机器学习-支持向量机

黄海广 副教授

2022年02月

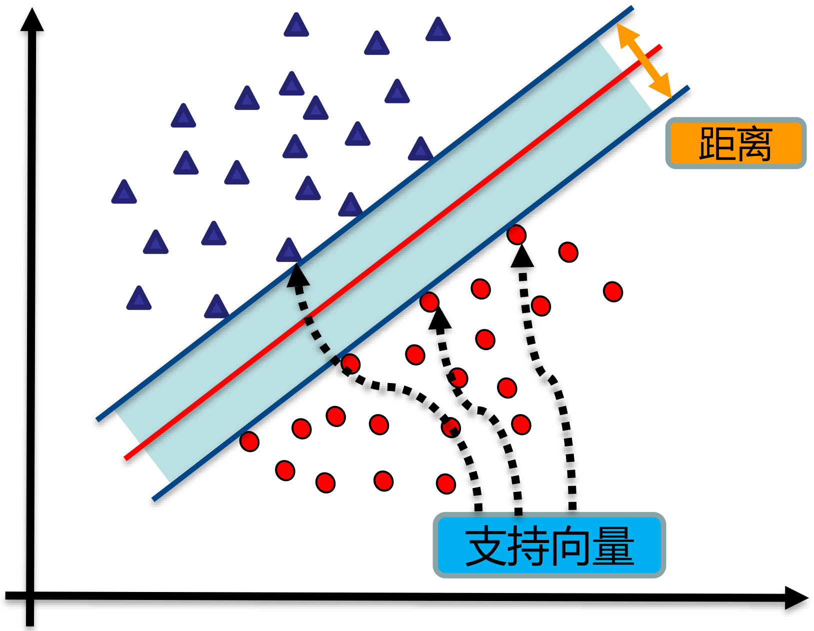
本章目录

1.支持向量机概述

1.支持向量机概述

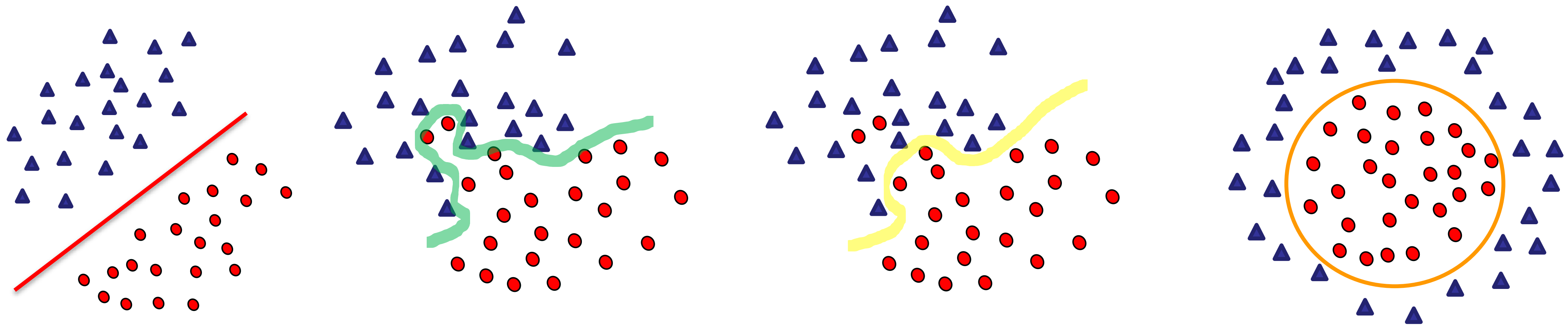
支持向量机（Support Vector Machine, SVM）是一类按监督学习（supervised learning）方式对数据进行二元分类的广义线性分类器（generalized linear classifier），其决策边界是对学习样本求解的最大边距超平面（maximum-margin hyperplane）。

