SVM 支持向量机

Dezeming Family

Dezeming

Art. No 0
ISBN 000–00–0000–00–0
Edition 0.0

Cover design by Dezeming Family

Published by Dezeming
Printed in China

目录

前言及简介

DezemingFamily 系列书和小册子因为是电子书，所以可以很方便地进行修改和重新发布。如果您获得了 *DezemingFamily* 的系列书，可以从我们的网站 *[https://dezeming.top/]* 找到最新版。对书的内容建议和出现的错误欢迎在网站留言。

0.1 本书前言

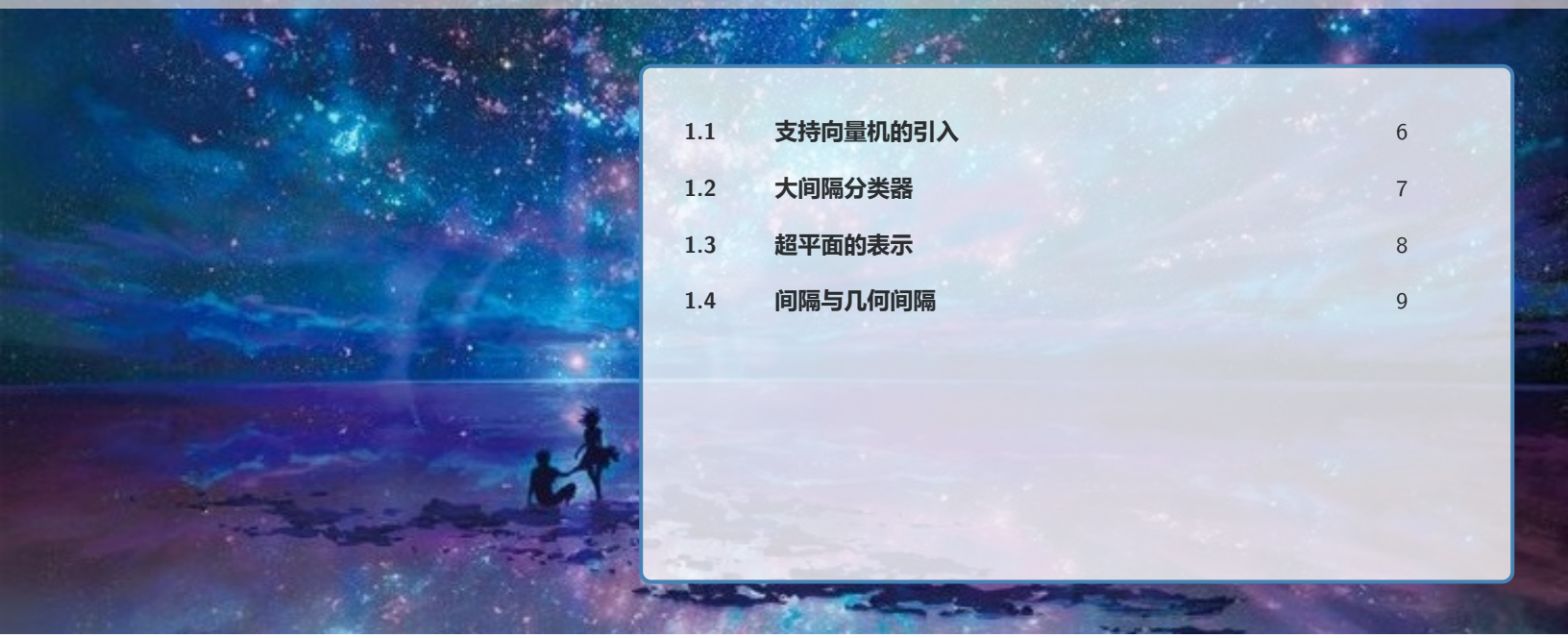
支持向量机自从被提出以后，在分类的等问题上大放异彩。大部分人使用支持向量机都是直接调用 python 的相关程序，而对支持向量机的内部工作原理和推导认识并不是很充分。网上的参考资料也都不是非常全面，不好理解。

我希望能对 SVM 支持向量机进行最详细和全面的原理介绍，并介绍一些实际使用上的术语和经验。相比于庞大的神经网络架构，支持向量机虽然也发展的越来越复杂，但终究好把握一些。

我们很难将 SVM 中所有的基础内容都研究透彻，尤其是涉及到凸优化和运筹学的一些专业知识更是如此。在这里，我的建议是，某些已经被数学家们证明过的定理，我们可以先直接使用（比如 KKT 条件）；在 SVM 上的推导和流程，我们要好好掌握，重点是先把整个流程走通。最后，细节部分再慢慢补充，就可以起到事半功倍的效果。本书只是一个基础和引子，更进阶的内容会在后续进行发布，包括一些对偶性的推导，以及 SMO 求解方法等。

本书的售价是 3 元（电子版），我们不直接收取任何费用，如果本书对大家学习有帮助，可以往我们的支付宝账户（17853140351）进行支持，您的赞助将是我们 Dezeming Family 继续创作各种计算机视觉、图形学、机器学习、以及数学原理小册子的动力！

1. 支持向量与大间隔分类器



本章讲解支持向量机的提出和基本原理，明确分类器的目标和相关约束项。

1.1 支持向量机的引入

SVM 由机器学习大牛 Vladimir Naumovich Vapnik（俄罗斯统计学家）和他的一些同事于 1995 年提出（主要是软间隔分类），由于原理清晰明确（当时的最优化理论已经发展到很高的水平了），功能强大，因此得到了很广泛的应用。

SVM 是建立在统计学习理论的 VC 维理论（Vapnik 的 VC 维理论为他带来了很多奖项和荣誉）和结构风险最小原理基础上的。

首先解释一下结构化风险，结构化风险 = 经验风险 + 置信风险，经验风险就是分类器在给定的样本上的分类误差，而置信风险是在未知样本上分类的误差。我们在训练模型时，会让模型的结构化风险变得非常小，但是至于这个模型能否预测未知样本，我们关注的是置信风险，显然，如果训练模型的样本数越多，那么置信风险就会变得越小；如果分类函数越复杂，则说明分类函数的普适性越差，则会增加置信风险。

这里衡量函数复杂性的方式就是 VC 维理论，我们这里只做一个简单的解释，具体详细的 VC 维理论可以参考 Dezeming Family 的《统计学习 VC 维理论》一书。比如我们使用一条直线来拟合样本，那么拟合度可能不会很好，因此，我们可以使用二次曲线、三次曲线甚至更高次的曲线，曲线越复杂，则经验风险就会越低，但是很容易过拟合，因而增大置信风险。因为置信风险是用来衡量未知样本的，所以我们无法真正地计算出精确值，只能进行估计。

