支持向量机简介

Apple Zhang

Shenzhen University

2020年11月8日

Contents

Contents

符号定义

- 第i个训练样本: $\mathbf{x}_i = [x_i^{(1)}, x_i^{(2)}, \cdots, x_i^{(d)}]^T \in \mathbb{R}^d$.
- 第i个训练样本的标签: $y_i \in \{-1, 1\}$.
- 训练样本矩阵: $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n] \in \mathbb{R}^{d\times n}$.
- 训练样本标签向量: $\mathbf{y} = [y_1, y_2, \cdots, y_n] \in \{-1, 1\}^n$.
- 超平面的法向量: $\mathbf{w} \in \mathbb{R}^d$.
- 超平面的偏移: $b \in \mathbb{R}$
- L_2范数算子:

$$\|x\|_2 = \sqrt{\sum_{i=1}^{d} x_i^2} = \sqrt{\mathbf{x}^T\mathbf{x}}.$$

- 损失函数: $\mathcal{L}(\cdot)$.
- 拉格朗日函数: $L(\cdot)$.
- 求导算子: ∇f.

深圳大学 SHENZHEN UNIVERSITY

Contents

什么是超平面?

- 在$\mathbb{R}^2$空间中, 一条直线可以表示为

$$w_1 x^{(1)} + w_2 x^{(2)} + b = 0.$$

- 在$\mathbb{R}^3$空间中, 一个平面可以表示为

$$w_1 x^{(1)} + w_2 x^{(2)} + w_3 x^{(3)} + b = 0.$$

- $\cdots$
- 在$\mathbb{R}^d$空间中, 一个超平面可以表示为

$$w_1 x^{(1)} + w_2 x^{(2)} + \cdots + w_d x^{(d)} + b = 0.$$

即:

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

数据点到超平面的距离

- 在$\mathbb{R}^2$空间中，数据点$\mathbf{x}_i = [x_i^{(1)}, x_i^{(2)}]^T$到一条直线的距离为

$$\mathcal{D} = \frac{|w_1 x_i^{(1)} + w_2 x_i^{(2)} + b|}{\sqrt{w_1^2 + w_2^2}}.$$

- 在$\mathbb{R}^3$空间中，数据点$\mathbf{x}_i = [x_i^{(1)}, x_i^{(2)}, x_i^{(3)}]^T$到一个平面的距离是

$$\mathcal{D} = \frac{|w_1 x_i^{(1)} + w_2 x_i^{(3)} + w_2 x_i^{(3)} + b|}{\sqrt{w_1^2 + w_2^2 + w_3^2}}.$$

- $\cdots$
- 在$\mathbb{R}^d$空间中，数据点$\mathbf{x}_i = [x_i^{(1)}, x_i^{(2)}, \cdots, x_i^{(d)}]^T$到超平面的距离是

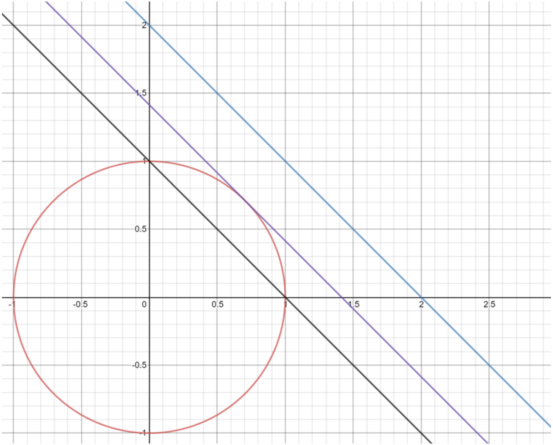
$$\begin{aligned}\mathcal{D} &= \frac{|w_1 x_i^{(1)} + w_2 x_i^{(2)} + \cdots + w_d x_i^{(d)} + b|}{\sqrt{w_1^2 + w_2^2 + \cdots + w_d^2}} \\ &= \frac{|\mathbf{w}^T \mathbf{x}_i + b|}{\|\mathbf{w}\|_2}.\end{aligned}$$

