Matrix X row-vector Y X Y\_bi

Y\_b=1(Y==3)

OVA

function name\_of\_file=PNC\_BinaryDataset(class)

S=load('ProjectNA.mat');

X=S.X;

Y=(S.Y==class);

fields\_to\_be\_copied={'size','feature\_s','F','Void\_pixels'};

for field\_=1:length(fields\_to\_be\_copied)

field=fields\_to\_be\_copied{field\_};

metadata.(field)=S.metadata.(field);

end

metadata.C=2;

metadata.classes\_l={S.metadata.classes\_l(class),'evrything\_else'};

end

h(x)=1(s\*x\_f<=s\*lambda)

%h(x)= 1(x\_f>=lambda)

function info=PNC\_train1(name\_of\_file);

S=load([name\_of\_file,'.mat']);

info.feature=ceil(rand(1)\*S.metadata.F);

Mx=max(S.X(:,S.info.feature));

Mn=min(S.X(:,S.info.feature));

info.threshold=rand(1)\*(Mx-Mn)+Mn;

info.sign=2\*(rand(1)>0.5) -1;

end

function y=PNC\_predict1(info,x)

y=(info.sign\*x(info.feature)<=info.sign(info.theshold));

end

function score=PNC\_score(info,K)

score.confusion\_matrix=zeros(2);

score.OA=0;

score.recall=0;

score.precision=0;

for k=1:K

S=load('training.mat');

confusion\_matrix=zeros(2);

for n=1:size(S.X,1)

x=S.X(n,:);

y\_hat=PNC\_predict1(info,x);

y=S.Y(n);

i=-y\_hat+2; j=-y+2;

confusion\_matrix(i,j)=confusion\_matrix(i,j)+1;

OA=sum(diag(confusion\_matrix))/sum(sum(confusion\_matrix)); %OA=(CM(1,1)+CM(2,2))/sum(sum(CM))

recall=confusion\_matrix(1,1)/sum(confusion\_matrix(:,1));

precision=confusion\_matrix(1,1)/ /sum(confusion\_matrix(1,:));

end

score.confusion\_matrix= score.confusion\_matrix+confusion\_matrix;

score.OA=score.OA+OA;

score.recall=score.recall+recall;

score.precision=score.precision+precision;

end

score.confusion\_matrix= score.confusion\_matrix/K;

score.OA=score.OA/K;

score.recall=score.recall/K;

score.precision=score.precision/K;

end

[NG,XG,N1,X1,N0,X0]=PNC\_show\_hist(name\_of\_file,feature\_number)

end

figure(2); plot(X1,N1/sum(N1),'g',X0,N0/sum(N0),'b',XG,NG/sum(NG),'r');

X

1.3

16

3

12

13

info=PNC\_train2(name\_of\_file)

exp\_n=5000;

info\_best=[];

OA\_best=-1;

for exp=1:exp\_n

info=PNC\_train1(name\_of\_file);

score=PNC\_score(info,K)

OA=score.OA;

if OA>OA\_best

info\_best=info;

OA\_best=OA;

disp(['OA=',num2str(OA),' exp=',num2str(exp)]),

end

end

info=info\_best;

end

Matlab simulation

X=rand(10,1)

figure(1); plot(X)

X=rand(50,1)

figure(1); plot(X)

figure(1); plot(sort(X))

[N,X]=hist(X);

figure(2); plot(X,N)

figure(2); plot(X,N/sum(N))

X=rand(500,1)

figure(1); plot(sort(X))

[N,X]=hist(X);

figure(2); plot(X,N/sum(N))

figure(2); plot(X,N/sum(N)); axis([-Inf Inf 0 1])

figure(2); bar(X,N/sum(N)); axis([-Inf Inf 0 1])

Y=(rand(500,1)<0.5);

X=10\*Y+10\*randn(500,1);

figure(1); plot(sort(X))

[N,X]=hist(X); figure(2); plot(X,N/sum(N));

figure(2); bar(X,N/sum(N)); axis([-Inf Inf 0 1])

[N0,X0]=hist(X(Y==0)); figure(2); plot(X0,N0/sum(N0));

X=10\*Y+10\*randn(500,1);

[N0,X0]=hist(X(Y==0)); figure(2); plot(X0,N0/sum(N0));

[N1,X1]=hist(X(Y==1)); figure(2); plot((X0,N0/sum(N0),X1,N1/sum(N1)));

[N1,X1]=hist(X(Y==1)); figure(2); plot(X0,N0/sum(N0),X1,N1/sum(N1));

[N,X\_]=hist(X);

figure(2); plot(X0,N0/sum(N0),'b',X1,N1/sum(N1),'g',X\_,N/sum(N),'r');

Mx=max(X);

Mn=min(X)

Mx

BW=(Mx-Mn)/50;

BW

x\_l=Mn+BW\*(0:K);

K+50;

K=50;

x\_l=Mn+BW\*(0:K);

x\_l

for k=1:(length(x\_l)-1)

h(k)=sum((X>=x\_l(k)).\*(X<=x\_l(k+1)))/N;

end

size(X)

N

N=size(X,1);

for k=1:(length(x\_l)-1)

h(k)=sum((X>=x\_l(k)).\*(X<=x\_l(k+1)))/N;

end

h

x\_tilde=(x\_k(1:end-1)+x\_k(2:end))/2;

x\_tilde=(x\_l(1:end-1)+x\_l(2:end))/2;

figure(3); plot(x\_tilde,h)