clc;

close all;

clear all;

%data=load('Heirtrain.txt');

data1=load('Heir.txt');

dat=csvread('HT.csv');

data=dat\*data1;

ds1=data;

[row,column]=size(data);

%let k=3;

k=3;

C=[];

for i=1:k

C(i,:)=data(i,:);

end

d=[];

for i=1:k

for j=1:row

d(i,j)=sqrt(sum((C(i,:)-data(j,:)).^2));

end

end

s=[];

l=d;

z=[];

for i=1:length(d)

[v,v1]=min(d(:,i));

d(:,i)=0;

d(v1,i)=1;

end

t=d+1;

while(t~=d)

siz=sum(d');

t=d;

C=d\*data;

for i=1:length(siz)

C(i,:)=C(i,:)/siz(i);

end

for i=1:k

for j=1:row

d(i,j)=sqrt(sum((C(i,:)-data(j,:)).^2));

end

end

l=d;

for i=1:length(d)

[v,v1]=min(d(:,i));

d(:,i)=0;

d(v1,i)=1;

end

end

d=d';

d

%data=load('iris1.txt');

%data=data(:,1:end-1);

%d=data(1:2,:)

%d1=data(51:52,:)

%d2=data(101:102,:)

%data=[d;d1;d2];

c=3;

%p1=round(rand(150,3));

p1=zeros(30,1);

p2=ones(30,1);

pl1=[p1;p2;p1;p2;p1;p1;p2;p1;p1;p1];

pl1=[pl1;0;0;0;1];

pl3=[p1;p1;p2;p1;p1;p2;p1;p2;p2;p1;0;1;1;0];

pl2=[p2;p1;p1;p1;p2;p1;p1;p1;p1;p2;1;0;0;0];

p=[pl1';pl2';pl3'];

centre=cent(p,data,c);

centre1=centre+1;

i=0;

while(centre~=centre1)

if(i<25)

dist=fuzzydist(centre,data);

p1=update1(p,dist);

[row1,column1]=size(p1);

centre1=centre;

centre=cent(p1',data,c);

i=i+1;

else

break;

end

end

for i=1:row1

m=p1(i,1);

for j=1:column1

if(p1(i,j)>m)

m=p1(i,j);

end

end

for l=1:column1

if(p1(i,l)==m)

p1(i,l)=1;

else

p1(i,l)=0;

end

end

end

disp('final classification')

disp(p1)

dfs=0;

dfs1=0;

dg=0;

for i=1:row

if(p1(i-dg,:)==d(i-dg,:))

dfs1=dfs1+1;

elseif(dg<50&&p1(i-dg,2)~=1&&p1(i-dg,1)~=1)

data(i-dg,:)=[];

p1(i-dg,:)=[];

d(i-dg,:)=[];

%dfs=dfs+1;

dg=dg+1

else

dfs=dfs+1;

end

end

[r1,c1]=size(find(p1(:,1)==1));

[r2,c2]=size(find(p1(:,2)==1));

[r3,c3]=size(find(p1(:,3)==1));

%ds=find(p1(:,3)==1)

%for i=1:.8\*r3

% ds1(ds(i)-i+1,:)=[];

%end

[r4,c4]=size(find(d(:,1)==1));

[r5,c5]=size(find(d(:,2)==1));

[r6,c6]=size(find(d(:,3)==1));

comparision=[r1,r2,r3;r4,r5,r6]

match=dfs1

mismatch=dfs

correlation=(match/(match+mismatch))\*100

%data=[];

%data=d;

y=[];

for i=1:length(d)

if(d(i,1)==1)

y(i)=0;

elseif(d(i,2)==1)

y(i)=1;

else

y(i)=2;

end

end

y=y';

data(:,end+1)=y;

y=[];

test=data(1:0.3\*end,:);

train=data((.3\*end):end ,:);

%data=load('iris1.txt');