深圳大学 SHENZHEN UNIVERSITY

拉格朗日乘数法 - 举个栗子

如何解下面的优化问题？

$$\max_{x,y} \ x+y, \quad \text{s.t. } x^2+y^2=1.$$



拉格朗日乘数法 - 举个栗子

$$\max_{x,y} x + y, \quad \text{s.t. } x^2 + y^2 = 1.$$

解的关键是：相切！

$f(x,y) = x + y$和$g(x,y) = x^2 + y^2 - 1$的梯度平行的:

$$\begin{aligned}
\nabla f = \lambda \nabla g \ &\Rightarrow (\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}) = \lambda(\frac{\partial g}{\partial x}, \frac{\partial g}{\partial y}) \\
&\Rightarrow (1,1) = \lambda(2x, 2y) \\
&\Rightarrow x = \frac{1}{2\lambda}, \ y = \frac{1}{2\lambda}.
\end{aligned}$$

别忘了我们还有$x^2 + y^2 = 1$的条件, 因此可以解出

$$\lambda = \frac{1}{\sqrt{2}}, \ x = \frac{1}{\sqrt{2}}, \ y = \frac{1}{\sqrt{2}}.$$

拉格朗日乘数法

对于包含等式约束的优化问题:

$$\min_{X} f(X), \quad \text{s.t.} \ \ g(X) = 0.$$

我们可以构造一个叫做**拉格朗日函数**的辅助函数:

$$L(X, \lambda) = f(X) + \lambda g(X).$$

其中λ叫**拉格朗日乘子**。我们可以得到

$$\left\{ \min_{X,\lambda} L(X, \lambda) \right\} \Leftrightarrow \left\{ \min_{X} f(X), \quad \text{s.t.} \ \ g(X) = 0 \right\}.$$

也就是

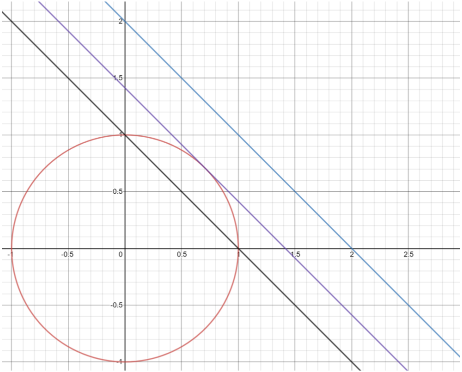
$$\frac{\partial L}{\partial X} = 0 \Rightarrow \frac{\partial f}{\partial X} + \lambda \frac{\partial g}{\partial X} = 0,$$
$$\frac{\partial L}{\partial \lambda} = 0 \Rightarrow g(X) = 0.$$

拉格朗日乘数法 - 又是个栗子

如果是这样呢？

$$\min_{x,y}\ x+y,\quad \text{s.t. } x^2+y^2=1,\ \boldsymbol{x}\geq -\mathbf{1}, \boldsymbol{y}\geq -\mathbf{1},$$

$$\min_{x,y}\ x+y,\quad \text{s.t. } x^2+y^2=1,\ \boldsymbol{x}\geq \mathbf{0}, \boldsymbol{y}\geq \mathbf{0}.$$



