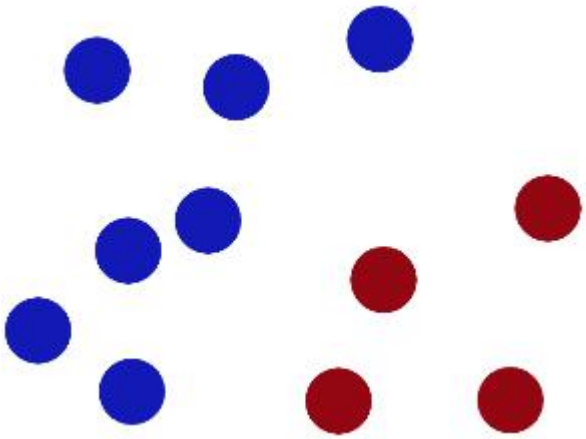
# 一、支持向量机(svm)

## 一、什么是支持向量机

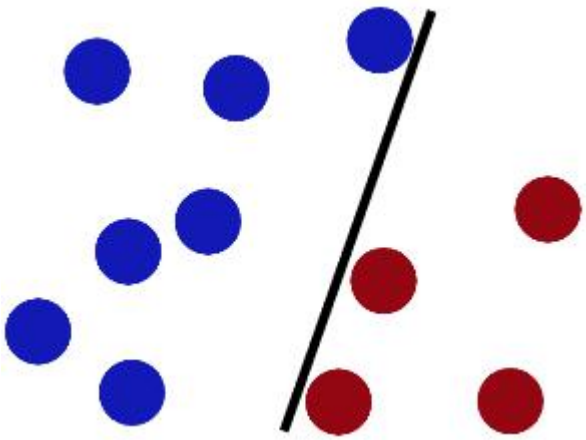
支持向量机，Support Vector Machines（ svm ）是用于分类的一种算法。

在很久以前的情人节，大侠要去救他的爱人，但魔鬼和他玩了一个游戏。

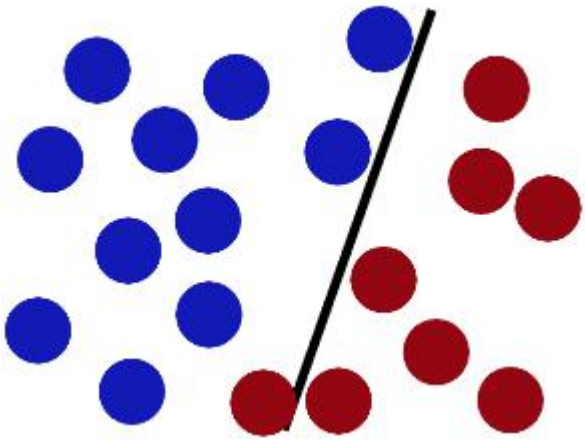
魔鬼在桌子上似乎有规律放了两种颜色的球，说：“你用一根棍分开它们？要求：尽量在放更多球之后，仍然适用。”



