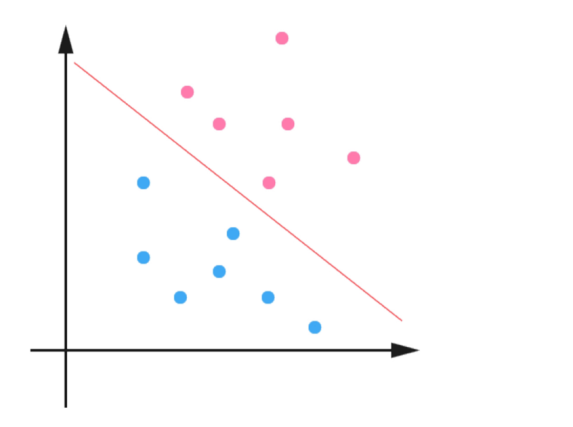
**支持向量机的目的：**

通俗来讲，它是一种二分类模型，模型基本定义为，在特征空间上，试图找到一个间隔最大的线或者平面乃至超平面，使得分类正确。支持向量机的学习策略便是间隔最大化，最终可转化为一个凸二次规划问题。

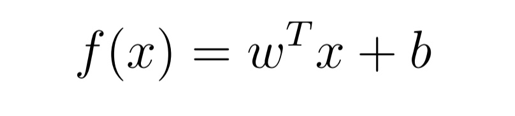
**线性分类：**

下图所示，平面上有两种不同的点，分别用两种不同的颜色表示，一种为红颜色的点，另一种则为蓝颜色的点，红颜色的线表示一个可行的超平面。

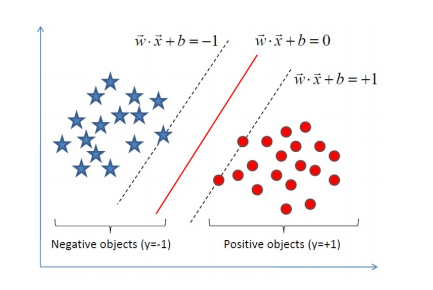


可以看出，这条红颜色的线把红颜色的点和蓝颜色的点分开来了。而这条红颜色的线就是我们上面所说的超平面，也就是说，这个所谓的超平面，把这两种不同颜色的数据点分隔开来，在超平面一边的数据点所对应的 y 全是-1，而在另一边全是 1 。

可以令分类函数：



如果 f(x) = 0 ，那么 x 是位于超平面上的点。我们不妨要求对于所有满足 f(x) < 0 的点，其对应的 y 等于 −1 ，而 f(x) > 0 则对应 y = 1 的数据点。



**前提知识储备：**

但是在认识并求出这个超平面的过程之前，我们先来了解一下间隔的概念。一般而言，一个点距离超平面的远近可以表示为分类预测的确信或准确程度。

在超平面 w ∗ x + b = 0 确定的情况下，|w ∗ x + b| 能够相对的表示点 x 到距离超平面的远近，而 w ∗ x + b 的符号与类标记 y 的符号是否一致表示分类是否正确，所 以，可以用量 y ∗ (w ∗ x + b) 的正负性来判定或表示分类的正确性和确信度。

于是，引出样本到分类间隔距离的函数间隔 functional margin 的概念。

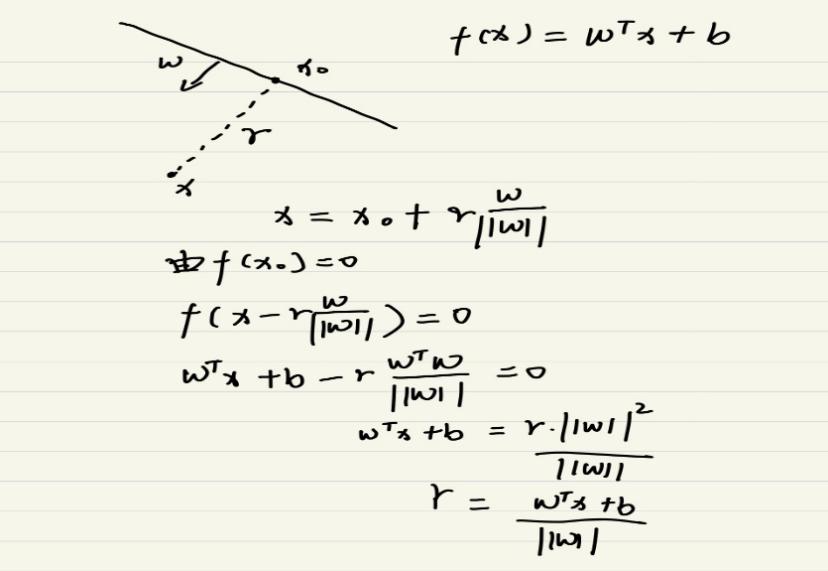
**函数间隔Function margin:**



接着，我们定义超平面 (w, b) 关于训练数据集 T 的函数间隔为超平面 (w, b) 关于 T 中所有样本点 (xi , yi) 的函数间隔最小值，其中，x 是特征，y 是结果标签，i 表示第 i 个 样本，有：

