**Source code of RF.**

%% RF

%% clear

clear all

clc

%%input data

load('C:\Users\Administrator\Desktop\SVM.mat')

data(1:104,:)=unnamed(1:104,:);

wine=data(:,1:2);

wine\_labels=data(:,3);

P\_train = [wine(1:30,:);wine(55:85,:)];

T\_train = [wine\_labels(1:30);wine\_labels(55:85)];

P\_test = [wine(31:54,:);wine(86:104,:)];

T\_test = [wine\_labels(31:54);wine\_labels(86:104)];

model = classRF\_train(P\_train,T\_train,1000);

[T\_sim,votes] = classRF\_predict(P\_test,model);

[T\_sim\_train,votes\_train] = classRF\_predict(P\_train,model);

total\_0 = length(find(data(:,3) == 0));

total\_1 = length(find(data(:,3) == 1));

count\_0 = length(find(T\_train == 0));

count\_1 = length(find(T\_train == 1));

number\_0 = length(find(T\_test == 0));

number\_1 = length(find(T\_test == 1));

number\_0\_0\_train = length(find(T\_sim\_train == 0 & T\_train == 0));

number\_1\_1\_train = length(find(T\_sim\_train == 1 & T\_train == 1));

number\_0\_1\_train = length(find(T\_sim\_train == 1 & T\_train == 0));

number\_1\_0\_train = length(find(T\_sim\_train == 0 & T\_train == 1));

number\_0\_0\_sim = length(find(T\_sim == 0 & T\_test == 0));

number\_1\_1\_sim = length(find(T\_sim == 1 & T\_test == 1));

number\_0\_1\_sim = length(find(T\_sim == 1 & T\_test == 0));

number\_1\_0\_sim = length(find(T\_sim == 0 & T\_test == 1));

disp(['total：' num2str(104)...

' 1：' num2str(total\_0)...

' 2：' num2str(total\_1)]);

disp(['total training set：' num2str(61)...

' 1：' num2str(count\_0)...

' 2：' num2str(count\_1)]);

disp(['total test set：' num2str(43)...

' 1：' num2str(number\_0)...

' 2：' num2str(number\_1)]);

disp(['test set：']);

disp(['1：' num2str(number\_0\_0\_sim)...

' error：' num2str(number\_0 - number\_0\_0\_sim)...

' correct p1=' num2str(number\_0\_0\_sim/number\_0\*100) '%']);

disp(['2：' num2str(number\_1\_1\_sim)...

' error：' num2str(number\_1 - number\_1\_1\_sim)...

' correct p3=' num2str(number\_1\_1\_sim/number\_1\*100) '%']);

disp(['total test set：' num2str(43)...

' correct p=' num2str((number\_0\_0\_sim+number\_1\_1\_sim)/(number\_0+number\_1)\*100) '%']);

disp([' 0 to 1：' num2str(number\_0\_1\_sim)]);

disp([' 1 to 0：' num2str(number\_1\_0\_sim)]);

disp(['traing set：']);

disp(['1：' num2str(number\_0\_0\_train)...

' error：' num2str(count\_0 - number\_0\_0\_train)...

' correct p1=' num2str(number\_0\_0\_train/count\_0\*100) '%']);

disp(['2：' num2str(number\_1\_1\_train)...

' error：' num2str(count\_1 - number\_1\_1\_train)...

' correct p3=' num2str(number\_1\_1\_train/count\_1\*100) '%']);

disp(['total training set：' num2str(61)...

