十二、支持向量机(Support Vector Machines)

12.1 优化目标（SVM 的由来与逻辑回归的关系）

一、背景与动机

- 在监督学习中，算法之间（如逻辑回归、神经网络等）性能差异不大。
- **真正重要的**是：
    - 特征设计（feature engineering）
    - 正则化参数的选择
    - 数据量与数据质量
- **支持向量机（Support Vector Machine, SVM）**是一种在工业界与学术界广泛应用的、功能强大的监督学习算法。

二、从逻辑回归到支持向量机

1. 逻辑回归的核心思想

- 假设函数：

$$[h_\theta(x) = \frac{1}{1 + e^{-\theta^T x}}]$$

- 对于样本：
    - $$若(y=1)：希望(h_\theta(x) \to 1 \Rightarrow \theta^T x \gg 0)$$
      $$若(y=0)：希望(h_\theta(x) \to 0 \Rightarrow \theta^T x \ll 0)$$

2. 逻辑回归的代价函数（单个样本）

- $$Cost(h_\theta(x), y) = \{ * \log(h_\theta(x)), \& if y = 1 \ * \log(1 - h_\theta(x)), \& if y = 0$$

- 当 ($y = 1$) 且 ($\theta^T x$) 很大时，代价趋近于 0。
- 当 ($y = 0$) 且 ($\theta^T x$) 很小时，代价趋近于 0。

三、从逻辑回归代价函数到 SVM 的代价函数

1. 逻辑回归代价曲线（关于 ($z = \theta^T x$)）

- 对于 (y = 1)：代价在 (z) 增大时迅速下降趋近 0。
- 对于 (y = 0)：代价在 (z) 减小时迅速下降趋近 0。

2. SVM 的代价函数（近似线性化）

SVM 将逻辑回归的代价函数近似为**分段线性函数**：

- 当 (y = 1):
  代价函数 ($cost_1(z)$)
  - 当 ($z \geq 1$)：代价 = 0
  - 当 ($z < 1$)：代价随 (z) 减小线性增大
- 当 (y = 0):
  代价函数 ($cost_0(z)$)
  - 当 ($z \leq -1$)：代价 = 0
  - 当 ($z > -1$)：代价随 (z) 增大线性增加

📈 **几何意义**：
SVM 不仅要求分类正确（($\theta^T x > 0$)），
还要求“**离得足够远**”（(y=1) 时 ($\theta^T x \geq 1$)，(y=0) 时 ($\theta^T x \leq -1$)）。
这种“留出间隔”的思想就是 **margin（间隔）**。

四、SVM 的优化目标

1. 从逻辑回归的代价函数出发

逻辑回归最小化：

$$[J(\theta) = \frac{1}{m}\sum_i Cost(h_\theta(x^{(i)}), y^{(i)}) + \frac{\lambda}{2m}\sum_j \theta_j^2]$$

2. 替换为 SVM 的代价函数（分段线性）

SVM 的目标函数：
$[\min_\theta; C\sum_i [y^{(i)} cost_1(\theta^T x^{(i)}) + (1 - y^{(i)}) cost_0(\theta^T x^{(i)})] + \frac{1}{2}\sum_j \theta_j^2]$

五、参数 (C) 与 (λ) 的关系

- 在逻辑回归中，我们使用 (\lambda) 来控制正则化强度。
- 在 SVM 中，使用 **(C)**（常数）代替。
  - (C)：表示**对训练误差的惩罚权重。**
  - (C) 越大 → 更关注训练样本分类正确（更少正则化）
  - (C) 越小 → 更关注保持参数较小、间隔更大（更多正则化）

因此：

$$[C \approx \frac{1}{\lambda}]$$

六、SVM 的预测方式

- 支持向量机的假设函数为：

$$[h_\theta(x) = \{1, \quad \text{if } \theta^T x \geq 0 \; 0, \quad \text{if } \theta^T x < 0 \,]$$

  与逻辑回归不同，SVM **不输出概率**，只输出分类结果。

12.2 大间距（大边界）的直观理解

一. 代价函数与分类条件

SVM 的代价函数由两部分组成：

- 一部分衡量样本的分类误差；
- 另一部分是正则化项（权重平方和的一半）。

以样本标签 y 和输入 x 为例，定义 $z = \theta^T x$。
对于正样本 ($y = 1$)，代价函数 $\text{cost}_1(z)$ 只有在 $z \geq 1$ 时为 0。
对于负样本 ($y = 0$)，代价函数 $\text{cost}_0(z)$ 只有在 $z \leq -1$ 时为 0。

