MATLAB 入門

譚立暉 財金所碩一

MATLAB簡介

- 現在最流行的數學軟體之一
- 強大的矩陣運算、數值分析和繪圖能力
- 內建多種不同用途的函數，包括統計、財務等

如何取得MATLAB

1. 自己想辦法取得光碟
2. 在計中使用
3. 軟體銀行

如何取得MATLAB – 軟體銀行

- 計中網站>一般服務>軟體銀行

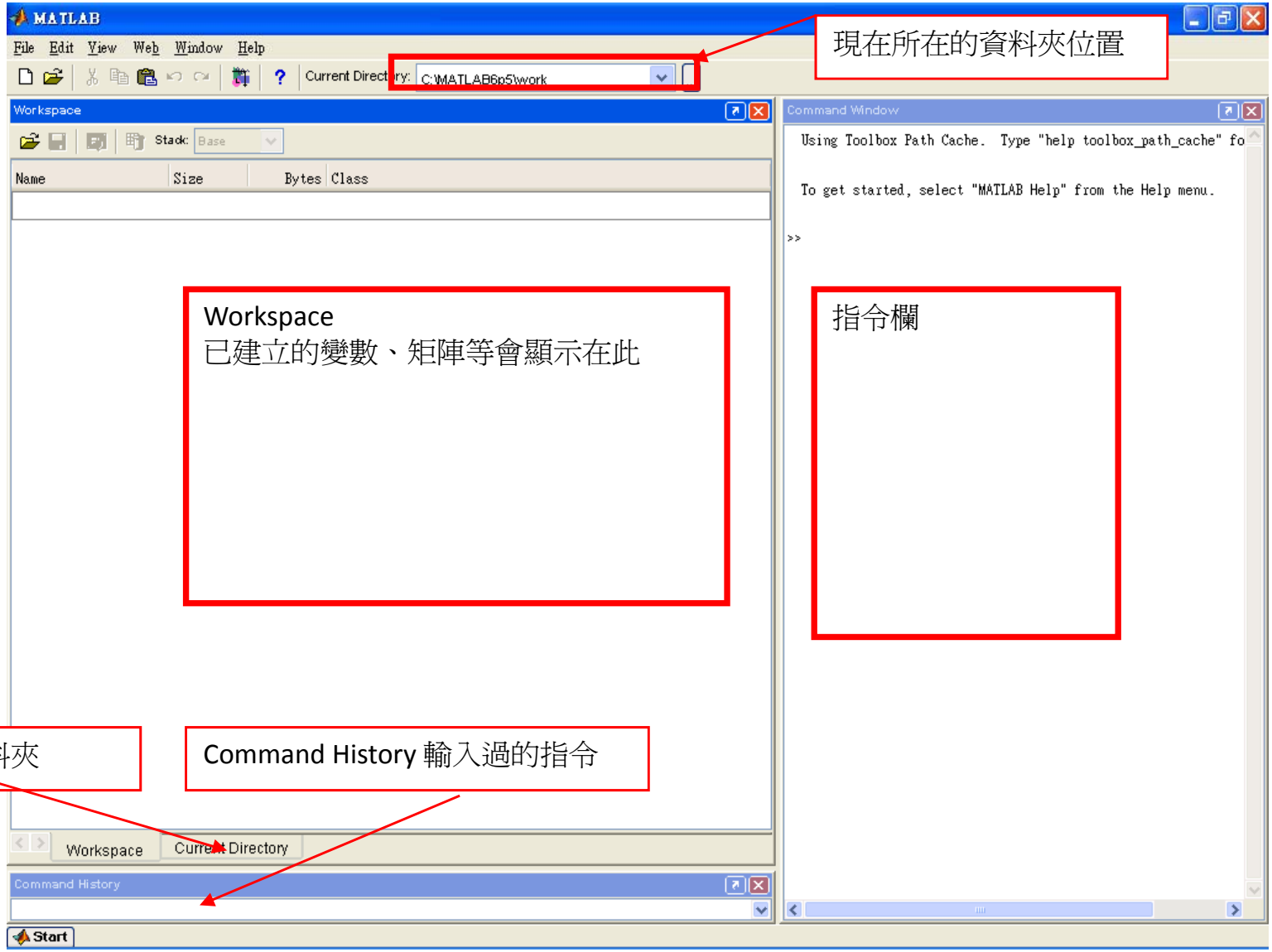
http://softbank.cc.ntu.edu.tw/User/NTUsoftbank.html

