Soft Margin SVM

一、Primal Soft Margin SVM

我们使用Gaussian kernel的时候over fitting的原因可能是因为参数选的不好，也有可能是我们的限制条件太苛刻——我们可以适当放宽条件，允许一些错误，于是这就从hard-margin转化到了soft-margin，借助pocket算法找到灵感，也就是找到犯错最少的而不是不犯错的模型

want: give up on some noisy examples

pocket

$$\min_{b,\mathbf{w}} \sum_{n=1}^{N} [\![y_n \neq \text{sign}(\mathbf{w}^T\mathbf{z}_n + b)]\!]$$

hard-margin SVM

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

$$\text{s.t.} \quad y_n(\mathbf{w}^T\mathbf{z}_n + b) \geq 1 \text{ for all } n$$

combination:

$$\min_{b,\mathbf{w}} \frac{1}{2}\mathbf{w}^T\mathbf{w} + C \cdot \sum_{n=1}^{N} [\![y_n \neq \text{sign}(\mathbf{w}^T\mathbf{z}_n + b)]\!]$$

$$\text{s.t.} \quad y_n(\mathbf{w}^T\mathbf{z}_n + b) \geq 1 \text{ for correct } n$$

$$y_n(\mathbf{w}^T\mathbf{z}_n + b) \geq -\infty \text{ for incorrect } n$$

我们引入C来表示**large margin**和**noise tolerance**之间的相对重要性。

但是存在两个问题：

1. 不是linear的问题，也就是不是QP问题我们就没有办法解决了。
2. 无法分辨错误的程度，也就是对于错误的类型没有一个定量的认知。

我们引入新的模型，使用变量ξ_n记录错误程度：

- $[\![\cdot]\!]$: non-linear, not QP anymore :-(
  —what about dual? kernel?
- cannot distinguish **small error** (slightly away from fat boundary)
  or **large error** (a...w...a...y... from fat boundary)

- record 'margin violation' by ξ_n—linear constraints
- penalize with margin violation instead of error count
  —quadratic objective

soft-margin SVM:

$$\min_{b,\mathbf{w},\xi} \frac{1}{2}\mathbf{w}^T\mathbf{w} + C \cdot \sum_{n=1}^{N} \xi_n$$

$$\text{s.t.} \quad y_n(\mathbf{w}^T\mathbf{z}_n + b) \geq 1 - \xi_n \text{ and } \xi_n \geq 0 \text{ for all } n$$

C用来在large margin和margin violation之间作权衡：

- parameter C: trade-off of large margin & margin violation
  - large C: want less margin violation
  - small C: want large margin
- QP of $\tilde{d} + 1 + N$ variables, $2N$ constraints

二、Dual SVM

引入lagrange因子来转化为Dual Problem:

$$\text{primal:} \quad \min_{b,\mathbf{w},\xi} \quad \frac{1}{2}\mathbf{w}^T\mathbf{w} + C \cdot \sum_{n=1}^{N} \xi_n$$

$$\text{s.t.} \quad y_n(\mathbf{w}^T\mathbf{z}_n + b) \geq 1 - \xi_n \text{ and } \xi_n \geq 0 \text{ for all } n$$

Lagrange function with Lagrange multipliers α_n and β_n

$$\mathcal{L}(b, \mathbf{w}, \xi, \alpha, \beta) = \frac{1}{2}\mathbf{w}^T\mathbf{w} + C \cdot \sum_{n=1}^{N} \xi_n + \sum_{n=1}^{N} \alpha_n \cdot (1 - \xi_n - y_n(\mathbf{w}^T\mathbf{z}_n + b)) + \sum_{n=1}^{N} \beta_n \cdot (-\xi_n)$$

want: Lagrange dual

$$\max_{\alpha_n \geq 0,\ \beta_n \geq 0} \left(\min_{b,\mathbf{w},\xi} \quad \mathcal{L}(b, \mathbf{w}, \xi, \alpha, \beta) \right)$$

利用KKT condition简化问题!

