```
"""
Created on 2017/8/4
machine-learning-course
@author: DSG
"""
```

# Support Vector Machines

如果数据在某一维度上线性不可分，那么只要提升他的维度，就会线性可分。

## 线性分类器引入：

给定一些数据点，它们分别属于两个不同的类，现在要找到一个线性分类器把这些数据分成两类。如果用 $x$ 表示数据点，用 $y$ 表示类别（ $y$ 可以取 1 或者-1，分别代表两个不同的类），一个线性分类器的学习目标便是要在 $n$ 维的数据空间中找到一个超平面（hyper plane），这个超平面的方程可以表示为（ $w^T$ 中的 $T$ 代表转置）：

$$w^T x+b=0$$

在逻辑回归中我们构建 $\theta^T x=\theta_0 x_0+\theta_1 x_1+\theta_2 x_2+\cdots+\theta_n x_n$，通过学习得到向量 $\theta$ 的值，就可以得到 logistics 分类器，现在我们对 logistics 模型做一点小小的变化。首先，将使用的结果标签 $y=0$ 和 $y=1$ 替换为 $y=0, y=-1$，然后将 $\theta^T x=\theta_0 x_0+\theta_1 x_1+\theta_2 x_2+\cdots+\theta_n x_n$ $(x_0=1)$ 中的 $\theta_0$ 替换为 $b$，最后把 $\theta_1 x_1+\theta_2 x_2+\cdots+\theta_n x_n$ 替换为 $w^T x$。这样就有了 $f(x)=w^T x+b$，这是一个线性分类器的模型，也就是说除了 $y$ 从 $y=0$ 变为了 $y=-1$ 之外，线性分类函数和 logistics 回归的形式 $h_\theta(x)=g(\theta^T x)=g(w^T x+b)$ 没有太大的区别。

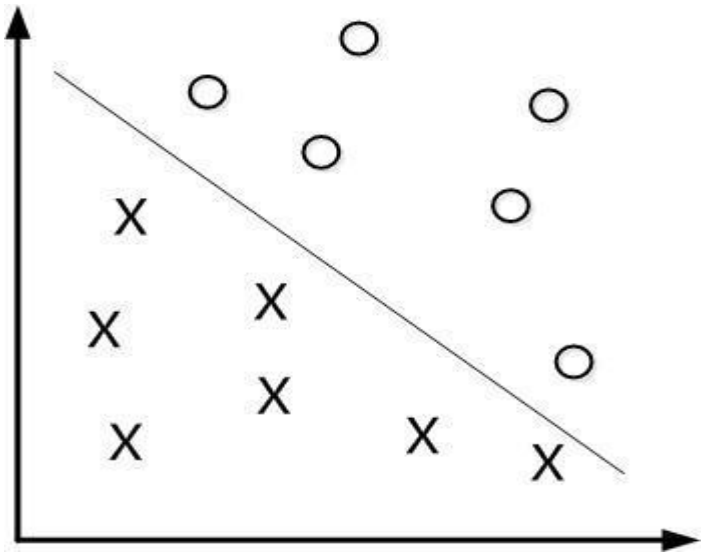
接下来我们可以将假设函数 $h_{\theta,b}(x)=g(\theta^T x+b)$ 中 $g(z)$ 做一个简化，将他映射到 $y=-1$ 和 $y=1$ 上，映射关系如下：

$$g(x)=\begin{cases}1 & z\geq 0\\ -1 & z\leq 0\end{cases}$$

## SVM 引入

先看一个简单的例子，如下图所示，现在有一个二维平面，平面上有两种不同的数据，分别用圈和叉表示。由于这些数据是线性可分的，所以可以用一条直线将这两类数据分开，这条直线就相当于一个超平面，超平面一边的数据点所对应的 $y$ 全是 -1 ，另一边所对应的 $y$ 全是 1。

图 1



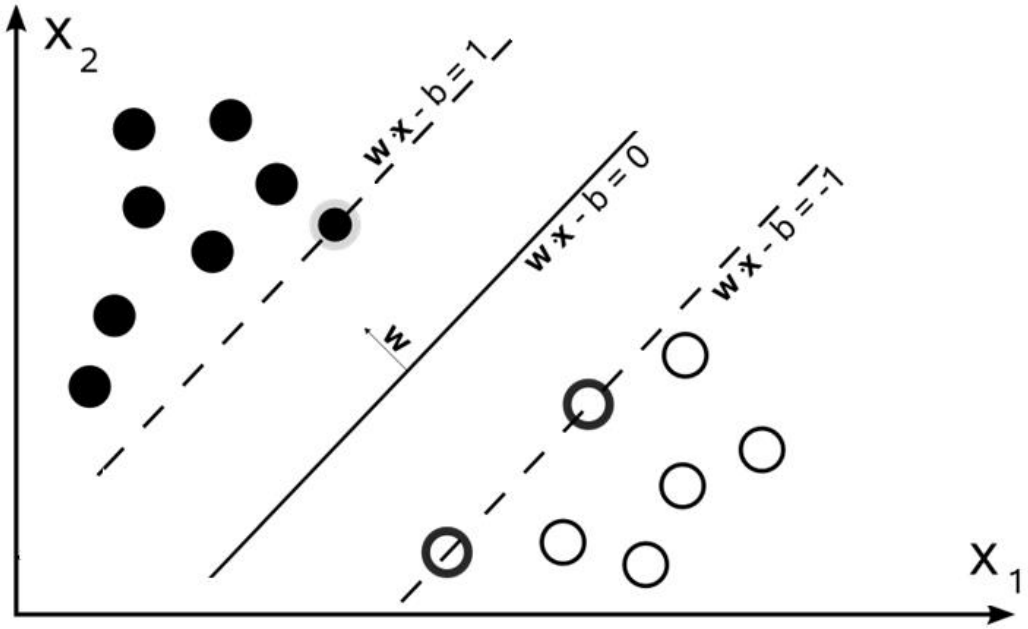
这个超平面可以用分类函数 $f(x)=w^Tx+b$ 表示，当 $f(x)$ 等于 0 的时候， $x$ 便是位于超平面上的点，而 $f(x)\geq 0$ 的点对应一个类别的数据点， $f(x)\leq 0$ 的点对应另外一个类别的数据点，如果这样可以发现有无数多的分割面都满足这样的需求，那怎样的分割面才算是最好的呢?

## 需求的数学模型

对一个数据点进行分类，当超平面和数据点的"间隔"越大，分类的确信度（confidence）也越大。所以，为了使得分类的确信度尽量高，需要让所选择的超平面能够最大化这个"间隔"值，我们需要这个超平面最大的分隔这两类。也就是这个超平面到这两个类的最近的那个样本的距离最大，两者都最大，所以相同，如图 2 所示。

我们称距离分割面距离为 1 的数据点为支撑向量，实质上就可以发现分割面实质上使用支撑向量确定的，因此支撑向量在 SVM 中起着至关重要的作用。这里的间隔就是指不同类别中支撑向量到分割面的距离。令分割面的方程为 $w^Tx+b=0$ 。

