CS 461: Machine Learning Principles

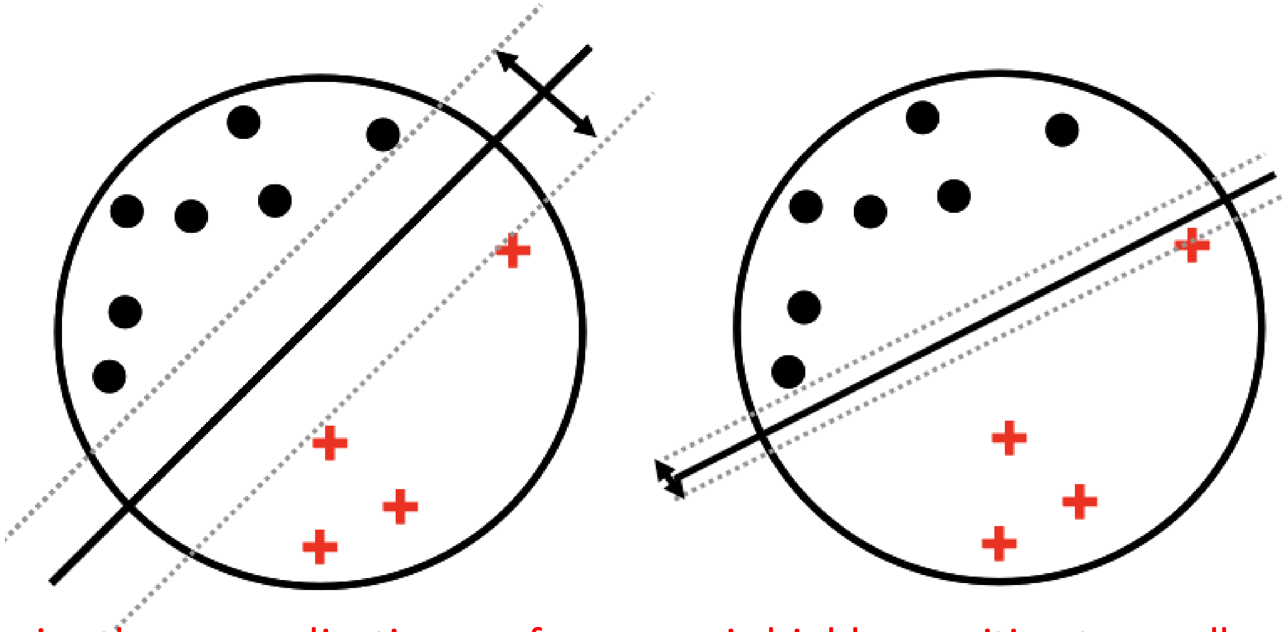
Class 13: Oct. 17

SVM (SMO Algorithm)

& Decision Tree

Instructor: Diana Kim

Computing a Large Margin Classifier



+ as we have a small margin, the generalization performance is highly sensitive to small perturbations in the position of the optimal hyperplane.

There are many hyperplanes to sperate the training data points.
But a maximum margin classifier on training set is desirable.

- high confident separation
- robust to perturbation of data (generalization)

SVM Problem [Hard Margin SVM]

$$w*, b* = \arg\max_{w,b} \frac{\Delta}{||w||}$$

$$\text{subject to} \quad t_n(w^t x_n + b) \geq \Delta \quad \forall n$$

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

$$\text{subject to} \quad t_n(w^t x_n + b) \geq 1 \quad \forall n$$

All data points are separated by the minimum distance of $\Delta/||W||$ hyperplane , we want to maximize the minimum distance.

SVM Primal and Dual Problem

- Primal

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

$$\text{subject to} \quad t_n(w^t x_n + b) \geq 1 \quad \forall n$$

- Dual

$$\lambda*_{n=1}^{N} = \arg\max_{\lambda*} \sum_{n=1}^{N} \lambda *_n - \frac{1}{2} \sum_{n=1}^{N} \sum_{m=1}^{N} \lambda_n^* \cdot t_n \lambda_m^* \cdot t_m \cdot \kappa(x_n, x_m)$$

$$\text{subject to} \quad \lambda_n^* \geq 0 \quad \forall n$$

$$\sum_{n=1}^{N} \lambda_n^* \cdot t_n = 0$$

Advantage of Dual Formulation

- Dual

$$\lambda *_{n=1}^{N} = \arg\max_{\lambda *} \sum_{n=1}^{N} \lambda *_n - \frac{1}{2} \sum_{n=1}^{N} \sum_{m=1}^{N} \lambda_n^* \cdot t_n \lambda_m^* \cdot t_m \cdot \kappa(x_n, x_m)$$

$$\text{subject to} \quad \lambda_n^* \geq 0 \quad \forall n$$

$$\sum_{n=1}^{N} \lambda_n^* \cdot t_n = 0$$

- By using kernel trick, maximum margin classifier can be applied efficiently to feature spaces whose dimensionality exceeds the number of data points. (M>>N)
- By using kernel trick, we can make training data linearly separable.