SVM 的意义是为了让经验风险和置信风险的和最小。SVM 可以进行非线性分类，这是由于其松弛变量（惩罚变量，用于软间隔）和核函数的作用（其实就是一种映射关系），我们会在后面进行详细解释。

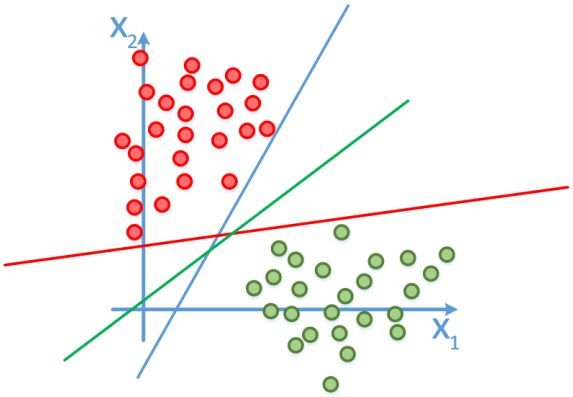
如果样本特征量很多，即样本维度很高，则相比于其他的分类器，SVM 的作用非常好，因为它只需要“支持向量”就可以分类，因此如果遇到几万维的样本（比如文本分类），则除了降维的方法（或者有效信息提取，整理和压缩等），得益于其核函数的 SVM 也可以做得非常好。在《机

器学士实战》[5] 的第一个机器学习例子中，对于 kNN 算法，如果是几万维的样本，比如 400X600 的图像，那么运算量岂不是超级大？而 SVM 可以通过一些加速算法有效避开不重要的样本，只需要少量的样本和构成支持向量的样本来不断优化，直到达到最优解。

关于 SVM 的基本介绍就讲这么多了，下面我们开始了解 SVM 分类器的基础思想。

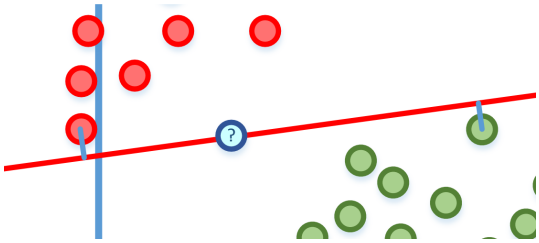
1.2 大间隔分类器

我们先假设一个最简单的情况：我们有一堆样本，这些样本一共分为两类，每个样本有两个特征，分别标记为 x_1 和 x_2，整个样本集的示意图表示为：

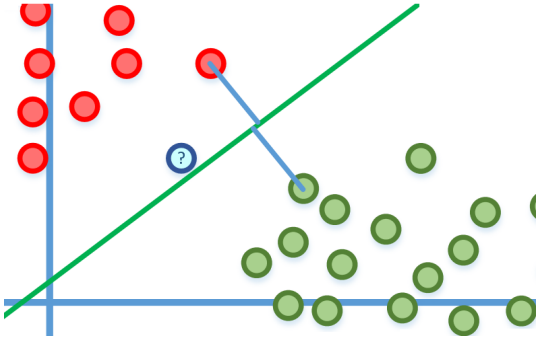


上图中，红色和绿色圆圈分别代表不同类别的样本；红线、蓝线和绿线分别是把这些样本分隔开的分界线（考虑到多维的情况，我们称之为超平面）。我们可以感受到，虽然红线和蓝线能把已经存在的样本正确地区分开，但是总归是不够好，而绿线则看起来非常完美。

红线和蓝线存在什么问题呢？我们看下图：



可以看到，两种样本中，距离超平面最近的样本离着超平面太近了，即图中蓝线标注出来的距离，而当出现一个新的样本，如图中蓝色圆圈，我们就很难把它进行分类。因此，我们希望样本距离超平面的距离尽可能远：



这样显而易见，蓝色圆圈应该被划分为红色圆圈一类。综上所述，我们需要我们的超平面能够尽可能离着最近的样本越远，这个距离（上图的蓝线）构成的向量我们称之为支持向量。根据超平面需要尽可能离着样本更远可知，我们需要让其间隔更大，即大间隔分类器。

Thinking 1.1 (间隔最大化的意义) 为什么要让间隔最大，这是因为几何间隔与样本误分次数 N 之间的关系为 $N \leq \left(\frac{2R}{\delta}\right)^2$，其中 δ 是样本集到分类面的间隔，R 是所有样本里向量长度最长的值，$R = max(||\boldsymbol{x}_i||)$（我们可以这么思考：当向量长度越长，则说明样本分布的范围越大，就越容易被误分；当样本间隔越大，说明不同的样本区别越明显，就越容易分类正确）。（这里关于误分次数和间隔的关系式是我在很多网络博客和发布在网站的 PPT 上找到的，但我实在不清楚它是哪篇论文或者著作来证明的，我也问了很多从事机器学习研究的人，他们都说不知道，据网上说这还是某个面试题。）

1.3 超平面的表示

在二维平面上，假如两轴各代表样本的一个特征 x_1 和 x_2，因此平面上的一个分隔线（高维就是超平面，下面用超平面来描述）可以定义为：

$$w_1 x_1 + w_2 x_2 + b = 0 \tag{1.3.1}$$

即可以写为：

$$\boldsymbol{w}^T \boldsymbol{x} + b = 0 \tag{1.3.2}$$

当扩展到多维时也是如此。

某点离超平面的距离可以计算为：

$$r = \frac{|\boldsymbol{w}^T \cdot \boldsymbol{x} + b|}{||\boldsymbol{w}||} \tag{1.3.3}$$

$$||\boldsymbol{w}|| = \sqrt{w_1^2 + w_2^2} \tag{1.3.4}$$

$||\boldsymbol{w}||$ 在数学上被称为 $\boldsymbol{w}$ 的二范数。因为很简单，上式的推导过程暂且不提（可以自行上网查找“点到直线的距离推导”）。

