

# 12

Kernel Trick

## 核技巧

将数据映射到高维特征空间



复杂理论，花拳绣腿；简单算法，立竿见影。

***Complex theories do not work; simple algorithms do.***

—— 弗拉基米尔·万普尼克 (Vladimir Vapnik) | 俄罗斯统计学家、数学家 | 1936 ~



- ◀ `sklearn.datasets.make_circles` 生成环形数据
- ◀ `sklearn.datasets.make_moons` 生成月牙形数据
- ◀ `sklearn.svm.SVC` 支持向量机函数

## 12.1 映射函数：实现升维

上一章简要介绍支持向量机核技巧 (kernel trick) 原理——将样本数据映射到高维特征空间中，使数据在高维空间中线性可分。常用的核函数有线性核函数、多项式核函数和径向基函数核函数等。这种核技巧可以有效地提高 SVM 的性能和灵活性，因为它可以将低维输入空间中的非线性分类问题转化为高维空间中的线性分类问题。通过使用核技巧，SVM 能够处理非线性问题并且具有更好的泛化能力。

核技巧应用广泛，本章依托支持向量机，展开讲解核技巧及常用的几种核函数。



请大家回顾《数据有道》介绍的高斯过程中的核函数和先验协方差的关系。

### 映射函数

首先，大家需要了解映射函数 (mapping function) 这个概念。 $x$  经过特征映射 (feature map) 后得到  $\phi(x)$  向量， $\phi()$  叫映射函数。如图 1 所示，从  $x$  到  $\phi(x)$  的过程便是一个升维过程。在原始数据特征空间线性不可分的数据，在新特征空间中变得线性可分。

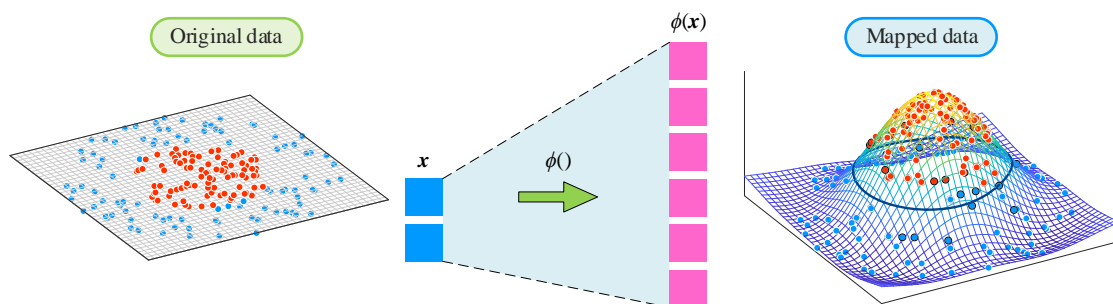


图 1. 映射原理示意图

从书前文一再提及降维 (dimension reduction)，核技巧则采用升维解决分类问题；这一点，听着有点不可思议。下面举几个例子解释。

### 第一个例子

图 2 所示为两组单一特征数据。图 2 (a) 中，原始数据左右两侧各 4 个点标签为 ●；中间 9 个点标签为 ●。

在  $x_1$  这个单一维度上，样本数据不能直接线性分类；但是，经过类似二次函数映射后，在全新二维空间中，样本数据便很容易被分类。

$$x = [x_1] \xrightarrow{\phi(\cdot)} \phi(x) = [x_1 \quad x_1^2]^T \quad (1)$$

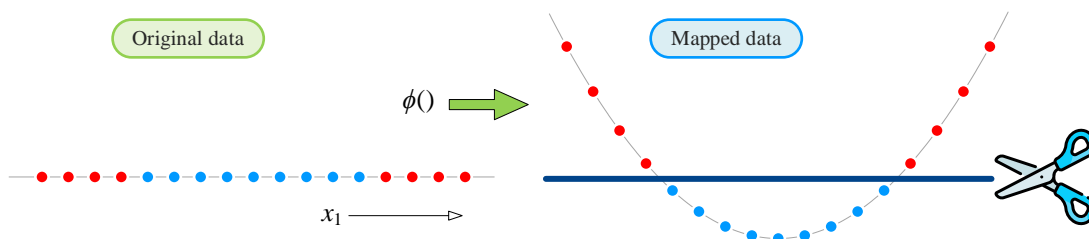


图 2. 核技巧，原始数据为单一特征

## 第二个例子

图 2 (a) 中原始数据标签交替出现。采用类似正弦函数映射到二维空间后，数据线性可分。

$$\mathbf{x} = [x_1] \xrightarrow{\phi(\cdot)} \phi(\mathbf{x}) = [x_1 \quad \sin(x_1)]^T \quad (2)$$

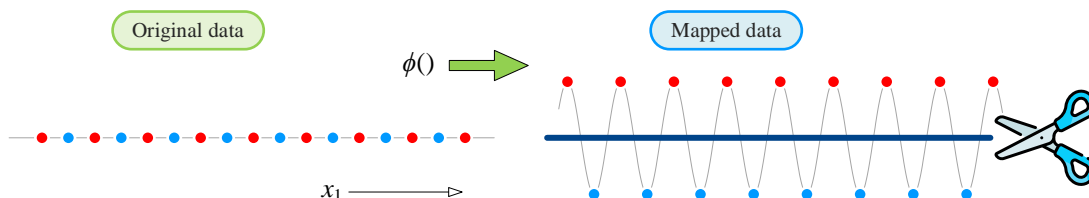


图 3. 核技巧，原始数据为单一特征

## 第三个例子

图 4 所示原始数据有两个特征，也是线性不可分。但是利用 XOR 函数映射之后，数据便容易分离。

$$\mathbf{x} = [x_1 \quad x_2]^T \xrightarrow{\phi(\cdot)} \phi(\mathbf{x}) = [x_1 \quad x_2 \quad \text{XOR}(x_1, x_2)]^T \quad (3)$$

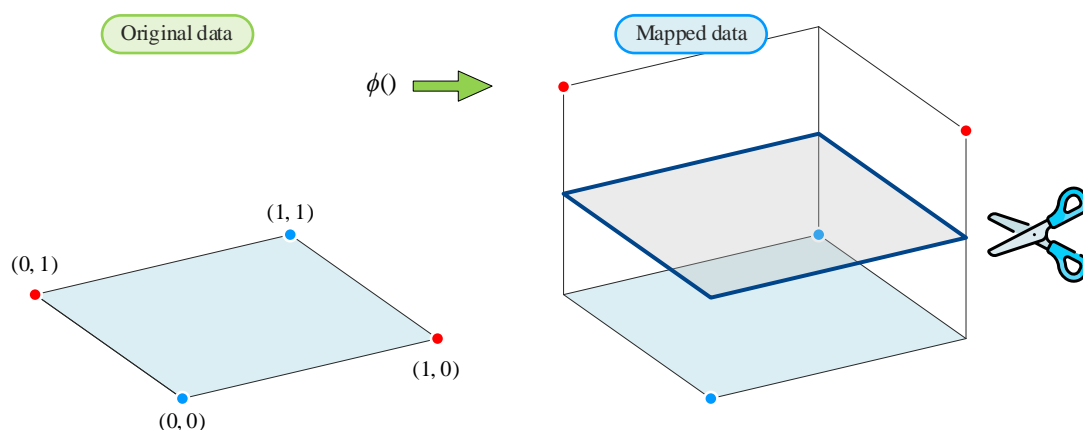


图 4. 核技巧，两特征数据

XOR 为**逻辑异或** (exclusive or) 函数，真值表如图 5。根据真值表， $\text{XOR}(0, 1) = 1$ ， $\text{XOR}(1, 0) = 1$ ；而， $\text{XOR}(1, 1) = 0$ ， $\text{XOR}(0, 0) = 0$ 。

本 PDF 文件为作者草稿，发布目的为方便读者在移动终端学习，终稿内容以清华大学出版社纸质出版物为准。

版权归清华大学出版社所有，请勿商用，引用请注明出处。

代码及 PDF 文件下载：<https://github.com/Visualize-ML>

本书配套微视频均发布在 B 站——生姜 DrGinger: <https://space.bilibili.com/513194466>

欢迎大家批评指教，本书专属邮箱：[jiang.visualize.ml@gmail.com](mailto:jiang.visualize.ml@gmail.com)

		A	
B	XOR(A, B)	False, 0	Truth, 1
	False, 0	False, 0	Truth, 1
	Truth, 1	Truth, 1	False, 0

图 5. XOR 逻辑异或真值表

## 第四个例子

上一章已经展示过类似图 6 环形数据，这种数据线性不可分。但是按照如下规则映射，在新的特征空间中，数据变得线性可分。

$$\mathbf{x} = [x_1 \ x_2]^T \xrightarrow{\phi(\cdot)} \phi(\mathbf{x}) = [x_1^2 \ \sqrt{2}x_1x_2 \ x_2^2]^T \quad (4)$$

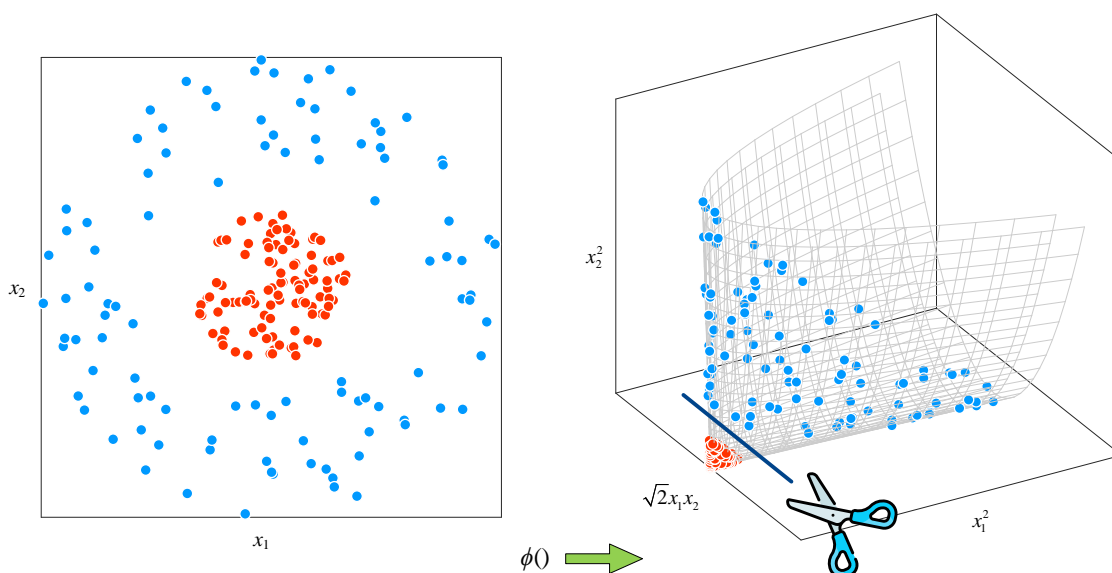


图 6. 环形数据映射到新特征空间，变得线性可分

## 改造支持向量机算法

通过上一章学习，我们知道 SVM 决策边界解析式为：

$$f(\mathbf{x}) = \mathbf{w}^T \mathbf{x} + b = 0 \quad (5)$$

(5) 解析式代表超平面。

经过特征映射后的，决策边界如下：

$$f(\mathbf{x}) = \mathbf{w}^T \phi(\mathbf{x}) + b = 0 \quad (6)$$

Mapping  
function

其中， $\mathbf{w}$  和  $b$  为模型参数。决策边界具体为哪一类曲面，取决于  $\phi(\mathbf{x})$ 。注意，(5) 和 (6) 中向量  $\mathbf{w}$  不同。(6) 中，列向量  $\mathbf{w}$  和  $\phi(\mathbf{x})$  行数一致。

