机器学习
第5节

涉及知识点：
支持向量机简介、支持向量机优化、序列最小优化算法、支持向量机核方法

支持向量机

张伟楠 – 上海交通大学

支持向量机简介

张伟楠 – 上海交通大学

目录

Contents

01 线性分类器

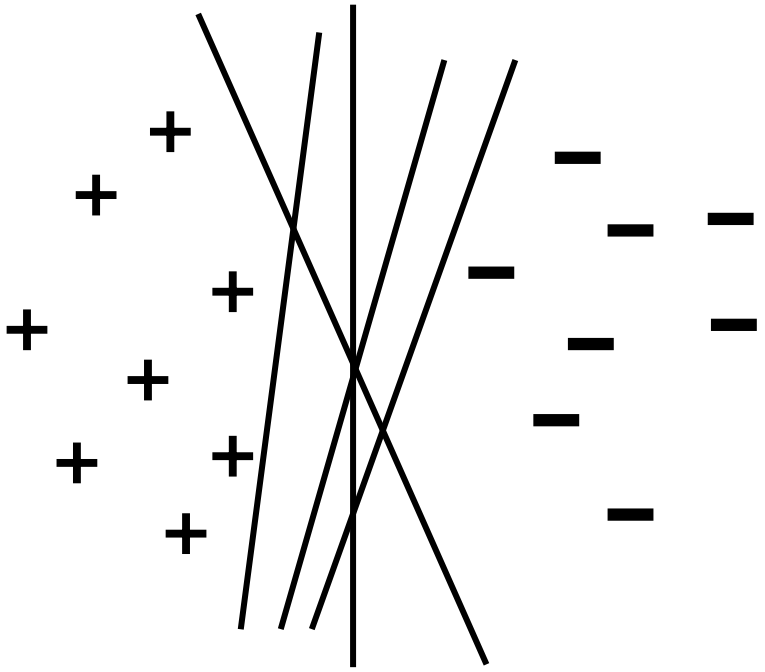
02 支持向量机

01
线性分类器

线性分类器

决策边界

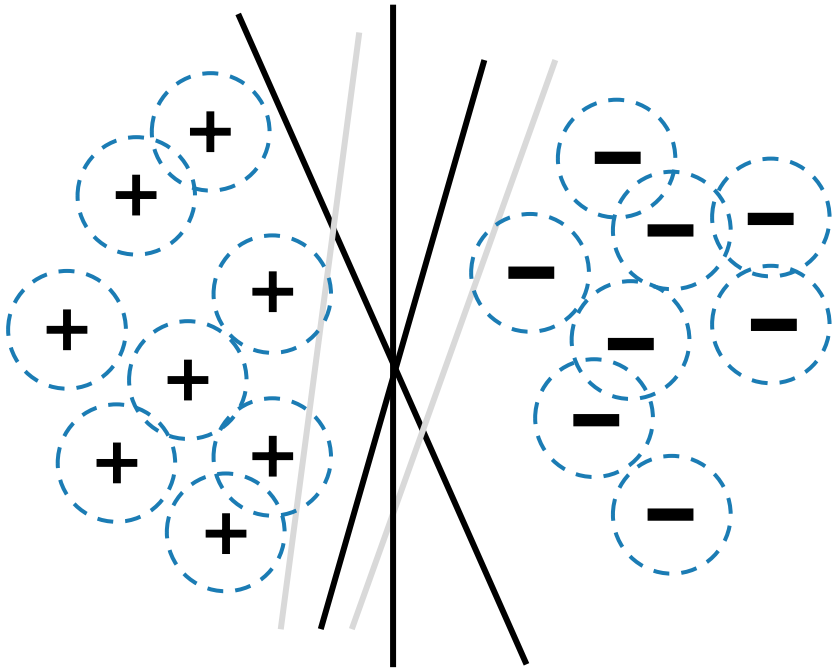
- 线性可分的情形下，决策边界可以是多样的



线性分类器

决策边界

- 线性可分的情形下，决策边界可以是多样的

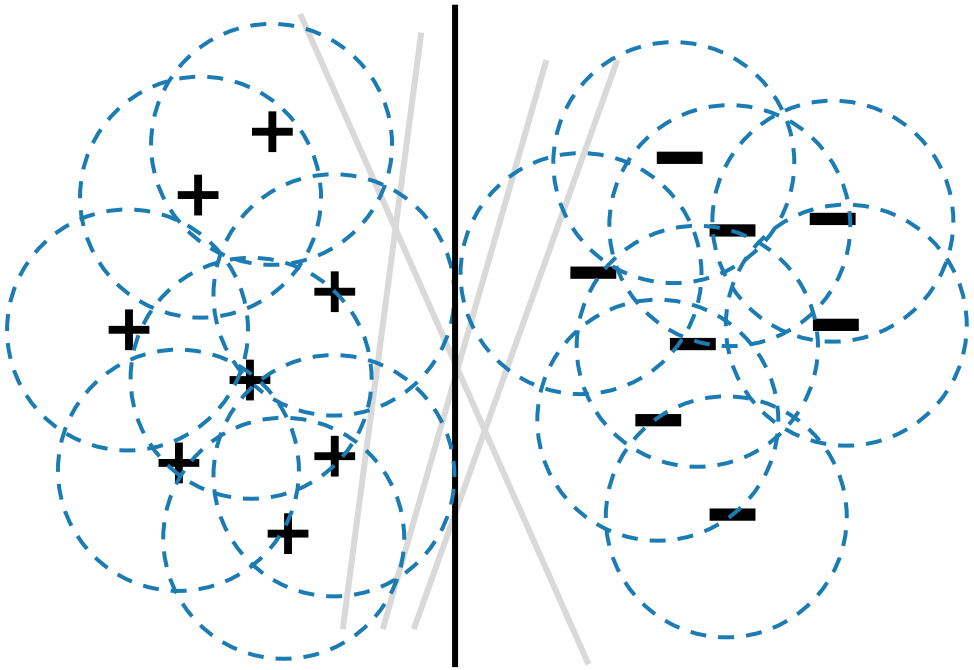


- 考虑数据噪声，可以去除一些划分

线性分类器

决策边界

- 线性可分的情形下，决策边界可以是多样的



- 一种直观的最优决策边界：最大间隔边界

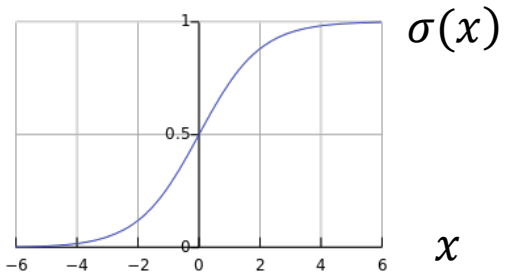
线性分类器

逻辑回归

- 逻辑回归是一种二分类模型

$$p_\theta(y=1|x)=\sigma(\theta^T x)=\frac{1}{1+e^{-\theta^T x}}$$

$$p_\theta(y=0|x)=\frac{e^{-\theta^T x}}{1+e^{-\theta^T x}}$$



- 交叉熵损失函数

$$\mathcal{L}(y,x,p_\theta)=-y\log\sigma(\theta^T x)-(1-y)\log\big(1-\sigma(\theta^T x)\big)$$

- 梯度函数

$$\frac{\partial\mathcal{L}(y,x,p_\theta)}{\partial\theta}=-y\frac{1}{\sigma(\theta^T x)}\sigma(z)(1-\sigma(z))x-(1-y)\frac{-1}{1-\sigma(\theta^T x)}\sigma(z)(1-\sigma(z))x$$

$$=(\sigma(\theta^T x)-y)x$$

$$\theta\leftarrow\theta+\eta\big(y-\sigma(\theta^T x)\big)x$$

线性分类器

标签决策

- 逻辑回归给出了每个类别的概率

$$p_\theta(y=1|x)=\sigma(\theta^T x)=\frac{1}{1+e^{-\theta^T x}}$$

$$p_\theta(y=0|x)=\frac{e^{-\theta^T x}}{1+e^{-\theta^T x}}$$

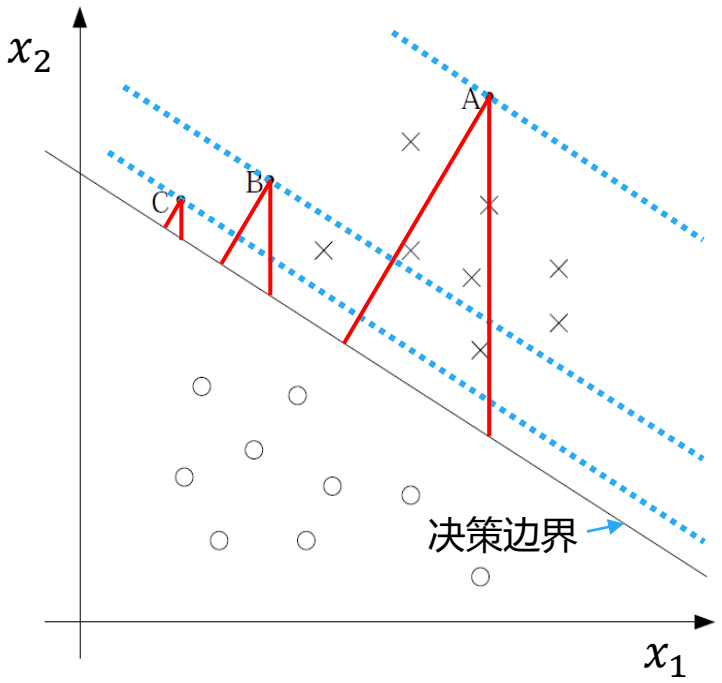
- 每个样例的最终标签由设定的阈值h决定

$$\hat{y}=\begin{cases}1, & p_\theta(y=1|x)>h\\ 0, & \text{otherwise}\end{cases}$$