与逻辑回归和神经网络相比，支持向量机，在学习复杂的非线性方程时提供了一种更为清晰，更加强大的方式。



1.支持向量机概述

硬间隔、软间隔和非线性 SVM



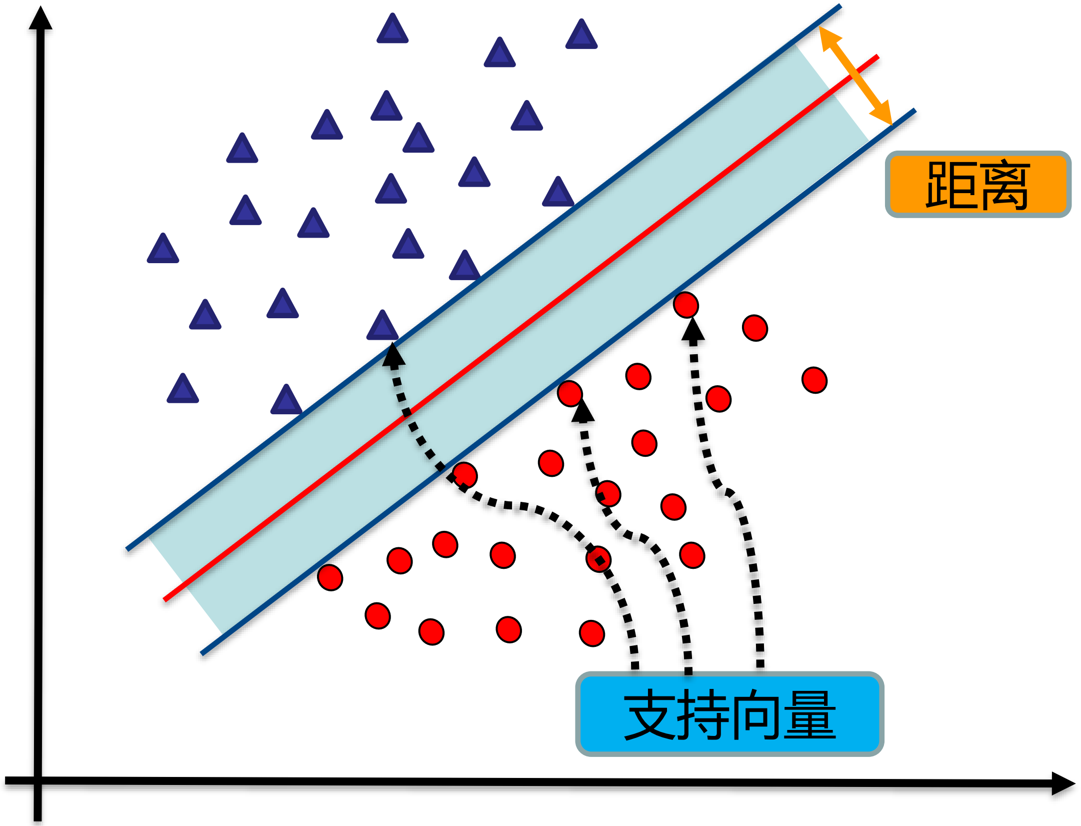
线性可分　　硬间隔　　软间隔　　线性不可分

假如数据是完全的线性可分的，那么学习到的模型可以称为硬间隔支持向量机。换个说法，硬间隔指的就是完全分类准确，不能存在分类错误的情况。软间隔，就是允许一定量的样本分类错误。

1.支持向量机概述

算法思想

找到集合边缘上的若干数据（称为支持向量（Support Vector））,用这些点找出一个平面（称为决策面），使得支持向量到该平面的距离最大。



1.支持向量机概述

背景知识

任意超平面可以用下面这个线性方程来描述：

$$w^{\mathrm{T}}x + b = 0$$

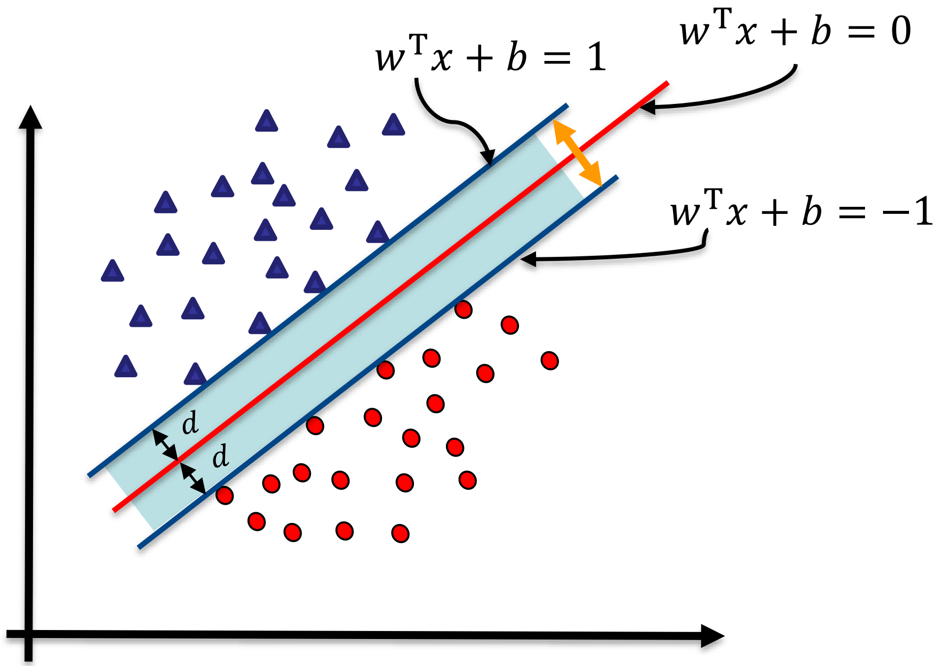
二维空间点 (x, y)到直线 $Ax + By + C = 0$的距离公式是：

