

# 用Python學數學

## 本章大綱

- 7-1 內建數學公式
- 7-2 數學函數繪圖
- 7-3 實例探討

Python 內建很多數學工具，通通寫在 `numpy` 模組，本章將介紹這些功能，這樣可以幫助理解或解決國中、高中的數學問題。

## 7-1 內建數學公式

### 一元多次方程式

解一元多次方程式是使用 `roots()` 方法，可解一元一次、一元二次、一元三次等方程式。例如，若有一元一次方程式如下：

$$x+2=0$$

則求其解的程式如下：

```
import numpy as a
b=a.roots([1,2])#x+2=0
print(b)#-2
```

又例如，一元二次方程式如下：

$$x^2-4x-12=0$$

求其解的程式如下：

```
import numpy as a
b=a.roots([1,-4,-12])#x**2-4x-12=0
print(b)#6,-2
```

又例如，一元三次方程式如下：

$$x^3-3x^2-16x-12=0$$

求其解的程式如下：

```
import numpy as a
b=a.roots([1,-3,-16,-12])#x**3-3x**2-16x-12=0
print(b)#6,-2,-1
```

## 二元一次或三元一次

二元一次或三元一次要使用 `linalg` 類別的 `solve` 方法。例如，若有二元一次方程式如下：

$$3x+y=5$$

$$x-2y=-3$$

則求其解的 Python 程式如下：

```
import numpy as a
c=a.array([[3,1],[1,-2]])
d=a.array([5,-3])
ans=a.linalg.solve(c,d)
print(ans)#1,2
```

又例如，三元一次方程式如下：

$$x+y-z=-2$$

$$x+z=2$$

$$x-y+2z=5$$

則求其解的 Python 程式如下：

```
import numpy as a
c=a.array([[1,1,-1],[1,0,1],[1,-1,2]])
d=a.array([-2,2,5])
ans=a.linalg.solve(c,d)
print(ans)#1,-2,1
```

### 向量內積（高中數學）

若有兩個向量如下：

```
m=(a1,a2,a3)
n=(b1,b2,b3)
```

向量內積公式是

```
m*n=(a1*b1+a2*b2+a3*b3)
```

運算結果是純量。例如，若有向量

```
m=(1,1,-1)
n=(2,0,1)
```

則其內積程式如下：

```
import numpy as a
m=a.array([1,1,-1])
n=a.array([2,0,1])
d=a.inner(m,n)
print(d)#1
```

## 向量外積（高中數學）

向量外積也是高中空間向量常用計算，過程有點繁瑣，把程式寫好，那隨時可代入求答案。

```
m=(a1,a2,a3)
n=(b1,b2,b3)
```

則向量外積  $m*n=(a2*b3-a3*b2,a3*b1-a1*b3,a1*b2-a2*b1)$ 。例如：

```
m=(1,1,-1)
n=(2,0,1)
```

則求解所需程式如下：

```
import numpy as a
m=a.array([1,1,-1])
n=a.array([2,0,1])
d=a.cross(m,n)
print(d)#1 -3 -2
```

## 集合的運算

Python 的集合表示如下：

```
a={1,2,3}
b={2,3,5,6}
```

若要求其交集、聯集、差集，則其程式如下：

```
a={1,2,3}#表示法竟然和集合的列舉法相同
b={2,3,4,5}
c={1,2,3,4}
print(a&b)#交集 2,3
print(a|b)#聯集1,2,3,4,5
print(a-b)#差集1
print(a^b)#對稱差集1,4,5
print(1 in a)#True
print(a==b)#False
```

```
print(5 not in b)#False
print(a < c)#包含於 True
print(a < b)#False
```

## 複數的運算（高中數學）

Python 的資料型態有複數，且運算子『+、-、\*、/、abs』也適用複數的加減乘除的運算。請鍵入以下程式，並觀察執行結果。

```
a=4+1j#1不能省
b=2-1j
print(a+b)
print(a-b)
print(a*b)
print(a/b)
print(abs(a))
```

## 7-2 數學函數繪圖

國高中數學有很多數學函數，本節就使用 `matplotlib.pyplot` 繪圖類別來繪製這些函數圖形，尤其是高一數學的指數、對數、三角函數，您只要多畫幾次，那就會更瞭解其內涵。其次，`matplotlib.pyplot` 繪圖類別還要使用 `numpy` 模組，來存放所需要的連續數據。（補充說明：若使用 `Anaconda3`，那以上模組都是預設模組，通通不用再額外安裝）