图 2

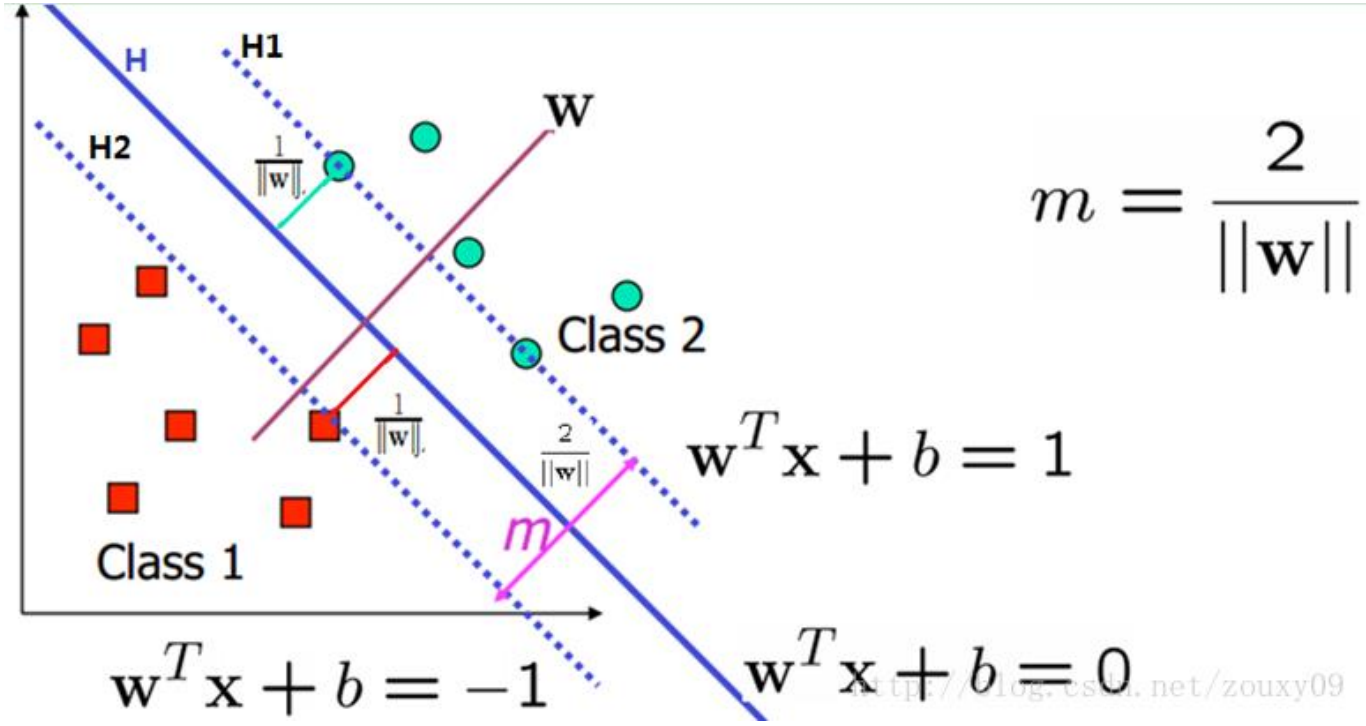



### 1. 用支持向量建立需求模型：

我们可以用不同类别中的支撑向量来固定两个平面 $H_1$ 和 $H_2$，$H_1$ 和 $H_2$ 的特点是他们之间没有任何数据点，同时在他们上面的数据点结果都输出-1 和 1，他们到 H 的距离是相等的。因此可以用 $H_1$ 和 $H_2$ 来描述我们的需求可以单的描述为：我们需要最大 $H_1$ 和 $H_2$ 到 $H$ 的距离且他们的距离相等，同时保证没有任何数据点在这两个分割面之间。

我们初中就学过，两条平行线的距离的求法，例如 $ax+by=c_1$ 和 $ax+by=c_2$，那他们的距离是 $\frac{|c_2-c_1|}{\sqrt{a^2+b^2}}$（值得注意的是，这里的 $x,y$ 是二维坐标系，而在我们的数据中他们分别表示数据的不同维度）。而用 $w$ 来表示就是 $H_1:w_1x_1+w_2x_2+\cdots+w_nx_n=1$，$H_2:w_1x_1+w_2x_2+\cdots+w_nx_n=1$ 和 $H:w_1x_1+w_2x_2+\cdots+w_nx_n=0$，那 $H_1$ 和 $H_2$ 到 $H$ 的距离就是 $\frac{|\pm 1-0|}{\sqrt{w_1^2+w_2^2+\cdots+w_n^2}}=\frac{1}{\|w\|}$（其中 $x_1,x_2,\cdots,x_n$ 表示数据的不同特征，也就是我们模型中的不同纬度，$w_n$ 分别是对应维度的系数，$\|w\|$ 为 $w$ 的二阶范数，范数是一个类似于模的表示长度的概念）。也就是说，我们需要最大化 $margin=\frac{1}{\|w\|}$ 。

图 3




## 2. 用几何间隔建立需求模型：

在超平面 $w^T x+b=0$ 确定的情况下，而通过观察 $w^T x+b$ 的符号与类标记 $y$ 的符号是否一致可判断分类是否正确，所以，可以用 $(w^T x^{(i)}+b)$ 的正负性来判定或表示分类的正确性。而 $\left|w^T x+b\right|$ 能够表示点 $x$ 到距离超平面的远近，所以样本点 $\left(x_1, x_2\right)$ 与超平面之间 $w^T x+b=0$ 的函数间隔定义为

$$\gamma_i = y(w^T x+b) = yf(x)$$

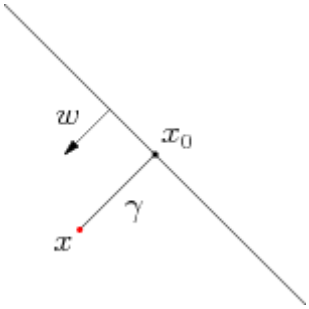
而超平面 $w^T x+b=0$ 关于数据集中所有样本点的函数间隔最小值便为超平面关于训练数据集的函数间隔：

$$\gamma = \min \gamma_{(i)},\ \ i=1,2,\cdots,n$$

函数间隔不适合用来最大化间隔值，因为在超平面固定以后，可以等比例地缩放 $w$ 的长度和 $b$ 的值，这样可以使得 $f(x)=w^T x+b$ 的值任意大，亦即函数间隔 $\gamma$ 可以在超平面保持不变的情况下被取得任意大。

假定对于一个点 $x$ ，令其垂直投影到超平面上的对应点为 $x_0$ ， $w$ 是垂直于超平面的一个向量， $\gamma$ 为样本 $x$ 到超平面的距离，如图 4 所示：

图 4




根据平面几何知识，有

$$x = x_0 + \gamma \frac{w}{\|w\|}$$

上述公式可以简单的理解为， $x_0$ 为一个向量的终点， $\frac{w}{\|w\|}$ 为单位向量， $\gamma \frac{w}{\|w\|}$ 就为为另外一个向量， $x$ 为一个向量的终点，那么整个式子就是两个向量求和。

又由于 $x_0$ 是超平面上的点，满足 $f(x_0)=0$，代入超平面的方程 $w^Tx+b=0$，可得 $w^Tx_0+b=0$，即 $w^Tx_0=-b$。随即让此式 $x=x_0+\gamma\frac{w}{\|w\|}$ 的两边同时乘以 $w^T$，再根据 $w^Tx_0=-b$ 和 $w^Tw=\|w\|^2$，即可算出：

$$\gamma=\frac{w^Tx+b}{\|w\|}=\frac{f(x)}{\|w\|}$$

为了得到 $\gamma$ 的绝对值，令 $\gamma$ 乘上对应的类别 $y$，即可得出几何间隔（用 $\hat{\gamma}$ 表示）的定义：

$$\hat{\gamma}=y\gamma=\frac{yf(x)}{\|w\|}$$

从上述函数间隔和几何间隔的定义可以看出：几何间隔就是函数间隔除以 $\|w\|$，而且函数间隔 $y(w^Tx+b)=yf(x)$ 实际上就是 $|f(x)|$，只是人为定义的一个间隔度量，而几何间隔 $\frac{|f(x)|}{\|w\|}$ 才是直观上的点到超平面的距离。

这里要找的最大间隔分类超平面中的“间隔”指的是几何间隔。通过上面的分析我们知道需求实质就是求支撑向量的 $\max\hat{\gamma}$。如果我们令 $yf(x)=1$，因此上面的公式就可以写为 $\hat{\gamma}=\frac{1}{\|w\|}$，那我们的需求就是求最大的 $\hat{\gamma}$，这个结果和我们用支持向量的到的结果相同。

## 3. 需求转化与约束条件

有上面的模型我们可以得到，关键就是求最大的间隔 $\frac{1}{\|w\|}$，为了最大化这个距离，我们应该最小化 $\|w\|$，因此我们的需求就是找出 $\min\|w\|$，因为 $\frac{1}{2}\|w\|^2$ 和 $\|w\|$ 具有相同的单调性和最优值，所以我们用 $\frac{1}{2}\|w\|^2$ 替换 $\|w\|$，理由是 $\frac{1}{2}\|w\|^2$ 在定义域内连续可导，而 $\|w\|$ 在对称轴出不连续不可导，处理起来麻烦。到此我们的需求就是 $\min\frac{1}{2}\|w\|^2$。

对于任何一个正样本 $y^{(i)}=+1$，它到分割面的几何间隔都要大于 1，也就是要保证：$y^{(i)}=w^Tx^{(i)}+b\geq 1$。对于任何一个负样本 $y^{(i)}=-1$，它到分割面的几何间隔也都要大于 1，也就是要保证：$y^{(i)}=w^Tx^{(i)}+b\leq 1$。因为我们把所有的类别规定为+1 和-1，所以这两个约束可以合并成同一个式子：$y^{(i)}(w^Tx^{(i)}+b)\geq 1$。这就是我们的条件。

