**SVM**

%% I. 清空环境变量

clear all

clc

%% II. 导入数据

data=xlsread('C:\Users\DELL\Desktop\海域地震信息.xlsx');

attributes=data(:,1:4);

strength=data(:,5);%strength=log(strength);%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%

% 1. 随机产生训练集和测试集

n = randperm(size(attributes,1));

%%

% 2. 训练集——80个样本

p\_train = attributes(n(1:495),:);

t\_train = strength(n(1:495),:);

%%

% 3. 测试集——23个样本

p\_test = attributes(n(496:end),:);

t\_test = strength(n(496:end),:);

%% III. 数据归一化

%%

% 1. 训练集

[pn\_train,inputps] = mapminmax(p\_train');

pn\_train = pn\_train';

pn\_test = mapminmax('apply',p\_test',inputps');

pn\_test = pn\_test';

%%

% 2. 测试集

[tn\_train,outputps] = mapminmax(t\_train');

tn\_train = tn\_train';

tn\_test = mapminmax('apply',t\_test',outputps);

tn\_test = tn\_test';

%% IV. SVM模型创建/训练

%%

% 1. 寻找最佳c参数/g参数

[c,g] = meshgrid(-10:0.25:10,-10:0.25:10);

[m,nn] = size(c);

cg = zeros(m,nn);

eps = 10^(-4);

v = 10;

bestc = 0;

bestg = 0;

error = Inf;

for i = 1:m

    step=i

    for j = 1:nn

        cmd = ['-v ',num2str(v),' -t 2',' -c ',num2str(2^c(i,j)),' -g ',num2str(2^g(i,j) ),' -s 3 -p 0.1'];

        cg(i,j) = svmtrain(tn\_train,pn\_train,cmd);

        if cg(i,j) < error

            error = cg(i,j);

            bestc = 2^c(i,j);

            bestg = 2^g(i,j);

        end

        if abs(cg(i,j) - error) <= eps && bestc > 2^c(i,j)

            error = cg(i,j);

            bestc = 2^c(i,j);

            bestg = 2^g(i,j);

        end

    end

end

%%

% 2. 创建/训练SVM

cmd = [' -t 2',' -c ',num2str(bestc),' -g ',num2str(bestg),' -s 3 -p 0.01'];

model = svmtrain(tn\_train,pn\_train,cmd);

%% V. SVM仿真预测

[Predict\_1,error\_1] = svmpredict(tn\_train,pn\_train,model);

[Predict\_2,error\_2] = svmpredict(tn\_test,pn\_test,model);

%%

% 1. 反归一化

predict\_1 = mapminmax('reverse',Predict\_1,outputps);

predict\_2 = mapminmax('reverse',Predict\_2,outputps);

%%

% 2. 结果对比

result\_1 = [t\_train predict\_1];

result\_2 = [t\_test predict\_2];

**SVM**

% 初始化参数

clear all;

close all;

clc;

% 读取两点的坐标

fid=xlsread('D:\温攀2018博 无记录\K-NET记录\2023\20230905132700-4.6n\station-lu');%源文件各台站位置%%%%%%%%%%%%%%%%%%

number\_b=size(fid,1);

for ii=1:number\_b

step=ii

coordinate\_a=fid(ii,:);

coordinate\_b=fid;

% 计算两点间的距离

distance\_ab = [];

for i = 1:number\_b

distance\_ab(i) = distance(coordinate\_a(1),coordinate\_a(2),coordinate\_b(i,1),coordinate\_b(i,2))/180\*pi\*6371;

end

distance\_ab=distance\_ab';

distance\_ab\_zong(:,ii)=distance\_ab;

end

L=tril(distance\_ab\_zong);

NN=size(L);NN=NN(:,1); %所选地震事件中台站记录总数

sort(L);

max\_fid=max(max(L));

min\_fid=min(min(L));

m=2; %%%%%%%%%%%%%%%每组的长度范围

n=ceil(max\_fid/m); %%%%%%%数据分为几组

for i=1:n

a(i)=length(find(L>m\*i-m&L<=m\*i));

end

xx=(m/2:m:m/2+(n-1)\*m);

zong=[xx;a]';

%

for i=1:50

step=i

[ii,jj]=find(L>m\*i-m&L<=m\*i);

position=[ii,jj];

N=length(ii);

%

data=xlsread('C:\Users\DELL\Desktop\海域相关性\PGA.xlsx');%%%%%%%%%%%%%%%%%%%%%%%%源文件%%%%%%%%%%%%%%%%%%%%%%%%%

x=data(:,1)';ij=find(~isnan(x));x=x(ij);%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

yrand=data(:,2)';ij=find(~isnan(yrand));yrand=yrand(ij);yrand=log(yrand);%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

plot(x,yrand,'ro');

%

% 最小二乘拟合

yf=yrand;

f=inline('b(1)+(b(2)+b(3)\*4.6)\*log(x.^2+b(4)^2)','b','x');% 内联函数: f(b,x) = b(1)+b(2).\*x

[q,r]=lsqcurvefit(f,[1,1,1,1],x,yf);%起点[1,0]拟合

plot(x,yrand,'ro','LineWidth',1) %绘制采样点

hold on;

% 绘制拟合曲线

yn = q(1)+(q(2)+q(3)\*4.6)\*log(x.^2+q(4)^2);

hold on;

grid on

%plot(x,y,'c','LineWidth',2);%原函数

plot(x,yn,'b','LineWidth',2);%拟合线

PGA=yrand';PGA=exp(PGA);

model\_PGA=yn';model\_PGA=exp(model\_PGA);

PGA\_x=PGA(ii,1); %%%%%%%%%%%%%%%%%%%%%%%%点x

PGA\_y=PGA(jj,1); %%%%%%%%%%%%%%%%%%%%%%%%点y

model\_PGA\_x=model\_PGA(ii,1);

model\_PGA\_y=model\_PGA(jj,1);

z\_x=log(PGA\_x)-log(model\_PGA\_x);

z\_y=log(PGA\_y)-log(model\_PGA\_y);

%z=[z\_x;z\_y];

%z\_zong(i,:)=z;

%z=unique(z);

%var\_z=var(z,1);

%var\_z\_zong(i,:)=var\_z;

z=log(PGA)-log(model\_PGA);

var\_z=var(z);

y=(1/(2\*N)\*sum((z\_x-z\_y).^2))./var\_z; %%%%%%%%%%%剩下的工作需把该值变为计算值与衰减模型之差%%%%%%%%%%%%

%y=1/(2\*N)\*sum(((z\_x-zx\_mean)/sqrt(var\_zx)-(z\_y-zy\_mean)/sqrt(var\_zy)).^2); %%%%%%%%%%%剩下的工作需把该值变为计算值与衰减模型之差%%%%%%%%%%%%

y\_zong(i,:)=y;

end