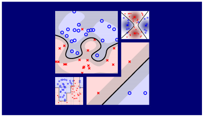
Machine Learning Techniques
(機器學習技法)



Lecture 4: Soft-Margin Support Vector Machine

Hsuan-Tien Lin (林軒田)
htlin@csie.ntu.edu.tw

Department of Computer Science
& Information Engineering

National Taiwan University
(國立台灣大學資訊工程系)

Roadmap

1. Embedding Numerous Features: Kernel Models

Lecture 3: Kernel Support Vector Machine

kernel as a shortcut to (transform + inner product) to **remove dependence on $\tilde{d}$**: allowing a spectrum of simple (**linear**) models to infinite dimensional (**Gaussian**) ones with margin control

Lecture 4: Soft-Margin Support Vector Machine

- Motivation and Primal Problem
- Dual Problem
- Messages behind Soft-Margin SVM
- Model Selection

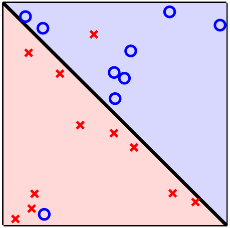
2. Combining Predictive Features: Aggregation Models
3. Distilling Implicit Features: Extraction Models

Cons of Hard-Margin SVM

recall: SVM can still overfit :-(

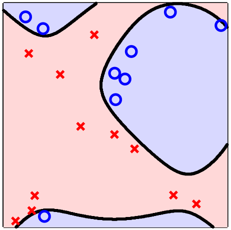
Cons of Hard-Margin SVM

recall: SVM can still overfit :-(



$\mathbf{\Phi}_1$

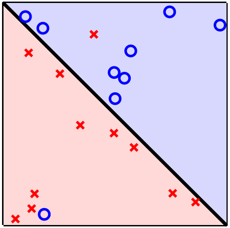
- part of reasons: $\mathbf{\Phi}$



$\mathbf{\Phi}_4$

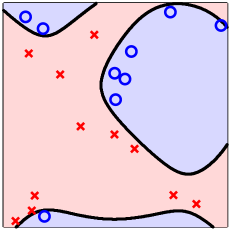
Cons of Hard-Margin SVM

recall: SVM can still overfit :-(



Φ_1

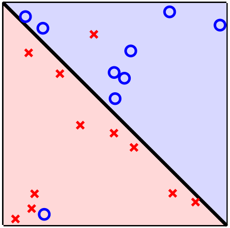
- part of reasons: Φ
- other part: **separable**



Φ_4

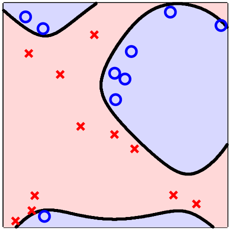
Cons of Hard-Margin SVM

recall: SVM can still overfit :-(



$\mathbf{\Phi}_1$

- part of reasons: $\mathbf{\Phi}$
- other part: **separable**



$\mathbf{\Phi}_4$

if always insisting on **separable** ($\Longrightarrow$ **shatter**), have power to **overfit to noise**

Give Up on Some Examples

want: **give up** on some noisy examples

Give Up on Some Examples

want: **give up** on some noisy examples

pocket

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

Give Up on Some Examples

want: **give up** on some noisy examples

pocket

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

hard-margin SVM

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

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

Give Up on Some Examples

want: **give up** on some noisy examples

pocket

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

hard-margin SVM

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

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

combination:

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

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

Give Up on Some Examples

want: **give up** on some noisy examples

pocket

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

hard-margin SVM

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

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

combination:

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

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

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

Give Up on Some Examples

want: **give up** on some noisy examples

pocket

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

hard-margin SVM

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

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

combination:

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

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

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

C: trade-off of large margin & noise tolerance

Soft-Margin SVM (1/2)

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

$$\text{s.t.} \quad y_n(\mathbf{w}^T\mathbf{z}_n + b) \geq 1 - \infty \cdot [\![y_n \neq \text{sign}(\mathbf{w}^T\mathbf{z}_n + b)]\!]$$

Soft-Margin SVM (1/2)

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

$$\text{s.t.} \quad y_n(\mathbf{w}^T\mathbf{z}_n + b) \geq 1 - \infty \cdot [\![y_n \neq \text{sign}(\mathbf{w}^T\mathbf{z}_n + b)]\!]$$

- $[\![\cdot]\!]$: non-linear, **not QP anymore :-(**

Soft-Margin SVM (1/2)

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

$$\text{s.t.} \quad y_n(\mathbf{w}^T\mathbf{z}_n + b) \geq 1 - \infty \cdot [\![y_n \neq \text{sign}(\mathbf{w}^T\mathbf{z}_n + b)]\!]$$

- $[\![\cdot]\!]$: non-linear, **not QP anymore :-(**
  —what about dual? kernel?

Soft-Margin SVM (1/2)

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

$$\text{s.t.} \quad y_n(\mathbf{w}^T\mathbf{z}_n + b) \geq 1 - \infty \cdot [\![y_n \neq \text{sign}(\mathbf{w}^T\mathbf{z}_n + b)]\!]$$

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

Soft-Margin SVM (1/2)

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

$$\text{s.t.} \quad y_n(\mathbf{w}^T\mathbf{z}_n + b) \geq 1 - \infty \cdot [\![y_n \neq \text{sign}(\mathbf{w}^T\mathbf{z}_n + b)]\!]$$

- $[\![\cdot]\!]$: 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)

Soft-Margin SVM (1/2)

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

$$\text{s.t.} \quad y_n(\mathbf{w}^T\mathbf{z}_n + b) \geq 1 - \infty \cdot [\![y_n \neq \text{sign}(\mathbf{w}^T\mathbf{z}_n + b)]\!]$$

- $[\![\cdot]\!]$: 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)

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

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

Soft-Margin SVM (1/2)

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

$$\text{s.t.} \quad y_n(\mathbf{w}^T\mathbf{z}_n + b) \geq 1 - \infty \cdot [\![y_n \neq \text{sign}(\mathbf{w}^T\mathbf{z}_n + b)]\!]$$

