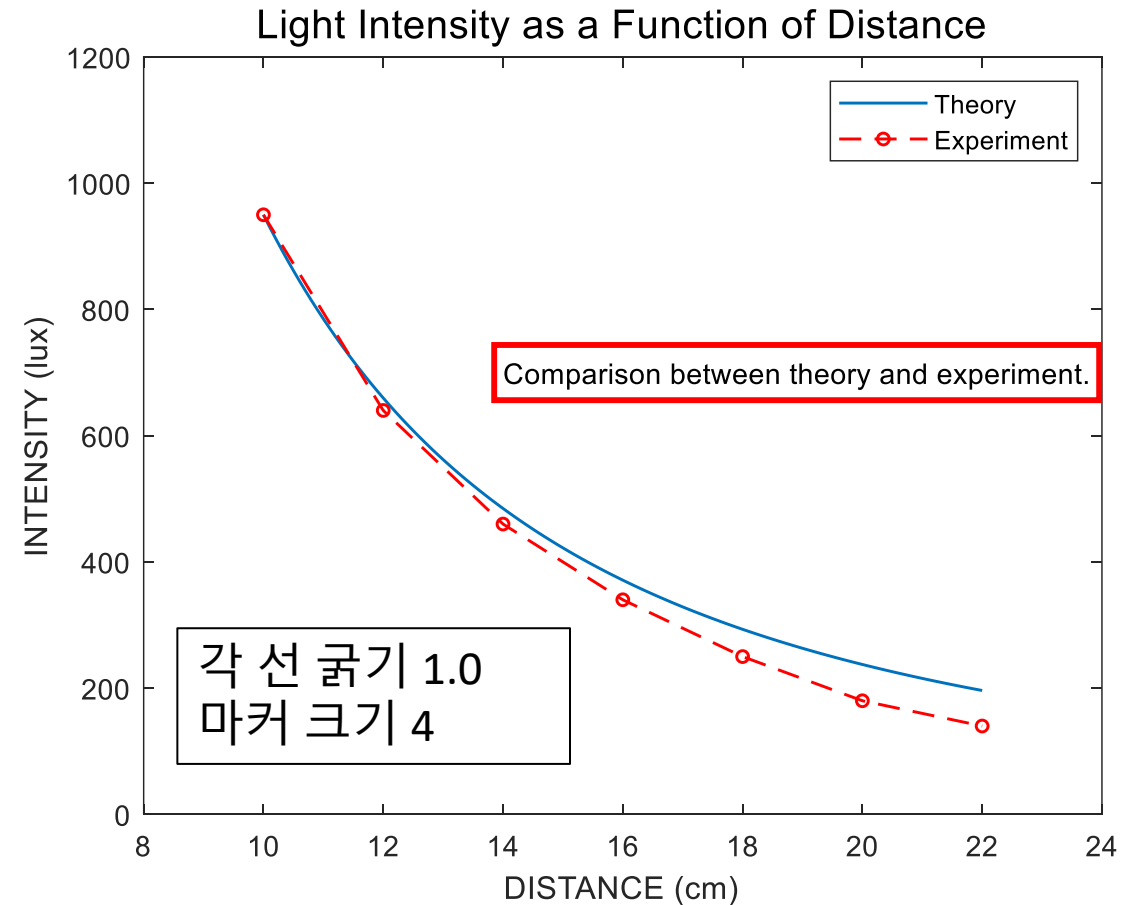
맨 처음 나왔던 그래프를 다시 그려보자.

```
close all
clc

x = 10:0.1:22;
y = 95000./(x.^2);

xd = 10:2:22;
yd = [950 640 460 340 250 180 140];

plot(x,y,'-','linewidth',1.0)
hold on
plot(xd,yd,'ro--', ...
     'linewidth',1.0,'markersize',4)
```



title, axis label, legend, textbox

※ 도움말을 보자.

```
xlabel('DISTANCE (cm)')  
ylabel('INTENSITY (lux)')  
title('Light Intensity as a Function of Distance');  
legend('Theory', 'Experiment');  
text(14, 700, 'Comparison between theory and experiment.');
```

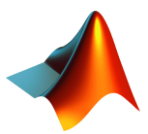
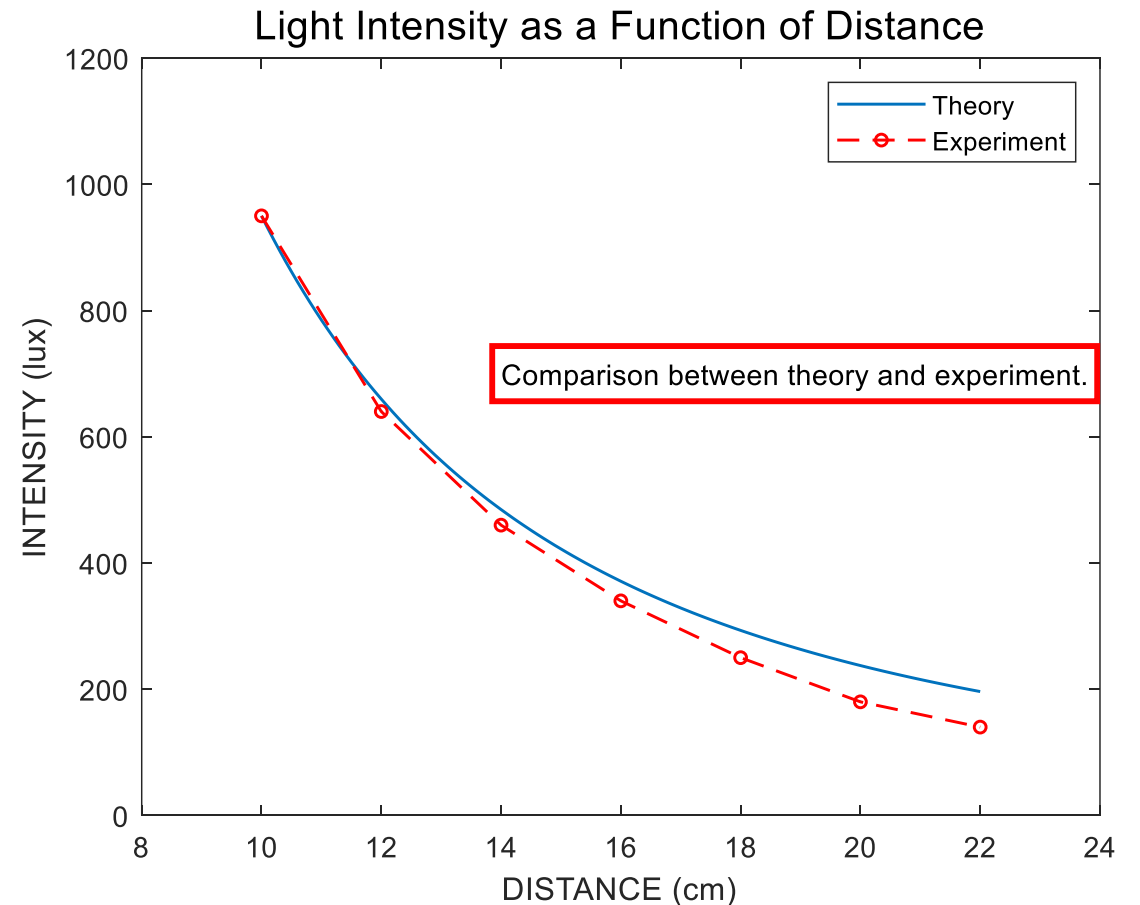
- 이 항목들도 property를 설정할 수 있다.
 - 어떤 항목들이 있는지는 도움말을 보자.
- 텍스트를 따로 변수로 두고 써도 된다.
- legend는 figure 창에 plot한 순서대로 매겨진다.
- gtext: 마우스로 textbox의 위치 지정 가능

※※ plot은 무조건 title과 label을 붙이는게 국룰이다.

축 범위를 조절해보자.

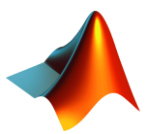
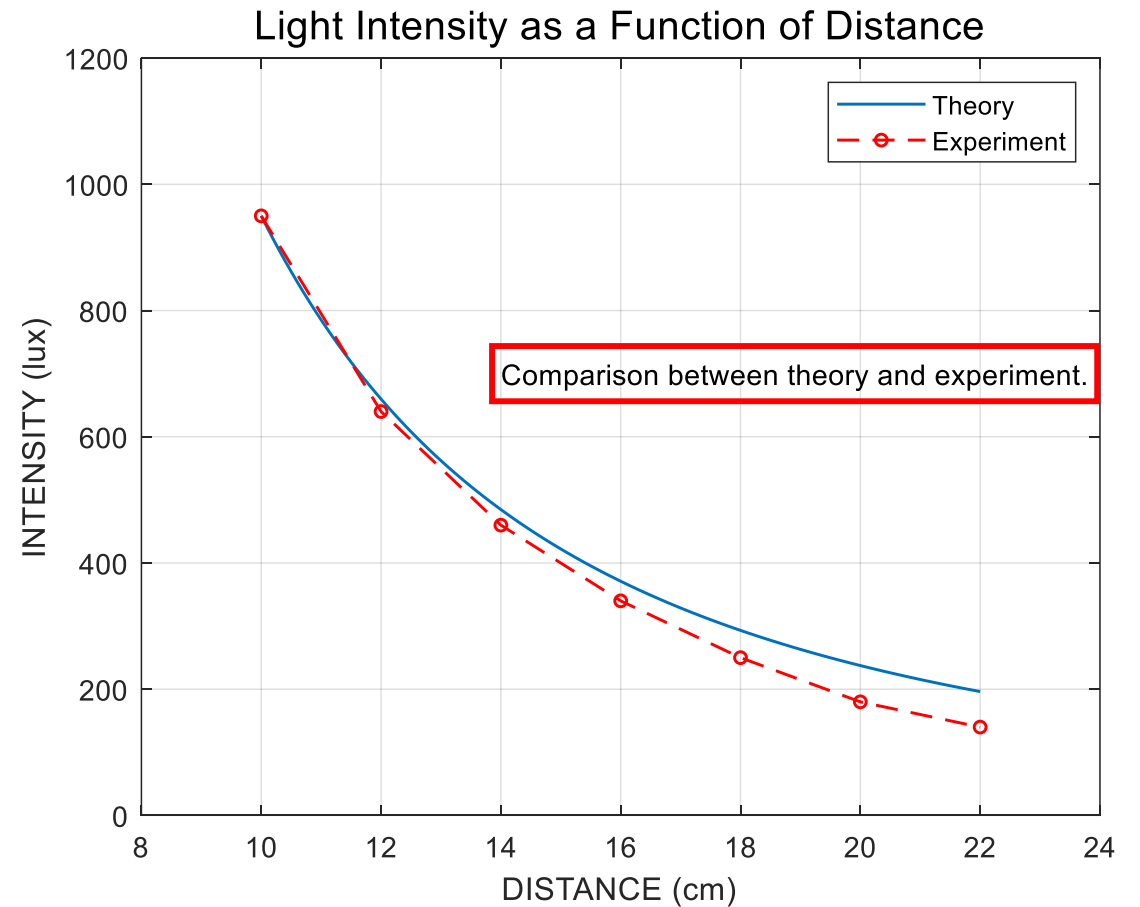
```
axis([8 24 0 1200]) % [xmin, xmax, ymin, ymax]  
xlim([8 24])        % [xmin, xmax]  
ylim([0 1200])      % [ymin, ymax]
```

- axis off : 축 지우기
- axis on : 축 표시하기
- axis tight : 여백 없애기
- axis square : 축을 정사각형으로
- axis auto : 알아서 해줘
- axis equal : x, y 같은 스케일로