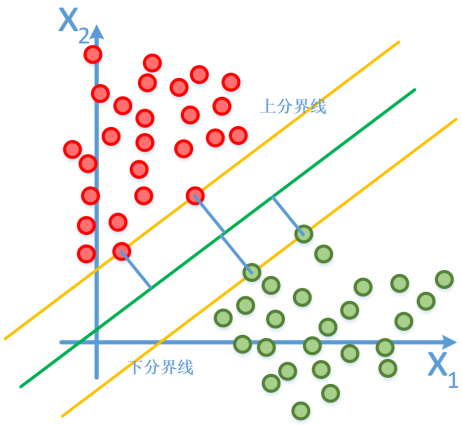
如何增大间隔，其实根据公式我们可以看出，可以减小 $||\boldsymbol{w}||$ 或者增大 $|\boldsymbol{w}^T \cdot \boldsymbol{x} + b|$。我们通常会选择固定 $|\boldsymbol{w}^T \cdot \boldsymbol{x} + b|$ 并努力缩小 $||\boldsymbol{w}||$ 的值，而不是固定 $||\boldsymbol{w}||$ 来寻找最大的间隔：想想看，比如上面的例子，$\boldsymbol{w}$ 表示的是曲线的倾斜程度，即不同的 $\boldsymbol{w}$ 描述的曲线倾斜度不一样，如果我们固定了倾斜度（注意这里的倾斜度不是斜率，因为横纵轴都是样本特征值，而不是笛卡尔坐标系来描述的函数关系），则无法找到更好的超平面了。

我们先不考虑固定 $|\boldsymbol{w}^T \cdot \boldsymbol{x} + b|$ 的意义，我们现在只考虑我们的目标，即 $min(||\boldsymbol{w}||)$，也就是等价于 $min(\frac{1}{2}||\boldsymbol{w}||^2)$，这里的 $\frac{1}{2}$ 是为了求导之后抵消掉平方项带来的 2。虽然我们的目标是要使得 $||\boldsymbol{w}||$ 最小，但我们还是需要加约束条件，因为其实当 $\boldsymbol{w}$ 里的分量都是 0 的时候它就是最小的，但任何样本计算以后得到的 $\boldsymbol{w}^T \boldsymbol{x} + b$ 值就都是 b 了，这样还怎么区分呢？

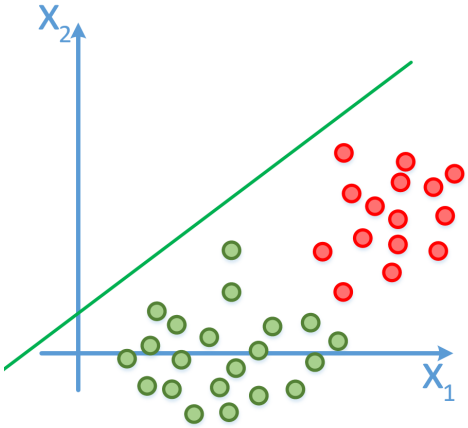
于是，我们设对于上下分隔面上的点来说，$|w^T \cdot x + b|$ 固定为 1，如下图。这样也就是说，对于所有样本点，都是要满足：

$$|w^T \cdot x + b| \geq 1 \tag{1.3.5}$$

至于为什么是 1，后面我会再进行描述。其实其他值也是可以的，但是 1 更简洁。



或许聪明的读者发现了一个问题，我们这个限定条件恐怕也不够呀？因为大家可以想象如果我们的中央分隔面在下面这个地方，不也能保证 $|w^T \cdot x + b| \geq 1$ 吗？



那么我们就这么来设定，即 $y_i = f(w^T x_i + b)$：

$$w^T x_i + b \geq +1 \quad y_i = +1 \tag{1.3.6}$$

$$w^T x_i + b \leq -1 \quad y_i = -1 \tag{1.3.7}$$

于是，我们的目标和限定条件就是：

$$aim : max\left(\frac{2}{\|w\|}\right) \Longrightarrow min(\frac{1}{2}\|w\|^2) \tag{1.3.8}$$

$$constraint : y_i \times \left(w^T x_i + b\right) \geq 1 \quad for\ all\ i \tag{1.3.9}$$

1.4 间隔与几何间隔

我们再来考虑前面的问题，即上下分隔面上的点 $|w^T \cdot x + b|$ 固定为常数以后会发生什么。

为了方便起见，我们把上下分隔面上的点 $\boldsymbol{x}$ 计算得到的 $2\frac{|\boldsymbol{w}^T\cdot\boldsymbol{x}+b|}{\|\boldsymbol{w}\|}$ 记做“几何间隔”（这里的 2 是因为几何间隔包括上分隔面和下分隔面距离中间间隔线的距离之和），把 $|\boldsymbol{w}^T\cdot\boldsymbol{x}+b|$ 称作“间隔”。以后我们都使用这些术语来描述。

也就是说，当间隔固定为 1 时，几何间隔就为 $\gamma=\frac{2}{\|\boldsymbol{w}\|}$，即两个异类的支持向量到分隔线的距离之和。

假如我们已经找到了最大几何间隔，我们可以得到：$\gamma\|\boldsymbol{w}\|=2$。

Thinking 1.2 (间隔值不影响超平面位置) 其实对于一个超平面而言，参数等比例变化不会影响超平面的位置，也就是说，下面的超平面其实是一样的：

$$w_1x_1+w_2x_2+b=0 \tag{1.4.1}$$

$$a\times(w_1x_1+w_2x_2+b)=0 \tag{1.4.2}$$

我们设间隔为 γ'，因为这个 γ' 是最小间隔，所以，对于其中所有的样本 $\boldsymbol{x}$：

$$min\Big(y_i\times(\boldsymbol{w}\boldsymbol{x}+b)\Big)=\gamma' \tag{1.4.3}$$

$$\Longrightarrow min\Big(y_i\times(\frac{\boldsymbol{w}}{\gamma'}\boldsymbol{x}+\frac{b}{\gamma'})\Big)=1 \tag{1.4.4}$$

所以我们可以很明确的得到：

$$y_i\times(\frac{\boldsymbol{w}}{\gamma'}\boldsymbol{x}+\frac{b}{\gamma'})\geq 1 \tag{1.4.5}$$

$$\Longrightarrow y_i\times(\boldsymbol{w}\boldsymbol{x}+b)\geq\gamma' \tag{1.4.6}$$

综上所述，我们无论设置间隔 γ' 为多少，都不会影响我们的原始优化问题，设置为 1 正是为了运算简单。当然，对于几何间隔而言，系数缩放并不影响几何间隔值：

$$\frac{2|\frac{1}{\gamma'}\boldsymbol{w}\cdot\boldsymbol{x}+b|}{\left\|\frac{1}{\gamma'}\boldsymbol{w}\right\|}=\frac{2|\boldsymbol{w}\cdot\boldsymbol{x}+b|}{\|\boldsymbol{w}\|} \tag{1.4.7}$$