$$\frac{|Ax + By + C|}{\sqrt{A^2 + B^2}}$$

扩展到 n 维空间后，点 $x = (x_1, x_2 \ldots x_n)$ 到超平面

$w^{\mathrm{T}}x + b = 0$ 的距离为：$\frac{|w^{\mathrm{T}}x+b|}{||w||}$

其中 $||w|| = \sqrt{w_1^2 + \cdots w_n^2}$



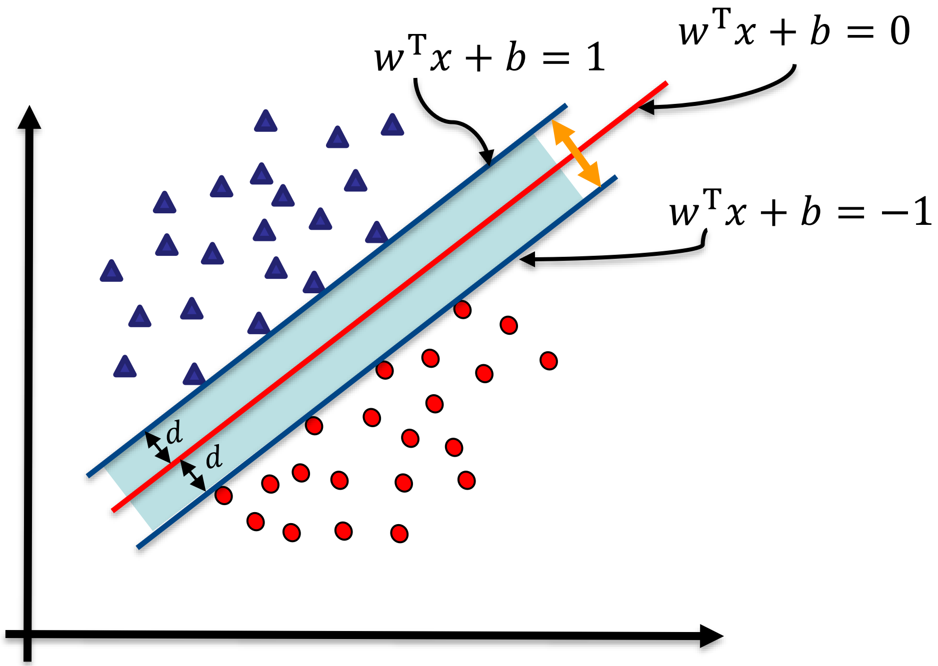
如图所示，根据支持向量的定义我们知道，支持向量到超平面的距离为 d，其他点到超平面的距离大于 d。每个支持向量到超平面的距离可以写为：$d = \frac{|w^{\mathrm{T}}x+b|}{||w||}$

1.支持向量机概述

背景知识

如图所示，根据支持向量的定义我们知道，支持向量到超平面的距离为 d，其他点到超平面的距离大于 d。

于是我们有这样的一个公式：故：$\begin{cases} \dfrac{w^{\mathrm{T}}x+b}{\|w\|} \geq d & y=1 \\ \dfrac{w^{\mathrm{T}}x+b}{\|w\|} \leq -d & y=-1 \end{cases}$



