# K–近邻算法

**同学们，早上好！**

咱们今天的助教导学即将开始，今天的主题是K–近邻算法，首先看一下今天的目录，我们今天的目录分为以下三点：

1、K–近邻算法的基本概念、核心思想

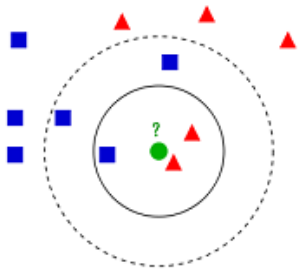
2、K–近邻算法的三要素：k值的选取、距离的度量、分类决策规则

3、k-近邻算法的一些个人总结

## 一.k-近邻算法的基本概念，原理以及应用

我们首先介绍以下k近邻的基本概念，让同学们对k近邻算法先有一个整体的把握。k近邻（k–nearest neighbor, k–NN）算法由Cover和Hart于1968年提出，是一种基本分类和回归方法。本篇文章只讨论分类问题的k近邻算法。我们首先叙述k近邻算法的基本概念，然后讨论k近邻算法的三要素，最后对k近邻算法做一个总结。

K近邻算法，通俗来说，就是给定一个训练数据集，对新的输入实例，在训练数据集中找到与该实例最邻近的K个实例，这K个实例的多数属于某个类，就把该输入实例分类到这个类中（类似于投票时少数服从多数的思想）。接下来我们来看下引自维基百科上的一幅图：




如上图所示，有两类不同的样本数据，分别用蓝色的小正方形和红色的小三角形表示，而图正中间的那个绿色的圆所标示的数据则是待分类的数据。这也就是我们的目的，来了一个新的数据点，我要得到它的类别是什么？下面我们根据k近邻的思想来给绿色圆点进行分类。

如果K=3，绿色圆点的最邻近的3个点是2个红色小三角形和1个蓝色小正方形，根据少数服从多数的思想，判定绿色的这个待分类点属于红色的三角形一类。如果K=5，绿色圆点最邻近的5个邻居是2个红色三角形和3个蓝色的正方形，根据少数服从多数的思想，判定绿色的这个待分类点属于蓝色的正方形一类。

上面的例子形象的展示了k近邻的算法思想，可以看出k近邻的算法思想非常简单。但是在上面的例子中有几个问题，不知道同学们有没有想到：

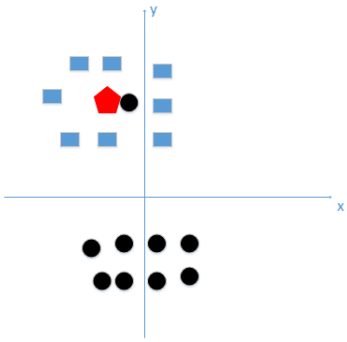
我们的K是怎么选取的，选多少合适？我们各个点之间的距离远近是如何度量的？判断输入样本属于哪一类时，少数服从多数思想背后的原因是什么？
上面三个问题就是我们k近邻算法的三个核心要素，别着急，接下来我们一一讨论。

## 二.k近邻算法三要素

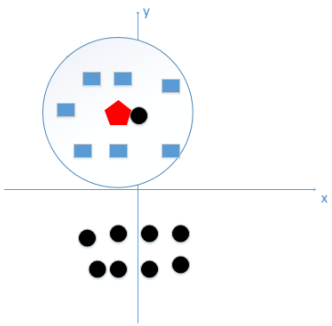
接下来我们讨论一下k近邻算法的三个基本要素。

### 2.1 k值的选取

首先我们来讨论k近邻算法的第一个核心要素：k值的选取
如果我们选取较小的k值，那么就会意味着我们的整体模型会变得复杂，容易发生过拟合！看完这个结论，是不是感觉有点困惑？我们通过具体的例子来讲解。
假设我们有训练数据和待分类点如下图：

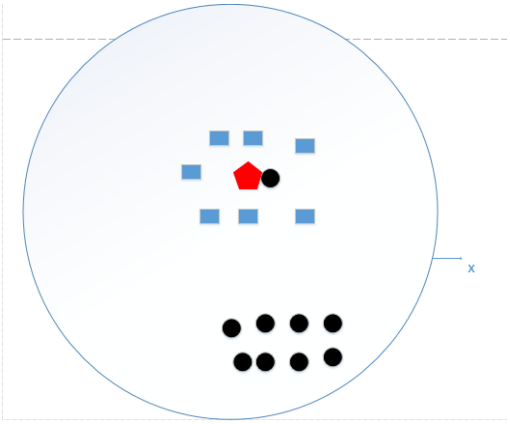



上图中有俩类，一个是黑色的圆点，一个是蓝色的长方形，现在我们的待分类点是红色的五边形。根据我们的k近邻算法步骤来决定待分类点应该归为哪一类。我们能够看出来五边形离黑色的圆点最近，k又等于1，因此我们最终判定待分类点是黑色的圆点。很明显我们这样分类是错误的，因为此时距离五边形最近的黑色圆点是一个噪声，因此如果我们选择的k太小了，比如上面k等于1，我们很容易学习到数据中的噪声，也就非常容易将待分类点判定为噪声类别，那么模型就太复杂了。在上图，如果，k大一点，k等于8，把长方形都包括进来，我们很容易得到我们正确的分类应该是蓝色的长方形！如下图：