### 直線

我們人類繪製直線是使用直尺，但電腦並沒有直尺，那電腦如何繪製直線呢？答案就是要先建立直線方程式，然後再根據直線方程式一點一點密集繪製，這些密集的点，看起來就是直線。例如，設有直線方程式如下：

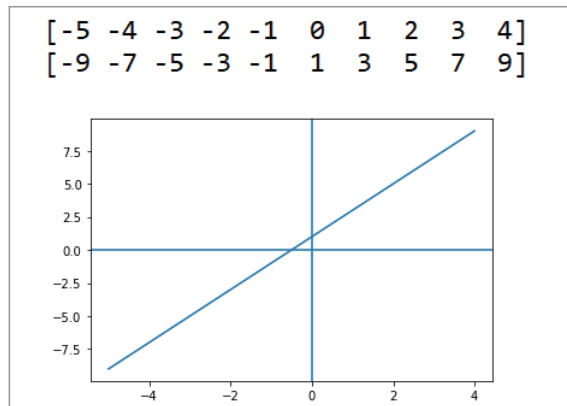
$$y=ax+b$$

那要繪製  $x$  從 -5 到 5 的直線，程式如下：

```
import matplotlib.pyplot as plt
import numpy as np
x = np.arange(-5, 5, 1) #從-5到5，間隔是1，不含結束點5，
且間隔可以實數
print(x)
y = 2*x + 1
print(y)
plt.plot(x,y)
plt.axhline(y=0)#x軸
plt.axvline(x=0)#y軸
plt.show()
```

執行結果如下，`arange()` 方法是產生一個 list（一群數字，不是一個，請看上一章），本例從 -5 到 5，間隔是 1，不含結束點 5，且間隔可以實數。請留意下圖第一、第二列數字是以下程式的輸出結果。

```
x = np.arange(-5, 5, 1) #從-5到5，間隔是1，不含結束點5，且間隔可以實數
print(x)
y = 2*x + 1
print(y)
```



### 自我練習

請同時繪製兩條直線，分別是  $y=x+6$  與  $y=-2x+3$ 。

## 二次曲線

前面  $x$  僅一次方  $y=ax+b$ ，那圖形是直線，若  $x$  是兩次  $y=ax^2+bx+c$ 、或三次  $y=ax^3+bx^2+cx+d$ ，那就形成曲線， $x$  二次會是拋物線，二次方係數為正就凹向上，會有極小值；二次方係數為負就凹向下，會有極大值，請觀察以下圖形。

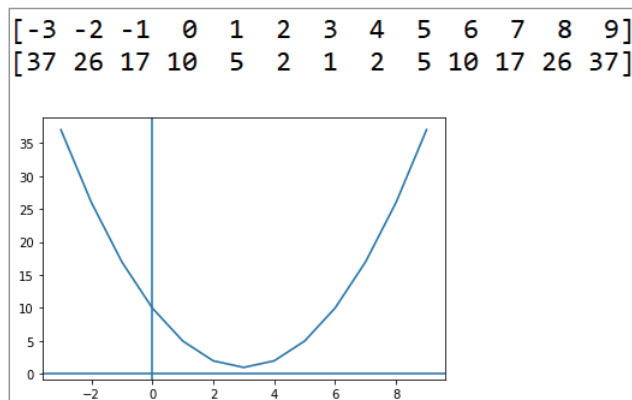
$$\begin{aligned}y &= (x-3)^2 + 1 \quad (x=3 \text{ 有極小值 } 1) \\ &= x^2 - 6x + 10\end{aligned}$$

繪製以上函數的程式如下：

```
import matplotlib.pyplot as plt
import numpy as np
x = np.arange(-3, 10, 1) #數字是我多次修正的結果，這樣可以表現圖形
最有變化的部分

print(x)
y = x *x -6*x +10
print(y)
plt.plot(x, y)
plt.axhline(y=0) #x軸
plt.axvline(x=0) #y軸
plt.show()
```

