KNN 算法实验报告

一、 算法模型

1. 算法简介

K 邻近（k-Nearest Neighbor，KNN）分类算法是一种基本分类与回归方法。其主要思想可以通过“近朱者赤，近墨者黑”来解释，即你的类别可以由你的邻居来推断出来。先计算待分类样本与已知类别的训练样本之间的距离，找到距离与待分类样本数据最近的 k 个邻居；再根据这些邻居所属的类别来判断待分类样本数据的类别。

2. 算法模型

K 近邻法使用的模型实际上就是对应于特征空间的划分，其中模型主要的要素是距离度量，k 值的选择，分类决策规则决定。

2.1 距离度量

特征空间的两个实例点的距离是两个实例点的相似程度来反映，通过空间两个点的距离类似的来度量，距离越大，两个点越不相似。距离的选择通常使用欧式距离。

对于距离的定义主要分为如下几种：

1) 欧式距离：

$$\mathrm{d}_{\mathrm{euc}}(x,y) = [\sum_{j=1}^{d}(x_j - y_j)^2]^{\frac{1}{2}} = [(x-y)(x-y)^T]^{\frac{1}{2}}$$

2) 曼哈顿距离：

$$\mathrm{d}_{\mathrm{man}}(x,y) = \sum_{i=1}^{d}|x_j - y_j|$$

3) 切比雪夫距离：

$$\mathrm{d}_{\mathrm{che}}(x,y) = \max_j(|x_j - y_j|)$$

4) 闵氏距离：r 取值为 2 时：曼哈顿距离；r 取值为 1 时：欧式距离。

$$d_{min}(x,y)=\left(\sum_{j=1}^{d}(x_j-y_j)^r\right)^{\frac{1}{r}},r\geq 1$$

5) 弦距离：

$\|\cdot\|_2$表示 2-范数，即$\|x\|_2=\sqrt{\sum_{j=1}^{d}x_j^2}$

$$d_{chord}(x,y)=\left(2-2\frac{\sum_{j=1}^{d}x_jy_j}{\|x\|_2\|y\|_2}\right)^{\frac{1}{2}}$$

2.2 K 值的选择

当 k 较小时，近似误差减小，估计误差增大， 易受噪声污染和过拟合；一般采用小的 K 值，再采用交叉验证法。

2.3 分类投票规则

投票决定 :少数服从多数，近邻中哪个类别的点最多就分为该类加权投票法： 根据距离的远近，对近邻的投票进行加权，距离越近则权重越大（权重为距离平 方的倒数）

2.4 优缺点

优点：

- 简单，易于理解，易于实现，无需估计参数，无需训练；
- 适合对稀有事件进行分类；
- 特别适合于多分类问题(multi-modal,对象具有多个类别标签)，kNN 比 SVM 的表现要好。

缺点：

- 懒惰算法，对测试样本分类时的计算量大，内存开销大，评分慢；

- 当样本不平衡时，如一个类的样本容量很大，而其他类样本容量很小时，有可能导致当输入一个新样本时，该样本的 K 个邻居中大容量类的样本占多数；
- 可解释性较差，无法给出决策树那样的规则。

2.5 常见问题

- K 值的设定

k 值选择过小，得到的近邻数过少，会降低分类精度，同时也会放大噪声数据的干扰；而如果 k 值选择过大，并且待分类样本属于训练集中包含数据数较少的类，那么在选择 k 个近邻的时候，实际上并不相似的数据亦被包含进来，造成噪声增加而导致分类效果的降低。

如何选取恰当的 K 值也成为 KNN 的研究热点。k 值通常是采用交叉检验来确定（以 k=1 为基准）。经验规则：k 一般低于训练样本数的平方根。

- 类别的判定方式

投票法没有考虑近邻的距离的远近，距离更近的近邻也许更应该决定最终的分类，所以加权投票法更恰当一些。

- 距离度量方式的选择

高维度对距离衡量的影响：众所周知当变量数越多，欧式距离的区分能力就越差。变量值域对距离的影响：值域越大的变量常常会在距离计算中占据主导作用，因此应先对变量进行标准化。

- 训练样本的参考原则

学者们对于训练样本的选择进行研究，以达到减少计算的目

的，这些算法大致可分为两类。第一类,减少训练集的大小。KNN 算法存储的样本数据,这些样本数据包含了大量冗余数据,这些冗余的数据增了存储的开销和计算代价。缩小训练样本的方法有:在原有的样本中删掉一部分与分类相关不大的样本样本,将剩下的样本作为新的训练样本;或在原来的训练样本集中选取一些代表样本作为新的训练样本；或通过聚类,将聚类所产生的中心点作为新的训练样本。

在训练集中，有些样本可能是更值得依赖的。可以给不同的样本施加不同的权重，加强依赖样本的权重，降低不可信赖样本的影响。

- 性能问题

KNN 是一种懒惰算法，而懒惰的后果：构造模型很简单，但在对测试样本分类地的系统开销大，因为要扫描全部训练样本并计算距离。

二、 算法流程

1. 算距离：给定测试对象，计算与训练集中的每个对象的距离；
2. 找邻居：圈定距离最近的 K 个训练对象，作为测试对象的近邻；
3. 做分类：根据这 K 个近邻归属的主要类别，来对测试对象分类。

三、 实验流程

- 处理数据集，在数据集上进行训练集其中数据的加载
- 进行距离的计算，对于每个数据进行数据的提取，进行各个数据点之间的计算，这里面选用欧拉距离

- 对于其中的 K 的值选取 5
- 加载数据集，在训练集训练后的进行相应的距离大小进行分类
- 确定各个预测值的分类