我们暂且令d为 1（之所以令它等于 1，是为了方便推导和优化，且这样做对目标函数的优化没有影响），

将两个方程合并，我们可以简写为： $y(w^{\mathrm{T}}x+b) \geq 1$

至此我们就可以得到最大间隔超平面的上下两个超平面： $d=\dfrac{|w^{\mathrm{T}}x+b|}{||w||}$

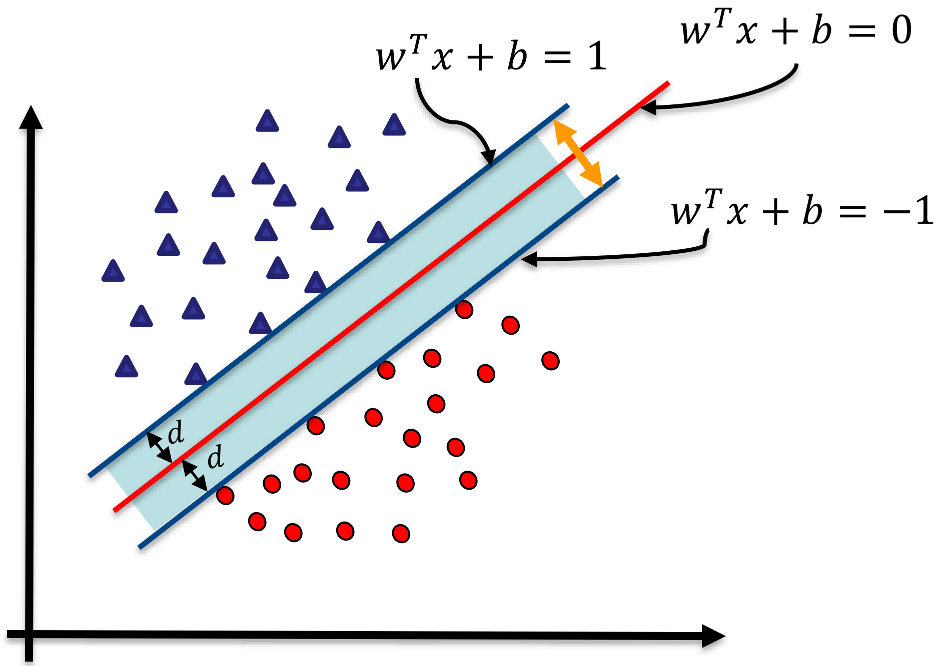
2.线性可分支持向量机

2.线性可分支持向量机

背景知识

点到面的距离公式

$$d = \frac{|Ax_0 + By_0 + Cz_0 + D|}{\sqrt{A^2 + B^2 + C^2}}$$



$$y(w^{\mathrm{T}}x + b) \geq 1 \qquad d = \frac{|w^{\mathrm{T}}x + b|}{||w||}$$

$$y(w^{\mathrm{T}}x + b) = |w^{\mathrm{T}}x + b|$$

支持向量机的最终目的是最大化d

2. 支持向量机求解

函数间隔： $d^* = y_i(w^{\mathrm{T}}x + b)$

几何间隔： $d = \frac{y(w^{\mathrm{T}}x+b)}{||w||}$，当数据被正确分类时，几何间隔就是点到超平面的距离

为了求几何间隔最大，SVM基本问题可以转化为求解:($\frac{d^*}{||w||}$为几何间隔，d^*为函数间隔)

$$\max_{w,b} \frac{d^*}{||w||}$$

$$(subject\ to)\ y_i(w^{\mathrm{T}}x_i + b) \geq d^*,\ i = 1,2,..,m$$

2.支持向量机求解

①转化为凸函数：

先令$d^* = 1$，方便计算（参照衡量，不影响评价结果）

$$\max_{w,b} \frac{1}{||w||}$$

$$s.t.\ y_i(w^{\mathrm{T}}x_i + b) \geq 1,\ i = 1,2,..,m$$

再将$\max\limits_{w,b} \frac{1}{||w||}$转化成 $\min\limits_{w,b} \frac{1}{2}||w||^2$求解凸函数，1/2是为了求导之后方便计算。

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

$$s.t.\ y_i(w^{\mathrm{T}}x_i + b) \geq 1,\ i = 1,2,..,m$$

2.支持向量机求解

②用拉格朗日乘子法和KKT条件求解最优值:

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

$$s.t.\ -y_i(w^{\mathrm{T}}x_i + b) + 1 \leq 0,\ i = 1,2,..,m$$

整合成：$L(w,b,\alpha) = \frac{1}{2}||w||^2 + \sum_{i=1}^{m} \alpha_i\,(-y_i(w^{\mathrm{T}}x_i + b) + 1)$　　其中α为拉格朗日乘子

推导:

根据Karush-Kuhn-Tucker（KKT）条件:

$$\frac{\partial}{\partial w}L(w,b,\alpha) = w - \sum_{i=1}^{m} \alpha_i\, y_i x_i = 0,\ w = \sum_{i=1}^{m} \alpha_i\, y_i x_i$$

$$\frac{\partial}{\partial b}L(w,b,\alpha) = \sum_{i=1}^{m} \alpha_i\, y_i = 0$$

2.支持向量机求解

$w = \sum_{i=1}^{m} \alpha_i y_i x_i$ $\sum_{i=1}^{m} \alpha_i y_i = 0$

代入$L(w, b, \alpha)$

$$\min_{w,b} L(w, b, \alpha) = \frac{1}{2}||w||^2 + \sum_{i=1}^{m} \alpha_i \left(-y_i(w^{\mathrm{T}} x_i + b) + 1\right)$$

$$= \frac{1}{2} w^{\mathrm{T}} w - \sum_{i=1}^{m} \alpha_i y_i w^{\mathrm{T}} x_i - b \sum_{i=1}^{m} \alpha_i y_i + \sum_{i=1}^{m} \alpha_i$$

$$= \frac{1}{2} w^{\mathrm{T}} \sum_{i=1}^{m} \alpha_i y_i x_i - \sum_{i=1}^{m} \alpha_i y_i w^{\mathrm{T}} x_i + \sum_{i=1}^{m} \alpha_i$$

$$= \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i=1}^{m} \alpha_i y_i w^{\mathrm{T}} x_i$$

$$= \sum_{i=1}^{m} \alpha_i - \sum_{i,j=1}^{m} \alpha_i \alpha_j y_i y_j \left(x_i \cdot x_j\right)$$

