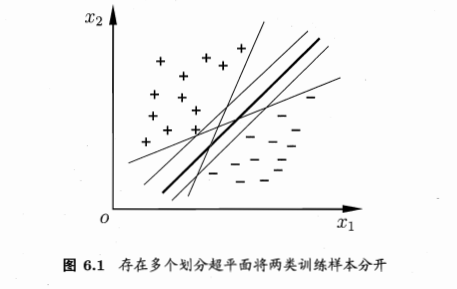
SVM

给定训练样本 , 如下图所以可以有多个超平面将其分开



该样本空间里，超平面可以用下面的表达式表示：

假设点到超平面S： 的距离为。（注：, w, x均为N维向量），假设在超平面S上的投影点为，所以我们有：

即

根据向量相乘的定义，有：

由于向量与超平面S的法向量 平行，所以：

其中 为向量的L2范数

所以由等式（4）和等式（5）可得：

等式（6）就是我们求出的空间里任意一点到超平面S的距离的表达式。

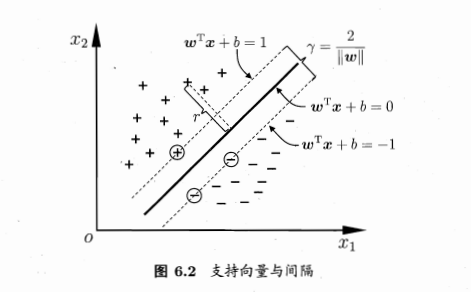
假设超平面S：(w, b)可以将所有样本正确分类，即对所有的样本，如果，有；如果，有。即：

从等式（6）可知，由于 和 都在变化，所以比较难优化 。所以不如假定训练样本的两个类别里离超平面S最近的点（该点被称为支持向量support vector）的 ，通过优化来优化支持向量到超平面的距离 。所以等式（7）可以表示为：

即

因为一旦选定一个超平面，w其实已经确定，所以 就确定，从而如果支持向量的 ，则其他点的 必然 。

总结来说，两类样本的各自支持向量之间间隔（margin）可以表示为等式（10），如下图：



由等式（9）和等式（10）我们的优化问题变为如下表达式：

即

我们可以通过拉格朗日乘子法来解等式（12）。（拉格朗日乘子法详见另一个word文档）我们通过拉格朗日乘子法得到下面的函数

其中m为样本的总数量，。

令L对w和b的偏导为0可得：

将等式（14）孤立出w代入等式（13）可得：

所以等式（15）（16）可以写为：

上述过程需满足KKT条件，KKT条件如下：

KKT条件里第二条即等式（9）。由KKT条件可知， 和 两者之间必有一个为0。如果，则等式（16）可知该样本 并不影响最终的L；如果，则由等式（9）可知该样本 必然是支持向量。所以支持向量机的本质是：训练完成后，大部分训练样本都不需要保留，最终模型仅与支持向量有关。

求解等式（16）可以使用一种有效的方法：SMO（Sequential Minimal Optimization）

SMO的思想是：选取一组和进行优化，保持其他不变（因为存在等式（15），所以不可能只有一个改变，其他不改变），即：

Repeat until convergence:

{

* 选取一组需要优化的和
* 固定和以外的其他，求解等式（16）中使得最大的和的值
* 更新和的值

}

由等式（15）可知：假设我们第一组选取的是和，则有：

因为我们固定了除和外的其他不变，等式（17）右边是一个常数，等式（17）又可以表示为：

从等式（16）的格式可以看出，如果保持除和的其他不变，为常数；用等式（18）里的表示，等式（16）的L可以表示为的二次函数，可写为：

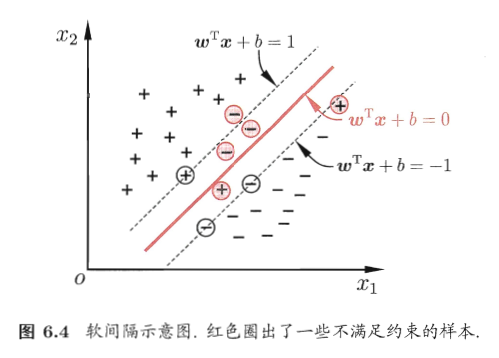
在等式（20）的条件下求等式（19）里L的极值，从而获得新的用于更新迭代。这样迭代收敛后获得的所有的值代回等式（14），即可求出SVM需要寻找的超平面S的第一个参数w。即：

因为从KKT条件下面的分析可知，对于非支持向量，必然有，所以等式（21）也可以简化为：

接下来要确定超平面S的第二个参数b。因为对于所有任意支持向量（），由等式（9）下面的分析可得：

等式（23）选用了其中一个支持向量（）来计算b，更鲁棒的做法是，对所有的支持向量都求一遍b，然后取这些b的平均值作为最终的b。

上面的推导都基于数据集线性可分的情况，如果数据集线性不可分，如下图：



则等式（12）可以写为：

上式表明允许一些样本到超平面的距离小于支持向量（），甚至允许一些样本被误判断（），但是会在损失函数上对其进行惩罚（加上），以便这些样本不会出现太多。

则等式（13）可以写为：

其中m为样本的总数量，，。

令L对w，b，的偏导为0可得：

将等式（14）（15）（26）代回等式（25）可得：

所以等式（15）（16）可以写为：

下面我们来看SMO算法中等式（19）如何求极值。首先我们可以看到，与等式（19）等效的等式（16）里， 这个部分可以算是下面这个矩阵里所有元素的和。

所以有关选取的和有关的项可以表示为：

其中：

所以等式（16）可以表示为：

由等式（17），（18）我们可知：

当时：

当时：

其中：

将等式（35）和等式（36）代入等式（34）即可得：

当时：

同理，当时：

总结来说等式（38）和（39）里面的红色部分就是等式（19）里面的A，蓝色部分就是等式（19）里的B。