눈금과 테두리를 넣어보자.

```
grid on % <-> grid off  
box on  % <-> box off
```

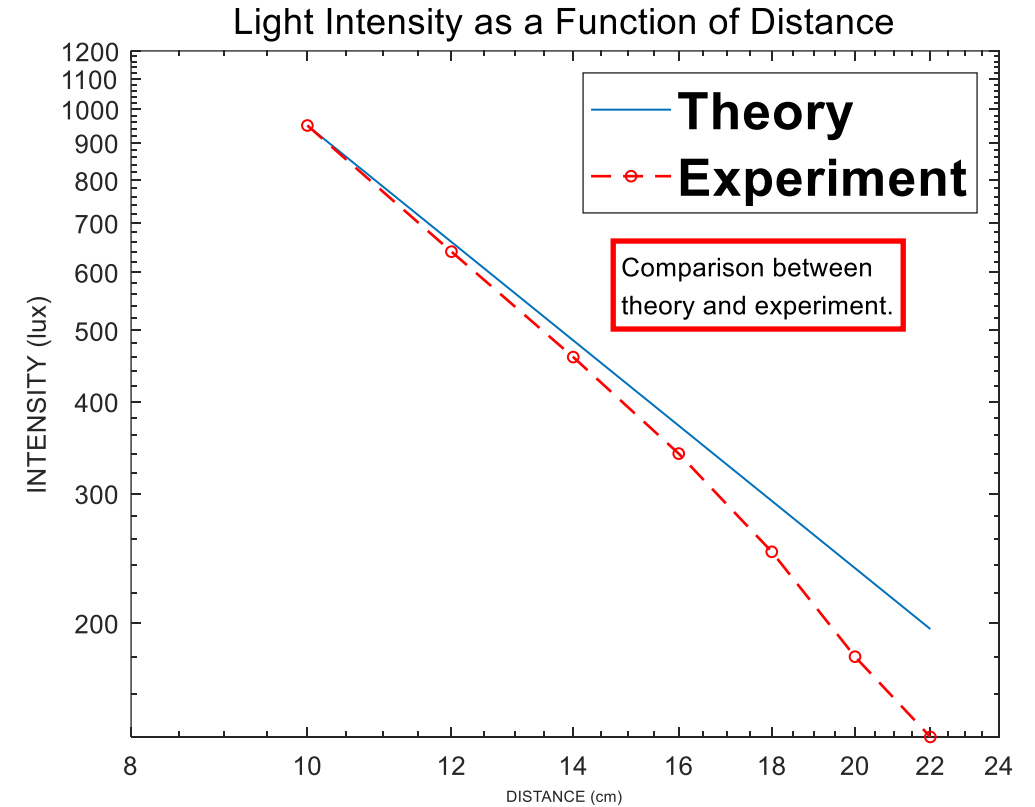


축을 log scale로 바꿔보자.

```
x = 10:0.1:22;  
y = 95000./(x.^2);  
  
xd = 10:2:22;  
yd = [950 640 460 340 250 180 140];  
  
loglog(x,y,'-','linewidth',1.0) ➡  
hold on  
plot(xd,yd,'ro--',...  
      'linewidth',1.0,'markersize',4)
```

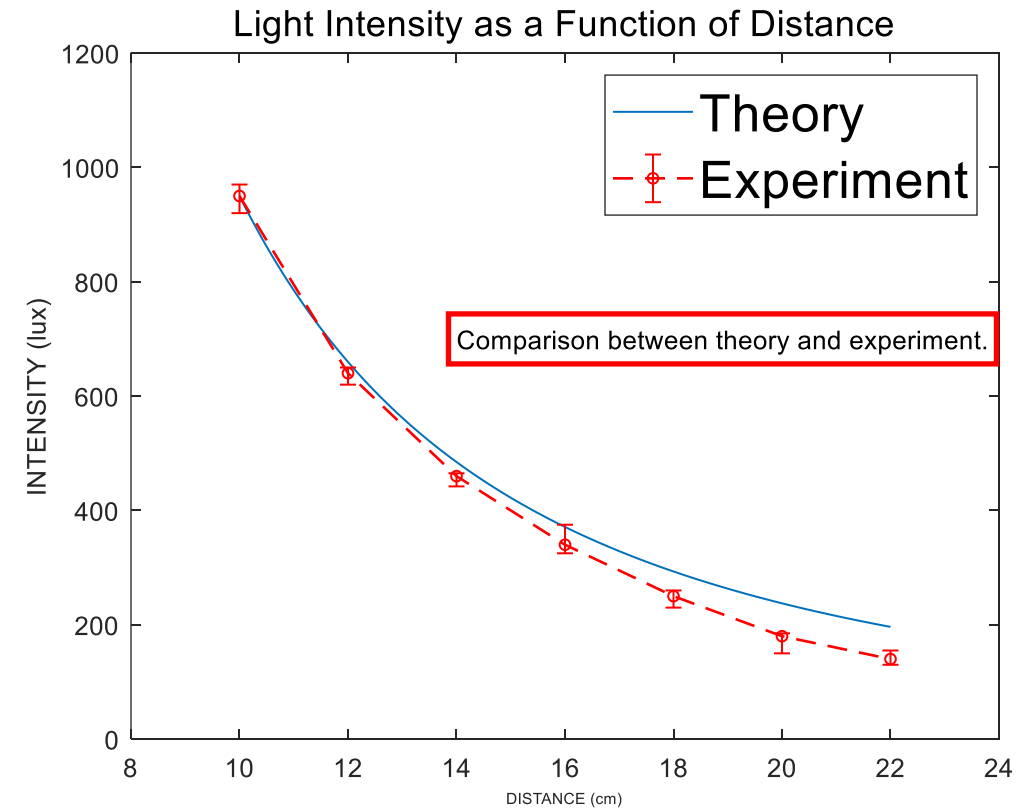
- semilogx → x축은 log, y축은 linear
- semilogy → x축은 linear, y축은 log
- loglog → x, y축 모두 log
- 주의

• 0보다 작거나 같은 수는 log 값이 정의되지 않는다. (실수 범위 내에서)

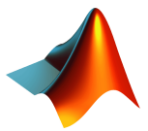


그래프에 error bar를 추가해보자.

```
x = 10:0.1:22;  
y = 95000./(x.^2);  
  
xd = 10:2:22;  
yd = [950 640 460 340 250 180 140];  
ydErrNeg = [30 20 18 15 20 30 10];  
ydErrPos = [20 10 5 35 10 5 15];  
  
plot(x,y, '-', 'linewidth',1.0)  
hold on  
errorbar(xd, yd, ydErrNeg, ydErrPos, ...  
         'ro--', 'linewidth',1.0, 'markersize', 4)
```

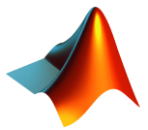
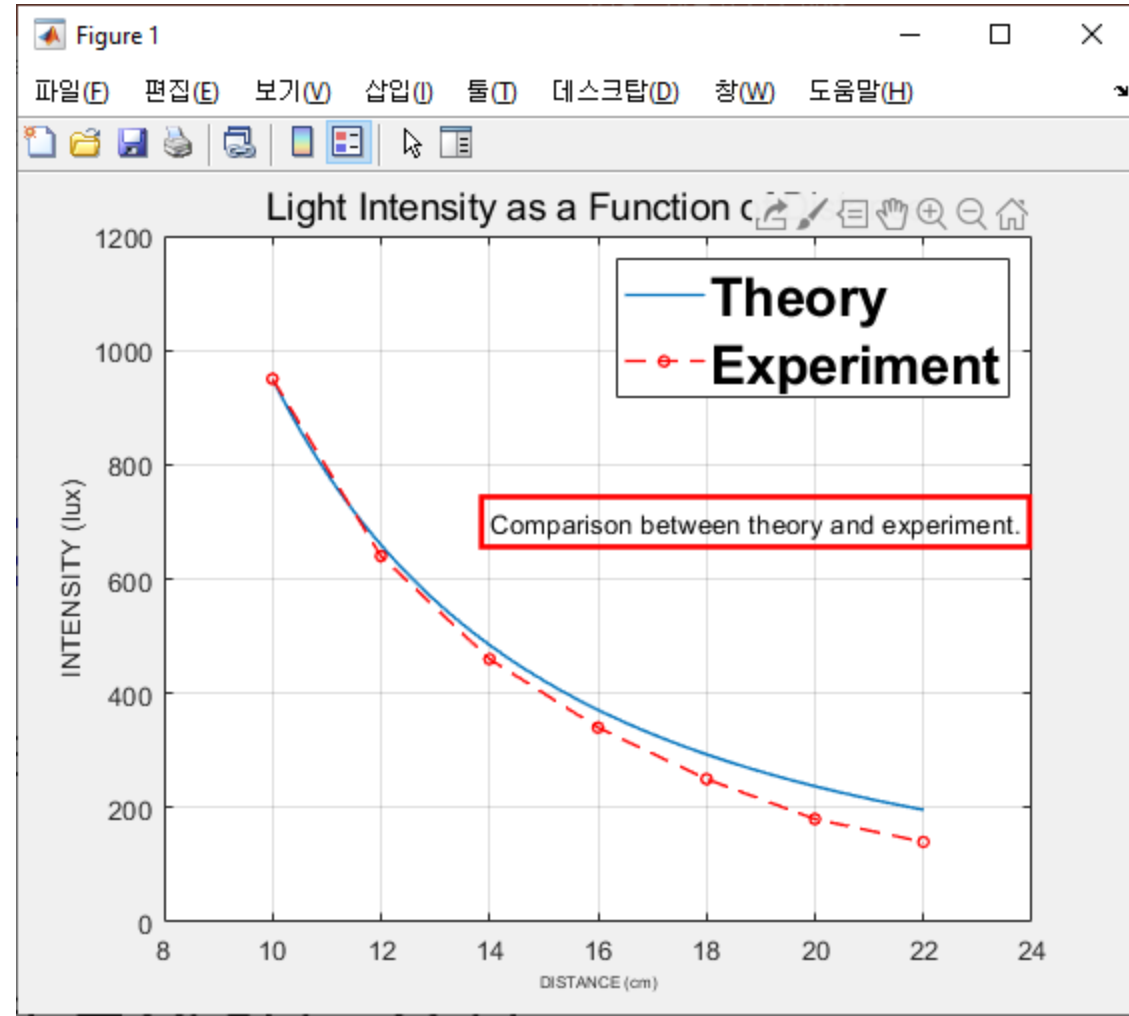


- 상한=하한이라면 error term은 한번만 적으면 됨
 - errorbar(xd, yd, ydErr, ...)



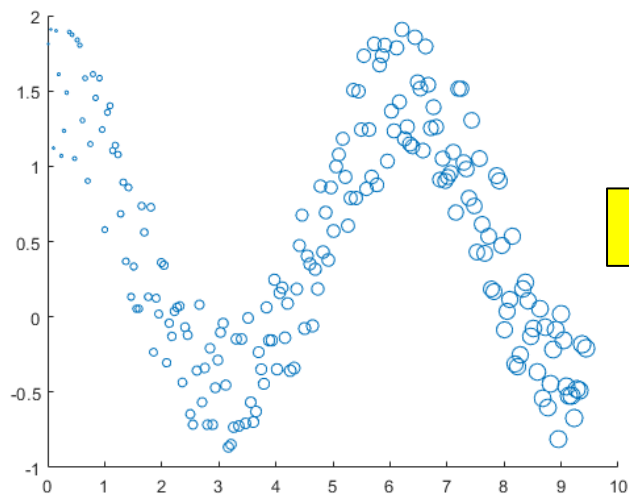
툴바와 조작패널을 이용해보자.

- legend on/off
- 그래픽 객체 선택 및 이동
- 데이터 브러쉬
- 데이터 팁
 - 팁 버튼 눌렀을 때 vs 안 눌렀을 때
 - 팁 지우기
- 팬, 확대, 축소, 원래대로
- 그림 복사하는 방법

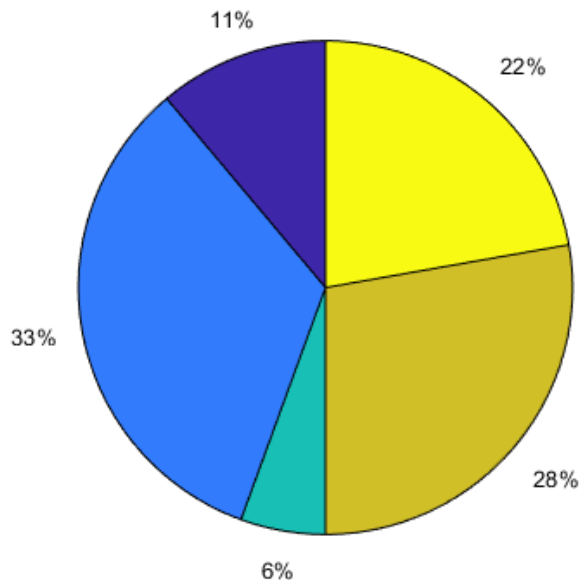


특별한 그래프들

※ 도움말을 보자.



scatter

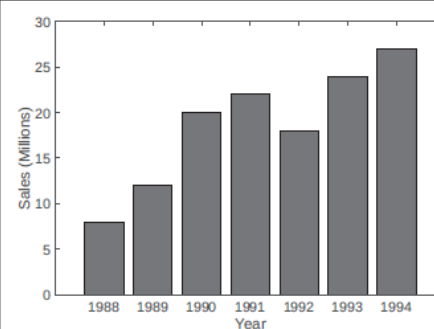


pie

Vertical Bar Plot

Function format:

`bar(x,y)`



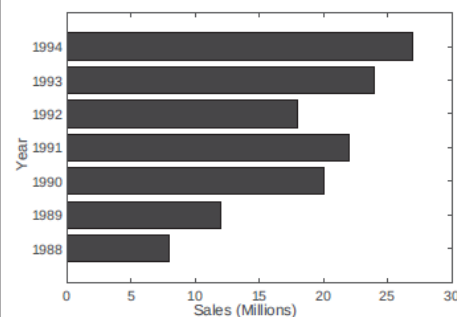
```
yr=[1988:1994];
sle=[8 12 20 22 18 24 27];
bar(yr,sle,'r')
xlabel('Year')
ylabel('Sales (Millions)')
```

The bars are in red.