拉格朗日乘数法

对于包含不等式约束的优化问题

$$\min_{X} f(X), \quad \text{s.t. } g(X) = 0,\ h(X) \leq 0.$$

我们还是可以构造

$$L(X,\lambda) = f(X) + \lambda g(X) + \mu h(X)$$

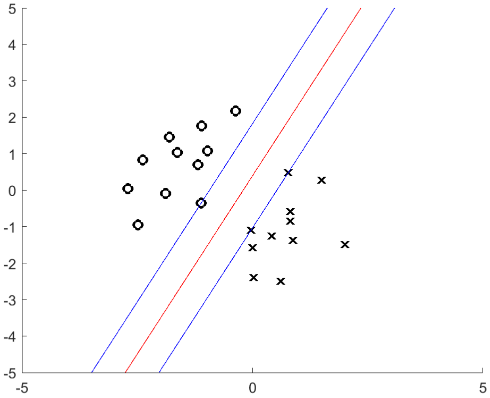
原问题的解需要满足KKT条件：

$$\frac{\partial L}{\partial X} = 0,\ \frac{\partial L}{\partial \lambda} = 0,$$
$$\mu h(X) = 0,\ \mu \geq 0,\ h(X) \leq 0.$$

如果$h(X) = 0$, 则$\frac{\partial L}{\partial \mu} = 0$;
如果$h(X) < 0$, 则$\mu = 0$.

问题构建

假设: 一个好的分类器应该最大化两个类间的间隔



问题构建

令两侧的超平面表示为

$$\mathbf{w}^T\mathbf{x} + b = -1, \ \ \mathbf{w}^T\mathbf{x} + b = 1.$$

所以可以得到这两个超平面中间的间隔是

$$\mathcal{D} = \frac{2}{\|\mathbf{w}\|_2}.$$

因此SVM最原始的优化问题定义为

$$\max_{\mathbf{w},b} \frac{2}{\|\mathbf{w}\|_2}, \ \text{s.t.} \begin{cases} \mathbf{w}^T\mathbf{x}_i + b \geq 1 & \text{if } y_i = 1 \\ \mathbf{w}^T\mathbf{x}_i + b \leq -1 & \text{if } y_i = -1 \end{cases}.$$

SVM的决策函数是

$$D(\mathbf{x}) = \text{sgn}(\mathbf{w}^T\mathbf{x} + b)$$

问题构建

又因为

$$\mathbf{w}^T\mathbf{x}_i + b \geq 1 \ \text{ or } \ \mathbf{w}^T\mathbf{x}_i + b \leq -1 \Leftrightarrow y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1,$$

以及

$$\max_{\mathbf{w},b} \frac{2}{\|\mathbf{w}\|_2} \Leftrightarrow \min_{\mathbf{w},b} \frac{1}{2}\|\mathbf{w}\|_2^2.$$

所以最终得到一个标准的优化形式

$$\min_{\mathbf{w},b} \frac{1}{2}\|\mathbf{w}\|_2^2, \ \text{s.t.} \ \ y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1.$$

分析

$$\min_{\mathbf{w},b} \frac{1}{2}\|\mathbf{w}\|_2^2, \text{ s.t. } y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1.$$

拉格朗日函数:

$$L(\mathbf{w}, b, \alpha) = \frac{1}{2}\|\mathbf{w}\|_2^2 + \sum_{i=1}^{n} \alpha_i \left[1 - y_i \left(\mathbf{w}^T\mathbf{x}_i + b\right)\right].$$

定理

上述*SVM*的优化问题等价于下面的对偶形式

$$\max_{\alpha} \min_{\mathbf{w},b} L(\mathbf{w}, b, \alpha), \text{ s.t. } \alpha \geq \mathbf{0}.$$

分析

简要解释：

$$\min_{\mathbf{w},b} \frac{1}{2}\|\mathbf{w}\|_2^2,\ \text{s.t.}\ \ y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1.$$

我们首先需要构造一个辅助函数$\theta(\mathbf{w}, b)$，且满足

$$\theta(\mathbf{w}, b) = \begin{cases} \frac{1}{2}\|\mathbf{w}\|_2^2 & \forall \mathbf{x}_i,\ 1 - y_i(\mathbf{w}^T\mathbf{x}_i + b) \leq 0 \\ \infty & \exists \mathbf{x}_i,\ 1 - y_i(\mathbf{w}^T\mathbf{x}_i + b) > 0 \end{cases}.$$

