8. The SMO algorithm

支持向量机的学习问题可以形式化为求解凸二次规划问题。这样的凸二次规划问题具有全局最优解，并且许多最优化算法可以用于对这一问题的求解，但是当训练样本容量很大时，这些算法往往变得非常低效，以致无法使用。所以，如何高效地实现支持向量机学习就成为一个重要的问题。目前人们已提出许多快速实现的算法。这里要介绍的就是 SMO 算法，SMO 是对 sequential minimal optimization（序列最小化优化） 的缩写，于1998年由 John Platt 在微软研究院提出。在这之前，我们先介绍坐标上升法（coordinate ascent algorithm）。

8.1 Coordinate ascent

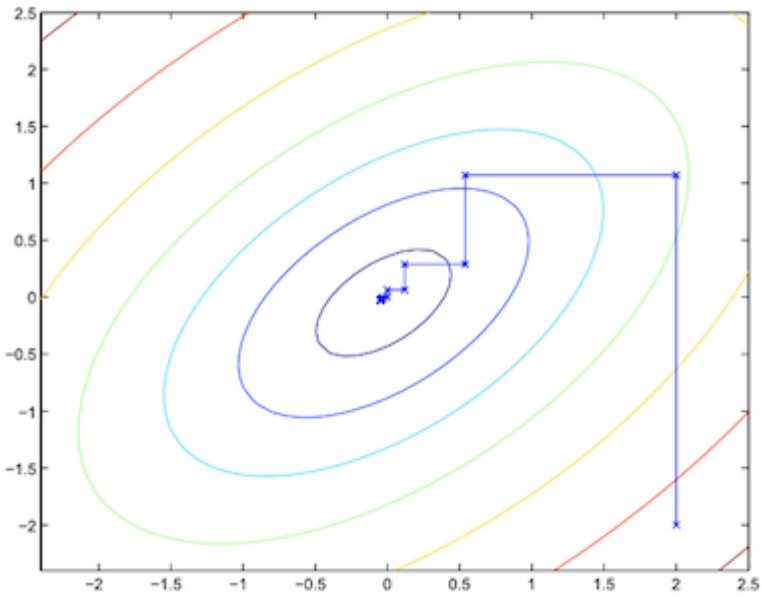
假设要解决如下无约束优化问题：

$$\max_{\alpha} \quad W(\alpha_1, \alpha_2, \cdots, \alpha_m)$$

这里的 W 是关于参数 α_i 的某种函数。坐标上升法是指：

$$\begin{aligned}
&Loop\ until\ convergence: \{ \\
&\quad For\ i = 1, \cdots, m, \{ \\
&\qquad \alpha_i := arg\ max_{\hat{\alpha}_i} W(\alpha_1, \cdots, \alpha_{i-1}, \hat{\alpha}_i, \alpha_{i+1}, \cdots, \alpha_m) \\
&\quad \} \\
&\}
\end{aligned}$$

在内层循环中，我们会固定除了 α_i 之外的所有变量，然后相对于 α_i 使函数取最大值。因为每次都只对一个变量进行优化，所以在坐标上升法的每一步中，移动的方向都是平行于某个坐标轴。如下图所示：



在内层循环中，我们对变量的优化顺序是按照变量排列的次序 $\alpha_1, \alpha_2, \cdots, \alpha_m, \alpha_1, \alpha_2, \cdots$。在其他更复杂的版本中会采取不同的方式，例如可以根据估计哪个变量可以使 $W(\alpha)$ 增加最多来进行选择。

但由于我们要求解的对偶问题存在着 $\sum_{i=1}^{m} \alpha_i y^{(i)} = 0$ 这个约束，使得我们不能直接使用坐标上升法。因此，如果我们要对 α_i 当中的一些值进行更新的话，就必须至少同时更新两个，这样才能保证满足约束条件。基于这个情况就衍生出 SMO 算法。

8.2 SMO

SMO 算法要求解如下凸二次规划的对偶问题：

$$\begin{aligned} \max_{\alpha} \quad & \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} \alpha_i \alpha_j y^{(i)} y^{(j)} K(x^{(i)}, x^{(j)}) \\ s.t. \quad & \sum_{i=1}^{m} \alpha_i y^{(i)} = 0 \\ & 0 \le \alpha_i \le C, \quad i = 1, \cdots, m \end{aligned}$$