- $[\![\cdot]\!]$: 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**

soft-margin SVM: $$\min_{b,\mathbf{w},\boldsymbol{\xi}} \quad \frac{1}{2}\mathbf{w}^T\mathbf{w} + C \cdot \sum_{n=1}^{N}$$

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

Soft-Margin SVM (1/2)

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

$$\text{s.t.} \quad y_n(\mathbf{w}^T\mathbf{z}_n + b) \geq 1 - \infty \cdot [\![y_n \neq \text{sign}(\mathbf{w}^T\mathbf{z}_n + b)]\!]$$

- $[\![\cdot]\!]$: 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: $$\min_{b,\mathbf{w},\boldsymbol{\xi}} \quad \frac{1}{2}\mathbf{w}^T\mathbf{w} + C \cdot \sum_{n=1}^{N} \xi_n$$

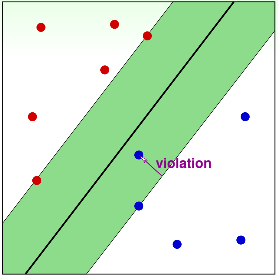
$$\text{s.t.} \quad y_n(\mathbf{w}^T\mathbf{z}_n + b) \geq 1 - \xi_n \text{ and } \xi_n \geq 0 \text{ for all } n$$

Soft-Margin SVM (2/2)

- record '**margin violation**' by ξ_n
- penalize with **margin violation**

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

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

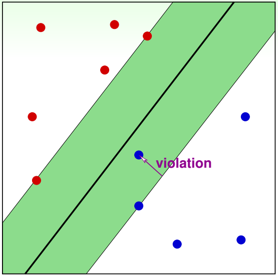


Soft-Margin SVM (2/2)

- record '**margin violation**' by ξ_n
- penalize with **margin violation**

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

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



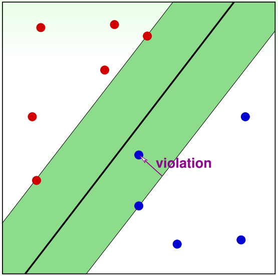
- parameter C: trade-off of large margin & margin violation

Soft-Margin SVM (2/2)

- record '**margin violation**' by ξ_n
- penalize with **margin violation**

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

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



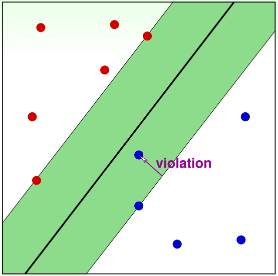
- parameter C: trade-off of large margin & margin violation
  - large C: want less margin violation

Soft-Margin SVM (2/2)

- record '**margin violation**' by ξ_n
- penalize with **margin violation**

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

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



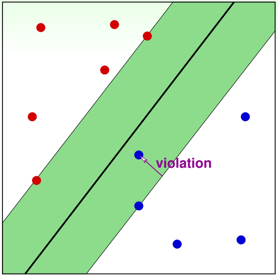
- parameter C: trade-off of large margin & margin violation
  - large C: want less margin violation
  - small C: want large margin

Soft-Margin SVM (2/2)

- record '**margin violation**' by ξ_n
- penalize with **margin violation**

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

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



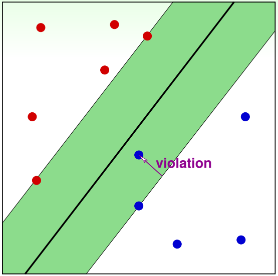
- parameter C: trade-off of large margin & margin violation
  - large C: want less margin violation
  - small C: want large margin
- QP of $\tilde{d} + 1 + N$ variables, $2N$ constraints

Soft-Margin SVM (2/2)

- record '**margin violation**' by ξ_n
- penalize with **margin violation**

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

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



- 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

next: remove dependence on $\tilde{d}$ by soft-margin SVM primal $\Rightarrow$ **dual**?

Fun Time

At the optimal solution of

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

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

assume that $y_1(\mathbf{w}^T\mathbf{z}_1 + b) = -10$. What is the corresponding ξ_1?

1. 1
2. 11
3. 21
4. 31

Fun Time

At the optimal solution of

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

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

assume that $y_1(\mathbf{w}^T\mathbf{z}_1 + b) = -10$. What is the corresponding ξ_1?

1. 1
2. 11
3. 21
4. 31

Reference Answer: 2

ξ_1 is simply $1 - y_1(\mathbf{w}^T\mathbf{z}_1 + b)$ when $y_1(\mathbf{w}^T\mathbf{z}_1 + b) \leq 1$.

Lagrange Dual

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

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

Lagrange Dual

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

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

Lagrange function with Lagrange multipliers α_n and β_n

Lagrange Dual

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

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

Lagrange function with Lagrange multipliers α_n and β_n

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

$$+$$

Lagrange Dual

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

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

Lagrange function with Lagrange multipliers α_n and β_n

$$\begin{aligned}\mathcal{L}(b, \mathbf{w}, \xi, \boldsymbol{\alpha}, \boldsymbol{\beta}) \quad = \quad & \frac{1}{2}\mathbf{w}^T\mathbf{w} + C \cdot \sum_{n=1}^{N} \xi_n \\ & + \sum_{n=1}^{N} \alpha_n \cdot \left(1 - \xi_n - y_n(\mathbf{w}^T\mathbf{z}_n + b)\right) +\end{aligned}$$

Lagrange Dual

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

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

Lagrange function with Lagrange multipliers α_n and β_n

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

Lagrange Dual

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

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

Lagrange function with Lagrange multipliers α_n and β_n

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

want: Lagrange dual

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

Simplify ξ_n and β_n

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

Simplify ξ_n and β_n

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

- $\frac{\partial \mathcal{L}}{\partial \xi_n} = 0 =$

Simplify ξ_n and β_n

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

- $\frac{\partial \mathcal{L}}{\partial \xi_n} = 0 = C - \alpha_n - \beta_n$

Simplify ξ_n and β_n

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

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

Simplify ξ_n and β_n

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

- $\frac{\partial \mathcal{L}}{\partial \xi_n} = 0 = C - \alpha_n - \beta_n$
- 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