线性分类器

打分函数

- 逻辑回归的打分函数 $s(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2$



$$p_\theta(y = 1|x) = \frac{1}{1 + e^{-s(x)}}$$

$$s(x) = \theta_0 + \theta_1 x_1^{(A)} + \theta_2 x_2^{(A)}$$

$$s(x) = \theta_0 + \theta_1 x_1^{(B)} + \theta_2 x_2^{(B)}$$

$$s(x) = \theta_0 + \theta_1 x_1^{(C)} + \theta_2 x_2^{(C)}$$

$$s(x) = 0$$

打分越高的样例越远离决策边界，具有更高的分类置信度

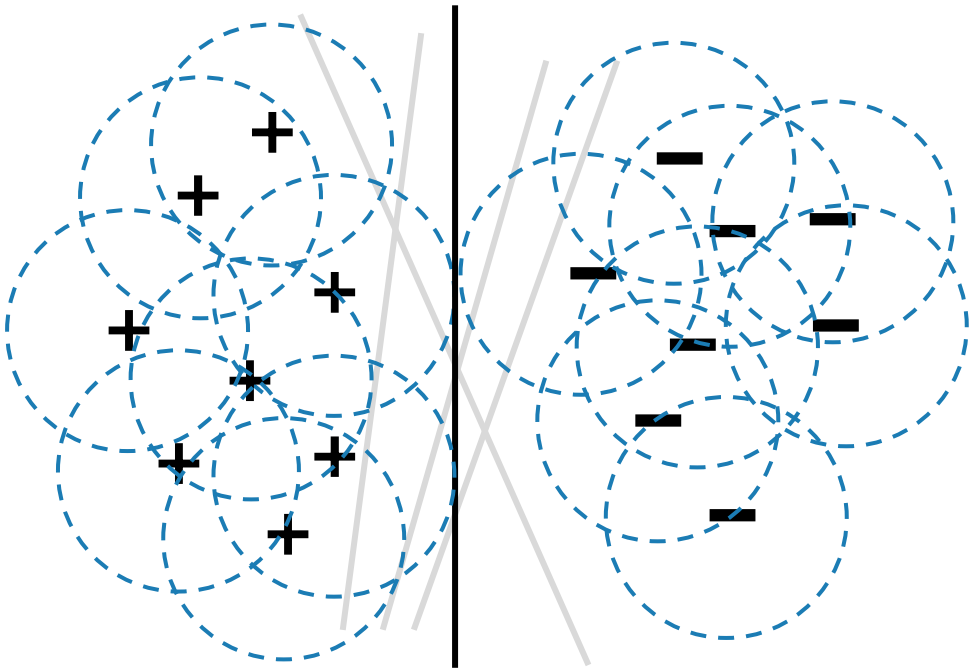
Hangout from Prof. Andrew Ng. https://cs229.stanford.edu/lectures-spring2022/main_notes.pdf

02
支持向量机

支持向量机

最优决策边界

- 直观的最优决策边界：最高的分类置信度



支持向量机

符号说明

- 特征向量 x
- 类别标签 $y \in \{-1, 1\}$
- 参数
  - 截距 b
  - 特征权重向量 w
- 标签预测

$$h_{w,b}(x) = g(w^{\top}x + b)$$

$$g(z) = \begin{cases} +1 & z \geq 0 \\ -1 & \text{otherwise} \end{cases}$$

支持向量机

边界间隔

- 函数间隔

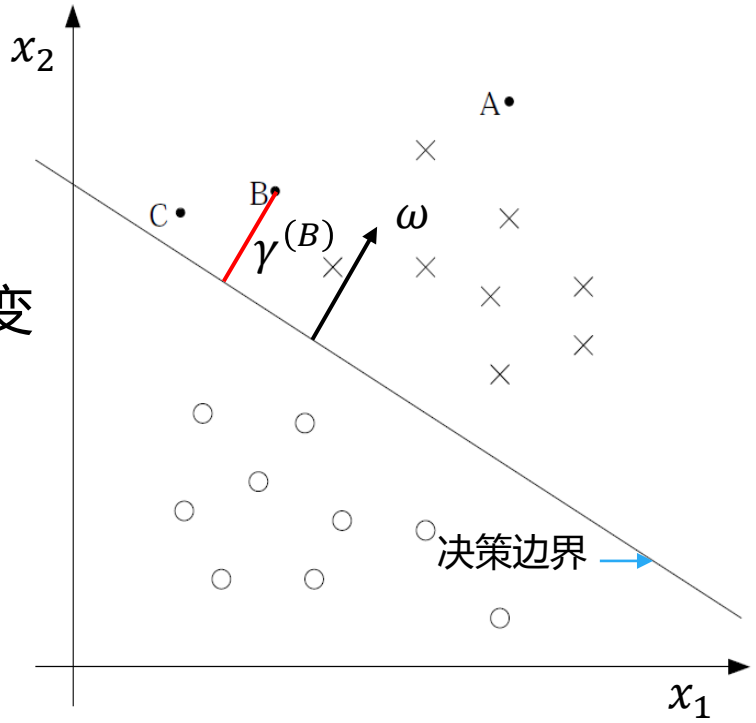
$$\hat{\gamma}^{(i)} = y^{(i)}\left(w^T x^{(i)} + b\right)$$

- 分割超平面不会随(ω, b)的幅值改变而改变

$$g(w^T x + b) = g(2w^T x + 2b)$$

- 几何间隔

$$\gamma^{(i)} = y^{(i)}\left(w^T x^{(i)} + b\right), \text{where } \|w\|^2 = 1$$



支持向量机

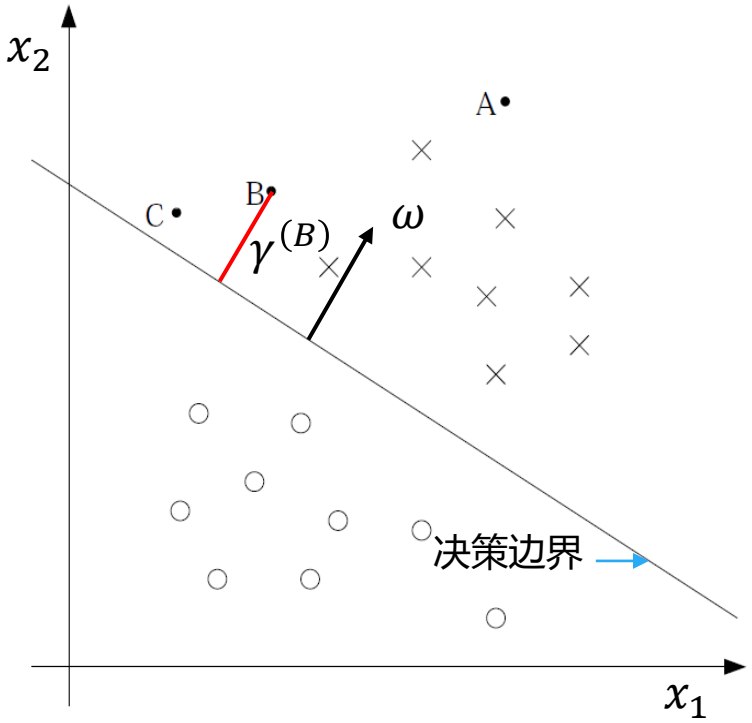
边界间隔

- 决策边界

$$w^T(x^{(i)} - \gamma^{(i)} y^{(i)} \frac{w}{\|w\|}) + b = 0$$

$$\Rightarrow \gamma^{(i)} = y^{(i)} \frac{w^T x^{(i)} + b}{\|w\|}$$

$$= y^{(i)} \left[\left(\frac{w}{\|w\|} \right)^T x^{(i)} + \frac{b}{\|w\|} \right]$$



- 给定训练集$S = \{(x_i, y_i)\}_{i=1,\ldots,m}$，最小几何间隔为

$$\gamma = \min_{i=1,\ldots,m} \gamma^{(i)}$$

支持向量机

目标函数

- 寻找一个使最小几何间隔达到最大值的分割超平面

$$\max_{\gamma,w,b} \quad \gamma$$

$$\text{s.t.} \quad y^{(i)}\left(w^T x^{(i)} + b\right) \geq \gamma, \qquad i = 1, \ldots, m$$

$$\|w\| = 1 \qquad \text{(非凸约束)}$$

- 等同于归一化函数间隔

$$\max_{\hat{\gamma},w,b} \quad \frac{\hat{\gamma}}{\|w\|} \qquad \text{(非凸目标函数)}$$

$$\text{s.t.} \quad y^{(i)}\left(w^T x^{(i)} + b\right) \geq \hat{\gamma}, \qquad i = 1, \ldots, m$$

支持向量机

目标函数

- 分类间隔的变化不会改变决策边界
  - 将函数间隔固定为1

$$\hat{\gamma} = 1$$

  - 目标函数重写成

$$\max_{w,b} \quad \frac{1}{\|w\|}$$
$$\text{s.t.} \quad y^{(i)}\left(w^T x^{(i)} + b\right) \geq 1, \qquad i = 1, \ldots, m$$

  - 目标函数等同于

$$\min_{w,b} \quad \frac{1}{2}\|w\|^2$$
$$\text{s.t.} \quad y^{(i)}\left(w^T x^{(i)} + b\right) \geq 1, \qquad i = 1, \ldots, m$$

- 此优化问题可以由二次规划算法有效求解

支持向量机优化

讲师：张伟楠 – 上海交通大学

目录

Contents

01 拉格朗日对偶问题

02 支持向量机优化求解

01
拉格朗日对偶问题

拉格朗日对偶问题

等式凸优化

- 对于凸优化问题

$$\min_{w} \quad f(w)$$
$$\text{s.t.} \quad h_i(w) = 0, \qquad i = 1, \dots, l$$

- 问题的拉格朗日函数定义为

$$\mathcal{L}(w, \beta) = f(w) + \sum_{i=1}^{l} \beta_i \, h_i(w)$$

拉格朗日乘子

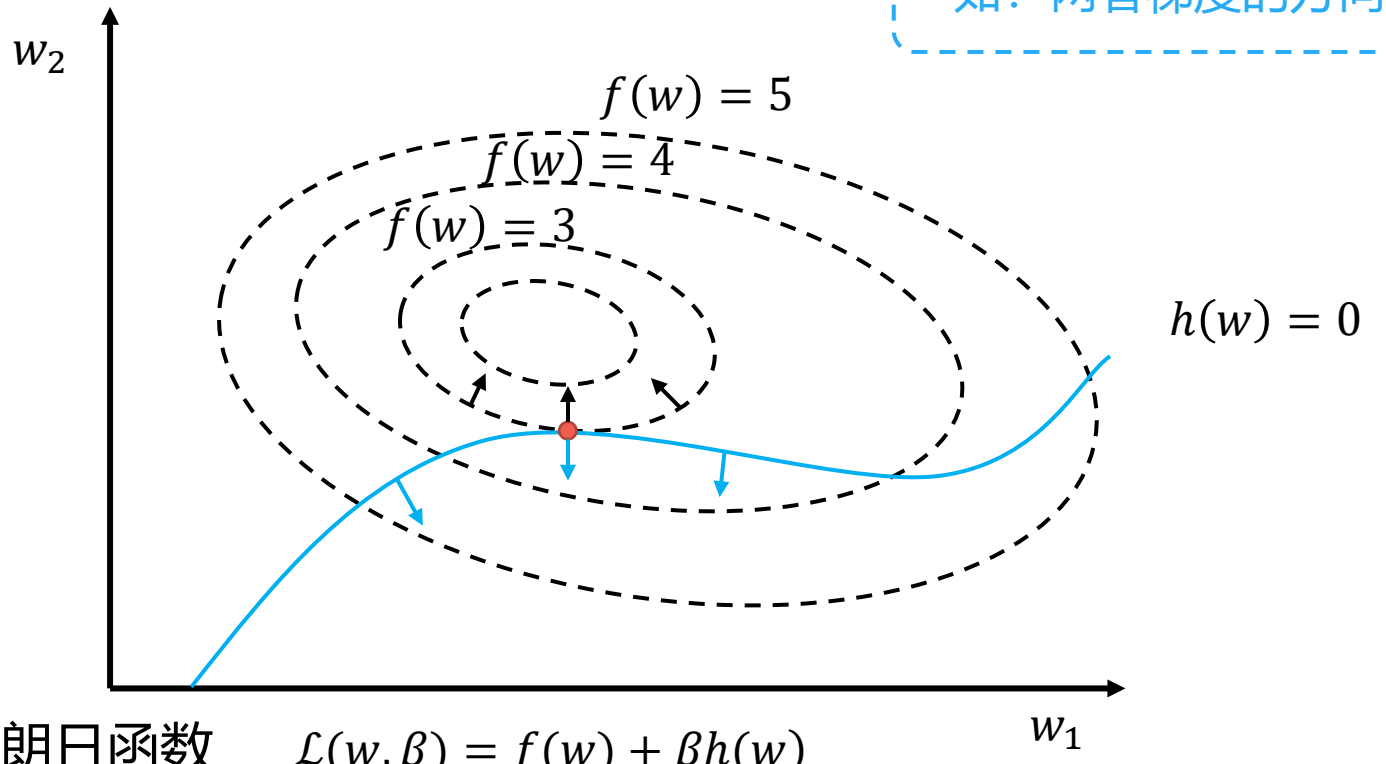
  - 求解 $\dfrac{\partial \mathcal{L}(w, \beta)}{\partial w} = 0$ $\dfrac{\partial \mathcal{L}(w, \beta)}{\partial \beta} = 0$