这个问题中，变量是拉格朗日乘子，一个变量 α_i 对应于一个样本点 $(x^{(i)}, y^{(i)})$。变量的总数等于训练样本容量 m。

SMO 算法是一种启发式算法，其基本思路是：如果所有变量的解都满足此最优化问题的 KKT 条件，那么这个最优化问题的解就得到了。因为 KKT 条件是该最优化问题的充分必要条件。否则，选择两个变量，固定其他变量，针对这两个变量构建一个二次规划问题。这个二次规划问题关于这两个变量的解应该更接近原始二次规划问题的解，因为这会使得原始二次规划问题的目标函数值变得更小。更重要的是，这时子问题可以通过解析方法求解，这样就可以大大提高整个算法的计算速度。子问题有两个变量，一个是违反 KKT 条件最严重的那一个，另一个由约束条件自动确定。如此，SMO 算法将原问题不断分解为子问题并对子问题求解，从而达到求解原问题的目的。

注意，子问题的两个变量中只有一个是自由变量。假设 α_1, α_2 为两个变量，$\alpha_3, \alpha_4, \cdots, \alpha_m$ 固定，那么由等式约束 $\sum_{i=1}^{m} \alpha_i y^{(i)} = 0$ 可知：

$$\alpha_1 = -y^{(1)} \sum_{i=2}^{m} \alpha_i y^{(i)}$$

所以如果 α_2 确定，那么 α_1 也随之确定。子问题中同时更新两个变量。

整个 SMO 算法包括两个部分：求解两个变量二次规划的解析方法和选择变量的启发式方法。

8.2.1 两个变量二次规划的求解方法

不失一般性，假设选择的两个变量是 α_1, α_2，其他变量 $\alpha_i (i = 3, 4, \cdots, m)$ 是固定的。于是 SMO 的最优化问题的子问题可以写成（这里把 max 改成了 min，同时省略了不含 α_1, α_2 的常数项）：

$$\min_{\alpha 1,\alpha 2} \quad W(\alpha_1,\alpha_2)=\sum_{i=1}^{m}\alpha_i\alpha_1 y^{(i)}y^{(1)}K_{i1}+\sum_{i=1}^{m}\alpha_i\alpha_2 y^{(i)}y^{(2)}K_{i2}-(\alpha_1+\alpha_2)$$

$$=\frac{1}{2}K_{11}\alpha_1^2+\frac{1}{2}K_{22}\alpha_2^2+y^{(1)}y^{(2)}K_{12}\alpha_1\alpha_2-(\alpha_1+\alpha_2)+y^{(1)}\alpha_1\sum_{i=3}^{m}y^{(i)}\alpha_i K_{i1}+y^{(2)}\alpha_2\sum_{i=3}^{m}$$

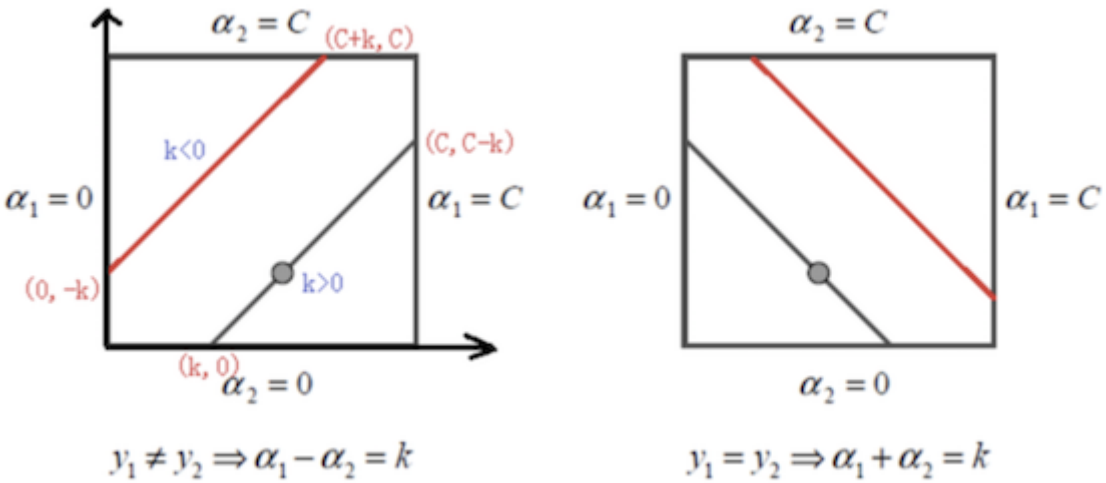
$$s.t. \quad \alpha_1 y^{(1)}+\alpha_2 y^{(2)}=-\sum_{i=3}^{m}y^{(i)}\alpha_i=\zeta$$