一般情况， $\phi(\mathbf{x})$  的特征数远多于原始数据  $\mathbf{x}$ 。因此，(6) 中，列向量  $\mathbf{w}$  行数一般比  $\mathbf{x}$  特征数多。极端情况， $\phi(\mathbf{x})$  的特征数量可能为无穷。

需要大家格外注意的是， $\phi(\mathbf{x})$  形式是并不重要！我们关心的是  $\phi(\mathbf{x})$  和自身内积结果。

有了映射函数的铺垫，下一节构造核技巧支持向量机优化问题。

## 12.2 核技巧 SVM 优化问题

在支持向量机中，核技巧将输入数据映射到高维空间中，使得原本的非线性问题转化为线性问题。这个转化的过程中，涉及到一个最优化问题，即要找到一个最优的决策函数，使得分类边界最优化。因此，通过核技巧转化后的 SVM 问题就是求解一个最优化问题。优化问题可以通过求解拉格朗日函数的最小值来解决。

### 优化问题

类似上一章硬间隔 SVM，构建核技巧 SVM 优化问题如下：

$$\begin{aligned} \arg \min_{\mathbf{w}, b} \quad & \frac{\mathbf{w} \cdot \mathbf{w}}{2} \\ \text{subject to } & y^{(i)} \left( \underbrace{\mathbf{w} \cdot \phi(\mathbf{x}^{(i)})}_{\text{Mapping function}} + b \right) \geq 1, \quad i = 1, 2, 3, \dots, n \end{aligned} \quad (7)$$

### 拉格朗日函数

同样，构造拉格朗日函数  $L(\mathbf{w}, b, \lambda)$ ：

$$L(\mathbf{w}, b, \lambda) = \frac{\mathbf{w} \cdot \mathbf{w}}{2} + \sum_{i=1}^n \lambda_i \left( 1 - y^{(i)} (\mathbf{w} \cdot \phi(\mathbf{x}^{(i)}) + b) \right) \quad (8)$$

### 偏导

$L(\mathbf{w}, b, \lambda)$  对  $\mathbf{w}$  和  $b$  偏导为 0，得到：

$$\begin{cases} \frac{\partial L(\mathbf{w}, b, \lambda)}{\partial \mathbf{w}} = \mathbf{w} - \sum_{i=1}^n \lambda_i y^{(i)} \phi(\mathbf{x}^{(i)}) = \mathbf{0} \\ \frac{\partial L(\mathbf{w}, b, \lambda)}{\partial b} = \sum_{i=1}^n \lambda_i y^{(i)} = 0 \end{cases} \quad (9)$$

整理 (9) 得到：

$$\begin{cases} \mathbf{w} = \sum_{i=1}^n \lambda_i y^{(i)} \phi(\mathbf{x}^{(i)}) \\ \sum_{i=1}^n \lambda_i y^{(i)} = 0 \end{cases} \quad (10)$$

类似上一章推导过程，将 (10) 两式带入 (8)，消去 (8) 中  $\mathbf{w}$  和  $b$ ：

$$\begin{aligned} L(\mathbf{w}, b, \lambda) &= \frac{\mathbf{w} \cdot \mathbf{w}}{2} + \sum_{i=1}^n \lambda_i \left( 1 - y^{(i)} (\mathbf{w} \cdot \phi(\mathbf{x}^{(i)}) + b) \right) \\ &= \sum_{i=1}^n \lambda_i - \frac{\sum_{i=1}^n \sum_{j=1}^n \lambda_i \lambda_j y^{(i)} y^{(j)} \overbrace{\phi(\mathbf{x}^{(i)}) \cdot \phi(\mathbf{x}^{(j)})}^{\text{Kernel function}}}{2} \end{aligned} \quad (11)$$

## 核函数

(11) 中  $\phi(\mathbf{x}^{(i)}) \cdot \phi(\mathbf{x}^{(j)})$  中一项，可以记做：

$$\kappa(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}) = \phi(\mathbf{x}^{(i)}) \cdot \phi(\mathbf{x}^{(j)}) = \langle \phi(\mathbf{x}^{(i)}), \phi(\mathbf{x}^{(j)}) \rangle \quad (12)$$

其中， $\kappa(\mathbf{x}^{(i)}, \mathbf{x}^{(j)})$  被称作**核函数** (kernel function)。白话说，核函数按照某种规则完成“向量  $\rightarrow$  标量”运算。

根据内积运算原理，下式成立：

$$\kappa(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}) = \kappa(\mathbf{x}^{(j)}, \mathbf{x}^{(i)}) \quad (13)$$

## 简化拉格朗日函数

利用核函数记法，(11) 可以整理为  $L(\lambda)$ ：

$$L(\lambda) = \sum_{i=1}^n \lambda_i - \frac{\sum_{i=1}^n \sum_{j=1}^n \lambda_i \lambda_j y^{(i)} y^{(j)} \overbrace{\kappa(\mathbf{x}^{(i)}, \mathbf{x}^{(j)})}^{\text{Kernel}}}{2} \quad (14)$$

## 二次型、格拉姆矩阵

特别地，当  $n = 2$ ，即两个样本数据，(14) 可以展开为：

$$L(\lambda) = (\lambda_1 + \lambda_2) - \frac{1}{2} \left( \lambda_1 \lambda_1 y^{(1)} y^{(1)} \kappa(\mathbf{x}^{(1)}, \mathbf{x}^{(1)}) + 2 \lambda_1 \lambda_2 y^{(1)} y^{(2)} \kappa(\mathbf{x}^{(1)}, \mathbf{x}^{(2)}) + \lambda_2 \lambda_2 y^{(2)} y^{(2)} \kappa(\mathbf{x}^{(2)}, \mathbf{x}^{(2)}) \right) \quad (15)$$

(15) 整理为如下二次型：

$$L(\lambda) = (\lambda_1 + \lambda_2) - \frac{1}{2} \begin{bmatrix} \lambda_1 y^{(1)} & \lambda_2 y^{(2)} \end{bmatrix} \begin{bmatrix} \kappa(\mathbf{x}^{(1)}, \mathbf{x}^{(1)}) & \kappa(\mathbf{x}^{(1)}, \mathbf{x}^{(2)}) \\ \kappa(\mathbf{x}^{(2)}, \mathbf{x}^{(1)}) & \kappa(\mathbf{x}^{(2)}, \mathbf{x}^{(2)}) \end{bmatrix} \begin{bmatrix} \lambda_1 y^{(1)} \\ \lambda_2 y^{(2)} \end{bmatrix} \quad (16)$$

类似地，(14) 可以整理为：

本 PDF 文件为作者草稿，发布目的为方便读者在移动终端学习，终稿内容以清华大学出版社纸质出版物为准。

版权归清华大学出版社所有，请勿商用，引用请注明出处。

代码及 PDF 文件下载：<https://github.com/Visualize-ML>

本书配套微视频均发布在 B 站——生姜 DrGinger：<https://space.bilibili.com/513194466>

欢迎大家批评指教，本书专属邮箱：[jiang.visualize.ml@gmail.com](mailto:jiang.visualize.ml@gmail.com)

$$L(\lambda) = \sum_{i=1}^n \lambda_i - \frac{1}{2} \begin{bmatrix} \lambda_1 y^{(1)} \\ \lambda_2 y^{(2)} \\ \vdots \\ \lambda_n y^{(n)} \end{bmatrix}^T \underbrace{\begin{bmatrix} \kappa(\mathbf{x}^{(1)}, \mathbf{x}^{(1)}) & \kappa(\mathbf{x}^{(1)}, \mathbf{x}^{(2)}) & \cdots & \kappa(\mathbf{x}^{(1)}, \mathbf{x}^{(n)}) \\ \kappa(\mathbf{x}^{(2)}, \mathbf{x}^{(1)}) & \kappa(\mathbf{x}^{(2)}, \mathbf{x}^{(2)}) & \cdots & \kappa(\mathbf{x}^{(2)}, \mathbf{x}^{(n)}) \\ \vdots & \vdots & \ddots & \vdots \\ \kappa(\mathbf{x}^{(n)}, \mathbf{x}^{(1)}) & \kappa(\mathbf{x}^{(n)}, \mathbf{x}^{(2)}) & \cdots & \kappa(\mathbf{x}^{(n)}, \mathbf{x}^{(n)}) \end{bmatrix}}_{\text{Gram matrix}} \begin{bmatrix} \lambda_1 y^{(1)} \\ \lambda_2 y^{(2)} \\ \vdots \\ \lambda_n y^{(n)} \end{bmatrix} \quad (17)$$

令**格拉姆矩阵** (Gram matrix)  $\mathbf{K}$  为:

$$\mathbf{K} = \underbrace{\begin{bmatrix} \kappa(\mathbf{x}^{(1)}, \mathbf{x}^{(1)}) & \kappa(\mathbf{x}^{(1)}, \mathbf{x}^{(2)}) & \cdots & \kappa(\mathbf{x}^{(1)}, \mathbf{x}^{(n)}) \\ \kappa(\mathbf{x}^{(2)}, \mathbf{x}^{(1)}) & \kappa(\mathbf{x}^{(2)}, \mathbf{x}^{(2)}) & \cdots & \kappa(\mathbf{x}^{(2)}, \mathbf{x}^{(n)}) \\ \vdots & \vdots & \ddots & \vdots \\ \kappa(\mathbf{x}^{(n)}, \mathbf{x}^{(1)}) & \kappa(\mathbf{x}^{(n)}, \mathbf{x}^{(2)}) & \cdots & \kappa(\mathbf{x}^{(n)}, \mathbf{x}^{(n)}) \end{bmatrix}}_{\text{Gram matrix}} \quad (18)$$

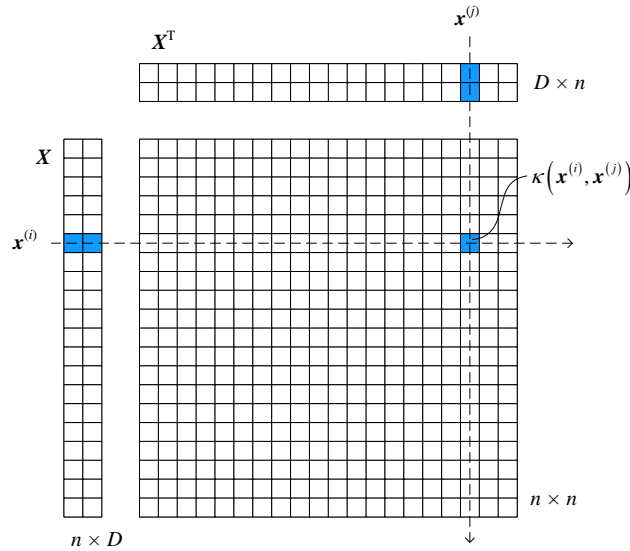


图 7. 格拉姆矩阵, 非线性核

(17) 可以整理为:

$$L(\lambda) = \sum_{i=1}^n \lambda_i - \frac{1}{2} \begin{bmatrix} \lambda_1 y^{(1)} \\ \lambda_2 y^{(2)} \\ \vdots \\ \lambda_n y^{(n)} \end{bmatrix}^T \mathbf{K} \begin{bmatrix} \lambda_1 y^{(1)} \\ \lambda_2 y^{(2)} \\ \vdots \\ \lambda_n y^{(n)} \end{bmatrix} \quad (19)$$