    可得原优化问题的解

拉格朗日对偶问题

拉格朗日函数解析

如：两者梯度的方向相同



- 拉格朗日函数 $\mathcal{L}(w, \beta) = f(w) + \beta h(w)$

$$\frac{\partial \mathcal{L}(w,\beta)}{\partial w} = \frac{\partial f(w)}{\partial w} + \beta \frac{\partial h(w)}{\partial w} = 0$$

拉格朗日对偶问题

不等式凸优化

- 对于凸优化问题

$$\begin{aligned} \min_{w} \quad & f(w) \\ s.t. \quad & g_i(w) \le 0, \qquad i = 1, \dots, k \\ & h_i(w) = 0, \qquad i = 1, \dots, l \end{aligned}$$

- 问题的拉格朗日函数定义为

$$\mathcal{L}(w, \alpha, \beta) = f(w) + \sum_{i=1}^{k} \alpha_i \, g_i(w) + \sum_{i=1}^{l} \beta_i \, h_i(w)$$

拉格朗日乘子

拉格朗日对偶问题

原始问题

- 凸优化问题

$$\begin{aligned} \min_{w} \quad & f(w) \\ s.t. \quad & g_i(w) \le 0, \qquad i = 1, \dots, k \\ & h_i(w) = 0, \qquad i = 1, \dots, l \end{aligned}$$

- 拉格朗日函数

$$\mathcal{L}(w, \alpha, \beta) = f(w) + \sum_{i=1}^{k} \alpha_i \, g_i(w) + \sum_{i=1}^{l} \beta_i \, h_i(w)$$

- 原问题

$$\theta_{\mathcal{P}}(w) = \max_{\alpha, \beta : \alpha_i \ge 0} \mathcal{L}(w, \alpha, \beta)$$

  - 对于不满足约束条件的w，例如

$$g_i(w) > 0 \text{ 或者 } h_i(w) \neq 0$$

可得 $\theta_{\mathcal{P}}(w) = +\infty$

拉格朗日对偶问题

原始问题

- 凸优化问题

$$\begin{aligned} \min_{w} \quad & f(w) \\ s.t. \quad & g_i(w) \le 0, \qquad i = 1, \dots, k \\ & h_i(w) = 0, \qquad i = 1, \dots, l \end{aligned}$$

- 拉格朗日函数

$$\mathcal{L}(w, \alpha, \beta) = f(w) + \sum_{i=1}^{k} \alpha_i \, g_i(w) + \sum_{i=1}^{l} \beta_i \, h_i(w)$$

- 原问题

$$\theta_{\mathcal{P}}(w) = \max_{\alpha, \beta : \alpha_i \ge 0} \mathcal{L}(w, \alpha, \beta)$$

  - 相反，对于满足所有约束条件的w

可得 $\theta_{\mathcal{P}}(w) = f(w)$

$$\theta_{\mathcal{P}}(w) = \begin{cases} f(w) & w\text{满足原始问题约束} \\ +\infty & \text{其他} \end{cases}$$

拉格朗日对偶问题

原问题

$$\theta_{\mathcal{P}}(w) = \begin{cases} f(w) & w\text{满足原始问题约束} \\ +\infty & \text{其他} \end{cases}$$

- 函数最小化问题

$$\min_{w} \theta_{\mathcal{P}}(w) = \min_{w} \max_{\alpha,\beta:\alpha_i \geq 0} \mathcal{L}(w,\alpha,\beta)$$

  - 等同于原来的优化任务

$$\begin{aligned} \min_{w} \quad & f(w) \\ \text{s.t.} \quad & g_i(w) \leq 0, \qquad i = 1,\dots,k \\ & h_i(w) = 0, \qquad i = 1,\dots,l \end{aligned}$$

- 定义原始问题的解为

$$p^* = \min_{w} \theta_{\mathcal{P}}(w)$$

拉格朗日对偶问题

对偶问题

- 略不相同的问题

$$\theta_{\mathcal{D}}(\alpha,\beta)=\min_{w}\mathcal{L}(w,\alpha,\beta)$$

- 定义对偶优化问题

$$\max_{\alpha,\beta:\alpha_i\geq 0}\theta_{\mathcal{D}}(\alpha,\beta)=\max_{\alpha,\beta:\alpha_i\geq 0}\min_{w}\mathcal{L}(w,\alpha,\beta)$$

- 交换了原始问题中的min和max

$$\min_{w}\theta_{\mathcal{P}}(w)=\min_{w}\max_{\alpha,\beta:\alpha_i\geq 0}\mathcal{L}(w,\alpha,\beta)$$

- 定义对偶问题的解为

$$d^*=\max_{\alpha,\beta:\alpha_i\geq 0}\min_{w}\mathcal{L}(w,\alpha,\beta)$$

拉格朗日对偶问题

问题对比

- d^*与p^*的大小关系

$$d^* = \max_{\alpha,\beta:\alpha_i \geq 0} \min_{w} \mathcal{L}(w,\alpha,\beta) \leq \min_{w} \max_{\alpha,\beta:\alpha_i \geq 0} \mathcal{L}(w,\alpha,\beta) = p^*$$

- 证明

$$\min_{w'} \mathcal{L}(w',\alpha,\beta) \leq \mathcal{L}(w,\alpha,\beta), \qquad \forall w, \alpha \geq 0, \beta$$

$$\Rightarrow \max_{\alpha,\beta:\alpha_i \geq 0} \min_{w'} \mathcal{L}(w',\alpha,\beta) \leq \max_{\alpha,\beta:\alpha_i \geq 0} \mathcal{L}(w,\alpha,\beta), \qquad \forall w$$

$$\Rightarrow \max_{\alpha,\beta:\alpha_i \geq 0} \min_{w'} \mathcal{L}(w',\alpha,\beta) \leq \min_{w} \max_{\alpha,\beta:\alpha_i \geq 0} \mathcal{L}(w,\alpha,\beta)$$

□

- 满足一定条件，可得 $d^* = p^*$

拉格朗日对偶问题

KKT条件

- 假设f以及g_i是凸函数，并且h_i为仿射函数，且g_i严格满足可行域
- 必然存在 (w^*, α^*, β^*)，满足
  - w^*是原始问题的解
  - α^*, β^*是对偶问题的解
  - 两个问题的解数值相等 $p^* = d^* = \mathcal{L}(w^*, \alpha^*, \beta^*)$
- 同时，(w^*, α^*, β^*) 满足KKT条件

拉格朗日对偶问题

KKT条件

- (w^*, α^*, β^*) 满足KKT条件
  - $\frac{\partial}{\partial w_i}\mathcal{L}(w^*, \alpha^*, \beta^*) = 0, i = 1, \dots, n$
  - $\frac{\partial}{\partial \beta_i}\mathcal{L}(w^*, \alpha^*, \beta^*) = 0, i = 1, \dots, l$
  - $\alpha_i^* g_i(w^*) = 0, i = 1, \dots, k$
  - $g_i(w^*) \leq 0, i = 1, \dots, k$
  - $\alpha^* \geq 0, i = 1, \dots, k$

KKT对偶互补条件

- 如果存在 (w^*, α^*, β^*)满足KKT条件，则这组参数同时也是原始问题以及对偶问题的解

Boyd, Stephen, and Lieven Vandenberghe. Convex optimization. Cambridge university press, 2004.
知乎KKT解释：https://zhuanlan.zhihu.com/p/26514613

02
支持向量机优化求解

REVIEW：支持向量机优化目标

目标函数

- 寻找一个使最小几何间隔达到最大值的分割超平面

$$\max_{\gamma,\omega,b} \quad \gamma$$

$$\text{s.t.} \quad y^{(i)}\left(w^T x^{(i)} + b\right) \geq \gamma, \quad i = 1, \dots, m$$

$$\|\omega\| = 1 \qquad \text{(非凸约束)}$$

$$\max_{\hat{\gamma},w,b} \quad \frac{\hat{\gamma}}{\|w\|} \qquad \text{(非凸目标函数)}$$

$$\text{s.t.} \quad y^{(i)}\left(w^T x^{(i)} + b\right) \geq \hat{\gamma}$$

$$i = 1, \dots, m$$

$$\hat{\gamma} = 1$$

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

$$\text{s.t.} \quad y^{(i)}\left(w^T x^{(i)} + b\right) \geq 1, \; i = 1, \dots, m$$

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

$$\text{s.t.} \quad y^{(i)}\left(w^T x^{(i)} + b\right) \geq 1,$$

$$i = 1, \dots, m$$

支持向量机优化求解

目标函数

- 支持向量机的目标函数：寻找最优间隔分类器

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

$$\text{s.t.} \quad y^{(i)}\left(w^T x^{(i)} + b\right) \geq 1, \qquad i = 1, \dots, m$$

  - 重写约束条件

$$g_i(w) = -y^{(i)}\left(w^T x^{(i)} + b\right) + 1 \leq 0$$

  - 对应标准优化形式

$$\min_{w} \ f(w)$$

$$s.t. \ \ g_i(w) \leq 0, \qquad i = 1, \dots, k$$

$$h_i(w) = 0, \qquad i = 1, \dots, l$$

REVIEW: 拉格朗日对偶问题

原始问题

- 凸优化问题

$$\begin{aligned} \min_{w} \quad & f(w) \\ s.t. \quad & g_i(w) \le 0, \qquad i = 1, \dots, k \\ & h_i(w) = 0, \qquad i = 1, \dots, l \end{aligned}$$

- 拉格朗日函数

$$\mathcal{L}(w, \alpha, \beta) = f(w) + \sum_{i=1}^{k} \alpha_i \, g_i(w) + \sum_{i=1}^{l} \beta_i \, h_i(w)$$