Horizontal Bar Plot

Function format:

`barh(x,y)`

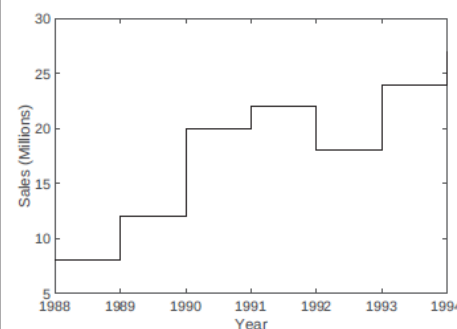


```
yr=[1988:1994];
sle=[8 12 20 22 18 24 27];
barh(yr,sle)
xlabel('Sales (Millions)')
ylabel('Year')
```

Stairs Plot

Function format:

`stairs(x,y)`

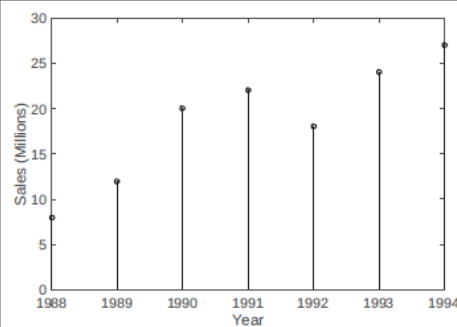


```
yr=[1988:1994];
sle=[8 12 20 22 18 24 27];
stairs(yr,sle)
xlabel('Year')
ylabel('Sales (Millions)')
```

Stem Plot

Function Format

`stem(x,y)`



```
yr=[1988:1994];
sle=[8 12 20 22 18 24 27];
stem(yr,sle)
xlabel('Year')
ylabel('Sales (Millions)')
```

히스토그램을 그려보자.

※ 도움말을 보자.

```
close all
clear

x = rand(10000,1);
histogram(x, 30)
histogram(x, linspace(0,1,100))

xn = randn(10000,1);
histogram(xn, 30)
histogram(xn, linspace(-3,3,100))
```

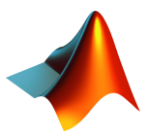
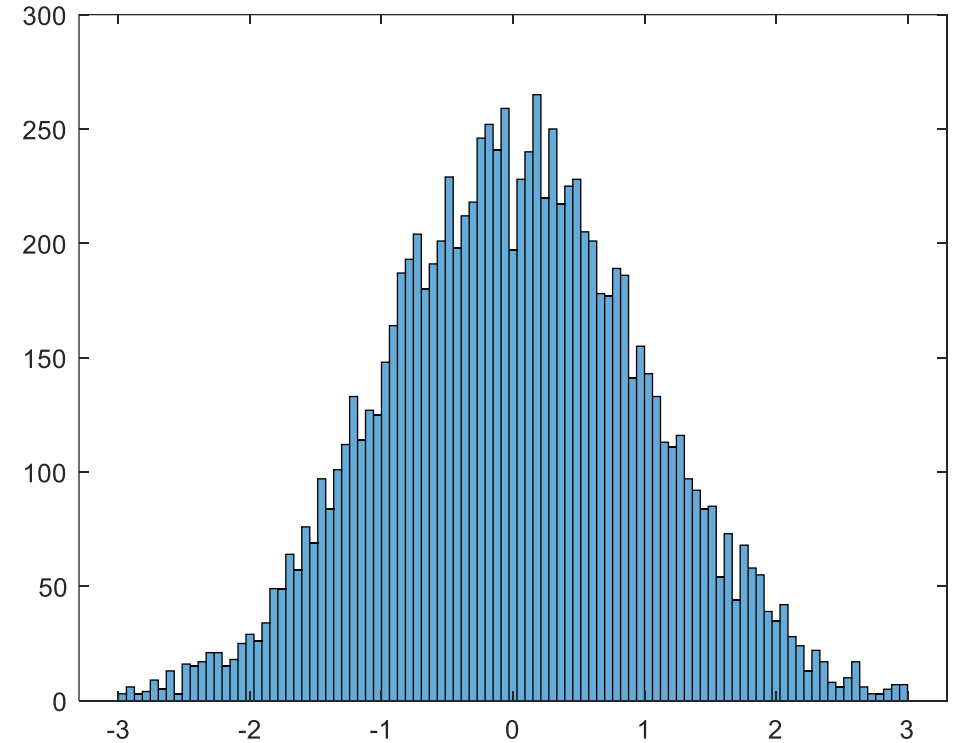


figure 창을 여러 개 띄워보자.

- 지금까지는 하나의 figure 창에 덮어썼다. (hold on, hold off로 조절)
 - plot을 할 때마다 현재 figure 창에 덮어쓴다. (현재 figure 창 = 현재 활성화된 창)

tip. 현재 활성화된 창 번호?
gcf

- 새로운 figure 창을 띄우고 싶다면?

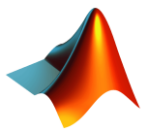
```
figure; % 새로운 창이 생성되고, 창 번호가 부여됨과 동시에 활성화
```

- 특별한 번호를 가진 창을 생성하고 싶다면?

```
figure(2); % 2번 창이: 없다면 2번 창이 생성됨, 있다면 2번 창 활성화  
figure(100); % 100번 창이: 없다면 100번 창이 생성됨, 있다면 100번 창 활성화
```

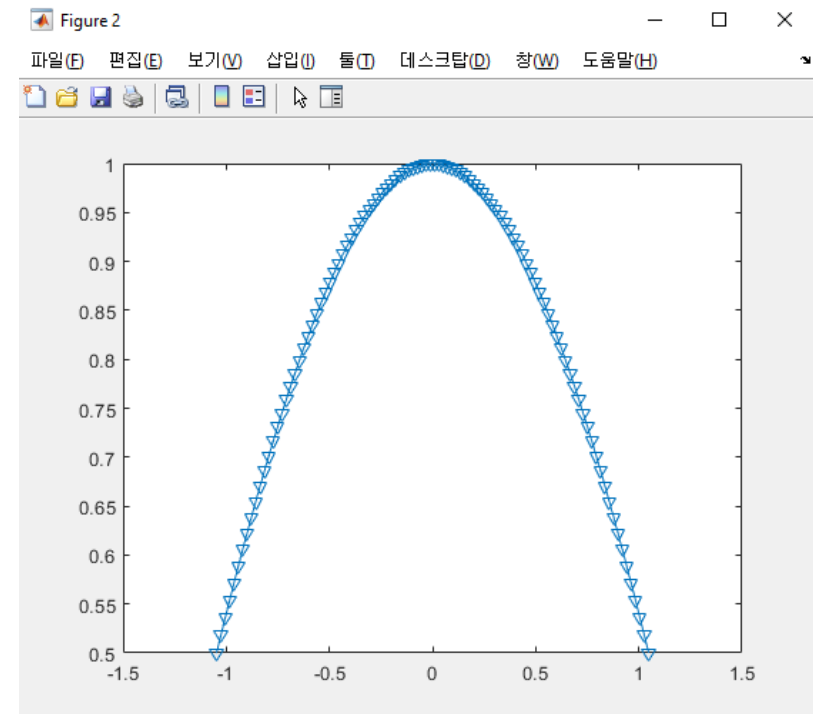
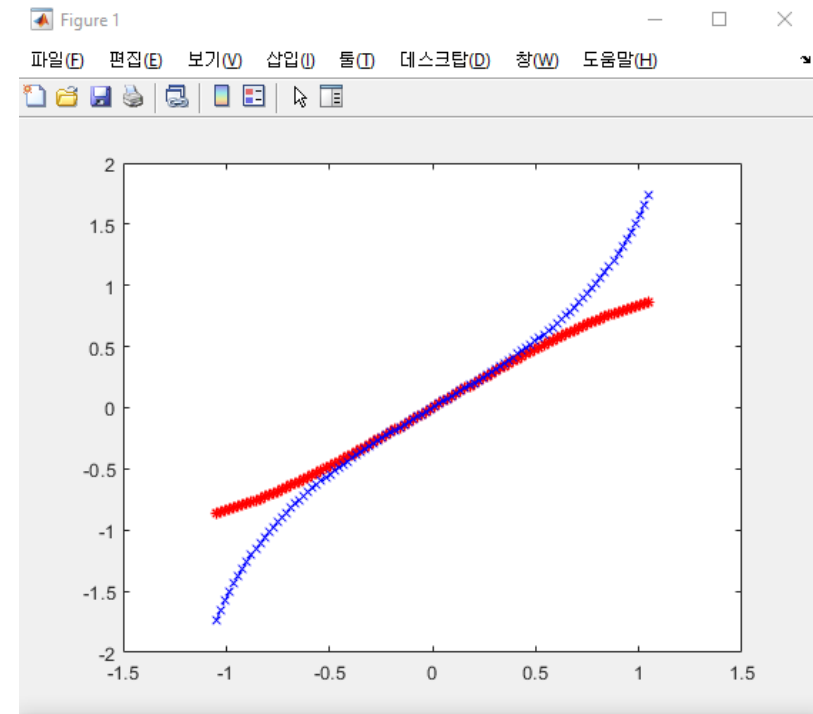
- 이 방식으로 figure 창 간 전환도 가능 (창 전환 = 활성화)
- 없는 창은 생성과 동시에 활성화
- 3번 figure 창에 plot하고 싶다면?

```
figure(3); % 3번 창이: 없다면 3번 창이 생김, 있다면 3번 창 활성화  
plot(x, y, ...)
```



여러 창에 plot을 해보자.

```
x = linspace(-pi/3,pi/3);  
sinx = sin(x);  
cosx = cos(x);  
tanx = tan(x);  
  
figure, % opening figure(1)  
plot(x, sinx, 'r*-','linewidth', 1)  
  
figure, % opening figure(2)  
plot(x, cosx, 'marker', 'v')  
  
figure(1), % activating figure(1)  
hold on  
plot(x, tanx, 'b-.x')
```



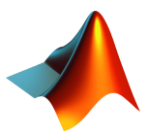
창이 너무 많다. 좀 닫고 싶은데...

```
clear
close all * CCC 패턴
clc

x = linspace(-pi/3,pi/3);
sinx = sin(x);
cosx = cos(x);
tanx = tan(x);

figure, plot(x, sinx, 'r*-','linewidth', 1)
figure, plot(x, cosx, 'marker', 'v')
figure, plot(x, tanx, 'b-.x')

close      % closing current active figure, which is figure(3)
close(1)   % closing figure(1)
close all  % closing all figures
```



한 figure 창에 여러 그래프를 그려보자.