2. 分类问题的优化方法

本章介绍 *SVM* 的优化方法，即如何找到分类间隔。

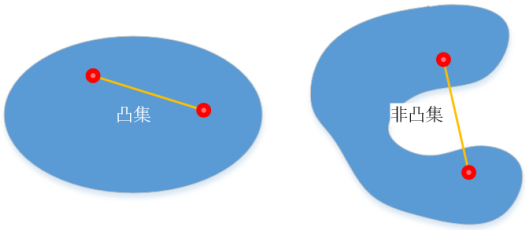
2.1 凸优化问题简介

上一章的公式中，$\boldsymbol{x}$ 可以被认为是变量，$\boldsymbol{w}$ 是系数，所有可以取值的 $\boldsymbol{w}$ 构成一个集合，但是根据前面所述，这个集合是有一定限定条件的，不能随便取值。

对于限定条件：

$$y_i \times \left(\boldsymbol{w}^T \boldsymbol{x}_i + b\right) \geq 1 \tag{2.1.1}$$

而它的可选区间是一个凸集，凸集就是，在集合中任选两点连线，它们连线上的任意一点都在这个集合中，比如下图左，而非凸集两点连线可能不在同一个集合中，如下图右。



对于上面的线性条件，$\boldsymbol{w}$ 的取值区域就是一个凸集，而又因为我们要最小化的目标是 $\frac{1}{2}\|\boldsymbol{w}\|^2$ 是一个二次函数，因此这就是一个凸二次规划问题（见 DezemingFamily 的《凸二次规划问题》）。凸二次规划问题不但有最优解，而且还可以通过计算来得到。

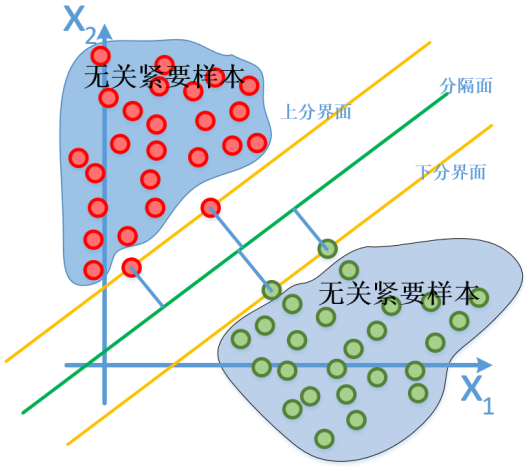
如何解不带约束项的函数最简单，只需要求极值即可。当只带等式约束项时，也不难解，只需要构造拉格朗日函数就可以转换为没有约束项的最优化问题，具体方法大家可以参考 DezemingFamily 的拉格朗日乘子法相关书。

但是我们当前的优化问题是带着不等式约束项的，这就给我们的优化问题带来了困难，不过好在，数学家们有办法解决，我们需要通过某种手段，将带不等式约束项的优化问题转换为只有等式约束项的优化问题。

2.2 转化最优解问题

在求解问题时，可以看到 b 只是起到截距的作用，也就是说，当你确定了 $\boldsymbol{w}$ 时也就确定了 b，同时，因为中间的分隔面是 $\boldsymbol{wx}+b=0$，你也会确定中间的分隔面。

我们思考一个重要问题，$\boldsymbol{w}$ 怎么确定？很显然，$\boldsymbol{w}$ 是恰好处于上下分界面上的点来确定的，而跟其他点无关。其他点只要分类正确就可以了，而上下分界面上的点就是决定其值的标准：



我们上一章讲过，因为间隔为 1，两个异类支持向量的间隔和就是 2。因为 $\boldsymbol{w}$ 只与支持向量有关，所以我们其实可以把这个优化问题理解为在一定约束条件下的优化（设样本数为 n）：

$$\begin{cases} \max_{w,b} \frac{2}{\|w\|} \\ s.t. \quad y_i \times \left(\boldsymbol{w}^T \boldsymbol{x_i} + b\right) \geq 1, \quad i = 1,2,...,n \end{cases} \tag{2.2.1}$$

我们其实可以转化为：

$$\begin{cases} \min_{w,b} \|w\| \\ s.t. \quad y_i \times \left(\boldsymbol{w}^T \boldsymbol{x_i} + b\right) \geq 1, \quad i = 1,2,...,n \end{cases} \tag{2.2.2}$$

根据《拉格朗日乘子法——带不等式约束项的函数优化》，可以把这个问题写作下面的表示方法：

$$\begin{cases} \min_{w,b} \frac{1}{2} \|w\|^2 \\ s.t. \quad 1 - y_i \times \left(\boldsymbol{w}^T \boldsymbol{x_i} + b\right) \leq 0, \quad i = 1,2,...,n \end{cases} \tag{2.2.3}$$

之前已经讲过，$\|\boldsymbol{w}\|^2 = \boldsymbol{w}^T \boldsymbol{w}$ 是为了方便求偏导计算，毕竟优化目标是一致的。

我们使用拉格朗日乘子法，得到下面的拉格朗日函数：

$$\begin{cases} L(\boldsymbol{w}, b, \boldsymbol{\alpha}) = \frac{1}{2} \|\boldsymbol{w}\|^2 + \sum_{i=1}^{n} \alpha_i \left(1 - y_i \times (\boldsymbol{w}^T \boldsymbol{x_i} + b)\right) \\ s.t. \quad \alpha_i \geq 0 \\ \qquad 1 - y_i \times (\boldsymbol{w}^T \boldsymbol{x_i} + b) \leq 0 \end{cases} \tag{2.2.4}$$

这个带约束项的问题可以转化为（这个转化的证明来自于运筹学与凸优化）：

$$\begin{cases} \min_{w,b} \max_{\alpha} L(\boldsymbol{w}, b, \boldsymbol{\alpha}) \\ s.t. \quad \alpha_i \geq 0 \end{cases} \tag{2.2.5}$$

Thinking 2.1 (转化的合理性思考) 我们简单来分析一下它的转化的合理性。由于 $\boldsymbol{w}$ 和 b 并没有明确的限制，所以理论上可以取任何值。

(1) 如果 $1-y_i\times(\boldsymbol{w}^T\boldsymbol{x_i}+b)>0$，则理论上 $\max_\alpha L(\boldsymbol{w},b,\boldsymbol{\alpha})$ 则可以无限大，因此就失去了意义。
(2) 如果 $1-y_i\times(\boldsymbol{w}^T\boldsymbol{x_i}+b)\le 0$，则理论上 $\max_\alpha L(\boldsymbol{w},b,\boldsymbol{\alpha})$ 就存在最大值，也就是 0（当 $\boldsymbol{\alpha}$ 都是 0 时），因此原式就等于 $\min_{w,b}\frac{1}{2}\|\boldsymbol{w}\|^2$。