(必需要在台大的IP下)

- 在家中如何使用?
  計中網站>網路服務>虛擬私有網路>
  安裝設定與說明>SSL VPN連線方式說明

http://ccnet.ntu.edu.tw/vpn/install.html#ssl

Section 1 認識Matlab 介面操作



Section 2 簡單的Matlab計算

- **2.1 如何用Matlab計算?**

- 1+1
- 3^5
- sin(pi/2)
- exp(0.5)
- sqrt(1+2*i)

2.2 基本的數學函數(Matlab內建函數)

- **Matlab語法**
  - exp(x)
  - sqrt(x)
  - log(x)
  - cos(x)　sin(x)
  - tan(x)
  - acos(x)　asin(x)

- **2.3 變數與變數型態**
- number = 1.32;
- message ='hello';

- 變數名稱只能用英文、數字和底線，不能有空格，要英文開頭
- 變數名稱的大小寫不一樣，會視為不同變數，如XAve和xave不一樣
- 在指令最後加上分號(;)，按Enter後不會出現計算結果
- Matlab有一些預設變數，如pi(圓週率)，i(虛數)

- **2.4 陣列(Array)和矩陣(Matrix)**
- AAA = [1, 2, 4, 8];
- BBB= [0.1, 5, 3; 6, 7, 8];

- 輸入AAA(3)　出現ans = 4
- 輸入BBB(2,3)　出現ans = 8

- 矩陣之運算：
- A .* B　代表A、B之對應元素相乘
- A ./ B
- A .^ 2　代表A裡面的所有元素各自平方

2.5 建立函數

- 方法一：
  - fun = @(x,y) x^2+sin(y);
  - 輸入fun(2,2)　出現 ans = 4.9093
- 方法二：
  - syms x y;　　創造兩個未知數 x 和 y
  - fun = x^2+sin(y);　　建立一個函數fun(x,y) = x^2+siny
  - 輸入 subs(fun,{x,y},{2,2})　出現 ans = 4.9093
- 方法三：
  - fun = inline('x^2+sin(y));

- **2.6 解一元方程**
- 解出f(x)=0
- 方法一/方法三：

  fun=@(x) x^2-1; 或 fun=inline('x^2-1');

  fzero(fun,0.5)　　程式會在x=0.5附近找根
- 方法二：

  f = x^2-1;

  solve(f)　　程式會解出所有的根

- **2.7 讀入檔案**
- A = [1,2,3;4,5,6];

- 寫：
- xlswrite('filename.xls',A);
- csvwrite('filename.csv',A);

- 讀：
- B = xlsread('filename.xls');
- B = csvread('filename.csv');

Section 3 繪圖

- **3.1 平面繪圖**
- 用描點

```
x = [0 1 2 3 4];
y = [0 1 4 9 16];
plot(x,y)
```

- 畫出函數

方法1

```
x2=inline('x^2');
fplot(x2,[0 1])
```

方法2

```
ezplot('x^2',[0,1]);
```

3.2 立體繪圖

- 畫出函數

方法1

```
[X,Y] = meshgrid(-2:0.2:2, -2:0.2:2);
Z = X .* exp(-X.^2 - Y.^2);
surf(X,Y,Z)
```

方法2

```
ezsurf('x*exp(-x^2 - y^2)',[-1 1 -2 2])
```

注意：函數要用單引號包住

- **3.2 立體繪圖**
- **畫出函數(曲線) plot3**

例

```
t=[0:pi/30:6*pi];
x=t.*cos(t);
y=t.*sin(t);
z=t;
plot3(x,y,z)
```

Section 4 邏輯運算 - if

if condition
  statement;
end
	if x > 2
	  sum = sum +1;
	end
		if x <2 && y > 1
		  z = x + y;
		end