Simplify ξ_n and β_n

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

- $\frac{\partial \mathcal{L}}{\partial \xi_n} = 0 = C - \alpha_n - \beta_n$
- 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

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

Simplify ξ_n and β_n

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

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

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

Simplify ξ_n and β_n

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

- $\frac{\partial\mathcal{L}}{\partial\xi_n} = 0 = C - \alpha_n - \beta_n$
- 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

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

Other Simplifications

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

Other Simplifications

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

familiar? :-)

Other Simplifications

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

familiar? :-)

- inner problem **same as hard-margin SVM**

Other Simplifications

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

familiar? :-)

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

Other Simplifications

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

familiar? :-)

- 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 $\mathbf{w} = \sum_{n=1}^{N} \alpha_n y_n \mathbf{z}_n$

Other Simplifications

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

familiar? :-)

- 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 $\mathbf{w} = \sum_{n=1}^{N} \alpha_n y_n \mathbf{z}_n$

standard dual can be derived
using the same steps as Lecture 2

Standard Soft-Margin SVM Dual

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

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

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

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

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

Standard Soft-Margin SVM Dual

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

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

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

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

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

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

Standard Soft-Margin SVM Dual

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

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

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

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

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

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

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

Fun Time

In the soft-margin SVM, assume that we want to increase the parameter C by 2. How shall the corresponding dual problem be changed?

1. the upper bound of α_n shall be halved
2. the upper bound of α_n shall be decreased by 2
3. the upper bound of α_n shall be increased by 2
4. the upper bound of α_n shall be doubled

Fun Time

In the soft-margin SVM, assume that we want to increase the parameter C by 2. How shall the corresponding dual problem be changed?

1. the upper bound of α_n shall be halved
2. the upper bound of α_n shall be decreased by 2
3. the upper bound of α_n shall be increased by 2
4. the upper bound of α_n shall be doubled

Reference Answer: 3

Because C is exactly the upper bound of α_n, increasing C by 2 in the primal problem is equivalent to increasing the upper bound by 2 in the dual problem.

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

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. $\boldsymbol{\alpha} \leftarrow \text{QP}(\mathrm{Q}_\text{D}, \mathbf{p}, \mathrm{A}, \mathbf{c})$

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. $\boldsymbol{\alpha} \leftarrow \mathrm{QP}(\mathrm{Q_D}, \mathbf{p}, \mathrm{A}, \mathbf{c})$
3. $b \leftarrow ?$

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. $\boldsymbol{\alpha} \leftarrow \mathrm{QP}(\mathrm{Q_D}, \mathbf{p}, \mathrm{A}, \mathbf{c})$
3. $b \leftarrow ?$
4. return SVs and their α_n as well as b such that for new $\mathbf{x}$,

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

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. $\alpha \leftarrow \mathrm{QP}(\mathrm{Q}_\mathrm{D}, \mathbf{p}, \mathrm{A}, \mathbf{c})$
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

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. $\boldsymbol{\alpha} \leftarrow \mathrm{QP}(\mathrm{Q_D}, \mathbf{p}, \mathrm{A}, \mathbf{c})$
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**

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. $\boldsymbol{\alpha} \leftarrow \mathrm{QP}(\mathrm{Q_D}, \mathbf{p}, \mathrm{A}, \mathbf{c})$
3. $b \leftarrow ?$
4. return SVs and their α_n as well as b such that for new $\mathbf{x}$,
$$g_{\text{SVM}}(\mathbf{x}) = \text{sign}\left(\sum_{\text{SV indices } n} \alpha_n y_n K(\mathbf{x}_n, \mathbf{x}) + b\right)$$

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

remaining question: step 3?

Solving for b

hard-margin SVM

complementary slackness:

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

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

Solving for b

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:

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

$$(C - \alpha_n)\xi_n = 0$$

Solving for b

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

Solving for b

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

Solving for b

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 ($\alpha_s < C$)
  $\Rightarrow \xi_s =$

Solving for b

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 ($\alpha_s < C$)
  $\Rightarrow \xi_s = 0$

Solving for b

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 $(\mathbf{x}_s, y_s)$:

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

Solving for b

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:

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

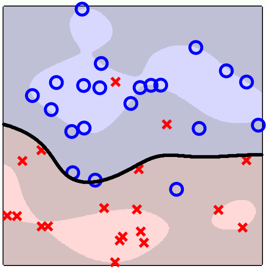
- SV ($\alpha_s > 0$)
  $\Rightarrow b = y_s - y_s\xi_s - \mathbf{w}^T\mathbf{z}_s$
- free ($\alpha_s < C$)
  $\Rightarrow \xi_s = 0$