综合（1）和（2），就能得到：

$$\min_{w,b}\max_{\alpha}L(\boldsymbol{w},b,\boldsymbol{\alpha})=\min_{w,b}\left(\infty,\frac{1}{2}\|\boldsymbol{w}\|^2\right) \tag{2.2.6}$$

这只是直觉上的解释，具体的证明还得依赖于凸优化原理。

转化以后的问题仍然需要继续转换，转化为对偶问题：

$$\begin{cases}\max_\alpha\min_{w,b}L(\boldsymbol{w},b,\boldsymbol{\alpha})\\ s.t.\quad \alpha_i\ge 0\end{cases} \tag{2.2.7}$$

Thinking 2.2 (对偶问题的简介) 弱对偶关系（凤尾大于鸡头）：

$$\min\max L\ge\max\min L \tag{2.2.8}$$

强对偶关系：

$$\min\max L=\max\min L \tag{2.2.9}$$

对于一个凸二次规划问题，是满足强对偶关系的，所以可以这么转换。

求解 $\max_\alpha L(\boldsymbol{w},b,\boldsymbol{\alpha})$，直接对 $\boldsymbol{w}$ 和 b 的偏导为 0，可以得到：

$$\begin{cases}\boldsymbol{w}=\sum_{i=1}^n\alpha_i y_i\boldsymbol{x_i}\\ 0=\sum_{i=1}^n\alpha_i y_i\end{cases} \tag{2.2.10}$$

Thinking 2.3 (一些简单的思考) 我们先不考虑把上面的式子代入到拉格朗日函数中，而是停下来，思考一下上面两个式子的形式。

可以看到，$\boldsymbol{w}$ 的值是样本点 $\boldsymbol{x_i}$ 的线性组合；其次，对 y_i 的同样系数的线性组合结果竟然是 0，为什么会这样？

首先，既然 $\boldsymbol{w}$ 肯定是仅仅由样本就能够确定的，所以 $\boldsymbol{w}=f(x_1,x_2,...,x_n)$。我们回想，平面

最少两点就能确定一条直线，立体空间最少三个点就能确定一个平面。

假如平面中已知两点 P_1 和 P_2，则过两点直线中任意一点可以表示为 $P = P_1 + k(P_2 - P_1)$。假如立体空间中已知不共线的三点 P_1、P_2 和 P_3，则过三点的平面中任意一点可以表示为 $P - P_1 = k_1(P_2 - P_1) + k_2(P_3 - P_1)$。由此看出，分隔面与样本点之间的关系符合线性关系。

同时，$\alpha_i \neq 0$ 仅在上下分界面上的样本点符合，其他点都是 $\alpha_i = 0$，我们就可以大概感受一下为什么对 $\alpha_i y_i$ 的线性组合得到的结果是 0 了。

我们把得到的两个式子代入到拉格朗日函数中，消去 $\boldsymbol{w}$ 和 b：

$$\begin{cases} \max_{\alpha} \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j \boldsymbol{x_i}^T \boldsymbol{x_j} \\ s.t. \quad \sum_{i=1}^{n} \alpha_i y_i = 0 \\ \qquad \alpha_i \geq 0, \quad i = 1, 2, ..., n \end{cases} \tag{2.2.11}$$

我们设函数模型为 $f(\boldsymbol{x}) = \boldsymbol{w}^T \boldsymbol{x} + b$，列出相关的 KKT 条件（原始、对偶问题具有强对偶关系的充分必要条件，这是由强对偶关系来得到的）：

$$\begin{cases} \alpha_i \geq 0 \\ y_i f(\boldsymbol{x_i}) - 1 \geq 0 \\ \alpha_i \Big(y_i f(\boldsymbol{x_i}) - 1 \Big) = 0 \end{cases} \tag{2.2.12}$$

其中，$\alpha_i \Big(y_i f(\boldsymbol{x_i}) - 1 \Big) = 0$ 被称为松弛互补条件，是最重要的 KKT 条件。α_i 是大于等于 0 的，括号里面的部分小于等于 0。

可以看到，当 $\alpha_i > 0$ 时，$y_i f(\boldsymbol{x_i}) = 1$ 就可以成立，说明该向量 $\boldsymbol{x_i}$ 是在上下分隔面上的点，构成支持向量。否则，$y_i f(\boldsymbol{x_i}) \neq 1$，则 $\alpha_i = 0$，$\boldsymbol{x_i}$ 不再是上下分隔面上的点，不构成支持向量。

因此，当训练完以后，SVM 只与支持向量有关，其他向量都不再会被考虑了。

由于样本里一定会存在 (x_k, y_k)，满足 $y_k f(\boldsymbol{x_k}) - 1 = 0$，可以推出：

$$y_k \Big(\boldsymbol{w}^T \boldsymbol{x}_k + b \Big) = 1 \tag{2.2.13}$$

$$\Longrightarrow y_k^2 \Big(\boldsymbol{w}^T \boldsymbol{x}_k + b \Big) = y_k \tag{2.2.14}$$

$$\Longrightarrow \Big(\boldsymbol{w}^T \boldsymbol{x}_k + b \Big) = y_k \tag{2.2.15}$$

$$\Longrightarrow b = y_k - \boldsymbol{w}^T \boldsymbol{x}_k = y_k - \sum_{i=1}^{N} \alpha_i y_i \boldsymbol{x}_i^T \boldsymbol{x}_k \tag{2.2.16}$$

这样就得到了该超平面的参数表示。

Thinking 2.4 (拉格朗日对偶性简述) 所谓二次规划问题，比如上面的 min 最优化函数：

$$\min_{w,b} \frac{1}{2} \|\boldsymbol{w}\|^2 \tag{2.2.17}$$

$$s.t. \quad 1 - y_i \times \Big(\boldsymbol{w}^T \boldsymbol{x_i} + b \Big) \leq 0, \quad i = 1, 2, ..., n \tag{2.2.18}$$

这里的 $\|\boldsymbol{w}\|^2$ 就是一个二次函数，所以称为二次规划。且由于这是凸函数，所以称为凸二次规划。

在做二次规划问题时，可以从主问题和对偶问题两个角度来思考，根据拉格朗日对偶性，将原始问题转化为对偶问题，然后通过解对偶问题来得到原始问题的解。对偶问题的复杂度往往会低于主问题。

