**华中科技大学计算机科学与技术学院**

**机器学习报告**



专 业： 计算机科学与技术

班 级： CS1803

学 号： U201814792

姓 名： 宋家垚

成 绩：

指导教师： 黄宏

**完成日期： 2020年 6月 3日**

# 实验一

## 实验题目：KNN（K近邻）

## 实验要求

自己动手实现KNN算法。语言限定为Python。数据集使用MNIST数据集。结果展示方式：

* 输入若干测试图片，输出对应每张图片k近邻的图片。
* 绘制knn算法的训练misclassification rate曲线，并做出分析。
* 可以自由发挥（人脸数据集等）

最终结果（算法测试及训练曲线）汇总在一个文档中即可。

## 算法设计

KNN的算法思想：取k个在样本空间中和测试样本“距离”最近的已知标签的样本，这k个样本中哪个标签的数量最多，则把该标签赋给测试样本。其中，“距离”可以是欧氏距离或曼哈顿距离，k可以自己输入。

首先将训练数据集读入内存，然后读取测试样例，并计算其和每一个训练数据的“距离”，计算完成后从小到大进行排序，对前k个训练数据的标签进行统计，取出个数最多的标签赋给测试样例，分类结束。

## 实验环境与平台

操作系统：MacOS Catalina 10.15.5

编辑器：VSCode 1.45.1

Python版本：3.7.7

Numpy版本：1.18.4

## 程序实现

* 1. 首先编写文件读取相关函数，需要用到标准库OS、Struct，第三方库numpy，函数具有两个参数，第一个参数path决定文件读取路径，第二个参数kind决定读取哪一个文件，初始值为’train’。返回值label和train为读入内存中的标签和图片，其中标签为1×60000的矩阵，图片为60000×784（28\*28）的矩阵。具体实现如图1所示。

电脑屏幕的截图

描述已自动生成

图1

* 1. 定义预测单个样本函数，输入参数有test测试数据，为1×784的向量，还有k，即KNN算法的参数，返回值为测试样例的标签。首先依次计算训练集和测试样例的“距离”并存放于一个列表中，然后排序切片得到k个邻居，再使用字典依次统计每个标签的数量，最终返回最多的标签。

手机的屏幕截图

描述已自动生成

图2

* 1. 进行主运行程序的编写，读入训练集并适当切片，在这里选用1000个测试用例做测试，并计算正确率，如图3所示。

屏幕上有字

描述已自动生成

图3

## 实验结果

表1

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| K | 1 | 3 | 5 | 7 | 9 |
| Accuracy | 68.500% | 73.900% | 76.800% | 78.000% | 79.000% |
| K | 11 | 13 | 15 | 17 | 19 |
| Accuracy | 79.300% | 79.200% | 80.100% | 79.500% | 79.000% |

## 结果分析

图4

整体分类准确度随k值增大而增大，刚开始增速明显，到后逐渐趋于平稳，原因可能是在k值较小的时候，误差较大，可供判断标签的数据具有一定的偶然性，导致准确度不高，k值增大到一定后继续增大，无关项增多，影响分类判断，导致准确度无法继续提升，故k值得选取比较随机，更需要依靠经验的判断，这也是knn算法的局限性所在，没有一个准确的标准或方法来确定一个合理的k值。

# 参考文献

[1]廖雪峰.Python教程[EB/OL].https://www.liaoxuefeng.com/wiki/1016959663602400

[2]liuchengxu\_.详解MNIST数据集[EB/OL]. https://blog.csdn.net/simple\_the\_best/article/details/75267863

[3] Jeremy.自己动手实现KNN分类器——识别MNIST手写数据[EB/OL]. http://blog.hitaoxy.cn/2018/10/11/%E8%87%AA%E5%B7%B1%E5%8A%A8%E6%89%8B%E5%AE%9E%E7%8E%B0KNN%E5%88%86%E7%B1%BB%E5%99%A8%E2%80%94%E2%80%94%E8%AF%86%E5%88%ABMNIST%E6%89%8B%E5%86%99%E6%95%B0%E6%8D%AE/