適應控制HW2\_0056C203\_廖育賢

**目標 :** 以RLS等方法估測未知系統之數學式(y=exp(-x))。

**步驟:**

(1)方法:

RLS:

RLS (exponential forgetting):

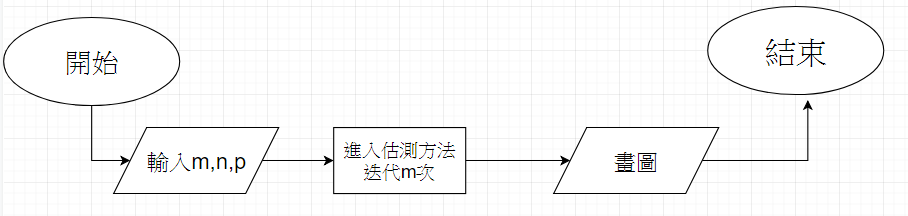
/

S.A.:

P.A.:

LMS:

(2)程式流程圖:



**測試結果:**

(1)程式碼:

…………………………… Main code ……………………………………

%%

%RLS;

clear ;

clc ;

format long;

m=300;

n=4;

p=100\*eye(n,n);

theta=zeros(n,m+1);

phi=phi(m,n);

y=y(m);

for k=1:m

[p ,theta(:,k+1)]=rls(p,theta(:,k),phi(k,:),y(k,:)) ;

end

draw(phi,theta,m,n);

%%

%RLS\_forgetting

clear ;

clc ;

format long;

m=500;

n=4;

theta=zeros(n,m+1);

phi=phi(m,n);

y=y(m);

p=100\*eye(n,n);

lambda=0.99;

for k=1:m [p ,theta(:,k+1)]=rls\_forgetting(p,theta(:,k),phi(k,:),y(k,:),lambda) ;

end

draw(phi,theta,m,n);

%%

%SA

clear ;

clc ;

format long;

m=600;

n=4;

theta=zeros(n,m+1);

phi=phi(m,n);

y=y(m)+ sqrt(3)\*(2\*rand(m,1)-1);

p=0;

gamma=0.2;

for k=1:m

p=p+phi(k,:)\* phi(k,:)';

[theta(:,k+1)]=SA(theta(:,k),phi(k,:),y(k,:),p,gamma) ;

end

draw(phi,theta,m,n);

%%

%PA

clear ;

clc ;

format long;

m=1000;

n=8;

theta=zeros(n,m+1);

phi=phi(m,n);

y=y(m);

alpha=0.1;

gamma=0.01;

for k=1:m

[theta(:,k+1)]=PA(theta(:,k),phi(k,:),y(k,:),gamma,alpha) ;

end

draw(phi,theta,m,n);

%%

%LMS

clear ;

clc ;

format long;

m=210;

n=8;

theta=zeros(n,m+1);

phi=phi(m,n);

y=y(m);

gamma=0.01

for k=1:m

[theta(:,k+1)]=LMS(theta(:,k),phi(k,:),y(k,:),gamma) ;

end

draw(phi,theta,m,n);

…………………… Functions ……………………………

function [ output\_args ] = draw( phi,theta,m,n)

theta(:,m+1)

for i=1:10

theta\_ans(i)=(-1)^(i-1)/factorial(i-1);

end

if(n==4)

figure,

subplot(411)

plot([1,m+1],[theta\_ans(1),theta\_ans(1)],'b--',1:m+1,theta(1,:),'r'),grid on,

axis([-inf, inf,0,1.2])

xlabel('Iteration'),ylabel('theta[1]'),legend('true value','estimation');

subplot(412)

plot([1,m+1],[theta\_ans(2),theta\_ans(2)],'b--',1:m+1,theta(2,:),'r'),grid on,

axis([-inf, inf,-1.2,inf])

xlabel('Iteration'),ylabel('theta[2]')

subplot(413)

plot([1,m+1],[theta\_ans(3),theta\_ans(3)],'b--',1:m+1,theta(3,:),'r'),grid on,

axis([-inf, inf,-inf,0.7])

xlabel('Iteration'),ylabel('theta[3]')

