

第五章 二維平面繪圖

本章學習目標

- 學習Matlab的基本繪圖函數plot()
- 編修函數的圖形
- 學習簡單易用的fplot() 與ezplot() 函數
- 學習如何利用Property editor來編修圖形

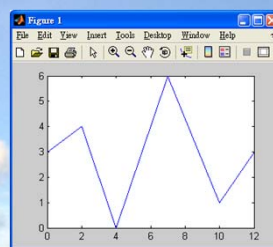
基本的二維繪圖函數

表 5.1.1 plot 函數的使用

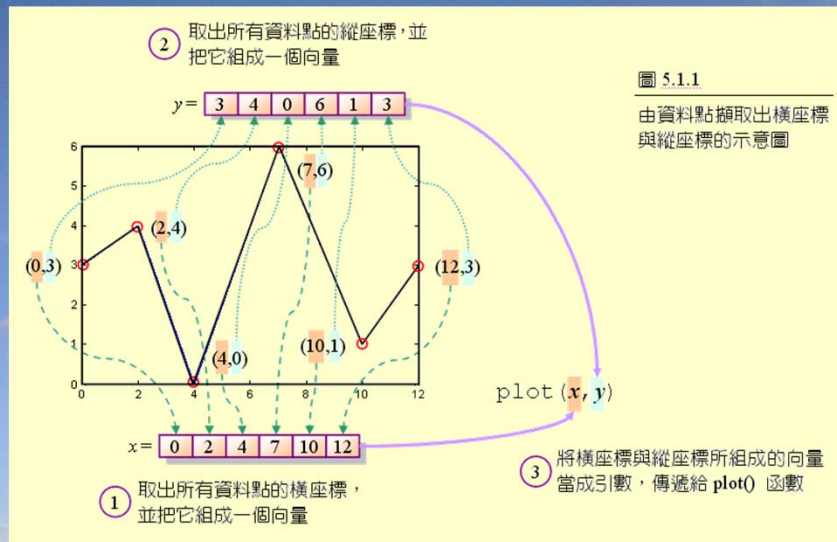
函 數	說 明
plot(x,y)	以 x 為資料點的橫座標所組成的向量， y 為縱座標所組成的向量，描點繪出 (x,y) 的曲線圖
plot(y)	x 的間距為 1，描點繪出 (x,y) 的曲線圖

```
>> x=[0 2 4 7 10 12]
x =
    0     2     4     7    10    12
>> y=[3 4 0 6 1 3]
y =
     3     4     0     6     1     3
```

```
>> plot(x,y)
```



由資料點取出橫座標與縱座標

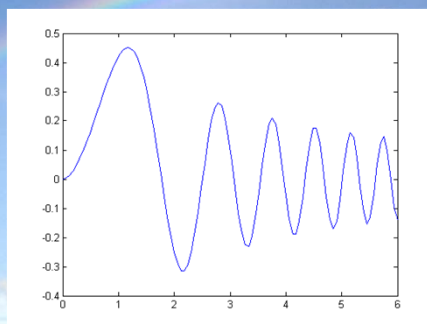


簡單的二維繪圖範例

```
>> x=linspace(0,6,100);
```

```
>> y=sin(x.^2)./(x+1);
```

```
>> plot(x,y)
```



二維圖形的編修

表 5.1.2 修飾 plot 函數所繪出的圖形

函 數	說 明
<code>plot(x, y, 'str')</code>	以字串 <i>str</i> 所指定的格式繪出二維圖形
<code>plot(x₁, y₁, 'str₁', x₂, y₂, 'str₂', ...)</code>	以字串 <i>str₁</i> 所指定的格式繪出 (x ₁ , y ₁) 的圖形，以 <i>str₂</i> 所指定的格式繪出 (x ₂ , y ₂) 的圖形，以此類推
<code>plot(x₁, y₁, 'str', 'p_str', property, ...)</code>	根據繪圖性質 <i>p_str</i> 來繪圖，其中 <i>p_str</i> 可為： <i>LineWidth</i> — 設定線條寬度 <i>MarkerFaceColor</i> — 設定標記的顏色 <i>MarkerEdgeColor</i> — 設定標記的邊框顏色 <i>MarkerSize</i> — 設定標記的大小