- 原问题

$$\theta_{\mathcal{P}}(w) = \max_{\alpha, \beta : \alpha_i \ge 0} \mathcal{L}(w, \alpha, \beta)$$

  - 对于不满足约束条件的w，例如$g_i(w) > 0$ 或者$h_i(w) \neq 0$ ，可得 $\theta_{\mathcal{P}}(w) = +\infty$

- 定义原始问题的解为 $p^* = \min_{w} \theta_{\mathcal{P}}(w) = \min_{w} \max_{\alpha, \beta : \alpha_i \ge 0} \mathcal{L}(w, \alpha, \beta)$ ← 不好直接解

- 定义对偶问题的解为 $d^* = \max_{\alpha, \beta : \alpha_i \ge 0} \min_{w} \mathcal{L}(w, \alpha, \beta)$ ← 容易直接解

REVIEW: 拉格朗日对偶问题

- d^*与p^*的大小关系

$$d^* = \max_{\alpha,\beta:\alpha_i \geq 0} \min_{w} \mathcal{L}(w,\alpha,\beta) \leq \min_{w} \max_{\alpha,\beta:\alpha_i \geq 0} \mathcal{L}(w,\alpha,\beta) = p^*$$

- 满足KKT条件，$d^* = p^*$成立，这时解对偶问题就是解原问题

KKT条件

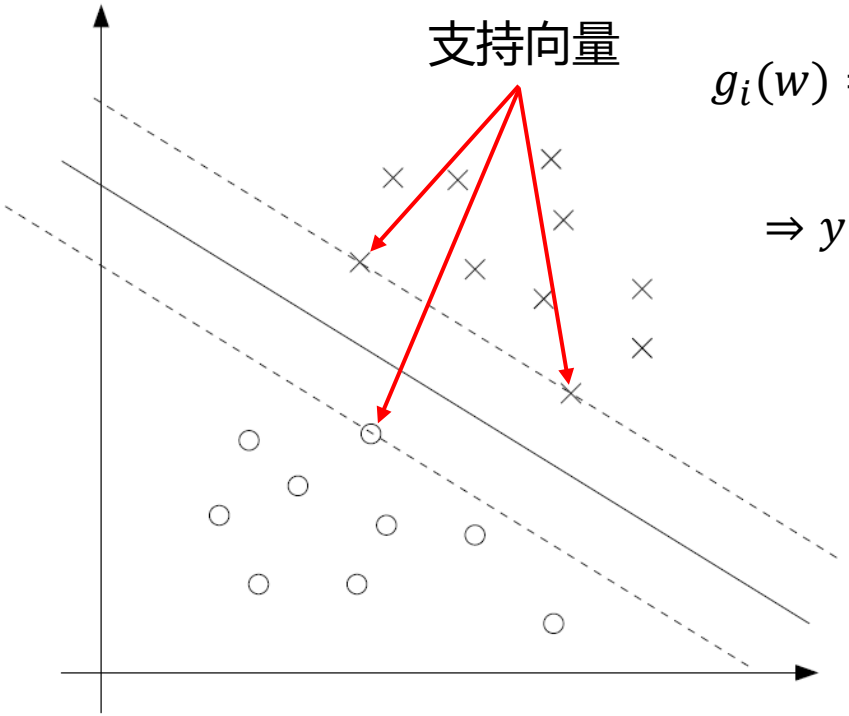
- $\frac{\partial}{\partial w_i}\mathcal{L}(w^*,\alpha^*,\beta^*) = 0, i = 1,\dots,n$
- $\frac{\partial}{\partial \beta_i}\mathcal{L}(w^*,\alpha^*,\beta^*) = 0, i = 1,\dots,l$
- $\alpha_i^* g_i(w^*) = 0, i = 1,\dots,k$ ← KKT对偶互补条件
- $g_i(w^*) \leq 0, i = 1,\dots,k$
- $\alpha^* \geq 0, i = 1,\dots,k$

Boyd, Stephen, and Lieven Vandenberghe. Convex optimization. Cambridge university press, 2004.
知乎KKT解释：https://zhuanlan.zhihu.com/p/26514613

支持向量机优化求解

等式情况

- 对于不等式约束条件，考虑等号成立的情况



$$g_i(w) = -y^{(i)}\left(w^T x^{(i)} + b\right) + 1 = 0$$

$$\Rightarrow y^{(i)}\left(\frac{w^T x^{(i)}}{\|w\|} + \frac{b}{\|w\|}\right) = \frac{1}{\|w\|}$$

几何间隔

当$g_i = 0$时，训练样本的函数间隔恰好等于1

支持向量机优化求解

目标函数

- 支持向量机的目标函数：寻找最优间隔分类器

$$\min_{w,b} \ \frac{1}{2}\|w\|^2$$
$$\text{s.t.} \quad -y^{(i)}\left(w^T x^{(i)} + b\right) + 1 \le 0, \qquad i = 1, \dots, m$$

- 拉格朗日函数

$$\mathcal{L}(w, b, \alpha) = \frac{1}{2}\|w\|^2 - \sum_{i=1}^{m} \alpha_i \left[y^{(i)}\left(w^T x^{(i)} + b\right) - 1\right]$$

  - 支持向量机中不存在β或者等式约束

支持向量机优化求解

问题求解

- 求解拉格朗日函数的极值点

$$\mathcal{L}(w,b,\alpha)=\frac{1}{2}\|w\|^2-\sum_{i=1}^{m}\alpha_i\left[y^{(i)}\left(w^Tx^{(i)}+b\right)-1\right]$$

  - 关于两个参数的偏导数

$$\frac{\partial}{\partial w}\mathcal{L}(w,b,\alpha)=w-\sum_{i=1}^{m}\alpha_i\,y^{(i)}x^{(i)}=0\Rightarrow w=\sum_{i=1}^{m}\alpha_i\,y^{(i)}x^{(i)}$$

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

  - 重写拉格朗日函数

$$\mathcal{L}(w,b,\alpha)=\frac{1}{2}\left\|\sum_{i=1}^{m}\alpha_i\,y^{(i)}x^{(i)}\right\|^2-\sum_{i=1}^{m}\alpha_i\,[y^{(i)}(\sum_{i=j}^{m}\alpha_j\,y^{(j)}x^{(j)^T}\cdot x^{(i)}+b)-1]$$

$$=\sum_{i=1}^{m}\alpha_i-\frac{1}{2}\sum_{i,j=1}^{m}y^{(i)}\,y^{(j)}\alpha_i\alpha_jx^{(i)^T}x^{(j)}-b\sum_{i=1}^{m}\alpha_i\,y^{(i)}=0$$

支持向量机优化求解

求解α^*

- 拉格朗日对偶问题

$$\max_{\alpha \geq 0} \theta_{\mathcal{D}}(\alpha) = \max_{\alpha \geq 0} \min_{w,b} \mathcal{L}(w, b, \alpha)$$

$$\max_{\alpha} \quad W(\alpha) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} y^{(i)} y^{(j)} \alpha_i \alpha_j x^{(i)T} x^{(j)}$$

$$\text{s.t.} \quad \alpha_i \geq 0, \qquad i = 1, \ldots, m$$

$$\sum_{i=1}^{m} \alpha_i y^{(i)} = 0$$

- 可以使用序列最小优化（SMO）算法对α^*进行求解

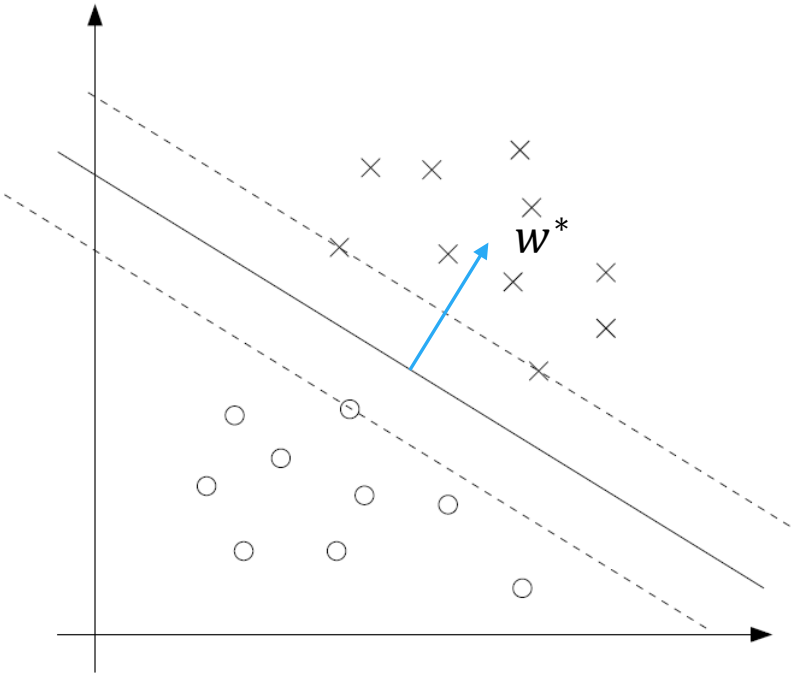
支持向量机优化求解

求解w^*和b^*

- 当求解得到α^*以后，w^*可以直接求解

$$w = \sum_{i=1}^{m} \alpha_i \, y^{(i)} x^{(i)}$$

  - 只有支持向量$\alpha > 0$



- 当求解得到w^*以后，b^*可以直接求解

$$b^* = -\frac{\max_{i:y^{(i)}=-1} w^{*T} x^{(i)} + \min_{i:y^{(i)}=1} w^{*T} x^{(i)}}{2}$$