[row121,column121]=size(train);

x=train(:,1:column121-1);

y=train(:,column121);

%r = randperm(150, .3\*row)'

%test=d(r,:);

v=1;

e(:,1)=find(y==0);

e1(:,1)=find(y==1);

e2(:,1)=find(y==2);

w1=x(e(:,1),:);

w2=x(e1(:,1),:);

w3=x(e2(:,1),:);

p=[length(find(y==0)),length(find(y==1)),length(find(y==2))];

p=p./row121;

mean1=sum(w1)./length(w1);

mean2=sum(w2)./length(w2);

mean3=sum(w3)./length(w3);

var=w1;

for i=1:length(w1)

var(i,:)=w1(i,:)-mean1;

end

std1=sqrt((sum(var.^2))/length(w2));

var=w2;

for i=1:length(w2)

var(i,:)=w2(i,:)-mean2;

end

std2=sqrt((sum(var.^2))/length(w1));

var=w3;

for i=1:length(w3)

var(i,:)=w3(i,:)-mean3;

end

std3=sqrt((sum(var.^2))/length(w2));

y111=[];

[rr,cc]=size(test);

for i=1:rr

x1=data(i,1:column121-1);

p0=p(1)\*postprob(x1(1),mean1(1),std1(1))\*postprob(x1(2),mean1(2),std1(2))\*postprob(x1(3),mean1(3),std1(3))\*postprob(x1(4),mean1(4),std1(4));

p1=p(2)\*postprob(x1(1),mean2(1),std2(1))\*postprob(x1(2),mean2(2),std2(2))\*postprob(x1(3),mean2(3),std2(3))\*postprob(x1(4),mean2(4),std2(4));

p2=p(3)\*postprob(x1(1),mean3(1),std3(1))\*postprob(x1(2),mean3(2),std3(2))\*postprob(x1(3),mean3(3),std3(3))\*postprob(x1(4),mean3(4),std3(4));

if(p0>p1&&p0>p2)

y111(i)=0;

elseif(p1>p0&&p1>p2)

y111(i)=1;

else

y111(i)=2;

end

end

match=0;

mismatch=0;

for i=1:length(y111)

if(y111(i)==test(i,5))

match=match+1;

else

mismatch=mismatch+1;

end

end

Naive\_accuracy=(match/rr)\*100

x=[];

y=[];

dataaa1=data;

size\_data = size(data);

eta1 = .02;

eta2 = .03;

no\_grp = 3;

no\_nodes = 4;

in =4;

itr = 200;

n0 = size\_data(1);

n= size\_data(1)/no\_grp;

weights = [0,0.9,0.6,0.3];

for i=1:size\_data(1)

if(data(i,in+1)==0)

data(i,in+1)=0.2;

elseif(data(i,in+1)==1)

data(i,in+1)=0.6;

else

data(i,in+1)=1;

end

end

test=data(1:0.3\*end,:);

train=data((.3\*end):end ,:);

%data=load('iris1.txt');

[row121,column121]=size(train);

x=train(:,1:column121-1);

y=train(:,column121);

size\_data = [];

size\_data = size(train);

%r = randi(size\_data(1),1,4);

r=[100,154,71,219];

c = data(r,1:4);

bias=0;

%r=randperm(150);

for i=1:10000%training

for k=1:size\_data(1)

z1(k) = euclidean(train(k,1:4),c(1,:));%euclidean distance calculation.

z2(k) = euclidean(train(k,1:4),c(2,:));

z3(k) = euclidean(train(k,1:4),c(3,:));

z4(k) = euclidean(train(k,1:4),c(4,:));

phi1(k) = exp(-((z1(k))^2));

phi2(k) = exp(-((z2(k))^2));

phi3(k) = exp(-((z3(k))^2));

phi4(k) = exp(-((z4(k))^2));

phi = [phi1(k) phi2(k) phi3(k) phi4(k)];

y(k)=(phi1(k)\*weights(1,1))+(phi2(k)\*weights(1,2))+(phi3(k)\*weights(1,3))+(phi4(k)\*weights(1,4));

y(k)=y(k)+bias;

for j=1:no\_nodes

c(j,:) = c(j,:) + eta1\*(train(k,in+1) -y(k))\*weights(j)\*(phi(j)\*2) \*(train(k,1:4)-c(j,:));%centre updation

weights(j) = weights(j) + eta2\*(train(k,in+1) -y(k))\*phi(j);%weight updation.