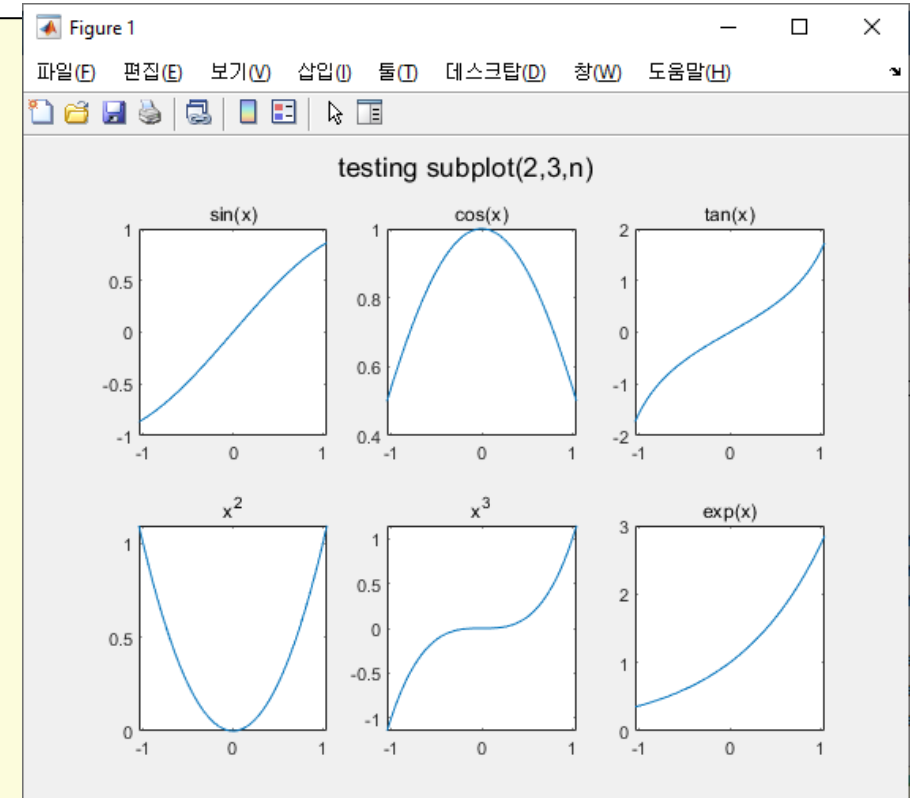
```
clear
close all
clc

x = linspace(-pi/3,pi/3);

figure,
subplot(2,3,1), plot(x, sin(x)), title('sin')
subplot(2,3,2), plot(x, cos(x)), title('cos')
subplot(2,3,3), plot(x, tan(x)), title('tan')
subplot(2,3,4), plot(x, x.^2), title('x^2')
subplot(2,3,5), plot(x, x.^3), title('x^3')
subplot(2,3,6), plot(x, exp(x)), title('exp(x)')
```

```
subplot(2,3,1), title('sin(x)') % activating subplot(2,3,1)
subplot(2,3,2), title('cos(x)') % activating subplot(2,3,2)
subplot(2,3,3), title('tan(x)') % activating subplot(2,3,3)
```

```
sgtitle('testing subplot(2,3,n)')
```



(2,3,1)	(2,3,2)	(2,3,3)
(2,3,4)	(2,3,5)	(2,3,6)

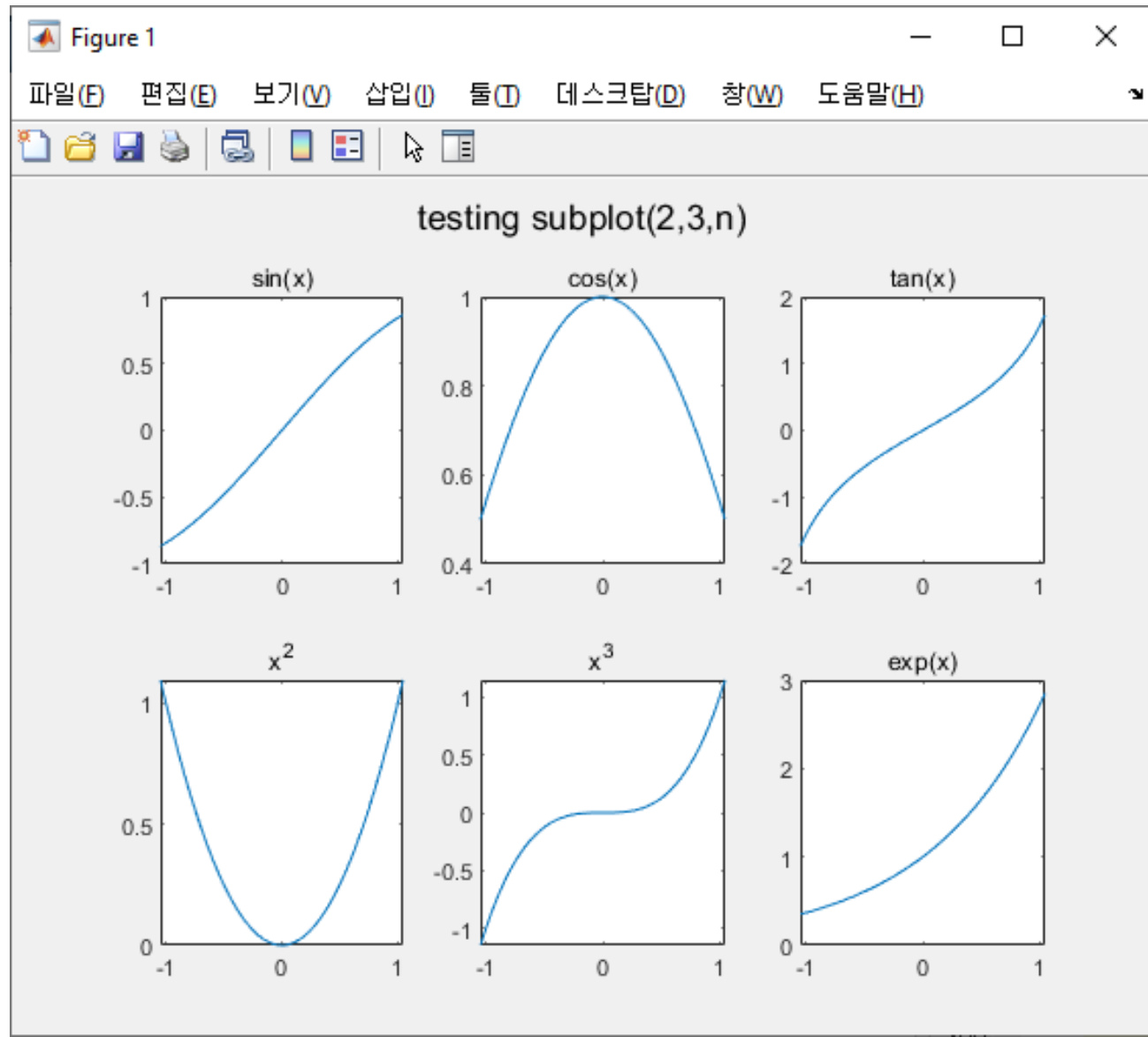
그래프를 저장하자.

- 그림 저장하는 방법

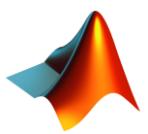
- 파일 메뉴 or 조작패널
- .fig 파일로 저장
- 그림 파일로 저장
- 명령창에서 저장
 - `saveas(figure번호, '파일명')`

※ save와 헛갈리지 말 것
※ 헛갈리면 자동완성 이용

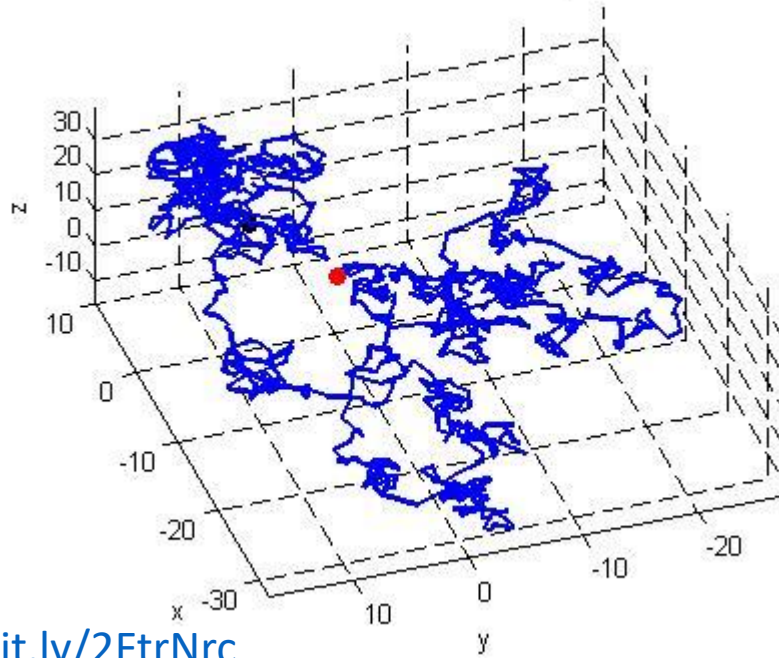
- 마우스(GUI) vs 명령창(CUI)



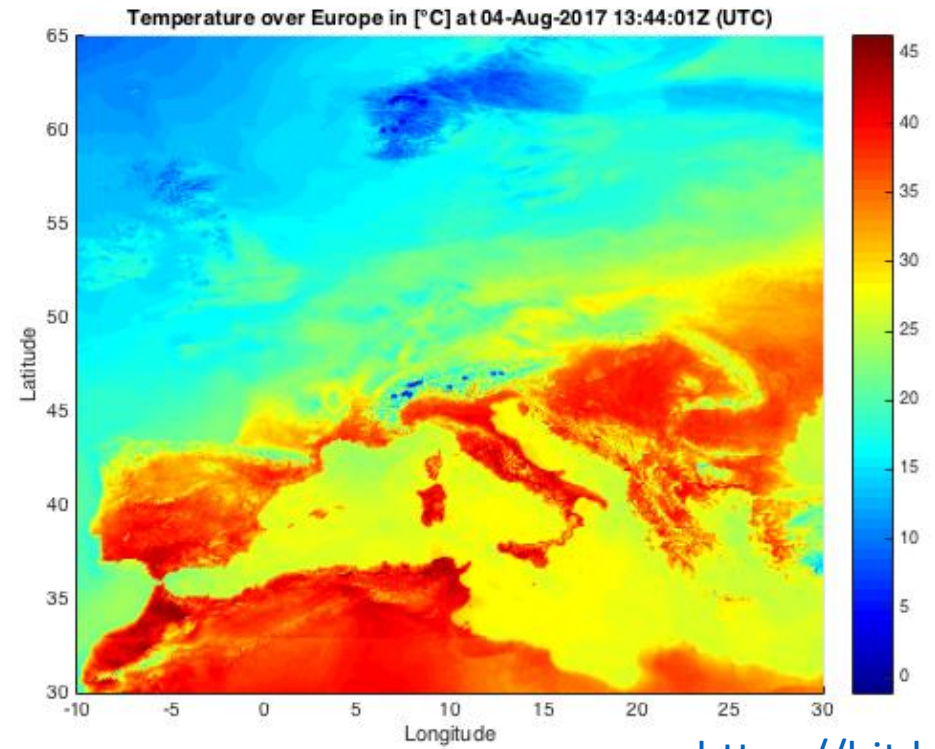
3차원 그래프



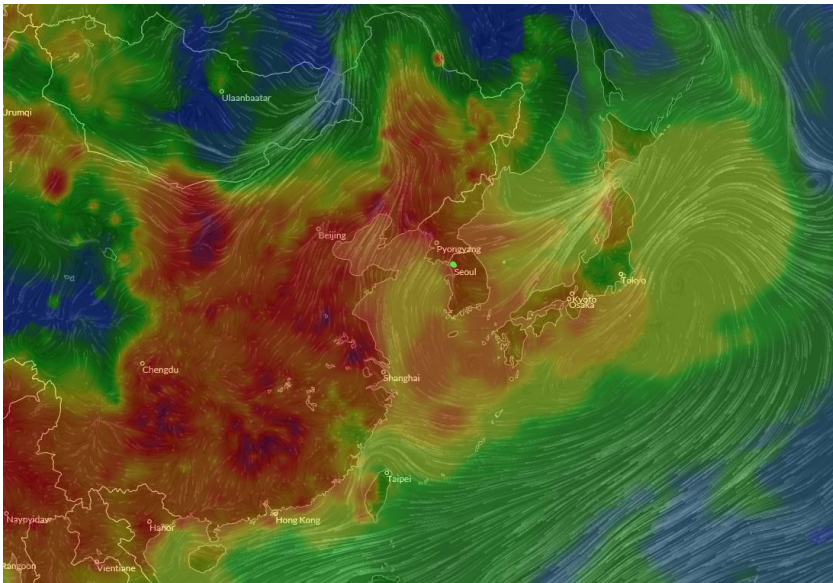
Three dimensional Brownian Motion, d=20.6188 units