关于对偶问题的原理我不在本书里详细介绍，因为如果单独拎出来一章讲解对偶会打乱 SVM 的讲解规划。不了解对偶性并不会耽误后面对于 SVM 和核函数的理解，该内容可以运筹学书籍中找到详细的解释和证明。

2.3 求解方法简述

虽然上述问题是一个二次规划问题，但我们很难用常规的二次规划方法求解，因为样本数量越多，这个问题的规模就越大，求解就越慢。

我们可以看出，SVM 难的地方并不是它的思想，而是它的求解方法——怎么样找到最合理的支持向量？

我们再来看一下我们要求解的问题：

$$\begin{cases} \max_{\alpha} \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j \boldsymbol{x}_i^T \boldsymbol{x}_j \\ s.t. \quad \sum_{i=1}^{n} \alpha_i y_i = 0 \\ \qquad \alpha_i \geq 0, \quad i = 1, 2, ..., n \end{cases} \tag{2.3.1}$$

一般现在主要使用 SMO 算法来求解，它首先选择其中的两个参数 α_i 和 α_j：

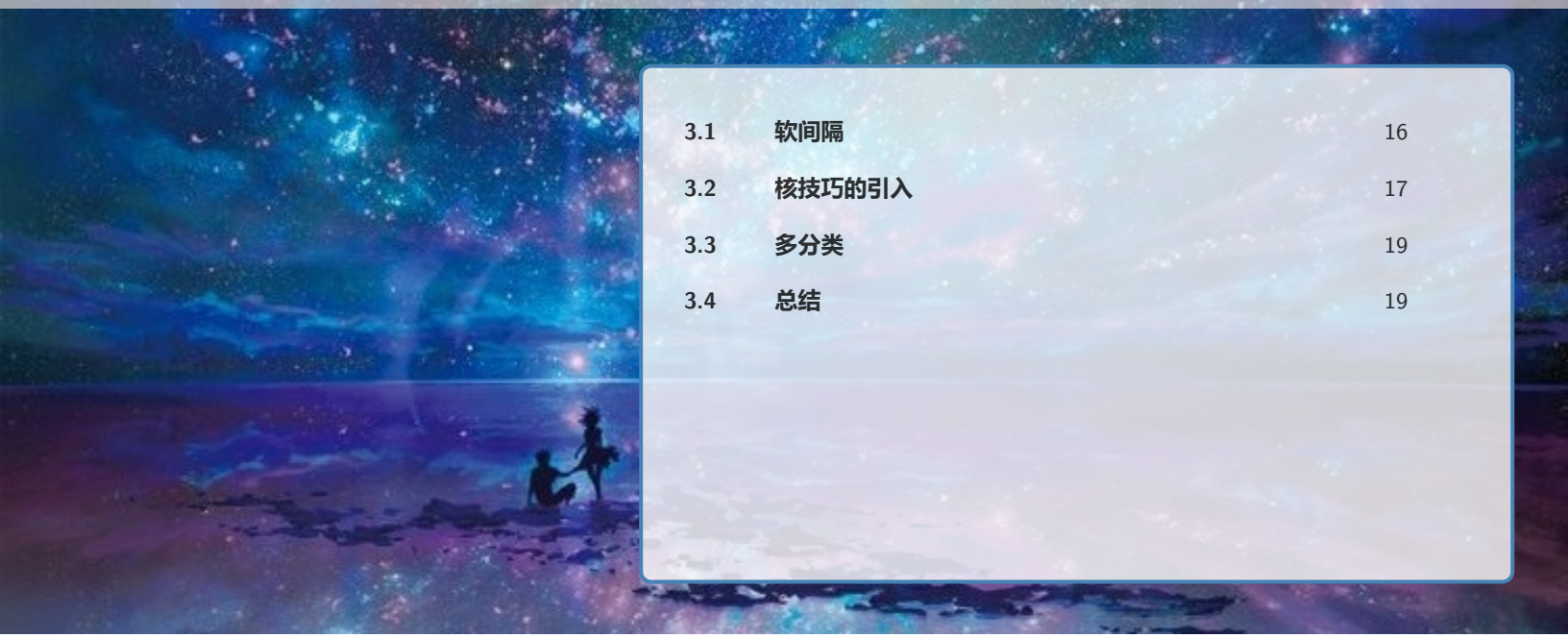
$$\sum_{i=1}^{n} \alpha_i y_i = 0 \tag{2.3.2}$$

$$\alpha_i y_i + \alpha_j y_j = -\sum_{k \neq i,j} \alpha_k y_k = c \tag{2.3.3}$$

我们再用 $\alpha_i y_i + \alpha_j y_j = c$ 消去变量 α_i，就得到了单变量二次规划问题，求解就会容易一些。

SMO 求解有一大堆理论基础，我们暂时先不考虑这些内容，我也不想把这个入门的小册子给搞成长篇大论（主要原因还是因为很久没有再看过优化理论了，很多内容也都忘了，但这并不妨碍我们使用 PRTools 或者 sklearn 的 svm，所以这里就暂时先不提这些内容）。

3. 更实际的问题



本章主要讲解核函数和软间隔这种实际分类中会遇到的情况。

3.1 软间隔

软间隔就是可以允许一定的错误，即下面的条件不一定对所有样本 $\boldsymbol{x_i}$ 都成立：

$$1-y_i\times(\boldsymbol{w}^T\boldsymbol{x_i}+b)\le 0 \tag{3.1.1}$$

最优化的目标就可以写为：

$$\min_{w,b}\frac{1}{2}\boldsymbol{w}^T\boldsymbol{w}+loss \tag{3.1.2}$$

我们可以使用距离来表示：

$$if\ \ 1-y_i\times(\boldsymbol{w}^T\boldsymbol{x_i}+b)\le 0\Longrightarrow loss=0 \tag{3.1.3}$$

$$if\ \ 1-y_i\times(\boldsymbol{w}^T\boldsymbol{x_i}+b)>0\Longrightarrow loss=1-y_i\times(\boldsymbol{w}^T\boldsymbol{x_i}+b) \tag{3.1.4}$$