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

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

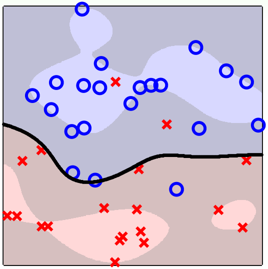
—range of b otherwise

Soft-Margin Gaussian SVM in Action

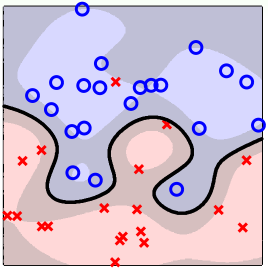


$C = 1$

Soft-Margin Gaussian SVM in Action

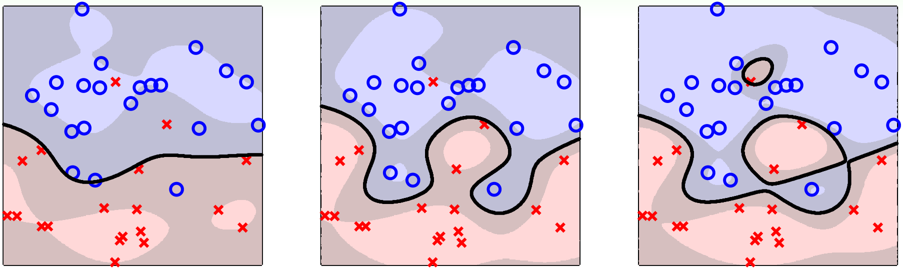


$C = 1$



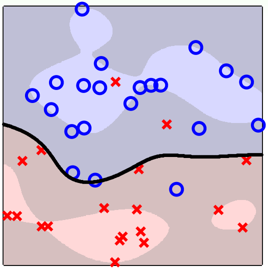
$C = 10$

Soft-Margin Gaussian SVM in Action

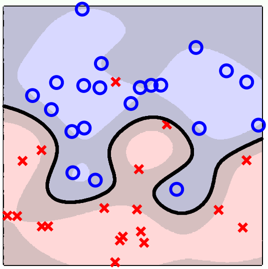


$C = 1$ $C = 10$ $C = 100$

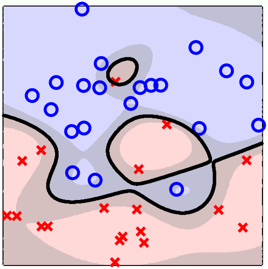
Soft-Margin Gaussian SVM in Action



$C = 1$



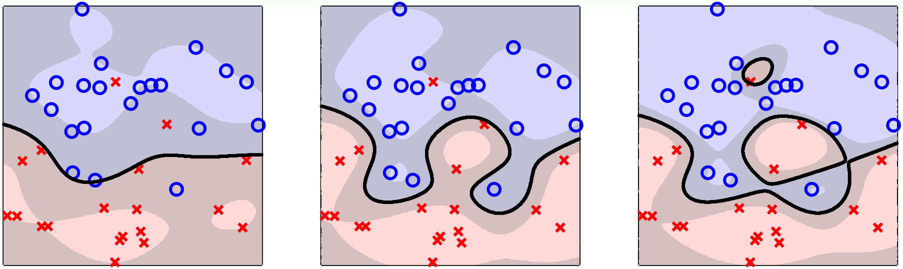
$C = 10$



$C = 100$

- large $C \Longrightarrow$ less noise tolerance $\Longrightarrow$ 'overfit'?

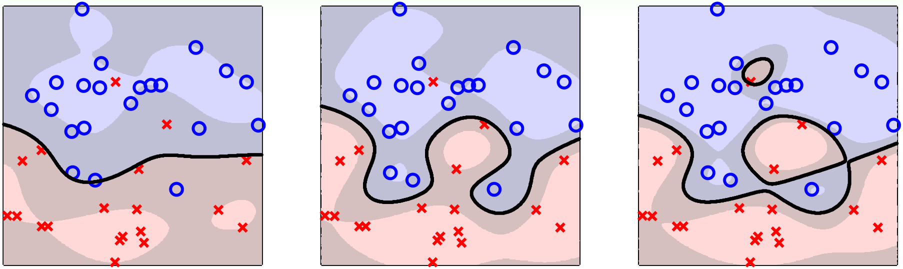
Soft-Margin Gaussian SVM in Action



$C = 1$ $C = 10$ $C = 100$

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

Soft-Margin Gaussian SVM in Action



$C = 1$ $C = 10$ $C = 100$

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

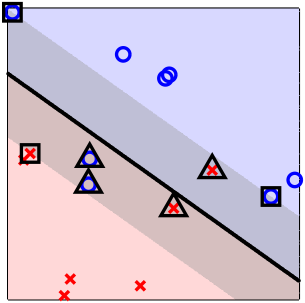
soft-margin Gaussian SVM:
need careful selection of (γ, C)

Physical Meaning of α_n

complementary slackness:

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

$$(C - \alpha_n)\xi_n = 0$$



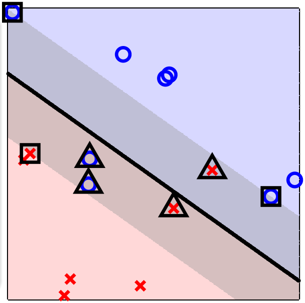
Physical Meaning of α_n

complementary slackness:

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

$$(C - \alpha_n)\xi_n = 0$$

- $\square$ free SV ($0 < \alpha_n < C$): $\xi_n = 0$, on fat boundary, locates b

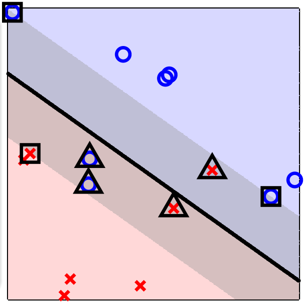


Physical Meaning of α_n

complementary slackness:

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

- non SV ($0 = \alpha_n$): $\xi_n =$,
- $\square$ free SV ($0 < \alpha_n < C$): $\xi_n = 0$, on fat boundary, locates b



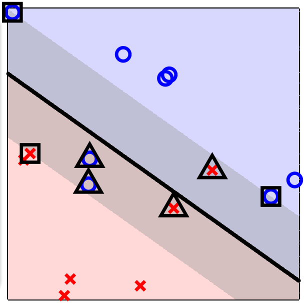
Physical Meaning of α_n

complementary slackness:

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

$$(C - \alpha_n)\xi_n = 0$$

- non SV ($0 = \alpha_n$): $\xi_n = 0$,
- □ free SV ($0 < \alpha_n < C$): $\xi_n = 0$, on fat boundary, locates b



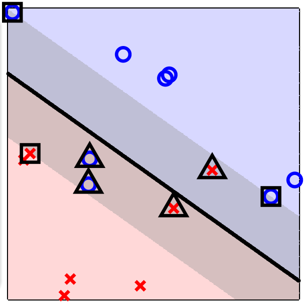
Physical Meaning of α_n

complementary slackness:

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

$$(C - \alpha_n)\xi_n = 0$$

- non SV ($0 = \alpha_n$): $\xi_n = 0$, 'away from'/on fat boundary
- □ free SV ($0 < \alpha_n < C$): $\xi_n = 0$, on fat boundary, locates b



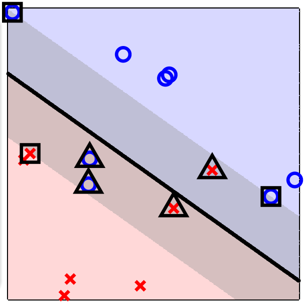
Physical Meaning of α_n

complementary slackness:

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

$$(C - \alpha_n)\xi_n = 0$$

- non SV ($0 = \alpha_n$): $\xi_n = 0$, 'away from'/on fat boundary
- $\square$ free SV ($0 < \alpha_n < C$): $\xi_n = 0$, on fat boundary, locates b
- $\triangle$ bounded SV ($\alpha_n = C$): $\xi_n =$,

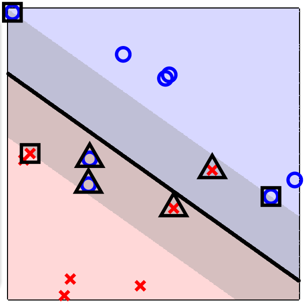


Physical Meaning of α_n

complementary slackness:

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

- non SV ($0 = \alpha_n$): $\xi_n = 0$, 'away from'/on fat boundary
- $\square$ free SV ($0 < \alpha_n < C$): $\xi_n = 0$, on fat boundary, locates b
- $\triangle$ bounded SV ($\alpha_n = C$): $\xi_n =$ violation amount,

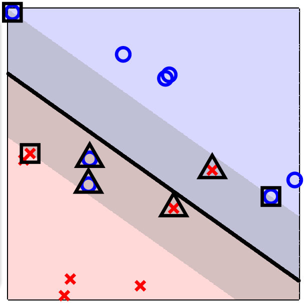


Physical Meaning of α_n

complementary slackness:

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

- non SV ($0 = \alpha_n$): $\xi_n = 0$, 'away from'/on fat boundary
- □ free SV ($0 < \alpha_n < C$): $\xi_n = 0$, on fat boundary, locates b
- △ bounded SV ($\alpha_n = C$): $\xi_n =$ violation amount, 'violate'/on fat boundary

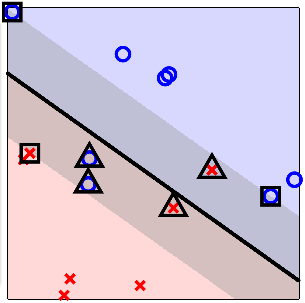


Physical Meaning of α_n

complementary slackness:

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

- non SV ($0 = \alpha_n$): $\xi_n = 0$, 'away from'/on fat boundary
- $\square$ free SV ($0 < \alpha_n < C$): $\xi_n = 0$, on fat boundary, locates b
- $\triangle$ bounded SV ($\alpha_n = C$): $\xi_n =$ violation amount, 'violate'/on fat boundary



α_n can be used for **data analysis**

Fun Time

For a data set of size 10000, after solving SVM, assume that there are 1126 support vectors, and 1000 of those support vectors are bounded. What is the possible range of $E_{\text{in}}(g_{\text{SVM}})$ in terms of 0/1 error?

1. $0.0000 \leq E_{\text{in}}(g_{\text{SVM}}) \leq 0.1000$
2. $0.1000 \leq E_{\text{in}}(g_{\text{SVM}}) \leq 0.1126$
3. $0.1126 \leq E_{\text{in}}(g_{\text{SVM}}) \leq 0.5000$
4. $0.1126 \leq E_{\text{in}}(g_{\text{SVM}}) \leq 1.0000$

Fun Time

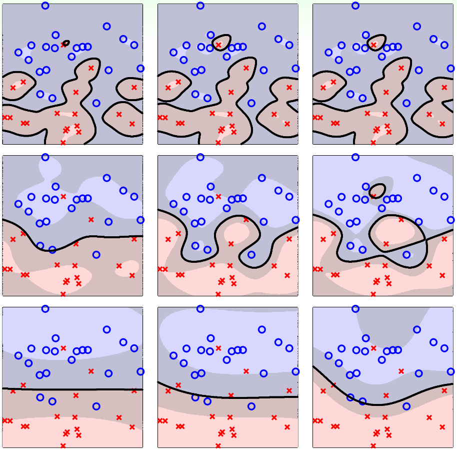
For a data set of size 10000, after solving SVM, assume that there are 1126 support vectors, and 1000 of those support vectors are bounded. What is the possible range of $E_{\text{in}}(g_{\text{SVM}})$ in terms of 0/1 error?

1. $0.0000 \le E_{\text{in}}(g_{\text{SVM}}) \le 0.1000$
2. $0.1000 \le E_{\text{in}}(g_{\text{SVM}}) \le 0.1126$
3. $0.1126 \le E_{\text{in}}(g_{\text{SVM}}) \le 0.5000$
4. $0.1126 \le E_{\text{in}}(g_{\text{SVM}}) \le 1.0000$

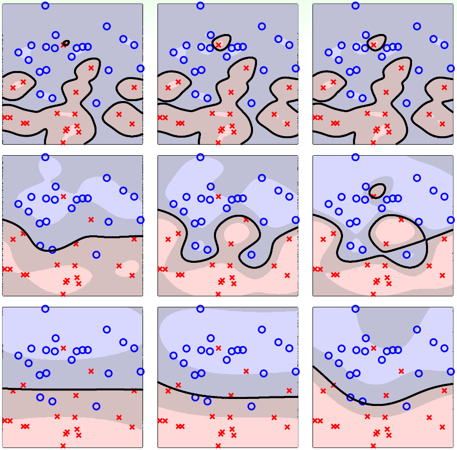
Reference Answer: 1

The bounded support vectors are the only ones that could violate the fat boundary: $\xi_n \ge 0$. If $\xi_n \ge 1$, then the violation causes a 0/1 error on the example. On the other hand, it is also possible that $\xi_n < 1$, and in that case the violation does not cause a 0/1 error.