### 线性核: 最简单的核函数

下式上一章获得  $L(\lambda)$  函数:

$$L(\lambda) = \sum_{i=1}^n \lambda_i - \frac{\sum_{j=1}^n \sum_{i=1}^n \lambda_i \lambda_j y^{(i)} y^{(j)} \overbrace{(\mathbf{x}^{(i)} \cdot \mathbf{x}^{(j)})}^{\text{Linear kernel}}}{2} \quad (20)$$

对比 (14) 和 (20)，可以发现  $(\mathbf{x}^{(i)} \cdot \mathbf{x}^{(j)})$  实际上也是一种核函数——**线性核** (linear kernel) ——最简单的核函数。

## 对偶问题

至此，我们得到核技巧 SVM 优化问题的对偶问题：

$$\begin{aligned} \arg \max_{\lambda} \quad & \sum_{i=1}^n \lambda_i - \frac{\sum_{j=1}^n \sum_{i=1}^n \lambda_i \lambda_j y^{(i)} y^{(j)} \overbrace{\phi(\mathbf{x}^{(i)}) \cdot \phi(\mathbf{x}^{(j)})}^{\text{Kernel}}}{2} \\ \text{subject to} \quad & \begin{cases} \sum_{i=1}^n \lambda_i y^{(i)} = 0 \\ \lambda_i \geq 0, \quad i, j = 1, 2, 3, \dots, n \end{cases} \end{aligned} \quad (21)$$

## 决策边界

整理得到核技巧 SVM 决策边界如下：

$$\begin{aligned} f(\mathbf{x}) &= \mathbf{w} \cdot \phi(\mathbf{x}) + b \\ &= \left( \underbrace{\sum_{i=1}^n \lambda_i y^{(i)} \phi(\mathbf{x}^{(i)})}_{\text{Coefficients}} \right) \cdot \phi(\mathbf{x}) + b = 0 \end{aligned} \quad (22)$$

将  $\phi(\mathbf{x})$  乘到求和符号  $\Sigma$  里，得到决策边界解析解为：

$$f(\mathbf{x}) = \sum_{i=1}^n \left( \lambda_i y^{(i)} \underbrace{\kappa(\mathbf{x}^{(i)}, \mathbf{x})}_{\text{Kernel}} \right) + b = 0 \quad (23)$$

再次强调， $\mathbf{x}^{(i)}$  代表一个已知样本点，而  $\mathbf{x}$  代表未知量。注意，以上推导过程不再区分行、列向量。

使用核技巧，二分类决策函数  $p(\mathbf{x})$  则可以表达为：

$$\begin{aligned} p(\mathbf{x}) &= \text{sign}(\mathbf{w} \cdot \phi(\mathbf{x}) + b) \\ &= \text{sign} \left( \sum_{i=1}^n \lambda_i y^{(i)} \underbrace{\kappa(\mathbf{x}^{(i)}, \mathbf{x})}_{\text{Kernel}} + b \right) \end{aligned} \quad (24)$$

对比线性核 SVM 分类决策函数：



$$\begin{aligned}
 p(\mathbf{x}) &= \text{sign}(\mathbf{w} \cdot \mathbf{x} + b) \\
 &= \text{sign}\left(\sum_{i=1}^n \lambda_i y^{(i)} \underbrace{(\mathbf{x}^{(i)} \cdot \mathbf{x})}_{\text{Linear kernel}} + b\right)
 \end{aligned} \tag{25}$$

请读者注意，scikit-learn 中 **决策函数** (decision function) 输出值指的是 (23) 结果。

## 软间隔 + 核技巧

引入惩罚因子  $C$ ，可以构造软间隔核技巧 SVM 优化问题：

$$\begin{aligned}
 \arg \min_{\mathbf{w}, b} \quad & \frac{\mathbf{w} \cdot \mathbf{w}}{2} + C \sum_{i=1}^n \xi_i \\
 \text{subject to} \quad & \begin{cases} y^{(i)} (\mathbf{w} \cdot \phi(\mathbf{x}^{(i)}) + b) \geq 1 - \xi_i, & i = 1, 2, 3, \dots, n \\ \xi_i \geq 0 \end{cases}
 \end{aligned} \tag{26}$$

## 四种核函数

本章后面将逐一介绍四种核函数：

- ◀ **线性核** (linear kernel)
- ◀ **多项式核** (polynomial kernel)
- ◀ **高斯核** (Gaussian kernel)，也叫**径向基核 RBF** (radial basis function kernel)
- ◀ **sigmoid 核** (sigmoid kernel)

本章利用上述四种核函数求解图 8 所示三组数据——线性可分、月牙形和环形数据。请读者注意比较不同核函数优劣以及决策边界形状。

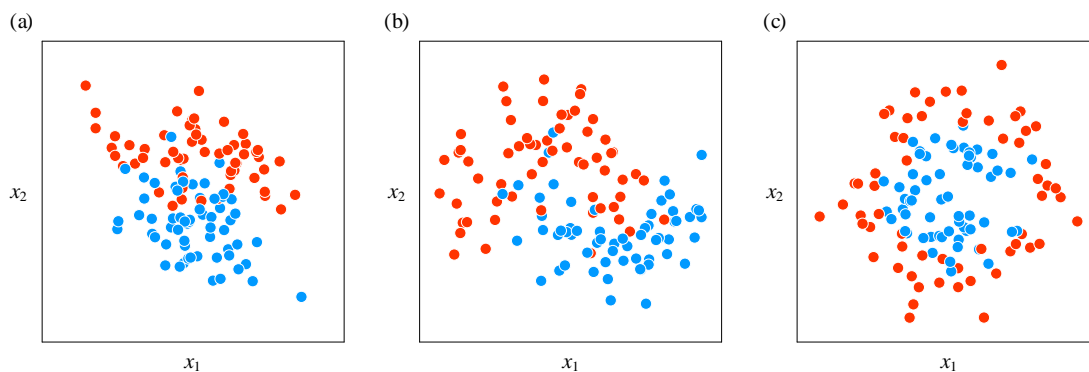


图 8. 三组数据——线性可分、月牙形和环形

## 12.3 线性核：最基本的核函数

**线性核** (linear kernel) 是支持向量机 SVM 的标配！线性核函数形式为：

$$\kappa(\mathbf{x}, \mathbf{q}) = \mathbf{x} \cdot \mathbf{q} = \langle \mathbf{x}, \mathbf{q} \rangle \quad (27)$$

线性核 SVM 决策边界形式如下：

$$f(\mathbf{x}) = \sum_{i=1}^n \left( \overbrace{\lambda_i y^{(i)} \mathbf{x}^{(i)}}^{\text{Coefficients Variables}} \underbrace{\mathbf{x}}_{\text{Hyperplane}} \right) + b = 0 \quad (28)$$

其中  $(\mathbf{x}^{(i)}, y^{(i)})$  为第  $i$  个样本数据点， $\lambda_i$  为求解得到拉格朗日乘子具体值。反复强调， $\mathbf{x}$  为变量构成的列向量。

### 超平面叠加

定义函数  $f_i(\mathbf{x})$ ：

$$f_i(\mathbf{x}) = \overbrace{\lambda_i y^{(i)} \mathbf{x}^{(i)}}^{\text{Coefficients}} \mathbf{x} \quad (29)$$

此外，观察 (29)，给定样本数据  $(\mathbf{x}^{(i)}, y^{(i)})$ ，优化得到的拉格朗日乘子  $\lambda_i$  相当于权重。相信通过丛书之前数学内容学习，大家已经清楚，(29) 所示  $f_i(\mathbf{x})$  空间形状为**超平面** (hyperplane)。

由此，(28) 可以记做：

$$f(\mathbf{x}) = \sum_{i=1}^n f_i(\mathbf{x}) + b = 0 \quad (30)$$

而 (30) 告诉我们，线性核 SVM 决策边界由  $n$  个超平面叠加而成，如图 9 所示。因此，线性核决策边界是超平面。

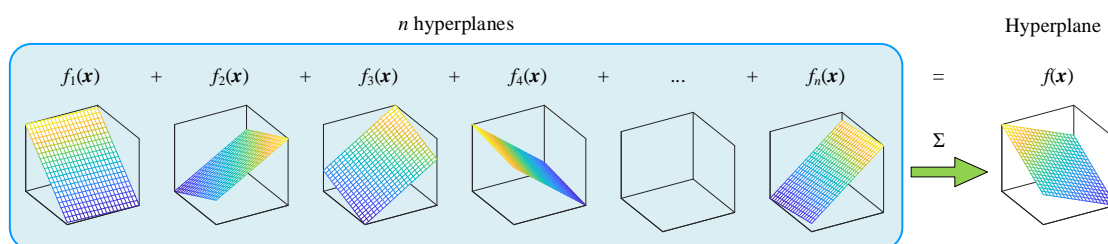


图 9. 线性核 SVM 决策超平面由  $n$  个超平面构造

### 决策边界

采用线性核预测图 8 所示三组形态不同数据分类。图 10、图 11 和图 12 分别为线性可分、月牙和环形数据预测分类结果。观察这三幅图，可以发现  $f(\mathbf{x})$  对应几何形状均为平面，决策边界  $f(\mathbf{x}) = 0$  为直线。

采用软间隔，线性核 SVM 尚可以分类图 10 所示样本数据。但是，对于图 11 和图 12 这种完全线性不可分数据，线性核 SVM 显得力不从心。

本 PDF 文件为作者草稿，发布目的为方便读者在移动终端学习，终稿内容以清华大学出版社纸质出版物为准。

版权归清华大学出版社所有，请勿商用，引用请注明出处。

代码及 PDF 文件下载：<https://github.com/Visualize-ML>

本书配套微课视频均发布在 B 站——生姜 DrGinger：<https://space.bilibili.com/513194466>

欢迎大家批评指教，本书专属邮箱：[jiang.visualize.ml@gmail.com](mailto:jiang.visualize.ml@gmail.com)