end

e(k) = train(k,in+1) -y(k);

end

err(i) = mse(e);%mean square error calculation

end

%figure;plot(err);title('Mean Square Error');xlabel('iteration --->');

y=[];

size\_data = [];

size\_data = size(test);

mismatch=0;

mismatch1=0;

for k=1:size\_data(1)%testing

z1(k) = euclidean(test((k),1:4),c(1,:));%euclidean distance calculation.

z2(k) = euclidean(test((k),1:4),c(2,:));

z3(k) = euclidean(test((k),1:4),c(3,:));

z4(k) = euclidean(test((k),1:4),c(4,:));

phi1(k) = exp(-((z1(k))^2));

phi2(k) = exp(-((z2(k))^2));

phi3(k) = exp(-((z3(k))^2));

phi4(k) = exp(-((z4(k))^2));

phi = [phi1(k) phi2(k) phi3(k) phi4(k)];

y(k)=(phi1(k)\*weights(1,1))+(phi2(k)\*weights(1,2))+(phi3(k)\*weights(1,3))+(phi4(k)\*weights(1,4));

y(k)=y(k)+bias;

if(y(k)>=test(k,5)+.2 || y(k)<=test(k,5)-.2)

mismatch1=mismatch1+1;

end

end

RBF\_percentage1 = (1 - mismatch1/ size\_data(1))\*100

data=[];

x=[];

y=[];

data=dataaa1;

test=data(1:0.3\*end,:);

train=data((.3\*end):end ,:);

%data=load('iris1.txt');

[row121,column121]=size(train);

x=train(:,1:column121-1);

y=train(:,column121);

for k=1:length(y)

if(y(k)==2)

y(k)=1;

end

end

x1=ones(row121,1);

x=[x1,x];

theta=zeros(column121,1);

alpha=.01;

for i=1:50000

theta(1)=theta(1)-(alpha/m)\*(sum((1./(1+exp(-x\*theta)))-y));

theta(2)=theta(2)-(alpha/m)\*(sum(((1./(1+exp(-x\*theta)))-y).\*x(:,2)));

theta(3)=theta(3)-(alpha/m)\*(sum(((1./(1+exp(-x\*theta)))-y).\*x(:,3)));

theta(4)=theta(4)-(alpha/m)\*(sum(((1./(1+exp(-x\*theta)))-y).\*x(:,4)));

theta(5)=theta(5)-(alpha/m)\*(sum(((1./(1+exp(-x\*theta)))-y).\*x(:,5)));

end

w=theta;

testx=test(:,1:column121-1);

testy=test(:,column121);

nTest = size(testx,1);

res = zeros(nTest,1);

for i = 1:nTest

sigm(i) = sigmoid(w(1)+testx(i,:) \* w(2:end));

%if sigm >= 0.5

% res(i) = 1;

%else

% res(i) = 0;

%end

end

%errors = abs(y - res);

%err = sum(errors)

%percentage = (1 - err / size(x, 1))\*100

test=data(1:0.3\*end,:);

train=data((.3\*end):end ,:);

%data=load('iris1.txt');