因此：

- 若 $y = 1$，希望 $\theta^T x \geq 1$；
- 若 $y = 0$，希望 $\theta^T x \leq -1$。

与逻辑回归相比，SVM 要求样本不仅被正确分开（即 $\theta^T x > 0$ 或 $\theta^T x < 0$），还要求其离决策边界有一定“安全间距”（margin）。

二. 当 C 很大时的情形

SVM 的优化目标为：

$$[\min_\theta ; C \sum_{i=1}^{m} \left[y^{(i)} \text{cost}_1(\theta^T x^{(i)}) + (1 - y^{(i)}) \text{cost}_0(\theta^T x^{(i)}) \right] + \frac{1}{2} \sum_{j=1}^{n} \theta_j^2]$$

当 C 很大（例如 $C = 100000$）时，算法会强烈倾向于使第一项（误差项）为 0。
此时，优化问题可近似为：

$$[\min_\theta \frac{1}{2} \sum_j \theta_j^2]$$

受约束条件：

$$[\{\theta^T x^{(i)} \geq 1, \quad 若\ y^{(i)} = 1 \; \theta^T x^{(i)} \leq -1, \quad 若\ y^{(i)} = 0]$$

这意味着我们希望在完全正确分类的前提下，使 θ 的范数尽可能小。

三. 最大间距分类的几何意义

考虑一个线性可分的数据集。
理论上，存在多条直线可以将正负样本完全分开。
但这些分界线的"稳定性"不同。

SVM 会选择那条能让样本距离决策边界最远的线，即**间距（margin）最大**的那条线。
几何上，黑色的最优决策边界距离最近样本的距离最大；
粉色或绿色的线虽然也能分开样本，但间距较小，分类鲁棒性较差。

因此，SVM 的优化本质上是：
在保证样本可分的条件下，使样本到决策边界的最小距离（margin）最大化。

四. 异常值与参数 C 的影响

当 C 非常大时，模型强制所有样本都被完全正确分类。
如果出现一个异常点（outlier），模型会为了"照顾"这个样本而显著改变决策边界（例如从黑线变为粉线），导致模型过拟合。

当 C 较小时，模型允许部分样本被错误分类（代价函数不为零），从而忽略异常点的影响。
此时决策边界更稳定、更合理（仍接近黑线）。

五. C 与正则化参数 λ 的关系

二者的关系为：

$$[C = \frac{1}{\lambda}]$$

因此：

- C **较大（λ 较小）**：更关注分类正确性，正则化弱，可能导致过拟合（高方差）。
- C **较小（λ 较大）**：更关注简化模型，正则化强，可能导致欠拟合（高偏差）。

六. 小结

- 支持向量机通过**最大化分类间距（margin）**来提高模型的鲁棒性。
- 当 C 很大时，SVM 退化为一个"硬间距分类器"（严格可分）。
- 当 C 适中时，SVM 能容忍少量分类错误，从而在实际数据中表现更好。
- 因此，SVM 被称为**大间距分类器**，其目标是在保证分类性能的前提下，最大化样本到决策边界的最小距离。

12.3 大间距分类背后的数学（选修）

一、内积与几何意义复习

1. 向量内积的定义

给定两个向量 u 和 v:

$$[u^T v = u_1 v_1 + u_2 v_2]$$

这是两向量的内积（dot product）。

几何意义上，若将 v 投影到 u 的方向上，投影长度记为 p，则:

$$[u^T v = p \cdot |u|]$$

其中 $|u| = \sqrt{u_1^2 + u_2^2}$ 表示向量 u 的长度（范数）。

---

2. 内积的符号含义

- 当 u 和 v 的夹角 **小于 90°** 时，$u^T v > 0$;
- 当夹角 **大于 90°** 时，$u^T v < 0$;
- 当夹角 **等于 90°** 时，$u^T v = 0$。

因此，内积不仅反映了两个向量的相似程度，还体现了它们的方向关系。

二、SVM 优化目标的几何意义

SVM 的目标函数如下:

$$[\min_\theta; C \sum_i \text{cost}(y^{(i)}, \theta^T x^{(i)}) + \frac{1}{2} \sum_j \theta_j^2]$$

为方便说明，做以下简化:

- 忽略截距项：设 $\theta_0 = 0$;
- 特征维度：$n = 2$。

此时，目标函数可写为:

$$[\frac{1}{2}(\theta_1^2 + \theta_2^2) = \frac{1}{2}|\theta|^2]$$

这说明 SVM 的优化目标是:

> **最小化参数向量 θ 的范数平方，即使参数向量尽可能短。**

三、约束条件的几何解释

SVM 在大 C 情形下的约束条件为:

$$[\{\theta^T x^{(i)} \geq 1, \quad y^{(i)} = 1 \; \theta^T x^{(i)} \leq -1, \quad y^{(i)} = 0]$$

根据前述内积的几何定义，有:

$$[\theta^T x^{(i)} = p^{(i)} \cdot |\theta|]$$

其中 $p^{(i)}$ 是样本 $x^{(i)}$ 在向量 θ 方向上的投影长度。

因此，约束可重写为：

$$[\{p^{(i)} \cdot |\theta| \geq 1, \quad y^{(i)} = 1\, p^{(i)} \cdot |\theta| \leq -1, \quad y^{(i)} = 0]$$

四、决策边界与间距的推导

1. 决策边界方向

决策边界总是**垂直于参数向量** θ 的。
当 $\theta_0 = 0$ 时，决策界通过原点；$\theta_0 \neq 0$ 时，决策界为一条平移后的平面（或直线）。

2. 较差的决策界

如果决策界靠近样本点：

- 样本的投影 $p^{(i)}$ 很小；
- 为了满足约束 $p^{(i)}|\theta| \geq 1$，必须让 $|\theta|$ 很大；
- 但目标函数要求 $|\theta|$ 尽量小；
- 因此这样的决策界**不是最优解**。

3. 优秀的决策界

如果决策界距离样本点较远：

- 样本的投影 $p^{(i)}$ 较大；
- 可以使用较小的 $|\theta|$ 满足约束；
- 因此能使目标函数 $\frac{1}{2}|\theta|^2$ 取更小的值；
- 最终得到一个**更大的间距（margin）**。

这说明：

> SVM 在最小化参数范数的同时，会选择能使样本在 θ 方向投影距离尽可能大的决策界，从而实现最大间距分隔。

五、数学结论：大间距的来源

- 支持向量机通过**最小化** $\frac{1}{2}|\theta|^2$，即使参数向量尽可能短；
- 同时必须满足所有样本的约束条件（正样本投影≥1，负样本投影≤−1）；
- 在几何上，这等价于**最大化决策边界与样本点之间的最小距离**；
- 因此，SVM 的最优解自然对应于**最大间距分类器（Large Margin Classifier）**。

六、扩展到 $\theta_0 \neq 0$ 的情形

当 $\theta_0 \neq 0$ 时：

- 决策界不再经过原点；
- 但推导过程和几何意义完全类似；
- 依然可以证明 SVM 会找到**最大间距的分隔超平面**。

七、小结

1. **内积的几何意义**：样本在参数向量方向上的投影。
2. **目标函数**：最小化 $\frac{1}{2}|\theta|^2$。
3. **约束条件**：保证样本被正确分类，并在两侧至少保持 1 的安全间距。
4. **核心思想**：
   - 当样本到边界的投影距离变大时，模型可使用更小的 $|\theta|$；
   - 这使得支持向量机自动选择最大间距的分隔界。
5. **结论**：SVM 之所以能产生大间距分类，是因为它在优化中同时平衡了分类正确性与参数范数的最小化。

12.4 核函数 1（Kernels I）

一、从多项式特征到核函数的动机

在非线性可分的分类问题中，我们可以通过添加多项式特征来改进模型。
例如，模型可以写成：

$$[h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_1 x_2 + \theta_4 x_1^2 + \theta_5 x_2^2 + \cdots]$$

这实际上是对输入特征进行组合、平方、交叉等处理，以形成更复杂的决策边界。
我们也可以定义新的特征：

$$[f_1 = x_1, \quad f_2 = x_2, \quad f_3 = x_1 x_2, \quad f_4 = x_1^2, \quad f_5 = x_2^2]$$

于是模型可以写为：

$$[h_\theta(x) = \theta_1 f_1 + \theta_2 f_2 + \cdots + \theta_n f_n]$$

但除了直接构造这些组合特征外，我们能否找到一种更灵活、更自动的方式来生成这些新特征？
这正是核函数（Kernel Function）的作用。

二、核函数与地标（Landmarks）

核函数的思想是：
给定一个样本 (x)，我们根据它与一组“地标”（landmarks）之间的相似度，定义新的特征 ($f_1, f_2, f_3, \ldots$)。

假设我们选取三个地标 ($l^{(1)}, l^{(2)}, l^{(3)}$)。
则新的特征可定义为：

$$[f_1 = \text{similarity}(x, l^{(1)}), \quad f_2 = \text{similarity}(x, l^{(2)}), \quad f_3 = \text{similarity}(x, l^{(3)})]$$