先对ξ_n进行求导，我们有如下:

Simplify ξ_n and β_n

$$\max_{\alpha_n \geq 0,\ \beta_n \geq 0} \left(\min_{b,\mathbf{w},\xi} \quad \frac{1}{2}\mathbf{w}^T\mathbf{w} + C \cdot \sum_{n=1}^{N} \xi_n + \sum_{n=1}^{N} \alpha_n \cdot \left(1 - \xi_n - y_n(\mathbf{w}^T\mathbf{z}_n + b)\right) + \sum_{n=1}^{N} \beta_n \cdot (-\xi_n) \right)$$

- $\frac{\partial \mathcal{L}}{\partial \xi_n} = 0 = C - \alpha_n - \beta_n$
- no loss of optimality if solving with implicit constraint $\beta_n = C - \alpha_n$ and explicit constraint $0 \leq \alpha_n \leq C$: β_n removed

ξ can also be removed :-), like how we removed b

$$\max_{0 \leq \alpha_n \leq C,\ \beta_n = C - \alpha_n} \left(\min_{b,\mathbf{w},\xi} \quad \frac{1}{2}\mathbf{w}^T\mathbf{w} + \sum_{n=1}^{N} \alpha_n\left(1 - y_n(\mathbf{w}^T\mathbf{z}_n + b)\right) \cancel{+ \sum_{n=1}^{N} (C - \alpha_n - \beta_n) \cdot \xi_n} \right)$$

$$C = \alpha_n + \beta_n \tag{1}$$

由于我们有$\beta_n \geq 0$所以就得到了

$$0 \leq \alpha_n \leq C \tag{2}$$

然后我们也把ξ干掉了

再对b, w进行求导，其实和之前的推导类似：

- inner problem **same as hard-margin SVM**
- $\frac{\partial \mathcal{L}}{\partial b} = 0$: no loss of optimality if solving with constraint $\sum_{n=1}^{N} \alpha_n y_n = 0$
- $\frac{\partial \mathcal{L}}{\partial w_i} = 0$: no loss of optimality if solving with constraint $\mathbf{w} = \sum_{n=1}^{N} \alpha_n y_n \mathbf{z}_n$

最终问题划归为如下问题：

$$\min_{\alpha} \quad \frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{N}\alpha_n\alpha_m y_n y_m \mathbf{z}_n^T\mathbf{z}_m - \sum_{n=1}^{N}\alpha_n$$

$$\text{subject to} \quad \sum_{n=1}^{N} y_n\alpha_n = 0;$$

$$0 \le \alpha_n \le C, \text{for } n = 1, 2, \ldots, N;$$

$$\text{implicitly} \quad \mathbf{w} = \sum_{n=1}^{N}\alpha_n y_n \mathbf{z}_n;$$

$$\beta_n = C - \alpha_n, \text{for } n = 1, 2, \ldots, N$$

—only difference to hard-margin: upper bound on α_n

another (convex) QP,
with N variables & $2N+1$ constraints

N variables, 2N+1 constraints!

三、求解Dual Soft-SVM

Kernel Soft-Margin SVM

Kernel Soft-Margin SVM Algorithm

1. $q_{n,m} = y_n y_m K(\mathbf{x}_n, \mathbf{x}_m)$; $\mathbf{p} = -\mathbf{1}_N$; $(\mathrm{A}, \mathbf{c})$ for equ./lower-bound/**upper-bound** constraints
2. $\alpha \leftarrow \mathrm{QP}(\mathrm{Q_D}, \mathbf{p}, \mathrm{A}, \mathbf{c})$
3. $b \leftarrow ?$
4. return SVs and their α_n as well as b such that for new $\mathbf{x}$,

$$g_{\mathrm{SVM}}(\mathbf{x}) = \mathrm{sign}\left(\sum_{\text{SV indices } n}\alpha_n y_n K(\mathbf{x}_n, \mathbf{x}) + b\right)$$

- almost the same as hard-margin
- more flexible than hard-margin
  —**primal/dual always solvable**

remaining question: step 3?