2.支持向量机求解

$w=\sum_{i=1}^{m}\alpha_i y_i x_i$ $\sum_{i=1}^{m}\alpha_i y_i=0$

再把max问题转成min问题： 添加负号

$$\max_{\alpha}\sum_{i=1}^{m}\alpha_i-\frac{1}{2}\sum_{i,j=1}^{m}\alpha_i\alpha_j y_i y_j(x_i\cdot x_j)=\min_{\alpha}\frac{1}{2}\sum_{i,j=1}^{m}\alpha_i\alpha_j y_i y_j(x_i\cdot x_j)-\sum_{i=1}^{m}\alpha_i$$

$$s.t.\sum_{i=1}^{m}\alpha_i y_i=0,$$

$$\alpha_i\geq 0, i=1,2,\ldots,m$$

得到最优解$\alpha^*=(\alpha_1^*,\alpha_2^*,\ldots,\alpha_m^*)^{\mathrm{T}}$

解出后，代入超平面模型也就是：

$y=w^{*\mathrm{T}}x+b^*=\sum_{i=1}^{m}\alpha_i^* y_i(x_i\cdot x_j)+b^*$，可得$b^*=y-\sum_{i=1}^{m}\alpha_i^* y_i(x_i\cdot x_j)$，$w^*=\sum_{i=1}^{m}\alpha_i^* y_i x_i$

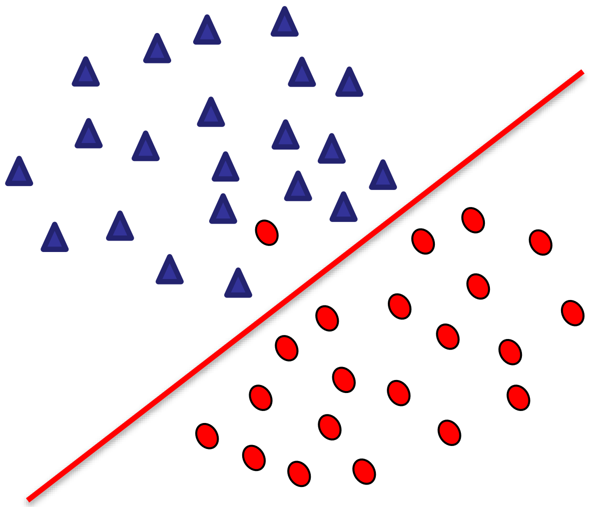
以上为SVM对偶问题的对偶形式

3.线性支持向量机

3.线性支持向量机

若数据线性不可分，则可以引入松弛变量$\xi \geq 0$，使函数间隔加上松弛变量大于等于1，则目标函数：

$$\min_{w,b,\xi} \ \frac{1}{2}||w||^2 + C\sum_{i=1}^{m}\xi_i \quad s.t.\ y_i(w^{\mathrm{T}}x_i + b) \geq 1 - \xi_i$$



对偶问题：

$$\max_{\alpha} \ \sum_{i=1}^{m}\alpha_i - \frac{1}{2}\sum_{i,j=1}^{m}\alpha_i\,\alpha_j y_i y_j\left(x_i \cdot x_j\right) = \min_{\alpha} \ \frac{1}{2}\sum_{i,j=1}^{m}\alpha_i\,\alpha_j y_i y_j(x_i \cdot x_j) - \sum_{i=1}^{m}\alpha_i$$

$$s.t.\ C \geq \alpha_i \geq 0, i = 1,2,\ldots,m \qquad \sum_{i=1}^{m}\alpha_i\, y_i = 0$$