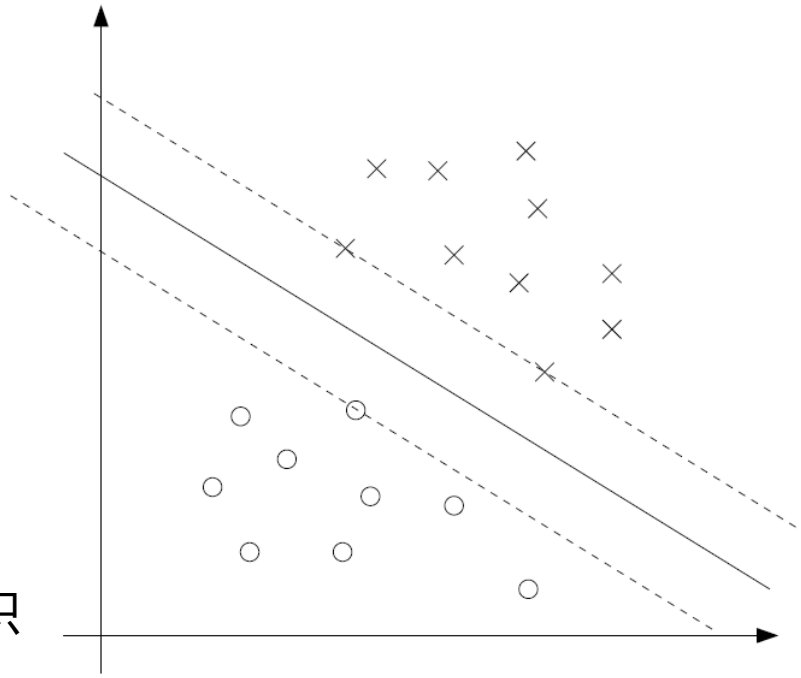
支持向量机优化求解

预测数值

- 当求解得到w^*和b^*以后，每个样例的预测数值（如：函数间隔）为

$$w^{*T}x + b^* = \left(\sum_{i=1}^{m} \alpha_i y^{(i)} x^{(i)}\right)^T x + b^*$$

$$= \sum_{i=1}^{m} \alpha_i y^{(i)} \langle x^{(i)}, x \rangle + b^*$$

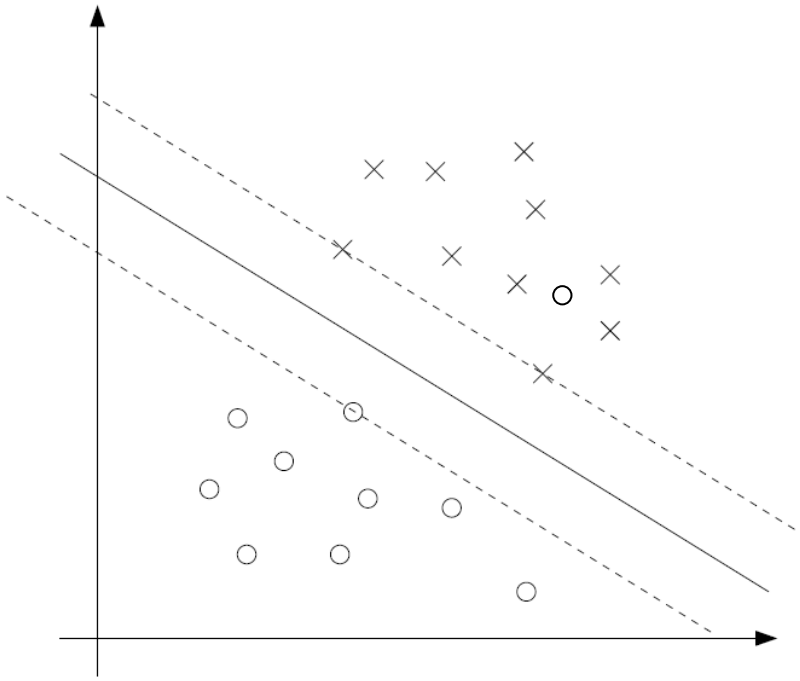


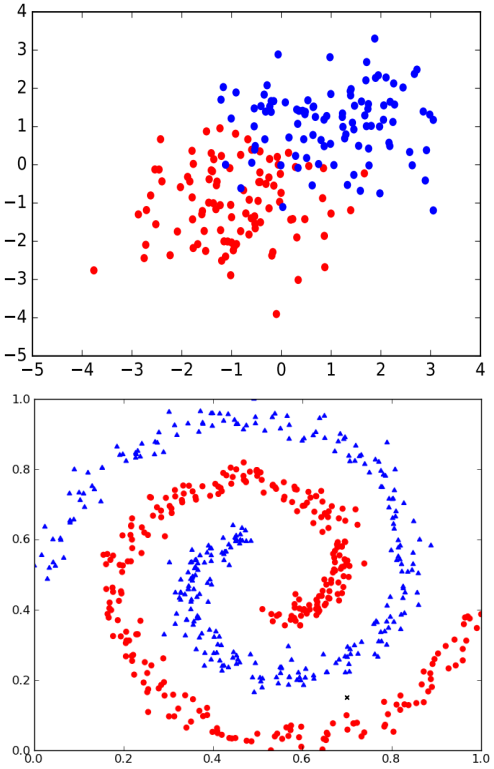
  - 只需要计算样例x与支持向量的内积

支持向量机优化求解

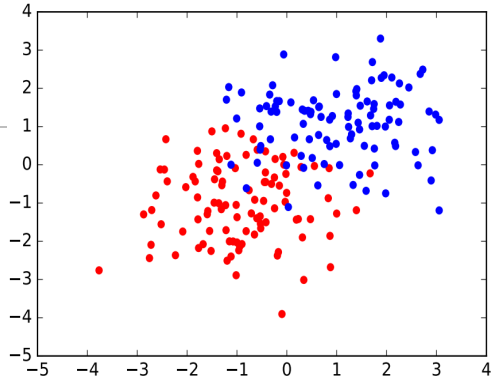
不可分情况

- 在之前支持向量机的推导过程中，数据被假定为线性可分的
- 应用场景中，数据往往是线性不可分的





支持向量机优化求解



处理不可分情况

- 增加松弛变量

$$\min_{w,b,\xi} \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{m}\xi_i$$

$$\text{s.t. } y^{(i)}\left(w^T x^{(i)} + b\right) \geq 1 - \xi_i, \qquad i = 1, \dots, m$$

$$\xi_i \geq 0, \qquad i = 1, \dots, m$$

- 拉格朗日函数

$$\mathcal{L}(w, b, \xi, \alpha, r) = \frac{1}{2}w^T w + C\sum_{i=1}^{m}\xi_i - \sum_{i=1}^{m}\alpha_i\left[y^{(i)}(x^T w + b) - 1 + \xi_i\right] - \sum_{i=1}^{m} r_i\,\xi_i$$

- 对偶问题

$$\max_{\alpha} W(\alpha) = \sum_{i=1}^{m}\alpha_i - \frac{1}{2}\sum_{i,j=1}^{m} y^{(i)}\,y^{(j)}\alpha_i\alpha_j {x^{(i)}}^T x^{(j)}$$

$$\text{s.t. } 0 \leq \alpha_i \leq C, \qquad i = 1, \dots, m$$

$$\sum_{i=1}^{m}\alpha_i\,y^{(i)} = 0$$

支持向量机优化求解

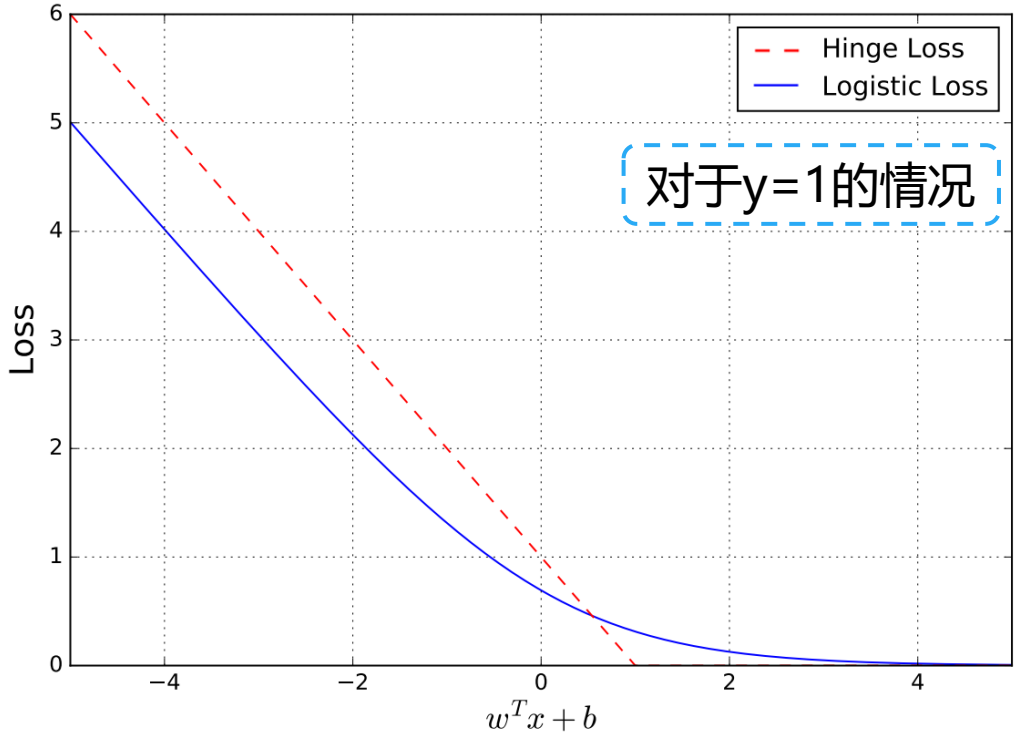
损失函数对比

支持向量机铰链损失（Hinge Loss）

$$\frac{1}{2}\|w\|^2 + C\sum_{i=1}^{m}\max\left(0,1-y_i(w^{\mathrm{T}}x_i+b)\right)$$

逻辑回归的对数损失

$$-y_i\log\sigma(w^Tx_i+b)-(1-y_i)\log\left(1-\sigma(w^Tx_i+b)\right)$$



序列最小优化算法

讲师：张伟楠 – 上海交通大学

序列最小优化算法

坐标上升法

- 对于优化问题

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

- 坐标上升法

循环直到收敛：{

对于$i = 1, \dots, m${

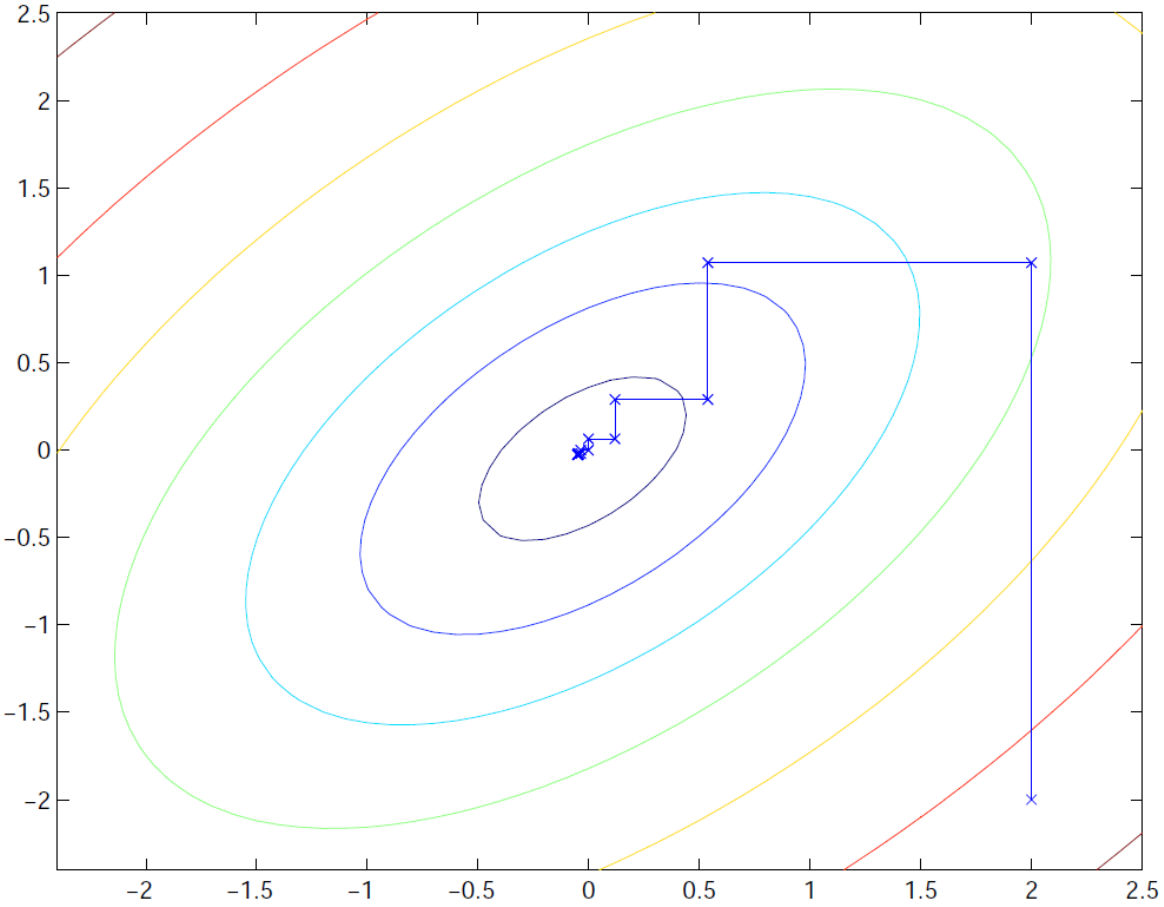
$$\alpha_i := \arg\max_{\hat{\alpha}_i} W(\alpha_1, \dots, \alpha_{i-1}, \hat{\alpha}_i, \alpha_{i+1}, \dots, \alpha_m)$$