' correct p=' num2str((number\_0\_0\_train+number\_1\_1\_train)/(count\_0+count\_1)\*100) '%']);

disp([' 0 to 1：' num2str(number\_0\_1\_train)]);

disp([' 1 to 0：' num2str(number\_1\_0\_train)]);

%%

figure;

hold on;

plot(T\_test,'ko');

plot(T\_sim,'r.','LineWidth',3);

set(gca,'ytick',1:1:3);

grid on;

xlabel('Test set samples','fontweight','bold','fontsize',12);

ylabel('Category label','fontweight','bold','fontsize',12);

legend('The actual test set classification','Predicted test set classification');

title('Test set of the actual classification and prediction classification figure','fontweight','bold','FontSize',14);

set(gcf,'color','white');

% gtext('Accuracy = 91.7%(366/399)','fontweight','bold','fontsize',10);

% I=getframe(gcf);

% imwrite(I.cdata,'Random forsts classification result.tif','tiff','Resolution',600);

% leave-one-out cross validation

disp('leave-one-out cross validation:');

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

target=data(:,3);

indices=[1:1:104]';

for k=1:104

test=(indices==k);

train=~test;

train\_data=dataNew(train,:);

train\_target=target(train,:);

test\_data=dataNew(test,:);

test\_target=target(test,:);

model = classRF\_train(train\_data,train\_target,1000);

[T\_sim,votes] = classRF\_predict(test\_data,model);

number\_0 = length(find(test\_target == 0));

number\_1 = length(find(test\_target == 1));

number\_0\_sim = length(find(T\_sim == 0 & test\_target == 0));

number\_1\_sim = length(find(T\_sim == 1 & test\_target == 1));

disp(['correct=' num2str((number\_0\_sim+number\_1\_sim)/(number\_0+number\_1)\*100) '%']);

number\_index(k)=number\_0+number\_1;

number\_sim\_index(k)=number\_0\_sim+number\_1\_sim;

total\_index(k)=(number\_0\_sim+number\_1\_sim)/(number\_0+number\_1)\*100;

result\_classificition(k)=T\_sim;

end

Average\_Cross\_Validation\_Accuracy=sum(number\_sim\_index(:))/sum(number\_index(:))\*100

result\_classificition=result\_classificition';

total\_0 = length(find(data(:,3) == 0));

total\_1 = length(find(data(:,3) == 1));

number\_0 = length(find(wine\_labels == 0));

number\_1 = length(find(wine\_labels == 1));

number\_0\_0\_cross = length(find(result\_classificition == 0 & wine\_labels == 0));

number\_1\_1\_cross = length(find(result\_classificition == 1 & wine\_labels == 1));

number\_0\_1\_cross = length(find(result\_classificition == 1 & wine\_labels == 0));

number\_1\_0\_cross = length(find(result\_classificition == 0 & wine\_labels == 1));

disp(['total：' num2str(104)...

' 1：' num2str(total\_0)...

' 2：' num2str(total\_1)]);

disp(['1：' num2str(number\_0\_0\_cross)...

' error：' num2str(number\_0 - number\_0\_0\_cross)...

' correct p1=' num2str(number\_0\_0\_cross/number\_0\*100) '%']);

disp(['2：' num2str(number\_1\_1\_cross)...

' error：' num2str(number\_1 - number\_1\_1\_cross)...

' correct p2=' num2str(number\_1\_1\_cross/number\_1\*100) '%']);

disp(['total：' num2str(104)...

' correct p=' num2str((number\_0\_0\_cross+number\_1\_1\_cross)/(number\_0+number\_1)\*100) '%']);

disp([' 0 to 1：' num2str(number\_0\_1\_cross)]);

disp([' 1 to 0：' num2str(number\_1\_0\_cross)]);

Accuracy = zeros(1,20);

for i = 50:50:1000

%i

accuracy = zeros(1,100);

for k = 1:100

model = classRF\_train(P\_train,T\_train,i);

T\_sim = classRF\_predict(P\_test,model);

accuracy(k) = length(find(T\_sim == T\_test)) / length(T\_test);

end

Accuracy(i/50) = mean(accuracy);

end

figure

plot(50:50:1000,Accuracy);

xlabel('The number of tree in random forests','fontweight','bold','fontsize',12);

ylabel('The test set classification','fontweight','bold','fontsize',12);

title('The influence of the tree number in random forests','fontweight','bold','FontSize',14);

set(gcf,'color','white');

gtext('');

I=getframe(gcf);

imwrite(I.cdata,'influence.tif','tiff','Resolution',600);