控制資料點的顯示符號

表 5.1.3 plot() 函數的控制碼（一），控制資料點的顯示符號

符 號	說 明	符 號	說 明
.	繪出點	^	繪出 △ 符號
*	繪出星號	v	繪出 ▽ 符號
o	繪出小圓 (小寫字母 o)	s 或 square	繪出正方形
+	繪出加號	d 或 diamond	繪出菱形
x	繪出打叉符號 (小寫字母 x)	p 或 pentagram	繪出五角形
<	繪出 ◁ 符號	h 或 hexagram	繪出六角形
>	繪出 ▷ 符號	none	不繪出任何形狀 (預設)

控制線條樣式與顏色

表 5.1.4 plot() 函數的控制碼（二），控制線條樣式

線條樣式	說明	線條樣式	說明
- (減號)	實線 (預設)	:	由點連成的線段
--	虛線	none	不繪出線段
-.	虛線和點連成的線段		

表 5.1.5 plot() 函數的控制碼（三），控制線條顏色

線條顏色	說明	線條顏色	說明
g	綠色 (green)	w	白色 (white)
m	洋紅色 (magenta)	r	紅色 (red)
b	藍色 (blue) (預設)	k	黑色 (black)
c	青藍色 (cyan)	y	黃色 (yellow)

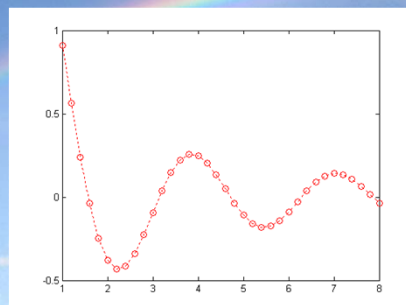
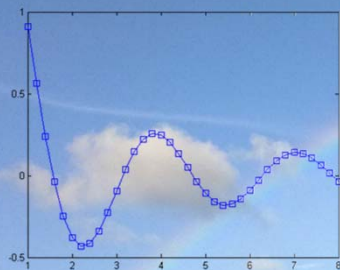
二維繪圖範例 (1/2)

```
>> x=linspace(1,8,36);
```

```
>> y1=sin(2*x)./x;
```

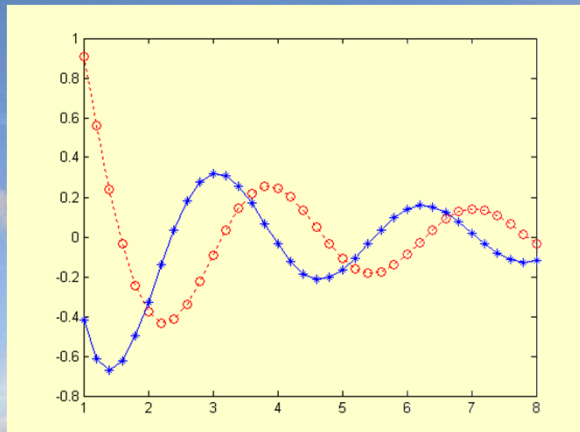
```
>> plot(x,y1,'-sb')
```

```
>> plot(x,y1,':ro')
```



二維繪圖範例 (2/2)

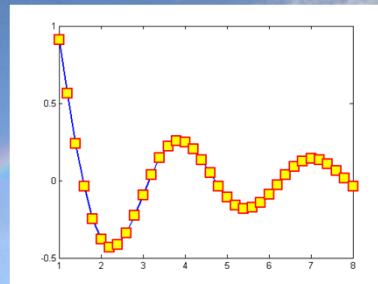
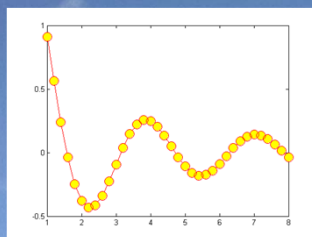
```
>> y2 = cos(2*x)./x;  
>> plot(x,y1,':ro',x,y2,'-b*')
```



9

二維繪圖範例 (2/2)

```
>> plot(x,y1,'-ro','MarkerSize',  
12,'MarkerFaceColor','y')
```



```
>> plot(x,y1,'-bs','LineWidth',2,  
'MarkerSize',12,'MarkerEdgeColor',  
'r','MarkerFaceColor','y')
```

