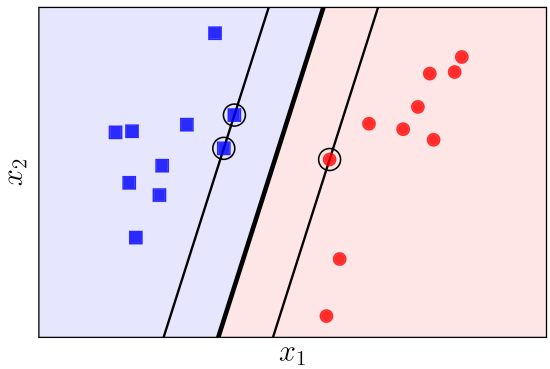
攀登传统机器学习的珠峰-SVM (中)

----------author：August助教（网易云课堂，机器学习微专业）----------

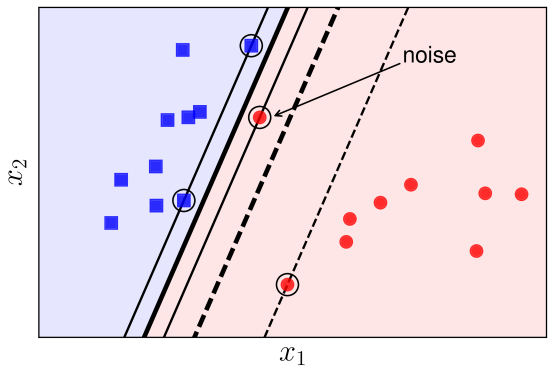
最大软间隔支持向量机

1. 线性分类SVM面临的问题

在上次课中，我们对线性可分SVM的算法的原理和流程进行了总结，如下图所示，为线性可分的数据集，我们可以采用线性可分的支持向量机，也称为硬间隔支持向量机。



当数据集中参杂了一些噪声，如下图所示，由于参杂了一个红色的噪声点，导致模型学习到的决策边界由下图中的粗虚线移动到了粗实线。



Q1：上图是粗实线作为决策边界更合理，还是粗虚线作为决策边界更合理？为什么？

很显然是粗虚线更合理，因为粗虚线忽略了噪音的影响，其margin更大，在测试集上的效果要优于粗实线。

如何解决这些问题呢？SVM引入了软间隔最大化的方法来解决。

2. 软间隔SVM

回顾下最大硬间隔的SVM：

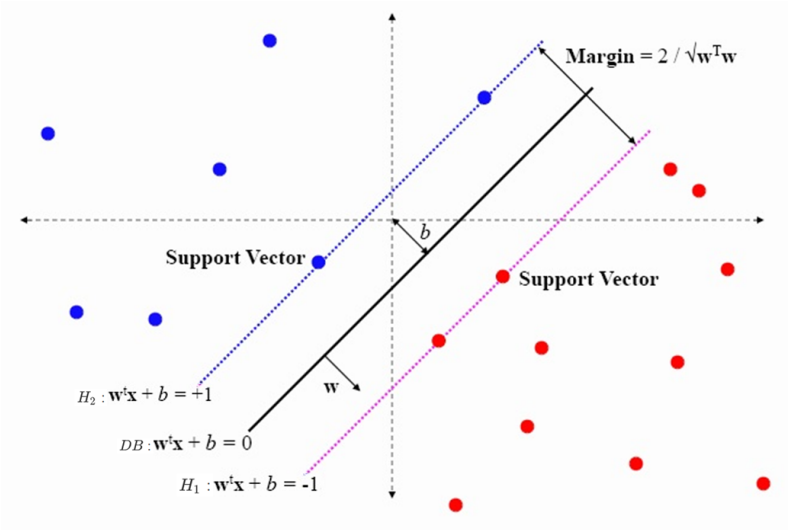
$$min\ \frac{1}{2}||w||_2^2 \quad s.t\ \ y_i(w^T x_i + b) \geq 1 (i = 1, 2, \dots m)$$

Q2：我们想想如何改造上式得到软间隔最大化呢？

上节课我们讲到，硬间隔SVM借鉴了EM的核心idea，包含两个核心的步骤：找到支持向量（样本下限），然后最大化支持向量间隔（最小化下限）。下面我们就这两个方面看看软间隔是怎样从硬间隔过渡而来的。

下限有什么变化

让我们回顾一下硬间隔支持向量机，如下图所示：

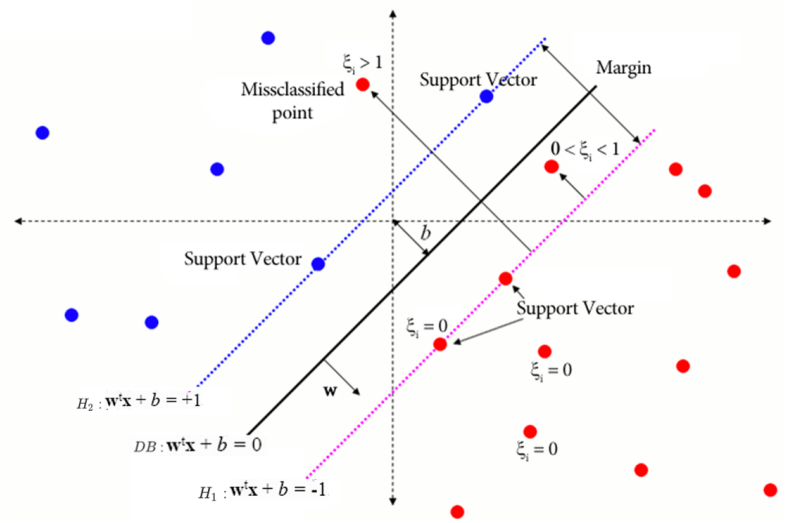


所有的样本点均到决策超平面的距离均不小于1，硬间隔SVM的下限为：

$$y_i(w^T x_i + b) \geq 1$$

H_1 和 H_2 像两堵墙一样将两类样本分隔开。

当数据中增加噪声点或误分类以后，比如在 H_1 和 H_2 之间，SVM 引入了一个神奇的变量 $\xi_i \geq 0$，这个变量被称为松弛变量，其几何含义由下图所示：