Practical Need: Model Selection

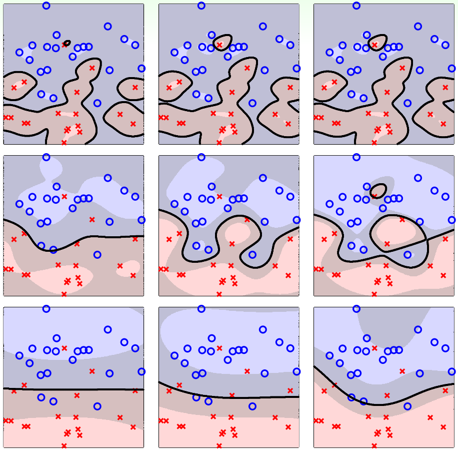


Practical Need: Model Selection



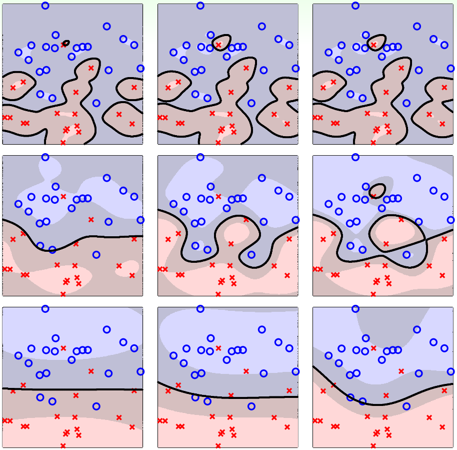
- complicated even for (C, γ) **of Gaussian SVM**

Practical Need: Model Selection



- complicated even for (C, γ) **of Gaussian SVM**
- more combinations if including other kernels or parameters

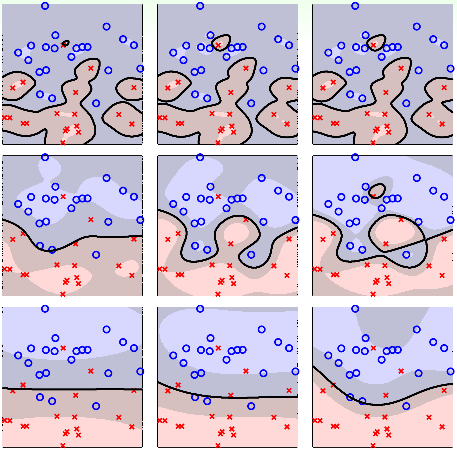
Practical Need: Model Selection



- complicated even for (C, γ) **of Gaussian SVM**
- more combinations if including other kernels or parameters

how to select?

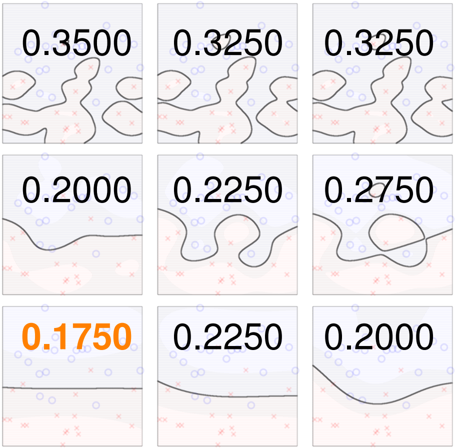
Practical Need: Model Selection



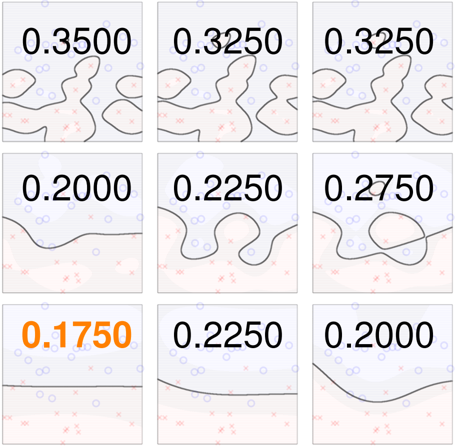
- complicated even for (C, γ) **of Gaussian SVM**
- more combinations if including other kernels or parameters

how to select? **validation :-)**

Selection by Cross Validation

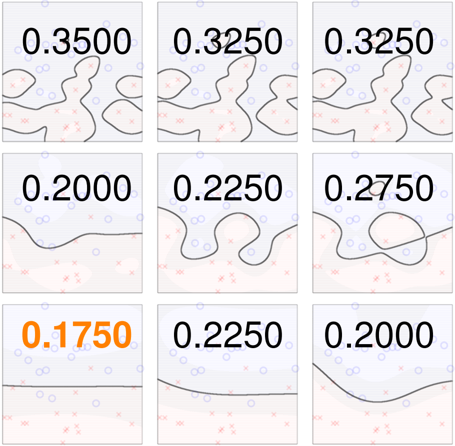


Selection by Cross Validation



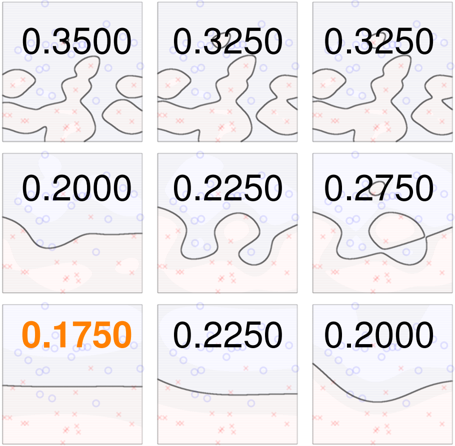
- $E_{\text{cv}}(C, \gamma)$: 'non-smooth' function of (C, γ) —**difficult to optimize**

Selection by Cross Validation



- $E_{\text{cv}}(C, \gamma)$: 'non-smooth' function of (C, γ) —**difficult to optimize**
- proper models can be chosen by ***V*-fold cross validation** on **a few grid values of (C, γ)**

Selection by Cross Validation



- $E_{\text{cv}}(C, \gamma)$: 'non-smooth' function of (C, γ) —**difficult to optimize**
- proper models can be chosen by ***V*-fold cross validation** on **a few grid values of (C, γ)**