Section 4 邏輯運算 - if

if condition1
  statement1;
elseif condition2
  statement2;
else
  statement3;
end
	if x > 0
	  output = ‘positive’;
	elseif x < 0
	  output = ‘negative’;
	else
	  output = ‘zero’;
	end

Section 4 邏輯運算 - for

for counter = a:b
  statement;
end
	for i = 2:10
	  A(i) = A(i-1) + 1;
	  sum = i;
	end
		for k = 1:-0.1:0
		  D = 2*k;
		end

Section 4 邏輯運算 - while

while condition
  statement;
end
	while k < 10
	  B(k) = k;
	  k = k + 1;
	end

Section 4 邏輯運算

- 邏輯相關運算

  || 或

  && 且

  == 等於

  ~= 不等於

  >= <= 大於等於 小於等於

Section 5 m-file

- **一般m-file：**
  把指令寫在m-file裡，若把檔名取為AAA.m，則在Matlab指令欄裡輸入AAA，則會執行AAA.m的指令。

Section 5 m-file - function型的m-file

function result = MyDivide(x,y)	在第一行寫function result是要輸出的結果 MyDivide是function名稱，m-file檔名一定要存成MyDivide.m (x,y)是需要輸入的參數
if y~=0 % We avoid y=0 result = x/y; end	在第二行開始可輸入指令，求出result %後面為註解，只供programmer參考，不會當作程式的一部分

Section 6 常用財務函數

Black-Scholes Option pricing model

[Call, Put] = blsprice(Price, Strike, Rate, Time, Volatility, Yield)

[CallDelta, PutDelta] = blsdelta(Price, Strike, Rate, Time, Volatility, Yield)

Gamma = blsgamma(Price, Strike, Rate, Time, Volatility, Yield)

Vega = blsvega(Price, Strike, Rate, Time, Volatility, Yield)

[CallRho, PutRho]= blsrho(Price, Strike, Rate, Time, Volatility, Yield)

blsimpv(S,K,r,T,C)

Binomial Option pricing model

[AssetPrice, OptionValue] = **binprice**(Price, Strike, Rate, Time, Increment, Volatility, Flag, DividendRate, Dividend, ExDiv)

Section 6 常用財務函數

- **一般函數**

平均數 mean(X)　其中X是一個矩陣

加總　sum(X)

標準差 std(X)

下高斯 floor(y)

上高斯 ceil(y)

最接近之整數 round(y)

絕對值 abs(y)

