CSE 5524 HW10

**Problem 1**

Not sure what does this problem aim to do…

Have attached the NCC code from hw7. Hopefully can receive partial points.

**Problem 2**

When K = 1, the accuracy is 0.9677. The predicted label is showing below with black circle for incorrectly classified points:

Chart, scatter chart

Description automatically generated

When K = 5, the accuracy is 0.9633. The predicted label is showing below with black circle for incorrectly classified points:

Chart, scatter chart

Description automatically generated

When K = 11, the accuracy is 0.9627. The predicted label is showing below with black circle for incorrectly classified points:

Chart, scatter chart

Description automatically generated

When K = 16, the accuracy is 0.9620. The predicted label is showing below with black circle for incorrectly classified points:

Chart, scatter chart

Description automatically generated

Observations: As we tune the K, the skill of the model does not go up when K increases. However, the misclassified points are different under different hyper parameters due to the distribution of the points in boundaries.

%% Problem 1

iml = imread('left.png');

imr = imread('right.png');

meanT = computeMean(template);

stdT = computeStd(template);

%% compute the NCC score

for i = 1:3

for r = 24:277

for c = 35:366

P = search(r-23:r+23,c-34:c+34,:);

T = template;

meanP = computeMean(P);

stdP = computeStd(P);

val = 0;

for x = 1:47

for y = 1:69

val = val+ ((P(x,y,i)-meanP(i))\*(T(x,y,i)-meanT(i))/(stdP(i)\*stdT(i)));

end

end

val = val/(47\*69-1);

scores(r,c,i) = val;

end

end

end

%% best match

scores = mean(scores,3);

k1 = max(max(scores));

[a,b] = find(scores ==k1);

imagesc(search(a-23:a+23,b-34:b+34,:));

%% plot the NCC scores

ls = reshape(scores,1,[]);

ls = sort(ls,'descend');

plot(ls,'-r');

xlabel('k');

ylabel('NCC');

%%

[a,b]= find(scores == ls(400));

imagesc(search(a-23:a+23,b-34:b+34,:));

%% helper function

% compute the mean of each channel in the image

function result = computeMean(image)

meanR = mean(image(:,:,1),'all');

meanG = mean(image(:,:,2),'all');

meanB = mean(image(:,:,3),'all');

result = [meanR,meanG,meanB];

end

function result = computeStd(image)

stdR = std(image(:,:,1),0,'all');

stdG = std(image(:,:,2),0,'all');

stdB = std(image(:,:,3),0,'all');

result = [stdR,stdG,stdB];

end

%% Problem 2 K = 1

X = readmatrix('train.txt');

Y = readmatrix('test.txt');

Idx = knnsearch(X(:,1:2),Y(:,1:2),'K',1,'Distance','euclidean');

predict = X(Idx,:);

[id,acc] = accuracy(predict(:,3),Y(:,3));

acc

Ywrong = Y(id,:);

hold on

plot(Ywrong(:,1),Ywrong(:,2),'ko');

% points in test.txt with predicted label

Z = [Y(:,1:2),predict(:,3)];

x1 = Z(:,1);

x2 = Z(:,2);

class = Z(:,3);

plot(x1(class == 1),x2(class == 1),'r.');

plot(x1(class == 2),x2(class == 2),'b.');

hold off;

%% Problem 2 K = 5

K = 16;

X = readmatrix('train.txt');

Y = readmatrix('test.txt');

Idx = knnsearch(X(:,1:2),Y(:,1:2),'K',K,'Distance','euclidean');

nb = [];

for i = 1:3000

nb = [nb;X(Idx(i,:),:)];

end

label = zeros(3000,1);

for i = 1:3000

label(i) = vote(nb(K\*(i-1)+1:K\*i,3),K);

end

[id,acc] = accuracy(label,Y(:,3));

acc

Z = [Y(:,1:2),label];

x1 = Z(:,1);

x2 = Z(:,2);

class = Z(:,3);

Ywrong = Y(id,:);

hold on

plot(Ywrong(:,1),Ywrong(:,2),'ko');

plot(x1(class == 1),x2(class == 1),'r.');

plot(x1(class == 2),x2(class == 2),'b.');

hold off

%% helper functions

% return the label from the vote

function result = vote(A,K)

num1 = 0;

num2 = 0;

result = 2;

for i = 1:K

if A(i) == 1

num1 = num1 + 1;

else

num2 = num2 + 1;

end

end

if num1 > num2

result = 1;

end

end

function [id,acc] = accuracy(A,B)

num = 0;

id = [];

for i = 1: 3000

if A(i) == B(i)

num = num + 1;

else

id = [id;i];

end

end

acc = num/3000;

end