

# 浙江大学

## 本科实验报告

k-近邻算法

课程名称： 人工智能实验

---

姓 名： 姚桂涛

---

学 院： 信息与工程学院

---

专 业： 信息工程

---

学 号： 3190105597

---

指导老师： 胡浩基、魏准

---

2022 年 3 月 11 日

## 一、 实验题目

### 1. kNN 代码实现-AB 分类

采用测量不同特征值之间的距离方法进行分类, 用所给的函数创建具有两个特征与一个标签类型的数据作为训练集, 编写 classify0 函数对所给的数据进行 AB 分类。

### 2. k-近邻算法改进约会网站的配对效果

k-近邻算法改进约会网站的配对效果

通过收集的一些约会网站的数据信息, 对匹配对象的归类: 不喜欢的人、魅力一般的人、极具魅力的人。

数据中包含了 3 种特征:

每年获得的飞行常客里程数、玩视频游戏所耗时间百分比、每周消费的冰淇淋公升数

## 二、 实验代码

### 1. kNN 代码实现-AB 分类

kNN

```
1 from http.client import ImproperConnectionState
2 from numpy import *
3 from collections import Counter
4 import operator
5 def createDataSet():
6     group = array([[1.0, 1.1], [1.0, 1.0], [0, 0], [0, 0.1]])
7     labels = ['A', 'A', 'B', 'B']
8     return group, labels
9
10 def classify0(inX, group, labels, k = 3):
11     res1 = (inX - group)**2
12     dist = res1[:,0] + res1[:,1]
13     dic = argsort(dist)
14     dic = dic[0:k:1]
15     newdic = []
16     for i in range(k):
17         newdic.append(labels[dic[i]])
18     c = Counter(newdic).most_common(1)
19     return c[0][0]
```

tests

```
1 import kNN
2 group, lables = kNN.createDataSet()
3 print('分类结果')
4 print('[0, 0] %c' %(kNN.classify0([0, 0], group, lables, k = 3)))
5 print('[0.8, 0.7] %c' %(kNN.classify0([0.8, 0.7], group, lables, k = 3)))
```

## 2. k-近邻算法改进约会网站的配对效果

```
1 import pandas as pd
2 import kNN
3 from sklearn.model_selection import train_test_split
4
5 df = pd.read_table('datingTestSet2.txt', sep='\s+', names = ['A', 'B', 'C', 'Y'])
6 # 对特征进行归一化处理
7 df2 = df.iloc[:, :3]
8 df2 = (df2 - df2.mean()) / df2.std()
9 label = df.iloc[:, 3:4]
10 df2.loc[:, 'Y'] = label
11 # 对数据集进行测试集和训练集划分, 90%作为训练集, 10%作为测试集
12 X_train, X_test, Y_train, Y_test = train_test_split(df2.iloc[:, :3], df2.Y, train_size
    =.90)
13 # 将DataFrame格式转化为numpy格式处理
14 group = X_train.values
15 label = Y_train.values
16 length = len(X_test)
17 X_test.iloc[0:1, :]
18 # res以储存测试结果
19 res = []
20 # 设置错误正确数count以计算正确率
21 Tnum = 0
22 Fnum = 0
23 for i in range(length):
24     inX = X_test.iloc[i:i+1, :].values
25     res.append(kNN.classify0(inX, group, label, k = 3))
26     if(kNN.classify0(inX, group, label, k = 3) == Y_test.values[i]):
27         Tnum += 1
28     else:
29         Fnum += 1
30 res1 = pd.DataFrame(data = res, columns=['TestResult'])
31 Y_test.reset_index(inplace=True, drop=True)
32 res1.loc[:, 'OriginTest'] = Y_test
33
34 print('前20个数据测试结果和原数据比较')
35 print('-----')
36 print(res1.head(20))
37 print('-----')
38 print('正确率%.2f%%' % (100 * Tnum / (Tnum + Fnum)))
```

## 三、实验结果及分析

### 1. kNN 代码实现-AB 分类

```
1 分类结果
2 [0, 0] B
3 [0.8, 0.7] A
```

## 2. k-近邻算法改进约会网站的配对效果

```
1  前20个数据测试结果和原数据比较
2  -----
3      TestResult OriginTest
4      0          2          2
5      1          3          3
6      2          1          3
7      3          2          2
8      4          2          2
9      5          3          3
10     6          3          3
11     7          2          2
12     8          1          1
13     9          1          1
14    10          1          1
15    11          3          3
16    12          2          2
17    13          2          2
18    14          1          1
19    15          2          2
20    16          1          1
21    17          2          2
22    18          1          1
23    19          3          3
24  -----
25  正确率97.00%
```

从实验结果可以看出,通过 k-近邻算法改进后的约会网站的配对效果比较显著,多次随机划分测试集和训练集后发现正确率基本可以达到 90% 以上。