[row121,column121]=size(train);

x=train(:,1:column121-1);

y=train(:,column121);

for k=1:length(y)

if(y(k)==2)

y(k)=0;

end

end

x1=ones(row121,1);

x=[x1,x];

theta=zeros(column121,1);

alpha=.01;

for i=1:50000

theta(1)=theta(1)-(alpha/m)\*(sum((1./(1+exp(-x\*theta)))-y));

theta(2)=theta(2)-(alpha/m)\*(sum(((1./(1+exp(-x\*theta)))-y).\*x(:,2)));

theta(3)=theta(3)-(alpha/m)\*(sum(((1./(1+exp(-x\*theta)))-y).\*x(:,3)));

theta(4)=theta(4)-(alpha/m)\*(sum(((1./(1+exp(-x\*theta)))-y).\*x(:,4)));

theta(5)=theta(5)-(alpha/m)\*(sum(((1./(1+exp(-x\*theta)))-y).\*x(:,5)));

end

w=theta;

testx=test(:,1:column121-1);

testy=test(:,column121);

nTest = size(testx,1);

res = zeros(nTest,1);

for i = 1:nTest

sigm1(i) = sigmoid(w(1)+testx(i,:) \* w(2:end));

%if sigm >= 0.5

% res(i) = 1;

%else

% res(i) = 0;

%end

end

test=data(1:0.3\*end,:);

train=data((.3\*end):end ,:);

%data=load('iris1.txt');

[row121,column121]=size(train);

x=train(:,1:column121-1);

y=train(:,column121);

for k=1:length(y)

if(y(k)==2)

y(k)=1;

else

y(k)=0;

end

end

x1=ones(row121,1);

x=[x1,x];

theta=zeros(column121,1);

alpha=.01;

for i=1:50000

theta(1)=theta(1)-(alpha/m)\*(sum((1./(1+exp(-x\*theta)))-y));

theta(2)=theta(2)-(alpha/m)\*(sum(((1./(1+exp(-x\*theta)))-y).\*x(:,2)));

theta(3)=theta(3)-(alpha/m)\*(sum(((1./(1+exp(-x\*theta)))-y).\*x(:,3)));

theta(4)=theta(4)-(alpha/m)\*(sum(((1./(1+exp(-x\*theta)))-y).\*x(:,4)));

theta(5)=theta(5)-(alpha/m)\*(sum(((1./(1+exp(-x\*theta)))-y).\*x(:,5)));

end

w=theta;

testx=test(:,1:column121-1);

testy=test(:,column121);

nTest = size(testx,1);

res = zeros(nTest,1);

for i = 1:nTest

sigm2(i) = sigmoid(w(1)+testx(i,:) \* w(2:end));

%if sigm >= 0.5

% res(i) = 1;

%else

% res(i) = 0;

%end

end

sig=[sigm',sigm1',sigm2'];

y1=[];

for i=1:length(sig)

if(sig(i,1)>sig(i,2)&&sig(i,1)>sig(i,3))

y1(i)=0;

elseif(sig(i,2)>sig(i,1)&&sig(i,2)>sig(i,3))

y1(i)=1;

else

y1(i)=2;

end

end

errors = abs(test(:,5) - y1');

err = sum(errors);

logistic\_percentage = (1 - err / size(x, 1))\*100

data1=data;

test=data(1:0.3\*end,:);

train=data((.3\*end):end ,:);

qww=[];

min=100000000;

count=0;

di=[];

vi=find(data(:,5)==2);

for i=1:length(vi)

data(vi,5)=1;

end

vi1=find(data(:,5)==1);

vi0=find(data(:,5)==0);

for i=1:length(vi1)

for j=i:length(vi0)

d=sqrt(sum(((data(vi1(i),:)-data(vi0(j),:)).^2)));

if(min>d)

min=d;

di(1,:)=data(vi1(i),:);

di(2,:)=data(vi0(j),:);

%di(2,end)=-1;

count=0;

end

if(min==d)

count=count+1;

end

end

end

di(:,end+1)=di(:,end);

di(:,end-1)=1;

di=di';

A=di(end,:);

A=A';

di=di(1:end-1,:);