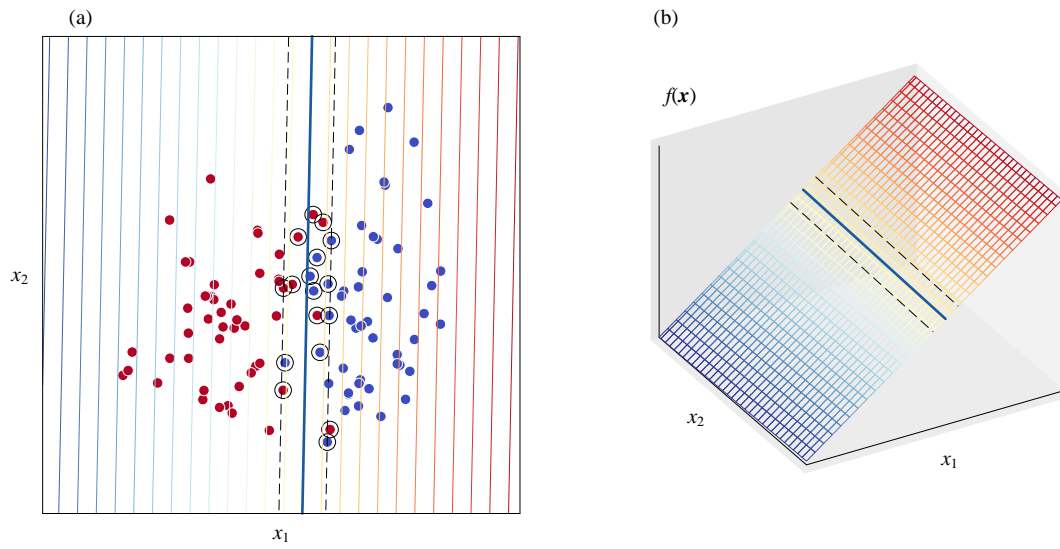


图 10. 线性可分数据，线性核 SVM

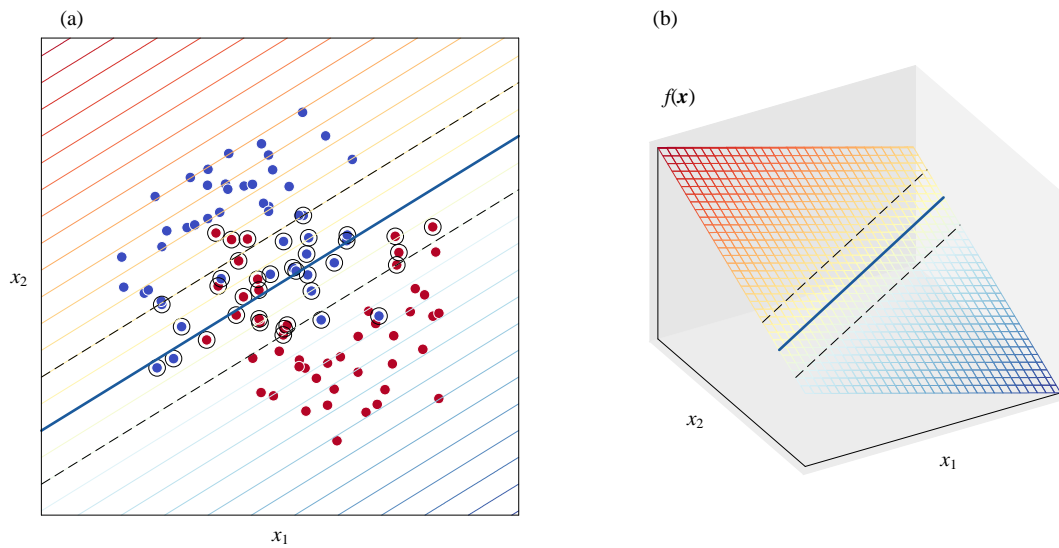


图 11. 月牙形数据，线性核 SVM

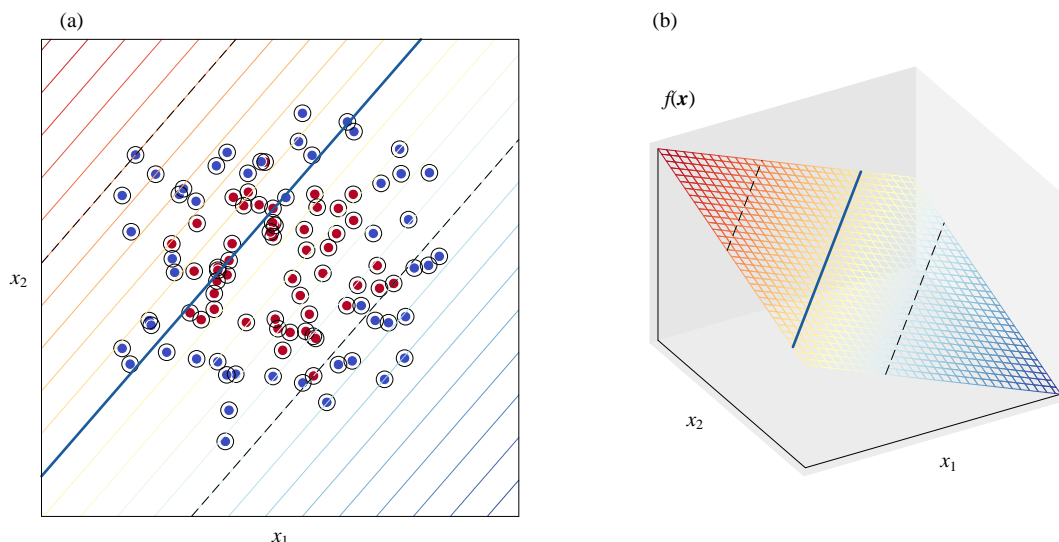


图 12. 环形数据，线性核 SVM

## 12.4 多项式核

**多项式核** (polynomial kernel) 形式如下：

$$\kappa_{\text{poly}(d)}(\mathbf{x}, \mathbf{q}) = (\gamma \mathbf{x} \cdot \mathbf{q} + r)^d = (\gamma \langle \mathbf{x}, \mathbf{q} \rangle + r)^d \quad (31)$$

其中， $\gamma$  (gamma) 为系数， $d$  为多项式次数， $r$  为常数。

(31) 中  $d$  主导多项式核形态，因此最为重要。线性核是多项式核特例，即次数  $d = 1$ 。 $d = 2$  时，(31) 为二次核； $d = 3$  时，(31) 为三次核。

### 以二次核为例

系数  $\gamma = 1$ ，常数  $r = 0$ ，次数  $d = 2$ ，特征数  $D = 2$  条件下，(31) 可以写作：

$$\kappa_{\text{poly}(2)}(\mathbf{x}, \mathbf{q}) = (\mathbf{x} \cdot \mathbf{q})^2 \quad (32)$$

其中，

$$\mathbf{x} = [x_1 \quad x_2]^T \quad \mathbf{q} = [q_1 \quad q_2]^T \quad (33)$$

将 (33) 代入 (32)，整理得到：

$$\begin{aligned} \kappa_{\text{poly}(2)}(\mathbf{x}, \mathbf{q}) &= (\mathbf{x} \cdot \mathbf{q})^2 = (x_1 q_1 + x_2 q_2)^2 \\ &= x_1^2 q_1^2 + x_2^2 q_2^2 + 2x_1 x_2 q_1 q_2 = x_1^2 \cdot q_1^2 + x_2^2 \cdot q_2^2 + \sqrt{2} x_1 x_2 \cdot \sqrt{2} q_1 q_2 \\ &= \phi(\mathbf{x}) \cdot \phi(\mathbf{q}) \end{aligned} \quad (34)$$

其中，

$$\begin{aligned}\phi(\mathbf{x}) &= \begin{bmatrix} x_1^2 & x_2^2 & \sqrt{2}x_1x_2 \end{bmatrix}^T \\ \phi(\mathbf{q}) &= \begin{bmatrix} q_1^2 & q_2^2 & \sqrt{2}q_1q_2 \end{bmatrix}^T\end{aligned}\quad (35)$$

(34) 还可以写作：

$$\begin{aligned}\kappa_{\text{poly}(2)}(\mathbf{x}, \mathbf{q}) &= x_1^2 \cdot q_1^2 + x_2^2 \cdot q_2^2 + x_1x_2 \cdot q_1q_2 + x_2x_1 \cdot q_2q_1 \\ &= \phi(\mathbf{x}) \cdot \phi(\mathbf{q})\end{aligned}\quad (36)$$

其中，

$$\begin{aligned}\phi(\mathbf{x}) &= \begin{bmatrix} x_1^2 & x_2^2 & x_1x_2 & x_2x_1 \end{bmatrix}^T \\ \phi(\mathbf{q}) &= \begin{bmatrix} q_1^2 & q_2^2 & q_1q_2 & q_2q_1 \end{bmatrix}^T\end{aligned}\quad (37)$$

比较 (35) 和 (37)，可以发现  $\phi()$  映射规则并不唯一。或者说， $\phi()$  的具体形式并不重要，我们关心的是映射规则和标量结果。 $\phi(\mathbf{x})$  搭建起从  $\mathbf{x}$  到核函数桥梁，如图 13 所示。

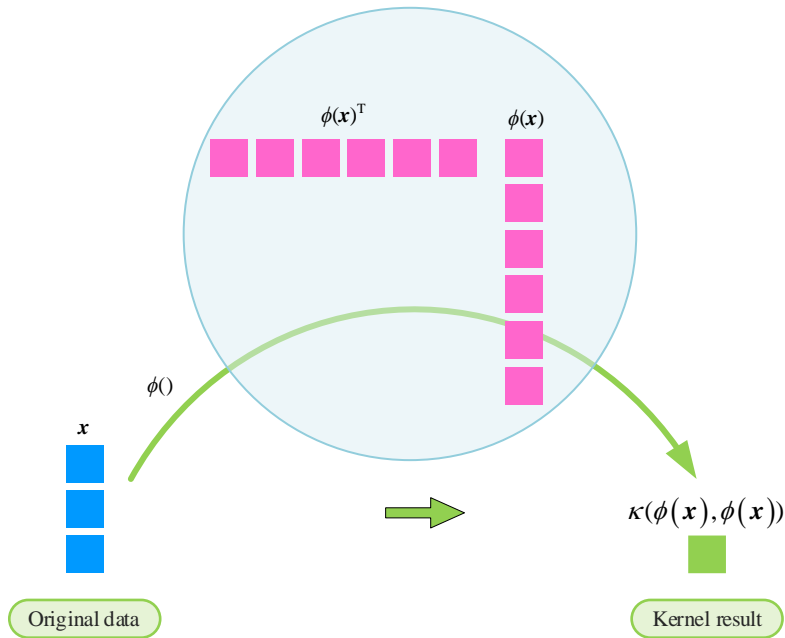


图 13. 映射原理

再看一个例子，系数  $\gamma = 1$ ，常数  $r = 1$ ，次数  $d = 2$ ，特征数  $D = 2$  条件下，(31) 可以写作：

$$\kappa_{\text{poly}(2)}(\mathbf{x}, \mathbf{q}) = (\mathbf{x} \cdot \mathbf{q} + 1)^2 \quad (38)$$

(38) 展开可以得到：

$$\begin{aligned}\kappa_{\text{poly}(2)}(\mathbf{x}, \mathbf{q}) &= (\mathbf{x} \cdot \mathbf{q} + 1)^2 \\ &= (x_1q_1 + x_2q_2 + 1)^2 \\ &= 1 + 2x_1q_1 + 2x_2q_2 + x_1^2q_1^2 + x_2^2q_2^2 + 2x_1x_2q_1q_2 \\ &= 1 \cdot 1 + \sqrt{2}x_1 \cdot \sqrt{2}q_1 + \sqrt{2}x_2 \cdot \sqrt{2}q_2 + x_1^2 \cdot q_1^2 + x_2^2 \cdot q_2^2 + \sqrt{2}x_1x_2 \cdot \sqrt{2}q_1q_2 \\ &= \phi(\mathbf{x}) \cdot \phi(\mathbf{q})\end{aligned}\quad (39)$$

本 PDF 文件为作者草稿，发布目的为方便读者在移动终端学习，终稿内容以清华大学出版社纸质出版物为准。

版权归清华大学出版社所有，请勿商用，引用请注明出处。

代码及 PDF 文件下载：<https://github.com/Visualize-ML>

本书配套微课视频均发布在 B 站——生姜 DrGinger：<https://space.bilibili.com/513194466>

欢迎大家批评指教，本书专属邮箱：[jiang.visualize.ml@gmail.com](mailto:jiang.visualize.ml@gmail.com)

其中,

$$\begin{aligned}\phi(\mathbf{x}) &= [1 \quad \sqrt{2}x_1 \quad \sqrt{2}x_2 \quad x_1^2 \quad x_2^2 \quad \sqrt{2}x_1x_2]^T \\ \phi(\mathbf{q}) &= [1 \quad \sqrt{2}q_1 \quad \sqrt{2}q_2 \quad q_1^2 \quad q_2^2 \quad \sqrt{2}q_1q_2]^T\end{aligned}\quad (40)$$

多项式核 SVM 决策边界解析式为:

$$f(\mathbf{x}) = \sum_{i=1}^n \left( \lambda_i y^{(i)} (\gamma \mathbf{x}^{(i)} \cdot \mathbf{x} + r) \right) + b = 0 \quad (41)$$