最终我们的问题就变为：

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

$$s.t.\ y^{(i)}(w^Tx^{(i)}+b)\geq 1,\forall x^{(i)},\ i=1,2,\cdots,n$$

值得注意的是，这里的约束条件是 $n$ 个，一个数据点就代表一个条件。

# 问题对偶

因为现在的目标函数是二次的，约束条件是线性的，所以它是一个凸二次规划问题。由于这个问题的特殊结构，这里将通过拉格朗日对偶函数来求解，即通过求解与原问题等价的对偶问题（dual problem）得到原始问题的最优解，这就是线性可分条件下支持 svm 的对偶算法，这样做的优点在于：一者对偶问题往往更容易求解；二者可以自然的引入核函数，进而推广到非线性分类问题。有关于拉格朗日对偶函数的介绍见文档 LagrangeDuality.docx

构建拉个朗日函数为：

$$L(\vec{w},b,\vec{\alpha})=\frac{1}{2}\|w\|^2-\sum_{i=1}^{n}\alpha_i(y^{(i)}(w^Tx^{(i)}+b)-1)$$

然后令

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

我们的目标是求$\min\frac{1}{2}\|w\|^2$，则对应的目标函数为：

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

这里我们是在参数$w,b$下求得$p^*$，这里如果我们直接求解问题，就要先求关于$\alpha$的函数$\max_{\alpha_i\geq 0}L(w,b,\alpha)$（把$w,b$当做常量）在条件$\alpha_i\geq 0$下$\max_{\alpha_i\geq 0}L(w,b,\alpha)$最优解下的$\alpha$，然后把$\alpha$带入到原函数中最终得到一个关于$w,b$的函数$\theta(w,b)$，最后求$\min_{w,b}\theta(w,b)$，在这个步骤下问题最终转换为关于$w,b$两个参数的函数，意味着我们要求$\min_{w,b}\theta(w,b)$就得首先面临这两个参数，而且$\alpha$是不等约束的系数他有着自己的约束条件，第一步很难消掉，所以这个求解过程非常麻烦，所以这里我求对偶问题的解：

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

在这里我们首先求解$\min_{w,b}L(w,b,\alpha)$这是关于$w,b$的函数，求关于$w,b$函数的最优值，我么可以令$w,b$的梯度为0得到方程，然后用$\alpha$表示$w,b$后代入到原函数中得到关于$\alpha$的函数$D(\alpha)$，最后求解$\max_{\alpha_i\geq 0}D(\alpha)$。

现在由对偶的定理可以知道$d^*\leq p^*$，在某些条件下两者可以取等号，这里的条件就是KKT条件，在我们你的实际情况中KKT条件全部满足。

## 求解问题

问题经过上面的转换得到：

$$d^*=\max_{\alpha_i\geq 0}D(\vec{\alpha})=\max_{\alpha_i\geq 0}\min_{w,b}\left[\frac{1}{2}\|w\|^2-\sum_{i=1}^{n}\alpha_i(y^{(i)}(w^Tx^{(i)}+b)-1)\right]$$

首先要求$\min_{w,b}L(w,b,\alpha)$，就要固定$\alpha$，让$L$关于$w,b$最小化，然后分别对$w,b$求偏导数，最后令$\frac{\partial L}{\partial w}$和$\frac{\partial L}{\partial b}$等于零。

$$\frac{\partial L}{\partial w}=0\ \Rightarrow\ w-\sum_{i=1}^{n}\alpha_iy^{(i)}x^{(i)}=0\ \Rightarrow\ w=\sum_{i=1}^{n}\alpha_iy^{(i)}x^{(i)}$$

$$\frac{\partial L}{\partial b}=0\ \Rightarrow\ \sum_{i=1}^{n}\alpha_iy^{(i)}=0$$

将上面的结果代入到$L(w,b,\alpha)$，就可以得到$\min_{w,b}L(w,b,\alpha)$的结果

$$\begin{aligned}\min_{w,b}L(\vec{w},b,\vec{\alpha})&=\frac{1}{2}\|w\|^2-\sum_{i=1}^{n}\alpha_i(y^{(i)}(w^Tx^{(i)}+b)-1)\\&=\frac{1}{2}w^Tw-\sum_{i=1}^{n}\alpha_iy^{(i)}w^Tx^{(i)}+\sum_{i=1}^{n}\alpha_iy^{(i)}b+\sum_{i=1}^{n}\alpha_i\\&=w^T\left(\frac{1}{2}w-\sum_{i=1}^{n}\alpha_iy^{(i)}x^{(i)}\right)+b\sum_{i=1}^{n}\alpha_iy^{(i)}+\sum_{i=1}^{n}\alpha_i\\&=\left(\sum_{i=1}^{n}\alpha_iy^{(i)}x^{(i)}\right)^T\left(\frac{1}{2}\sum_{i=1}^{n}\alpha_iy^{(i)}x^{(i)}-\sum_{i=1}^{n}\alpha_iy^{(i)}x^{(i)}\right)+\sum_{i=1}^{n}\alpha_i\end{aligned}$$

$$= -\frac{1}{2}\sum_{i=1}^{n}\alpha_i y^{(i)}\left(x^{(i)}\right)^T\sum_{i=1}^{n}\alpha_i y^{(i)}x^{(i)} + \sum_{i=1}^{n}\alpha_i$$

$$= \sum_{i=1}^{n}\alpha_i - \frac{1}{2}\sum_{i,j=1}^{n}\alpha_i\alpha_j\left(x^{(j)}\right)^T x^{(i)}y^{(i)}y^{(j)}$$

代入后的对偶问题就是：

$$d^* = \max_{\alpha_i\geq 0} D(\vec{\alpha}) = \max_{\alpha_i\geq 0}\left[\sum_{i=1}^{n}\alpha_i - \frac{1}{2}\sum_{i,j=1}^{n}\alpha_i\alpha_j y^{(i)}y^{(j)}\left\langle x^{(i)}, x^{(j)}\right\rangle\right]$$

$$s.t.\ \ \alpha_i \geq 0,\ \sum_{i=1}^{n}\alpha_i y^{(i)} = 0,\ i = 1,2,\cdots n$$

而原始问题为：

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

$$s.t.\ \ y^{(i)}(w^T x^{(i)} + b) \geq 1, \forall x^{(i)},\ \ i = 1,2,\cdots n$$

现在求的$\alpha$之后根据 $w = \sum_{i=1}^{n}\alpha_i y^{(i)}x^{(i)}$ 求得$w$，然后代入到超平面的点里面既可以得到$b$：

$$b = -\frac{\max\limits_{i:y^{(i)}=-1} w^T x^{(j)} + \min\limits_{i:y^{(i)}=1} w^T x^{(j)}}{2}$$

有上面的分析知道 $w = \sum_{i=1}^{n}\alpha_i y^{(i)}x^{(i)}$，我们将这个带入到超平面方程中 $f(x) = w^T x + b$ 中可以得到：

$$f(x) = \left(\sum_{i=1}^{n}\alpha_i y^{(i)}x^{(i)}\right)^T x + b \qquad \text{公式 1}$$

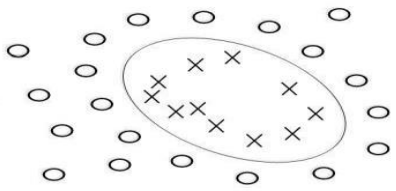
$$= \sum_{i=1}^{n}\alpha_i y^{(i)}\left\langle x^{(i)}, x\right\rangle + b$$

因此决策函数为 $f(x) = \sum_{i=1}^{n}\alpha_i y^{(i)}\left\langle x^{(i)}, x\right\rangle + b$，其实有这个式子就可以发现对于新的的数据点做决策，实质上只需要和训练集里的所有数据做内积就可以得到结果。在这里值得一提的是在 $\max\limits_{\alpha_i\geq 0} D(\alpha) = \max\limits_{\alpha_i\geq 0}\min\limits_{w,b}\left[\frac{1}{2}\|w\|^2 - \sum_{i=1}^{n}\alpha_i(y^{(i)}(w^T x^{(i)} + b) - 1)\right]$ 中，由上面的分析可以知道，支撑向量在上面的蓝色部分的计算结果为 0，而其他非支撑向量的计算结果都是大于 0 的，又因为$\alpha$是非负的。而结果是求最小值，所以只能让$\alpha$ =0，所以除了支撑向量外，其余的$\alpha$都为 0。所以在预测新数据点的时候只需要和支撑向量做内积就可以得到结果

## 处理线性不可分数据集——核函数

事实上，大部分时候数据并不是线性可分的，这个时候满足这样条件的超平面就根本不存在，如图 5:

图 5



[1] $\|w\|^2 = w^T w$

- 核函数

假设 $X_1, X_2$ 分别为二维平面的两个维度，现在有一个二次曲线的方程为：

$$a_1x_1 + a_2x_1^2 + a_3x_2 + a_4x_2^2 + a_5x_1x_2 + a_6 = 0$$

如果定义一个两维到五维的变换 $\mathbb{Z}: Z_1 = x_1,\ Z_2 = x_1^2,\ Z_3 = x_2,\ Z_4 = x_2^2,\ Z_5 = x_1x_2$，那么上面的曲线方程又可以写为：

$$a_1Z_1 + a_2Z_2 + a_3Z_3 + a_4Z_4 + a_5Z_5 + a_6 = 0$$

$$\sum_{i=1}^{5} a_iZ_i + a_6 = 0$$

在新坐标 $\mathbb{Z}$ 下原来的曲线方程变成线性的方程了，如果按照这个映射把原来的数据集映射到 5 维特征空间那么就可以继续使用 SVM 了。

如果用这个方法，在用线性分割器学习一个非线性关系时，需要选择一个非线性特征集（变换），并且将数据写成新的表达形式，这等价于用一个固定的非线性映射，将数据映射到其他的特征空间，在新特征空间中使用线性学习器，因此，考虑的假设数据集是线性不可分的，则它的决策规则类似以下类型：

$$f(x) = \sum_{i=1}^{n} w_i\phi(x^{(i)}) + b$$

这里 $\phi: X \to F$ 是从输入空间到某个特征空间的映射，这意味着建立非线性学习器分为两步：

1. 首先使用一个非线性映射将数据变换到一个特征空间 $F$，
2. 然后在特征空间使用线性学习器分类。

然而这种计算方式是非常低效的，比如最初的特征是 $n$ 维的，我们将其映射到 $n^2$ 维，然后再计算，这样需要 $O\left(n^2\right)$ 的时间，如果有一种方式可以在原特征空间中直接对输入数据点做某种计算就等价于计算映射后特征的内积，就像作用在原始输入数据点的函数一样，就有可能将两个步骤融合到一起建立一个非线性的学习器。**先看一个简单的例子：**

设两个向量 $x_1 = (\eta_1, \eta_2),\ x_2 = (\xi_1, \xi_2)$ 而 $\phi(X_1, X_2) = \left(\sqrt{2}X_1, X_1^2, \sqrt{2}X_2, X_2^2, \sqrt{2}X_1X_2, 1\right)^T$ 是一个 2 维到 6 维的映射，那么这两个向量在映射 $\phi(\cdot)$ 过后的内积为：

$$\langle\phi(x_1), \phi(x_2)\rangle = 2\eta_1\xi_1 + \eta_1^2\xi_1^2 + 2\eta_2\xi_2 + \eta_2^2\xi_2^2 + 2\eta_1\xi_1\eta_2\xi_2 + 1$$

再来看一个关于向量的函数 $f(X_1, X_2) = \left(\langle X_1, X_2\rangle + 1\right)^2$，那么向量 $x_1 = (\eta_1, \eta_2),\ x_2 = (\xi_1, \xi_2)$ 在这个函数下的结果为：

$$f(x_1, x_2) = \left(\langle x_1, x_2\rangle + 1\right)^2 = \eta_1\xi_1 + \eta_1^2\xi_1^2 + 2\eta_2\xi_2 + \eta_2^2\xi_2^2 + 2\eta_1\xi_1\eta_2\xi_2 + 1$$

神奇的是他们的结果是一样的，那他们的区别在哪里呢?

1. 一个是映射到高维空间中，然后再根据内积的公式进行计算；
2. 而另一个则只是在原来的低维空间中进行计算就完到高维成映加计算，而不需要显式地写出映射后的结果。

像这样我们把这里**的计算两个向量在隐式映射过后的空间中的内积的函数叫做核函数** (Kernel Function)。

如果原始特征内积是 $\langle x_1, x_2\rangle$，映射后为 $\langle\phi(x_1), \phi(x_2)\rangle$，如果存在一个函数 $K$ 使得 $K(\vec{x}_1, \vec{x}_2) = \phi(\vec{x}_1)^T\phi(\vec{x}_2)$，那我们称 $K(\vec{x}_1, \vec{x}_2)$ 为核函数。其实在上面的 $f(X_1, X_2) = \left(\langle X_1, X_2\rangle + 1\right)^2$ 就是一个核函数。

在我们的 SVM 决策规则中 $f(x) = \sum_{i=1}^{n} \alpha_i y^{(i)} \langle x^{(i)}, x\rangle + b$，使用原来的方法，加入数据的映射后就为：

$$f(x) = \sum_{i=1}^{n} \alpha_i y^{(i)} \langle \phi\left(x^{(i)}\right), \phi(x)\rangle + b$$

需要先计算 $\phi\left(x^{(i)}\right)$ 和 $\phi(x)$，然后计算 $\langle \phi\left(x^{(i)}\right), \phi(x)\rangle$ 才行，如果使用核函数决策规则就变成了：

$$f(x) = \sum_{i=1}^{n} \alpha_i y^{(i)} K\left(x^{(i)}, x\right) + b$$

他们的效果完全相同，而且复杂度会大大的降低。将核函数应用到对偶问题中，上面的对偶问题就是：

$$d^* = \max_{\alpha_i \geq 0} D(\vec{\alpha}) = \max_{\alpha_i \geq 0} \left[ \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y^{(i)} y^{(j)} K\left(x^{(i)}, x^{(j)}\right) \right]$$

$$s.t.\ 0 \leq \alpha_i,\ \sum_{i=1}^{n} \alpha_i y^{(i)} = 0,\ i = 1, 2, \cdots n$$

这样一来计算的问题就算解决了，避开了直接在高维空间中进行计算，而结果却是等价的！当然，因为我们这里的例子非常简单，所以我可以手工构造出对应于 $\phi(\cdot)$ 的核函数出来，如果对于任意一个映射，想要构造出对应的核函数就很困难了。

## ● Mercer 定理

问题：怎么判定一个核函数是否有效性，即给定一个函数 $K$，我们能否使用 $K$ 来替代计算 $\phi(\vec{x}_1)^T \phi(\vec{x}_2)$，也就说，是否能够找出一个 $\phi$，使得对于所有的 $\vec{x}$ 和 $\vec{z}$，都有 $K(\vec{x}_1, \vec{x}_2) = \phi(\vec{x}_1)^T \phi(\vec{x}_2)$ ？

Mercer 定理：如果函数 $K$ 是 $\mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ 上的映射（也就是从两个 $n$ 维向量映射到实数域）。那么如果 $K$ 是一个有效核函数（也称为 Mercer 核函数），那么当且仅当对于训练样例 $\{\alpha_1, \alpha_2, \cdots, \alpha_n\}$，其相应的核函数矩阵是对称半正定的。

Mercer 定理表明为了证明 K 是有效的核函数，那么我们不用去寻找 $\phi$，定义矩阵 $K$ 的元素 $K_{ij} = K(\vec{x}_i, \vec{x}_j)$，这个矩阵式 $n*n$ 的而只需要在训练集上求出各个 $K_{ij}$（这个矩阵是对称的），然后判断矩阵 $K$ 是否是半正定（使用左上角主子式大于等于零等方法）即可，如果矩阵 $K$ 是半正定的那么 $K(\vec{x}_i, \vec{x}_j)$ 就称为半正定的函数。

## ● 常用核函数

a) 多项式核，显然刚才我们举的例子是这里多项式核的一个特例 $(R = 1, d = 2)$。虽然比较麻烦，而且没有必要，不过这个核所对应的映射实际上是可以写出来的，该空间的维度是 $\binom{m+d}{d}$，其中 $m$ 是原始空间的维度。（不常用）

b) 高斯核 $k(x_1, x_2) = \exp\left(-\frac{\|x_1 - x_2\|^2}{2\sigma^2}\right)$，这时，如果 x 和 z 很相近 $\|x_1 - x_2\| \approx 0$，那么核函数值为 1，如果 x 和 z 相差很大 $\|x_1 - x_2\| \geq 0$，那么核函数值约等于 0。由于这个函数类似于高斯分布，因此称为高斯核函数，也叫做径向基函数(Radial Basis Function 简称 RBF)。它能够把原始特征映射到无穷维。不过，如果 $\sigma$ 选得很大的话，高次特征上的权重实际上衰减得非常快，所以实际上（数值上近似一下）相当于一个低维的子空间；反过来，如果 $\sigma$ 选得很小，则可以将任意的数据映射为线性可分——当然，这并不一定是好事，因为随之而来的可能是非常严重的过拟合问题。不过，总的来说，通过调控参数 $\sigma$，高斯核实际上具有相当高的灵活性，也是使用最广泛的核函数之一。下图所示的例子便是把低维线性不可分的数据通过高斯核函数映射到了高维空间，图 6 是高斯函数的图像。