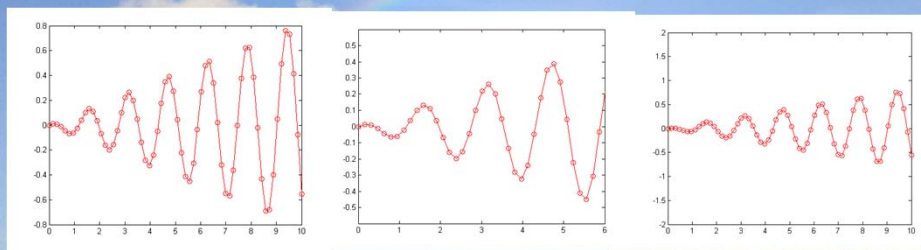
10

更改繪圖的範圍與顯示方式

表 5.2.1 設定繪圖的範圍

函數	說明
<code>axis([xmin,xmax,ymin,ymax])</code>	指定繪圖的範圍， x 方向從 $xmin$ 到 $xmax$ ， y 方向從 $ymin$ 到 $ymax$

```
>> x=linspace(0,10,64);
>> y=x.*cos(4*x)./12;
>> plot(x,y,'-ro')    >> axis([0,6,-0.6,0.6])    >> axis([-inf,inf,-2,2])
```

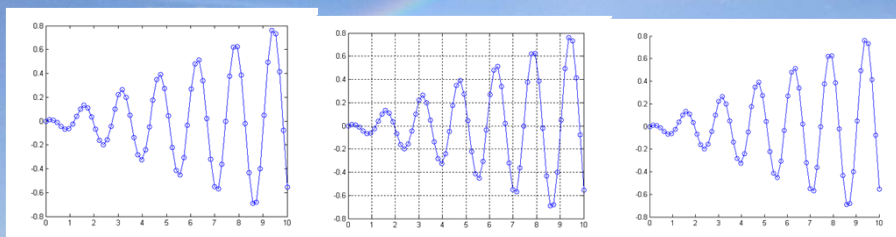


設定是否顯示圖形的格線與外框

表 5.2.2 設定是否顯示圖形的格線與外框

指令	說明
<code>grid</code>	設定是否顯示格線，設定 <code>on</code> 為顯示，設定 <code>off</code> 則不顯示
<code>box</code>	設定是否顯示圖形的外框，設定 <code>on</code> 顯示，設定 <code>off</code> 不顯示

```
>> plot(x,y,'-bo')    >> grid on    >> grid off; box off
```



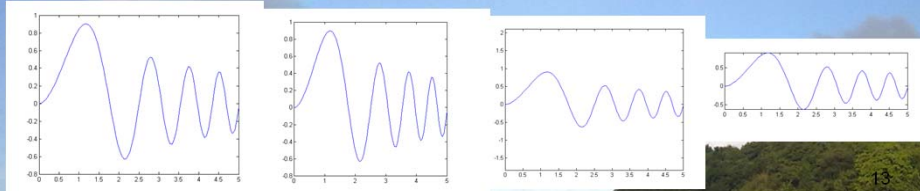
修改x與y軸的顯示比例

表 5.2.3 設定座標軸顯示的比例

指令	說明
<code>axis normal</code>	使用 Matlab 預設的寬高比，且拉動視窗即可調整其比例
<code>axis square</code>	圖形輸出外框的寬與高比例為 1:1
<code>axis equal</code>	圖形座標軸的比例為 1:1
<code>axis tight</code>	圖形緊貼繪圖區域的外框（tight 是緊縮之意）

```
>> x=linspace(0,5,100);
>> y=2*sin(x.^2)./(x+1);
```

```
>> plot(x,y)    >> axis square    >> axis equal    >> axis equal tight
```



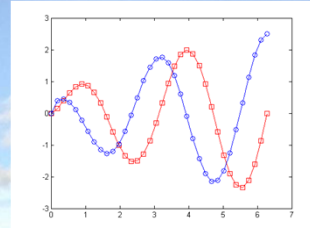
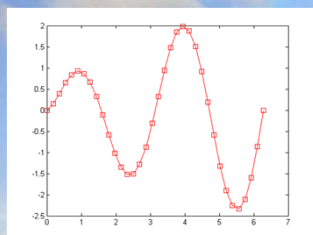
13