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

其中，$K_{ij}=K(x^{(i)},x^{(j)}), i,j=1,2,\cdots,m$, ζ 是常数。

首先，我们先分析约束条件。

由于只有两个变量 (α_1,α_2)，约束可以用二维空间中的图形表示。



不等式约束 $0\le\alpha_i\le C, i=1,2$ 使得 (α_1,α_2) 在盒子 $[0,C]\times[0,C]$ 内，等式约束 $\alpha_1 y^{(1)}+\alpha_2 y^{(2)}=\zeta$ 使 (α_1,α_2) 在平行于盒子 $[0,C]\times[0,C]$ 的对角线的直线上。因此要求的使目标函数在一条平行于对角线的线段上的最优值。这使得两个变量的最优化问题成为实质上的单变量的最优化问题，不妨考虑为变量 α_2 的最优化问题。

假设该问题的初始可行解为 $\alpha_1^{old},\alpha_2^{old}$，最优解为 $\alpha_1^{new},\alpha_2^{new}$，并假设在沿着约束方向未经剪辑时 α_2 的最优解为 $\alpha_2^{new,unc}$。

由于 α_2^{new} 需要满足不等式约束，所以最优值 α_2^{new} 的取值范围必须满足条件：

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

其中，L 和 H 是 α_2^{new} 所在对角线端点的界。

如上图，当 $y^{(1)}\neq y^{(2)}$ 时，线性限制条件可以写成:$\alpha_1-\alpha_2=k$，根据 k 的正负可以得到不同的上下界，因此可以统一表示为：

$$L = \max(0, \alpha_2^{old} - \alpha_1^{old})$$
$$H = \min(C, C + \alpha_2^{old} - \alpha_1^{old})$$

当 $y^{(1)} = y^{(2)}$ 时，限制条件可以写成:$\alpha_1 + \alpha_2 = k$，上下界表示为：

$$L = max(0, \alpha_1^{old} + \alpha_2^{old} - C)$$
$$H = min(C, \alpha_1^{old} + \alpha_2^{old})$$

根据得到的上下界，我们可以得到修剪后的 α_2^{new}：

$$\alpha_2^{new} = \begin{cases} H & \alpha_2^{new,unc} > H \\ \alpha_2^{new,unc} & L \le \alpha_2^{new,unc} \le H \\ L & \alpha_2^{new,unc} < L \end{cases}$$

其次，先不考虑不等式约束，即先求得沿着约束方向未经剪辑的 α_2 的最优解 $\alpha_2^{new,unc}$；接着，根据上面得到的公式求剪辑后的 α_2 的解 α_2^{new}。

记

$$g(x) = \sum_{i=1}^{m} \alpha_i y^{(i)} K(x^{(i)}, x) + b$$
$$E_i = g(x^{(i)}) - y^{(i)} = \left(\sum_{j=1}^{m} \alpha_j y^{(j)} K(x^{(j)}, x^{(i)}) + b \right) - y^{(i)}, \quad i = 1, 2$$

当 $i = 1, 2$ 时，E_i 为函数 $g(x)$ 对输入 $x^{(i)}$ 的预测值与真实输出 $y^{(i)}$ 之差。

令

$$v_i = \sum_{j=3}^{m} \alpha_j y^{(j)} K(x^{(i)}, x^{(j)}) = g(x^{(i)}) - \sum_{j=1}^{2} \alpha_j y^{(j)} K(x^{(i)}, x^{(j)}) - b, \quad i = 1, 2$$

目标函数可以写成：

$$W(\alpha_1, \alpha_2) = \frac{1}{2} K_{11} \alpha_1^2 + \frac{1}{2} K_{22} \alpha_2^2 + y^{(1)} y^{(2)} K_{12} \alpha_1 \alpha_2 - (\alpha_1 + \alpha_2) + y^{(1)} v_1 \alpha_1 + y^{(2)} v_2 \alpha_2$$

由 $\alpha_1 y^{(1)} = \zeta - \alpha_2 y^{(2)}$ 及 $(y^{(i)})^2 = 1$，可将 α_1 表示为

$$\alpha_1 = (\zeta - \alpha_2 y^{(2)}) y^{(1)}$$

将其代入目标函数，这样就得到只是 α_2 的函数的目标函数：

$$W(\alpha_2) = \frac{1}{2} K_{11} (\zeta - \alpha_2 y^{(2)})^2 + \frac{1}{2} K_{22} \alpha_2^2 + y^{(2)} K_{12} (\zeta - \alpha_2 y^{(2)}) \alpha_2 - (\zeta - \alpha_2 y^{(2)}) y^{(1)} - \alpha_2 + v_1 (\zeta - \alpha_2 y^{(2)}) +$$