简单明了， 以红点样本为例， ξ 表示到 H_1 的距离：

当样本点 H_1 右边时，包括支持向量和支持向量以外的样本点时 $\xi_i = 0$，即我们可以按照硬间隔来处理。

对于H_1左侧的样本点，样本的$\xi > 0$， 包括两种情况，一种是正确分类，但是在 margin 范围内的样本点 (也就是超平面附近的点)，此时$0 < \xi < 1$； 一类是误分类的点，此时$\xi > 1$。

Q2:* $1-\xi$ *有什么样的几何含义？

对于噪声点和误分类的样本点来说，$1-\xi$表示该样本点到决策超平面之间的距离，而且是有向距离。具体而言，噪声点到超平面的距离为$0 < 1-\xi < 1$，误分类点到超平面的距离为$1-\xi < 0$。

加入松弛变量ξ之后，我们的H_1和H_2像弹簧一样，针对不同的样本点做不同的处理，变“软”了。写成数学公式为：

$$y_i(w^T x_i + b) \geq 1 - \xi_i, \quad \xi_i \geq 0$$

怎样最优化下限

我们知道，只要我们拉伸弹簧，我们就会消耗能量，付出代价。同理松弛变量的添加也是有成本的，每一个松弛变量ξ_i都支付了一个代价ξ_i， 现在代价函数变成了：

$$min \ \ \frac{1}{2}||w||_2^2 \Rightarrow min \ \ \frac{1}{2}||w||_2^2 + C\sum_{i=1}^{m}\xi_i$$

这个公式的原理还是不够明了，让我们还原一下，看看这个到底是什么个意思。由：

$$min \ \ \frac{1}{2}||w||_2^2 \Rightarrow max \ \ \frac{2}{||w||_2^2}$$

$$min \ \ \frac{C}{m}\sum_{i=1}^{m}\xi_i \Rightarrow max \ \ \frac{C}{m}\sum_{i=1}^{m}(1-\xi_i)$$

显然对于支持向量和支持向量以外的点，$max \ \frac{2}{||w||_2^2}$，相当于最大化margin。$max \ \frac{C}{m}\sum_{i=1}^{m}(1-\xi_i)$ 对于margin内的噪声点，最大化噪声点和决策边界之间的距离$1-\xi_i$； 对于误分类的点，其到决策边界之间的距离为$-(1-\xi_i)$，那么$max \ \ C\sum_{i=1}^{m}(1-\xi_i) \Rightarrow min \ \ |C\sum_{i=1}^{m}(1-\xi_i)|$，也就是最小化误分类点到决策边界之间的距离，翻译成汉语就是让其尽量不要错得那么离谱。

综合起来看一下

综合起来，我们就得到了最大软间隔SVM的优化目标：

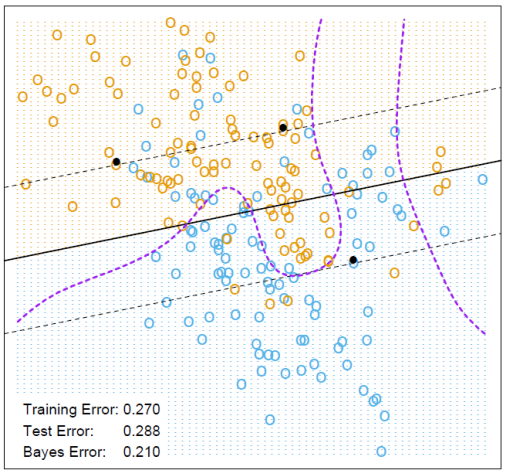
$$min \ \ \frac{1}{2}||w||_2^2 + \frac{C}{m}\sum_{i=1}^{m}\xi_i$$

$$s.t. \ \ y_i(w^T x_i + b) \geq 1 - \xi_i \ \ (i = 1, 2, \dots m)$$

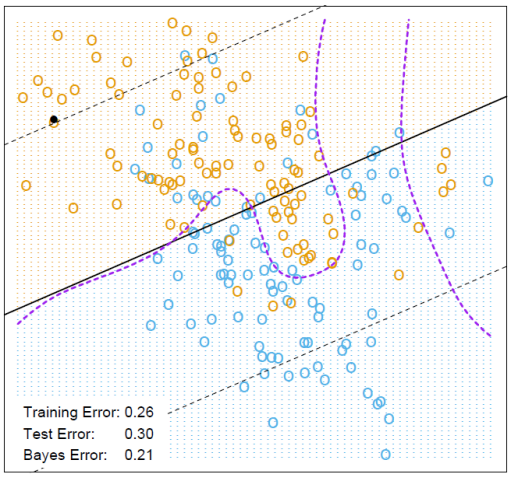
$$\xi_i \geq 0 \ \ (i = 1, 2, \dots m)$$

最小化$\frac{1}{2}||w||_2^2$意味着让支持向量距离超平面的距离尽可能大，最小化$\sum_{i=1}^{m}\xi_i$意味着保证 margin 内的点尽可能远离超平面，对误分类的点不要偏离的太远。C是协调两者关系的系数，需要调参来选择。

下面是取不同的C值的分类器对比情况，这两种分类器的性能相差不大，都比较差。C越小，margin越大，C越大，margin 越小。C越大，$\sum_{i=1}^{m}\xi_i$作用越大，模型会更多得关注噪声点和误分类点，即决策边界周围的点，C越小，模型会更多关注距离决策边界更远的点，当$C = 0$时，软间隔变成了硬间隔。 C的取值可以采用交叉验证的方式求得。