於已存在的圖中加入新圖

表 5.2.4 設定圖形產生的方式

指令	說明
<code>hold</code>	設定 hold 為 on 時，則新產生的圖形會疊加在原有圖形的上面，若是設定 off，則原有的圖形會被新產生的圖形覆蓋掉。

```
>> x=linspace(0,2*pi,36);
>> y1=sqrt(x).*sin(2*x);
>> y2=sqrt(x).*cos(2*x);
>> plot(x,y1,'-rs'); hold on    >> plot(x,y2,'-bo'); hold off
```



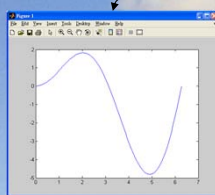
14

建立新的繪圖視窗

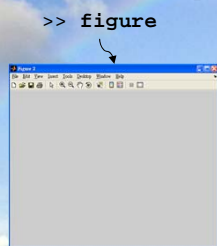
表 5.2.5 設定圖形產生的方式

函數	說明
figure	建立一個新的繪圖視窗，視窗的標題為 Matlab 自動設定
figure(n)	建立一個新的繪圖視窗，視窗的標題為 Figure n。若 Figure n 為已經存在的視窗，則 figure(n) 會把此視窗變成作用中視窗

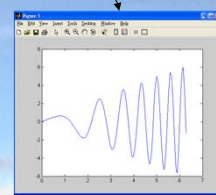
```
>> x=linspace(0,2*pi,100);
>> plot(x,x.*sin(x))
```



```
>> plot(x,x.*cos(x))
```



```
>> figure(5);
>> plot(x,x.*cos(x.^2))
```



15

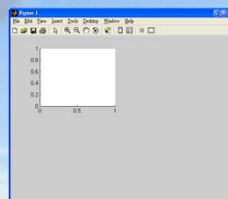
將數張圖合併成一張大圖(1/2)

表 5.2.6 subplot() 函數的用法

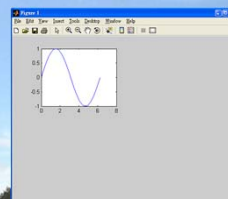
函數	說明
subplot(m,n,p)	把繪圖視窗分成 $m \times n$ 個區域，並在第 p 個位置建立一個子繪圖區。位置 p 的計算方式是由左而右，由上而下排列
subplot(m,n,p,'replace')	於第 p 個位置建立一個子繪圖區，若此繪圖區內已有其它圖形存在，則新繪的圖會取代掉原有的圖

```
>> x=linspace(0,2*pi,50);
```

```
>> subplot(2,2,1)
```



```
>> plot(x,sin(x))
```



16

將數張圖合併成一張大圖(2/2)

```
>> subplot(2,2,2);plot(x,cos(x));  
>> grid on
```

```
>> subplot(2,2,3);plot(x,sqrt(x))
```

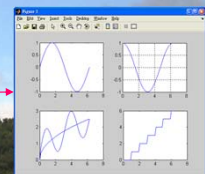
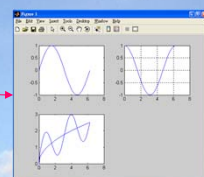
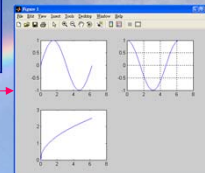
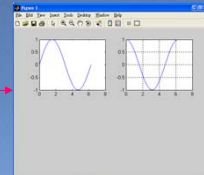
```
>> hold on
```

```
>> plot(x,sqrt(x)+sin(2*x))
```

```
>> hold off
```

```
>> subplot(2,2,4);plot(x,floor(x))
```

```
>> close all
```



```
t = 0:0.1:2*pi;
```

```
x = 3*cos(t);
```

```
y = sin(t);
```