类似 (29), 可以用  $n$  个  $f_i(\mathbf{x})$  函数构造决策边界解析式。

$$f(\mathbf{x}) = \sum_{i=1}^n (f_i(\mathbf{x})) + b = 0 \quad (42)$$

其中,

$$f_i(\mathbf{x}) = \lambda_i y^{(i)} (\gamma \mathbf{x}^{(i)} \cdot \mathbf{x} + r)^d \quad (43)$$

同理, (42) 可以解读为, 多项式核超曲面 (hypersurface) 相当有  $n$  个超曲面构造。图 14 所示为这一过程。下面, 我们用两节内容, 分别介绍如何用二次核和三次核分类图 8 所示三组形数据。

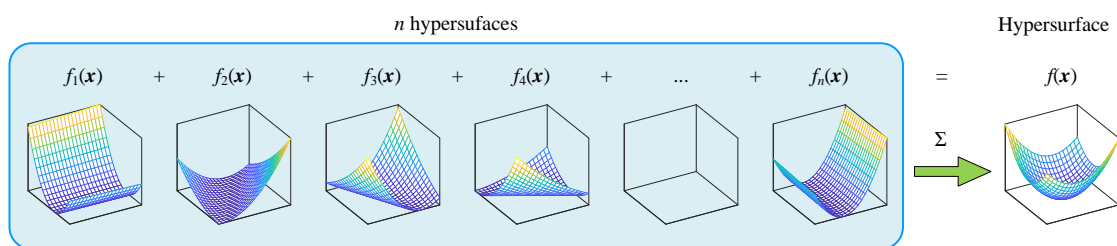


图 14. 多项式核超曲面相当有  $n$  个超曲面构造

## 12.5 二次核：二次曲面

二次核形式如下:

$$\kappa_{\text{poly}(d)}(\mathbf{x}, \mathbf{q}) = (\gamma \mathbf{x} \cdot \mathbf{q} + r)^2 = (\gamma \langle \mathbf{x}, \mathbf{q} \rangle + r)^2 \quad (44)$$

图 15、图 16 和图 17 所示为二次核 SVM 对线性可分、月牙形和环形数据预测分类结果。图 15 的决策面为山谷面; 图 16 为双曲抛物面; 图 17 看似椭圆抛物面。

请大家翻阅《数学要素》, 回顾常见的二次曲面, 并温习《矩阵力量》的正定性这个概念。

不难发现, 二次核适用范围比较窄, 分类效果一般。要想构造复杂曲面, 就需要提高多项式核次数。

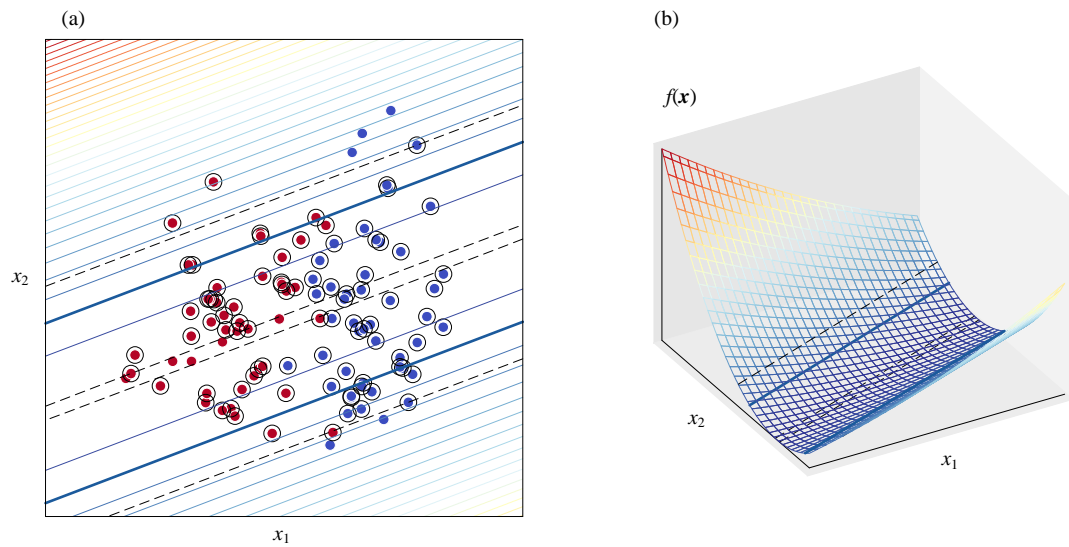


图 15. 线性可分数据，二次核 SVM

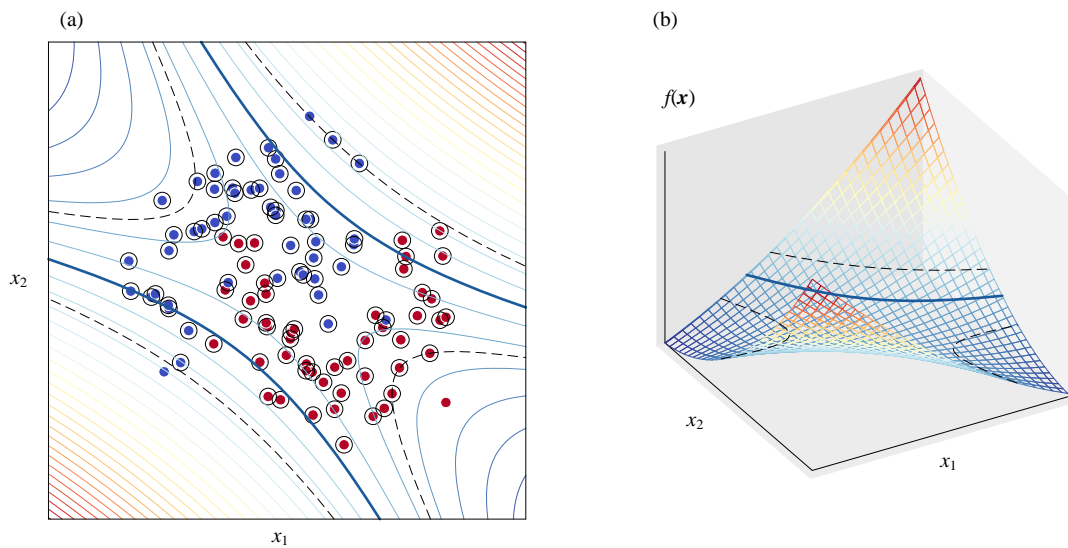


图 16. 月牙形数据，二次核 SVM

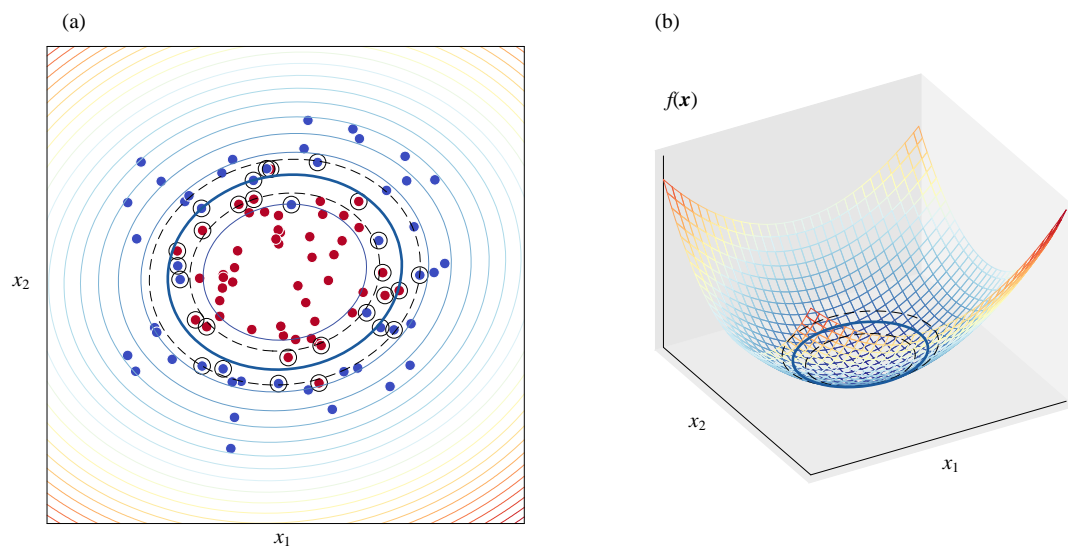


图 17. 环形数据，二次核 SVM

本 PDF 文件为作者草稿，发布目的为方便读者在移动终端学习，终稿内容以清华大学出版社纸质出版物为准。

版权归清华大学出版社所有，请勿商用，引用请注明出处。

代码及 PDF 文件下载：<https://github.com/Visualize-ML>

本书配套微课视频均发布在 B 站——生姜 DrGinger：<https://space.bilibili.com/513194466>

欢迎大家批评指教，本书专属邮箱：[jiang.visualize.ml@gmail.com](mailto:jiang.visualize.ml@gmail.com)



## 12.6 三次核：三次曲面

三次核形式如下：

$$\kappa_{\text{poly}(3)}(\mathbf{x}, \mathbf{q}) = (\gamma \mathbf{x} \cdot \mathbf{q} + r)^3 = (\gamma \langle \mathbf{x}, \mathbf{q} \rangle + r)^3 \quad (45)$$

图 18、图 19 和图 20 所示为三次核 SVM 分类线性可分、月牙形和环形数据结果。对比二次核 SVM 结果，可以发现三次核分类结果远好于二次核。本章最后给出代码，请读者尝试不断提高多项式核次数，并观察不同次数多项式核预测分类结果，仔细观察决策边界形状。

多项式核次数不是越高越好；次数过高会带来过拟合 (overfitting)，泛化能力低的问题。过拟合指的是机器学习模型在训练数据上表现良好，但在测试数据上表现不佳的现象。这是因为模型过度拟合训练数据，学习了数据的噪声和细节，而忽略了数据的潜在模式。过拟合可能会导致模型的泛化性能下降，因此需要采用一些技术来避免过拟合。

