作业七

6.15、考虑一阶模型：的线性预测。假设白噪声的方差为。使用抽头数为的滤波器，用和算法实现的线性预测。并对两种方法进行分析、比较。

**解：**应用RLS算法迭代求解最优权向量：

①初始化：，是小的正数，，遗忘因子；

②当时，完成以下迭代运算：

增益向量 

先验估计误差 





③令，转第2步。

MATLAB程序及结果：

clear

clc

a1 = 0.99; %AR模型系数

sigma = 0.995; %白噪声

N = 1000; %数据个数

n0 = 1; %需实现n0步线性预测

M = 2; %滤波器阶数

JFWC = zeros(500,N-1);

W1 = zeros(500,N);

W2 = zeros(500,N);

for m = 1:500

%产生AR模型输出序列

vn = sqrt(sigma) \* randn(N,1); %产生白噪声

nume = 1; %分子系数

deno = [1 a1]; %分母系数

u0 = zeros(length(deno) - 1,1); %初始数据

xic = filtic(nume,deno,u0); %初始条件

un = filter(nume,deno,vn,xic); %产生数据

%产生期望响应信号和观测数据矩阵

b = un(n0+1:N); %预测的期望响应

L = length(b);

un1 = [zeros(M-1,1).' , un.'].'; %扩展数据

A = zeros(M,L);

%构建观测数据矩阵

for k = 1:L

A(:,k) = un1(M-1+k:-1:k);

end

%应用RLS算法进行迭代求最优权向量

delta = 0.004; %调整参数

lambda = 0.98; %遗忘因子

w = zeros(M,L+1);

epsilon = zeros(L,1);

P1 = eye(M) / delta;

%RLS算法迭代过程

for k = 1:L

PIn = P1 \* A(:,k);

denok = lambda + A(:,k)' \* PIn;

kn = PIn / denok;

epsilon(k) = b(k) - w(:,k)' \* A(:,k);

w(:,k+1) = w(:,k) + kn \* conj(epsilon(k));

P1 = P1 / lambda - kn \* A(:,k)' \* P1 / lambda;

end

MSE = abs(epsilon) .^ 2;

JFWC(m,:) = MSE;

W1(m,:) = w(1,:);

W2(m,:) = w(2,:);

end

%求500次均值

mse = mean(JFWC);

w1 = mean(W1);

w2 = mean(W2);

figure(1)

plot(1:500,mse(1:500),'r');

xlabel('迭代次数');

ylabel('MSE');

title('均方误差');

grid on;

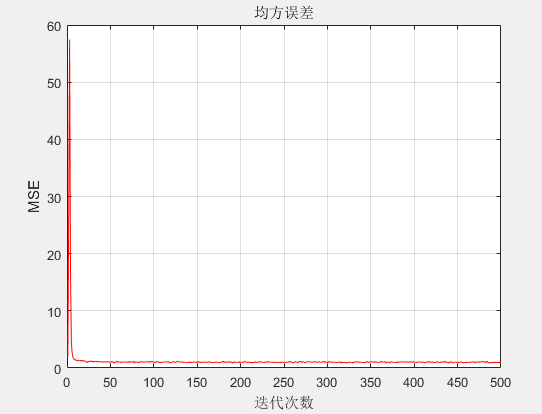


图7-1 均方误差

%权值

figure(2)

plot(1:500,w1(1:500),'r');

hold on;

plot(1:500,w2(1:500),'b');

hold off;

xlabel('迭代次数');

ylabel('权值');

title('计算权值');

grid on;

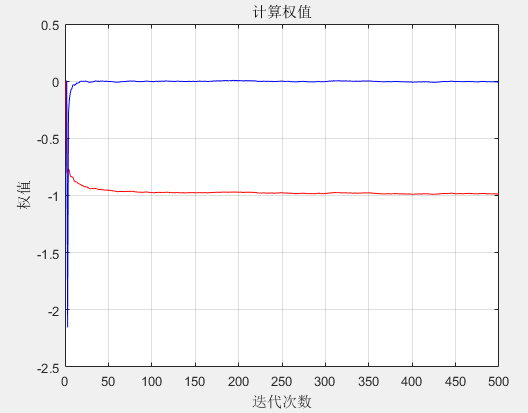


图7-2 计算权值

%100次试验

W1 = zeros(100,4500);

W2 = zeros(100,4500);

for m = 1:100

%产生样本序列

L = 5000;

a1 = 0.99;

s = 0.995;

n = 1:L;

v = sqrt(s) \* randn(L,1);

u(1) = v(1);

for k = 2:L

u(k) = -a1 \* u(k-1) + v(k);

end

u=u(500:end);

%LMS迭代算法

M = 2;

w(1,:) = zeros(1,M);

e(1) = u(1);

mu = 0.001;

uu = zeros(1,M);

w(2,:) = w(1,:) + mu \* e(1) \* uu;

uu = [u(1),uu(1:M-1)];

dd = (w(2,:)\*uu')';

e(2) = u(3) - dd;

for k = 3:4501

w(k,:) = w(k-1,:) + mu \* e(k-1) \* uu;

uu = [u(k-1),uu(1:M-1)];

dd = (w(k,:)\*uu')';

e(k) = u(k)-dd;

end

W1(m,:) = w(1:4500,1)';

W2(m,:) = w(1:4500,2)';

end

W11 = mean(W1);

W22 = mean(W2);

plot(1:4500,W11,'r');

hold on;

plot(1:4500,W22,'b');

xlabel('迭代次数');

ylabel('抽头权值');

title('步长0.05');

grid on;

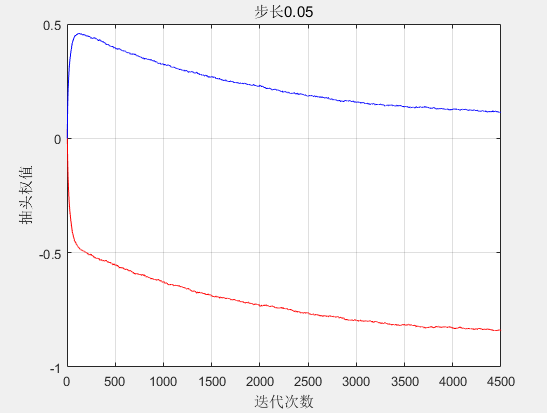


图7-3 两种方法比较图

比较可得，LMS计算权值时收敛相较RLS算法收敛非常慢，实际应用中更多的运用到RLS算法。