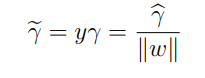


然与此同时，问题就出来了。上述定义的函数间隔虽然可以表示分类预测的正确性 和确信度，但在选择分类超平面时，只有函数间隔还远远不够，因为如果成比例的改变 w 和 b，如将他们改变为 2w 和 2b，虽然此时超平面没有改变，但函数间隔的值 f(x) 却 变成了原来的 2 倍。 其实，我们可以对法向量 w 加些约束条件，使其表面上看起来规范化，如此，我们 很快又将引出真正定义点到超平面的距离 --几何间隔 geometrical margin 的概念（很快 你将看到，几何间隔就是函数间隔除以个 ∥w∥，即 yf(x)/∥w∥）。

**几何间隔 Geometrical margin:**

（1）

如上图中所示，直线外一点x到直线上一点x0距离为r，w为直线在x方向上的法向量，将w与w的模值的比值作为分度值，则得到式（1），又因为x0在直线上，则f(x0)=0，反解出r的值。此时r是带符号的，我们需要的只是它的绝对值，因此类似地，也乘上对应的类别 y 即可，因此实际上我们定义几何间隔 geometrical margin 为：

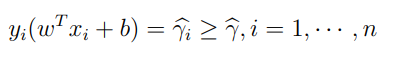


**原问题的引出：**

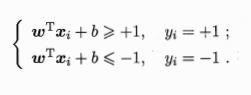
我们从函数间隔和几何间隔的概念了解到，函数间隔不适合用来描述定义超平面所需要的最大间隔，而几何间隔可以用来描述这个间隔。因为除上了∥w∥这个分母，所以缩放 w 和 b 的时候的值是不会改变的，它只随着超平面 hyper plane 的变动而变动，因此，这是更加合 适的一个margin 。这样一来，我们的 maximum margin classifier 的目标函数可以定义为：



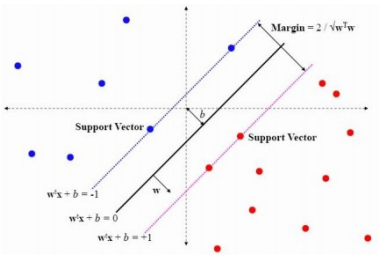
当然，还需要满足一些条件，根据 margin 的定义，我们有 ：



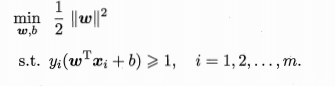
假设超平面（w,b）能够将数据样本正确分类，即对于（xi,yi）∈D，若yi=+1，则有wTxi+b>0;若yi=-1，则有wTxi+b<0。



两个支撑着中间的 gap 的超平面，它们到中间的纯红线 separating hyper plane 的距离相等，即我们所能得到的最大的 geometrical marginγe，而“支撑”这两个超平面的必定会有一些点，而这些“支撑”的点便叫做支持向量 Support Vector。Support Vector 便是那蓝色虚线和粉红色虚线上的点：



在红色跟蓝色虚线上的样本点分别满足wTxi+b=+1和wTxi+b=-1，分析知样本到超平面间隔最大只需满足，支持向量到超平面的几何距离最大即可，即使得1/最大 ，即使得最小，yi为符号位，条件是超平面可以将样本分类正确，以上可表示为：



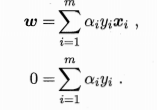
求解这个问题，即可求出超平面，所以这又叫做SVM的**原问题**。

**对偶问题的引出：**

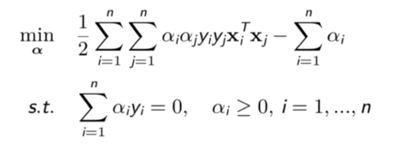
有拉格朗日定理可以知道，可以引入拉格朗日乘子，将约束条件与目标函数结合形成对新函数，将原函数求解转化成对新函数的求解。



令w和b求偏导数得：



将其带入拉格朗日函数中，消去w和b得：



求解对偶问题跟求解原问题是等价的，他们都可以求得一个相同的超平面。

**核函数的引出：**

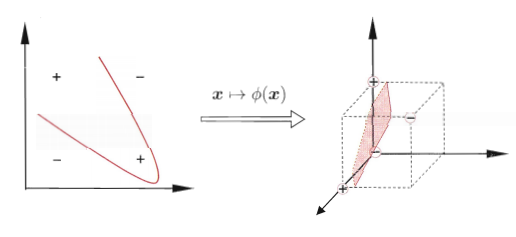
在了解核函数之前，我们需要先了解一下特征变换的概念及其作用：

特征变换：将原始数据或者图像通过一定的数字变换生成一组新的特征数据或者图像，使得这组新的数据或者图像集中反映在少数几个特征数据或者特征图像上，这样，使得数据量有所减少。