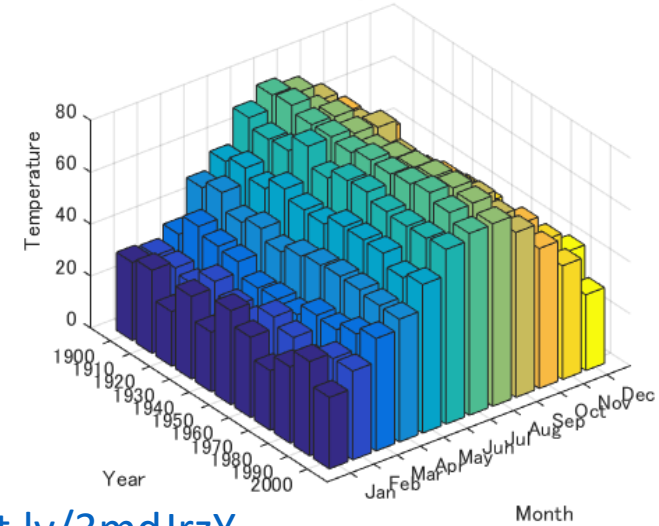
<https://bit.ly/2FtrNrc>



<https://bit.ly/3hs1Y7U>



Boston Monthly Temperatures 1900-2000



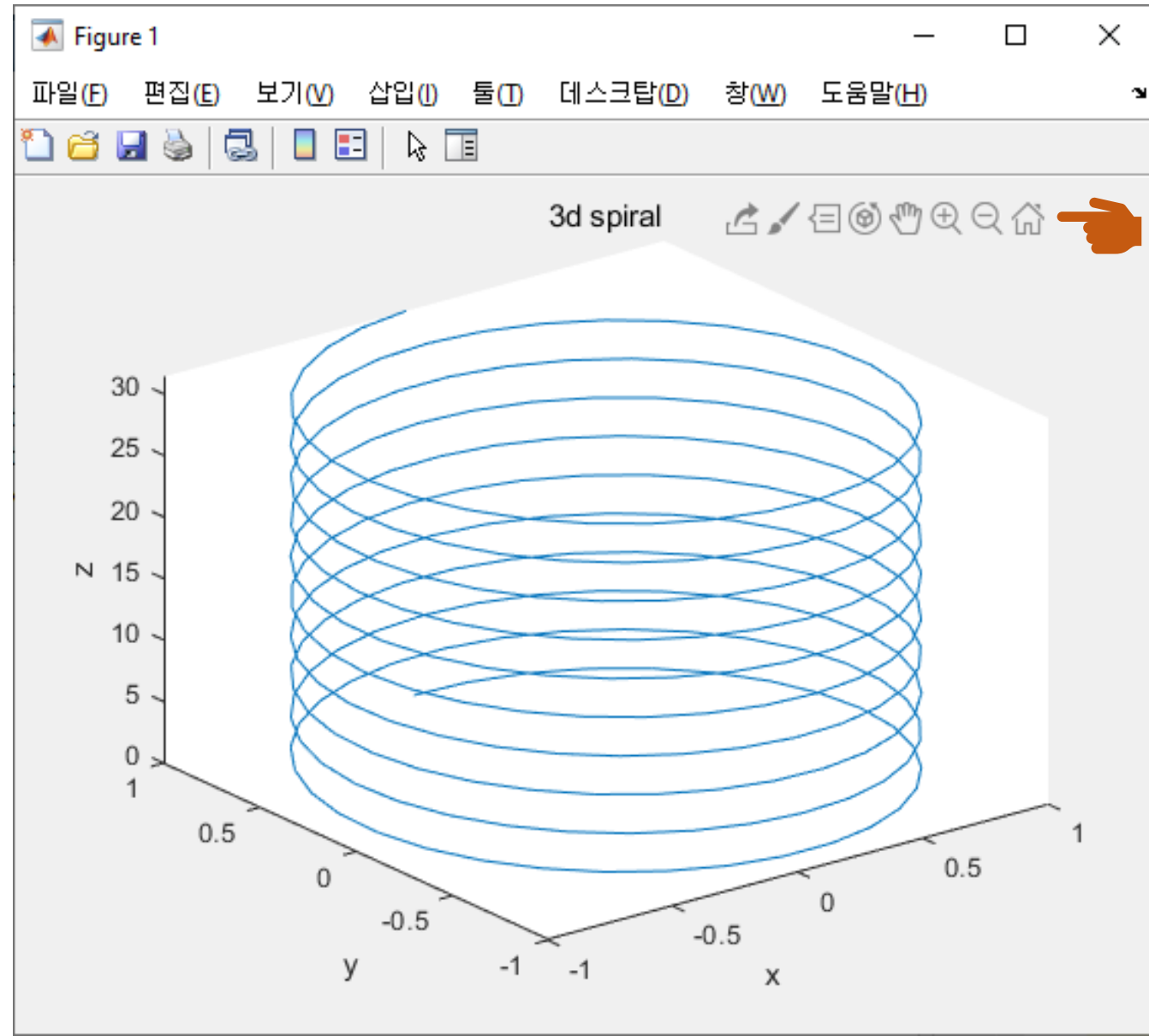
<https://bit.ly/3mdJrzY>

plot이 2차원? 3차원은 plot3!

* 조작패널을 간단히 이용해보자.

```
t = 0:0.1:10*pi;  
  
x = sin(2*t);  
y = cos(2*t);  
z = t;  
  
figure,  
plot3(x, y, z)  
xlabel('x'), ylabel('y'), zlabel('z')  
title('3d spiral')
```

- x, y, z 값을 확인해보자.
 - workspace에서 확인
 - 그래프에서 점을 찍어 확인
 - 3차원 그래프는 각 점을 잇는 선분의 모음

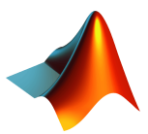


plot3도 문법은 plot과 똑같다.

※ 도움말을 보자.

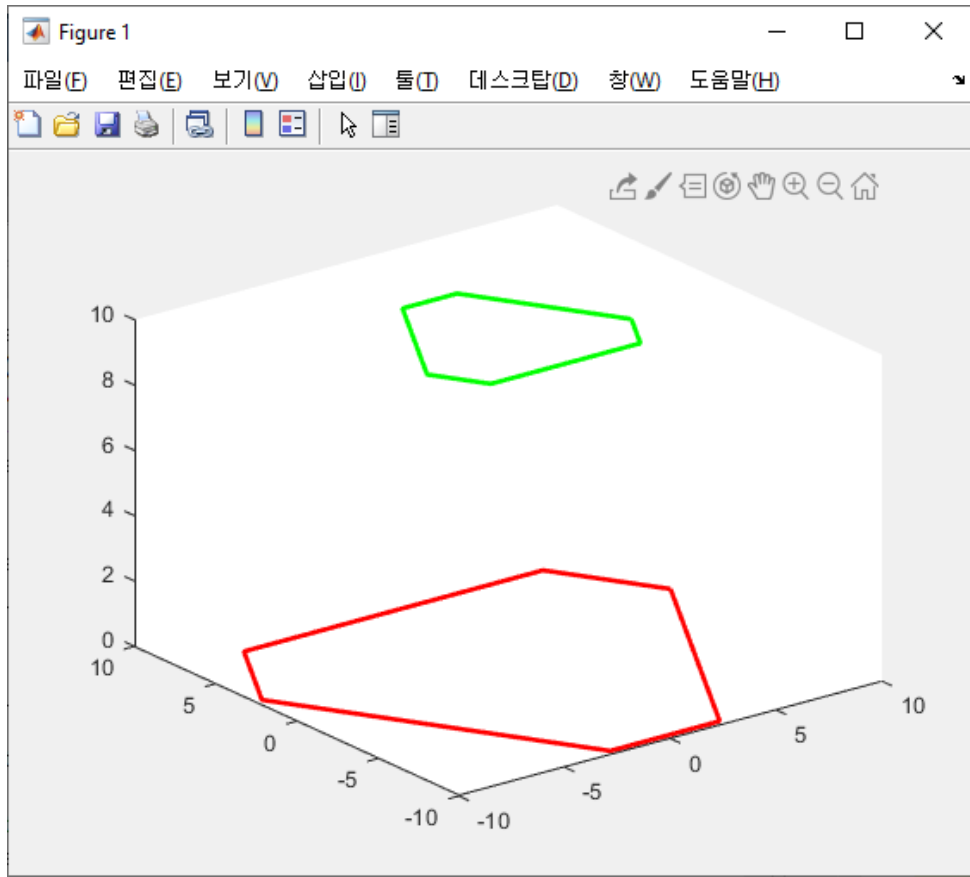
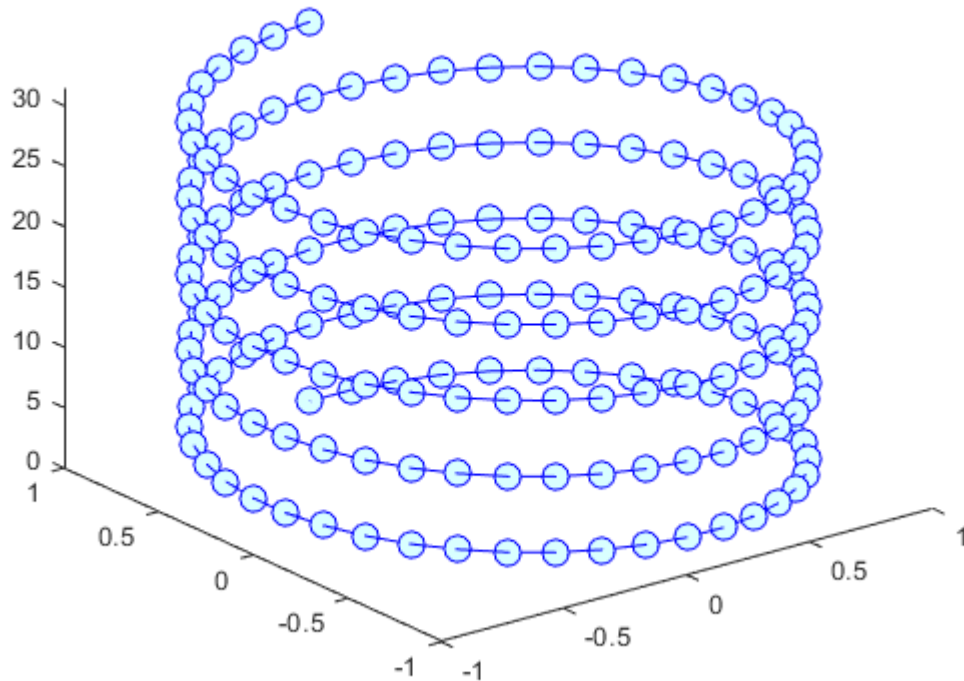
```
t = 0:0.1:10*pi;  
plot3(sin(2*t), cos(2*t), t)  
plot(x, y, 'LineStyle', 'propertyName', propertyValue)
```

	propertyName	의미	값	line specifier?	default
마커	marker	마커 종류	+o*.x^vsdph<>	Y	none
	markeredgecolor	마커 테두리 색	rgbmcykw 또는 1x3 벡터	N	auto
	markerfacecolor	마커 면 색		N	none
	markersize	마커 크기	양의 실수	N	6
선	color	선 색	rgbmcykw 또는 1x3 벡터	Y	[0 0.4470 0.7410]
	linestyle	선 스타일	- , -- , : , -.	Y	solid (-)
	linewidth	선 두께	양의 실수	N	0.5



plot3도 문법은 plot과 똑같다.

```
t = 0:pi/20:10*pi;  
xt = sin(t);  
yt = cos(t);  
plot3(xt,yt,t, '-o', 'Color','b', 'MarkerSize',10, 'MarkerFaceColor', '#D9FFFF')
```



이번엔 면(surface)을 그려보자.