E_{cv}: very popular criteria for soft-margin SVM

Leave-One-Out CV Error for SVM

recall: $E_{\text{loocv}} = E_{\text{cv}}$ with N folds

Leave-One-Out CV Error for SVM

recall: $E_{\text{loocv}} = E_{\text{cv}}$ with N folds

claim: $E_{\text{loocv}} \le \frac{\#\text{SV}}{N}$

Leave-One-Out CV Error for SVM

recall: $E_{\text{loocv}} = E_{\text{cv}}$ with N folds

claim: $E_{\text{loocv}} \leq \frac{\#\text{SV}}{N}$

- for $(\mathbf{x}_N, y_N)$: if optimal $\alpha_N = 0$ (non-SV)
  $\Longrightarrow$

Leave-One-Out CV Error for SVM

recall: $E_{\text{loocv}} = E_{\text{cv}}$ with N folds

claim: $E_{\text{loocv}} \leq \frac{\#\text{SV}}{N}$

- for $(\mathbf{x}_N, y_N)$: if optimal $\alpha_N = 0$ (non-SV) $\Longrightarrow (\alpha_1, \alpha_2, \ldots, \alpha_{N-1})$ still optimal when leaving out $(\mathbf{x}_N, y_N)$

Leave-One-Out CV Error for SVM

recall: $E_{\text{loocv}} = E_{\text{cv}}$ with N folds

claim: $E_{\text{loocv}} \leq \frac{\#\text{SV}}{N}$

- for $(\mathbf{x}_N, y_N)$: if optimal $\alpha_N = 0$ (non-SV)
  $\Longrightarrow (\alpha_1, \alpha_2, \ldots, \alpha_{N-1})$ still optimal when leaving out $(\mathbf{x}_N, y_N)$
  key: **what if there's better** α_n?

Leave-One-Out CV Error for SVM

recall: $E_{\text{loocv}} = E_{\text{cv}}$ with N folds

claim: $E_{\text{loocv}} \leq \frac{\#\text{SV}}{N}$

- for $(\mathbf{x}_N, y_N)$: if optimal $\alpha_N = 0$ (non-SV)
  $\Longrightarrow (\alpha_1, \alpha_2, \ldots, \alpha_{N-1})$ still optimal when leaving out $(\mathbf{x}_N, y_N)$
  key: **what if there's better** α_n?
- SVM: $g^- = g$ when leaving out non-SV

Leave-One-Out CV Error for SVM

recall: $E_{\text{loocv}} = E_{\text{cv}}$ with N folds

claim: $E_{\text{loocv}} \leq \frac{\#\text{SV}}{N}$

- for $(\mathbf{x}_N, y_N)$: if optimal $\alpha_N = 0$ (non-SV)
  $\Longrightarrow (\alpha_1, \alpha_2, \ldots, \alpha_{N-1})$ still optimal when leaving out $(\mathbf{x}_N, y_N)$
  key: **what if there's better** α_n?
- SVM: $g^- = g$ when leaving out non-SV

$$\begin{aligned} e_{\text{non-SV}} &= \text{err}(g^-, \text{non-SV}) \\ &= \end{aligned}$$

Leave-One-Out CV Error for SVM

recall: $E_{\text{loocv}} = E_{\text{cv}}$ with N folds

claim: $E_{\text{loocv}} \leq \frac{\#\text{SV}}{N}$

- for $(\mathbf{x}_N, y_N)$: if optimal $\alpha_N = 0$ (non-SV)
  $\Longrightarrow (\alpha_1, \alpha_2, \ldots, \alpha_{N-1})$ still optimal when leaving out $(\mathbf{x}_N, y_N)$
  key: **what if there's better** α_n?
- SVM: $g^- = g$ when leaving out non-SV

$$
\begin{aligned}
e_{\text{non-SV}} &= \text{err}(g^-, \text{non-SV}) \\
&= \text{err}(g, \text{non-SV}) = 0
\end{aligned}
$$

Leave-One-Out CV Error for SVM

recall: $E_{\text{loocv}} = E_{\text{cv}}$ with N folds

claim: $E_{\text{loocv}} \leq \frac{\#\text{SV}}{N}$

- for $(\mathbf{x}_N, y_N)$: if optimal $\alpha_N = 0$ (non-SV)
  $\Longrightarrow (\alpha_1, \alpha_2, \ldots, \alpha_{N-1})$ still optimal when leaving out $(\mathbf{x}_N, y_N)$
  key: **what if there's better** α_n?
- SVM: $g^- = g$ when leaving out non-SV

$$\begin{aligned} e_{\text{non-SV}} &= \text{err}(g^-, \text{non-SV}) \\ &= \text{err}(g, \text{non-SV}) = 0 \\ e_{\text{SV}} &\leq 1 \end{aligned}$$

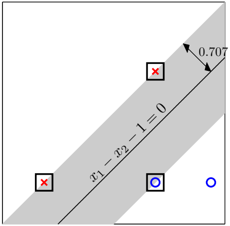
Leave-One-Out CV Error for SVM

recall: $E_{\text{loocv}} = E_{\text{cv}}$ with N folds

claim: $E_{\text{loocv}} \le \frac{\#\text{SV}}{N}$

- for $(\mathbf{x}_N, y_N)$: if optimal $\alpha_N = 0$ (non-SV) $\Longrightarrow (\alpha_1, \alpha_2, \ldots, \alpha_{N-1})$ still optimal when leaving out $(\mathbf{x}_N, y_N)$
  key: **what if there's better α_n?**
- SVM: $g^- = g$ when leaving out non-SV

$$\begin{aligned} e_{\text{non-SV}} &= \text{err}(g^-, \text{non-SV}) \\ &= \text{err}(g, \text{non-SV}) = 0 \\ e_{\text{SV}} &\le 1 \end{aligned}$$



motivation from hard-margin SVM: only **SVs needed**

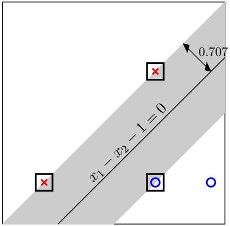
Leave-One-Out CV Error for SVM

recall: $E_{\text{loocv}} = E_{\text{cv}}$ with N folds

claim: $E_{\text{loocv}} \le \frac{\#\text{SV}}{N}$

- for $(\mathbf{x}_N, y_N)$: if optimal $\alpha_N = 0$ (non-SV)
  $\Longrightarrow (\alpha_1, \alpha_2, \ldots, \alpha_{N-1})$ still optimal when leaving out $(\mathbf{x}_N, y_N)$
  key: **what if there's better** α_n?
- SVM: $g^- = g$ when leaving out non-SV