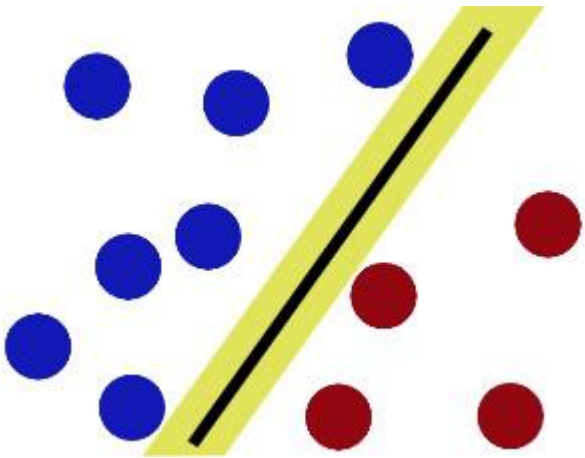
于是大侠这样放，干的不错？



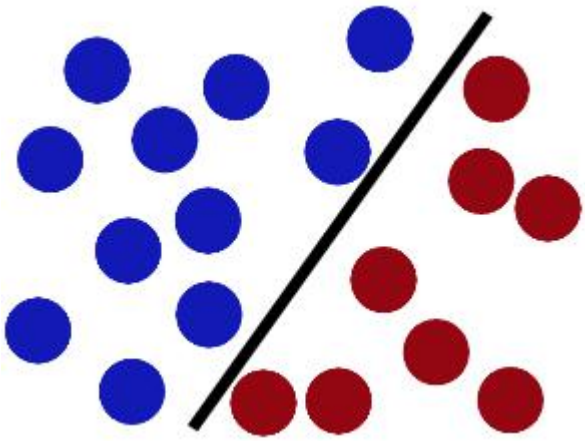
然后魔鬼，又在桌上放了更多的球，似乎有一个球站错了阵营。



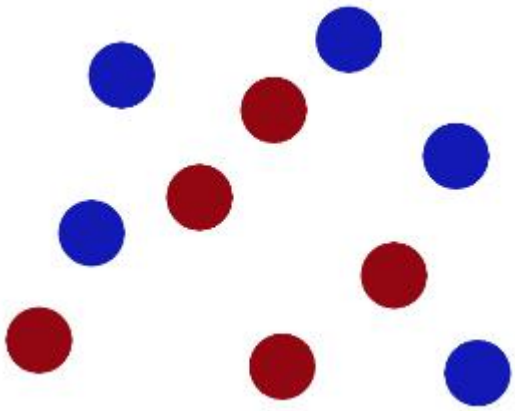
SVM就是试图把棍放在最佳位置，好让在棍的两边有尽可能大的间隙。



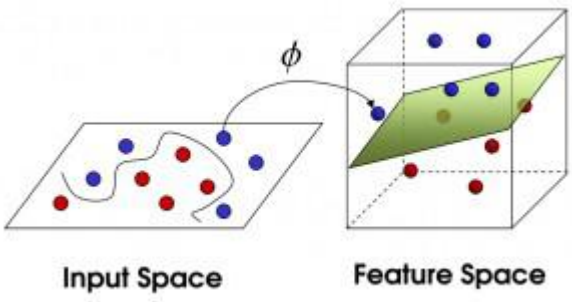
现在即使魔鬼放了更多的球，棍仍然是一个好的分界线。



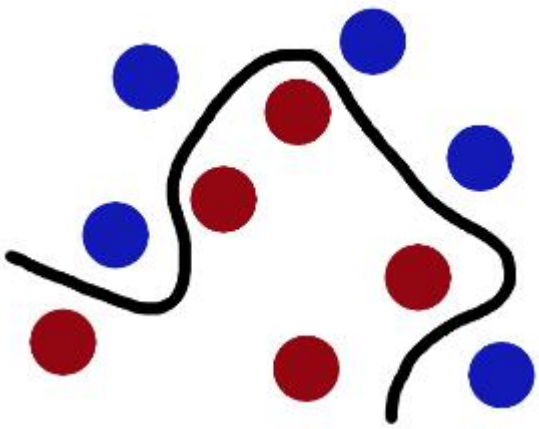
然后，在SVM 工具箱中有另一个更加重要的 **trick**。 魔鬼看到大侠已经学会了一个trick，于是魔鬼给了大侠一个新的挑战。



现在，大侠没有棍可以很好帮他分开两种球了，现在怎么办呢？当然像所有武侠片中一样大侠桌子一拍，球飞到空中。然后，凭借大侠的轻功，大侠抓起一张纸，插到了两种球的中间。




现在，从魔鬼的角度看这些球，这些球看起来像是被一条曲线分开了。



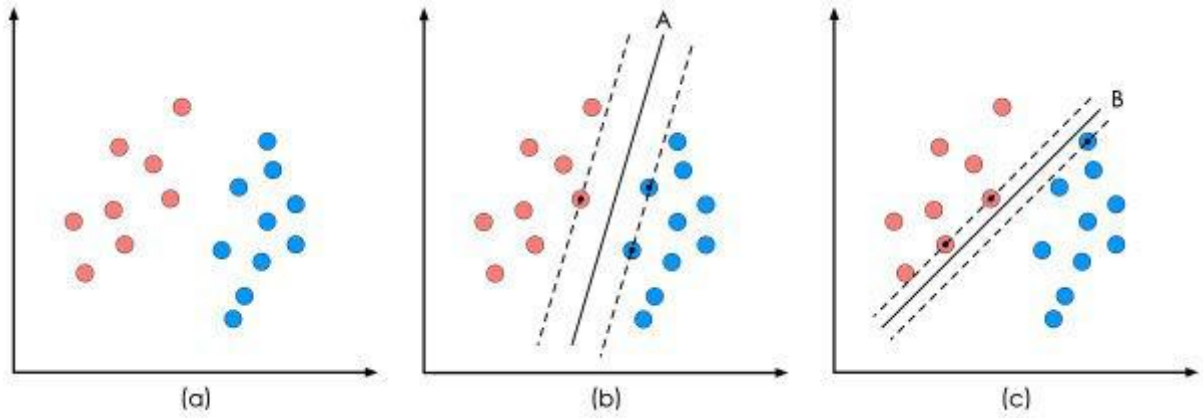
再之后，无聊的大人们，把这些球叫做 **「data」** ，把棍子 叫做 **「classifier」**，最大间隙trick 叫做 **「optimization」** ， 拍桌子叫做**「kernelling」**，那张纸叫做**「hyperplane」。**

当一个分类问题，数据是线性可分的，也就是用一根棍就可以将两种小球分开的时候，我们只要将棍的位置放在让小球距离棍的距离最大化的位置即可，寻找这个最大间隔的过程，就叫做最优化。但是，现实往往是很残酷的，一般的数据是线性不可分的，也就是找不到一个棍将两种小球很好的分类。这个时候，我们就需要像大侠一样，将小球拍起，用一张纸代替小棍将小球进行分类。想要让数据飞起，我们需要的东西就是核函数(kernel)，用于切分小球的纸，就是超平面。