对 α_2 求导：

$$\frac{\partial W}{\partial \alpha_2} = K_{11}\alpha_2 + K_{22}\alpha_2 - 2K_{12}\alpha_2 - K_{11}\zeta y^{(2)} + K_{12}\zeta y^{(2)} + y^{(1)}y^{(2)} - 1 - v_1 y^{(2)} + v_2 y^{(2)}$$

令其为零可得：

$$\begin{aligned}(K_{11} + K_{22} - 2K_{12})\alpha_2 &= K_{11}\zeta y^{(2)} - K_{12}\zeta y^{(2)} - y^{(1)}y^{(2)} + 1 + v_1 y^{(2)} - v_2 y^{(2)} \\ &= y^{(2)}(y^{(2)} - y^{(1)} + \zeta K_{11} - \zeta K_{12} + v_1 - v_2) \\ &= y^{(2)}\left[y^{(2)} - y^{(1)} + \zeta K_{11} - \zeta K_{12} + \left(g(x^{(1)}) - \sum_{j=1}^{2}\alpha_j y^{(j)} K_{1j} - b\right) - \left(g(x^{(2)}) - \sum_{j=\right.\right. \\ &= y^{(2)}\left(\zeta K_{11} - \zeta K_{12} + \alpha_1 y^{(1)} K_{21} + \alpha_2 y^{(2)} K_{22} - \alpha_1 y^{(1)} K_{11} - \alpha_2 y^{(2)} K_{12} + y^{(2)} - y^{(1)}\right.\end{aligned}$$

将 $\zeta = \alpha_1^{old} y^{(1)} + \alpha_2^{old} y^{(2)}$ 代入，得到：

$$\begin{aligned}(K_{11} + K_{22} - 2K_{12})\alpha_2^{new,unc} &= y^{(2)}\left((K_{11} + K_{22} - 2K_{12})\alpha_2^{old} y^{(2)} + y^{(2)} - y^{(1)} + g(x^{(1)}) - g(x^{(2)})\right) \\ &= (K_{11} + K_{22} - 2K_{12})\alpha_2^{old} + y^{(2)}(E_1 - E_2)\end{aligned}$$

令 $\eta = (K_{11} + K_{22} - 2K_{12}) = \|\phi(x^{(1)}) - \phi(x^{(2)})\|^2$，可得：

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

根据上面给出的 α_2^{new} 与 $\alpha_2^{new,unc}$ 的关系，我们就可以求得 α_2^{new}

最后，我们可以根据 $\alpha_1^{old} y^{(1)} + \alpha_2^{old} y^{(2)} = \alpha_1^{new} y^{(1)} + \alpha_2^{new} y^{(2)}$ 得到 α_1^{new}：

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

这样，就得到该最优化子问题的解 $(\alpha_1^{new}, \alpha_2^{new})$。

8.2.2 变量的选择方法

SMO 算法在每个子问题中选择两个变量优化，其中至少一个变量是违反 KKT 条件的。

1. 第1个变量的选择

SMO 称选择第1个变量的过程为外层循环。外层循环在训练样本中选取违反 KKT 条件最严重的样本点，并将其对应的变量作为第1个变量。具体地，检验训练样本点 $(x^{(i)}, y^{(i)})$ 是否满足 KKT 条件，即：

$$
\begin{aligned}
\alpha_i = 0 \quad &\Longleftrightarrow \quad y^{(i)} g(x^{(i)}) \geq 1 \\
\alpha_i = C \quad &\Longleftrightarrow \quad y^{(i)} g(x^{(i)}) \leq 1 \\
0 < \alpha_i < C \quad &\Longleftrightarrow \quad y^{(i)} g(x^{(i)}) = 1
\end{aligned}
$$

其中，$g(x^{(i)}) = \sum_{j=1}^{m} \alpha_j y^{(j)} K(x^{(i)}, x^{(j)}) + b$

该检验是在 ϵ 范围内进行的。在检验过程中，外层循环首先遍历所有满足条件 $0 < \alpha_i < C$ 的样本点，即在间隔边界上的支持向量点，检验它们是否满足 KKT 条件。如果这些样本点都满足 KKT 条件，那么遍历整个训练集，检验它们是否满足 KKT 条件。