}

}

序列最小优化算法

坐标上升法



二维坐标上升法样例

序列最小优化算法

序列最小优化（SMO）算法

- 支持向量机的优化问题

$$\max_{\alpha} W(\alpha) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} y^{(i)} y^{(j)} \alpha_i \alpha_j x^{(i)^T} x^{(j)}$$

$$\text{s.t.} \quad 0 \le \alpha_i \le C, \qquad i = 1, \dots, m$$

$$\sum_{i=1}^{m} \alpha_i y^{(i)} = 0$$

- 无法直接使用坐标上升法

$$\sum_{i=1}^{m} \alpha_i y^{(i)} = 0 \Rightarrow \alpha_i y^{(i)} = -\sum_{j \neq i} \alpha_j y^{(j)}$$

序列最小优化算法

序列最小优化（SMO）算法

- 每次优化两个变量

  循环直到收敛：{

  1. 选取一组变量α_i和α_j进行更新
  2. 以α_i和α_j为变量，对$W(\alpha)$进行再次优化

  }

- 收敛判别：当$W(\alpha)$的变化小于一个预设值，如：0.01

- 序列最小优化算法核心优势：更新变量α_i和α_j（步骤2）十分高效

序列最小优化算法

序列最小优化（SMO）算法

$$\max_{\alpha} W(\alpha) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} y^{(i)} y^{(j)} \alpha_i \alpha_j x^{(i)^T} x^{(j)}$$

支持向量机的优化问题

$$\text{s.t.} \quad 0 \le \alpha_i \le C, \qquad i = 1, \dots, m$$

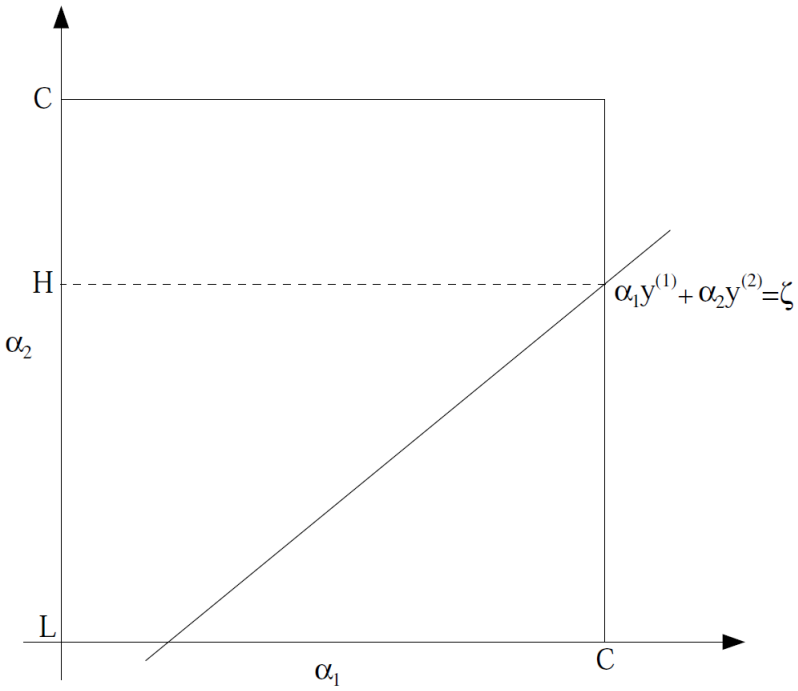
$$\sum_{i=1}^{m} \alpha_i y^{(i)} = 0$$

- 不失一般性，固定$\alpha_3, \dots, \alpha_m$，以$\alpha_1$和$\alpha_2$为变量，对$W(\alpha)$进行再次优化

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

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

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



序列最小优化算法

序列最小优化（SMO）算法

- 由$\alpha_1 = \left(\zeta - \alpha_2 y^{(2)}\right) y^{(1)}$，优化目标函数可以重写为

$$W(\alpha_1, \alpha_2, \ldots, \alpha_m) = W\left(\left(\zeta - \alpha_2 y^{(2)}\right) y^{(1)}, \alpha_2, \ldots, \alpha_m\right)$$

- 原始的优化问题

$$\max_{\alpha} \; W(\alpha) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} y^{(i)} y^{(j)} \alpha_i \alpha_j {x^{(i)}}^T x^{(j)}$$

$$\text{s.t.} \quad 0 \le \alpha_i \le C, \qquad i = 1, \ldots, m$$

$$\sum_{i=1}^{m} \alpha_i y^{(i)} = 0$$

  - 转化为以α_2为变量的二次优化问题

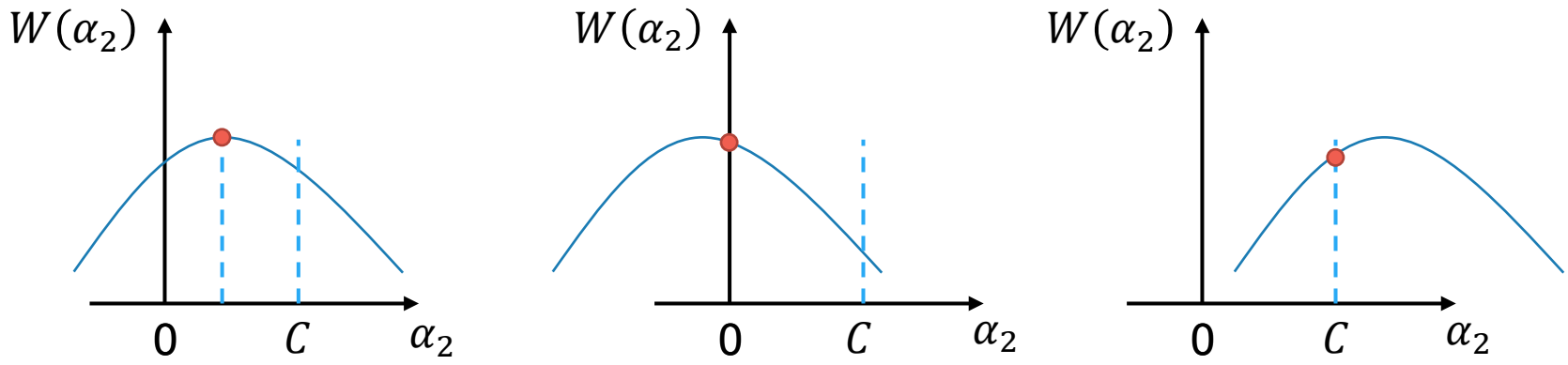
$$\max_{\alpha_2} \; W(\alpha_2) = a\alpha_2^2 + b\alpha_2 + c$$