经过上例，我们可以得到k太小会导致过拟合（过拟合就是在训练集上准确率非常高，而在测试集上准确率低），很容易将一些噪声（如上图距离五边形最近的黑色圆点）学习到模型中，而忽略了数据真实的分布！
如果我们选取较大的k值，就相当于用较大邻域中的训练数据进行预测，这时与输入实例较远的（不相似）训练实例也会对预测起作用，使预测发生错误，k值的增大意味着整体模型变得简单。为什么k值增大就意味着模型变得简单了？
我们假设，如果k=N（N为训练样本的个数）,那么无论输入实例是什么，都将简单地预测它属于在训练实例中最多的类。这时，模型是不是非常简单，这相当于你压根就没有

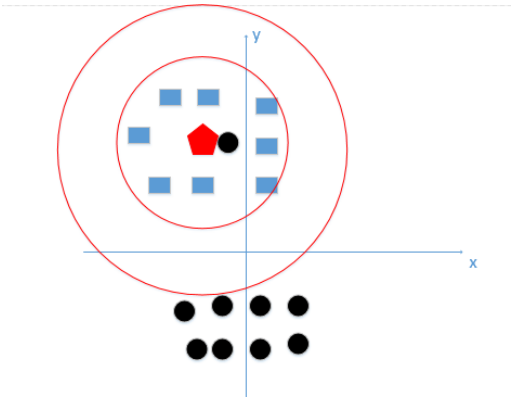
训练模型！直接拿训练数据统计了一下各个数据的类别，找最大的而已！这好像下图所示：




我们统计了黑色圆形是8个，长方形个数是7个，如果k=N，那么，红色五边形是属于黑色圆形的（很明显是错误的）。

这个时候，模型过于简单，完全忽略训练数据实例中的大量有用信息，是不可取的。

所以k值既不能过大，也不能过小，在我举的这个例子中，我们k值的选择，在下图红色圆边界之间这个范围是最好的，如下图：




（注：这里只是为了更好让大家理解，真实例子中不可能只有俩维特征，但是原理是一样的，我们就是想找到较好的k值大小）

那么我们一般怎么选取k值呢？李航老师的书上讲到（统计学习方法2），我们一般选取一个较小的数值，通常采取交叉验证法来选取最优的k值，也就是说，选取k值关键是实验调参，从这里也能看出来机器学习是一门实践学科，所以需要同学们多多写代码，理论与实践相结合才是最好的学习方式。

## 2.2 距离的度量

接下来我们讨论k近邻算法的第二个核心要素：距离的度量。

我们在上面的几个例子中，经常会使用到距离待分类样本最近的k个样本，那么这最近的k个样本是如何选取出来的呢？我们怎么知道哪个样本距离待测样本距离最近呢？

通常情况下我们使用欧氏距离作为距离的度量，欧式距离度量方式的公式如下所示：

设特征空间 $\chi$ 是 $n$ 维实数向量空间 $R^n$ ，$x_i, x_j \in \chi$ ，$x_i = (x_i^1, x_i^2, \ldots, x_i^n)^T$ ，$x_j = (x_j^1, x_j^2, \ldots, x_j^n)^T$ ，$x_i, x_j$ 的欧氏距离如下所示：

$$L_2(x_i, x_j) = ((x_i^1 - x_j^1)^2 + (x_i^2 - x_j^2)^2 + \ldots + (x_i^n - x_j^n)^2)^{1/2}$$

在实际应用中，距离函数的选择应该根据数据的特性和分析的需要而定，一般选取p=2欧式距离表示，这不是本文的重点。

## 2.3 分类决策规则

最后我们来讨论一下k近邻算法的第三个要素：分类决策规则

我们在上面几个例子中，判断待决策样本属于哪一类时，都是根据少数服从多数的思想。为什么根据这种思想做分类决策，背后的原理是什么呢?

假设分类的损失函数为0–1损失函数，分类函数为

$$f: R^n \to \{c_1, c_2, \ldots, c_K\}$$

$c_1, c_2, \ldots, c_K$ 是我们数据集的 $K$ 个类别，那么误分类的概率是：

$$P(Y \neq f(x)) = 1 - P(Y = f(x))$$

其中 $Y$ 是我们样本的真实类别， $f(x)$ 是我们算法预测的类别；对于给定的实例 $x \in X$ ,其最近邻的 $k$ 个训练实例点构成集合 $N_K(x)$ 。如果涵盖 $N_K(x)$ 的区域的类别是 $c_j$ ，那么误分类率是

$$1/k \sum_{x_i \in N_k(x)} I(y_i \neq c_j) = 1 - \frac{1}{k} \sum_{x_i \in N_k(x)} I(y_i = c_j)$$

其中， $y_i$ 是我们第 $i$ 个样本的所属类别，要使误分类率最小即经验风险最小，就要使 $\sum_{x_i \in N_k(x)} I(y_i = c_j)$ 最大，所以多数表决规则等价于经验风险最小化。

讲到这里，k近邻算法三个核心要素我们已经讲解完了，最后我们对本文做一下总结。

三.本文的一点总结

1.我们提出了k近邻算法，算法的核心思想是，给定一个训练数据集，对新的输入实例，在训练数据集中找到与该实例最邻近的K个实例，这K个实例的多数属于某个类，就把该输入实例分类到这个类中。更通俗的说一遍算法的过程，来了一个新的输入实例，我们算出该实例与每一个训练点的距离，然后找到前k个，这k个哪个类别数最多，我们就判断新的输入实例就是哪类！

2.其次我们对k近邻算法的核心三要素进行了讨论：如何选取k值（根据交叉验证，通过实验调参）、样本之间的距离如何度量（一般选取欧式距离）、分类决策的规则（多数表决规则等价于经验风险最小化）。

相信通过阅读本篇文章,我们能够对k近邻算法有一个比较清晰的了解。

## 参考文献：

[1]一文搞懂k近邻（k–NN）算法，文章链接：https://zhuanlan.zhihu.com/p/25994179

[2]统计学习方法2，李航