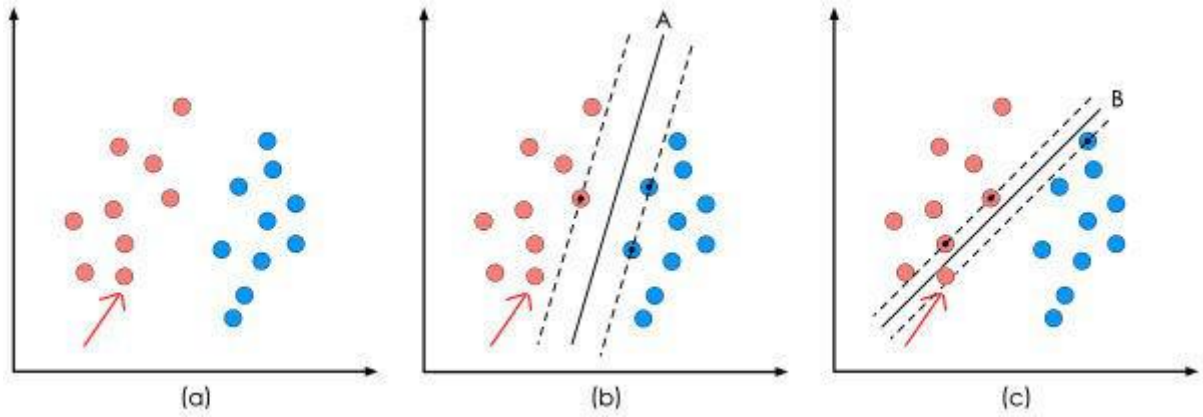
也许这个时候，你还是似懂非懂，没关系。根据刚才的描述，可以看出，问题是从线性可分延伸到线性不可分的。那么，我们就按照这个思路，进行原理性的剖析。

## 二、线性SVM

先看下线性可分的二分类问题。




上图中的(a)是已有的数据，红色和蓝色分别代表两个不同的类别。数据显然是线性可分的，但是将两类数据点分开的直线显然不止一条。上图的(b)和(c)分别给出了B、C两种不同的分类方案，其中黑色实线为分界线，术语称为“决策面”。每个决策面对应了一个线性分类器。虽然从分类结果上看，分类器A和分类器B的效果是相同的。但是他们的性能是有差距的，看下图：




在"决策面"不变的情况下，我又添加了一个红点。可以看到，分类器B依然能很好的分类结果，而分类器C则出现了分类错误。显然分类器B的"决策面"放置的位置优于分类器C的"决策面"放置的位置，SVM算法也是这么认为的，它的依据就是分类器B的分类间隔比分类器C的分类间隔大。这里涉及到第一个SVM独有的概念"分类间隔"。在保证决策面方向不变且不会出现错分样本的情况下移动决策面，会在原来的决策面两侧找到两个极限位置（越过该位置就会产生错分现象），如虚线所示。虚线的位置由决策面的方向和距离原决策面最近的几个样本的位置决定。而这两条平行虚线正中间的分界线就是在保持当前决策面方向不变的前提下的最优决策面。两条虚线之间的垂直距离就是这个最优决策面对应的分类间隔。显然每一个可能把数据集正确分开的方向都有一个最优决策面（有些方向无论如何移动决策面的位置也不可能将两类样本完全分开），而不同方向的最优决策面的分类间隔通常是不同的，那个具有“最大间隔”的决策面就是SVM要寻找的最优解。而这个真正的最优解对应的两侧虚线所穿过的样本点，就是SVM中的支持样本点，称为"支持向量"。

## 1、数学建模

求解这个"决策面"的过程，就是最优化。一个最优化问题通常有两个基本的因素：1）目标函数，也就是你希望什么东西的什么指标达到最好；

2）优化对象，你期望通过改变哪些因素来使你的目标函数达到最优。在线性SVM算法中，目标函数显然就是那个"分类间隔"，而优化对象则是决策面。所以要对SVM问题进行数学建模，首先要对上述两个对象（"分类间隔"和"决策面"）进行数学描述。按照一般的思维习惯，我们先描述决策面。

数学建模的时候，先在二维空间建模，然后再推广到多维。

**1)"决策面"方程**

我们都知道二维空间下一条直线的方式如下所示：

$$y = ax + b$$

现在我们做个小小的改变，让原来的x轴变成x1，y轴变成x2。

$$x_2 = ax_1 + b$$

移项得：

$$ax_1 - x_2 + b = 0$$

将公式向量化得：

$$[a \quad -1]\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + b = 0$$

进一步向量化，用w列向量和x列向量和标量γ进一步向量化：

$$\boldsymbol{\omega}^T \boldsymbol{x} + \gamma = 0$$

其中，向量w和x分别为：

$$\boldsymbol{\omega} = [\omega_1, \omega_2]^T, \boldsymbol{x} = [x_1, x_2]^T$$

这里w1=a，w2=-1。我们都知道，最初的那个直线方程a和b的几何意义，a表示直线的斜率，b表示截距，a决定了直线与x轴正方向的夹角，b决定了直线与y轴交点位置。那么向量化后的直线的w和r的几何意义是什么呢？