图 6

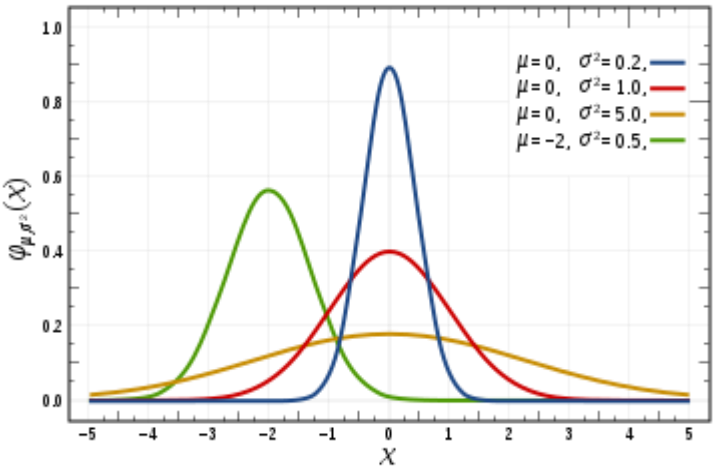
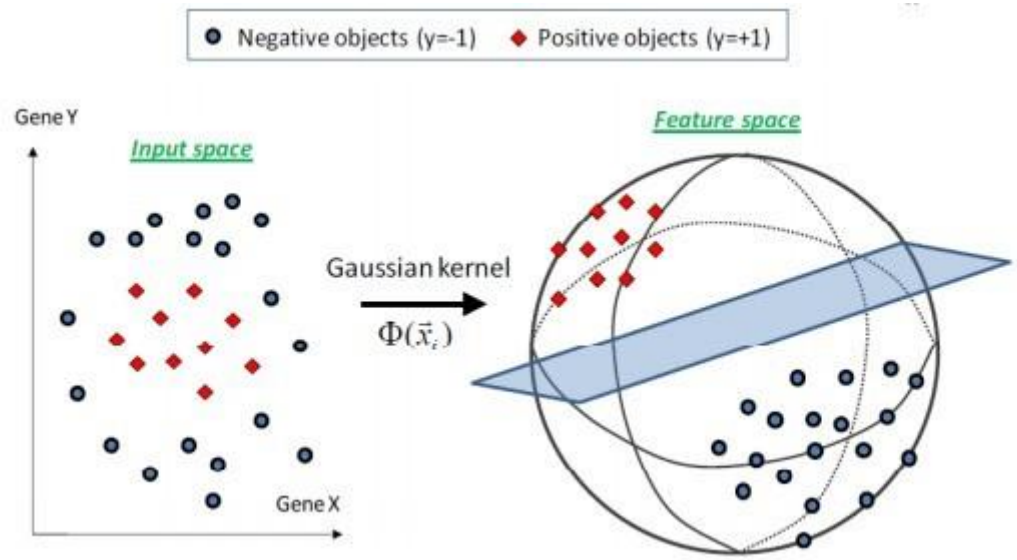



图 7 是使用高斯核函数建立的超平面。

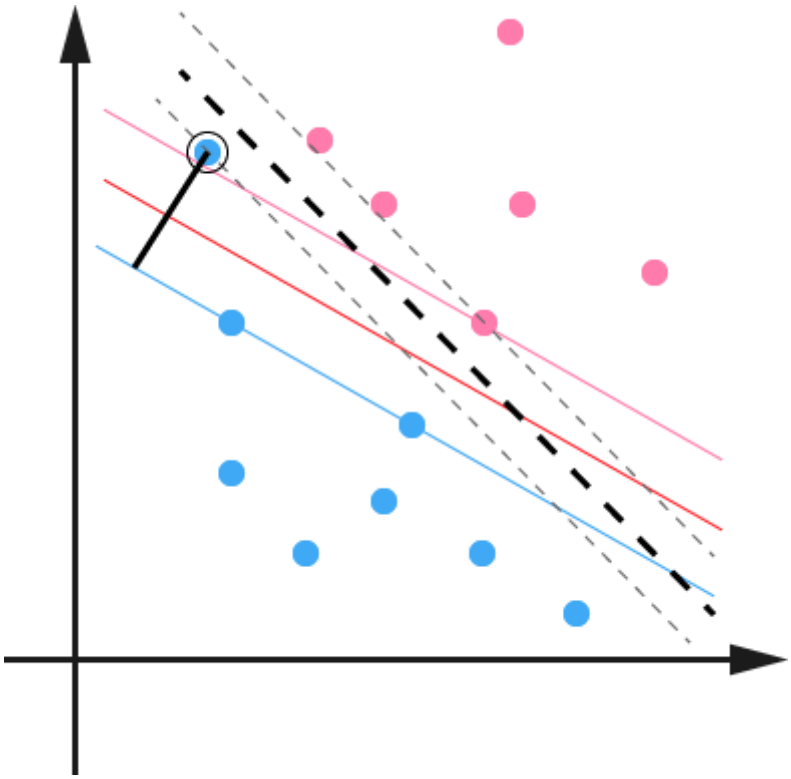
图 7




## 使用松弛变量处理 outliers

实际情况中数据集中可能有噪音存在，就算使用核函数也不能完全处理。对于这种偏离正常位置很远的数据点，我们称之为 outlier ，在我们原来的 SVM 模型里，outlier 的存在有可能造成很大的影响，因为超平面本身就是只有少数几个支撑向量组成的，如果这些支撑向量里又存在 outlier 的话，其影响就很大了。如图 8：

图 8



用黑圈圈起来的那个蓝点是一个 outlier ，它偏离了自己原本所应该在的那个半空间，如果直接忽略掉它的话，原来的分隔超平面还是挺好的，但是由于这个 outlier 的出现，导致分隔超平面不得不被挤歪了，变成途中黑色虚线所示（这只是一个示意图，并没有严格计算精确坐标），同时 margin 也相应变小了。当然，更严重的情况是，如果这个 outlier 再往右上移动一些距离的话，我们将无法构造出能将数据分开的超平面来。

为了处理这种情况，SVM 允许数据点在一定程度上偏离一下超平面。例如上图中，黑色实线所对应的距离，就是该 outlier 偏离的距离，如果把它移动回来，就刚好落在原来的超平面蓝色间隔边界上，而不会使得超平面发生变形了。

原来的问题中的约束条件是：

$$y^{(i)}(w^T x^{(i)} + b) \geq 1, \forall x^{(i)},\ i = 1, 2, \cdots n$$

这个约束条件的意思是离分隔面最近的样本点函数间隔也要比 1 大,在这种硬间隔下,有可能无法构建出超平面，现在考虑到 outlier 问题，我们引入松弛变量 (slack variable) $\xi$，约束条件改为：

$$y^{(i)}(w^T x^{(i)} + b) \geq 1 - \xi_i,\ \forall x^{(i)},\ \xi_i \geq 0,\ i = 1, 2, \cdots n$$

$\xi$ 的作用是允许对应的数据点 $x^{(i)}$ 和超平面的距离不为 1，$\xi_i$ 的数值表示对应的数据点 $x^{(i)}$ 允许偏离的函数间隔大小。

引入松弛变量后，虽然允许一步分数据点可以在支持向量和超平面之间，甚至可以在另一个类别的一边，因此就有可能出现过于松弛，对于任意的超平面都是符合条件的了，所以可以通过限制这些 $\xi$ 的总和最小（对于一个数据点尽可能的不使用这个变量），在原来的目标函数上加上这个限制：

$$\min\left[\frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\xi_i\right]$$

其中 $C$ 是一个常数，用于控制目标函数中两项（“寻找 margin 最大的超平面”和“保证数据点偏差量最小”）之间的权重。注意，其中 $\xi$ 是需要优化的变量（之一），而 $C$ 是一个事先确定好的常量。加入松弛变量后的需求为：

$$\begin{aligned}
&\min\left[\frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\xi_i\right] \\
&s.t.\ \ y^{(i)}(w^T x^{(i)} + b) \geq 1 - \xi_i, \forall x^{(i)} \\
&\xi_i \geq 0 \\
&i = 1, 2, \cdots, n
\end{aligned}$$

用之前的方法将限制或约束条件加入到目标函数中，得到新的拉格朗日函数为：

$$L(\vec{w},b,\vec{\alpha},\vec{\xi},\vec{r})=\frac{1}{2}\|w\|^2+C\sum_{i=1}^{n}\xi_i-\sum_{i=1}^{n}\alpha_i(y^{(i)}(w^Tx^{(i)}+b)-1+\xi_i)-\sum_{r_i\geq 0,i=1}^{n}r_i\xi_i$$