$C = 10000$



$C = 0.01$

对于硬间隔和软间隔的理解，下面我们举一个更形象的例子。比如我们要用 svm 来划分金庸武侠里的人物性别。

对于《射雕英雄传》，《神雕侠侣》，《倚天屠龙记》，《天龙八部》等小说中的人物，性格鲜明，性别明显，不存在噪声点和误分类点（例如tai jian，ren yao等），显然硬间隔SVM就能较好区分这一类小说中人物的性别。

然而，对于《笑傲江湖》，由于葵花宝典（据说辟邪剑谱和日月神教的葵花宝典都是原本葵花宝典的残本）的这部神奇的变性秘籍，可以将男人变成女人（典型的如"东方姑娘"）出现了我们所说的噪声点或者误分类的点，这样我们就必须采用软间隔SVM。具体可以把里面的角色分为以下几类，对于令狐冲，任盈盈，岳灵珊等人，很显然，他们性格鲜明，性别明显，我们可以把他们归为 $\xi = 0$ 的人群里面（支持向量还没有想到是谁，恒山派和少林派？希望大家帮忙补充，囧）。对于岳不群，林平之这种噪声点，我们可以把它们划分到margin中区，即 $0 < \xi < 1$，对于东方不败（徐克版和于正版的东方不败都直接启用林青霞和陈乔恩扮演，而且还和令狐冲谈起了恋爱），很显然"她"处于误分类的点，即 $\xi > 1$。而我们的代价函数是用来做什么的呢？是让令狐冲，任盈盈，岳灵珊等人的性别更加鲜明，男的更man，女的更woman；让岳不群、林平之尽量靠向男人的这一方（是不是像岳不群伪君子行为）；让"东方姑娘"不要太woman了。

Q3：影响软间隔SVM决策超平面的样本是否和硬间隔SVM一样，只有支持向量呢？

是的，影响软间隔SVM的决策超平面的只有支持向量。之所以不包含噪声点和误分类点，是为了防止过拟合。

3. 软间隔SVM优化算法

和线性可分SVM的优化方式类似，优化过程分为以下五步：

- 转化为拉格朗日函数
- 转化为对偶问题
- 简化对偶问题
- SMO 算法求解 α
- 根据 α 求解出 w 和 b

转化为拉格朗日函数

根据凸优化理论，代价函数满足 KKT 条件，我们可以通过拉格朗日函数将我们的优化目标转化为无约束的优化函数：

$$L(w, b, \xi, \alpha, \mu) = \frac{1}{2}||w||_2^2 + C\sum_{i=1}^{m}\xi_i - \sum_{i=1}^{m}\alpha_i[y_i(w^T x_i + b) - 1 + \xi_i] - \sum_{i=1}^{m}\mu_i\xi_i$$

其中 $\mu_i \geq 0, \alpha_i \geq 0$ ，均为拉格朗日系数。

我们的优化目标变成：

$$\underbrace{min}_{w,b,\xi}\ \underbrace{max}_{\alpha_i\geq 0,\mu_i\geq 0,}\ L(w,b,\alpha,\xi,\mu)$$

转化为对偶问题

这个拉格朗日函数满足KKT条件，我们可以通过拉格朗日对偶将该问题转化为等价的对偶问题来求解。即：

$$\underbrace{max}_{\alpha_i\geq 0,\mu_i\geq 0,}\ \underbrace{min}_{w,b,\xi}\ L(w,b,\alpha,\xi,\mu)$$

简化对偶问题

首先我们来求优化函数对于 w,b,ξ 的极小值，这个可以通过求偏导数求得：

$$\frac{\partial L}{\partial w}=0\ \Rightarrow w=\sum_{i=1}^{m}\alpha_i y_i x_i$$

$$\frac{\partial L}{\partial b}=0\ \Rightarrow \sum_{i=1}^{m}\alpha_i y_i=0$$

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

好了，我们可以利用上面的三个式子去消除 w，b 和 C 了。

将 $C=\alpha_i+\mu_i$ (记住这个式子，以后还会用得到) 带入式子，并进行化简：

$$\begin{aligned}L(w,b,\xi,\alpha,\mu)&=\frac{1}{2}||w||_2^2+C\sum_{i=1}^{m}\xi_i-\sum_{i=1}^{m}\alpha_i[y_i(w^Tx_i+b)-1+\xi_i]-\sum_{i=1}^{m}\mu_i\xi_i\\&=\frac{1}{2}||w||_2^2-\sum_{i=1}^{m}\alpha_i[y_i(w^Tx_i+b)-1+\xi_i]+\sum_{i=1}^{m}\mu_i\xi_i\\&=\frac{1}{2}||w||_2^2-\sum_{i=1}^{m}\alpha_i[y_i(w^Tx_i+b)-1]\end{aligned}$$

这个式子是否似曾相识，大家想想有没有在哪见过呢？像不像我们硬间隔SVM里讲到的 $\psi(\alpha)$？

下面按照惯例，我们进行化简（那一连串让人懵逼又超级简单的数学公式大家还记得吗？），得到：

$$\underbrace{max}_{\alpha}\sum_{i=1}^{m}\alpha_i-\frac{1}{2}\sum_{i=1,j=1}^{m}\alpha_i\alpha_j y_i y_j x_i^Tx_j$$

$$s.t.\ \sum_{i=1}^{m}\alpha_i y_i=0\quad C-\alpha_i-\mu_i=0\quad \alpha_i\geq 0\quad \mu_i\geq 0\ (i=1,2,\ldots,m)$$