- 변수가 2개인 그래프: $z = f(x, y)$

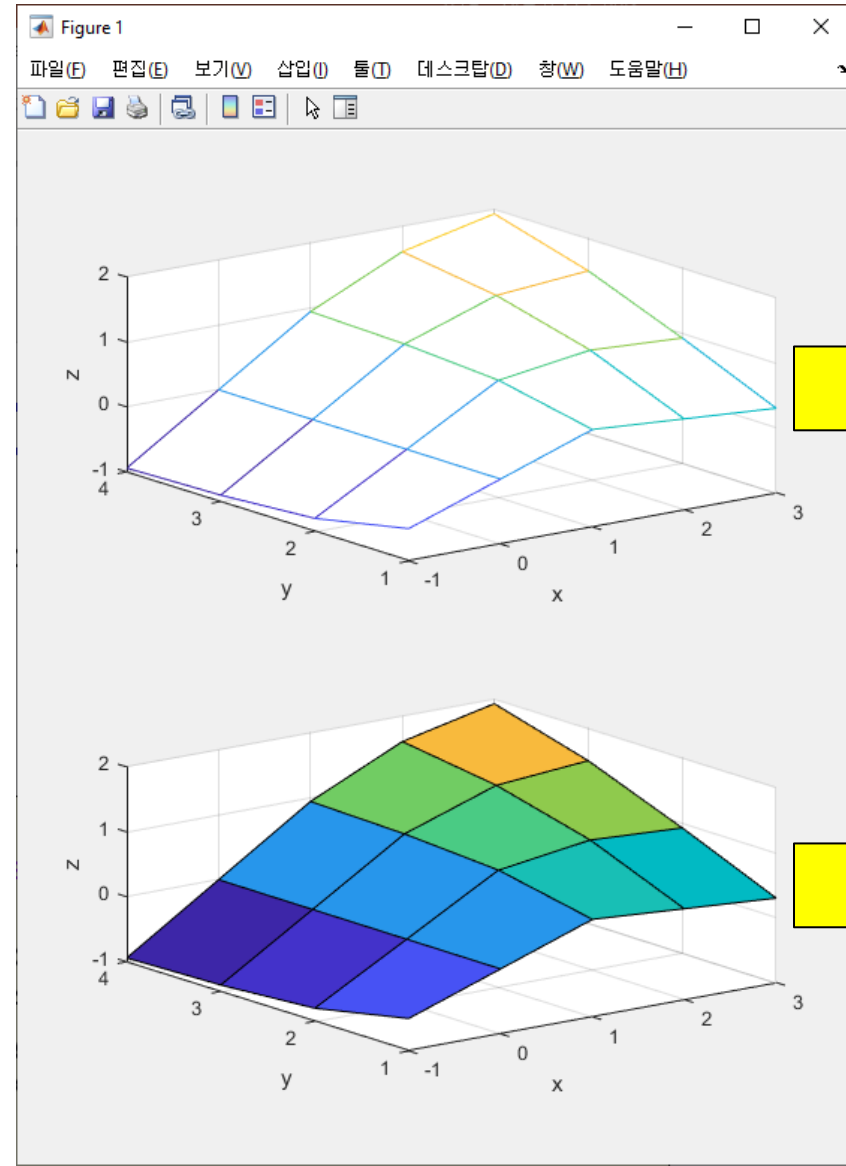
ex.) $Z = \frac{xy^2}{x^2+y^2}$

```
clear
close all

x = -1:3;
y = 1:4;
[xx, yy] = meshgrid(x, y);
zz = xx.*yy.^2./(xx.^2+yy.^2);

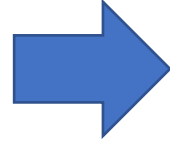
figure,
subplot(2,1,1), mesh(xx, yy, zz)
xlabel('x'); ylabel('y'); zlabel('z')

subplot(2,1,2), surf(xx, yy, zz)
xlabel('x'); ylabel('y'); zlabel('z')
```

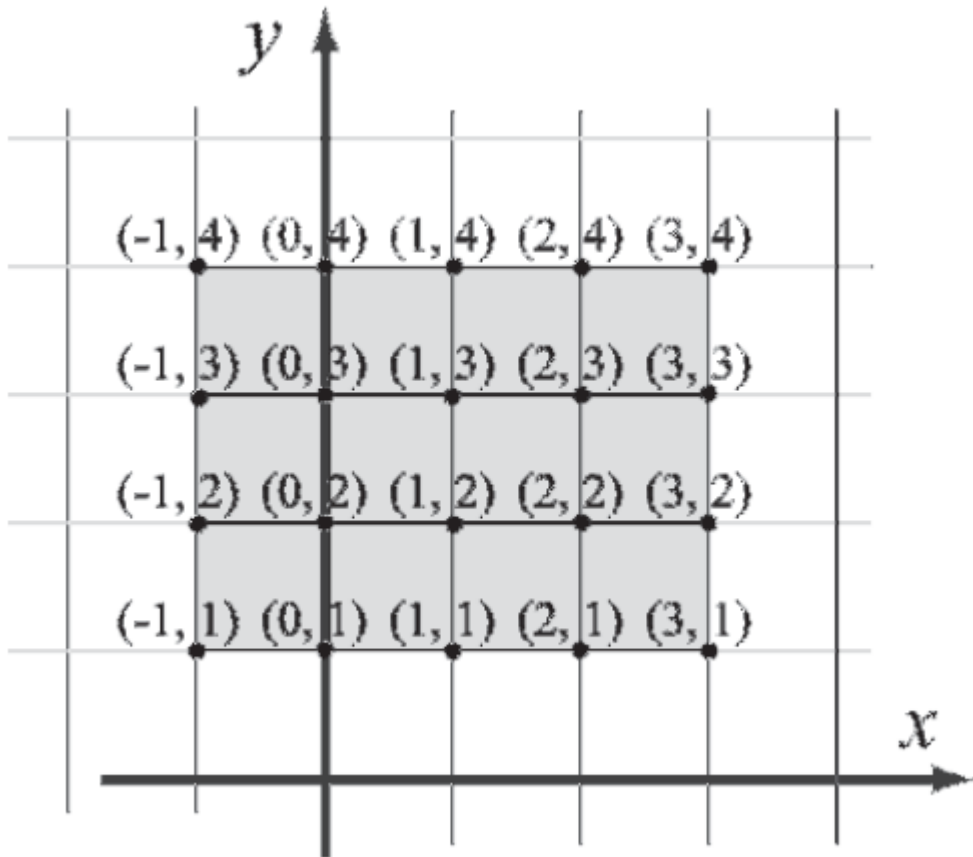


meshgrid

```
x = -1:3;  
y = 1:4;  
[xx, yy] = meshgrid(x, y);
```



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

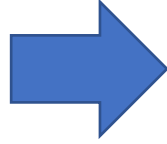


$$X = \begin{bmatrix} -1 & 0 & 1 & 2 & 3 \\ -1 & 0 & 1 & 2 & 3 \\ -1 & 0 & 1 & 2 & 3 \\ -1 & 0 & 1 & 2 & 4 \end{bmatrix}$$

$$\text{and } Y = \begin{bmatrix} 4 & 4 & 4 & 4 & 4 \\ 3 & 3 & 3 & 3 & 3 \\ 2 & 2 & 2 & 2 & 2 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

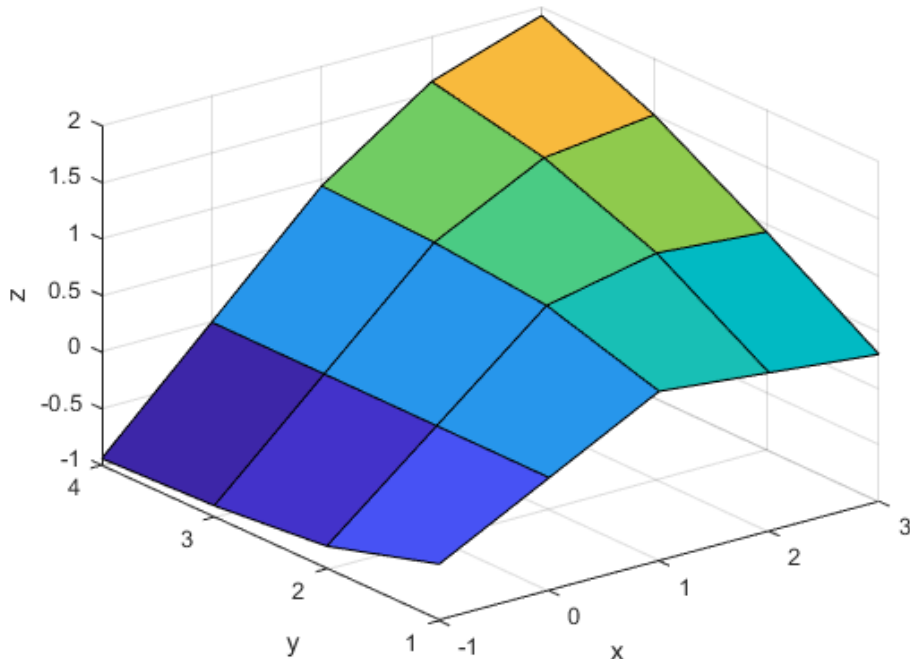
meshgrid → surf

```
x = -1:3;  
y = 1:4;  
[xx, yy] = meshgrid(x, y);  
zz = xx.*yy.^2./(xx.^2+yy.^2);  
  
surf(xx, yy, zz)  
xlabel('x'); ylabel('y'); zlabel('z')
```



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

```
>> zz = xx.*yy.^2./(xx.^2+yy.^2)  
zz =  
    -0.5000     0     0.5000     0.4000     0.3000  
    -0.8000     0     0.8000     1.0000     0.9231  
    -0.9000     0     0.9000     1.3846     1.5000  
    -0.9412     0     0.9412     1.6000     1.9200
```



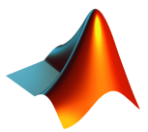
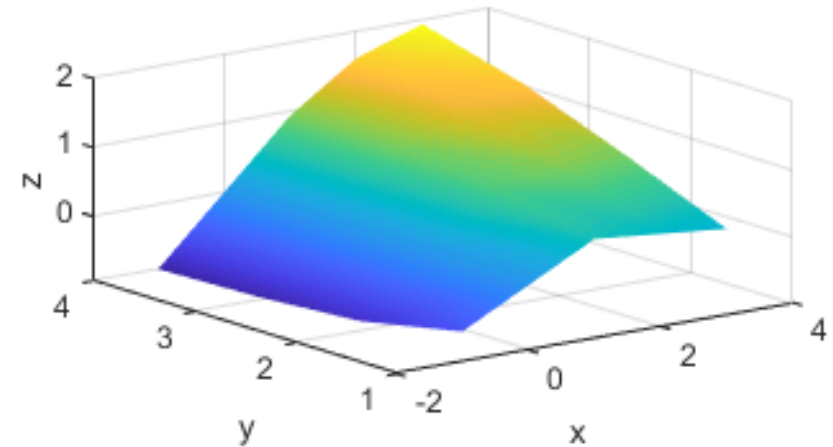
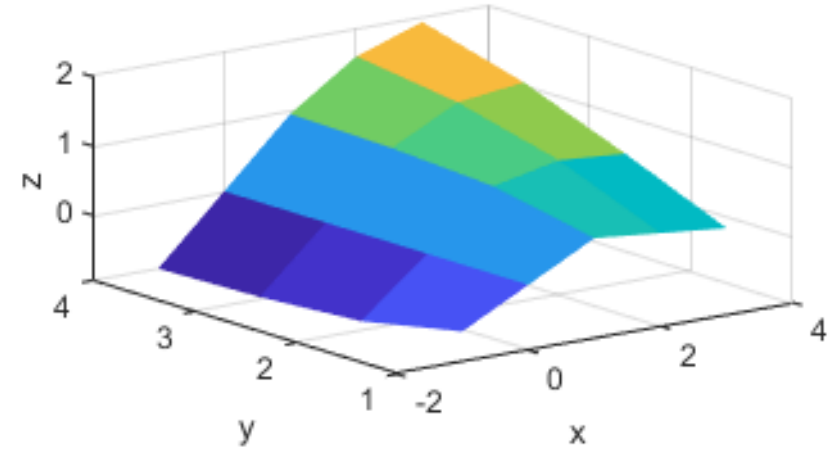
간격을 바꿔보자. + shading 옵션

```
dx = 1;
dy = 1;
x = -1:dx:3;
y = 1:dy:4;
[xx, yy] = meshgrid(x, y);
zz = xx.*yy.^2./(xx.^2+yy.^2);
```

```
figure,
subplot(2,1,1), surf(xx, yy, zz), shading flat
xlabel('x'); ylabel('y'); zlabel('z')
```










```
subplot(2,1,2), surf(xx, yy, zz), shading interp
xlabel('x'); ylabel('y'); zlabel('z')
```