三、高斯核函数（Gaussian Kernel）

一种常用的核函数是高斯核（Gaussian Kernel）：

$$[\text{similarity}(x, l) = e^{-\frac{|x-l|^2}{2\sigma^2}}]$$

其中：

$$[|x - l|^2 = \sum_{j=1}^{n}(x_j - l_j)^2]$$

表示样本 (x) 与地标 (l) 之间的欧氏距离平方。
需要注意：此函数与正态分布并无实际关联，只是形式相似。

其性质如下：

- $$若(x)与地标(l)距离很近，则(|x - l|^2 \approx 0)，于是(\text{similarity} \approx e^0 = 1)；$$
- $$若距离很远，则(\text{similarity} \approx 0)。$$

因此，每个地标定义了一个"局部区域"，样本越靠近该地标，其对应的特征值 (f) 越接近 1。

四、参数 (σ) 的影响

参数 (σ) 控制函数变化的"宽度"：

- (σ) 较大：函数变化缓慢，影响范围广；
- (σ) 较小：函数变化剧烈，影响范围小。

因此，(σ) 决定了"局部性"的强弱。

五、基于核函数的决策示例

假设我们有三个地标 (l^{(1)}, l^{(2)}, l^{(3)})，并为每个输入样本计算：
$[f_1 = e^{-\frac{|x-l^{(1)}|^2}{2\$\sigma\$^2}}, \quad f_2 = e^{-\frac{|x-l^{(2)}|^2}{2\$\sigma\$^2}}, \quad f_3 = e^{-\frac{|x-l^{(3)}|^2}{2\$\sigma\$^2}}]$

若一个样本点靠近 ($l^{(1)}$)，则 ($f_1 \approx 1$)，而 ($f_2, f_3 \approx 0$)。
模型预测：
$[h_\theta(x) = \theta_0 + \theta_1 f_1 + \theta_2 f_2 + \theta_3 f_3 > 0 \Rightarrow y = 1]$

这样一来，我们并非直接使用 (x_1, x_2) 等原始特征，而是使用由核函数生成的新特征 (f_1, f_2, f_3)。
这些新特征能自动产生非线性决策边界。

12.5 核函数 2（Kernels II）

一、地标的选择

通常我们会将训练集中每个样本都作为一个地标：

$$[l^{(1)} = x^{(1)}, \quad l^{(2)} = x^{(2)}, \quad \dots, \quad l^{(m)} = x^{(m)}]$$

于是每个样本 (x) 都会对应一个 (m) 维的新特征向量：

$$[f = [\text{similarity}(x, l^{(1)}), \text{similarity}(x, l^{(2)}), \dots, \text{similarity}(x, l^{(m)})]]$$

二、核函数在支持向量机中的使用

修改后的假设函数为：

$$[h_\theta(x) = \{1, \quad 若\ \theta^T f \geq 0\ 0, \quad 否则]$$

对应的代价函数为：

$$[\min_\theta; C \sum_{i=1}^{m} \left[y^{(i)} \cos t_1(\theta^T f^{(i)}) + (1 - y^{(i)}) \cos t_0(\theta^T f^{(i)}) \right] + \frac{1}{2}\theta^T M \theta]$$

其中矩阵 (M) 由所选核函数决定，用于简化计算。

三、为什么逻辑回归不常用核函数

理论上，逻辑回归也可以使用核函数。
但由于逻辑回归没有 SVM 中的矩阵简化技巧（即 ($\theta^T M\theta$) 替代 ($\theta^T\theta$)），
计算开销会极大，因此在实际中很少采用。

四、SVM 的实现与核选择

在实践中，通常使用现成的软件包来训练 SVM，例如：

- **liblinear**
- **libsvm**

在使用高斯核函数前，应对数据进行**特征缩放（feature scaling）**。
若选择不使用核函数，则称为**线性核函数（linear kernel）**。
当特征维度高、样本量较小时，线性 SVM 通常表现良好。

五、参数 (C) 与 (σ) 的影响

- $$(C = 1/\lambda)$$

参数	变化方向	模型表现	偏差 / 方差趋势
(C) 大	正则化弱	过拟合倾向	高方差、低偏差
(C) 小	正则化强	欠拟合倾向	高偏差、低方差
(σ) 大	核范围宽	平滑、欠拟合	高偏差、低方差
(σ) 小	核范围窄	边界复杂、过拟合	低偏差、高方差