对于 $C-\alpha_i-\mu_i=0,\ \alpha_i\geq 0,\ \mu_i\geq 0$ 这 3 个式子进行化简。得到最终结果：

$$\underbrace{min}_{\alpha}\frac{1}{2}\sum_{i=1,j=1}^{m}\alpha_i\alpha_j y_i y_j x_i^Tx_j-\sum_{i=1}^{m}\alpha_i$$

$$s.t.\ \sum_{i=1}^{m}\alpha_i y_i=0$$

$$0\leq\alpha_i\leq C$$

Q4：软间隔最大SVM的代价函数，和硬间隔最大SVM相比，发生了什么变化呢？为什么？

样本下限从 $0 \leq \alpha_i$ 变为 $0 \leq \alpha_i \leq C$， 对 α 限制得更厉害了。支持向量最大化的的方式不变。

SMO算法求解 α

只要我们可以求出上式极小化时对应的 α 向量就可以求出 w 和 b 了 (需要用到SMO算法)。在这里，我们假设通过SMO算法，我们得到了对应的 α 的值 α^*。

根据 α 求解出 w 和 b

那么我们根据 $w = \sum_{i=1}^{m} \alpha_i y_i x_i$， 可以求出对应的 w 的值：

$$w^* = \sum_{i=1}^{m} \alpha_i^* y_i x_i$$

再求出 b 我们就大功告成啦。注意到，对于支持向量 (Q3)，都有：

$y_s(w^{*T} x_s + b) = 1$

将 $w^* = \sum_{i=1}^{m} \alpha_i^* y_i x_i$ 带入上式可以得到：

$y_s(\sum_{i=1}^{m} \alpha_i^* y_i x_i^T x_s + b) = 1$

求得：

$b_s^* = y_s - \sum_{i=1}^{m} \alpha_i y_i x_i^T x_s$

假设我们有 S 个支持向量，则对应我们求出 S 个 b_s^*，然后将其平均值作为最后的结果。

Q5：上面的公式怎么得到的呢？ $\frac{1}{y_s} = y_s$ 是什么鬼？大家解释一下

因为在SVM中 y 只能取 1 或 -1，显然 $\frac{1}{y_s} = y_s$。

那么问题来了，怎样求解支持向量、噪声点以及误分类点呢？

在硬间隔最大化时，根据 KKT 条件中的对偶互补条件 $\alpha_i^*(y_i(w^T x_i + b) - 1) = 0$。

- 如果 $\alpha_i^* > 0$ 则有 $y_i(w^T x_i + b) = 1$ 即点在支持向量上，
- 如果 $\alpha_i^* = 0$ 则有 $y_i(w^T x_i + b) \geq 1$，即样本在支持向量上或者已经被正确分类。

软间隔最大化时 KKT 条件中的对偶互补条件包含两个：

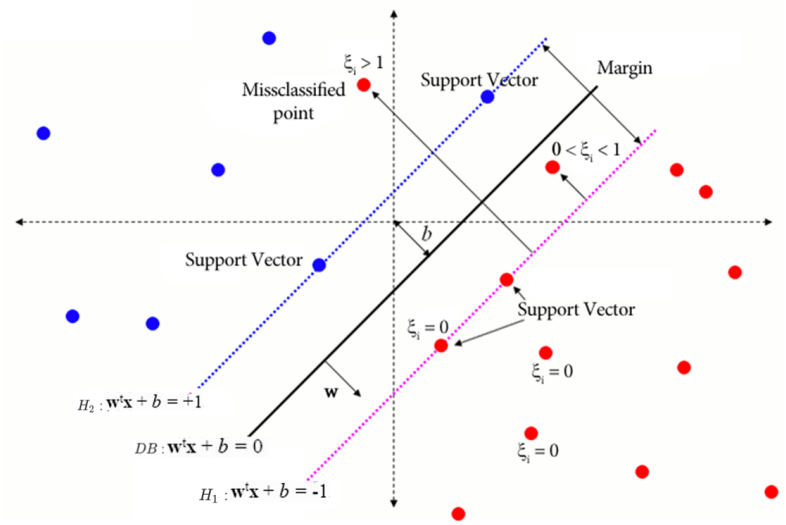
$\alpha_i^*(y_i(w^T x_i + b) - 1 + \xi_i^*) = 0$

$\mu_i \xi_i = 0 \Rightarrow (C - \alpha_i)\xi_i = 0$

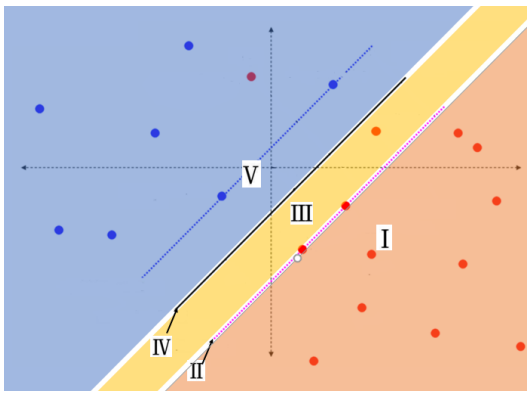
我们有：

- 如果 $\alpha = 0$，那么 $\xi_i = 0,\ \ y_i(w^T x_i + b) - 1 \geq 0$，即样本在支持向量上或者已经被正确分类。
- 如果 $0 < \alpha < C$ ，那么 $\xi_i = 0,\ \ y_i(w^T x_i + b) - 1 = 0$，即该点为支持向量。
- 如果 $\alpha = C$，说明这要么是噪声点，要么是误分类点，需要检查此时 ξ_i
    - 如果 $0 < \xi_i < 1$，该点被正确分类，但是却在超平面和自己类别的支持向量之间。