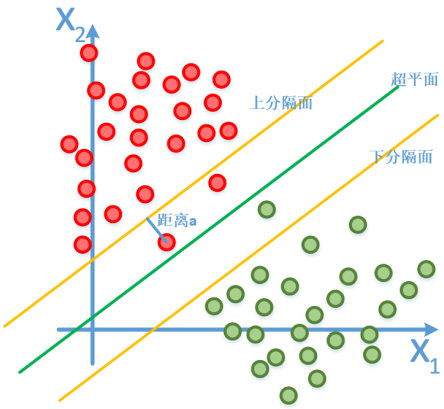
于是最优化目标就是（其中，C 是常数）：

$$\min_{w,b}\left\{\frac{1}{2}\boldsymbol{w}^T\boldsymbol{w}+C\cdot\sum_{i=1}^{N}\max\left[0,1-y_i\times(\boldsymbol{w}^T\boldsymbol{x_i}+b)\right]\right\} \tag{3.1.5}$$

令 $\max\left[1-y_i\times(\boldsymbol{w}^T\boldsymbol{x_i}+b)\right]=\xi_i$，于是 $\xi_i\ge 0$。原式就可以写为：

$$\min_{w,b}\left\{\frac{1}{2}\boldsymbol{w}^T\boldsymbol{w}+C\cdot\sum_{i=1}^{N}\xi_i\right\} \tag{3.1.6}$$

$y_i\times(\boldsymbol{w}^T\boldsymbol{x_i}+b)$ 是样本距离上下分隔面的距离。在超平面上，$\boldsymbol{w}^T\boldsymbol{x_i}+b=0$，见下图。对于红点来说，在上分隔面以及以上的点，$y_i\times(\boldsymbol{w}^T\boldsymbol{x_i}+b)\ge 1$，此时 $\xi_i=0$；否则，$y_i\times(\boldsymbol{w}^T\boldsymbol{x_i}+b)<1$，此时 $\xi_i>0$。



注意，会出现 $y_i \times (\boldsymbol{w}^T \boldsymbol{x_i} + b)$ 小于 0 甚至小于-1 的情况，这说明超平面并不能完全把两类样本分开。

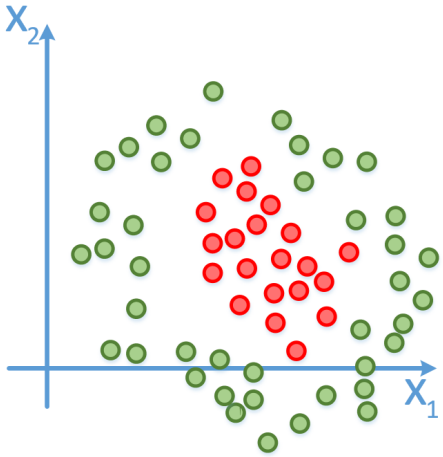
上图的“距离 a”的间隔（注意是间隔，不是几何间隔）就是 ξ_i，该红点距离超平面的距离间隔就是 $1-\xi_i$，注意这个距离间隔可以为负（当超平面没法把这个红点区分开时，红点在超平面下方）。加上这些限定条件，优化目标就是：

$$\begin{cases} \min_{w,b} \left\{ \frac{1}{2} \boldsymbol{w}^T \boldsymbol{w} + C \cdot \sum_{i=1}^{N} \xi_i \right\} \\ s.t. \quad y_i(\boldsymbol{w}^T \boldsymbol{x_i} + b) \geq 1 - \xi_i \\ \qquad \xi_i \geq 0 \end{cases} \tag{3.1.7}$$

针对软间隔的情况，我们依然可以运用拉格朗日乘子法和转换成对偶问题来解。

3.2 核技巧的引入

在分类中，最简单的情况是二维线性可分（一维线性可分就没有什么意义了）；稍微复杂一点，就是没法完全分开（你中由我，我中有你），需要软间隔来容错；最复杂的情况就是二维线性不可分，比如：



对于异或问题非线性不可分，如果能通过一种转换，转换为线性，则就可以变得线性可分了。对于某个异或问题，类别 1：$[(1,0),(0,1)]$；类别 2：$[(0,0),(1,1)]$，我们发现类别 2 的两个坐标值

都相等，因此做下面一种映射就可以用一个平面来分开了（高维比低维更容易可分）：

$$\boldsymbol{x}_k \longrightarrow \phi(\boldsymbol{x}_k) \tag{3.2.1}$$

$$(z_1, z_2) \longrightarrow (z_1, z_2, (z_1 - z_2)^2) \tag{3.2.2}$$

我们再回顾一下优化目标：

$$\begin{cases} \max_{\alpha} \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j \boldsymbol{x}_i^T \boldsymbol{x}_j \\ s.t. \quad \sum_{i=1}^{n} \alpha_i y_i = 0 \\ \qquad \alpha_i \geq 0, \quad i = 1, 2, ..., n \end{cases} \tag{3.2.3}$$

注意里面有内积表示形式 $\boldsymbol{x}_i^T \boldsymbol{x}_j$，当对 $\boldsymbol{x}_k$ 做映射来升维时，其实优化目标应该就变为了：

$$\begin{cases} \max_{\alpha} \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j \phi(\boldsymbol{x}_i)^T \phi(\boldsymbol{x}_j) \\ s.t. \quad \sum_{i=1}^{n} \alpha_i y_i = 0 \\ \qquad \alpha_i \geq 0, \quad i = 1, 2, ..., n \end{cases} \tag{3.2.4}$$

我们可以定义一个函数，直接获得内积表示，而不是先进行映射，然后再求出内积，否则在高维甚至无限维情况下，计算量过大。这个函数其实就是核函数：

$$\mathcal{K}(\boldsymbol{x}_i, \boldsymbol{x}_j) = \phi(\boldsymbol{x}_i)^T \phi(\boldsymbol{x}_j) \tag{3.2.5}$$

比如我们举个例子，我们不用考虑 $\phi(\boldsymbol{x}_k)$ 的形式，而是直接定义一个核函数：

$$\mathcal{K}(\boldsymbol{x}_i, \boldsymbol{x}_j) = e^{-\frac{(x_i - x_j)^2}{2\sigma^2}} \tag{3.2.6}$$

因为我们不再需要求在优化时先求映射再求内积，所以它经常被称为核技巧 (kernel trick)。

我们一般会使用正定核函数，正定核函数中，$\phi \in \mathcal{H}$(希尔伯特空间)。希尔伯特空间是一个线性空间 (其实是向量空间)，它是完备的，可能是无限维空间，可以进行内积运算。我们当前只考虑实数域上的希尔伯特空间，完备性表示对于序列来说是收敛的，而且对加减数乘等都是封闭的。内积运算需要保证对称性（$< f, g >=< g, f >$）、正定性（$< f, f >\geq 0$，$< f, f >= 0 \Longleftrightarrow f = 0$）以及线性 $< r_1 f_1 + r_2 f_2, g >= r_1 < f_1, g > + r_2 < f_2, g >$。