分析方法和前面一样，转换为另一个问题之后，我们先让 $L$ 针对 $w,b,\xi$ 最小化：

$$\frac{\partial L}{\partial b}=0 \quad \Rightarrow \quad \sum_{i=1}^{n}\alpha_i y^{(i)}=0$$

$$\frac{\partial L}{\partial \xi}=0 \quad \Rightarrow \quad C-\alpha_i-r_i=0$$

$$i=1,2,\cdots,n$$

将 $w$ 代回 $L$ 中得到和原来一样的结果：

$$\max_{\alpha_i\geq 0}D(\vec{\alpha})=\max_{\alpha_i\geq 0}\left[\sum_{i=1}^{n}\alpha_i-\frac{1}{2}\sum_{i,j=1}^{n}\alpha_i\alpha_j y^{(i)}y^{(j)}\left\langle x^{(i)},x^{(j)}\right\rangle\right]$$

不过，由于我们得到 $C-\alpha_i-r_i=0$ 而又有 $r_i\geq 0$（作为 Lagrange multiplier 的条件），因此有 $\alpha_i\leq C$，所以整个 dual 问题现在写作：

$$d^*=\max_{\alpha_i\geq 0}D(\vec{\alpha})=\max_{\alpha_i\geq 0}\left[\sum_{i=1}^{n}\alpha_i-\frac{1}{2}\sum_{i,j=1}^{n}\alpha_i\alpha_j y^{(i)}y^{(j)}\left\langle x^{(i)},x^{(j)}\right\rangle\right] \quad s.t.\ 0\leq\alpha_i\leq C,\ \sum_{i=1}^{n}\alpha_i y^{(i)}=0,\ i=1,2,\cdots,n$$

引入松弛变量前的对偶问题为：

$$d^*=\max_{\alpha_i\geq 0}D(\vec{\alpha})=\max_{\alpha_i\geq 0}\left[\sum_{i=1}^{n}\alpha_i-\frac{1}{2}\sum_{i,j=1}^{n}\alpha_i\alpha_j y^{(i)}y^{(j)}\left\langle x^{(i)},x^{(j)}\right\rangle\right] \quad s.t.\ \alpha_i\geq 0,\ \sum_{i=1}^{n}\alpha_i y^{(i)}=0,\ i=1,2,\cdots,n$$

唯一的区别就是 $\alpha_i$ 多了一个上限 $C$。加入核函数就得到了最终的对偶问题：

$$d^*=\max_{\alpha_i\geq 0}D(\vec{\alpha})=\max_{\alpha_i\geq 0}\left[\sum_{i=1}^{n}\alpha_i-\frac{1}{2}\sum_{i,j=1}^{n}\alpha_i\alpha_j y^{(i)}y^{(j)}K\left(x^{(i)},x^{(j)}\right)\right]$$

$$s.t.\ 0\leq\alpha_i\leq C,\ \sum_{i=1}^{n}\alpha_i y^{(i)}=0,\ i=1,2,\cdots n$$

# 使用 SMO 算法训练 $\alpha$

- 参数更新

到现在,一个强大完美的 SVM 已经形成了，通过上面的分析可以知道一条数据对应一个 $\alpha$，但是随着训练集数据量增大一下训练这么多 $\alpha$ 也不是一件简单的事， 1998 年，Microsoft Research 的 John C. Platt 在论文《Sequential Minimal Optimization：A Fast Algorithm for Training Support Vector Machines》中提出针对上述问题的解法 :SMO 算法，它很快便成为最快的二次规划优化算法，特别是在针对线性 SVM 和数据稀疏时性能更优。

在使用 SMO 之前还得对我们的最终的问题做一点点小小的变化，原问题为：

$$\min_{w,b}\max_{\alpha_i\geq 0}L(\vec{w},b,\vec{\alpha},\vec{\xi},\vec{r})=\min_{w,b}\max_{\alpha_i\geq 0}\left[\frac{1}{2}\|w\|^2-\sum_{i=1}^{n}\alpha_i(y^{(i)}(w^Tx^{(i)}+b)-1+\xi_i)-\sum_{r_i\geq 0,i=1}^{n}r_i\xi_i+C\sum_{i=1}^{n}\xi_i\right]$$

原对偶问题为：

$$\max_{\alpha_i\geq 0}L(\vec{w},b,\vec{\alpha},\vec{\xi},\vec{r})=\max_{\alpha_i\geq 0}\min_{w,b}\left[\frac{1}{2}\|w\|^2-\sum_{i=1}^{n}\alpha_i(y^{(i)}(w^Tx^{(i)}+b)-1+\xi_i)-\sum_{r_i\geq 0,i=1}^{n}r_i\xi_i+C\sum_{i=1}^{n}\xi_i\right]$$

经过处理转化为：

$$d^* = \max_{\alpha_i \geq 0} D(\vec{\alpha}) = \max_{\alpha_i \geq 0} \left[ \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y^{(i)} y^{(j)} K\left(x^{(i)}, x^{(j)}\right) \right]$$

$$s.t.\ 0 \leq \alpha_i \leq C,\ \sum_{i=1}^{n} \alpha_i y^{(i)} = 0,\ i = 1, 2, \cdots n$$

取负号后：

$$\min_{\alpha_i} \Psi(\vec{\alpha}) = \min_{\alpha_i} \left[ \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y^{(i)} y^{(j)} K\left(x^{(i)}, x^{(j)}\right) - \sum_{i=1}^{n} \alpha_i \right]$$

$$s.t.\ 0 \leq \alpha_i \leq C,\ \sum_{i=1}^{n} \alpha_i y^{(i)} = 0,\ i = 1, 2, \cdots n$$

某一数据点的决策函数为 $f(x) = \sum_{i=1}^{n} \alpha_i y^{(i)} K(x^{(i)}, x) + b$，现在的问题中与之对应的 KKT 条件为：

$$\alpha_i = 0 \ \Leftrightarrow\ y_i f(x_i) \geq 1 - \xi_i,$$
$$0 < \alpha_i < C \ \Leftrightarrow\ y_i f(x_i) = 1 - \xi_i,$$
$$\alpha_i = C \ \Leftrightarrow\ y_i f(x_i) \leq 1 - \xi_i.$$

原始约束条件：

$$s.t.\ -y^{(i)}(w^T x^{(i)} + b) - 1 + \xi_i \leq 0, \forall x^{(i)},\ i = 1, 2, \cdots, n$$

其中 $\xi_i$ 为松弛参数，只有在离群点的 $\xi_i$ 不为 0，其余的 $\xi_i$ 都为 0。$C$ 由我们预先设定，也是已知数。这个 KKT 条件说明，在两条间隔线外面的点，对应前面的系数 $\alpha_i$ 为 0，在两条间隔线里面的对应 $\alpha_i$ 为 $C$，在两条间隔线上的对应的系数 $\alpha_i$ 在 0 和 $C$ 之间。

要解决的是在参数 $\{\alpha_1, \alpha_2, \cdots, \alpha_n\}$ 上求 $\Psi(\vec{\alpha})$ 最小值的问题，至于 $x^{(i)}, y^{(i)}$ 都是已知数。按照坐标上升的思路，我们首先固定除 $\alpha_1$ 以外的所有参数，然后在 $\alpha_1$ 上求极值。等一下，这个思路有问题，因为问题中规定了 $\sum_{i=1}^{n} \alpha_i y^{(i)} = 0$，如果固定 $\alpha_1$ 以外的所有参数，那么 $\alpha_1$ 将不再是变量（可以由其他值推出）。因此，我们需要一次选取两个参数做优化，比如 $\alpha_1$ 和 $\alpha_2$，此时 $\alpha_2$ 可以由 $\alpha_1$ 和其他参数表示出来。这样回带到 $\Psi(\vec{\alpha})$ 中，$\Psi(\vec{\alpha})$ 就只是关于 $\alpha_1$ 的函数了：

$$\Psi(\alpha_1, \alpha_2) = \frac{1}{2} \alpha_1 y^{(1)} \sum_{j=1}^{n} \alpha_j y^{(j)} K\left(x^{(i)}, x^{(j)}\right) + \frac{1}{2} \alpha_2 y^{(2)} \sum_{j=1}^{n} \alpha_j y^{(j)} K\left(x^{(i)}, x^{(j)}\right) + \frac{1}{2} \sum_{i=3}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y^{(i)} y^{(j)} K\left(x^{(i)}, x^{(j)}\right) - \sum_{i=1}^{n} \alpha_i$$

$$\Psi(\alpha_1, \alpha_2) = \frac{1}{2} \alpha_1 y^{(1)} \sum_{j=1}^{n} \alpha_j y^{(j)} K\left(x^{(i)}, x^{(j)}\right) + \frac{1}{2} \alpha_2 y^{(2)} \sum_{j=1}^{n} \alpha_j y^{(j)} K\left(x^{(i)}, x^{(j)}\right) + \frac{1}{2} \sum_{i=3}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y^{(i)} y^{(j)} K\left(x^{(i)}, x^{(j)}\right) - \sum_{i=1}^{n} \alpha_i$$