- 如果 $\xi_i = 1$，该点在分离超平面上，无法被正确分类。
- 如果 $\xi_i > 1$，该点就是误分类的点。



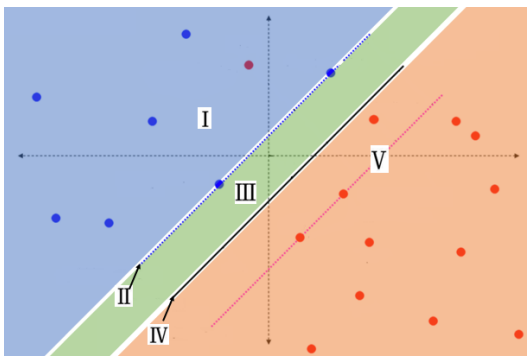
上面说了一大堆是什么意思呢，我们以红色的样本为例，将空间划分为以下五部分：



其中

- $\alpha = 0$ 对应 Ⅰ + Ⅱ
- $0 < \alpha < C$ 对应 Ⅱ
- $\alpha = C$ 对应 Ⅲ + Ⅳ + Ⅴ
- $\alpha = C$ 且 $0 < \xi_i < 1$ 对应 Ⅲ
- $\alpha = C$ 且 $\xi_i = 1$ 对应 Ⅳ
- $\alpha = C$ 且 $\xi_i > 1$ 对应 Ⅴ

Q6：大家可以用用蓝色的样本点来解释吗？



5. 最大软间隔SVM的算法总结

输入是 m 个线性可分的样本 $(x_1,y_1),(x_2,y_2),\ldots,(x_m,y_m)$,

其中 x 为 n 维特征向量。y 为二元分类结果 1 或 -1。

输出是分离超平面的参数 w^* 和 b^* 和分类决策函数。

算法过程如下：

1) 构造代价函数：

$$\underbrace{min}_{\alpha}\frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}\alpha_i\alpha_j y_i y_j(x_i^T x_j)-\sum_{i=1}^{m}\alpha_i$$

$$s.t.\ \sum_{i=1}^{m}\alpha_i y_i=0$$

$$0\le\alpha_i\le C\ i=1,2,\ldots m$$

2) 用 SMO 算法求出 α 向量的值 $\alpha^*=(\alpha_1^*,\alpha_2^*,\cdots,N)$ 。

3) 计算 $w^*=\sum_{i=1}^{m}\alpha_i^* y_i x_i$。

4) 找出满足$0<\alpha^*<C$ 的所有的α^*分量 ，求得 b_s^*

$$b^*=y_j-\sum_{i=1}^{N}\alpha_i^* y_i(x_i^T x_j)$$

取平均得到 b

5) 最终的分类决策函数为：$f(x)=sign(w^{*T}x+b^*)=sign(\sum_{i=1}^{m}\alpha_i^* y_i(x_i^T x)+b^*)$

6. SVM损失函数详解

总结一下，关于线性支持向量机我们学了三种代价函数：

- 合页损失函数 (hinge loss function)：

$$min\ \ \frac{1}{m}\sum_{i=1}^{m}max\{0,1-y_i(w^T x_i+b)\}+\frac{\lambda}{2}||w||_2^2$$

- 硬间隔损失函数：

$$min\ \ \frac{1}{2}||w||_2^2\quad s.t\ \ y_i(w^T x_i+b)\ge 1(i=1,2,\ldots m)$$

- 软间隔损失函数：

$$min\ \ \frac{1}{2}||w||_2^2+\frac{C}{m}\sum_{i=1}^{m}\xi_i$$

$$s.t.\ \ y_i(w^T x_i+b)\ge 1-\xi_i\ \ (i=1,2,\ldots m)$$

$$\xi_i\ge 0\ \ (i=1,2,\ldots m)$$

其实归根到底都是合页损失函数。

推导如下：

- 对于硬间隔损失函数：

  由以下条件：

$$y_i(w^Tx_i+b) \geq 1 \Rightarrow 1-y_i(w^Tx_i+b) \leq 0$$

可得：

$$max\{0, 1-y_i(w^Tx_i+b)\} = 0$$
$$\frac{1}{m\lambda}\sum_{i=1}^{m} max\{0, 1-y_i(w^Tx_i+b)\} = 0$$

则：

$$\begin{aligned} min\ \ \frac{1}{2}||w||_2^2 &= min\ \ \frac{1}{2}||w||_2^2 + \frac{1}{m\lambda}\sum_{i=1}^{m} max\{0, 1-y_i(w^Tx_i+b)\} \\ &= min\ \ \frac{1}{\lambda}(\frac{1}{m}\sum_{i=1}^{m} max\{0, 1-y_i(w^Tx_i+b)\} + \frac{\lambda}{2}||w||_2^2) \end{aligned}$$

得证。

- 对于软间隔损失函数：

  由以下两个条件：

$$y_i(w^Tx_i+b) \geq 1-\xi_i \Rightarrow \xi_i \geq 1-y_i(w^Tx_i+b)$$
$$\xi_i \geq 0$$

  合并为：

$$\xi_i \geq max\{0, 1-y_i(w^Tx_i+b)\}$$

  得到 ξ_i 的下限 $max\{0, 1-y_i(w^Tx_i+b)\}$，翻译成公式就是：

$$min\ \ \xi_i = max\{0, 1-y_i(w^Tx_i+b)\}$$

$$\begin{aligned} min\ \ \frac{1}{2}||w||_2^2 + C\sum_{i=1}^{m}\xi_i &= min\ \ \frac{1}{2}||w||_2^2 + \frac{C}{m}\sum_{i=1}^{m} max\{0, 1-y_i(w^Tx_i+b)\} \\ &= C(min\ \ \frac{1}{2C}||w||_2^2 + \frac{1}{m}\sum_{i=1}^{m} max\{0, 1-y_i(w^Tx_i+b)\}) \end{aligned}$$