※ 기본값은
shading faceted

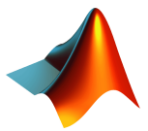
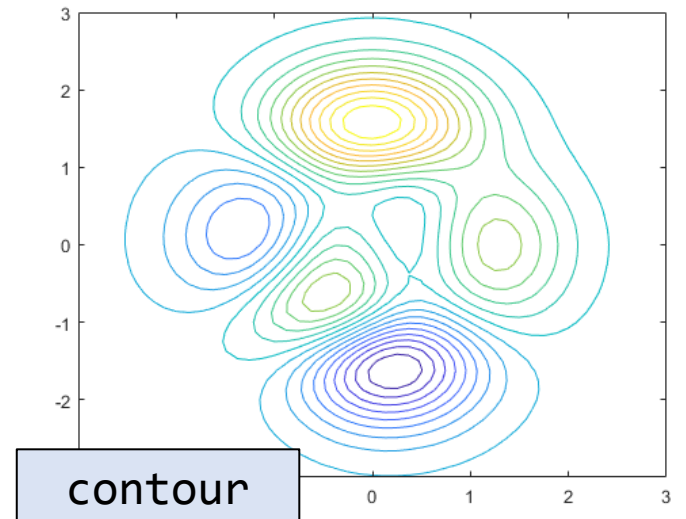
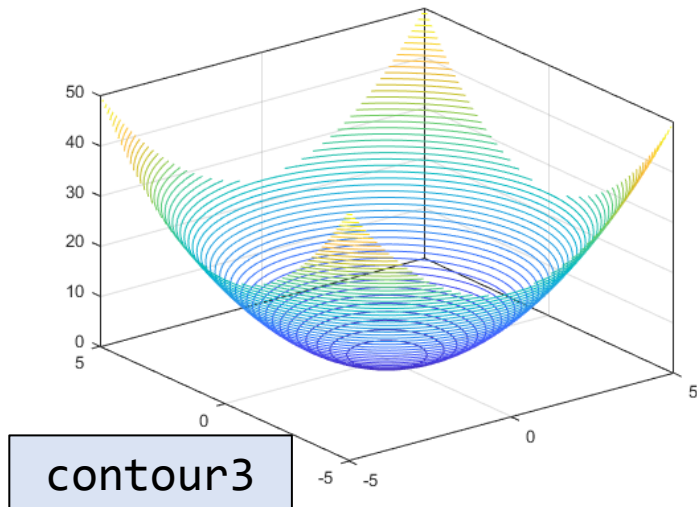
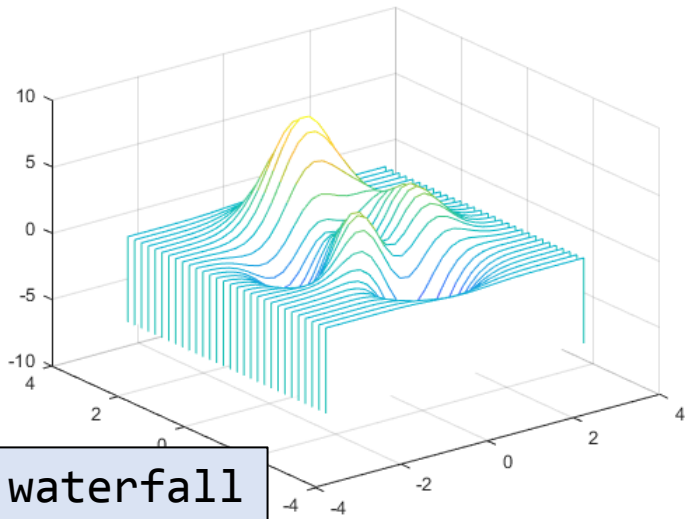
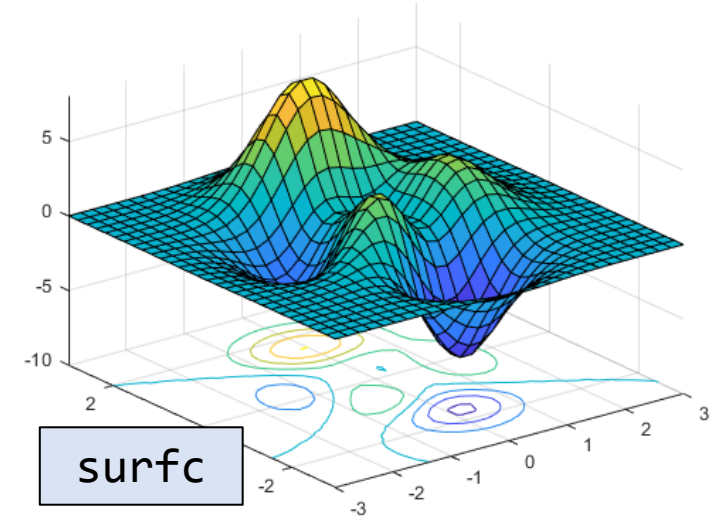
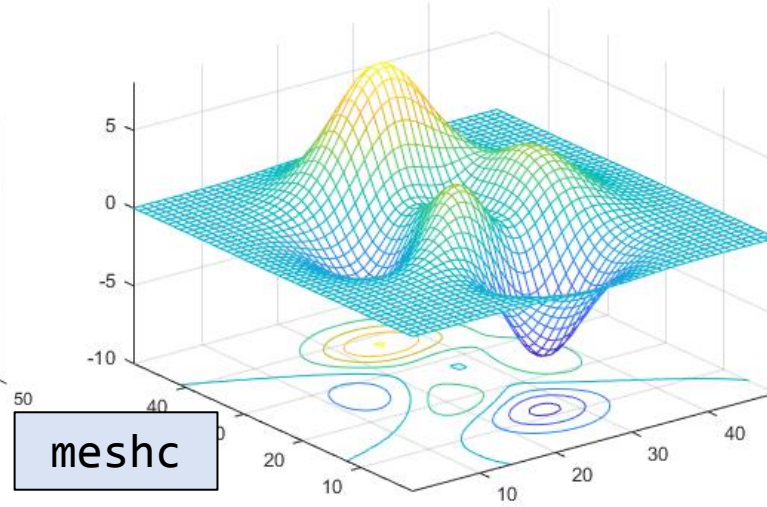
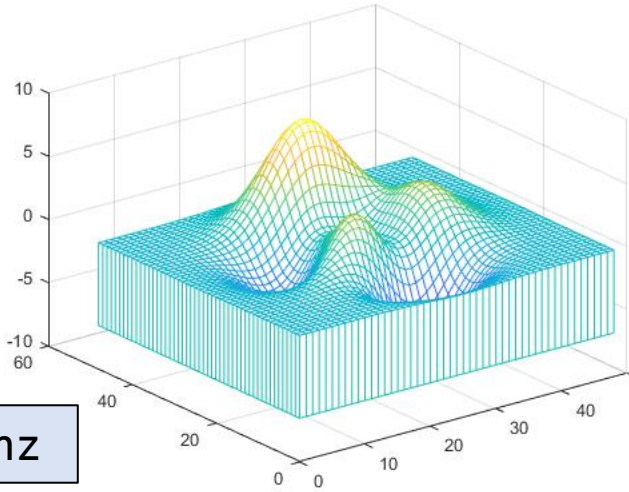


colormap을 바꿔보자. + colorbar를 달아보자.

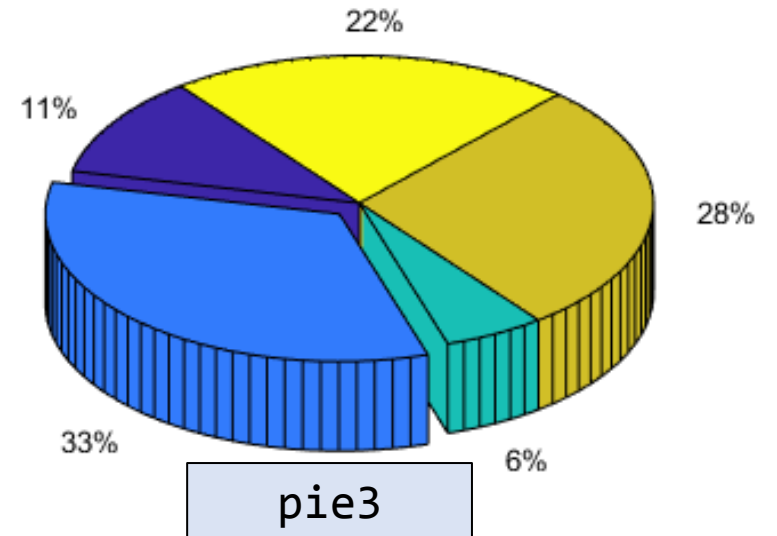
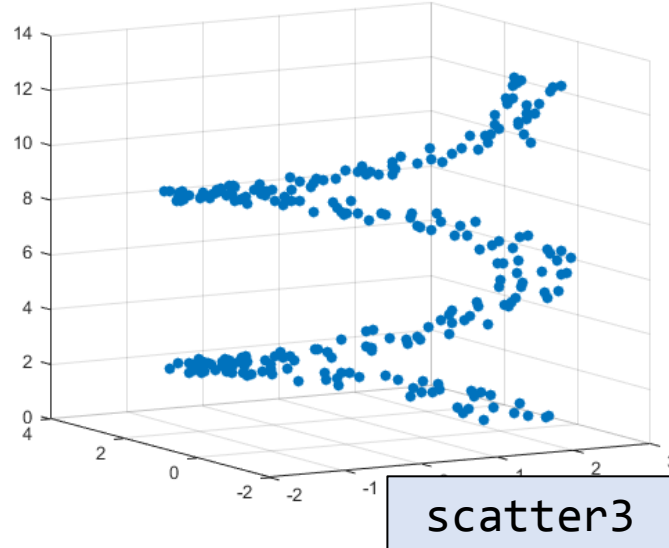
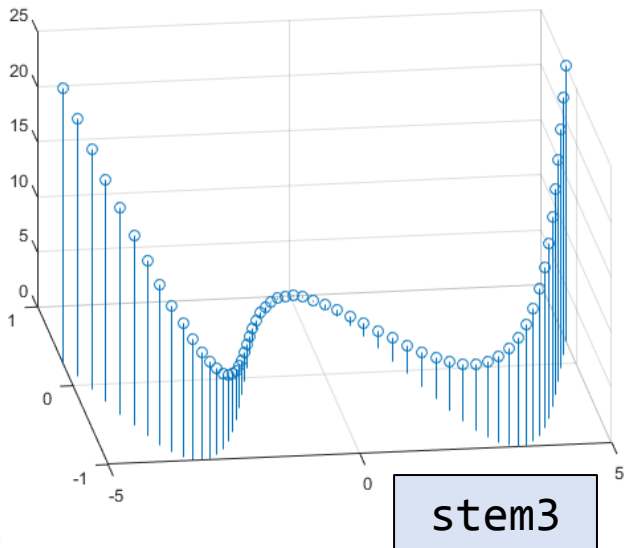
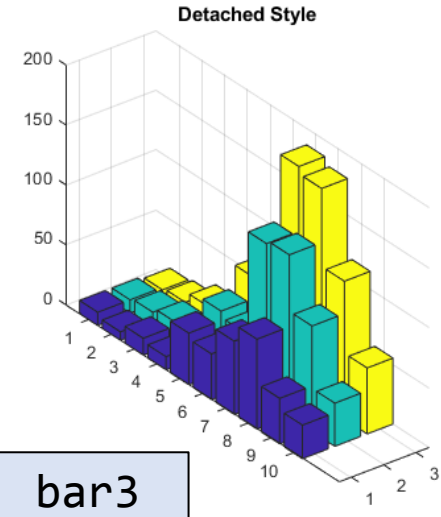
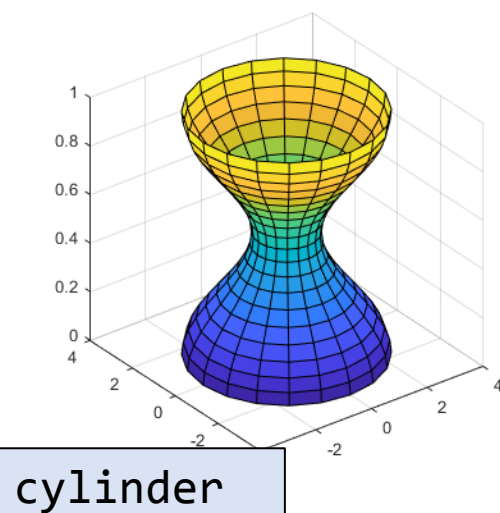
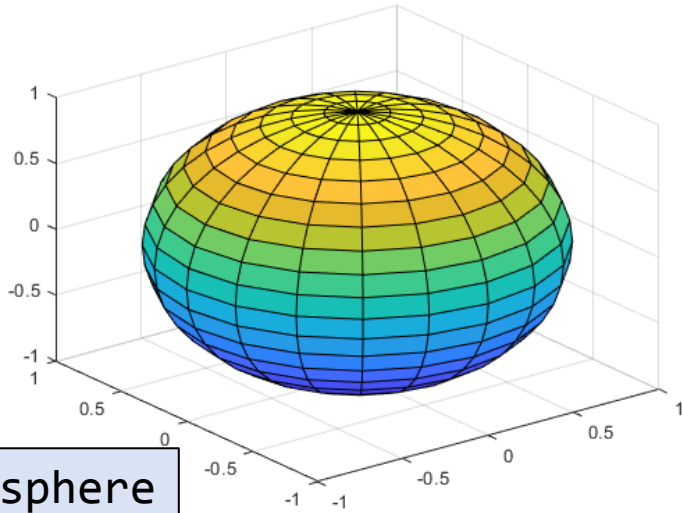
※ 도움말 – colormap