C为惩罚参数，C 值越大，对分类的惩罚越大。跟线性可分求解的思路一致，同样这里先用拉格朗日乘子法得到拉格朗日函数，再求其对偶问题。

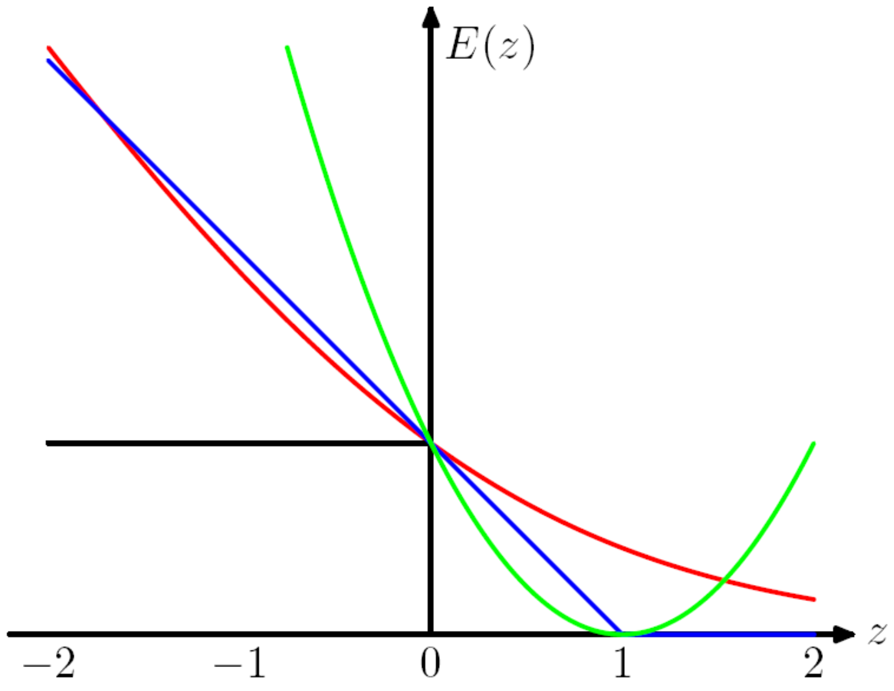
3.线性支持向量机

绿色的线为 square loss
蓝色的线为 hinge loss
红的的线为负 log loss

ξ 为"松弛变量"

$$\xi_i = \max(0, 1 - y_i(w^{\mathrm{T}} x_i + b))$$

即hinge损失函数。每一个样本都有一个对应的松弛变量，表征该样本不满足约束的程度。



损失函数hinge loss

3.线性支持向量机

求解原始最优化问题的解w^*和b^*，得到线性支持向量机，其分离超平面为

$$w^{*\mathrm{T}}x + b^* = 0$$

分类决策函数为：$f(x) = \mathrm{sign}\left(w^{*\mathrm{T}}x + b^*\right)$

线性可分支持向量机的解w^*唯一，但b^*不唯一。对偶问题是

$$\min_{\alpha} \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}\alpha_i\,\alpha_j y_i y_j\left(x_i \cdot x_j\right) - \sum_{i=1}^{m}\alpha_i$$

$$s.t. \quad \sum_{i=1}^{m}\alpha_i\,y_i = 0$$

$$0 \le \alpha_i \le C, \quad i = 1,2,\cdots,m$$

3.线性支持向量机

解出后，代入超平面模型：

$$w^{*\mathrm{T}}x + b^* = 0$$

可得

$$b^* = y_i - \sum_{i=1}^{m} \alpha_i^* y_i (x_i \cdot x_j)$$

$$w^* = \sum_{i=1}^{m} \alpha_i^* y_i x_i$$

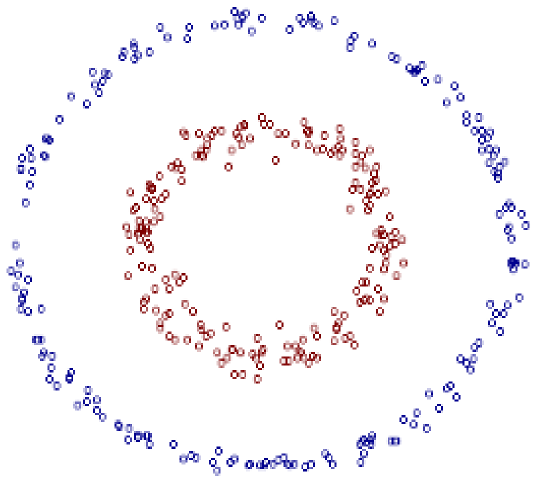
其中：$0 < \alpha_i^* < C$

4.线性不可分支持向量机

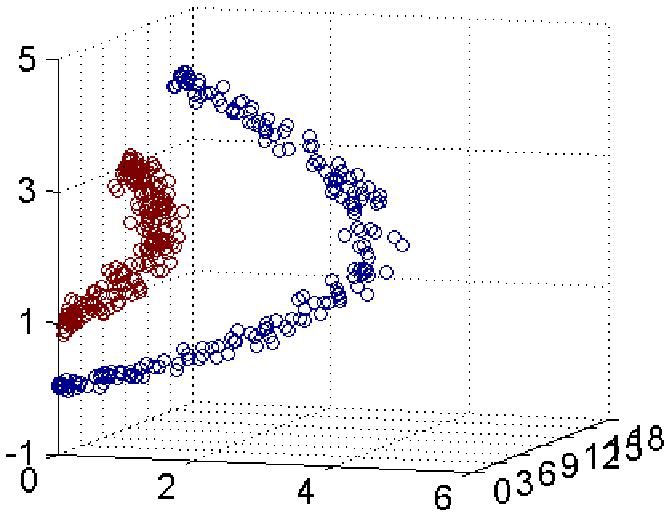
4.线性不可分支持向量机

核技巧

在低维空间计算获得高维空间的计算结果，满足高维，才能在高维下线性可分。 我们需要引入一个新的概念：**核函数。它可以将样本从原始空间映射到一个更高维的特质空间中，使得样本在新的空间中线性可分**。这样我们就可以使用原来的推导来进行计算，只是所有的推导是在新的空间，而不是在原来的空间中进行，即用核函数来替换当中的内积。



