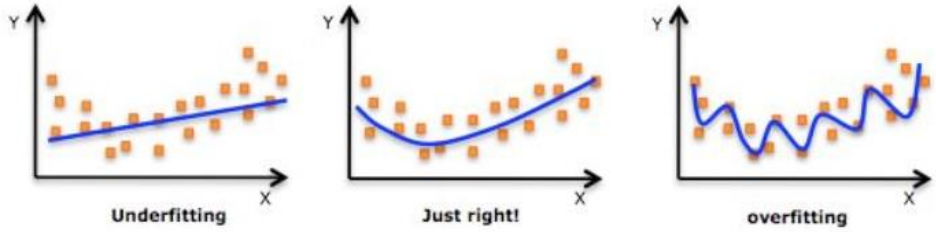
1. Overfitting 与 Underfitting 产生的原因



过拟合（overfitting）与欠拟合（underfitting）是统计学中的一组现象。过拟合是由于使用的参数过多而导致模型对训练数据过度拟合（某些噪声也被拟合进去），以至于用该模型来预测其他测试样本输出的时候与实际输出或者期望值相差很大。欠拟合则刚好相反，是由于模型使用的参数过少，以至于得到的模型难以拟合训练数据的现象。实际问题中就是不断寻求一种介于过拟合与欠拟合之间的状态。

欠拟合在实际的问题中相对容易被发现，只要通过继续训练模型或者更换学习算法就可以得到解决；而过拟合的解决方法主要有：增加训练数据量使得噪声影响尽可能降低或者说数据的规律更好的体现，减少模型参数的数量，还有就是在误差函数中添加正则项。

2. 贝叶斯分类与决策树分类的优缺点

在分析贝叶斯分类与决策树分类的优缺点之前，首先对两种方法的基本逻辑进行对比。

贝叶斯分类方法的核心就是贝叶斯公式：

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

很多实际问题中，直接求概率$P(A|B)$往往是非常困难的，贝叶斯公式给出了一个间接求此概率的方法，这种方法就类似于我们经常会通过求某个事件的对立事件的概率（相对容易求得）来间接得到该事件发生的概率。

我们在进行贝叶斯分类时，假设目标分类集合为$\mathrm{A} = \{A_1, A_2, \dots A_k, \dots A_K\}$,即总共有K种分类结果，而$B = \{B_1, B_2, \dots B_n, \dots B_N\}$表示我们考虑的因素(feature)有$N$个。那么分类的目的就是在考虑 N 个因素的情况下判定某个未知分类的样本是属于哪个子集A_i。

步骤一：求得该样本在每种分类中的概率(最后一步假设任意两个因素B_m, B_n之间相互独立)，实际在求此概率的时候需要灵活运用下面公式

$$\mathrm{P}(A_i|\mathrm{B}) = \frac{P(B|A_i)P(A_i)}{P(B)} = \frac{P(B|A_i)P(A_i)}{\sum_i P(B|A_i)P(A_i)}$$

$$= \frac{P(B_1 \cup B_2 \cup \dots B_n \cup \dots B_N|A_i)P(A_i)}{\sum_i P(B_1 \cup B_2 \cup \dots B_n \cup \dots B_N|A_i)P(A_i)} = \frac{P(A_i)\prod_n P(B_n|A_i)}{\sum_i P(A_i)\prod_n P(B_n|A_i)}$$

步骤二：将样品分类到概率最大的那个子集中

$$arg\ \max_i \frac{P(A_i)\prod_n P(B_n|A_i)}{\sum_i P(A_i)\prod_n P(B_n|A_i)}$$

决策树分类方法的核心在信息熵公式：

$$H = -\sum_i p(x_i)\, log_2\, p(x_i)$$

信息熵的物理意义表示的是混乱程度，数据越混乱，信息熵的值越大，而分类的目的则是将数据变得不那么混乱，找出其中的分类规律。
同样的，我们考虑分类的因素(feature)有多种，那么决策树分类方法的一个关键问题就是如何通过数据集来获得决策树，更准确的说，我们如何确定这些 feature 在决策树上占据的节点位置。整个决策树的构造其实是一个递归的过程，每个节点以及向下的分支都是类似的决策过程，以下就以某一个节点为例进行分析。

递归步骤：在所有未考虑的 feature 中，计算出依据每种 feature 对当前子集进行分类后的信息熵，选取信息熵最小（数据变得最整齐）的那个 feature 作为当前该节点的分类依据

当通过以上递归步骤对所有的 feature 进行分类之后，该决策树就构造完成，对于每一个待分类的样本，只要在决策树快速找出对应的分类即可。

通过以上的分析可总结两种方法的优缺点如下：

贝叶斯分类：需要训练估计的参数较少，因此对于数据缺失相对不敏感，该分类方法假设 feature 之间相互独立，而这个条件在实际中往往很难严格成立，当 feature 之间的相关较大或者 feature 较多时，分类效率相对决策树较低，但是当 feature 之间相关性较小时，其性能好于决策树分类；
决策树分类：决策树分类方法的优点是操作起来比较简单，计算量相对较小，一旦通过数据集训练得到决策树后，对新样本的分类十分方便，也非常容易理解。缺点就是我们很难确定到底应该用多少 feature 去生成决策树，这就导致很容易出现过拟合现象，需要额外的技术进行改进，同时一旦某些关键数据缺失，可能对决策树的生成造成困难。