这里我们来单独说一下b的求解，和hard margin是否一样呢，之前hard margin使用complimentary slackness

hard-margin SVM

complementary slackness:

$$\alpha_n(1 - y_n(\mathbf{w}^T\mathbf{z}_n + b)) = 0$$

- SV ($\alpha_s > 0$)
  $\Rightarrow b = y_s - \mathbf{w}^T\mathbf{z}_s$

soft-margin SVM

complementary slackness:

$$\alpha_n(1 - \xi_n - y_n(\mathbf{w}^T\mathbf{z}_n + b)) = 0$$
$$(C - \alpha_n)\xi_n = 0$$

- SV ($\alpha_s > 0$)
  $\Rightarrow b = y_s - y_s\xi_s - \mathbf{w}^T\mathbf{z}_s$
- free ($\alpha_s < C$)
  $\Rightarrow \xi_s = 0$

solve unique b with free SV $(\mathbf{x}_s, y_s)$:

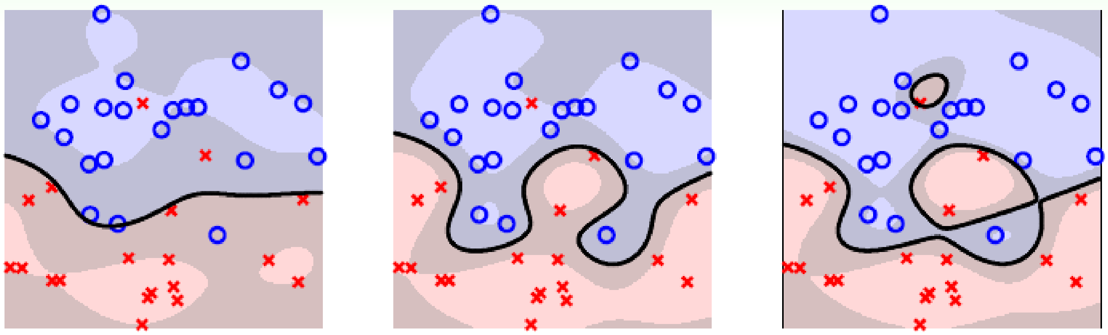
$$b = y_s - \sum_{\text{SV indices } n} \alpha_n y_n K(\mathbf{x}_n, \mathbf{x}_s)$$

—range of b otherwise

找一个 free SV（一般情况下都会有free SV)