$$\text{s.t.} \quad 0 \le \alpha_2 \le C$$

序列最小优化算法

序列最小优化（SMO）算法

- 二次函数的优化十分高效



$$\max_{\alpha_2} \ W(\alpha_2) = a\alpha_2^2 + b\alpha_2 + c$$

$$\text{s.t.} \ \ 0 \le \alpha_2 \le C$$

支持向量机核方法

讲师：张伟楠 – 上海交通大学

目录

Contents

01 支持向量机核方法

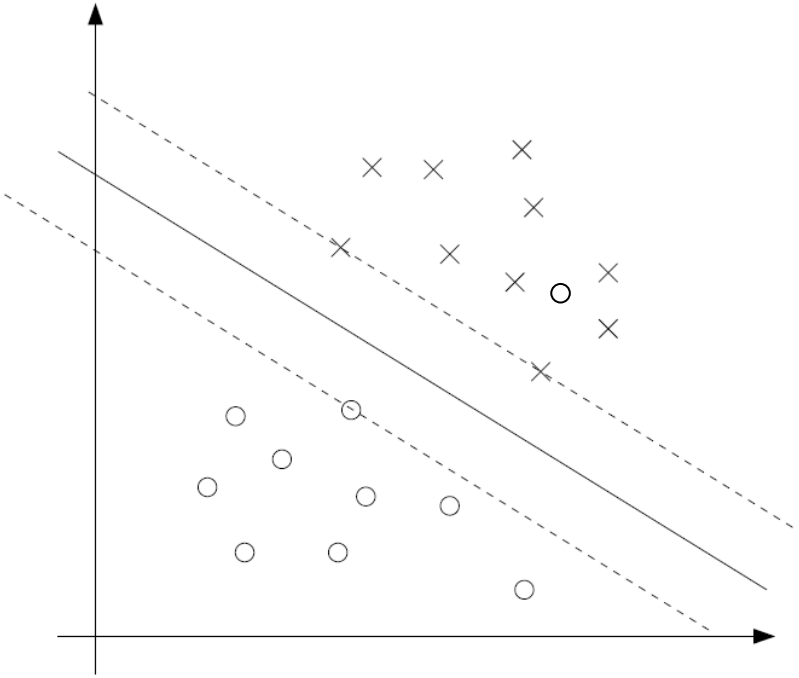
02 广义线性模型

01
支持向量机核方法

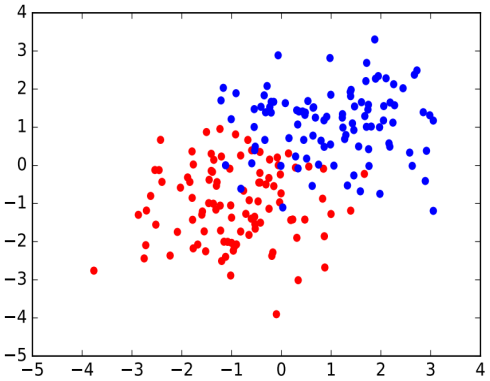
支持向量机核方法

不可分情况

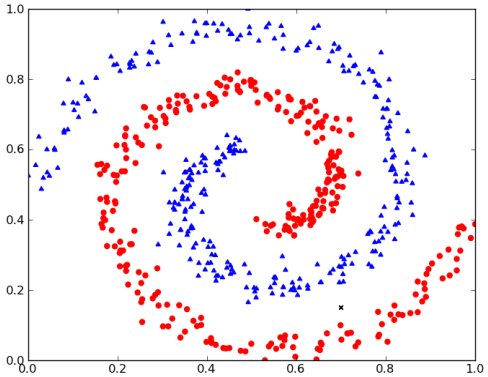
- 应用场景中，数据往往是线性不可分的



线性可分情况



通过松弛变量可解



无法通过松弛变量求解

支持向量机核方法

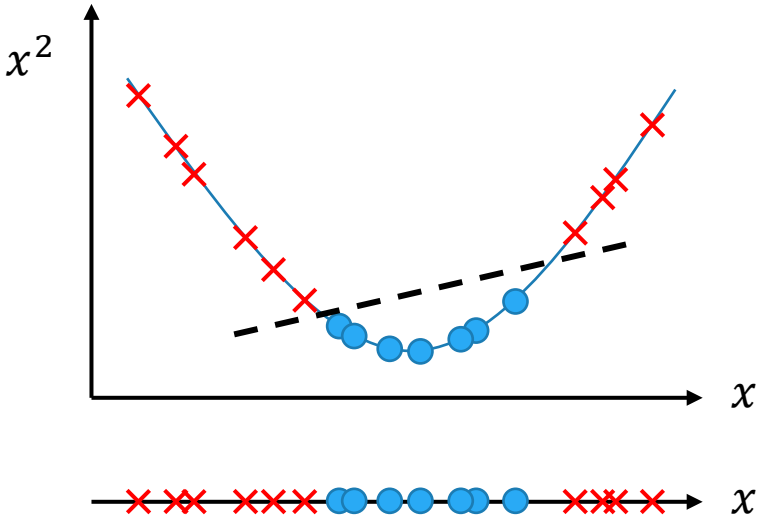
不可分情况

- 应用场景中，数据往往是线性不可分的
- 解决方案：将特征向量映射到高维空间中

$$\phi(x)$$

- 一个例子

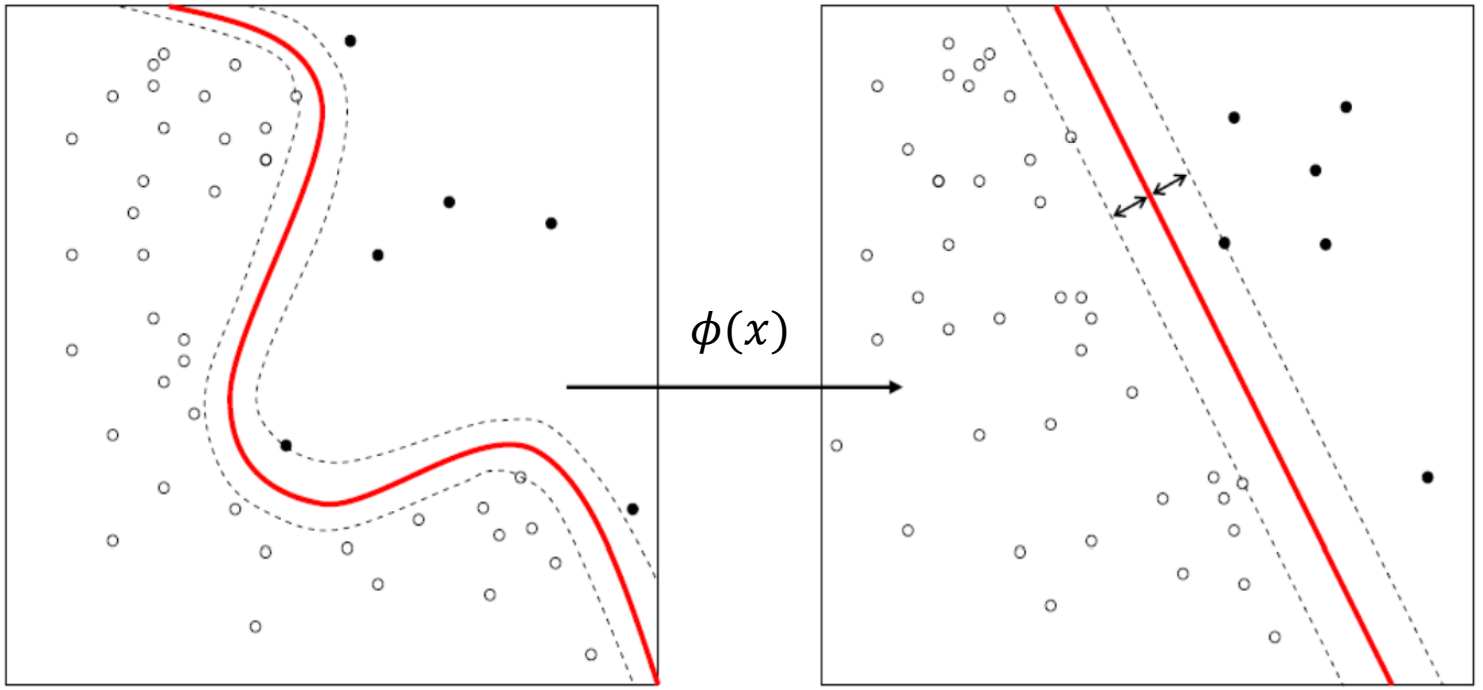
$$\phi(x) = (x, x^2)$$



支持向量机核方法

不可分情况

- 更广义地，将特征向量映射到不同空间中



支持向量机核方法

特征映射函数

- 基础的支持向量机只着眼于内积计算

$$W(\alpha) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} y^{(i)} y^{(j)} \alpha_i \alpha_j x^{(i)^T} x^{(j)}$$

- 定义特征映射函数$\phi(x)$

$$W(\alpha) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} y^{(i)} y^{(j)} \alpha_i \alpha_j K(x^{(i)}, x^{(j)})$$

- 核函数

$$K\left(x^{(i)}, x^{(j)}\right) = \phi\left(x^{(i)}\right)^T \phi(x^{(j)})$$

支持向量机核方法

核函数

- 对于特征映射函数

$$\phi(x) = \begin{bmatrix} x \\ x^2 \\ x^3 \end{bmatrix}$$

- 其对应核函数为

$$K\left(x^{(i)}, x^{(j)}\right) = \phi\left(x^{(i)}\right)^T \phi\left(x^{(j)}\right)$$
$$= x^{(i)}x^{(j)} + x^{(i)^2}x^{(j)^2} + x^{(i)^3}x^{(j)^3}$$

核技巧（Kernel Trick）

- 在多数情况下，可以直接定义$K\left(x^{(i)}, x^{(j)}\right)$，从而不需要显式定义$\phi\left(x^{(i)}\right)$
  - 例如，假定$x^{(i)}, x^{(j)} \in \mathbb{R}^n$，$K\left(x^{(i)}, x^{(j)}\right) = \left(x^{(i)^T} x^{(j)}\right)^2$

支持向量机核方法

训练和预测

- 给定核函数，α可以通过序列最小优化算法(SMO)求得

$$W(\alpha) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} y^{(i)} y^{(j)} \alpha_i \alpha_j K(x^{(i)}, x^{(j)})$$

- 当求解得到α 以后，w^*和b^*可以进行求解

$$w^* = \sum_{i=1}^{m} \alpha_i y^{(i)} \phi(x^{(i)})$$

w^*不需要被求解

$$b^* = -\frac{\max_{i:y^{(i)}=-1} w^{*T} \phi(x^{(i)}) + \min_{i:y^{(i)}=1} w^{*T} \phi(x^{(i)})}{2}$$

$$= -\frac{\max_{i:y^{(i)}=-1} \sum_{i=1}^{m} \alpha_i y^{(i)} K(x^{(i)}, x^{(j)}) + \min_{i:y^{(i)}=1} \sum_{i=1}^{m} \alpha_i y^{(i)} K(x^{(i)}, x^{(j)})}{2}$$

支持向量机核方法

训练和预测

- 当求解得到w^*和b^*以后，每个样例的预测数值（如：函数间隔）为

$$w^{*T}x + b^* = \left(\sum_{i=1}^{m} \alpha_i \, y^{(i)} \phi\left(x^{(i)}\right) \right)^T \phi(x) + b^*$$
$$= \sum_{i=1}^{m} \alpha_i \, y^{(i)} K\left(x^{(i)}, x\right) + b^*$$

- 假如预测数值为正，样例被预测为正例，反之亦然

注意：整个过程没有真正引入特征映射函数$\phi(\cdot)$的计算

支持向量机核方法

根据核函数反算映射函数

- 假定 $x^{(i)}, x^{(j)} \in \mathbb{R}^n$

$$K\left(x^{(i)}, x^{(j)}\right) = \left(x^{(i)^T} x^{(j)}\right)^2$$

$$= \left(\sum_{k=1}^{n} x_k^{(i)} x_k^{(j)}\right)\left(\sum_{l=1}^{n} x_l^{(i)} x_l^{(j)}\right)$$

$$= \sum_{k=1}^{n}\sum_{l=1}^{n} x_k^{(i)} x_k^{(j)} x_l^{(i)} x_l^{(j)} \quad \Rightarrow \quad \phi(x) = \begin{bmatrix} x_1x_1 \\ x_1x_2 \\ x_1x_3 \\ x_2x_1 \\ x_2x_2 \\ x_2x_3 \\ x_3x_1 \\ x_3x_2 \\ x_3x_3 \end{bmatrix}$$

$$= \sum_{k,l=1}^{n} \left(x_k^{(i)} x_l^{(i)}\right)\left(x_k^{(j)} x_l^{(j)}\right)$$

注意：计算$\phi(x)$需要$O(n^2)$的时间复杂度

然而计算$K\left(x^{(i)}, x^{(j)}\right)$仅仅需要$O(n)$的时间复杂度

支持向量机核方法

相似性度量

- 直观上对于x和z两个样例，如果$\phi(x)$和$\phi(z)$足够接近，我们希望

$$K(x,z) = \phi(x)^T\phi(z)$$

更大，反之亦然

- 高斯核函数（十分常用）

$$K(x,z) = \exp\left(-\frac{\|x-z\|^2}{2\sigma^2}\right)$$

  - 也被称为径向基函数（RBF）核
  - 那么，该核函数的特征映射函数是什么？

知乎回答：https://www.zhihu.com/question/35602879

支持向量机核方法

核矩阵

- 对于有限样例集合$\{x^{(1)},\dots,x^{(m)}\}$，其对应的核矩阵$K$定义为$\{K_{i,j}\}_{i,j=1,\dots,m}$
- 核矩阵K必定是对称矩阵，由于

$$K_{i,j}=K\left(x^{(i)},x^{(j)}\right)=\phi\left(x^{(i)}\right)^T\phi\left(x^{(j)}\right)=\phi\left(x^{(j)}\right)^T\phi\left(x^{(i)}\right)=K\left(x^{(j)},x^{(i)}\right)=K_{j,i}$$