我们先来看正定核函数满足的第一条性质，需要满足对称性（希尔伯特空间内积运算的对称性）：

$$\mathcal{K}(\boldsymbol{x}_i, \boldsymbol{x}_j) = \mathcal{K}(\boldsymbol{x}_j, \boldsymbol{x}_i) \tag{3.2.7}$$

然后还需要满足正定性，任取 N 个元素，$\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_N$，对于的 Gram 矩阵（$K = \left[\mathcal{K}(\boldsymbol{x}_i, \boldsymbol{x}_j)\right]$）是半正定的。由于核函数满足对称性，所以 Gram 是一个实对称矩阵。可以很简单就能证明，对于任意 N 维向量 $\boldsymbol{t}$：

$$\boldsymbol{t}^T \left[\mathcal{K}(\boldsymbol{x}_i, \boldsymbol{x}_j)\right] \boldsymbol{t} = \boldsymbol{t}^T \begin{bmatrix} \mathcal{K}(\boldsymbol{x}_1, \boldsymbol{x}_1) & \mathcal{K}(\boldsymbol{x}_1, \boldsymbol{x}_2) & \cdots & \mathcal{K}(\boldsymbol{x}_1, \boldsymbol{x}_N) \\ \mathcal{K}(\boldsymbol{x}_2, \boldsymbol{x}_1) & \mathcal{K}(\boldsymbol{x}_2, \boldsymbol{x}_2) & \cdots & \mathcal{K}(\boldsymbol{x}_2, \boldsymbol{x}_N) \\ \cdot & \cdot & \cdots & \cdot \\ \mathcal{K}(\boldsymbol{x}_N, \boldsymbol{x}_1) & \mathcal{K}(\boldsymbol{x}_N, \boldsymbol{x}_2) & \cdots & \mathcal{K}(\boldsymbol{x}_N, \boldsymbol{x}_N) \end{bmatrix} \boldsymbol{t} \geq 0 \tag{3.2.8}$$

3.3 多分类

我们其实很容易就能感受到，SVM 设计的初衷是为了进行二分类，如果想进行多分类，就需要一些改进。

最简单的多分类

最容易想到的方法，就是两分类——先把一个类别与其他所有类别区分开，然后再把剩下的类别再用一个 SVM 模型区分开一类。如果有 5 个类别，就需要 4 个 SVM 模型。但是这样可能会有问题，比如稳定性差——如果其中某个 SVM 模型对分类效果不好，则就会影响其他的分类，比如如果第二个 SVM 不能很好地把第二类和第三、四、五类区分开，那么第三个 SVM 就要从不好的分类中继续做分类。

假如某个样本本应该属于第四类，我们使用了第一个 SVM 以后判断其不属于第 1 类，那么就进入下一个环节。使用第二个 SVM 以后，假设第二个 SVM 分类效果不好，则有可能因为误差被判为属于第二类，即使被判为了不属于第二类，由于在训练时第二个 SVM 分类不好用，导致在训练第三个 SVM 时就会训练的也不好。

改进的多分类器

改进的方法就是每定义 5 个 SVM，每个的作用都是把一个类别与其他所有的类别区分开。比如假设一共有 5 个类别，第一个 SVM 是区分第 1 类与第 2,3,4,5, 类；第二个 SVM 是区分第 2 类与第 1,3,4,5 类；以此类推。

有可能，某一个样本在通过所有这 5 个 SVM 模型以后，都被判断为属于它那一类，我们可以根据其距离超平面的距离来决定它最终属于哪一类。如果通过所有 SVM 以后，发现这个样本不属于其中任何一类，我们一般就只能让它作为第 6 类——一个新类。

小结

其实关于 SVM 的多分类问题有很多理论性的研究，而这些方法与基础思想都是一致的，但是可以加强鲁棒性。大家可以参考比较老的几篇文献 [10][11][12] 来入门。

3.4 总结

到目前为止，关于支持向量机的基本概念和方法已经讲完了。虽然里面还有很多概念并没有详细解释，但由于这些概念都是与凸优化理论相关的，没有基础知识很难进行讲解，所以就暂时跳过了。本书作为 SVM 的入门学习已经基本足够，SVM 的重点内容就是：大间隔分类-> 对偶转化-> 软间隔-> 核技巧-> 求解方法。其中，对偶转化我们只讲解了概念和理解；求解方法我们仅仅做了引子（如果不把凸二次规划讲清楚，SMO 没法讲得很透彻）。

本书开始于 2021 年 5 月，但由于科研任务和安排，导致 12 月才再次开始进行编写。期间关于这第一本 SVM 的书应该介绍哪些内容也是做了很多次安排和规划，同时为了学习不同学者的算法理解，也观看了很多视频和文章，有数学方向的，也有非数学方向偏工程的。总之，这本轻体量的小书就写到这里啦，关于其他更高阶的内容，会陆续在其他的小书中发布。

Bibliography

[1] 周志华. 《机器学习》[J]. 中国民商, 2016, 03(No.21):93-93.

[2] https://study.163.com/course/courseMain.htm?courseId=1004570029

[3] http://www.blogjava.net/zhenandaci/category/31868.html

[4] https://blog.csdn.net/marising/article/details/

[5] Harrington P . Machine Learning in Action[M].

[6] https://www.bilibili.com/video/BV1Hs411w7ci?spm_id_from=333.999.0.0

[7] https://blog.csdn.net/williamchin/article/details/116590902 (对超平面间隔取 1 做了解释)

[8] https://blog.csdn.net/crazy_programmer_p/article/details/38553663 多分类简述

[9] https://www.zhihu.com/topic/20687611/top-answers 凸二次规划的解释

[10] https://zhuanlan.zhihu.com/p/35755150 对偶问题

[11] Krebel, U. H.-G. 1998. Pairwise classification and support vector machines. In Advances in Kernel Methods –Support Vector Learning, B. Scholkopf, C. J. C. Burges, and A. J. Smola, Eds., MIT Press, Cambridge, MA, 255-268.

[12] C. W. Hsu and C. J. Lin. 2002. A comparison of methods for multi-class support vector machines. IEEE Trans. Neural Netw. 13, 2, 415-425.

[13] J. Weston and C. Watkins. 1998. Multi-class support vector machines. Technical Report CSD-TR-98-04, Department of Computer Science, Royal Holloway, University of London.