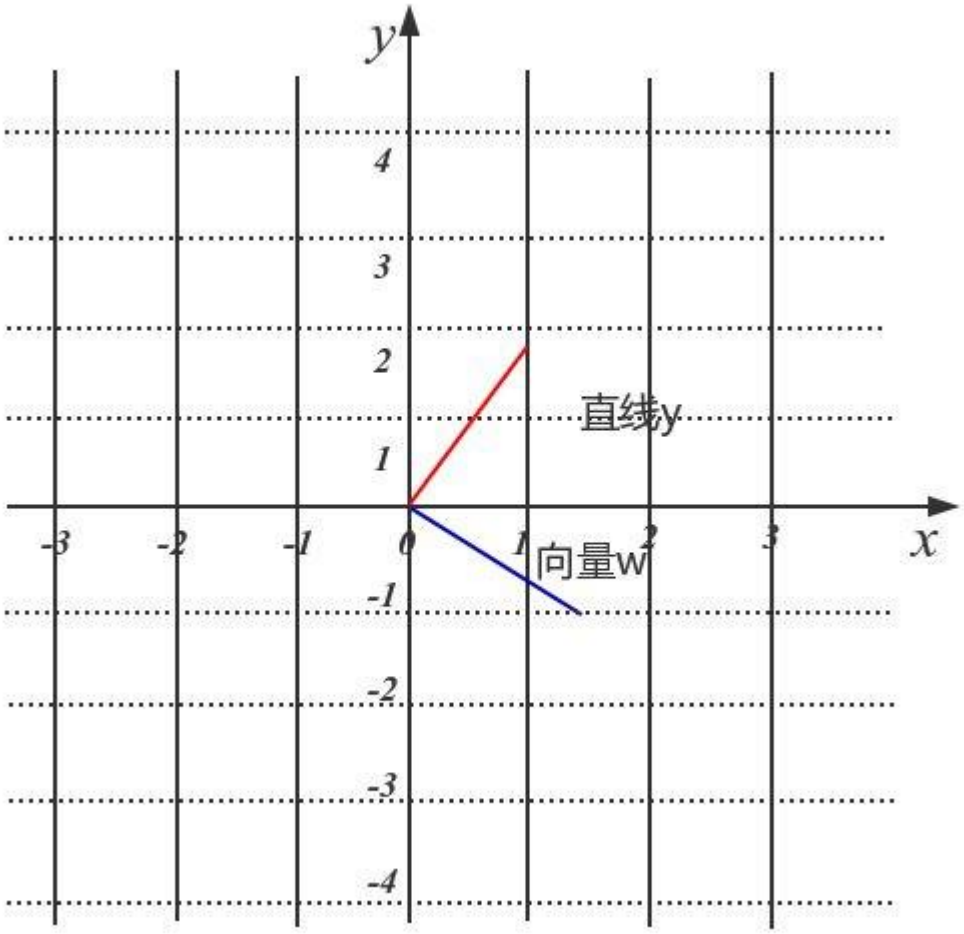
现在假设：

$$a = \sqrt{3}, b = 0$$

可得：

$$\boldsymbol{\omega} = [\sqrt{3}, -1]^T$$

在坐标轴上画出直线和向量w：




蓝色的线代表向量w，红色的先代表直线y。我们可以看到向量w和直线的关系为垂直关系。这说明了向量w也控制这直线的方向，只不过是与这个直线的方向是垂直的。标量γ的作用也没有变，依然决定了直线的截距。此时，我们称w为直线的法向量。