$$\Psi(\alpha_1, \alpha_2) = \frac{1}{2} \alpha_1^2 (y^{(1)})^2 K\left(x^{(1)}, x^{(1)}\right) + \frac{1}{2} \alpha_2^2 (y^{(2)})^2 \left(x^{(2)}, x^{(2)}\right) + \alpha_1 y^{(1)} \alpha_2 y^{(2)} K\left(x^{(1)}, x^{(2)}\right) +$$
$$\frac{1}{2} \alpha_1 y^{(1)} \sum_{j=3}^{n} \alpha_j y^{(j)} K\left(x^{(1)}, x^{(j)}\right) + \frac{1}{2} \alpha_2 y_2 \sum_{j=3}^{n} \alpha_j y^{(j)} K\left(x^{(2)}, x^{(j)}\right) + \frac{1}{2} \alpha_1 y^{(1)} \sum_{j=3}^{n} \alpha_j y^{(j)} K\left(x^{(i)}, x^{(1)}\right) +$$
$$\frac{1}{2} \alpha_2 y^{(2)} \sum_{j=3}^{n} \alpha_j y^{(j)} K\left(x^{(j)}, x^{(2)}\right) + \frac{1}{2} \sum_{i=3}^{n} \sum_{j=3}^{n} \alpha_i \alpha_j y^{(i)} y^{(j)} K\left(x^{(i)}, x^{(j)}\right) - \alpha_1 - \alpha_2 + \sum_{i=3}^{n} \alpha_i$$

$$\Psi(\alpha_1, \alpha_2) = \frac{1}{2} \alpha_1^2 K\left(x^{(1)}, x^{(1)}\right) + \frac{1}{2} \alpha_2^2 \left(x^{(2)}, x^{(2)}\right) + \alpha_1 y^{(1)} \alpha_2 y^{(2)} K\left(x^{(1)}, x^{(2)}\right) +$$
$$\alpha_1 y^{(1)} \sum_{j=3}^{n} \alpha_i y^{(j)} K\left(x^{(j)}, x^{(1)}\right) + \alpha_2 y^{(2)} \sum_{j=3}^{n} \alpha_j y^{(j)} K\left(x^{(j)}, x^{(2)}\right) - \alpha_1 - \alpha_2 + \Psi_{Constant}$$

值得注意的是上面的展开中 $\Psi_{Constant}$ 表示所有不含 $\alpha_1, \alpha_2$ 项的总和，同时用到了以下的等式代换：

$$(y^{(i)})^2 = 1\ ;\ K\left(x^{(i)}, x^{(j)}\right) = K\left(x^{(j)}, x^{(i)}\right)\ ;\ \sum_{i=3}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y^{(i)} y^{(j)} K\left(x^{(i)}, x^{(j)}\right) = \sum_{i=1}^{n} \sum_{j=3}^{n} \alpha_i \alpha_j y^{(i)} y^{(j)} K\left(x^{(i)}, x^{(j)}\right)$$

令 $v_i=\sum_{j=3}^{n}\alpha_i y^{(j)}K\left(x^{(j)},x^{(i)}\right)$， $y^{(1)}y^{(2)}=s$ 则上式的展开结果为：

$$\Psi(\alpha_1,\alpha_2)=\frac{1}{2}\alpha_1^2K_{11}+\frac{1}{2}\alpha_2^2K_{22}+s\alpha_1\alpha_2K_{12}+\alpha_1y^{(1)}v_1+\alpha_2y^{(2)}v_2-\alpha_1-\alpha_2+\Psi_{Constant}$$

我们现在的问题就是：

$$\min\Psi(\alpha_1,\alpha_2)=\min[\frac{1}{2}\alpha_1^2K_{11}+\frac{1}{2}\alpha_2^2K_{22}+s\alpha_1\alpha_2K_{12}+\alpha_1y^{(1)}v_1+\alpha_2y^{(2)}v_2-\alpha_1-\alpha_2+\Psi_{Constant}]$$

其中 $K_{11}$ 为 $K\left(x^{(1)},x^{(1)}\right)$，如果引入 $f(x)$，则可以做以下变形：

$$f(x)=\sum_{i=1}^{n}\alpha_i y^{(i)}K(x^{(i)},x)+b$$

$$v_i=\sum_{j=3}^{n}\alpha_j y^{(j)}K\left(x^{(j)},x^{(i)}\right)=f(x)-b-\alpha_1^*y^{(1)}K_{1i}-\alpha_2^*y^{(2)}K_{2i}$$

其中 $\alpha_1^*,\alpha_2^*$ 为某次迭代之前的值, 即上次迭代之后的值， $f(x)$ 是一个已知的数字, 所以在得到的里面的为已知。

当固定了 $\alpha_1,\alpha_2$ 了之后，$\alpha_1,\alpha_2$ 满足下面的等式：

$$\alpha_1y^{(1)}+\alpha_2y^{(2)}=-\sum_{i=3}^{m}\alpha_i y^{(i)}$$

由于 $\{\alpha_1,\alpha_2,\cdots,\alpha_n\}$ 都是已知固定值，因此为了简单处理，可将等式右边标记成实数值 $\zeta$ 。

因此，如果假设选择的两个乘子 $\alpha_1,\alpha_2$，它们在更新之前分别是 $\alpha_1^{old},\alpha_1^{old}$，更新之后分别是 $\alpha_1^{new},\alpha_1^{new}$，那么更新前后的值需要满足以下等式才能保证 $\sum_{i=1}^{n}\alpha_i y^{(i)}=0$ ：

$$\alpha_1^{old}y^{(1)}+\alpha_2^{old}y^{(2)}=\alpha_1^{new}y^{(1)}+\alpha_2^{new}y^{(2)}=\zeta$$

两个因子不好同时求解，所以可先求第二个乘子 $\alpha_2$ 的解 $\alpha_2^{new}$，得到 $\alpha_2$ 的解 $\alpha_2^{new}$ 之后，再用 $\alpha_2$ 的解 $\alpha_2^{new}$ 表示 $\alpha_1$ 的解 $\alpha_1^{new}$。 为了求解 $\alpha_2^{new}$，得先确定 $\alpha_2^{new}$ 的取值范围。假设它的上下边界分别为 H 和 L，那么有：

$$L\le\alpha_2^{new}\le H$$

接下来，综合 $0\le\alpha_i\le C,\ i=1,2,\cdots,n$ 和 $\alpha_1^{old}y^{(1)}+\alpha_2^{old}y^{(2)}=\alpha_1^{new}y^{(1)}+\alpha_2^{new}y^{(2)}=\zeta$ 这两个约束条件，求取 $\alpha_2^{new}$ 的取值范围。

当 $y^{(1)}\ne y^{(2)}$ 时，根据 $\alpha_1^{old}y^{(1)}+\alpha_2^{old}y^{(2)}=\alpha_1^{new}y^{(1)}+\alpha_2^{new}y^{(2)}=\zeta$ 可得 $\alpha_1^{old}-\alpha_2^{old}=\zeta$，所以有 $L=\max(0,-\zeta)$， $H=\min\left(C,C-\zeta\right)$，如图 9 所示。

当 $y^{(1)}=y^{(2)}$ 时，根据 $\alpha_1^{old}y^{(1)}+\alpha_2^{old}y^{(2)}=\alpha_1^{new}y^{(1)}+\alpha_2^{new}y^{(2)}=\zeta$ 可得 $\alpha_1^{old}+\alpha_2^{old}=\zeta$，所以有 $L=\max(0,\zeta-C)$， $H=\min\left(C,\zeta\right)$，如图 10 所示。

如此，根据 $y^{(1)}$ 和 $y^{(2)}$ 异号或同号，可得出 $\alpha_2^{new}$ 的上下界分别为：

$$\begin{cases}L=\max(0,\alpha_2^{old}-\alpha_1^{old}),H=\min\left(C,C+\alpha_2^{old}-\alpha_1^{old}\right)\ if\ y^{(1)}\ne y^{(2)}\\ L=\max(0,\alpha_2^{old}+\alpha_1^{old}-C),H=\min\left(C,\alpha_2^{old}+\alpha_1^{old}\right)\ if\ y^{(1)}=y^{(2)}\end{cases}$$

公式 2

现在这个式子 $\alpha_1^{old}y^{(1)}+\alpha_2^{old}y^{(2)}=\alpha_1^{new}y^{(1)}+\alpha_2^{new}y^{(2)}=\zeta$ 两边同乘以 $y^{(1)}$ ：

$$\alpha_1+s\alpha_2=\alpha_1^*+s\alpha_2^*=-y^{(1)}\sum_{i=3}^{m}\alpha_i y^{(i)}$$