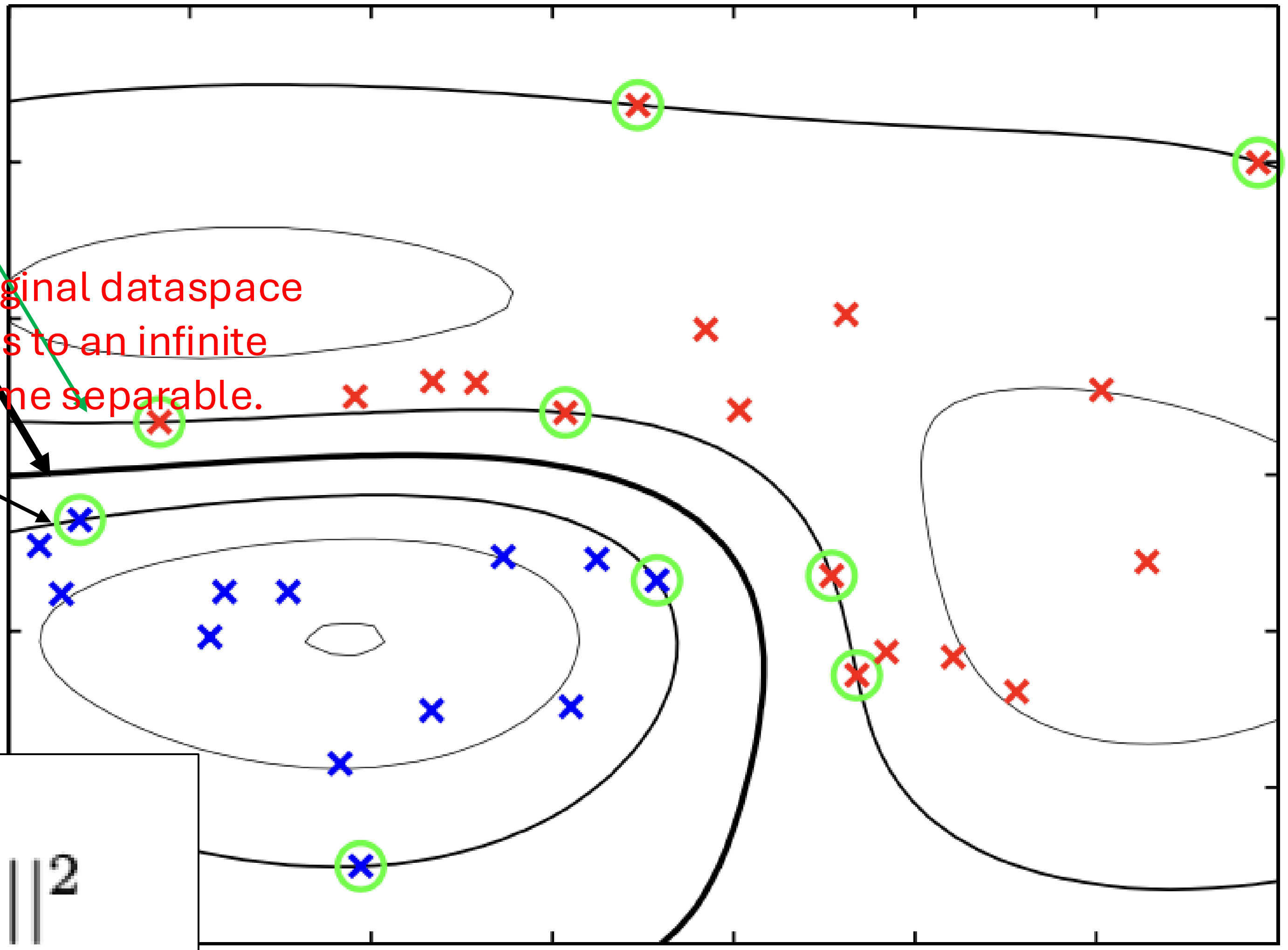
The Outcomes of Gaussian Kernel SVM

(1) **Decision hyperplane**

(2) Support Vectors (Samples)

(3) Margins

+ the data points are not linearly separable in the original dataspace
But Gaussian kernel implicitly project the data points to an infinite dimensional feature space, so the data points become separable.