线性不可分

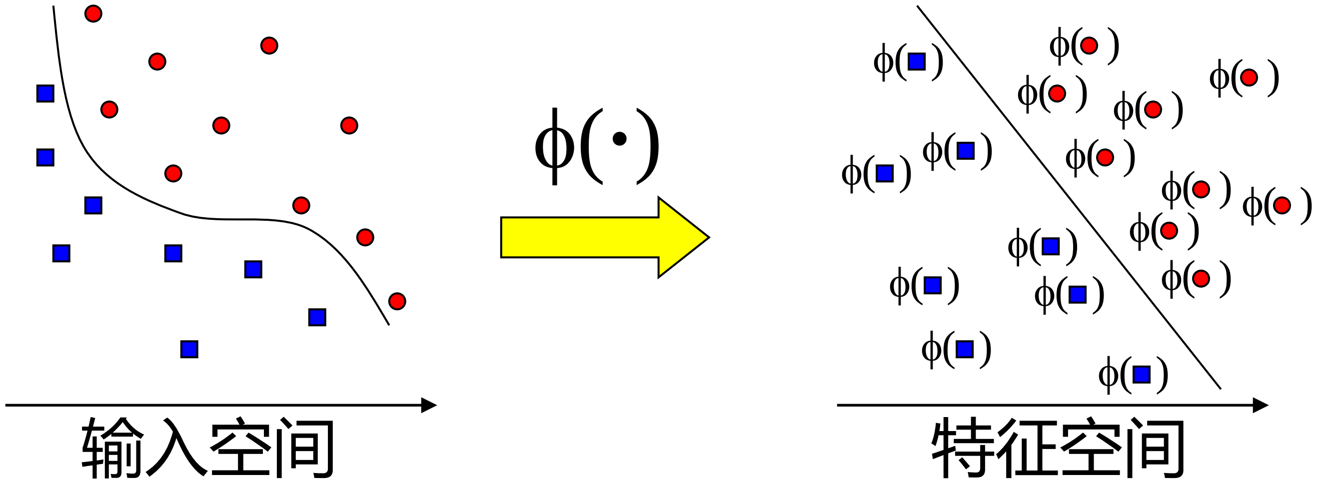


高维下线性可分

4.线性不可分支持向量机

核技巧

用核函数来替换原来的内积。



即通过一个非线性转换后的两个样本间的内积。具体地，$K(x,z)$是一个核函数，或正定核，意味着存在一个从输入空间到特征空间的映射，对于任意空间输入的x,z有:

$$K(x,z) = \phi(x) \cdot \phi(z)$$

4.线性不可分支持向量机

在线性支持向量机学习的对偶问题中，用核函数$K(x,z)$替代内积，求解得到的就是非线性支持向量机

$$f(x) = \operatorname{sign}\left(\sum_{i=1}^{N} \alpha_i^* y_i K(x, x_i) + b^*\right)$$