这样构造的原因是:

$$\min_{\mathbf{w},b} \frac{1}{2}\|\mathbf{w}\|_2^2,\ \text{s.t.}\ \ y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1$$
$$\Leftrightarrow \min_{\mathbf{w},b} \theta(\mathbf{w}, b)$$

深圳大学 SHENZHEN UNIVERSITY

分析

$$\theta(\mathbf{w}, b)=\begin{cases}\frac{1}{2}\|\mathbf{w}\|_2^2 & \forall \mathbf{x}_i,\ 1-y_i(\mathbf{w}^T\mathbf{x}_i+b)\leq 0\\ \infty & \exists \mathbf{x}_i,\ 1-y_i(\mathbf{w}^T\mathbf{x}_i+b)>0\end{cases}.$$

断言:

$$\theta(\mathbf{w}, b)=\max_{\alpha} L(\mathbf{w}, b, \alpha),\ \text{s.t. } \alpha \geq \mathbf{0}$$

. 解释: 对于这个优化问题，由于

$$h(\mathbf{w}, b)=\max_{\alpha}\ \frac{1}{2}\|\mathbf{w}\|_2^2+\sum_{i=1}^{n} \alpha_i\left[1-y_i\left(\mathbf{w}^T\mathbf{x}_i+b\right)\right],\ \text{s.t. } \alpha \geq \mathbf{0}$$

- $\forall \mathbf{x}_i,\ 1-y_i(\mathbf{w}^T\mathbf{x}_i+b)\leq 0 \Rightarrow h(\mathbf{w}, b)=\frac{1}{2}\|\mathbf{w}\|_2^2$
- $\exists \mathbf{x}_i,\ 1-y_i(\mathbf{w}^T\mathbf{x}_i+b)>0 \Rightarrow h(\mathbf{w}, b)=\infty$

因此, $\theta(\mathbf{w}, b)=h(\mathbf{w}, b)$

分析

记录一下, 我们现在得到了:

$$
\begin{aligned}
&\min_{\mathbf{w},b} \frac{1}{2}\|\mathbf{w}\|_2^2, \text{ s.t. } \ y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1 \\
\Leftrightarrow &\min_{\mathbf{w},b} \theta(\mathbf{w}, b) \\
\Leftrightarrow &\min_{\mathbf{w},b} \max_{\alpha} L(\mathbf{w}, b, \alpha), \text{ s.t. } \alpha \geq \mathbf{0}
\end{aligned}
$$

由于弱对偶性:

$$
\max_{Y} \min_{X} g(X, Y) \leq \min_{X} \max_{Y} g(X, Y)
$$

而在我们的问题里，等号是成立的（强对偶性）

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

深圳大学 SHENZHEN UNIVERSITY

分析

$$\max_{\alpha \geq \mathbf{0}} \min_{\mathbf{w},b} \frac{1}{2}\|\mathbf{w}\|_2^2 + \sum_{i=1}^{n} \alpha_i \left[1 - y_i \left(\mathbf{w}^T \mathbf{x}_i + b\right)\right].$$

首先对于$\mathbf{w}, b$最小化$L(\mathbf{w}, b, \alpha)$:

$$\frac{\partial L}{\partial \mathbf{w}} = 0 \Rightarrow \mathbf{w} = \sum_{i=1}^{n} \alpha_i y_i \mathbf{x}_i,$$

$$\frac{\partial L}{\partial b} = 0 \Rightarrow 0 = \sum_{i=1}^{n} \alpha_i y_i.$$

得到了

$$\max_{\alpha \geq \mathbf{0}} \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i y_i \alpha_j y_j \mathbf{x}_i^T \mathbf{x}_j, \quad \text{s.t.} \sum_{i=1}^{n} \alpha_i y_i = 0$$

分析

最后我们得到了一个二次规划问题：

$$\begin{aligned} &\max_{\alpha} \mathbf{1}^T\alpha - \frac{1}{2}\alpha^T\mathbf{Q}\alpha, \\ &\text{s.t. } \alpha \geq \mathbf{0},\ \mathbf{y}^T\alpha = 0. \end{aligned} \tag{1}$$