二维空间的直线方程已经推导完成，将其推广到n为空间，就变成了超平面方程。(一个超平面，在二维空间的例子就是一个直线)但是它的公式没变，依然是：

$$\boldsymbol{\omega}^T\boldsymbol{x}+\gamma=\mathbf{0}$$

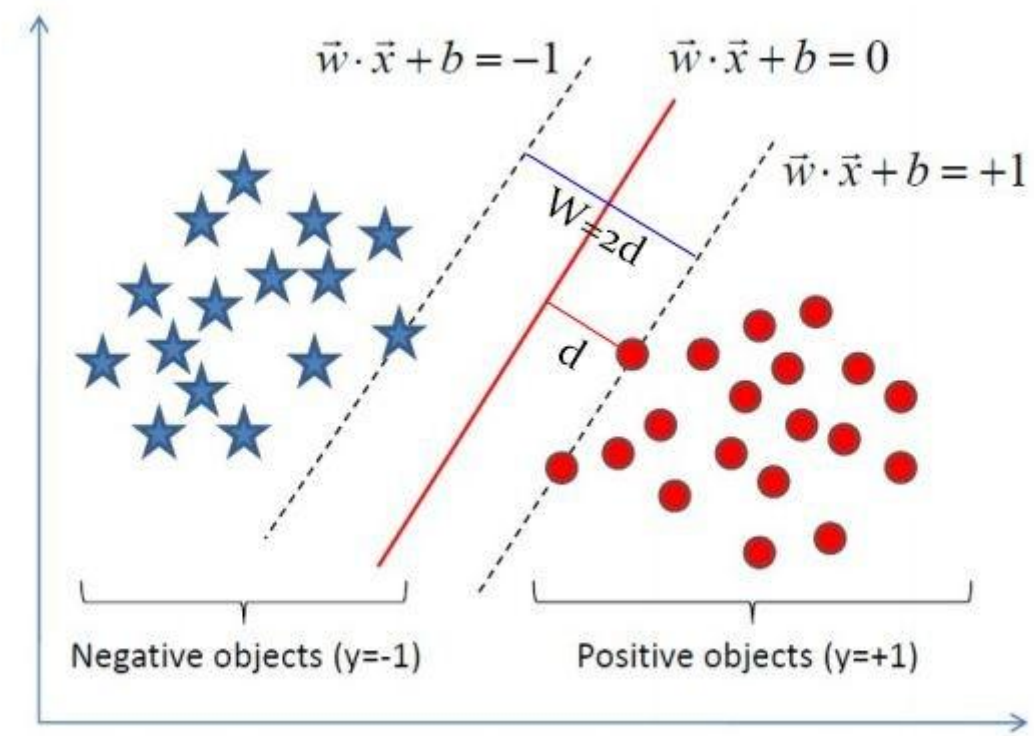
不同之处在于：

$$\boldsymbol{\omega}=[\omega_1,\omega_2,\cdots,\omega_n]^T$$
$$\boldsymbol{x}=[x_1,x_2,\cdots,x_n]^T$$

我们已经顺利推导出了"决策面"方程，它就是我们的超平面方程，之后，我们统称其为超平面方程。

**2)分类间隔"方程**

现在，我们依然对于一个二维平面的简单例子进行推导。




我们已经知道间隔的大小实际上就是支持向量对应的样本点到决策面的距离的二倍。那么图中的距离d我们怎么求？我们高中都学过，点到直线的距离距离公式如下：

$$d = \left| \frac{Ax_0 + By_0 + C}{\sqrt{A^2 + B^2}} \right|$$

公式中的直线方程为Ax0+By0+C=0，点P的坐标为(x0,y0)。

现在，将直线方程扩展到多维，求得我们现在的超平面方程，对公式进行如下变形：

$$d = \frac{|\boldsymbol{\omega}^T \boldsymbol{x} + \gamma|}{||\boldsymbol{\omega}||}$$

这个d就是"分类间隔"。其中||w||表示w的二范数，求所有元素的平方和，然后再开方。比如对于二维平面：

$$\boldsymbol{\omega} = [\omega_1, \omega_2]^T$$

那么，

$$||\boldsymbol{\omega}|| = \sqrt[2]{{\omega_1}^2 + {\omega_2}^2}$$

我们目的是为了找出一个分类效果好的超平面作为分类器。分类器的好坏的评定依据是分类间隔W=2d的大小，即分类间隔W越大，我们认为这个超平面的分类效果越好。此时，求解超平面的问题就变成了求解分类间隔W最大化的为题。W的最大化也就是d最大化的。

**3)约束条件**

看起来，我们已经顺利获得了目标函数的数学形式。但是为了求解w的最大值。我们不得不面对如下问题：

- 我们如何判断超平面是否将样本点正确分类？
- 我们知道要求距离d的最大值，我们首先需要找到支持向量上的点，怎么在众多的点中选出支持向量上的点呢？

上述我们需要面对的问题就是约束条件，也就是说我们优化的变量d的取值范围受到了限制和约束。事实上约束条件一直是最优化问题里最让人头疼的东西。但既然我们已经知道了这些约束条件确实存在，就不得不用数学语言对他们进行描述。但SVM算法通过一些巧妙的小技巧，将这些约束条件融合到一个不等式里面。

这个二维平面上有两种点，我们分别对它们进行标记：红

- 颜色的圆点标记为1，我们人为规定其为正样本；
- 蓝颜色的五角星标记为-1，我们人为规定其为负样本。

对每个样本点xi加上一个类别标签yi：

$$y_i = \begin{cases} +1 & \text{红色点} \\ -1 & \text{蓝色点} \end{cases}$$

如果我们的超平面方程能够完全正确地对上图的样本点进行分类，就会满足下面的方程：

$$\begin{cases} \boldsymbol{\omega}^T x_i + \gamma > 0 & y_i = 1 \\ \boldsymbol{\omega}^T x_i + \gamma < 0 & y_i = -1 \end{cases}$$

如果我们要求再高一点，假设决策面正好处于间隔区域的中轴线上，并且相应的支持向量对应的样本点到决策面的距离为d，那么公式进一步写成：

$$\begin{cases} \dfrac{\boldsymbol{\omega}^T x_i + \gamma}{||\boldsymbol{\omega}||} \geq \mathrm{d} & \forall y_i = 1 \\ \dfrac{\boldsymbol{\omega}^T x_i + \gamma}{||\boldsymbol{\omega}||} \leq -\mathrm{d} & \forall y_i = -1 \end{cases}$$

上述公式的解释就是，对于所有分类标签为1和-1样本点，它们到直线的距离都大于等于d(支持向量上的样本点到超平面的距离)。公式两边都除以d，就可以得到：

$$\begin{cases} \boldsymbol{\omega_d}^T x_i + \gamma_d \geq 1 & \forall y_i = 1 \\ \boldsymbol{\omega_d}^T x_i + \gamma_d \leq -1 & \forall y_i = -1 \end{cases}$$