以上程式執行結果如下：(  $x$  從 -3 到 10，這是我慢慢觀察圖形，所調整出來的範圍，這樣才能畫出函數最有變化的範圍)



以下  $x$  是三次方，三次方圖形大部份有兩個臨界點，或稱相對極點。例如：

$$y=2x^3-3x^2-12x+3$$

將以上對  $x$  微分，得到斜率方程式如下：

$$dy/dx=6x^2-6x-12$$

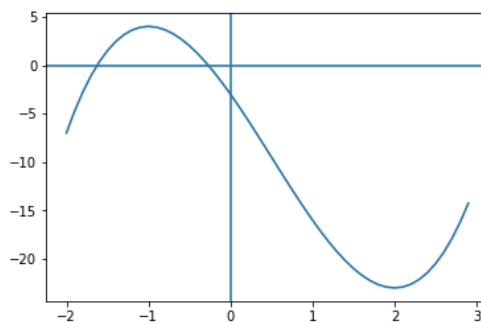
設其斜率為零如下：

$$dy/dx=6x^2-6x-12=0$$

解出  $x=-1$  或  $x=2$  時斜率為零，此點即為臨界點，撰寫程式驗證如下：

```
import matplotlib.pyplot as plt
import numpy as np
x = np.arange(-2, 3, 0.1) # 這是我慢慢觀察圖形，所調整出來的範圍
y=2*x**3-3*x**2-12*x-3
plt.plot(x,y)
plt.axhline(y=0) #x軸
plt.axvline(x=0) #y軸
plt.show()
```

執行結果如下：(請留意  $x=-1,2$  的數化)



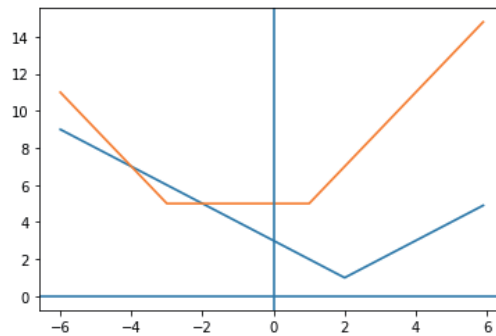


## 絕對值函數

方程式有一個絕對值，就會有一個轉折點，請看以下程式。

```
import matplotlib.pyplot as plt
import numpy as np
x = np.arange(-6, 6, 0.1)
y1=abs(x-2)+1#一個折點
y2=abs(x+3)+abs(x-1)+1#兩個折點
plt.plot(x,y1)
plt.plot(x,y2)
plt.axhline(y=0)#x軸
plt.axvline(x=0)#y軸
plt.show()
```

以上執行結果如下圖：



### 自我練習

1. 範例 4\_1a 的自我練習，我們請同學自己在紙上描點繪圖，現在請您用電腦直接繪製函數圖形。
2. 請同時繪出  $y=x+3$  與  $y=x^2-2$ ，並觀察其交點個數。
3. 請寫一個程式，可以輸入兩個座標，並於螢幕畫出連接此兩點的直線。
4. 請寫一個程式，可以輸入兩個截距，並畫出此直線。例如，輸入 x 截距是 3，y 截距是 4，然後繪出此直線。

5. 請寫一程式，可以輸入一個二次方程式  $y=f(x)=ax^2+bx+c$ ，計算其判別式  $d=b^2-4ac$ ，求其解，並繪出其圖形。例如，請分別輸入  $y=x^2-2x+1$ 、 $y=x^2+x-6$ 、 $y=3x^2+x+1$ 、 $y=-x^2-x+6$ 、 $y=-3x^2+x-3$ ，計算其判別式，求其解，繪出其圖形，觀察與  $y$  軸的交點個數。

## 7-3 課程學習成果

111 學年度學測個人申請入學需要提供『課程學習成果』，108 課綱強調『素養導向』，什麼是素養導向？簡單的說就是打破各科疆界，把知識應用在實際的現實生活中，也就是學生要能將各科的學習融會貫通，統整在一起。例如，本書就很強調跨領域學習，所以書中不斷幫大家解決數學、英文等問題，課程學習成果當然是把大家的作業與報告加深或加廣，以下是我提供的『課程學習成果』參考範例。

