一、算法概述

1、kNN算法又称为k近邻分类(k-nearest neighbor classification)算法。最简单平凡的分类器也许是那种死记硬背式的分类器，记住所有的训练数据，对于新的数据则直接和训练数据匹配，如果存在相同属性的训练数据，则直接用它的分类来作为新数据的分类。这种方式有一个明显的缺点，那就是很可能无法找到完全匹配的训练记录。

kNN算法则是从训练集中找到和新数据最接近的k条记录，然后根据他们的主要分类来决定新数据的类别。该算法涉及3个主要因素：训练集、距离或相似的衡量、k的大小。

2、代表论文

Discriminant Adaptive Nearest Neighbor Classification
Trevor Hastie and Rolbert Tibshirani
IEEE TRANSACTIONS ON PAITERN ANALYSIS AND MACHINE INTELLIGENCE, VOL. 18, NO. 6, JUNE 1996
http://www.stanford.edu/~hastie/Papers/dann_IEEE.pdf

3、行业应用

客户流失预测、欺诈侦测等（更适合于稀有事件的分类问题）

二、算法要点

1、指导思想

kNN算法的指导思想是“近朱者赤，近墨者黑”，由你的邻居来推断出你的类别。

计算步骤如下：

1）算距离：给定测试对象，计算它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

2、距离或相似度的衡量
什么是合适的距离衡量？距离越近应该意味着这两个点属于一个分类的可能性越大。
觉的距离衡量包括欧式距离、夹角余弦等。
对于文本分类来说，使用余弦(cosine)来计算相似度就比欧式(Euclidean)距离更合适。

3、类别的判定
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）

三、优缺点

1、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（例如当流失率很低时，比如低于0.5%，构造流失预测模型）
特别适合于多分类问题(multi-modal,对象具有多个类别标签)，例如根据基因特征来判断其功能分类，kNN比SVM的表现要好

2、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
可解释性较差，无法给出决策树那样的规则。

四、常见问题

1、k值设定为多大?

k太小，分类结果易受噪声点影响；k太大，近邻中又可能包含太多的其它类别的点。（对距离加权，可以降低k值设定的影响）
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

2、类别如何判定最合适?
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。

3、如何选择合适的距离衡量?
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。

4、训练样本是否要一视同仁?
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。

5、性能问题?
kNN是一种懒惰算法，平时不好好学习，考试（对测试样本分类）时才临阵磨枪（临时去找k个近邻）。
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。
已经有一些方法提高计算的效率，例如压缩训练样本量等。

6、能否大幅减少训练样本量，同时又保持分类精度?
浓缩技术(condensing)
编辑技术(editing)