什么是支持向量机

支持向量机，即 SVM(Support Vector Machine)，是常见的一种分类方法，在机器学习中，SVM 是有监督的学习模型。监督的学习指的是我们需要事先对数据打上分类标签，这样机器就知道这个数据属于哪个分类。SVM 作为有监督的学习模型，通常可以帮我们模式识别、分类以及回归分析。

SVM 的工作原理

在复杂的分类问题中，同一个平面上内很难将数据进行划分。将二维平面变成了三维空间。原来的分割曲线变成了一个平面（超平面）。用 SVM 计算的过程就是帮我们找到那个超平面的过程，这个超平面就是我们的 SVM 分类器。

SVM 有个特有的概念：**分类间隔**。实际上，分类环境不是在二维平面中的，而是在多维空间中，这样分割线直线 就变成了决策面 。在保证决策面不变，且分类不产生错误的情况下，我们可以移动决策面，直到产生两个极限的位置。极限的位置是指，如果越过了这个位置，就会产生分类错误。这样的话，两个极限位置之间的分界线就是最优决策面。极限位置到最优决策面的距离，就是“分类间隔”，英文叫做 margin。

如果转动这个最优决策面，会发现可能存在多个最优决策面，它们都能把数据集正确分开，这些最优决策面的分类间隔可能是不同的，而那个拥有“最大间隔”（max margin）的决策面就是 SVM 要找的最优解。

超平面的数学表达式：

$$g(x)=\omega^{T}x+b, 其中\omega, x\in \mathrm{R}^{\mathrm{n}}$$

w、x 是 n 维空间里的向量，其中 x 是函数变量；w 是法向量。法向量这里指的是垂直于平面的直线所表示的向量，它决定了超平面的方向。

SVM 就是帮我们找到一个超平面，这个超平面能将不同的样本划分开，同时使得样本集中的点到这个分类超平面的最小距离（即分类间隔）最大化。在这个过程中，**支持向量**就是离**分类超平面**最近的样本点，实际上如果确定了支持向量也就确定了这个超平面。所以支持向量决定了分类间隔到底是多少，而在最大间隔以外的样本点，其实对分类都没有意义。

所以说， SVM 就是求解最大分类间隔的过程，我们还需要对分类间隔的大小进行定义。

用 di 代表点 xi 到超平面 wxi+b=0 的欧氏距离：

$$d_i = \frac{|\omega x_i + b|}{\|\omega\|}$$

其中||w||为超平面的范数。

目标就是找出所有分类间隔中最大的那个值对应的超平面。在数学上，这是一个凸优化问题。通过凸优化问题，最后可以求出最优的 w 和 b，也就是我们想要找的最优超平面。

硬间隔、软间隔和非线性 SVM

假如数据是完全的线性可分的，那么学习到的模型可以称为硬间隔支持向量机。**换个说法，硬间隔指的就是完全分类准确，不能存在分类错误的情况。软间隔，就是允许一定量的样本分类错误。**

线性可分是个理想情况。实际工作中的数据没有那么“干净”，或多或少都会存在一些噪点。这时，我们需要使用到软间隔 SVM（近似线性可分）。

另外还存在一种情况，就是非线性支持向量机。不论是多高级的分类器，只要映射函数是线性的，就没法处理，SVM 也处理不了。这时，需要引入一个新的概念：**核函数。它可以将样本从原始空间映射到一个更高维的特质空间中，使得样本在新的空间中线性可分。**这样我们就可以使用原来的推导来进行计算，只是所有的推导是在新的空间，而不是在原来的空间中进行

在非线性 SVM 中，核函数的选择就是影响 SVM 最大的变量。最常用的核函数有线性核、多项式核、高斯核、拉普拉斯核、sigmoid 核，或者是这些核函数的组合。这些函数的区别在于映射方式的不同。通过这些核函数，我们就可以把样本空间投射到新的高维空间中。

软间隔和核函数的提出，都是为了方便我们对上面超平面公式中的 w* 和 b* 进行求解，从而得到最大分类间隔的超平面。

用 SVM 如何解决多分类问题

SVM 本身是一个二值分类器，最初是为二分类问题设计的。实际上我们要解决的问题，可能是多分类的情况，比如对文本进行分类，或者对图像进行识别。

针对这种情况，可以将多个二分类器组合起来形成一个多分类器，常见的方法有“一对多法”和“一对一法”两种。

1. 一对多法

   假设我们要把物体分成 A、B、C、D 四种分类，那么我们可以先把其中的一类作为分类 1，其他类统一归为分类 2。这样我们可以构造 4 种 SVM，分别为以下的情况：

   （1）样本 A 作为正集，B，C，D 作为负集；

   （2）样本 B 作为正集，A，C，D 作为负集；

   （3）样本 C 作为正集，A，B，D 作为负集；

   （4）样本 D 作为正集，A，B，C 作为负集。

   这种方法，针对 K 个分类，需要训练 K 个分类器，分类速度较快，但训练速度较慢，因为每个分类器都需要对全部样本进行训练，而且负样本数量远大于正样本数量，会造成样本不对称的情况，而且当增加新的分类，比如第 K+1 类时，需要重新对分类器进行构造。

2. 一对一法

   一对一法的初衷是想在训练的时候更加灵活。可以在任意两类样本之间构造一个SVM，这样针对 K 类的样本，就会有 C(k,2) 类分类器。

   比如想要划分 A、B、C 三个类，可以构造 3 个分类器：

   （1）分类器 1：A、B；

   （2）分类器 2：A、C；

   （3）分类器 3：B、C。

   当对一个未知样本进行分类时，每一个分类器都会有一个分类结果，即为 1 票，最终得票最多的类别就是整个未知样本的类别。这样做的好处是，如果新增一类，不需要重新训练所有的 SVM，只需要训练和新增这一类样本的分类器。而且这种方式在训练单个 SVM 模型的时候，训练速度快。

   但这种方法的不足在于，分类器的个数与 K 的平方成正比，所以当 K 较大时，训练和测试的时间会比较慢。