一．问题描述

现有 USPS 手写体 0 到 9 共十组数据，每一组数据中有 1100 个 256 维的样本，用 K 近邻分类和最近邻分类方法分出 10 个手写体数字，然后进行 10 倍交叉验证。

二．原理描述

1．最近邻法

最近邻以每个训练样本为一个子类，不同类的两个样本之间用最小距离作为分类准则，对于一个新样本，把它逐一与已知样本比较，找出距离新样本最近的已知样本，以该样本的类别作为新样本的类别。通过判断它到两类样本的距离来进行决策。

已知样本集 $S_N=\{(x_1,\theta_1),(x_2,\theta_2),\ldots,(x_N,\theta_N)\}$，其中 x_i 是样本 i 的特征向量，θ_i 是它对应的类别，设有 c 个类，即 $\theta_i\in\{1,2,\ldots,c\}$。定义两个样本之间的距离度量 $\delta(x_i,x_j)$，采用欧式距离 $\delta(x_i,x_j)=\|x_i-x_j\|$，对于未知样本 x，求 S_N 中与之距离最近的样本，设为 x'（对应的类别为 θ'），即

$$\delta(x,x')=\min_{j=1,\ldots,N}\delta(x,x_j) \qquad \cdots\cdots\cdots\cdots(1)$$

则将 x 决策为 θ' 类。这种决策方法称作最近邻决策。

2. K 近邻法

k 近邻算法可以表示为：设有 N 个已知样本分属于 c 个类 $w_i,i=1,\ldots,c$，考查新样本 x 在这些样本中的前 k 个近邻，设其中有 k_i 个属性属于 w_i 类，则 w_i 类的判别函数就是

$$g_i(x)=k_i,i=1,\ldots,c \qquad \cdots\cdots\cdots\cdots(2)$$

决策规则是

$$\text{若 } g_k(x)=\max_{i=1,\ldots,c}g_i(x),\text{则 } x\in w_k \qquad \cdots\cdots\cdots\cdots(3)$$

因此，最近邻算法可以看做 K=1 时的 K 近邻算法的特殊情况。

三．代码实现

最近邻

```
clear
load('usps');
data=double(data);
load('T');%T 是标签集
M=11000;
K=10;
index= crossvalind('Kfold',M, K);%十倍交叉验证
count = zeros(1,K);
accuracy = zeros(1,K);
for i=1:K
    test=(index==i);
    train=~test;
    data_train=data(train,:);
    data_test=data(test,:);
    trainT=T(train,:);
    testT=T(test,:);
    Dist = pdist2(data_test,data_train);%训练数据
    [dmin,id]=min(Dist');
    temp=trainT(id);
    %计算正确率：
    for j=1:(M/K)
        if(temp(j) == testT(j))
            count(i)=count(i)+1;
        end
    end
```

```
    accuracy(i) = count(i)/(M/K);
    disp(['第',num2str(i),'次正确率为',num2str(accuracy(i))]);
end
Average = mean(accuracy);
disp(['平均正确率为',num2str(Average)]);
```

```
第 1 次正确率为 0.94
第 2 次正确率为 0.96182
第 3 次正确率为 0.95727
第 4 次正确率为 0.96
第 5 次正确率为 0.96636
第 6 次正确率为 0.96091
第 7 次正确率为 0.96
第 8 次正确率为 0.96091
第 9 次正确率为 0.96364
第 10 次正确率为 0.95636
平均正确率为 0.95873
```

K 近邻

```
clear
load('usps');
data=double(data);
load('T');
M=11000;
N=10;
K=3;%设置 K 值为 3
%十倍交叉验证：
indices= crossvalind('Kfold',M, N);
count = zeros(1,N);
```

```
accuracy = zeros(1,N);
for i=1:N
    test=(indices==i);
    train=~test;
    dataTrain=data(train,:);
    dataTest=data(test,:);
    trainT=T(train,:);
    testT=T(test,:);
    %%训练数据：
    A = unique(trainT);
    %取矩阵 T 的不同元素构成的向量，其中 A 可能是行向量也可能是列向量。
    L=length(A);
    ST=size(dataTest,1);
    B = pdist2(dataTest,dataTrain);
    [~,id] = sort(B,2,'ascend');
    %对 B 每一行进行升序排序
    k = zeros(L,ST);
    for x=1:L
        if(ST==1)
            k(x) = sum(trainT(id(:,1:K))==A(x));
        else
            k(x,:) = sum(trainT(id(:,1:K))==A(x),2);
        end
    end
    [~,j] = max(k);
    temp = A(j);
    %计算正确率：
    for b=1:(M/N)
        if(temp(b) == testT(b))
            count(i) = count(i)+1;
```

```
            end
        end
        accuracy(i) = count(i)/(M/N);
        disp(['正确率为',num2str(accuracy(i))]);
end
average = mean(accuracy);
disp(['平均正确率为',num2str(average)]);
```

```
正确率为 0.95727
正确率为 0.95909
正确率为 0.95182
正确率为 0.95182
正确率为 0.96091
正确率为 0.95636
正确率为 0.95727
正确率为 0.96
正确率为 0.95091
正确率为 0.95636
平均正确率为 0.95618
```