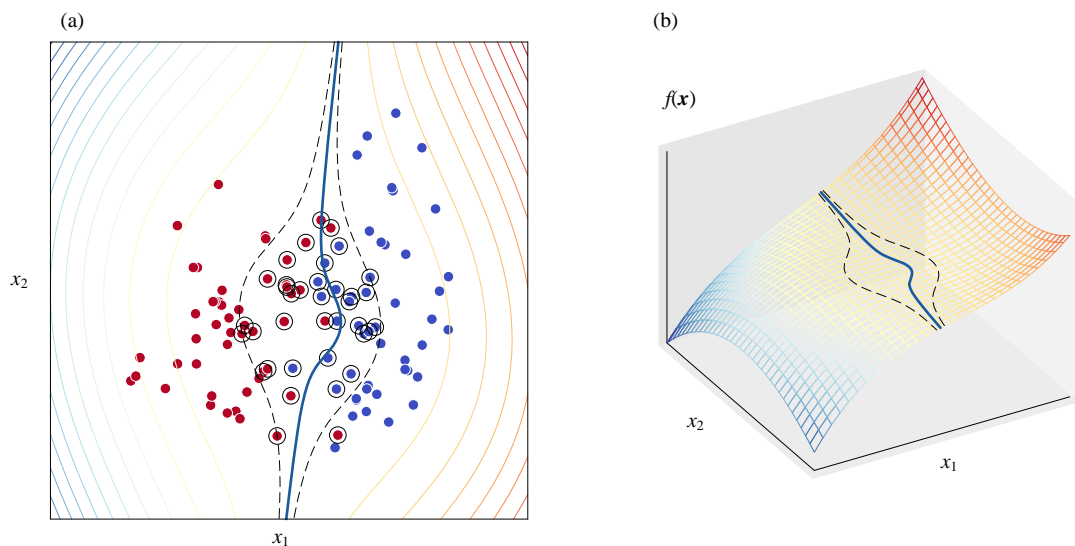
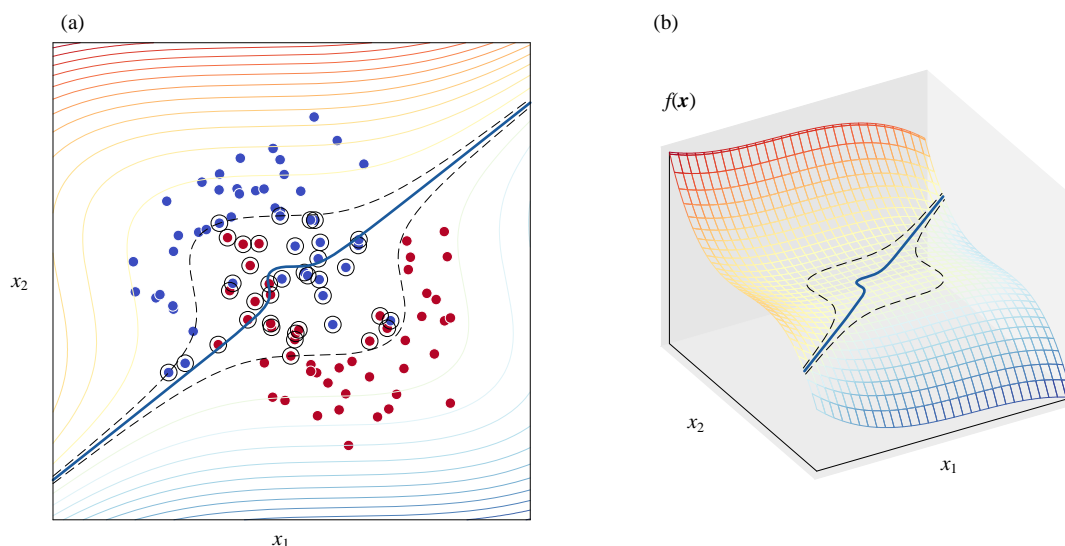


图 18. 线性可分数据，三次核 SVM



本 PDF 文件为作者草稿，发布目的为方便读者在移动终端学习，终稿内容以清华大学出版社纸质出版物为准。

版权归清华大学出版社所有，请勿商用，引用请注明出处。

代码及 PDF 文件下载：<https://github.com/Visualize-ML>

本书配套微视频均发布在 B 站——生姜 DrGinger：<https://space.bilibili.com/513194466>

欢迎大家批评指教，本书专属邮箱：[jiang.visualize.ml@gmail.com](mailto:jiang.visualize.ml@gmail.com)



图 19. 月牙形数据，三次核 SVM

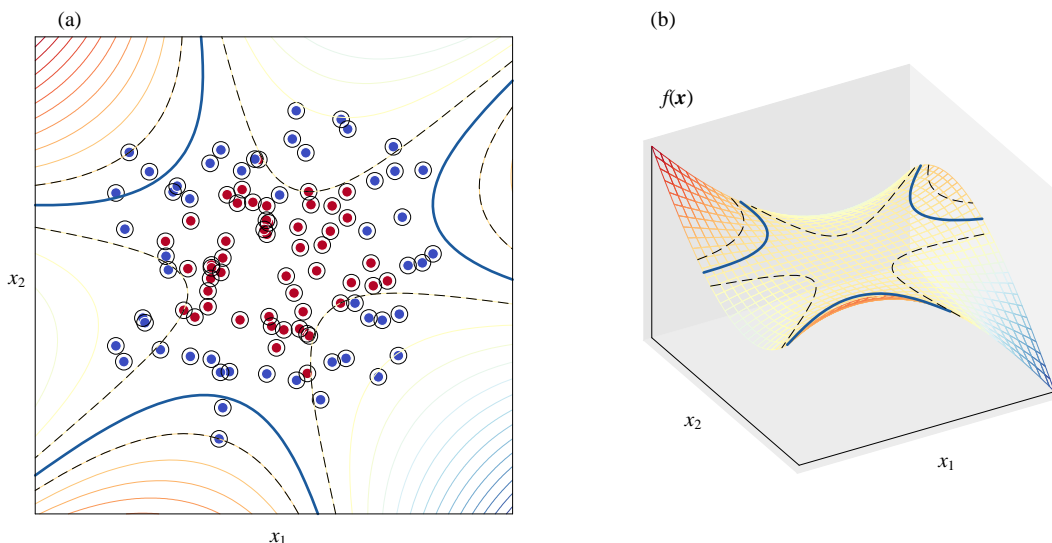


图 20. 环形数据，三次核 SVM

## 12.7 高斯核：基于径向基函数

对于刚接触支持向量机的读者，高斯核是谜一样的存在。它看上去那么简单，又那么神秘。本节就帮助大家揭开高斯核的面纱一角。

**高斯核** (Gaussian kernel)，也叫**径向基核函数** (radial basis function kernel, RBF kernel)，具体形式为：

$$\kappa_{\text{RBF}}(\mathbf{x}, \mathbf{q}) = \exp(-\gamma \|\mathbf{x} - \mathbf{q}\|^2) \quad (46)$$

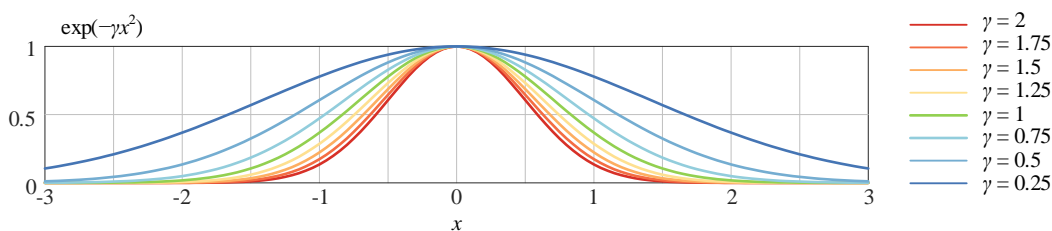
(46) 有一个参数， $\gamma$  ( $\gamma > 0$ )； $\gamma$  决定高斯核函数曲面的开口大小。

### 形状

对于单特征 ( $D = 1$ )，且  $q = 0$  时，(46) 可以写作：

$$\kappa_{\text{RBF}}(x, 0) = \exp(-\gamma x^2) \quad (47)$$

图 21 给出  $\gamma$  如何影响高斯核曲线。

图 21.  $\gamma$  决定高斯核函数曲面的开口大小

## 试着找到映射函数

$\gamma = 1/2$  时,

$$\begin{aligned}\kappa_{\text{RBF}}(\mathbf{x}, \mathbf{q}) &= \exp\left(-\frac{1}{2}\|\mathbf{x} - \mathbf{q}\|^2\right) \\ &= \exp\left(-\frac{1}{2}(\mathbf{x} - \mathbf{q}) \cdot (\mathbf{x} - \mathbf{q})\right) \\ &= \exp\left(-\frac{1}{2}(\|\mathbf{x}\|^2 + \|\mathbf{q}\|^2 - 2\mathbf{x} \cdot \mathbf{q})\right) \\ &= \exp\left(-\frac{1}{2}(\|\mathbf{x}\|^2 + \|\mathbf{q}\|^2)\right) \exp(\mathbf{x} \cdot \mathbf{q})\end{aligned}\quad (48)$$

对  $\exp(\mathbf{x} \cdot \mathbf{q})$  泰勒展开 (Taylor expansion):

$$\begin{aligned}\exp(\mathbf{x} \cdot \mathbf{q}) &= \sum_{j=0}^{\infty} \frac{(\mathbf{x} \cdot \mathbf{q})^j}{j!} \\ &= 1 \cdot 1 + \mathbf{x} \cdot \mathbf{q} + \frac{(\mathbf{x} \cdot \mathbf{q})^2}{2} + \dots\end{aligned}\quad (49)$$

可以发现  $\exp(\mathbf{x} \cdot \mathbf{q})$  展开得到无限项。

$D = 2$  时, 仅考虑 (49) 前三项,  $\exp(\mathbf{x} \cdot \mathbf{q})$  可以展开并整理为:

$$\exp(\mathbf{x} \cdot \mathbf{q}) \approx 1 \cdot 1 + (\mathbf{x}_1 \cdot \mathbf{q}_1 + \mathbf{x}_2 \cdot \mathbf{q}_2) + \frac{\mathbf{x}_1^2 \cdot \mathbf{q}_1^2 + \mathbf{x}_2^2 \cdot \mathbf{q}_2^2 + \sqrt{2}\mathbf{x}_1\mathbf{x}_2 \cdot \sqrt{2}\mathbf{q}_1\mathbf{q}_2}{2}\quad (50)$$

将 (50) 代入 (48), 整理得到:

$$\begin{aligned}\kappa_{\text{RBF}}(\mathbf{x}, \mathbf{q}) &\approx \exp\left(-\frac{1}{2}\|\mathbf{x}\|^2\right) \exp\left(-\frac{1}{2}\|\mathbf{q}\|^2\right) \left(1 \cdot 1 + (\mathbf{x}_1 \cdot \mathbf{q}_1 + \mathbf{x}_2 \cdot \mathbf{q}_2) + \frac{\mathbf{x}_1^2 \cdot \mathbf{q}_1^2 + \mathbf{x}_2^2 \cdot \mathbf{q}_2^2 + \sqrt{2}\mathbf{x}_1\mathbf{x}_2 \cdot \sqrt{2}\mathbf{q}_1\mathbf{q}_2}{2}\right) \\ &= \phi(\mathbf{x}) \cdot \phi(\mathbf{q})\end{aligned}\quad (51)$$

其中, 两个映射函数可以记做:

$$\begin{aligned}\phi(\mathbf{x}) &= \exp\left(-\frac{1}{2}\|\mathbf{x}\|^2\right) \begin{bmatrix} 1 & x_1 & x_2 & \frac{x_1^2}{\sqrt{2}} & \frac{x_2^2}{\sqrt{2}} & x_1x_2 \end{bmatrix} \\ \phi(\mathbf{q}) &= \exp\left(-\frac{1}{2}\|\mathbf{q}\|^2\right) \begin{bmatrix} 1 & q_1 & q_2 & \frac{q_1^2}{\sqrt{2}} & \frac{q_2^2}{\sqrt{2}} & q_1q_2 \end{bmatrix}\end{aligned}\quad (52)$$

如上文所述, 高斯核经过泰勒展开后, 实际上映射函数有无数项。本书后文还会用到核技巧。

## 试着找到映射函数

高斯核 SVM 对应的决策边界解析式:

$$f(\mathbf{x}) = \sum_{i=1}^n \left( \lambda_i y^{(i)} \exp\left(-\gamma \|\mathbf{x}^{(i)} - \mathbf{x}\|^2\right) \right) + b = 0\quad (53)$$

和前文一样, 用  $n$  个  $f_i(\mathbf{x})$  函数构造  $f(\mathbf{x})$ :

$$f(\mathbf{x}) = \sum_{i=1}^n (f_i(\mathbf{x})) + b = 0 \quad (54)$$

其中,

$$f_i(\mathbf{x}) = \lambda_i y^{(i)} \exp\left(-\gamma \|\mathbf{x}^{(i)} - \mathbf{x}\|^2\right) \quad (55)$$

对于两特征  $D=2$ , (55) 可以展开得到:

$$f_i(\mathbf{x}) = \lambda_i y^{(i)} \exp\left(-\gamma \left((x_1 - x_{i,1})^2 + (x_2 - x_{i,2})^2\right)\right) \quad (56)$$