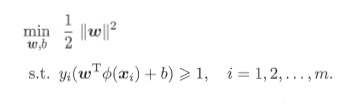
二维空间内，当样本可以被一条线分类正确时，称作线性可分，即在样本空间内，样本可以被跟样本同一维度下的超平面正确分类。但是，存在一种情况，即在同一维度下，不存在超平面使得样本分类正确，这叫做线性不可分。下面进入线性不可分情况的探讨：

线性不可分：

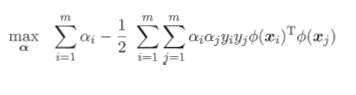
在同一维度的样本特征空间内不可分时，我们将其映射到高维空间内，使得其存在一个超平面可以将其正确分类。X经过特征变换成，直观感受如下图：

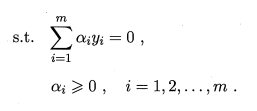


原问题转换成：



将对偶问题转换成：





在原问题上，我们假设xi中一共有d个特征，即有 d个变量，n个约束条件；在对偶问题上，我们在形式上变成了对n个变量，n+1个约束的问题的求解。我们将原问题转换成对偶问题的动机是求解与无关，但是依旧与相关。于是就有了我们的核函数。

先看一个例子：

假设x有d个特征，分别为（x1，x2，…，xd）,对其进行二阶多项式变换则：

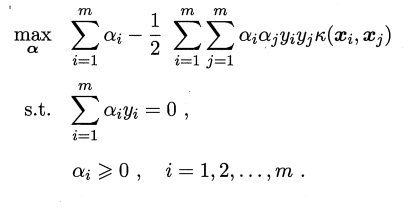
所以整理后其内积1+

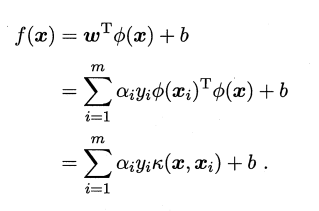
由此可知，在经过特征变换之后，的运算复杂度为，但是经过整理后其运算复杂度为d，即只需要计算的运算复杂度，大大减少了数据处理难度。在此处，我们将核函数定义为：



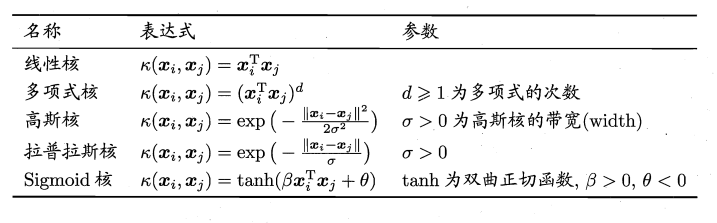
即与在特征空间的内积为他们在原始样本空间中通过k(.)计算出的结果。

于是可以将对偶问题写成：



求解可以得到：

但是在实际过程中，我们可以完全不知道核映射具体形式是什么，因为映射的维度可能会很高，表达会十分复杂，所以我们不知道什么样的核函数是合适的。若核函数选择不佳，我们很有可能会将样本映射到一个不合适的特征空间中，导致性能不佳。以下我们介绍几种常用的核函数：

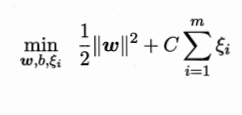


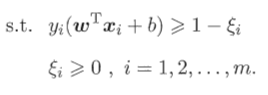
基本经验：文本数据常用线性核，情况不明常用高斯核。

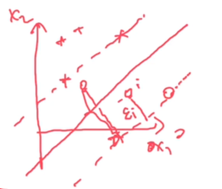
**引入软间隔：**

在实际应用中，由于数据噪音的存在，或者强行数据本身的不正常分布，我们如果依旧要求数据强行可分，则容易造成过拟合；此外，如果映射到过于高阶的空间中，也容易造成过拟合。为避免过拟合，我们尝试在一定限度内允许支持向量机在某些样本分类出错，我们把这种向量机叫软间隔支持向量机。

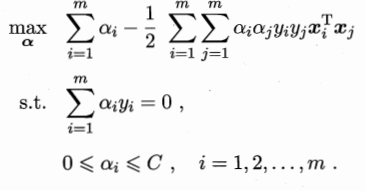
采用hinge损失函数，则原问题改写成：







若1，则代表该样本处于间隔内部，若，则代表该样本被分类错误。

对偶问题则改写成：