看一下Soft-Margin SVM,也有可能over fitting

Soft-Margin Gaussian SVM in Action

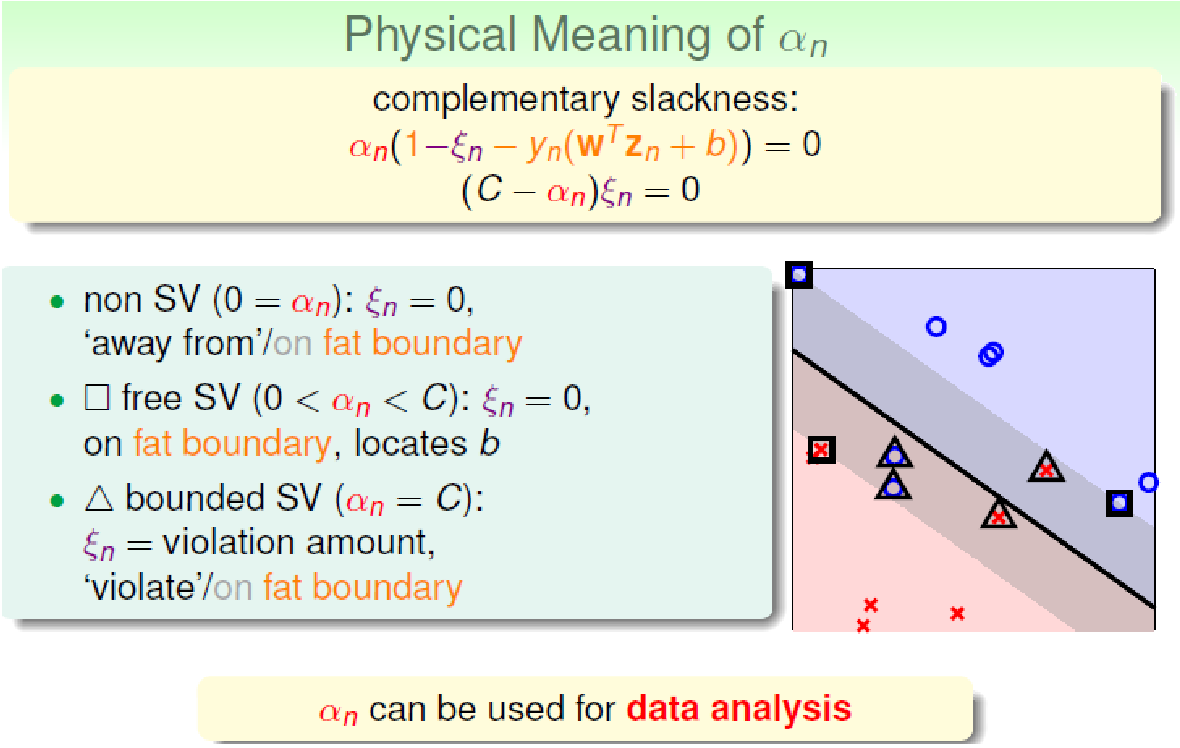


$C = 1$　　　$C = 10$　　　$C = 100$

- large $C \Longrightarrow$ less noise tolerance $\Longrightarrow$ 'overfit'?
- **warning: SVM can still overfit :-(**

soft-margin Gaussian SVM:
need careful selection of (γ, C)

试图解释一下这个模型：



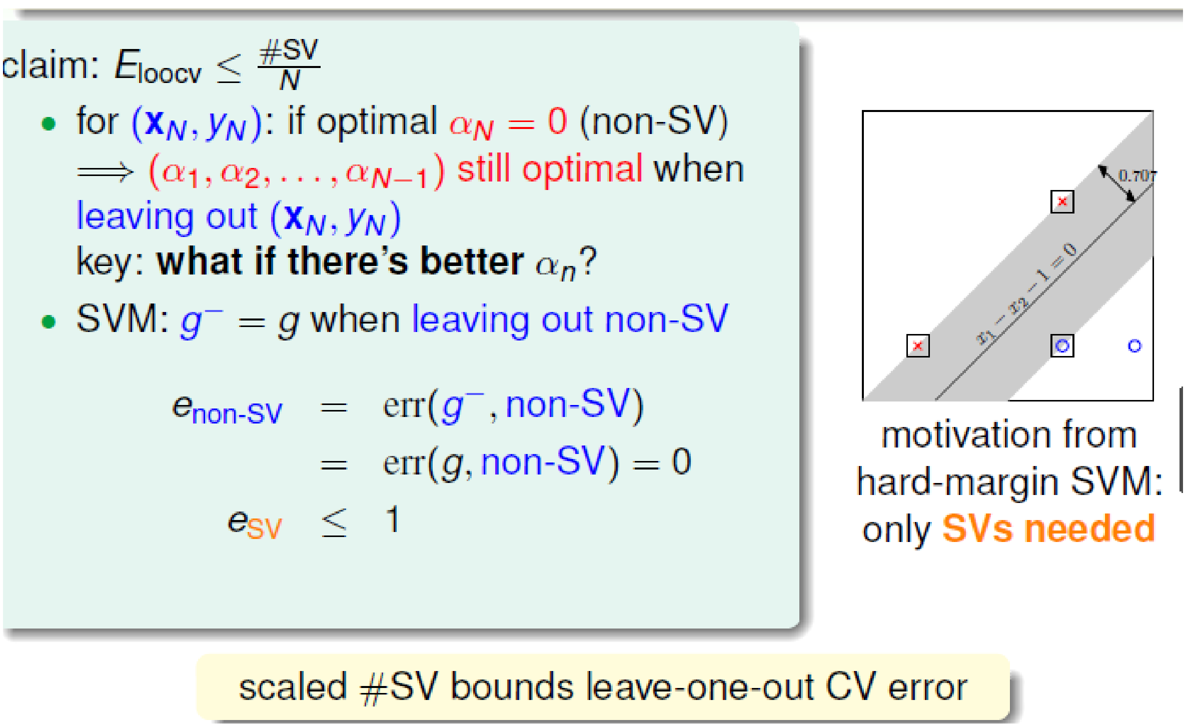
- free SV: 正好在边界上
- non SV: 在边界外
- bounded SV：在边界内（违反但分类正确，违反且分类错误）唯一有可能犯错的点!

所以，α_n可以用来分析这个模型的数据

四、模型的选择

使用validation方法来选择

对于SVM的E_{loocv}，我们有一些特别的结论：



non-SV对于最佳解g没有什么影响!

所以SV数量可以作为safety-check!