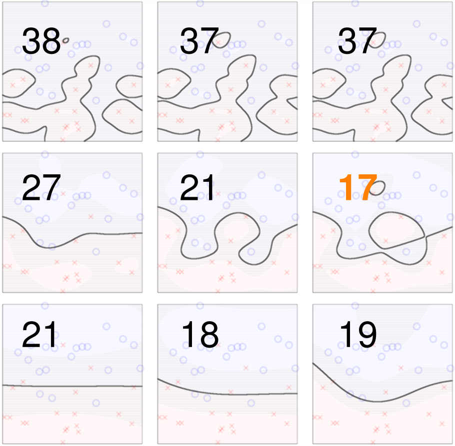
$$
\begin{aligned}
e_{\text{non-SV}} &= \text{err}(g^-, \text{non-SV}) \\
&= \text{err}(g, \text{non-SV}) = 0 \\
e_{\text{SV}} &\le 1
\end{aligned}
$$



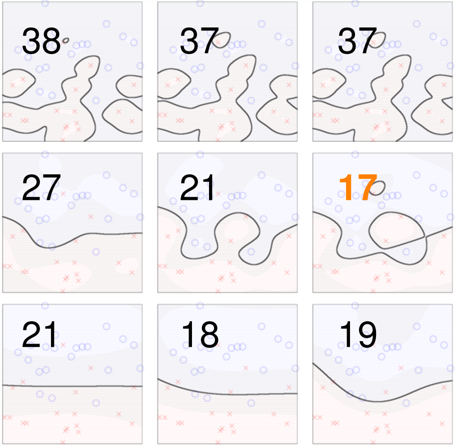
motivation from hard-margin SVM: only **SVs needed**

scaled $\#$SV bounds leave-one-out CV error

Selection by # SV

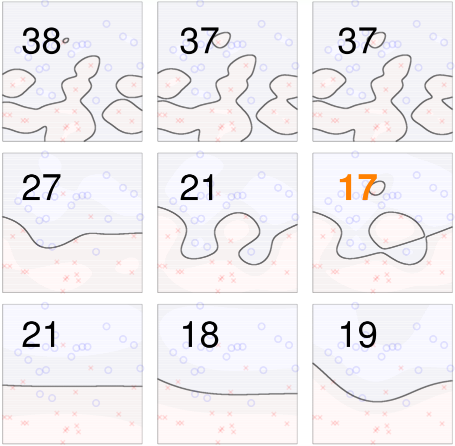


Selection by # SV



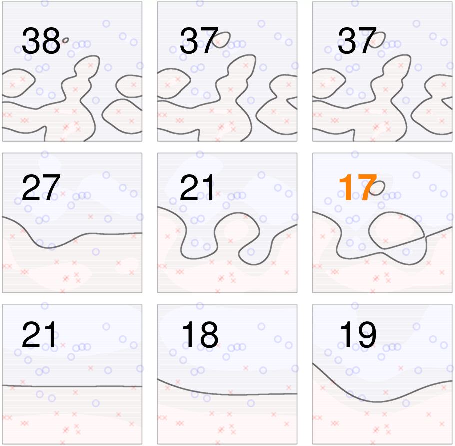
- nSV(C, γ): 'non-smooth' function of (C, γ) —**difficult to optimize**

Selection by # SV



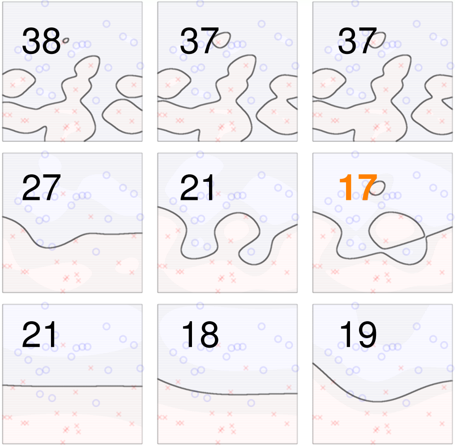
- nSV(C, γ): 'non-smooth' function of (C, γ) —**difficult to optimize**
- **just an upper bound!**

Selection by # SV



- nSV(C, γ): 'non-smooth' function of (C, γ) —**difficult to optimize**
- **just an upper bound!**
- dangerous models can be ruled out by **nSV** on **a few grid values of (C, γ)**

Selection by # SV



- nSV(C, γ): 'non-smooth' function of (C, γ) —**difficult to optimize**
- **just an upper bound!**
- dangerous models can be ruled out by **nSV** on **a few grid values of (C, γ)**

nSV: often used as a **safety check** if computing E_{cv} is too time-consuming

Fun Time

For a data set of size 10000, after solving SVM on some parameters, assume that there are 1126 support vectors, and 1000 of those support vectors are bounded. Which of the following cannot be E_{loocv} with those parameters?

1. 0.0000
2. 0.0805
3. 0.1111
4. 0.5566

Fun Time

For a data set of size 10000, after solving SVM on some parameters, assume that there are 1126 support vectors, and 1000 of those support vectors are bounded. Which of the following cannot be E_{loocv} with those parameters?

1. 0.0000
2. 0.0805
3. 0.1111
4. 0.5566

Reference Answer: 4

Note that the upper bound of E_{loocv} is 0.1126.

Summary

1. Embedding Numerous Features: Kernel Models

Lecture 4: Soft-Margin Support Vector Machine

- Motivation and Primal Problem

  add margin violations ξ_n
- Dual Problem

  upper-bound α_n by C
- Messages behind Soft-Margin SVM

  bounded/free SVs for data analysis
- Model Selection

  cross-validation, or approximately nSV

- **next: other kernel models for soft binary classification**

2. Combining Predictive Features: Aggregation Models
3. Distilling Implicit Features: Extraction Models