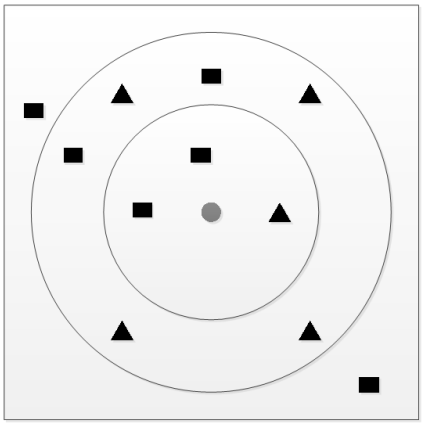
KNN 算法

1. 原理：

根据测试数据与每一条训练数据的距离，选择前 K 个邻近的训练数据并统计k中各个类别的数目，选择类别最多的作为该测试数据所属的类别；



该算法中k的取值尤为关键：

（a）一般而言，在数据样本充足且分布均匀的情况下，k值越大，划分类别相对越正确；当K=1时，其抗干扰能力就较差。由于假如样本中出现了某种偶然的类别，那么新的数据非常有可能被分错。为了添加分类的可靠性，能够考察待测数据的K个近期邻样本 。统计这K个近邻样本中属于哪一类别的样本最多，就将样本X判属于该类。

(b) 假设在样本有限的情况下，KNN算法的误判概率和距离的详细测度方法就有了直接关系。即用何种方式判定哪些数据与新数据近邻。不同的样本选择不同的距离测量函数，这能够提高分类的正确率。通常情况下，KNN能够采用Euclidean（欧几里得）、Manhattan（曼哈顿）、Mahalanobis（马氏距离）等距离用于计算。

KNN伪代码：

```
Algorithm KNN(A[n],k,x)
  Input:
    A[n]为N个训练样本的特征，K为近邻数，x为新的样本；
  Initialize:
    取A[1]~A[k]作为x的初始近邻。
    计算测试样本与x间的欧式距离d(x,A[i]),i=1,2...,k;
    按d(x,A[i])升序排序。
    计算最远样本与x间距离D。即max{d(x,A[i])}；
  for(i=k+1;i<=n;i++)
    计算A[i]与x之间的距离d(x,A[i])；
    if （d(x,A[i])）<D then 用A[i]取代最远样本。
```

依照d(x,A[i])升序排序；

计算最远样本与x间的距离D，即max{d(x,A[i])}；

End for

计算前K个样本A[i],i=1,2...,k所属类别的概率。

具有最大概率的类别即为样本x的类；

Output:x所属的类别。

KNN 存在的不足：

1. 样本数量不均衡时，该样本的K个近邻中，大容量类的样本占多数，从而导致误分类。通常采用加权法，提高分类精度

2. 分类时须要先计算待分类样本和全体已知样本的距离。才干求得所需的K近邻点，计算量较大，尤其是样本数量较多时。

针对这样的情况能够事先对已知样本点进行剪辑。去除对分类作用不大的样本，这一处理步骤仅适用于样本容量较大的情况，假设在原始样本数量较少时採用这样的处理。反而会添加误分类的概率。

改进的knn算法：

组合分类器的KNN改进算法

首先随机选择属性子集。构建多个K近邻分类器；然后对未分类元组进行分类。最后把分类器的分类结果依照投票法进行组合，将得票最多的分类器作为终于组合近邻分类器的输出。

如何选择一个合适的K值？

选择一个**大的k会减少噪声数据对于模型的影响**,但它会使分类器产生**偏差。**`k值`的选取决定于要学习的概念的难度和训练数据中记录的数量.

1. 一种常见的方法是**从k等于训练集中案例数量的平方根开始.**比如训练集中有100个案例,则k可以从10开始进行筛选。

2.基于各种测试数据测试多个`k值`,并选择一个可以提供最好分类性能的`k值`.并且,除非数据的噪声非常大,否则大的训练集可以使`k值`的选择不那么重要.

3. 还有种方法是选择一个较大的`k值`,同时用一个**权重投票**`(weighted voting)`,在这个过程中,认为较近邻的投票比远的投票更有权威.

特征范围标准化：

在应用kNN算法之前通常需要将特征数据转换为**一个标准的范围内**,这个步骤的合理性高度依赖于特征是如何被度量的. 如果某个特征具有比其他特征更大范围的值,那么距离的度量就会强烈地被这个具有较大范围的特征所支配.

(1) min-max 标准化

标准化的方法便是搜索或者放大特征的范围来重新调整特征,使得每个特征对距离公式的贡献相对平均.传统的归一化方法是**min-max标准化**(min-max normalizatio).该过程变换特征,使被变换数据的所有值都落在**0~1**的范围内。

$$Xnew = (X-min(X))/(max(X)-min(X))$$

(2) z-score standardzation (z-分数标准化)

$$Xnew = (X - mean(X))/stdDev(X)$$

公式如下,含义是原始值减去特征X的均值后,再除以X的标准差.

z分数落在一个无界的负数和正数构成的范围内,与**min-max标准化**后的值不同,没有预定义的最小值和最大值.

用于kNN训练集的标准化方法也应该用于要待分类的测试集样本.但是对于**min-max标准化**,这可能就导致一种棘手的情形,即**测试集**的最小值或最大值可能在**训练集**的范围之外.比如**测试集**的一个特征的值为1, 3, 5, 7, 9。经**min-max标准化**后为0, 0.25, 0.5, 0.75, 1.00；但**训练集**中如果这个特征值存在0, 10等处于测试集范围之外的数值，如果也使用**min-max标准化**则导致预测结果偏离不准确。**所以，在测试集范围大于训练集范围的情况下，使用z分数标准化是种更好的方法。但前提是两者具有相同的均值和标准差。**

参考：

https://www.jianshu.com/p/3dcb39de04aa