  - 定义$\phi_k(x)$为向量$\phi(x)$的第k维坐标值，那么对于任何向量$z\in\mathbb{R}^m$

$$\begin{aligned}z^TKz&=\sum_i\sum_j z_i\,K_{ij}z_j\\&=\sum_i\sum_j z_i\,\phi\left(x^{(i)}\right)^T\phi\left(x^{(j)}\right)z_j=\sum_i\sum_j z_i\sum_k\phi_k\left(x^{(i)}\right)\phi_k\left(x^{(j)}\right)z_j\\&=\sum_k\sum_i\sum_j z_i\,\phi_k\left(x^{(i)}\right)\phi_k\left(x^{(j)}\right)z_j=\sum_k\left(\sum_i z_i\,\phi_k\left(x^{(i)}\right)\right)^2\geq 0\end{aligned}$$

- 因此，K为半正定矩阵

支持向量机核方法

有效核

James Mercer
英国数学家
1883-1932

- Mercer定理
  - 给定$K:\mathbb{R}^n \times \mathbb{R}^n \mapsto \mathbb{R}$，如果$K$为一个有效（Mercer）核，对于任意集合$\{x^{(1)},\dots,x^{(m)}\}, m<\infty$，其对应的核矩阵为对称半正定矩阵
- 有效核举例
  - RBF核 $K(x,z)=\exp\left(-\frac{\|x-z\|^2}{2\sigma^2}\right)$
  - 多项式核 $K(x,z)=(x^Tz)^d$
  - 余弦相似度核 $K(x,z)=\frac{x^Tz}{\|x\|\cdot\|z\|}$

支持向量机核方法

Sigmoid核

$$K(x,z) = \tanh(\alpha x^T z + c)$$

$$\tanh(b) = \frac{1 - e^{-2b}}{1 + e^{-2b}}$$

- 神经网络使用Sigmoid函数作为激活函数
- 使用Sigmoid核的支持向量机相似于一个二层的感知机
  - 但二层感知机还可以学习

Lin, Hsuan-Tien, and Chih-Jen Lin. "A study on sigmoid kernels for SVM and the training of non-PSD kernels by SMO-type methods." 2003.

02

泛线性模型（复习）

泛线性模型

线性回归

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}^{(1)} \\ \mathbf{x}^{(2)} \\ \vdots \\ \mathbf{x}^{(n)} \end{bmatrix} = \begin{bmatrix} x_1^{(1)} & x_2^{(1)} & x_3^{(1)} & \cdots & x_d^{(1)} \\ x_1^{(2)} & x_2^{(2)} & x_3^{(2)} & \cdots & x_d^{(2)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_1^{(n)} & x_2^{(n)} & x_3^{(n)} & \cdots & x_d^{(n)} \end{bmatrix} \quad \boldsymbol{\theta} = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \vdots \\ \theta_d \end{bmatrix} \mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}$$

- 预测

$$\hat{\mathbf{y}} = \mathbf{X}\boldsymbol{\theta} = \begin{bmatrix} \mathbf{x}^{(1)}\boldsymbol{\theta} \\ \mathbf{x}^{(2)}\boldsymbol{\theta} \\ \vdots \\ \mathbf{x}^{(n)}\boldsymbol{\theta} \end{bmatrix}$$

- 目标函数

$$J(\boldsymbol{\theta}) = \frac{1}{2}(\mathbf{y} - \hat{\mathbf{y}})^T(\mathbf{y} - \hat{\mathbf{y}}) = \frac{1}{2}(\mathbf{y} - \mathbf{X}\boldsymbol{\theta})^T(\mathbf{y} - \mathbf{X}\boldsymbol{\theta})$$

泛线性模型

线性回归矩阵形式

- 目标函数

$$J(\boldsymbol{\theta}) = \frac{1}{2}(\mathbf{y} - \mathbf{X}\boldsymbol{\theta})^T(\mathbf{y} - \mathbf{X}\boldsymbol{\theta}) \qquad \min_{\boldsymbol{\theta}} J(\boldsymbol{\theta})$$

- 梯度

$$\frac{\partial J(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = -\mathbf{X}^{\top}(\mathbf{y} - \mathbf{X}\boldsymbol{\theta})$$

- 求解

$$\frac{\partial J(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = 0 \rightarrow -\mathbf{X}^{\top}(\mathbf{y} - \mathbf{X}\boldsymbol{\theta}) = \mathbf{0}$$

$$\rightarrow \mathbf{X}^{\top}\mathbf{y} = \mathbf{X}^{\top}\mathbf{X}\boldsymbol{\theta}$$

$$\rightarrow \widehat{\boldsymbol{\theta}} = (\mathbf{X}^{\top}\mathbf{X})^{-1}\mathbf{X}^{\top}\mathbf{y}$$

泛线性模型

泛线性模型

映射关系

$$y = f(\theta^T \phi(x))$$

- 特征映射函数$\phi(x)$: $\mathbb{R}^d \mapsto \mathbb{R}^{\mathrm{h}}$
- 映射后的特征矩阵$\boldsymbol{\Phi}_{n\times h}$

$$\boldsymbol{\Phi} = \begin{bmatrix} \phi(x^{(1)}) \\ \phi(x^{(2)}) \\ \vdots \\ \phi(x^{(i)}) \\ \vdots \\ \phi(x^{(n)}) \end{bmatrix} = \begin{bmatrix} \phi_1(x^{(1)}) & \phi_2(x^{(1)}) & \cdots & \phi_h(x^{(1)}) \\ \phi_1(x^{(2)}) & \phi_2(x^{(2)}) & \cdots & \phi_h(x^{(2)}) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_1(x^{(i)}) & \phi_2(x^{(i)}) & \cdots & \phi_h(x^{(i)}) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_1(x^{(n)}) & \phi_2(x^{(n)}) & \cdots & \phi_h(x^{(n)}) \end{bmatrix}$$

泛线性模型

核线性回归矩阵形式

- 目标函数

$$J(\boldsymbol{\theta}) = \frac{1}{2}(\mathbf{y} - \boldsymbol{\Phi}\ \boldsymbol{\theta})^T(\mathbf{y} - \boldsymbol{\Phi}\ \boldsymbol{\theta}) \qquad \min_{\boldsymbol{\theta}} J(\boldsymbol{\theta})$$

- 梯度

$$\frac{\partial J(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = -\boldsymbol{\Phi}^T(\mathbf{y} - \boldsymbol{\Phi}\boldsymbol{\theta})$$

- 求解

$$\frac{\partial J(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = 0 \rightarrow -\boldsymbol{\Phi}^T(\mathbf{y} - \boldsymbol{\Phi}\boldsymbol{\theta}) = \mathbf{0}$$

$$\rightarrow\ \boldsymbol{\Phi}^T\mathbf{y} = \boldsymbol{\Phi}^T\boldsymbol{\Phi}\boldsymbol{\theta}$$

$$\rightarrow \widehat{\boldsymbol{\theta}} = (\boldsymbol{\Phi}^\top\boldsymbol{\Phi})^{-1}\boldsymbol{\Phi}^T\mathbf{y}$$

泛线性模型

核线性回归矩阵形式

- 通过矩阵运算的代数技巧

$$(\mathbf{P}^{-1}+\mathbf{B}^T\mathbf{R}^{-1}\mathbf{B})^{-1}\mathbf{B}^T\mathbf{R}^{-1}=\mathbf{P}\mathbf{B}^T(\mathbf{B}\mathbf{P}\mathbf{B}^T+\mathbf{R})^{-1}$$

- 使用L2范数作为正则项，最优参数为

$$\boldsymbol{P}=\frac{1}{\lambda}\boldsymbol{I}_{h\times h}\quad \boldsymbol{R}=\boldsymbol{I}_{n\times n}\quad \boldsymbol{B}=\boldsymbol{\Phi}_{n\times h}$$

$$\begin{aligned}\widehat{\boldsymbol{\theta}} &= (\boldsymbol{\Phi}^T\boldsymbol{\Phi}+\lambda\mathbf{I}_h)^{-1}\boldsymbol{\Phi}^T\mathbf{y}\\ &= \boldsymbol{\Phi}^T(\boldsymbol{\Phi}\boldsymbol{\Phi}^T+\lambda\mathbf{I}_n)^{-1}\mathbf{y}\end{aligned}$$

- 在预测时，我们不需要真正求出$\boldsymbol{\Phi}$

$$\begin{aligned}\hat{\mathbf{y}}=\boldsymbol{\Phi}\widehat{\boldsymbol{\theta}} &= \boldsymbol{\Phi}\boldsymbol{\Phi}^T(\boldsymbol{\Phi}\boldsymbol{\Phi}^T+\lambda\mathbf{I}_n)^{-1}\mathbf{y}\\ &= \mathbf{K}(\mathbf{K}+\lambda\mathbf{I}_n)^{-1}\mathbf{y}\end{aligned}$$

  - 其中，核矩阵为$\mathbf{K}=\{K(\boldsymbol{x}^{(i)},\boldsymbol{x}^{(j)})\}$

https://web2.qatar.cmu.edu/~gdicaro/10315-Fall19/additional/welling-notes-on-kernel-ridge.pdf

总结支持向量机

原始优化问题

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

$$\text{s.t.} \quad y^{(i)}\left(w^T x^{(i)} + b\right) \geq 1, \quad i = 1, \ldots, m$$

对偶优化问题

$$\max_{\alpha} \ W(\alpha) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} y^{(i)} y^{(j)} \alpha_i \alpha_j {x^{(i)}}^T x^{(j)}$$

$$\text{s.t.} \quad 0 \leq \alpha_i \leq C, \quad i = 1, \ldots, m$$

$$\sum_{i=1}^{m} \alpha_i y^{(i)} = 0$$

- 原始问题本身求解很方便
- 带线性不等式约束的凸优化问题
- 方便用标准求解器求解
- 参数化方法

- 对偶问题通过SMO算法建模成α的二次函数，快速求解
- 更方便直接用代码手写完成
- 直接导出了核技巧
- 非参数化方法

总结支持向量机

- 支持向量机是一种线性模型，优化目标是最大化决策边界距离数据点的最小距离，由此获得更好的分类鲁棒性
- 支持向量机的原问题可转化为一个二次函数优化问题，最终由序列最小优化算法来高效求解
- 当对原始特征数据做了映射变换，支持向量机可以被看成一个泛线性模型，而使用核方法可以让研究者仅仅关注定义两个数据点之间的相似性
- 支持向量机和逻辑回归的本质区别是什么？
- 基于统计的机器学习的本质思维方式是什么？

THANK YOU