  令 $C=\frac{1}{\lambda}$ 即可求证。

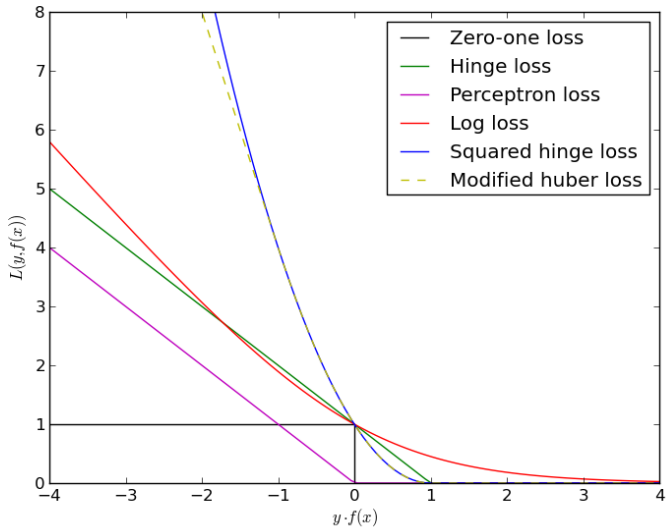
如下图中的绿线：如果点被正确分类，且在支持向量以外，损失是 0，否则损失是 $1-y(w\bullet x+b)$。如果点被正确分类，且在margin之内，损失为小于1的小数；如果点被分类错误，损失函数大于1，且随样本到超平面距离的增大，损失函数增大。

从下图中我们还可以看出其他各种模型损失和函数间隔的关系：

对于 0-1损失函数，如下图黑线，如果正确分类，损失是0，误分类损失 1， 且 0-1 损失函数是不可导的。

对于感知机模型，感知机的损失函数是 $[-y_i(w \bullet x + b)]_+$，如下图紫线。当样本被正确分类时，损失是 0，误分类时，损失是 $-y_i(w \bullet x + b)$。

对于逻辑回归之类和最大熵模型对应的对数损失，损失函数是 $log[1 + exp(-y(w \bullet x + b))]$，如下图红线所示。



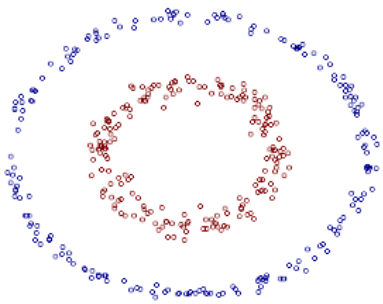
非线性支持向量机与核函数

1. 核函数的引入

以上介绍的都是SVM作为线性分类器的作用，那对于非线性问题，SVM该怎样做呢？

对于非线性问题，我们采取的做法是将进行一个非线性变换映射到特征空间中，将原空间非线性问题转变为特征空间的线性问题，然后再用线性分类器SVM求解。什么意思呢？我们举例说明：

图中的两类数据，分别分布为两个圆圈的形状，因为这样的数据本身就是线性不可分的，线性分类器是没法处理。那我们该如何处理呢？

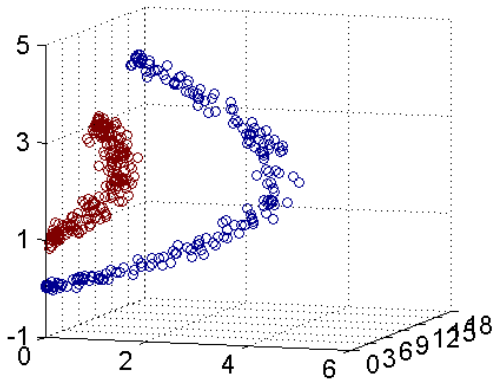


对于上图的数据，我们可以表示为：

$a_1 X_1^2 + a_2 (X_2 - c)^2 + a_3 = 0$

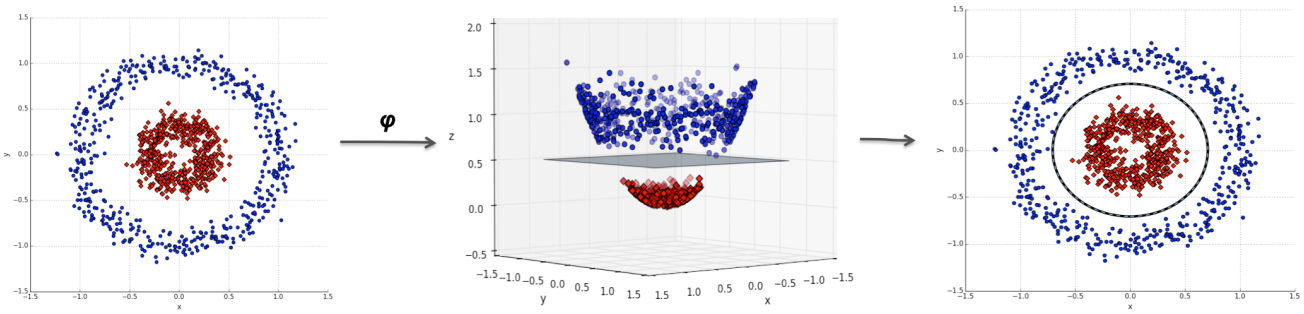
其中 X_1 和 X_2 是两个坐标系。

我们令 $Z_1 = X_1^2, Z_2 = X_2^2, Z_3 = X_2$（其中$Z_1, Z_2, Z_3$ 为三维空间的三个坐标）将其映射到三维空间中进行求解，如下图：