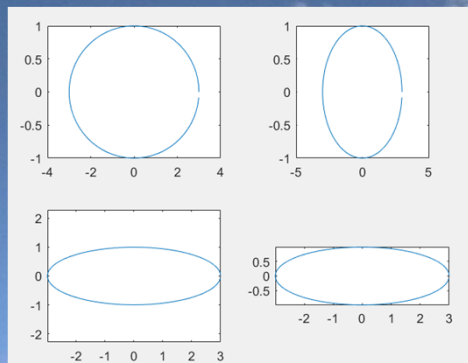
```
subplot(2, 2, 1); plot(x, y); axis normal
```

```
subplot(2, 2, 2); plot(x, y); axis square
```

```
subplot(2, 2, 3); plot(x, y); axis equal
```

```
subplot(2, 2, 4); plot(x, y); axis equal tight
```

```
saveas(gcf,'ML_plot.jpg','jpg')
```

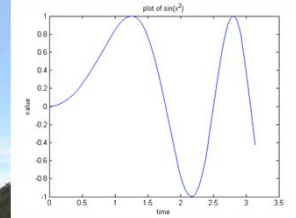
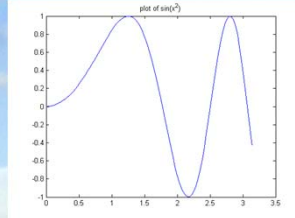
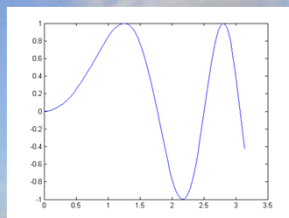


於圖形內加入文字

表 5.3.1 於圖形內加入文字

函數	說明
<code>title('text')</code>	設定圖形的標題文字為 <i>text</i>
<code>xlabel('text')</code>	設定 <i>x</i> 軸的解說文字為 <i>text</i>
<code>ylabel('text')</code>	設定 <i>y</i> 軸的解說文字為 <i>text</i>
<code>zlabel('text')</code>	設定 <i>z</i> 軸的解說文字為 <i>text</i> (用於三維的繪圖)

```
>> x=linspace(0,pi,100); >> title('plot of sin(x^2)') >> xlabel('time');
>> plot(x,sin(x.^2)) >> ylabel('value');
```

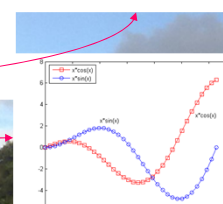
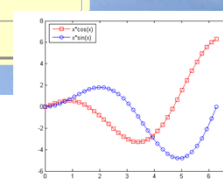
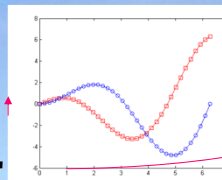


加入圖形的註解

表 5.3.2 加入圖形的註解

函數	說明
<code>legend(str₁,str₂,...)</code>	加入曲線說明的文字
<code>legend(str₁,str₂,...,pos)</code>	設定曲線說明文字的位置，pos 設 1 代表將說明文字放在右上角，2 是左上，3 是左下，4 則是放在右下角
<code>legend off</code>	清除曲線說明文字
<code>text(x,y,'text')</code>	在圖形中位置為 (x,y) 之處加入註解文字
<code>gtext('text')</code>	利用滑鼠來設定文字輸入的位置

```
>> x=linspace(0,2*pi,36);
>> y1=x.*cos(x);y2=x.*sin(x);
>> plot(x,y1,'-rs',x,y2,'-bo')
>> legend('x*cos(x)','x*sin(x)')
>> text(2,2.5,'x*sin(x)')
>> text(5.5,3,'x*cos(x)')
>> gtext('Text','color','r')
```

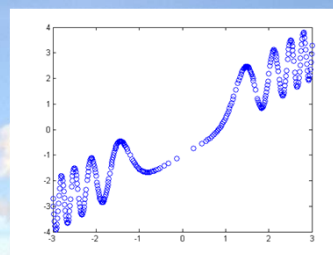
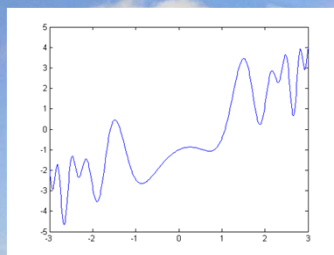