这里$\mathbf{1}$是全1的列向量，$\mathbf{Q} \in \mathbb{R}^{n\times n}$的每一个元素是

$$\mathbf{Q}_{ij} = y_i y_j \mathbf{x}_i^T \mathbf{x}_j.$$

这个优化问题有很多算法来解决. (Matlab function: `quadprog`) 最常用的是序列最小化算法 (Sequential Minimal Optimization, SMO).
最后，我们可以把决策函数写为

$$D(\mathbf{x}) = \mathrm{sgn}(\sum_{i=1}^{n} \alpha_i y_i \mathbf{x}_i^T \mathbf{x} + b)$$

小贴士

$$
\begin{aligned}
&\max_{\alpha} \mathbf{1}^T \alpha - \frac{1}{2} \alpha^T \mathbf{Q} \alpha, \\
&\text{s.t. } \alpha \geq \mathbf{0},\ \mathbf{y}^T \alpha = 0.
\end{aligned}
$$

- 上式解出的α是稀疏的. 其中对应的训练样本$\mathbf{x}_i$称为支持向量(**Support Vector**, SV), 如果它对应的$\alpha_i > 0$.
- b有很多种计算方法. 在SMO算法中, b是“顺便”被算出来的. 其他情况下, b可以用下面的公式算出:

$$
b = \frac{1}{\|\alpha\|_0} \sum_{\alpha_i > 0} \left(\frac{1}{y_i} - \sum_{j=1}^{n} \alpha_j y_j \mathbf{x}_j^T \mathbf{x}_i \right)
$$

其中$\|\cdot\|_0$表示L_0范数算子, 返回向量非零元素个数.

小贴士

$$\max_{\alpha} \mathbf{1}^T\alpha - \frac{1}{2}\alpha^T\mathbf{Q}\alpha,$$
$$\text{s.t. } \alpha \geq \mathbf{0},\ \mathbf{y}^T\alpha = 0. \tag{2}$$

Matlab函数：`x = quadprog(H, f, A, b, Aeq, Beq, lb, ub, x0);`

$$\min_{\mathbf{x}} \frac{1}{2}\mathbf{x}^T\mathbf{H}\mathbf{x} + \mathbf{f}^T\mathbf{x},\ \text{s.t.} \begin{cases} \mathbf{A}\cdot\mathbf{x} \leq \mathbf{b} \\ \mathbf{Aeq}\cdot\mathbf{x} = \mathbf{beq} \\ lb \leq \mathbf{x} \leq ub \end{cases}$$

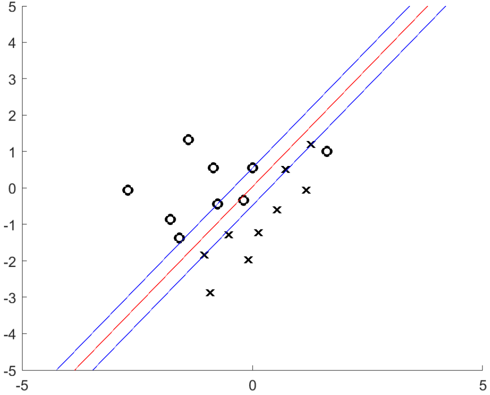
因此解SVM的二次规划问题可以按以下方式调用：

$$\texttt{alpha = quadprog}(\mathbf{Q},\ -\mathbf{1},\ \texttt{[]},\ \texttt{[]},\ \mathbf{y}^T,\ 0,\ \mathbf{0},\ \texttt{[]},\ \mathbf{x}_0\texttt{)}$$

注意：需要安装Matlab Optimization toolbox!