### 開根號計算之研究

專題學生：洪國勝

指導老師：洪國勝

高雄市立新莊高中

#### 壹、摘要

分別使用循序法、二分猜值法、遞迴法、與二項式定理，配合 Python 程式設計工具，解出開根號運算。

#### 貳、研究動機

於高一資訊科技的演算法我們學到了『循序法、二分猜值法、遞迴法』、於國中數學我們學到『二項式』定理，於多元選修我學到

『Python』程式設計，由於對資訊科學非常有興趣，所以我將開根號的四個演算法，以 Python 程式設計實現，希望作為申請入學課程學習成果的依據。

## 參、專題原理

### 一、Python語法

#### 算術運算子 (Arithmetic operators)

下表是 Python 的算術運算子列表：

運算子	定義	優先順序	結合律
**	次方	1	由左至右
+/-	正負號	2	由右至左
*	乘法運算	3	由左至右
/	除法，得實數商	3	由左至右
//	除法，得整數商	3	由左至右
%	求餘數 (Modulus)	3	由左至右
+/-	加法 / 減法運算	4	由左至右
=	指派	14	由右至左

例如：

```
a=5;b=2
print(a**b)
```

結果是 25。

#### 決策運算

Python 的決策指令是『if~else~』，其語法如下：

```
if運算式:
    敘述區塊1;
[else :
    敘述區塊2;]
```

例如，以下程式可評判成績是否及格。

```
a=66
if a>=60:
    b="Pass"
else:
    b="Fail"
print(b)
```

### 迴圈運算

Python 迴圈有兩種，分別是 for 與 while，for 用於程式設計階段已知執行次數，例如，以下程式可進行乘法運算。

```
a=5
b=8
s=0
for b in range(1,b+1) :
    s=s+a
print(s)#40
```

while 用於設計階段不知迴圈要重複幾次。例如，除法運算是，只要被除數大於除數，就要重複執行以下敘述

```
被除數=被除數-除數
商=商+1
```

所以程式如下：

```
a=8#被除數
b=3#除數
q=0#商
while (a>=b):#只要(被除數>除數) 就執行迴圈
    a=a-b    # (被除數)-(除數)
    q=q+1    #商每次遞增1
print(q)# 商數
print(a)#餘數
```

## 串列

前面的 a,b,c 適用少量的資料處理，串列則適用於大數據分析的資料結構。例如，若有資料 5 筆如下：

```
3,5,6,8,1
```

若使用 5 個變數儲存，那計算總和、極大與極小等問題，程式會很冗長，若使用串列儲存如下：

```
a=[3,5,6,8,1]
```

這樣就一口氣宣告與指派 a[0]、a[1]、a[2]、a[3]、a[4] 等 5 筆資料。其次，串列可配合迴圈，那就可大批處理資料。例如，以下迴圈就可求其和。

```
a=[3,5,6,8,1]
s=0
for i in range(len(a)):
    s=s+a[i]
print(s)
```

## 二、循序法猜值法

循序法是由小而大，逐一將所有可能的答案通通代入。例如，以下程式就可解出開根號整數解。

```
a=9
for i in range(9):
    if i*i>=a:
        print(i)#3
        break
```

以下程式，可解出開根號實數解。

```
a=9
s=10#放大倍數
```

```

for i in range(a*s):
    if (i/s)*(i/s)>=a:
        print(i/s)#3.0
        break

```

### 三、二分猜值法

前面循序法是一個一個猜，每猜一個，猜值範圍僅減少一個；二分猜值法則是每次猜其一半，這樣每次就可將猜值範圍減少一半。使用二分猜值法解平方根的程式如下：

```

y=9.0
x1=0.0 #下齊
x2=y #上齊
n=1 #猜值次數
while (abs(x1-x2)>0.01):#距離>0.01,繼續猜
    print("%d:x1=%2.2f,x2=%2.2f" %(n,x1,x2))
    x=(x1+x2)/2#每次都猜中間
    t=x*x#計算結果
    if t<y :#猜的太小
        x1=x #調整下界
    else:
        x2=x#調整上界
    n=n+1
print(x)#2.99