subplot(414)

plot([1,m+1],[theta\_ans(4),theta\_ans(4)],'b--',1:m+1,theta(4,:),'r'),grid on

xlabel('Iteration'),ylabel('theta[4]')

end

if(n==6)

figure,

subplot(321)

plot([1,m+1],[theta\_ans(1),theta\_ans(1)],'b--',1:m+1,theta(1,:),'r'),grid on

,axis([-inf, inf,0,1.2])

xlabel('Iteration'),ylabel('theta[1]')

subplot(322)

plot([1,m+1],[theta\_ans(2),theta\_ans(2)],'b--',1:m+1,theta(2,:),'r'),grid on,

axis([-inf, inf,-1.2,inf])

xlabel('Iteration'),ylabel('theta[2]')

subplot(323)

plot([1,m+1],[theta\_ans(3),theta\_ans(3)],'b--',1:m+1,theta(3,:),'r'),grid on,

axis([-inf, inf,-inf,0.7])

xlabel('Iteration'),ylabel('theta[3]')

subplot(324)

plot([1,m+1],[theta\_ans(4),theta\_ans(4)],'b--',1:m+1,theta(4,:),'r'),grid on

xlabel('Iteration'),ylabel('theta[4]')

subplot(325)

plot([1,m+1],[theta\_ans(5),theta\_ans(5)],'b--',1:m+1,theta(5,:),'r'),grid on

xlabel('Iteration'),ylabel('theta[5]')

subplot(326)

plot([1,m+1],[theta\_ans(6),theta\_ans(6)],'b--',1:m+1,theta(6,:),'r'),grid on

xlabel('Iteration'),ylabel('theta[6]')

end

if(n==8)

figure,

subplot(421)

plot([1,m+1],[theta\_ans(1),theta\_ans(1)],'b--',1:m+1,theta(1,:),'r'),grid on,

axis([-inf, inf,0,1.2])

xlabel('Iteration'),ylabel('theta[1]')

subplot(422)

plot([1,m+1],[theta\_ans(2),theta\_ans(2)],'b--',1:m+1,theta(2,:),'r'),grid on,

axis([-inf, inf,-1.2,inf])

xlabel('Iteration'),ylabel('theta[2]')

subplot(423)

plot([1,m+1],[theta\_ans(3),theta\_ans(3)],'b--',1:m+1,theta(3,:),'r'),grid on,

axis([-inf, inf,-inf,0.7])

xlabel('Iteration'),ylabel('theta[3]')

subplot(424)

plot([1,m+1],[theta\_ans(4),theta\_ans(4)],'b--',1:m+1,theta(4,:),'r'),grid on

xlabel('Iteration'),ylabel('theta[4]')

subplot(425)

plot([1,m+1],[theta\_ans(5),theta\_ans(5)],'b--',1:m+1,theta(5,:),'r'),grid on

xlabel('Iteration'),ylabel('theta[5]')

subplot(426)

plot([1,m+1],[theta\_ans(6),theta\_ans(6)],'b--',1:m+1,theta(6,:),'r'),grid on

xlabel('Iteration'),ylabel('theta[6]')

subplot(427)

plot([1,m+1],[theta\_ans(7),theta\_ans(7)],'b--',1:m+1,theta(7,:),'r'),grid on

xlabel('Iteration'),ylabel('theta[7]')

subplot(428)

plot([1,m+1],[theta\_ans(8),theta\_ans(8)],'b--',1:m+1,theta(8,:),'r'),grid on

xlabel('Iteration'),ylabel('theta[8]')

end

o=phi\*theta(:,m+1);

figure,

x=0.01:0.01:m/100;

plot(x,exp(-x),'b--',0.01:0.01:(m/100),o(1:1:m),'r')

xlabel('x'),ylabel('y=exp(-x)'),axis([-inf, inf,-2,2])

legend('true value','estimation');title('matching')

end

function [ y ] = y( m)

y=zeros(m,1);

for s=1:m

y(s)=exp(-1\*0.01\*s);

end

end

function [ theta ] =SA( theta,phi,y,temp,gamma)

p=gamma/temp;