4.线性不可分支持向量机

常用核函数有：

线性核函数

$$K(x_i, x_j) = x_i^T x_j$$

多项式核函数

$$K(x_i, x_j) = (x_i^T x_j)^d$$

高斯核函数

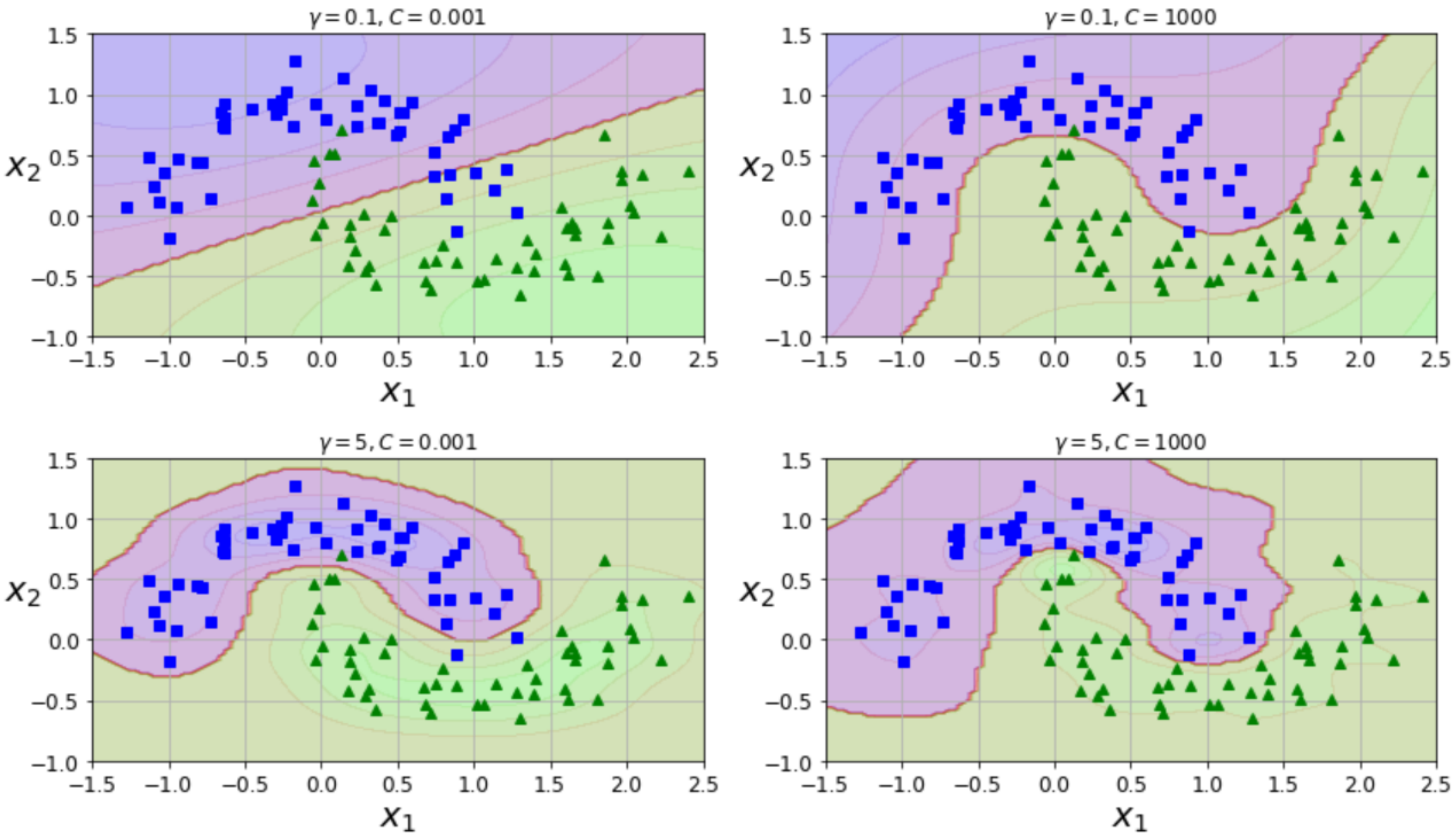
$$K(x_i, x_j) = exp(-\frac{||x_i - x_j||}{2\gamma^2})$$

这三个常用的核函数中,只有高斯核函数是需要调参的。

4.线性不可分支持向量机

SVM的超参数

γ越大，支持向量越少，γ值越小，支持向量越多。其中 C是惩罚系数，即对误差的宽容度。 C越高，说明越不能容忍出现误差，容易过拟合。C越小，容易欠拟合。



总结

下面是一些SVM普遍使用的准则：

n为特征数，m为训练样本数。

(1)如果相较于m而言，n要大许多，即训练集数据量不够支持我们训练一个复杂的非线性模型，我们选用逻辑回归模型或者不带核函数的支持向量机。

(2)如果n较小，而且m大小中等，例如n在 1-1000 之间，而m在10-10000之间，使用高斯核函数的支持向量机。

(3)如果n较小，而m较大，例如n在1-1000之间，而m大于50000，则使用支持向量机会非常慢，解决方案是创造、增加更多的特征，然后使用逻辑回归或不带核函数的支持向量机。

参考文献

[1] CORTES C, VAPNIK V. Support-vector networks[J]. Machine learning, 1995, 20(3): 273–297.

[2] Andrew Ng. Machine Learning[EB/OL]. StanfordUniversity,2014.https://www.coursera.org/course/ml

[3] 李航. 统计学习方法[M]. 北京: 清华大学出版社,2019.

[4] Hastie T., Tibshirani R., Friedman J. The Elements of Statistical Learning[M]. New York: Springer,2001.

[5] CHRISTOPHER M. BISHOP. Pattern Recognition and Machine Learning[M]. New York: Springer,2006.

[6] Stephen Boyd, Lieven Vandenberghe, Convex Optimization[M]. Cambridge: Cambridge University Press, 2004.

[7] PLATT J. Sequential Minimal Optimization: A Fast Algorithm for Training Support Vector Machines[J]. Advances in Kernel Methods-Support Vector Learning, 1998, 208.


谢 谢！