其中, 仅  $x_1$  和  $x_2$  为变量。 $(x_{i,1}, x_{i,2})$  决定曲面中心位置,  $\gamma$  决定曲面“胖瘦”,  $\lambda_i y^{(i)}$  决定曲面高矮。

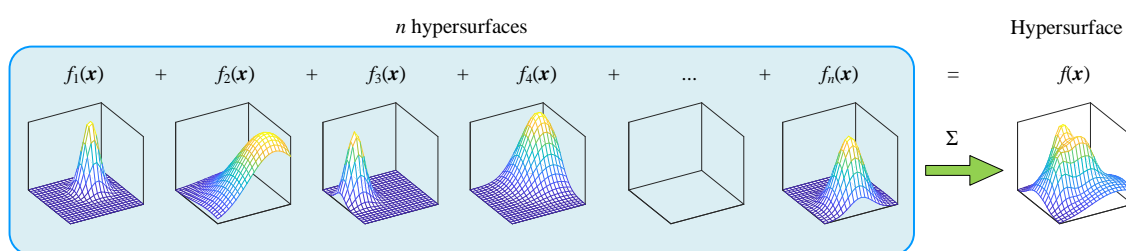


图 22. 高斯核 SVM 决策曲面相当有  $n$  个“高斯核曲面”构造

## 分类结果

图 23、图 24 和图 25 所示为高斯核 SVM 求解线性可分、月牙形和环形数据分类问题结果。可以发现高斯核 SVM 得到的决策边界形状丰富多样, 分类预测结果要好于前文介绍的二次核和三次核。

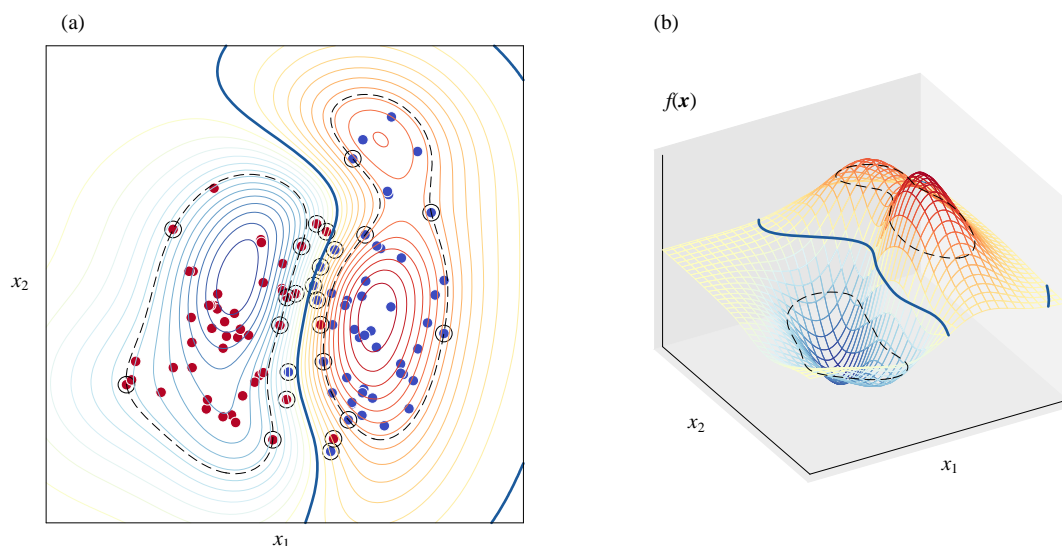


图 23. 线性可分数据, 高斯核 SVM

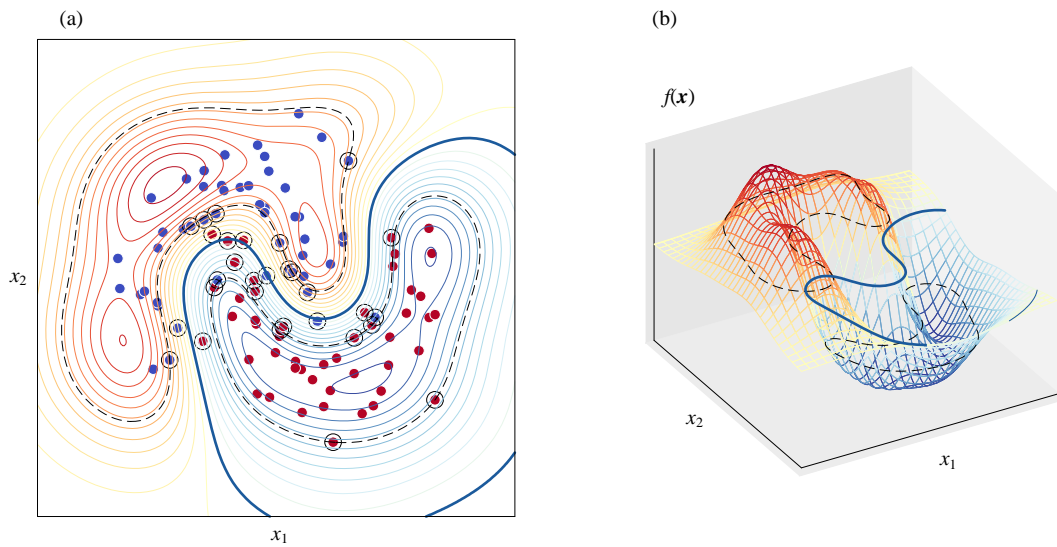


图 24. 月牙形数据，高斯核 SVM

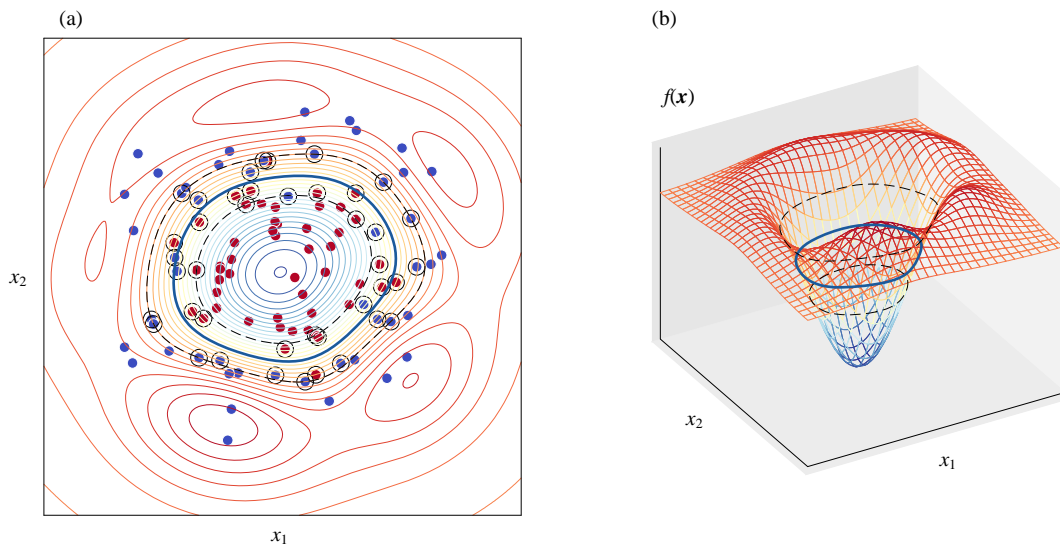


图 25. 环形数据，高斯核 SVM

图 26 所示为用 Streamlit 搭建的展示 Gamma 对高斯核 SVM 分类影响的 App。读过《编程不难》的读者对这个 App 应该很熟悉了。

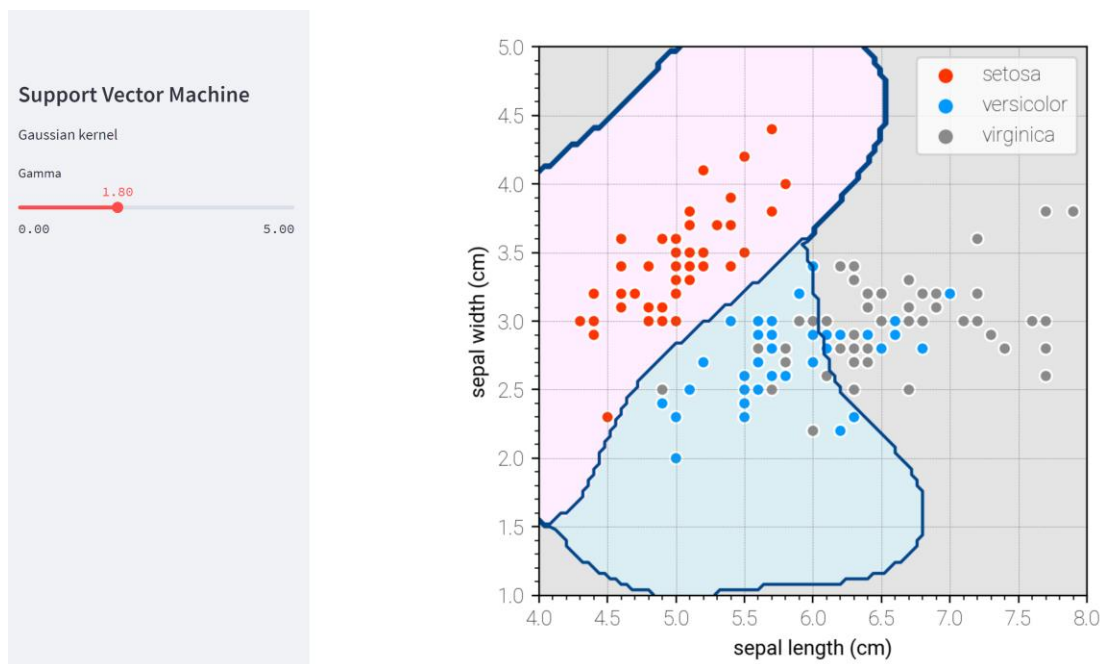


图 26. 展示 Gamma 对高斯核 SVM 分类影响的 App, Streamlit 搭建 | [Streamlit\\_Bk7\\_Ch12\\_02.py](#)

## 12.8 Sigmoid 核

**Sigmoid 核** (sigmoid kernel), 也叫 S 形核, 形式如下:

$$\kappa_{\text{Sigmoid}}(\mathbf{x}, \mathbf{q}) = \tanh(\gamma \mathbf{x} \cdot \mathbf{q} + r) \quad (57)$$

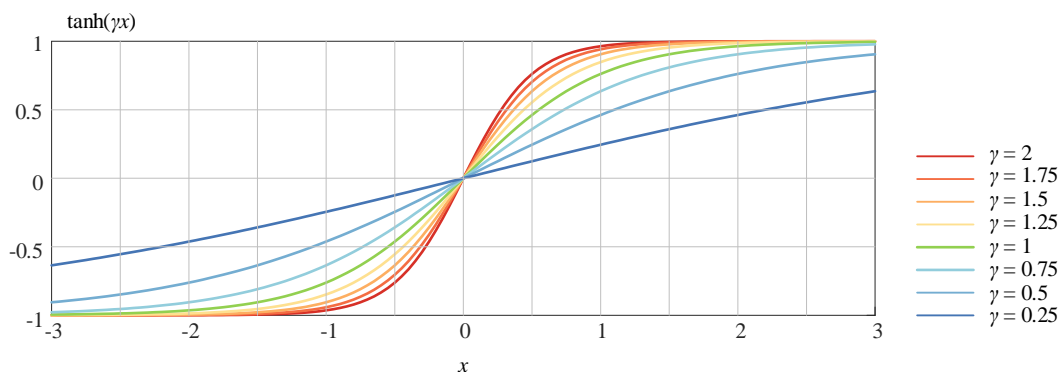
其中,  $\tanh()$  为**双曲正切函数** (hyperbolic tangent)。