컬러맵 이름	색조
parula	
jet	
hsv	
hot	
cool	
spring	
summer	
autumn	
winter	
gray	
bone	
copper	
pink	
lines	
colorcube	
prism	
flag	
white	

그 외 이상한 3차원 그래프들

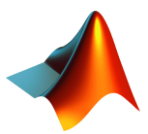
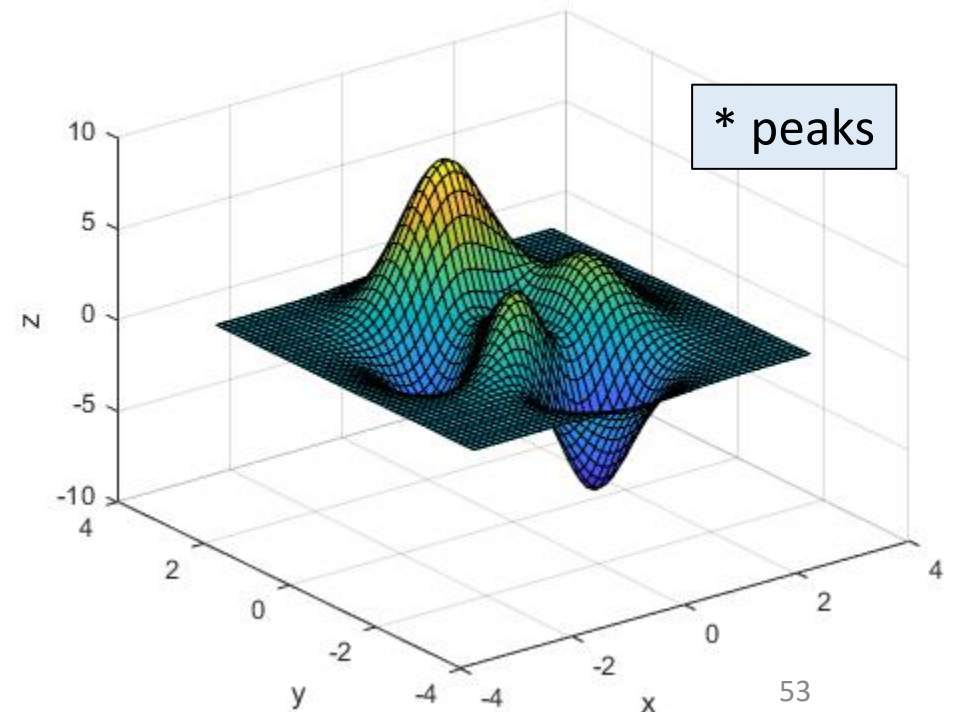
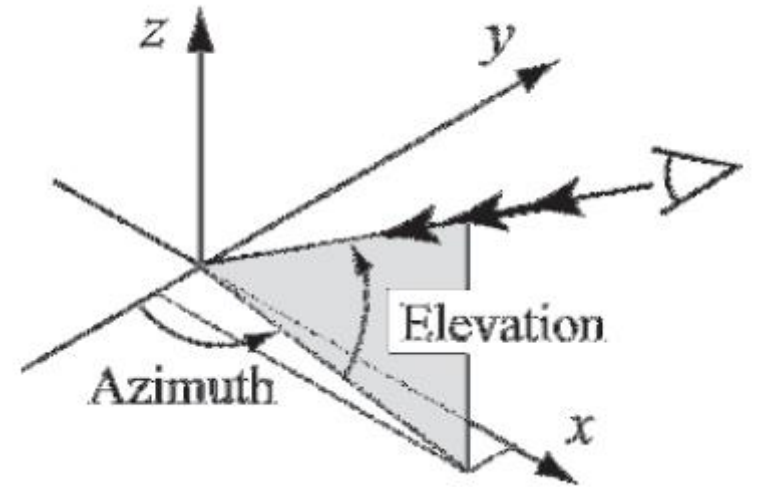


그 외 이상한 3차원 그래프들



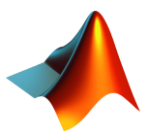
그래프를 돌려보자.

- 조작패널 이용
 - 그냥 회전시킬 때 vs 조작패널로 회전시킬 때
- `view(az, el)`
- `view(2)`: 2차원 보기
 - `view(0, 90)`과 같음
- `view(3)`: 3차원 보기
 - `view(-37.5, 30)`과 같음

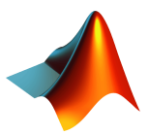
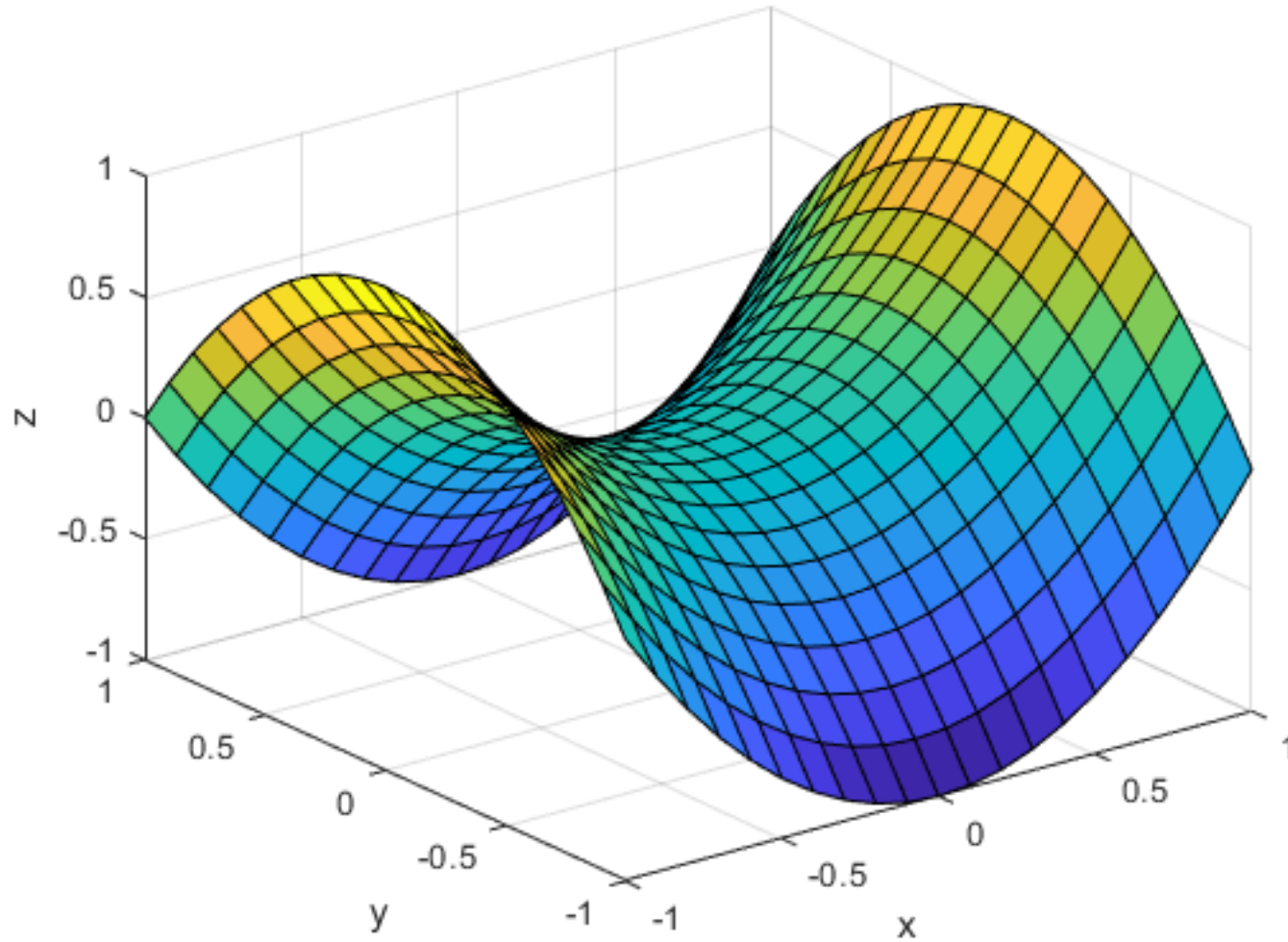


기타

- 탭 자동완성
 - 우리가 배운 것보다 훨씬 많은 옵션들이 있다.
- `clf` = clear figure
- `cla` = clear axes

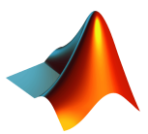
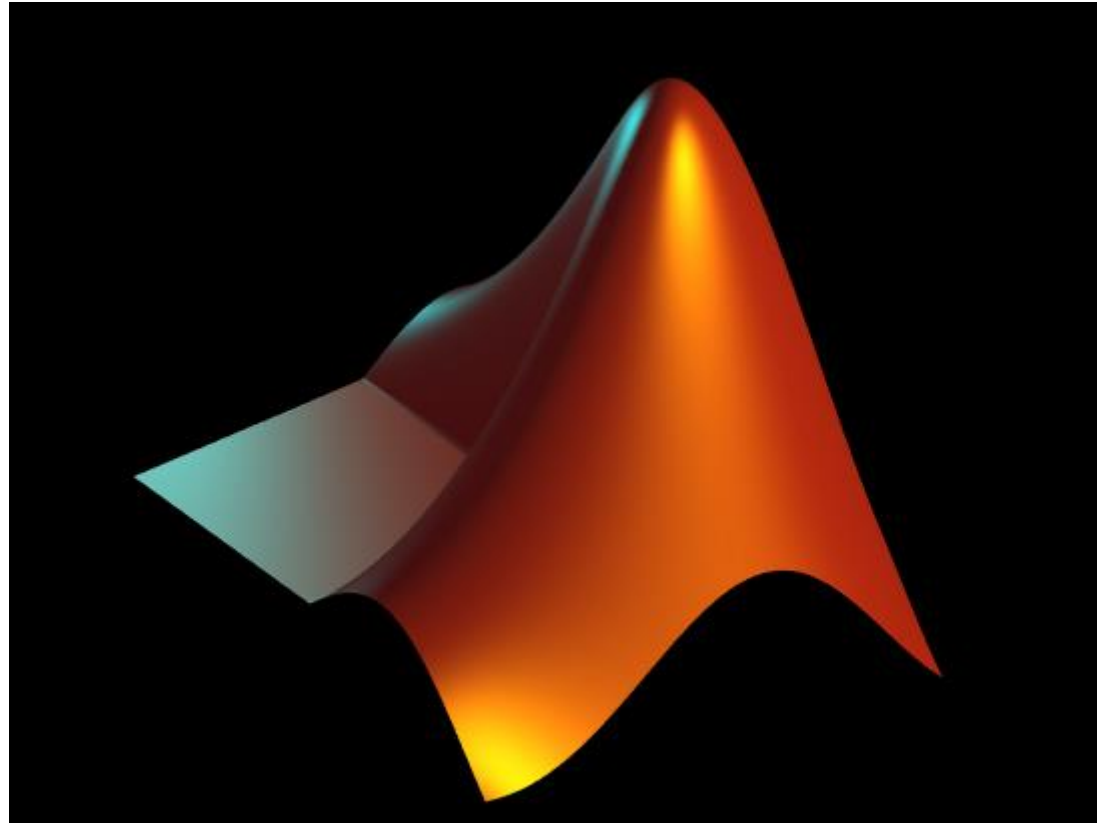


saddle point



매트랩 로고를 그려보자.

- <https://www.mathworks.com/help/matlab/visualize/creating-the-matlab-logo.html>
- 명령창에 아래 입력
 - `web(fullfile(docroot, 'matlab/visualize/creating-the-matlab-logo.html?s_tid=doc_srchttitle'))`



Q&A