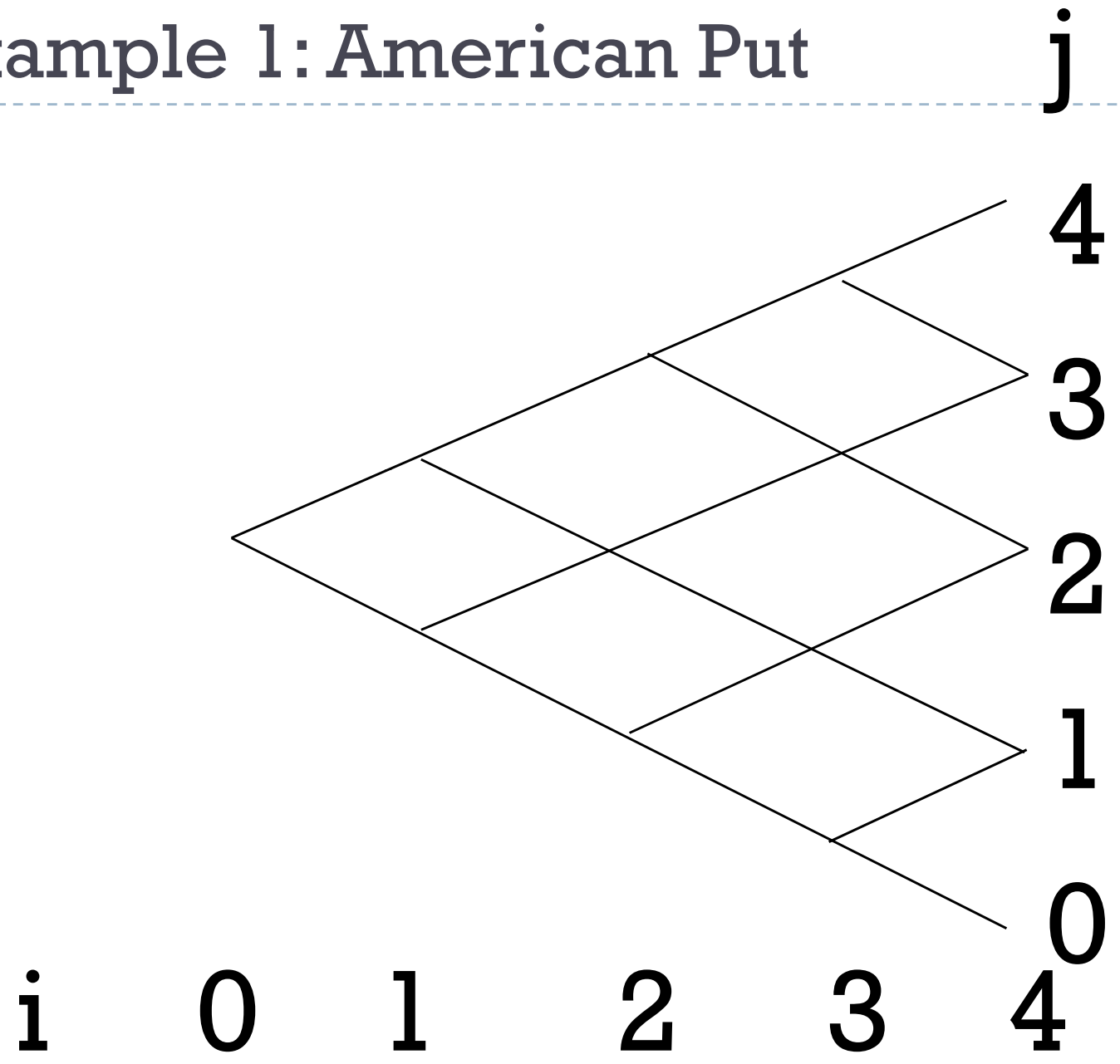
PDF　pdf('Normal',x,mu,v)

CDF　cdf('Normal',x,mu,v)

隨機量 randn 產生標準常態分配的一個隨機量

Gamma function Gamma(alpha)

Example 1: American Put



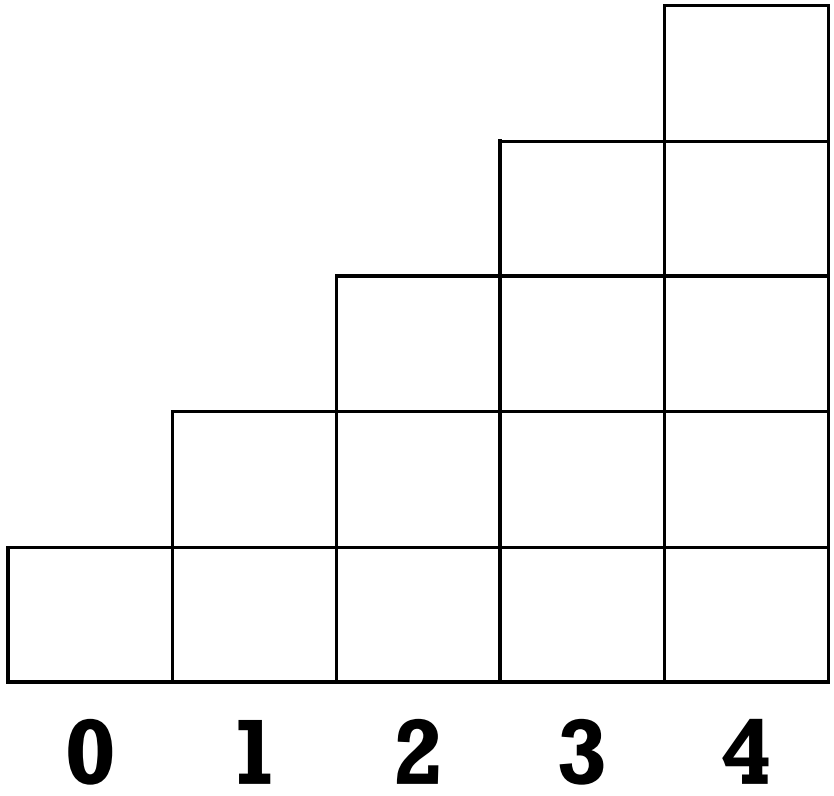
Example 1: American Put(1/2)

```
function opt1 = AmPut(s,x,r,q,sig,t,n)
% s:present stock price
% x:strike price
% r:riskfree rate
% q:dividend yield
% sig:standard derivation of return
% t: time to maturity
% n: number of steps

dt  = t/n;
u = exp(sig*sqrt(dt));
d = exp(-sig*sqrt(dt));
a = exp((r-q)*dt);
p = (a-d)/(u-d);
```