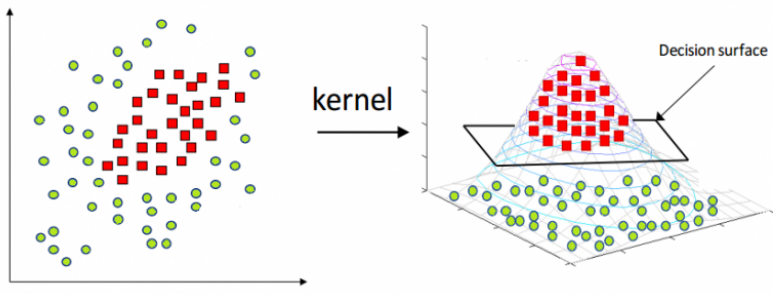


可以找到一个平面，将红色的样本和蓝色样本区分开。

下面展示了另一种特征映射和线性分类相结合的方式：



世界上怎么会有如此奇妙的求解手段，如此美妙的图片！！！我们再来欣赏一组：



也就是说对于在低维线性不可分的数据，在映射到了高维以后，就变成线性可分的了。这个思想我们同样可以运用到 SVM 的线性不可分数据上。也就是说，对于 SVM 线性不可分的低维特征数据，我们可以将其映射到高维，就可以运用线性可分 SVM 的进行求解了。

Q7：特征变换还有没有其他应用？

特征变换可以说是遍布整个机器学习界。

机器学习处理分类和回归问题本身就是特征变换。对于分类问题，输入空间维n维特征空间，输出为只有0 (-1) 和1的一维向量。对于回归问题，输入空间维n维特征空间，输出为连续的一维向量。这是高维空间转换为一维空间的情况。

对于熟悉深度学习的学生而言，$w^T x + b$ 为特征空间的线性变化，激活函数为特征空间的非线性变换（空间扭曲）。

我们这种思想运用到我们的SVM的算法上。回顾线性可分SVM的优化目标函数：

$$\underbrace{min}_{\alpha}\frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}\alpha_i\alpha_j y_i y_j(x_i \bullet x_j)-\sum_{i=1}^{m}\alpha_i$$

$$s.t.\ \sum_{i=1}^{m}\alpha_i y_i=0$$

$$0\le\alpha_i\le C\ i=1,2,\dots m$$

再次友情提示，对数学符号不太敏感的小朋友注意啦，对于任意两个向量 x 和 y， $x^Ty=x\bullet y$ 始终成立！！！

我们定义一个低维特征空间到高维特征空间的映射 ϕ，将所有输入空间映射到一个更高维度的特征空间，让数据线性可分，我们就可以利用 SVM 的优化目标函数求出分类决策边界了。也就是说现在的 SVM 的优化目标函数变成：

$$\underbrace{min}_{\alpha}\frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}\alpha_i\alpha_j y_i y_j[\phi(x_i) \bullet \phi(x_j)]-\sum_{i=1}^{m}\alpha_i$$

$$s.t.\ \sum_{i=1}^{m}\alpha_i y_i=0$$

$$0\le\alpha_i\le C\ i=1,2,\dots m$$

可以看到，和线性可分SVM的优化目标函数的区别仅仅是将内积 $x_i\bullet x_j$ 替换为 $\phi(x_i)\bullet\phi(x_j)$。

得到的超平面即为：

$$f(x)=sign(\sum_{i=1}^{m}\alpha_i^* y_i(\phi(x_i)\bullet\phi(x))+b^*)$$

我们发现在SVM算法的求解过程中 $\phi(x)$ 并不是单独存在的， 而是始终以 $\phi(x_i)\bullet\phi(x_j)$ 的形式出现，我们不如把他们写在一起，定义一个新的函数：

$$K(x,z)=\phi(x)\bullet\phi(z)$$

这玩意就是著名的核函数 $K(x,z)$（在这里我也解释了为什么核函数长成这个鬼样子）。

下面我们来看看官方是怎么定义核函数的的：

假设 ϕ 是一个从低维的输入空间 χ（欧式空间的子集或者离散集合）到高维的希尔伯特空间的 H 映射。对于所有的 $x,z\in\chi$，都有 $K(x,z)$ 满足：

$$K(x,z)=\phi(x_i)\bullet\phi(x_j)$$

那么我们就称 $K(x,z)$ 为核函数， $\phi(x)$ 为映射函数 。

左看看，右看看，这核函数也没什么神奇之处啊。核函数的神奇之处在与：用低维的运算来解决高维的运算问题。什么意思呢？

至今为止我们所使用的思想是拿到非线性数据，就找一个映射 $\phi(\cdot)$ ，然后一股脑把原来的数据映射到高维空间中，再在高维空间内做线性 SVM 。下面借鉴一下李航大神的案例：

例 7.3 假设输入空间是 $\mathbf{R}^2$，核函数是 $K(x,z)=(x\cdot z)^2$，试找出其相关的特征空间 $\mathcal{H}$ 和映射 $\phi(x):\mathbf{R}^2\to\mathcal{H}$.

解 取特征空间 $\mathcal{H}=\mathbf{R}^3$，记 $x=(x^{(1)},x^{(2)})^{\mathrm{T}}$，$z=(z^{(1)},z^{(2)})^{\mathrm{T}}$，由于

$$(x\cdot z)^2=(x^{(1)}z^{(1)}+x^{(2)}z^{(2)})^2=(x^{(1)}z^{(1)})^2+2x^{(1)}z^{(1)}x^{(2)}z^{(2)}+(x^{(2)}z^{(2)})^2$$

所以可以取映射

$$\phi(x)=((x^{(1)})^2,\sqrt{2}x^{(1)}x^{(2)},(x^{(2)})^2)^{\mathrm{T}}$$

容易验证 $\phi(x)\cdot\phi(z)=(x\cdot z)^2=K(x,z)$.

仍取 $\mathcal{H}=\mathbf{R}^3$ 以及

$$\phi(x)=\frac{1}{\sqrt{2}}((x^{(1)})^2-(x^{(2)})^2,2x^{(1)}x^{(2)},(x^{(1)})^2+(x^{(2)})^2)^{\mathrm{T}}$$

同样有 $\phi(x)\cdot\phi(z)=(x\cdot z)^2=K(x,z)$.