```

### 四、遞迴

前面的循序法、二分猜值法都是正向思考，遞迴則是一種反向思考，強調的是，先解決一個，或一部份，而剩下的竟然和原方法相同，這樣是否可繼續使用原方法呢，答案是肯定的，這就是電腦所提供的『遞迴』工具，這樣就可以輕鬆解決問題。以下是以『遞迴』法解開根號的程式。

```

def f(x,x1,x2):
    if abs(x1-x2)<0.01:
        return(x1)

```

```

else:
    y=(x1+x2)/2
    t=y*y
    if t<x:#猜的太小(此時，只是改變猜值範圍而已，方法還是一樣，
        所以繼續呼叫原方法)
        return(f(x,y,x2)) #剩下的一半和原方法相同
    else:# 猜的太大
        return(f(x,x1,y)) #剩下的一半和原方法相同

a=9
x=a
x1=0#下界
x2=a#上界
print(f(x,x1,x2))#2.997

```

## ※五、二項式定理

二項和的二項式定理如下：

$$(10a+b)^2=100a^2+20ab+b^2=100 a^2+b(20a+b)$$

以 138384 為例，開根號運算過程如下：

1. 由以上  $100 a^2+b(20a+b)$ ，表示我們由左到右，每兩個一組，先找 a，再找 b。
2. 找出 a。從 9,8,7 到 1 的平方依序找出小於等於最左邊那一組的數字(本例是 13)，因為 81(9\*9),64(8\*8),49,36,25,16 都太大，所以找到 3，並扣掉 3 的平方 9，剩下 4，請看下圖運算步驟 1。
3. 使用迴圈，從第二組數字開始逐一找  $b_1b_2b_3\cdots$ 。
  - (1) 計算每一次的餘數。餘數  $d = \text{前面餘數} \times 100 + \text{這一組數字}$ ，本例  $d$  是 483。
  - (2) 用迴圈找 b。b 從 9,8,7 到 1，找  $b(20a + b)$  小於餘數  $d$ 。

例如，本例 a 是 3，d 是 483，所以運算步驟如下：

$$b=9,9*(20*3+9)=621 \text{ 太大，不行}$$

$$b=8,8*(20*3+8)=544 \text{ 太大，不行}$$

$b=7,7*(20*3+7)=469$  已經小於 483，所以可以，此迴圈結束，請看下圖步驟 2。

(3) 餘數  $d$  扣掉  $b(20a + b)$ 。 $d = d - b(20a + b)$ 。

5. 輸出  $ab_1b_2b_3\cdots$ ，即為所求。

步驟		3	7	2
1		1 3 8 3 8 4		
		9		
2	$3 \times 20 + 7 = 67$ $67 \times 7 = 469$	4 8 3		
		4 6 9		
3	$37 \times 20 + 2 = 742$ $742 \times 2 = 1484$		1 4 8 4	
			1 4 8 4	
4				0

6. 以下是 Python 程式列印

```
aa=138384
bb=[0]
i=0
#每兩個數字一組
while aa>0:
    i=i+1
    bb.append(aa %100)
    aa=aa//100
print(bb)
n=i
print(n)
#找出a
d=bb[n]
a=10
f=100
while f>d:
    a=a-1
    f=a*a
d=d-f
print('1. a=%d,d=%d'% (a,d) )
n=n-1
```



```

print('n=%d'% n)
#找b
for i in range(n,1-1,-1):
    d=d*100+bb[i]
    b=10
    print('2. a=%d,d=%d'% (a,d) )
    f=10000
    while f>d:
        b=b-1
        f=b*(20*a+b)
        d=d-f #d 是每次扣剩的
        a=a*10+b
    print('平方根是:%d'% a)
    print('d=%d'% d)

```

## 肆、心得

程式設計是一個學習數學的好工具，自己把所有演算法，用程式設計工具實現，真的可以更加瞭解數學，感謝洪老師的指導，讓我使用程式設計工具解決很多數學問題，更加強我往資訊科學研究的動力。

## 六、參考資料

Python 程式設計入門教材，洪國勝，泉勝 108 年出版。

## ※七、延伸專題

開立方原理如下：

$$(10a+b)^3=1000a^3+300a^2b+30ab^2+b^3=1000a^3+(300a^2+30ab+b^2)b$$

請根據以上演算法實際操作開立方如下：

	2 5
	1 5 6 2 5
	8
300*4+30*2*5+25=1525	7 6 2 5
	7 6 2 5
	0

**MEMO**