更簡潔的繪圖函數

表 5.4.1 繪圖函數 `fplot()` 的用法

函 數	說 明
<code>fplot('f_str', [xmin, xmax])</code>	繪出函數 f_str 的圖形， x 軸的範圍取 $xmin$ 到 $xmax$
<code>fplot('f_str', [xmin, xmax, ymin, ymax])</code>	繪出函數 f_str 的圖形， x 軸的範圍取 $xmin$ 到 $xmax$ ， y 軸的範圍取 $ymin$ 到 $ymax$

```
>> fplot('x-cos(x^3)-sin(2*x^2)', [-3, 3]) >> fplot('x-cos(x^3)', [-3, 3], 'o')
```



21

繪圖函數 `ezplot()`

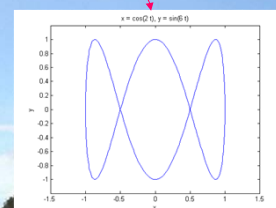
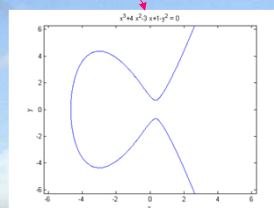
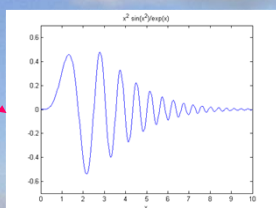
表 5.4.2 繪圖函數 `ezplot()` 的用法

函 數	說 明
<code>ezplot('f_str', [xmin, xmax])</code>	繪出函數 f_str 的圖形，繪圖範圍在 x 與 y 方向均取 $xmin$ 到 $xmax$
<code>ezplot('f_str', [xmin, xmax, ymin, ymax])</code>	繪出函數 f_str 的圖形，繪圖範圍在 x 方向取 $xmin$ 到 $xmax$ 在 y 方向均取 $ymin$ 到 $ymax$
<code>ezplot('fx', 'fy', [tmin, tmax])</code>	參數繪圖，繪出 $(fx(t), fy(t))$ ， t 從 $tmin$ 到 $tmax$ 的參數圖

```
>> ezplot('x^2*sin(x^2)/exp(x)', ...  
[0, 10, -0.7, 0.7])
```

```
>  
ezplot('cos(2*t)', 'sin(6*t)', [0, pi]), ...  
axis([-1.5, 1.5, -1.2, 1.2])
```

```
>> ezplot('x^3+4*x^2-3*x+1-y^2')
```



2

利用「屬性編輯區」編修圖形

繪圖工具列

關閉「屬性編輯區」

開啟「屬性編輯區」

點選曲線，於屬性編輯區中會出現相對應的屬性可供修改

屬性編輯區

圖 5.5.1
利用「屬性編輯區」編修圖形的一些屬性

23

利用「屬性編輯區」
編修圖形的一些屬性

點選曲線，於屬性編輯區中會出現相對應的屬性可供修改

屬性編輯區

23

「Data Cursor」按鈕

- 繪圖工具列上的按鈕可對圖形進行旋轉、縮放，或者是進行其它的編修

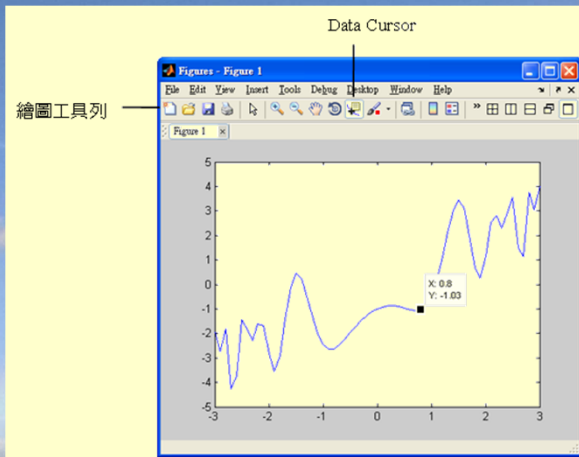


圖 5.5.2

繪圖工具列可協助執行特定的功能。左圖是利用「Data Cursor」按鈕來顯示函數曲線的座標

- 繪圖工具列上的按鈕可對圖形進行旋轉、縮放，或者是進行其它的編修

繪圖工具列可協助執行特定的功能。左圖是利用「Data Cursor」按鈕來顯示函數曲線的座標