其中，

$$\boldsymbol{\omega}_d = \frac{\boldsymbol{\omega}}{||\boldsymbol{\omega}||d}, \quad \gamma_d = \frac{\gamma}{||\boldsymbol{\omega}||d}$$

因为||w||和d都是标量。所上述公式的两个矢量，依然描述一条直线的法向量和截距。

$$\boldsymbol{\omega}_d^T \boldsymbol{x} + \gamma_d = 0$$

$$\boldsymbol{\omega}^T \boldsymbol{x} + \gamma = 0$$

上述两个公式，都是描述一条直线，数学模型代表的意义是一样的。现在，让我们对wd和γd重新起个名字，就叫它们w和γ。因此，我们就可以说："对于存在分类间隔的两类样本点，我们一定可以找到一些超平面面，使其对于所有的样本点均满足下面的条件："

$$\begin{cases} \boldsymbol{\omega}^T x_i + \gamma \geq 1 & \forall y_i = 1 \\ \boldsymbol{\omega}^T x_i + \gamma \leq -1 & \forall y_i = -1 \end{cases}$$

上述方程即给出了SVM最优化问题的约束条件。这时候，可能有人会问了，为什么标记为1和-1呢？因为这样标记方便我们将上述方程变成如下形式：

$$y_i\left(\boldsymbol{\omega}^T x_i + \gamma\right) \geq 1 \quad \forall x_i$$

正是因为标签为1和-1，才方便我们将约束条件变成一个约束方程，从而方便我们的计算。

**4)线性SVM优化问题基本描述**

现在整合一下思路，我们已经得到我们的目标函数：

$$d = \frac{|\boldsymbol{\omega}^T \boldsymbol{x} + \gamma|}{||\boldsymbol{\omega}||}$$

我们的优化目标是是d最大化。我们已经说过，我们是用支持向量上的样本点求解d的最大化的问题的。那么支持向量上的样本点有什么特点呢？

$$|\boldsymbol{\omega}^T x_i + \gamma| = 1 \quad \forall 支持向量上的样本点 x_i$$

你赞同这个观点吗？所有支持向量上的样本点，都满足如上公式。如果不赞同，请重看"分类间隔"方程推导过程。

现在我们就可以将我们的目标函数进一步化简：

$$\mathrm{d} = \frac{1}{||\boldsymbol{\omega}||}$$

因为，我们只关心支持向量上的点。随后我们求解d的最大化问题变成了||w||的最小化问题。进而||w||的最小化问题等效于

$$\min \frac{1}{2} ||\omega||^2$$

为什么要做这样的等效呢？这是为了在进行最优化的过程中对目标函数求导时比较方便，但这绝对不影响最优化问题最后的求解。我们将最终的目标函数和约束条件放在一起进行描述：

$$\min \frac{1}{2} ||\omega||^2$$
$$\text{s.t.} \quad y_i\left(\boldsymbol{\omega}^T x_i + b\right) \geq 1, i = 1,2, \ldots, n$$

这里n是样本点的总个数，缩写s.t.表示"Subject to"，是"服从某某条件"的意思。上述公式描述的是一个典型的不等式约束条件下的二次型函数优化问题，同时也是支持向量机的基本数学模型。

**5)求解准备**

我们已经得到支持向量机的基本数学模型，接下来的问题就是如何根据数学模型，求得我们想要的最优解。在学习求解方法之前，我们得知道一点，想用我下面讲述的求解方法有一个前提，就是我们的目标函数必须是凸函数。理解凸函数，我们还要先明确另一个概念，凸集。在凸几何中，凸集(convex set)是在)凸组合下闭合的放射空间的子集。看一幅图可能更容易理解：