令人困惑的决策边界

上面讨论的是**硬间隔SVM.**
硬间隔SVM: 不能犯任何错误! 因此对异常数据点很敏感!
因此就有了**软间隔SVM.**



软间隔SVM

模型建构:

$$\min_{\mathbf{w},b} \frac{1}{2}\|\mathbf{w}\|_2^2 + C\sum_{i=1}^{n}\mathcal{L}_{\text{hinge}}(y_i(\mathbf{w}^T\mathbf{x}_i + b)), \text{ s.t. } \exists i,\ y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1$$

此处C是惩罚参数, $\mathcal{L}_{\text{hinge}}(\cdot)$称为**铰链损失**, 其定义为

$$\mathcal{L}_{\text{hinge}}(z) = \max(0, 1 - z).$$

铰链损失度量了分类的误差ξ_i

$$\xi_i = \left(1 - y_i(\mathbf{w}^T\mathbf{x}_i + b)\right)_+,$$

这个误差参数称为**松弛变量**.

软间隔SVM

重写一下目标函数

$$\min_{\mathbf{w},b,\xi} \frac{1}{2}\|\mathbf{w}\|_2^2 + C\sum_{i=1}^{n}\xi_i,$$
$$\text{s.t. } y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1 - \xi_i,\ \xi_i \geq 0$$

拉格朗日函数:

$$L(\mathbf{w}, b, \xi, \alpha, \beta) = \frac{1}{2}\|\mathbf{w}\|_2^2 + C\sum_{i=1}^{n}\xi_i + \sum_{i=1}^{n}\left[\alpha_i\left(1 - \xi_i - y_i(\mathbf{w}^T\mathbf{x}_i + b)\right)\right] - \sum_{i=1}^{n}\beta_i\xi_i$$

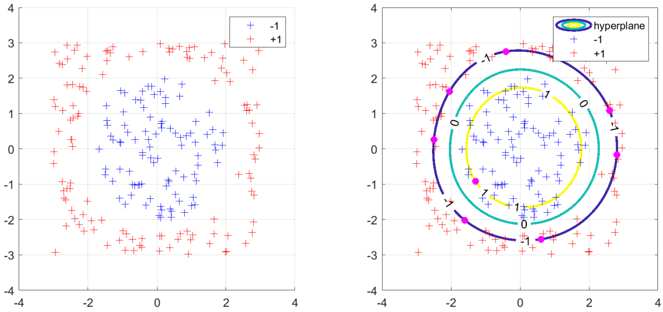
对偶问题:

$$\max_{\alpha} \mathbf{1}^T\alpha - \frac{1}{2}\alpha^T\mathbf{Q}\alpha,$$
$$\text{s.t. } \mathbf{0} \leq \alpha \leq C\mathbf{1},\ \mathbf{y}^T\alpha = 0.$$

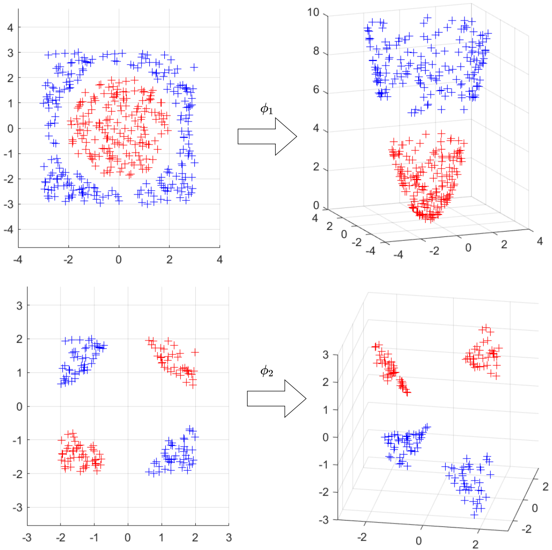
可以发现, 硬间隔SVM是软间隔SVM当$C \to \infty$的特殊情况.

非线性分布的数据

之前的讨论基于一个假设: 数据是线性可分的. 当数据呈现非线性分布时, 效果可能就比较差.
我们需要**非线性SVM** (或核SVM).



