

# 基础训练 1: 最近邻/K 近邻算法

滕明卓

2021 年 3 月 12 日

## 1 算法简介

### 1.1 算法原理

k 近邻 (k-Nearest Neighbor, 简称 kNN) 学习是一种常用的监督学习方法, 其工作机制非常简单: 给定测试样本, 基于某种距离度量找出训练集中与其最靠近的  $k$  个训练样本, 然后基于这  $k$  个“邻居”的信息来进行预测。

k 近邻算法可以用来解决分类和回归问题。

### 1.2 算法过程

1. 读取训练数据和测试数据
2. 计算测试样本与训练样本之间的距离
3. 选取距离最近的  $k$  个训练样本, 用投票法 (分类) 或平均法 (回归) 计算预测结果

### 1.3 特点

- 懒惰学习: 在训练阶段仅仅是把样本保存起来, 训练时间开销为零, 待收到测试样本后再进行处理。
- 泛化性能: 在训练样本的采样密度足够大的情况下, 它的泛化错误率不超过贝叶斯最优分类器的错误率的两倍。但是实际中, 尤其是在高维情形下, 样本是很稀疏的。

数据集	样本数	任务	属性数	属性类型
iris	150	多分类	4	实数
adult	48842	二分类	14	类别、整数
forest fire	517	回归	13	实数

表 1: 数据集

- k 的选取: k 如果过小, 会很容易学到噪声, 造成过拟合; 反之, k 过大容易造成欠拟合。所以选取合适的 k 很重要, 通常使用验证集选取合适的 k 值。

## 2 数据集

选用 3 个数据集进行实验, 数据集概览见表 1。实验中以固定随机数种子按 6:4 随机划分训练集和测试集。

## 3 程序实现

程序实现了 k 近邻分类和 k 近邻回归算法。以分类器为例, 函数接口如下:

```
KNNClassifier(self, n_neighbors=5, weights='uniform',
               algorithm='brute', metric='euclidean')
```

程序实现了如下功能:

1. 设置不同的邻居个数 k。
2. 加权投票、加权平均。距离越近, 权重越大。默认权值是相同的。
3. 数据结构改进, 支持线性表和 KD 树, 便于找出最近的 k 个邻居。
4. 距离度量可以传入自定义函数, 默认为欧氏距离。

此外, 对于数据集有进行如下处理:

1. 离散属性处理: 对于属性值为类别的属性, 转化为 one-hot 编码, 然后计算距离。
2. 属性标准化: 将各个属性值标准化, 避免某个属性影响过大。

## 4 库函数

sklearn 中, KNeighborsClassifier 和 KNeighborsRegressor 两个函数是 k 近邻分类器。

经检验, 库函数和自己实现的程序预测结果几乎相同。