### 形状

对于单特征 ( $D = 1$ ), 且  $q = 1$ 、 $r = 0$  时, (57) 可以写作:

$$\kappa_{\text{Sigmoid}}(x, 1) = \tanh(\gamma x) \quad (58)$$

丛书第一本数学部分介绍过双曲正切函数, 图 27 给出  $\gamma$  如何影响 sigmoid 核曲线。 $\tanh()$  是机器学习的常客, 比如在神经网络中  $\tanh()$  便是常用的激励函数之一。

图 27.  $\gamma$  影响双曲正切函数形状

## 决策边界

Sigmoid 核 SVM 决策边界解析式如下：

$$f(\mathbf{x}) = \sum_{i=1}^n \left( \lambda_i y^{(i)} \tanh(\gamma \mathbf{x}^{(i)} \cdot \mathbf{x} + r) \right) + b = 0 \quad (59)$$

和前文一样， $f(\mathbf{x})$  可以由  $n$  个  $f_i(\mathbf{x})$  构造：

$$f(\mathbf{x}) = \sum_{i=1}^n \left( f_i(\mathbf{x}) \right) + b = 0 \quad (60)$$

其中，

$$f_i(\mathbf{x}) = \lambda_i y^{(i)} \tanh(\gamma \mathbf{x}^{(i)} \cdot \mathbf{x} + r) \quad (61)$$

如图 28 所示，sigmoid 核曲面可以由若干 sigmoid 曲面构造得到。

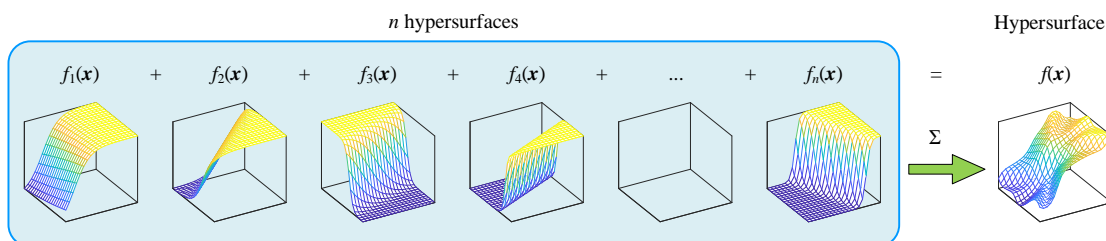


图 28. Sigmoid 核曲面可以由若干 sigmoid 曲面构造

## 分类结果

图 29、图 30 和图 31 所示为 sigmoid 核 SVM 解决三组样本数据结果。类似高斯核，sigmoid 核能够构造比较复杂的曲面形状。

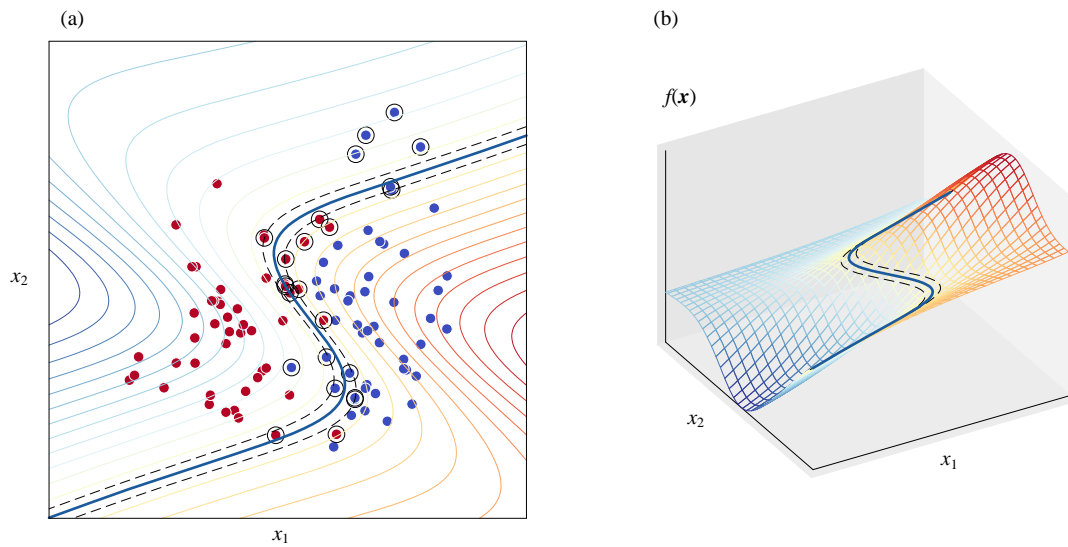


图 29. 线性可分数据, sigmoid 核 SVM

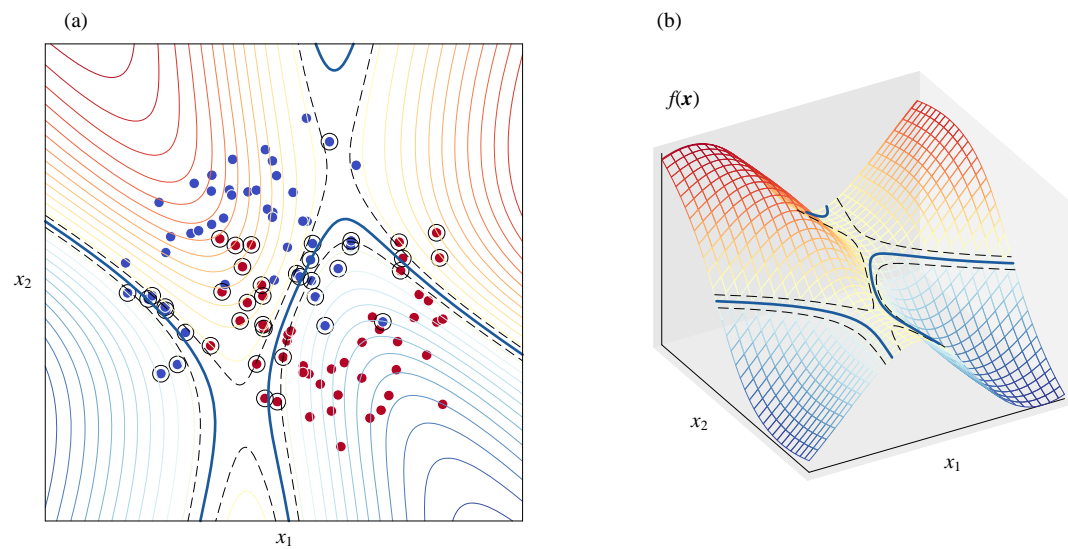


图 30. 月牙形数据, sigmoid 核 SVM



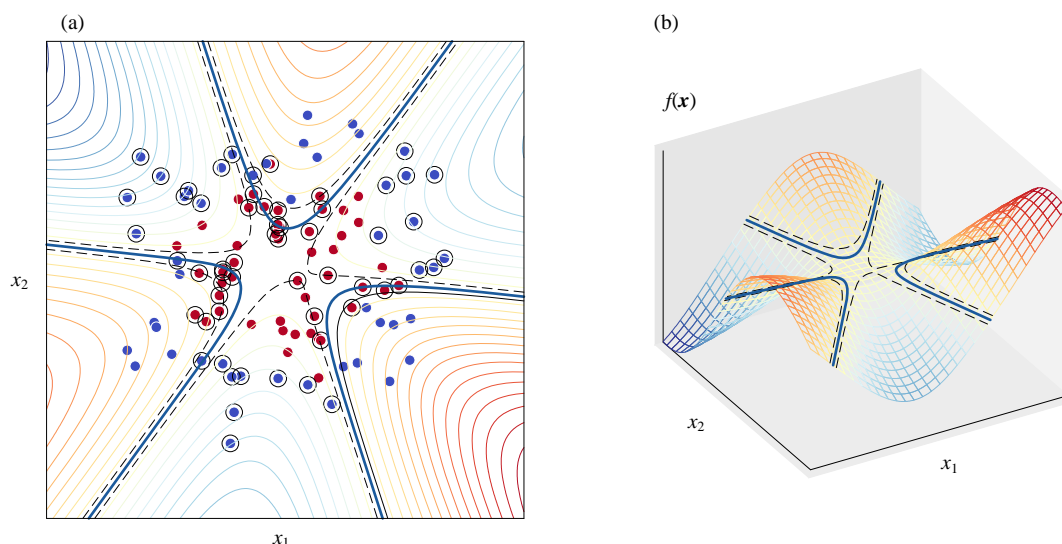


图 31. 环形数据, Sigmoid 核 SVM

代码 Bk7\_Ch12\_01.ipynb 可以获得本章线性可分、月牙形和环形数据，并且利用线性核、多项式核、高斯核和 sigmoid 核求解这三类数据分类。下面聊聊其中关键语句。

**a** 用 `sklearn.datasets.make_classification()` 生成样本数据。`n_features=2` 代表两个特征。`n_redundant` 设置冗余特征数量，`n_informative` 设置信息特征数量，`random_state` 设置随机种子，`n_clusters_per_class` 设置每个类别中的簇数量。

**b** 用 `sklearn.preprocessing.StandardScaler()` 标准化处理将特征数据  $X$  进行标准化，使其均值为 0，方差为 1。

**c** 用 `sklearn.svm.SVC()` 创建一个径向基核的支持向量机分类器 (`clf`)，设置 `gamma` 值为 0.7，正则化参数为 3。

**d** 使用 `fit` 方法训练支持向量机分类器，将标准化后的数据  $X$  和目标标签  $y$  传递给模型。

**e** 使用决策函数方法 `decision_function()` 获取输入网格点的决策值 ( $Z$ )，其中 `xx` 和 `yy` 是网格的坐标。

**f** 将决策值  $Z$  重新塑造成与网格坐标相同的形状，以便后续可视化。

**g** 使用决策函数获取训练数据的决策函数值。



```

from sklearn.preprocessing import StandardScaler
from sklearn.datasets import make_classification
from sklearn import svm

# 生成数据
a X, y = make_classification(n_features=2,
                             n_redundant=0,
                             n_informative=2,
                             random_state=1,
                             n_clusters_per_class=1)

# 标准化处理
b X = StandardScaler().fit_transform(X)

# 创建SVM分类器
c clf = svm.SVC(kernel='rbf', gamma=0.7, C=3)

# 训练数据
d clf.fit(X, y)

# 计算网格点决策函数值
e Z = clf.decision_function(np.c_[xx.ravel(), yy.ravel()])
f Z = Z.reshape(xx.shape)

# 计算样本点决策函数值
g Z_0 = clf.decision_function(X)

# 提取支持向量
x1_sp_vec = clf.support_vectors_[:, 0]
x2_sp_vec = clf.support_vectors_[:, 1]

```

代码 1. 用 sklearn.svm.SVC() 完成分类 | Bk7\_Ch12\_01.ipynb

在 SVM 中，核技巧可以将输入数据映射到高维空间中，使得原本的非线性问题转化为线性问题，从而提高模型的分类性能。常用的核函数包括线性核函数、多项式核函数和径向基函数 RBF 核函数等。

线性核函数适用于线性可分的数据集，可以在低维空间中很好地工作，但在处理非线性问题时效果较差。多项式核函数可以处理一些简单的非线性问题，但需要调整多项式的次数。RBF 核函数是最常用的核函数之一，能够处理更加复杂的非线性问题。

不同的核函数适用于不同的数据集和问题类型。选择适当的核函数和调整参数是使用 SVM 的关键。在实践中，可以使用交叉验证等技术来选择最佳的核函数和参数。