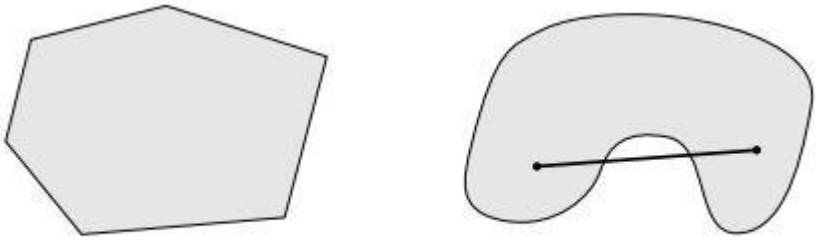


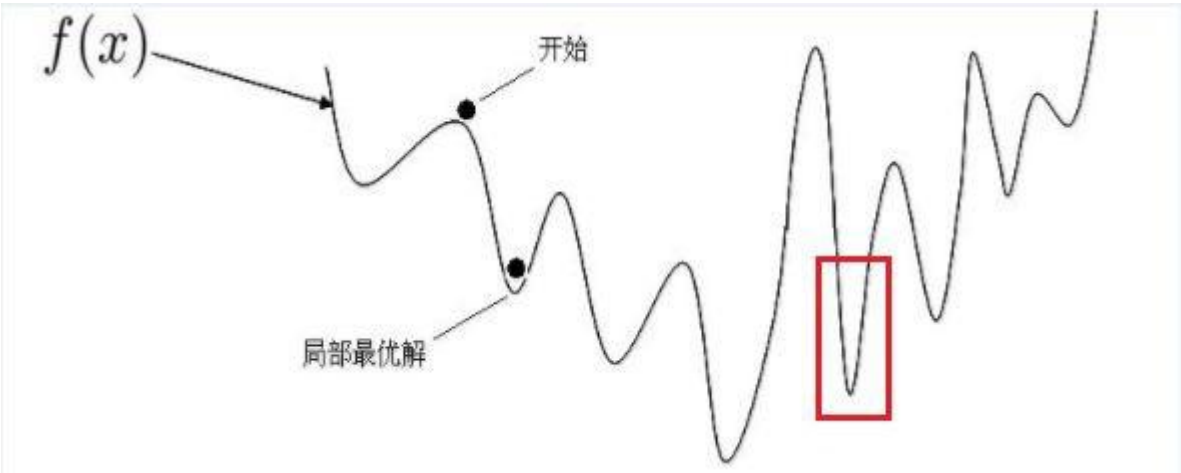
Figure 1: Examples of a convex set (a) and a non-convex set (b).

左右量图都是一个集合。**如果集合中任意2个元素连线上的点也在集合中，那么这个集合就是凸集。**显然，上图中的左图是一个凸集，上图中的右图是一个非凸集。

凸函数的定义也是如此，其几何意义表示为函数任意两点连线上的值大于对应自变量处的函数值。若这里凸集C即某个区间L，那么，设函数f为定义在区间L上的函数，若对L上的任意两点x1，x2和任意的实数λ，λ属于(0,1)，总有：

$$f(\lambda x_1 + (1-\lambda)x_2) \leq \lambda f(x_1) + (1-\lambda) f(x_2)$$

则函数f称为L上的凸函数，当且仅当其上镜图（在函数图像上方的点集）为一个凸集。再看一幅图，也许更容易理解：




像上图这样的函数，它整体就是一个非凸函数，我们无法获得全局最优解的，只能获得局部最优解。比如红框内的部分，如果单独拿出来，它就是一个凸函数。对于我们的目标函数：

$$\min \frac{1}{2} ||\omega||^2$$

很显然，它是一个凸函数。所以，可以使用我接下来讲述的方法求取最优解。

通常我们需要求解的最优化问题有如下几类：

- 无约束优化问题，可以写为：

$$min\ f\ (x)$$

- 有等式约束的优化问题，可以写为：

$$\begin{aligned} &min\ f\ (x) \\ &s.t.\ h_{i(x)} = 0, \quad i = 1,2,\dots,n \end{aligned}$$

- 有不等式约束的优化问题，可以写为：

$$\begin{aligned} &min\ f\ (x) \\ &s.t.\ g_{i(x)} \leq 0, \quad i = 1,2,\dots,n \\ &h_{j(x)} = 0, \qquad j = 1,2,\dots,m \end{aligned}$$

对于第(a)类的优化问题，尝尝使用的方法就是费马大定理(Fermat)，即使用求取函数f(x)的导数，然后令其为零，可以求得候选最优值，再在这些候选值中验证；如果是凸函数，可以保证是最优解。这也就是我们高中经常使用的求函数的极值的方法。

对于第(b)类的优化问题，常常使用的方法就是拉格朗日乘子法（ Lagrange Multiplier），即把等式约束h_i(x)用一个系数与f(x)写为一个式子，称为拉格朗日函数，而系数称为拉格朗日乘子。通过拉格朗日函数对各个变量求导，令其为零，可以求得候选值集合，然后验证求得最优值。

对于第(c)类的优化问题，常常使用的方法就是KKT条件。同样地，我们把所有的等式、不等式约束与f(x)写为一个式子，也叫拉格朗日函数，系数也称拉格朗日乘子，通过一些条件，可以求出最优值的**必要条件**，这个条件称为KKT条件。

必要条件和充要条件如果不理解，可以看下面这句话：

- A的**必要条件**就是A可以推出的**结论**
- A的**充分条件**就是可以推出A的**前提**

了解到这些，现在让我们再看一下我们的最优化问题：

$$\begin{aligned} &\min \frac{1}{2}||\omega||^2 \\ &s.t.\ \ y_i\left(\boldsymbol{\omega}^T x_i + b\right) \geq 1, i = 1,2,\dots,n \end{aligned}$$

现在，我们的这个对优化问题属于哪一类？很显然，它属于第(c)类问题。在学习求解最优化问题之前，我们还要学习两个东西：拉格朗日函数和KKT条件。

**6)拉格朗日函数**

首先，我们先要从宏观的视野上了解一下**拉格朗日对偶问题出现的原因和背景。**

我们知道我们要求解的是最小化问题，所以一个直观的想法是如果我能够构造一个函数，使得该函数在可行解区域内与原目标函数完全一致，而在可行解区域外的数值非常大，甚至是无穷大，那么这个**没有约束条件的新目标函数的优化问题**就与原来**有约束条件的原始目标函数的优化问题**是等价的问题。这就是使用拉格朗日方程的目的，它将约束条件放到目标函数中，**从而将有约束优化问题转换为无约束优化问题。**