theta=theta+p\*phi'\*(y-phi\*theta)/(phi\*phi');

end

function [ p,theta ] = rls\_forgetting( p,theta,phi,y ,lambda)

k = p \*phi'./(lambda\*1+phi\*p\*phi');

p=p-k\*phi\*p/lambda;

theta=theta+k\*(y-phi\*theta);

end

function [ p,theta ] = rls( p,theta,phi,y )

k = p \*phi'./(1+phi\*p\*phi');

p=p-k\*phi\*p;

theta=theta+k\*(y-phi\*theta);

end

function [ theta ] =LMS( theta,phi,y,gamma)

theta=theta+gamma\*phi'\*(y-phi\*theta);

end

function [ PHI ] = phi(m,n)

PHI=zeros(m,n);

for f=1:m

for s=1:n

PHI(f,s)=(f/100)^(s-1);

end

end

end

function [ theta ] =PA( theta,phi,y,gamma,alpha)

theta=theta+gamma\*phi'\*(y-phi\*theta)/(alpha+phi\*phi');

end

(2)估測圖:

RLS:

|  |  |  |
| --- | --- | --- |
|  | y-x |  |
| 4階  =10 |  |  |
| 6階  =10 |  |  |
| 8階  =10 |  |  |
| 4階  =100 |  |  |
| 6階  =100 |  |  |
| 8階  =100 |  |  |

RLS():

|  |  |  |
| --- | --- | --- |
|  | y-x |  |
| 4階  =10 |  |  |
| 4階  =10 |  |  |
| 4階=100 |  |  |
| 6階  =10 |  |  |
| 6階  =10 |  |  |
| 6階  =10 |  |  |
| 8階  =10 |  |  |
| 8階  =10 |  |  |

S.A.:

|  |  |  |
| --- | --- | --- |
|  | y-x |  |
| 4階0.2 |  |  |
| 6階  0.2 |  |  |
| 8階  0.2 |  |  |
| 4階  0.8 |  |  |
| 6階  0.8 |  |  |
| 8階  0.8 |  |  |

P.A.:

|  |  |  |
| --- | --- | --- |
|  | y-x |  |
| 4階0.01 |  |  |
| 6階  0.01 |  |  |
| 8階  0.01 |  |  |

LMS:

|  |  |  |
| --- | --- | --- |
|  | y-x |  |
| 4階  0.005 |  |  |
| 6階  0.0001 |  |  |
| 8階  0.000001 |  |  |

(3)表:

估測值(θ):















**結論:**

在RLS方法中initial gain P(0)= ，是決定我們估測方法之重要因素，當=100時的模擬結果比=10時還要準確，所以我們的P(0)要夠大，可以使結果跑得較漂亮；若用高階model(order=6,8)去估測，我們的估測值(θ)收斂得很好，四階model較為粗糙，但若要以更高階model估測，要避免overfitting

的情況發生；另外，經觀察發現，估測迭代越多次，高階待估測係數收斂地越漂亮，而對於低階係數影響效果不大。

當我們遇到Time-Varying系統時，可以採用exponential forgetting方法，以forgetting factor來調整控制器，而對於此y=exp(-x)系統來說，選用0.9模擬之結果還不錯，而時，估測值(θ)在收斂時，抖動的比較厲害。

對於Simplified Algorithm之三種估測方法S.A、P.A.、LMS，實際模擬之結果不理想，原因可能為scalar gain P(t)並不適用於此次系統，以S.A.(8階model)來說，從P(0)=0.7999到P(300)= 7\* e-09非常接近0，這種單一的gain值，對於八項待估測值(θ)來說並不合適，而在RLS方法中的gain 為一矩陣，針對每一估測值分別對應到不同的數值，這種方法考慮到每一參數的情況來做運算，較為合適；另外，理論上S.A.的收斂時間比其他方法慢，而LMS的敏感度大於P.A.，而每種方法都有各自的優勢，若想要節省運算時間則用Simplified Algorithm，若想要精確度，則用RLS；藉由此次模擬，我們可以知道估測器的好壞受到很多因素的影響，如何將影響降低是我們可以深入研究的課題。