还可以取 $\mathcal{H}=\mathbf{R}^4$ 和

$$\phi(x)=((x^{(1)})^2,x^{(1)}x^{(2)},x^{(1)}x^{(2)},(x^{(2)})^2)^{\mathrm{T}}$$ ■

上面是什么意思呢？估计还是很多同学没有看懂。我们有原始的二维空间，要求解 $\phi(x)\bullet\phi(z)$，有很多途径：

- 法1：直接在原始空间 R^2 中求解**核函数**得到：
  $(x^{(1)}z^{(1)})^2+2x^{(1)}z^{(1)}x^{(2)}z^{(2)}+(x^{(2)}z^{(2)})^2$
- 法2：我们也可以先映射到三维空间 R^3 得到映射函数
  $\phi(x)=\frac{1}{\sqrt{2}}((x^{(1)})^2-(x^{(2)})^2,2x^{(1)}x^{(2)},(x^{(1)})^2+(x^{(2)})^2)^T$
  然后再求解$\phi(x)\bullet\phi(z)$
- 法3：我们也可以先映射到四维空间 R^4 得到映射函数
  $\phi(x)=((x^{(1)})^2,x^{(1)}x^{(2)},x^{(1)}x^{(2)},(x^{(2)})^2)^T$
  然后再求解 $\phi(x)\bullet\phi(z)$

当然我们还可以映射到 $R^5,R^6\cdots$

显然第一种方法更为简单。原始空间三维空间，四维空间，五维空间呢？特征空间会呈爆炸性增长的，这给 $\phi(\cdot)$ 的计算带来了非常大的困难，而且如果遇到无穷维的情况，就根本无从计算了。

也就是说，核函数可以让我们好好享受在高维特征空间线性可分的同时，避免了高维特征空间恐怖的内积计算量。

至此，我们总结下线性不可分时核函数的引入过程：

我们遇到线性不可分的样例时，常用做法是把样例特征映射到高维空间中去，但是有时候这个维度大小是会高到令人恐怖的，通常我们会采用核函数来处理这类问题。核函数好在它在低维上进行计算，而将实质上的分类效果（向量内积）表现在了高维上，避免了直接在高维空间中的复杂计算。

2. 核函数的介绍

从上面的分析发现，因此我们只需要定义核函数 $K(x,z)$，而不用显示的定义映射函数 ϕ， 即可求出 $\phi(x_i)\bullet\phi(x_j)$， 这样就省去了寻找映射函数的麻烦（映射函数由无数个）。但是却带来了另一个问题：我们怎样定义核函数 $K(x,z)$ 呢？

其实已经有人帮我们找到了很多的核函数，而且常用的核函数也仅仅只有那么几个（专业的人做专业的事）。下面我们简要介绍 sklearn 中默认可选的几个核函数。

2.1 线性核函数

线性核函数（Linear Kernel）其实就是线性可分SVM，表达式为：

$$K(x,z)=x\bullet z$$

也就是说，线性SVM是非线性SVM的一个特殊的情况，即线性SVM是使用线性核函数的SVM。

2.2 多项式核函数

多项式核函数（Polynomial Kernel）是线性不可分SVM常用的核函数之一，表达式为：

$$K(x,z)=（\gamma x\bullet z+r)^p$$

相当于将原始空间映射到 p 维空间。其中，γ,r,p 都需要调参。

2.3 高斯核函数

高斯核函数（Gaussian Kernel）也称为径向基核函数（Radial Basis Function，RBF），它是非线性分类SVM最主流的核函数。表达式为：

$$K(x,z)=exp(-\gamma||x-z||^2)$$

相当于将原始空间映射到无穷维空间。其中，非负参数γ需要调参。

2.4 Sigmoid核函数

Sigmoid核函数（Sigmoid Kernel）也是线性不可分SVM常用的核函数之一，表达式为：

$$K(x,z)=tanh（\gamma x\bullet z+r)$$

其中，γ,r 都需要自己调参定义。

3. 分类SVM的算法过程

现在我们对分类SVM的算法过程做一个总结。不再区别是否线性可分。

输入是 m 个线性可分的样本 $(x_1,y_1),(x_2,y_2),...,(x_m,y_m)$,

其中 x 为 n 维特征向量。y 为二元分类结果 1或 -1。

输出是分离超平面的参数 w^* 和 b^* 和分类决策函数。

算法过程如下：

1）选择适当的核函数 $K(x,z)$ 和一个惩罚系数 $C>0$, 构造约束优化问题构造代价函数：

$$\underbrace{min}_{\alpha}\frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}\alpha_i\alpha_jy_iy_jK(x_i,x_j)-\sum_{i=1}^{m}\alpha_i$$

$$s.t.\ \sum_{i=1}^{m}\alpha_iy_i=0$$

$$0\le\alpha_i\le C\ \ i=1,2,\cdots,N$$

2) 用 SMO 算法求出 α 向量的值 $\alpha^*=(\alpha_1^*,\alpha_2^*,\cdots,N)$ 。

3) 计算 $w^*=\sum_{i=1}^{m}\alpha_i^*y_i\phi(x_i)$。

4) 找出满足$0 < \alpha^* < C$ 的所有的α^*分量 ，求得 b_s^*

$$b^* = y_j - \sum_{i=1}^{N} \alpha_i^* y_i K(x_i, x_j)$$

取平均得到 b

5) 最终的分类决策函数为：$f(x) = sign(w^{*T}\phi(x) + b^*) = sign(\sum_{i=1}^{m} \alpha_i^* y_i K(x_i, x) + b^*)$

参考文献：

支持向量机原理(一) 线性支持向量机 http://www.cnblogs.com/pinard/p/6097604.html

T. Hastie/ R. Tibshirani / J. H. Friedman 《The Elements of Statistical Learning》

pluskid 《支持向量机: Kernel》 http://blog.pluskid.org/?p=685

李航 《统计学习方法》