非线性分布的数据



核方法

假设有一个特征映射:

$$\phi(\mathbf{x}) : \mathbb{R}^d \mapsto \mathcal{H}$$

这个映射唯一对应了一个核函数（核函数需要满足Mercer定理）

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

假设: 在映射后的特征空间$\mathcal{H}$中, 数据是线性可分的.
决策边界:

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

而且$\mathbf{w} \in \mathcal{H}$.
新的优化问题:

$$\min_{\mathbf{w},b} \frac{1}{2}\|\mathbf{w}\|_{\mathcal{H}}^2 + \sum_{i=1}^{n} \xi_i$$

$$\text{s.t. } y_i(\mathbf{w}^T \phi(\mathbf{x}_i) + b) \geq 1 - \xi_i$$

深圳大学 SHENZHEN UNIVERSITY

核方法

拉格朗日函数:

$$L(\mathbf{w}, b, \xi, \alpha, \beta) = \frac{1}{2}\|\mathbf{w}\|_{\mathcal{H}}^2 + C\sum_{i=1}^{n}\xi_i + \sum_{i=1}^{n}\left[\alpha_i\left(1 - \xi_i - y_i(\mathbf{w}^T\phi(\mathbf{x}_i) + b)\right)\right] - \sum_{i=1}^{n}\beta_i\xi_i$$

$$\frac{\partial L}{\partial \mathbf{w}} = 0 \Rightarrow \mathbf{w} = \sum_{i=1}^{n}\alpha_i y_i \phi(\mathbf{x}_i), \quad \frac{\partial L}{\partial b} = 0 \Rightarrow 0 = \sum_{i=1}^{n}\alpha_i y_i,$$
$$\frac{\partial L}{\partial \xi} = 0 \Rightarrow 0 = \sum_{i=1}^{n}(C\xi_i - \alpha_i - \beta_i)\,.$$

从这里就可以知道核SVM的决策函数是

$$D(\mathbf{x}) = \mathrm{sgn}(\mathbf{w}^T\phi(\mathbf{x}) + b) = \mathrm{sgn}(\sum_{i=1}^{n}\alpha_i y_i \mathcal{K}(\mathbf{x}_i, \mathbf{x}) + b)$$

核方法

对偶问题:

$$\max_{\alpha} \mathbf{1}^T\alpha - \frac{1}{2}\alpha^T\mathbf{Q}^{\Phi}\alpha,$$
$$\text{s.t. } \mathbf{0} \leq \alpha \leq C\mathbf{1},\ \mathbf{y}^T\alpha = 0.$$

其中$\mathbf{Q}^{\Phi} \in \mathbb{R}^{n\times n}$的每一个元素是:

$$\mathbf{Q}^{\Phi}_{ij} = y_i y_j \phi(\mathbf{x}_i)^T\phi(\mathbf{x}_j) = y_i y_j \mathcal{K}(\mathbf{x}_i, \mathbf{x}_j)$$

关键: 也许映射$\phi(\cdot)$的表达式是未知的, 但它对应的核函数$\mathcal{K}(\cdot,\cdot)$是已知的
e.g. RBF核:

$$\mathcal{K}(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(\frac{-\|\mathbf{x}_i - \mathbf{x}_j\|_2^2}{4t}\right)$$

更多有关SVM...

高效的SVM解决方案(C++实现): **LIBSVM, LIBLINEAR**

- 已有C++, Java, MATLAB, Python接口.
- 贼快.

其他SVM的变种:

- (1999) Least square support vector machine (LSSVM).
- (2003 **NIPS**) L_1-regularized support vector machine (L_1-SVM).
- (2006 **JMLR**) Laplacian support vector machine.
- (2007 **TPAMI**) Twin support vector machine (TWSVM).
- (2011 **TNNLS**) Twin bound support vector machine (TBSVM).
- (2012 **Neural Networks**) Laplacian twin support vector machine.
- (2019 **Machine Learning**) Robst twin support vector machine.
- ...

恭喜你撑下来了

谢谢!