图 9

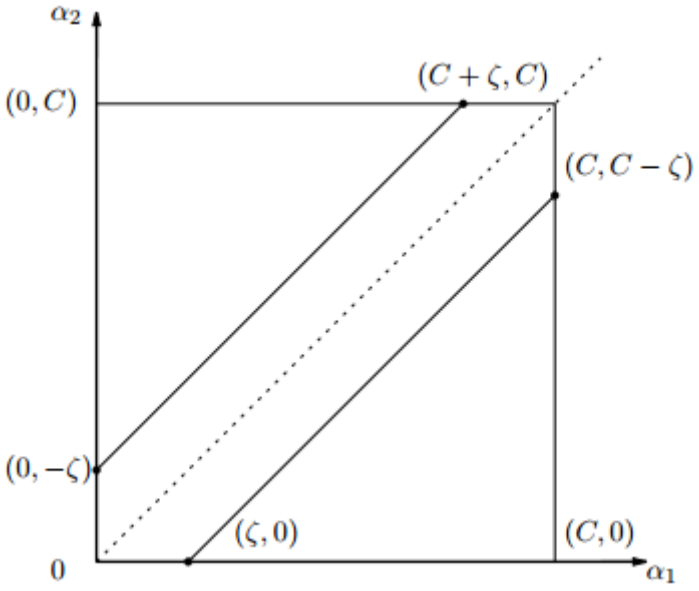
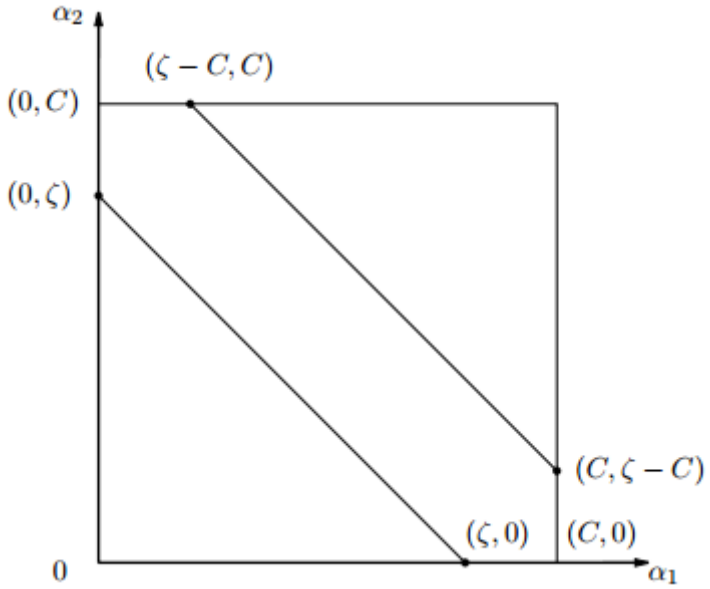



图 10




令 $-y^{(1)}\sum_{i=3}^{m}\alpha_i y^{(i)} = t$ ,则 $\alpha_1 + s\alpha_2 = \alpha_1^* + s\alpha_2^* = t$ ， 因此 $\alpha_1 = t - s\alpha_2$ ， 从而就把式子替换为只包含 $\alpha_2$ 的问题：

$$\min \Psi(\alpha_1, \alpha_2) = \min[\frac{1}{2}\alpha_1^2 K_{11} + \frac{1}{2}\alpha_2^2 K_{22} + s\alpha_1\alpha_2 K_{12} + \alpha_1 y^{(1)} v_1 + \alpha_2 y^{(2)} v_2 - \alpha_1 - \alpha_2 + \Psi_{Constant}]$$

$$\min \Psi(\alpha_2) = \min[\frac{1}{2}K_{11}(t - s\alpha_2)^2 + \frac{1}{2}\alpha_2^2 K_{22} + s\alpha_2 K_{12}(t - s\alpha_2) + y^{(1)} v_1 (t - s\alpha_2) + \alpha_2 y^{(2)} v_2 - t + s\alpha_2 - \alpha_2 + \Psi_{Constant}]$$

对 $\alpha_2$ 求导：

$$\frac{d\Psi}{d\alpha_2} = -sK_{11}(t - s\alpha_2) + K_{22}\alpha_2 - K_{12}\alpha_2 + sK_{12}(t - s\alpha_2) - y^{(2)} v_1 + s + y^{(2)} v_2 - 1$$

整理得：

$$(K_{11} + K_{22} - 2K_{12})\alpha_2 = s(K_{11} - K_{12})t + y^{(2)}(v_1 - v_2) + 1 - s$$

$$K_{ij} = K(x^{(i)}, x^{(j)})$$

将 $s = y^{(1)} y^{(2)}, \alpha_1 + s\alpha_2 = \alpha_1^* + s\alpha_2^* = t, v_i = \sum_{j=3}^{n}\alpha_j y^{(j)} K\left(x^{(j)}, x^{(i)}\right) = f(x) - b - \alpha_1^* y^{(1)} K_{1i} - \alpha_1^* y^{(2)} K_{2i}$ 带输入上式
得：

$$\alpha_2^{new}(K_{11}+K_{22}-2K_{12})=\alpha_2^{old}(K_{11}+K_{22}-2K_{12})+y^{(2)}\left[f(x_1)-f(x_2)+y^{(2)}-y^{(1)}\right]$$

令 $E_i=f(x_1)-y^{(1)}$（表示估计误差），$\eta=(K_{11}+K_{22}-2K_{12})$ 可以得到

$$\alpha_2^{new}=\alpha_2^{old}+\frac{y^{(2)}(E_1-E_2)}{\eta}$$ 公式 3

这个便是 $\alpha_2$ 的迭代公式，但是我们需要对 $\alpha_2^{new}$ 的取值进行约束才可以：

$$\alpha_2^{new,clipped}=\begin{cases}H & if \quad \alpha_2^{new}\geq H,\\ \alpha_2^{new} & if \quad L<\alpha_2^{new}<H,\\ L & if \quad \alpha_2^{new}\leq L.\end{cases}$$ 公式 4

现在只需要用坐标上升的方法求得 $\alpha_2$ 的最优值，然后求出 $\alpha_1$。

$$\alpha_1^{new}=\alpha_1+s(\alpha_2-\alpha_2^{new,clipped})$$ 公式 5

接下来就差 $b$ 的更新了，每计算两个 $\alpha$ 就要更新一次 $b$，$b$ 的更新公式如下：

$$b=\begin{cases}b_1 & if \quad 0<\alpha_1^{new}<c\\ b_2 & if \quad 0<\alpha_2^{new,clipped}<c\\ b_1=b_2 & if \quad 0<\alpha_1^{new}<c \quad and \quad 0<\alpha_2^{new,clipped}<c\\ \frac{b_1+b_2}{2} & otherwise\end{cases}$$ 公式 6

$$\begin{aligned}b_1&=E_1+y^{(1)}\left(\alpha_1^{new}-\alpha_1^{old}\right)K(\vec{x}_1,\vec{x}_1)+y^{(2)}\left(\alpha_2^{new,clipped}-\alpha_2^{old}\right)K(\vec{x}_1,\vec{x}_2)+b^{old}\\ b_1&=E_2+y^{(1)}\left(\alpha_1^{new}-\alpha_1^{old}\right)K(\vec{x}_1,\vec{x}_2)+y^{(2)}\left(\alpha_2^{new,clipped}-\alpha_2^{old}\right)K(\vec{x}_2,\vec{x}_2)+b^{old}\end{aligned}$$ 公式 7

前面两个的公式推导可以根据 $\alpha_1^{old}y^{(1)}+\alpha_2^{old}y^{(2)}=-\sum_{i=3}^{m}\alpha_i y^{(i)}=\alpha_1^{new}y^{(1)}+\alpha_2^{new,clipped}y^{(2)}$ 和对于 $0<\alpha_i<C$ 有 $y^{(i)}f(x^{(i)})=1$ 的 KKT 条件推出。

## ● SMO 中拉格朗日乘子的启发式选择方法

$\alpha_1$：先“扫描”所有乘子，把第一个违反 KKT 条件的作为更新对象，令为 a1；

$\alpha_2$：在所有不违反 KKT 条件的乘子中，选择使 $|E_1-E_2|$ 最大的 $\alpha_2$ 进行更新，使得能最大限度增大目标函数的值 $\alpha_2$ 值更新公式

最后，每次更新完两个乘子的优化后，都需要再重新计算 b，及对应的 Ei 值。