随后，人们又发现，使用拉格朗日获得的函数，使用求导的方法求解依然困难。进而，需要对问题再进行一次转换，即使用一个数学技巧：**拉格朗日对偶。**

所以，显而易见的是，我们在拉格朗日优化我们的问题这个道路上，**需要进行下面二个步骤：**

- 将有约束的原始目标函数转换为无约束的新构造的拉格朗日目标函数
- 使用拉格朗日对偶性，将不易求解的优化问题转化为易求解的优化

下面，进行第一步：**将有约束的原始目标函数转换为无约束的新构造的拉格朗日目标函数**

公式变形如下：

$$\mathcal{L}(\boldsymbol{w}, b, \boldsymbol{\alpha}) = \frac{1}{2}\|\boldsymbol{w}\|^2 - \sum_{i=1}^{n} \alpha_i \left(y_i(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_i + b) - 1\right)$$

其中αi是拉格朗日乘子，αi大于等于0，是我们构造新目标函数时引入的系数变量(我们自己设置)。现在我们令：

$$\theta(w) = \max_{\alpha_i \geq 0} \mathcal{L}(w, b, \alpha)$$

当样本点不满足约束条件时，即**在可行解区域外：**

$$y_i(\boldsymbol{\omega}^T x_i + b) < 1$$

此时，我们将αi设置为正无穷，此时θ(w)显然也是正无穷。

当样本点满足约束条件时，即**在可行解区域内：**

$$y_i(\boldsymbol{\omega}^T x_i + b) \geq 1$$

此时，显然θ(w)为原目标函数本身。我们将上述两种情况结合一下，就得到了新的目标函数：

$$\theta(\boldsymbol{\omega}) = \begin{cases} \frac{1}{2}||\boldsymbol{\omega}||^2 & x \in \text{可区域} \\ +\infty & x \in \text{非可行区域} \end{cases}$$

此时，再看我们的初衷，就是为了建立一个在可行解区域内与原目标函数相同，在可行解区域外函数值趋近于无穷大的新函数，现在我们做到了。

现在，我们的问题变成了求新目标函数的最小值，即：

$$\min_{w,b} \theta(w) = \min_{w,b} \max_{\alpha_i \geq 0} \mathcal{L}(w, b, \alpha) = p^*$$

这里用p*表示这个问题的最优值，且和最初的问题是等价的。

接下来，我们进行第二步：**将不易求解的优化问题转化为易求解的优化**

我们看一下我们的新目标函数，先求最大值，再求最小值。这样的话，我们首先就要面对带有需要求解的参数w和b的方程，而αi又是不等式约束，这个求解过程不好做。所以，我们需要使用拉格朗日函数对偶性，将最小和最大的位置交换一下，这样就变成了：

$$\max_{\alpha_i \geq 0} \min_{w,b} \mathcal{L}(w, b, \alpha) = d^*$$

交换以后的新问题是原始问题的对偶问题，这个新问题的最优值用d来表示。而且$d$<=p*。我们关心的是d=p的时候，这才是我们要的解。需要什么条件才能让d=p呢？

- 首先必须满足这个优化问题是凸优化问题。
- 其次，需要满足KKT条件。

凸优化问题的定义是：**求取最小值的目标函数为凸函数的一类优化问题。**目标函数是凸函数我们已经知道，这个优化问题又是求最小值。所以我们的最优化问题就是凸优化问题。

接下里，就是探讨是否满足KKT条件了。

**7)KKT条件**

我们已经使用拉格朗日函数对我们的目标函数进行了处理，生成了一个新的目标函数。通过一些条件，可以求出最优值的必要条件，这个条件就是接下来要说的KKT条件。一个最优化模型能够表示成下列标准形式：

$$\begin{aligned} &\min f(\boldsymbol{x}) \\ &s.t. \quad h_j(\boldsymbol{x}) = 0, j = 1, 2, \cdots, p \\ &\qquad g_k(\boldsymbol{x}) \leq 0, k = 1, 2, \cdots, q \\ &\qquad \boldsymbol{x} \in X \subset \mathbb{R}^n \end{aligned}$$

KKT条件的全称是Karush-Kuhn-Tucker条件，KKT条件是说最优值条件必须满足以下条件：

- 条件一：经过拉格朗日函数处理之后的新目标函数L(w,b,α)对x求导为零：
- 条件二：$\boldsymbol{h_j(x) = 0}$;
- 条件三：$\boldsymbol{\alpha * g_k(k) = 0}$;

对于我们的优化问题：

$$\begin{aligned} &\min \frac{1}{2} ||\omega||^2 \\ &\text{s.t.} \quad y_i\left(\boldsymbol{\omega}^T x_i + b\right) \geq 1, i = 1,2, \ldots, n \end{aligned}$$

显然，条件二已经满足了。另外两个条件为啥也满足呢？

感兴趣的可以移步这里：http://blog.csdn.net/xianlingmao/article/details/7919597

现在，凸优化问题和KKT都满足了，问题转换成了对偶问题。而求解这个对偶学习问题，可以分为三个步骤：首先要让L(w,b,α)关于w和b最小化，然后求对α的极大，最后利用SMO算法求解对偶问题中的拉格朗日乘子。