## 第一个代码

1. **from** numpy **import** \* #导入numpy科学计算包
2. **import** operator     #导入运算符模块
4. #加载数据的方法，返回样本数据（每一行是一个样本）和样本标签
5. **def** createDataSet():
6. group = array([[90,100],[88,90],[85,95],[10,20],[30,40],[50,30]])    #样本点数据
7. labels = ['A','A','A','D','D','D'] #样本标签
8. **return** group,labels#返回样本数据和标签
10. #分类方法  传入的dataset需是array数组
11. **def** classify0(inX, dataSet, labels, k):    #inX为输入样本，例如[85,90]，dataset是样本数据，labels是样本标签，k
12. dataSetSize = dataSet.shape[0]              #求出输入数据矩阵的行数（样本个数）
13. diffMat = tile(inX, (dataSetSize,1)) - dataSet    #求矩阵差,tile(inX, (dataSetSize,1))将inX沿y轴复制dataSetSize倍
14. sqDiffMat = diffMat \*\* 2
15. sqDistance = sqDiffMat.sum(axis = 1)          #平方和
16. distance = sqDistance \*\* 0.5               #测试样本点距离每个样本点的距离
17. sortedDistance = distance.argsort()          #将距离按升序排列
19. classCount = {}
20. **for** i **in** range(k):
21. voteLabel = labels[sortedDistance[i]]      #遍历前k个样本的标签
22. classCount[voteLabel] = classCount.get(voteLabel,0) + 1  #对标签进行计数，即每一类出现的次数
23. sortedClassCount = sorted(classCount.items(),key = operator.itemgetter(1),reverse = True)  #将计数后的标签按降序进行排列，得到元组列表
24. **return** sortedClassCount[0][0]
26. group,labels=createDataSet()
27. **print** (group)
28. **print** (labels)
29. g=classify0([0,0],group,labels,3)#样本[0,0],k=3，判断测试样本标签
30. **print** (g)

这是一个python写的KNN算法（K最近邻分类算法），它是机器学习十大算法之一，也是其中最最简单的一个算法。

这是我学到的第一个关于机器学习的算法，对我而言也是很有意义的一个代码，因为我觉得机器学习还蛮厉害的样子，也有点感兴趣，也希望能对这个领域有所了解。

因此我觉得这一个代码很值得去写，这是机器学习的入门。

## 第二个代码

1. #include<stdio.h>
2. **int** main()
3. {
4. printf("hello world");
5. **return** 0;
6. }

这个代码很简单，但是很多人学习计算机都是从这一个程序开始的，所以对于我们也是很有意义的。“hello world”也代表这我们对于世界的好奇，正是这一份好奇才能让我们去探索与发现。