2. 第2个变量的选择

SMO 称选择第2个变量的过程为内层循环。假设在外层循环中已经找到第1个变量 α_1，现在要在内层循环中找第2个变量 α_2。第2个变量选择的标准是希望能使 α_2 有足够大的变化。

由上面的推导我们知道，α_2^{new} 是依赖于 $\left|E_1 - E_2\right|$ 的，为了加快计算速度，一种简单的做法是选择 α_2，使其对应的 $\left|E_1 - E_2\right|$ 最大。

因为 α_1 已定，E_1 也确定了。如果 E_1 是正的，那么选择最小的 E_i 作为 E_2；如果 E_1 是负的，那么选择最大的 E_i 作为 E_2。为了节省计算时间，将所有 E_i 的值保存在一个列表中。

在特殊情况下，如果内层循环通过以上方法选择的 α_2 不能使目标函数有足够的下降，那么采用以下启发式规则继续选择 α_2。遍历在间隔边界上的支持向量点，依次将其对应的变量作为 α_2 试用，直到目标函数有足够的下降。若找不到适合的 α_2，那么遍历训练数据集；若仍找不到合适的 α_2，则放弃第1个 α_1，再通过外层循环寻求另外的 α_1。

3. 计算阈值 b 和差值 E_i

在每次完成两个变量的优化后，都要重新计算阈值 b。

当 $0 \leq \alpha_1^{new} \leq C$ 时，由 KKT 条件可知：

$$
\sum_{i=1}^{m} \alpha_i y^{(i)} K_{i1} + b = y^{(1)}
$$

于是，

$$
b_1^{new} = y^{(1)} - \sum_{i=3}^{m} \alpha_i y^{(i)} K_{i1} - \alpha_1^{new} y^{(1)} K_{11} - \alpha_2^{new} y^{(2)} K_{21}
$$

由

$$
E_1 = \left(\sum_{i=1}^{m} \alpha_i y^{(i)} K(x^{(i)}, x^{(1)}) + b \right) - y^{(1)}
$$

可得

$$E_1 = \sum_{i=3}^{m} \alpha_i y^{(i)} K_{i1} + \alpha_1^{old} y^{(1)} K_{11} + \alpha_2^{old} y^{(2)} K_{21} + b^{old} - y^{(1)}$$

$$y^{(1)} - \sum_{i=3}^{m} \alpha_i y^{(i)} K_{i1} = -E_1 + \alpha_1^{old} y^{(1)} K_{11} + \alpha_2^{old} y^{(2)} K_{21} + b^{old}$$

因此

$$\begin{aligned} b_1^{new} &= -E_1 + \alpha_1^{old} y^{(1)} K_{11} + \alpha_2^{old} y^{(2)} K_{21} + b^{old} - \alpha_1^{new} y^{(1)} K_{11} - \alpha_2^{new} y^{(2)} K_{21} \\ &= -E_1 - y^{(1)} K_{11} (\alpha_1^{new} - \alpha_1^{old}) - y^{(2)} K_{21} (\alpha_2^{new} - \alpha_2^{old}) + b^{old} \end{aligned}$$

同理，如果 $0 \le \alpha_2^{new} \le C$，那么

$$b_2^{new} = -E_2 - y^{(1)} K_{12} (\alpha_1^{new} - \alpha_1^{old}) - y^{(2)} K_{22} (\alpha_2^{new} - \alpha_2^{old}) + b^{old}$$

如果 $\alpha_1^{new}, \alpha_2^{new}$ 同时满足条件 $0 \le \alpha_i^{new} \le C, i = 1, 2$，那么 $b_1^{new} = b_2^{new}$

如果 $\alpha_1^{new}, \alpha_2^{new}$ 是 0 或者 C，那么 b_1^{new} 和 b_2^{new} 以及它们之间的数都是符合 KKT 条件的阈值，这时选择它们的中点作为 b^{new}

在每次完成两个变量的优化之后，还必须更新对应的 E_i 值，并将它们保存在列表中，E_i 值的更新要用到 b^{new} 值，以及所有支持向量对应的 α_j:

$$E_i^{new} = \sum_{S} y^{(j)} \alpha_j K(x^{(i)}, x^{(j)}) + b^{new} - y^{(i)}$$

其中，S 是所有支持向量 $x^{(j)}$ 的集合。

参考资料:

1. 李航，《统计学习方法》
2. 吴恩达，cs229 讲义
3. 机器学习算法实践-SVM中的SMO算法，https://zhuanlan.zhihu.com/p/29212107 (https://zhuanlan.zhihu.com/p/29212107)

In [ ]: