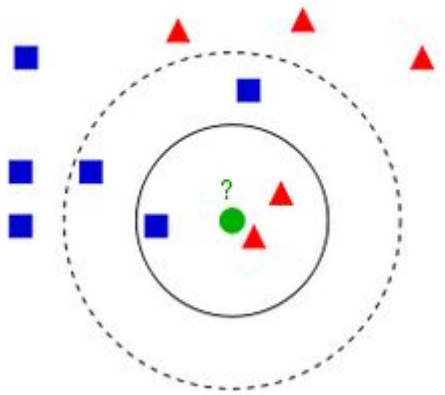
kNN 算法的一些补充

1、图示 kNN

下图是一个对于 kNN 算法解释最清楚的图，如下：



蓝色小方块和红色三角形均是已有分类的数据，也就是有 label（标签）的数据，可以看出来是监督学习算法。当前的任务是将绿色圆块进行分类判断，判断是属于蓝色块或者红三角。

当然这里的分类还跟 K 值是有关的：

如果 K=3（图中的实线圈），红三角占比 2/3 ，则判断为绿色圆块为红三角；

如果 K=5（图中的虚线圈），蓝色块占比 3/5 ，则判断为绿色圆块为蓝色块。

不知道大家发没发现一个问题，kNN 算法实际上根本就不用进行训练，而是直接进行计算的，训练的时间为 0，计算时间为训练集规模 n 。

2、基本要素

kNN 算法的基本要素大致有 3 个：

- k 值的选择
- 距离的度量
- 分类决策规则

相对应的使用方式：

- k 值会对算法的结果产生重大影响。k 值较小意味着只有与输入实例较近的训练实例才会对预测结果起作用，容易产生过拟合；如果 k 值较大，优点是可以减少学习的估计误差，缺点是学习的近似误差增大，这时与输入实例较远的训练实例也会对预测结果其作用，使预测发生错误。在实际应用中，k 值一般选择一个较小的数值，通常采用交叉验证的方法来选择一个最优的 k 值。
- 算法中的分类决策规则往往是多数表决，即由输入实例的 k 个最邻近的训练实例中的多数类决定输入实例的类别
- 距离度量一般采用 欧氏距离，在度量之前，应该将每个属性的值规范化，这样有助于防止具有较大初始值域的属性比具有较小初始值域的属性的权重过大。

3、kNN 算法用于回归

通过找出一个样本的 k 个最近邻居，将这些邻居的属性的平均值赋给该样本，就可以得到该样本的属性。更有用的方法是将不同距离的邻居对该样本产生的影响给予不同的权重（weight），如权值与距离成正比。

4、kNN 算法不足

- 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。因此可以采用权值的方法（和该样本距离小的邻居权值大）来改进。
- 另一个不足之处是计算量巨大，因为对每一个待分类的文本都要计算它到全体已知样本的距离，才能求得它的 k 个最近邻点。

一个常用的解决办法是，事先对已知样本点进行剪辑，事先去除对分类作用不大的样本。该算法比较适用于样本容量比较大的类域的自动分类，而那些样本容量较小的类域采用这种算法比较容易产生误分。

5、kd 树

由于我们上面说的 kNN 算法的不足之处，为了提高 kNN 算法搜索的速度，可以利用特殊的数据存储形式来减少计算距离的次数。kd 树就是一种以二叉树的形式存储数据的方法。

kd 树就是对 k 维空间的一个划分。构造 kd 树相当于不断用垂直于坐标轴的超平面将 k 维空间切分，构成一系列 k 维超矩阵区域。kd 树的每一个节点对应一个超矩阵区域。（kd树想要深入了解的同学可以参考 李航老师的 《统计学习方法》的 P41 ）

很生动的一个讲解（思路篇）：https://www.joinquant.com/post/2627

算法详解篇：https://zhuanlan.zhihu.com/p/23966698

参考资料：https://zhuanlan.zhihu.com/p/24405864

6、距离和相似度度量

- 一个很好的参考文章： https://zhuanlan.zhihu.com/p/27305237
- 一个简单的认识：
    - 闵可夫斯基距离，这个距离最常用的 p 是 2 和 1，前者是欧几里得距离（Euclidean distance），也就是我们常说的 欧氏距离，后者是曼哈顿距离（Manhattan distance）。当 p 趋近于无穷时，闵可夫斯基距离转化成 切比雪夫距离（Chebyshev distance）。闵可夫斯基距离比较直观，但是与数据的分布无关，具有一定的局限性，在数据各个维度不相关的情况下，如果 x 方向的幅度值远远大于 y 方向的幅度值，那么这个距离公式就会过度的放大 x 维度的作用。
    - 如果数据维度相互之间数据相关（比如，身高较高的信息很有可能会带来体重较重的信息，因为两者是有关联的），这时候就要用到 马氏距离（Mahalanobis distance）。