六、小结

1. 核函数用于将原始特征映射到高维空间，从而实现非线性分类。
2. 高斯核是最常用的核函数，其核心思想是基于距离的相似度。
3. 选择合适的参数 (C) 和 (σ) 对模型的泛化能力至关重要。
4. 支持向量机通过核函数能有效处理复杂的分类边界，而无需显式计算高维特征。

12.6 使用支持向量机（Using an SVM）

一、概述

在之前的内容中，我们主要从理论层面理解了支持向量机（SVM）的思想与优化目标。
本节主要讨论如何**在实际中使用 SVM**，包括参数选择、核函数选取、与其他算法的对比等内容。

二、SVM 的实际使用建议

支持向量机的核心是一个优化问题。
但是，不建议自己编写求解 SVM 的优化代码。
原因在于：

- 求解 SVM 涉及复杂的数值优化算法；
- 高性能实现需要多年的优化研究；
- 现有的 SVM 库已经非常成熟。

常用的 SVM 软件包包括：

- **liblinear**：适合线性核（无核函数）SVM；
- **libsvm**：支持多种核函数（高斯核、多项式核等）。

这些库在多种主流编程语言中都有接口，可直接调用。

三、常见核函数类型

除了常用的高斯核（Gaussian Kernel）外，SVM 还支持多种核函数：

- **多项式核函数（Polynomial Kernel）**
- **字符串核函数（String Kernel）**
- **卡方核函数（Chi-square Kernel）**
- **直方图交集核函数（Histogram Intersection Kernel）**
- 以及其他根据特定应用定制的核函数

这些核函数的共同目标是：
根据样本与地标之间的相似度，生成新的特征映射。

使用的前提是核函数需满足 **Mercer 定理**，以确保优化问题可解。

四、多类分类（Multi-class Classification）

支持向量机原生为**二分类算法**，
若要用于多类问题，可以使用**一对多（One-vs-All）**方法：

- 对于 (k) 个类别，训练 (k) 个 SVM 模型；
- 每个模型学习“该类 vs 其他类”的分类边界。

在实践中，绝大多数 SVM 库（如 **libsvm**）已经内置了多类分类功能，
用户无需手动构建多个模型。

五、使用 SVM 前需要做的准备

虽然无需自己实现 SVM 优化器，但仍需完成以下工作：

1. **选择正则化参数 (C)**
   - 控制模型的偏差与方差；
   - 大 (C)：低偏差，高方差；
   - 小 (C)：高偏差，低方差。
2. **选择核函数及其参数（如高斯核中的 (σ)）**
   - (σ) 决定了核函数的“作用范围”；
   - 大 (σ)：平滑、偏差高；
   - 小 (σ)：复杂、方差高。
3. **特征缩放（Feature Scaling）**
   - 使用高斯核函数前必须对特征进行缩放；
   - 保证每个维度的数值尺度相似，避免距离计算失真。

六、SVM 与逻辑回归的选择

SVM 与逻辑回归在很多场景下表现相似，选择的主要依据是数据规模和特征维度。
设：

- (n)：特征数；
- (m)：样本数。

情况	建议使用算法	原因
(1) $(n \gg m)$（特征多、样本少）	逻辑回归 或 **线性 SVM**	数据量小，不宜用复杂非线性模型
(2) (n) 较小，(m) 中等（如 $(n \in [1, 1000])$，($m \in [10, 10000]$)）	**高斯核 SVM**	可捕捉非线性关系，计算可接受
(3) (n) 较小，(m) 很大（如 (m > 50000)）	逻辑回归 或 **线性 SVM**	非线性核计算代价过高

七、SVM、逻辑回归与神经网络的比较

1. **逻辑回归 vs 线性 SVM**
   - 两者结构非常相似；
   - 在相同数据集上，表现通常接近；
   - 实现方式或数值优化差异可能导致一方更快。
2. **非线性 SVM**
   - 使用核函数可拟合复杂的非线性边界；
   - 但计算复杂度高；
   - 当样本量过大时，训练速度会显著变慢。
3. **神经网络**
   - 理论上能在各种情况下表现良好；
   - 但训练较慢；
   - SVM 的优势在于优化问题是**凸优化问题**，不存在局部最优陷阱；
   - 优秀的 SVM 软件能保证找到**全局最优解**。

八、算法选择建议

- 当训练集规模较小、特征数中等时：
  使用高斯核 SVM。
- 当数据量很大或特征很多时：
  使用逻辑回归或线性核 SVM。
- 当问题复杂且需要自动提取高阶特征时：
  神经网络是可行选择，但训练时间较长。