Example 1: American Put(2/2)

```
for j = 0:n;
  pr(j+1,2) = s*(d^(n-j))*(u^j);
  opt(j+1,2) = max(x-pr(j+1,2),0);
end;

for i = n-1:-1:0;
    for j = 0:i;
        pr(j+1,1) = s*(d^(i-j))*(u^j);
        opt(j+1,1) = max( x-pr(j+1,1),(p*opt(j+2,2)+(1-p)*opt(j+1,2))*exp(-r*dt));
    end
opt(:,2) = opt(:,1);
pr(:,2) = pr(:,1);
end
opt1 = opt(1,1);
```

Example 1: American Put



Example 2: Plotting the European Call Price

```
% European Call Price
r = 0.03;
sigma = 0.4;
K = 50;
T = 1;
Yield = 0;
for s = 1:100
   Stock(s) = s;
   Call(s) = blsprice(s,K,r,T,sigma,Yield);
end

plot(Stock,Call);
legend('Call Price');
xlabel('Stock Price');
ylabel('Call Price');
```

Example 3: Monte Carlo Simulation(1/2)

```
function StockMC(S_int,sigma,r,T,n)
% S_int: Initial stock value
% sigma: Standard derivation
% r: Risk free interest rate
% T: Time to maturity
% n: The number of periods

dt=T/n;
S=zeros(n+1);
S(1)=S_int;
t=zeros(n+1);  %先把Array建立好
```

Example 3: Monte Carlo Simulation(1/2)

```
for j=1:n
  t(j+1)=dt*j;
end

for j=1:n
   S(j+1)=S(j)*exp( dt*(r-0.5*(sigma^2)) + sigma*sqrt(dt)*randn );
end

plot(t,S,'r');
xlabel('Time');
ylabel('Stock Price');
hold on;
```

Example 4: Fixed Strike Lookback Put – Using MC method

- 現有一個選擇權，其期末payoff是

$$\text{Max}\{ S(T)-K, S(T/2)-K, 0\}$$

Example 4: Fixed Strike Lookback Put – Using MC method

```
function price =
  FixStkLBCallMC(S_int,k,sigma,r,T,n)
% S_int: Initial stock value
% sigma: Standard derivation
% r: Risk free interest rate
% T: Time to maturity
% n: The number of repeat
R = exp(r*T);
```

Example 4: Fixed Strike Lookback Put – Using MC method

```
sum = 0;
for i = 1:n
    dt=T/2;
    S=zeros(3);
    S(1)=S_int;
    for j=1:2
        S(j+1)=S(j)*exp( dt*(r-0.5*(sigma^2)) + sigma*sqrt(dt)*randn );
    end
    c = max(S(3)-k,S(2)-k)/R;
    c = max(c,0);
    sum = sum + c;
end
price = sum/n;
```

Extended Binomial Tree

其他指令

- 在一個圖中畫兩條以上的曲線，例：

```
plot(x,y1);
hold on;
plot(x,y2);
```