S=[sum(di(:,1).\*di(:,1)),sum(di(:,1).\*di(:,2));sum(di(:,1).\*di(:,2)),sum(di(:,2).\*di(:,2))];

alpha=linsolve(S,A)

for i=1:length(alpha)

w(:,i)=alpha(i)\*di(:,i);

end

w=sum(w');

qww(1,:)=w;

data=data1;

w=[]

min=100000000;

count=0;

di=[];

vi=find(data(:,5)==2);

for i=1:length(vi)

data(vi,5)=0;

end

vi1=find(data(:,5)==1);

vi0=find(data(:,5)==0);

for i=1:length(vi1)

for j=i:length(vi0)

d=sqrt(sum(((data(vi1(i),:)-data(vi0(j),:)).^2)));

if(min>d)

min=d;

di(1,:)=data(vi1(i),:);

di(2,:)=data(vi0(j),:);

%di(2,end)=-1;

count=0;

end

if(min==d)

count=count+1;

end

end

end

di(:,end+1)=di(:,end);

di(:,end-1)=1;

di=di';

A=di(end,:);

A=A';

di=di(1:end-1,:);

S=[sum(di(:,1).\*di(:,1)),sum(di(:,1).\*di(:,2));sum(di(:,1).\*di(:,2)),sum(di(:,2).\*di(:,2))];

alpha=linsolve(S,A)

for i=1:length(alpha)

w(:,i)=alpha(i)\*di(:,i);

end

w=sum(w');

qww(2,:)=w;

data=data1;

w=[]

min=100000000;

count=0;

di=[];

vi=find(data(:,5)==0);

vi2=find(data(:,5)==2);

for i=1:length(vi)

data(vi,5)=1;

%data(vi2,5)=0;

end

vi1=find(data(:,5)==1);

vi0=find(data(:,5)==2);

for i=1:length(vi1)

for j=i:length(vi0)

d=sqrt(sum(((data(vi1(i),:)-data(vi0(j),:)).^2)));

if(min>d)

min=d;

di(1,:)=data(vi1(i),:);

di(2,:)=data(vi0(j),:);

%di(2,end)=-1;

count=0;

end

if(min==d)

count=count+1;

end

end

end

di(:,end+1)=di(:,end);

di(:,end-1)=1;

di=di';

A=di(end,:);

A=A';

di=di(1:end-1,:);

S=[sum(di(:,1).\*di(:,1)),sum(di(:,1).\*di(:,2));sum(di(:,1).\*di(:,2)),sum(di(:,2).\*di(:,2))];

alpha=linsolve(S,A)

for i=1:length(alpha)

w(:,i)=alpha(i)\*di(:,i);

end

w=sum(w');

qww(3,:)=w;

sdsd=[];

for i=1:length(data1)

x1=data1(i,1:end-1);

y11=data1(1,end);

y1=qww(1,1:end-1)\*x1'+qww(1,end);

y2=qww(2,1:end-1)\*x1'+qww(2,end);

y3=qww(3,1:end-1)\*x1'+qww(3,end);

if(y1<0)

d1=0;

else

d1=1;

end

if(y2>1)

d2=1;

else

d2=0;

end

if(y3>2)

d3=2;

else

d3=1;

end

if(d1==0&&(d2==0||d3==1))

%disp('0')

sdsd(i)=0;

elseif((d2==1)&&(d2==1||d3==1))

%disp('1')

sdsd(i)=1;

else

%disp('2')

sdsd(i)=2;

end

end

match=0;

sp=[];

a=1;

mm=0;

for i=1:length(sdsd)

if(sdsd(i)==data1(i,5))

match=match+1;

else

mm=mm+1;

sp(a)=i;

a=a+1;

end

end

svm\_accuracy=(1-mm/length(sdsd))\*100

OUTPUT:

Naive\_accuracy =

88.1579

RBF\_percentage1 =

60.5263

logistic\_percentage =

67.4157

svm\_accuracy =

63.3858