Gaussian Kernel

$$\kappa(x, x') = \exp \frac{-||x - x'||^2}{2\sigma^2}$$

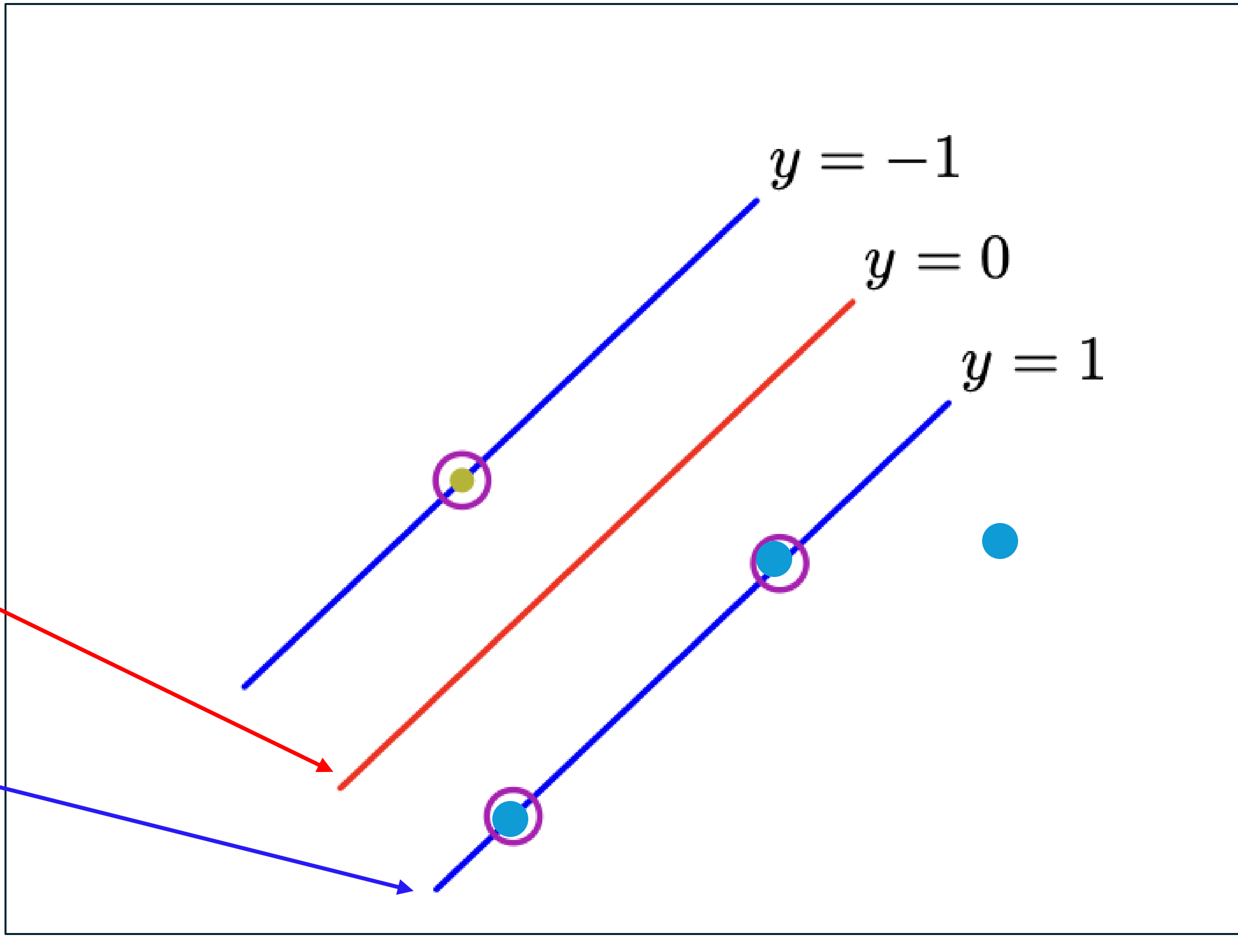
From the dual function, we compute $\lambda *$ (n = 1,…N).
Then we can build a maximum margin classifier.
A subset of the training data points are used to build the classifier.

$$y(x) = \sum_{n=1}^{N} \lambda *_n t_n \kappa(x_n, x) + b$$

$$\text{The second and third KKT} = \begin{cases} \boxed{\lambda *_n = 0} & \text{if} \quad t_n(w *^t x_n + b) > 1 \\ \boxed{\lambda *_n > 0} & \text{if} \quad t_n(w *^t x_n + b) = 1 \end{cases}$$

Support Vectors x_n!

From Bishop Figure 7.1



The Outcomes of Linear Kernel SVM

(1) Decision hyperplane

(2) Support Vectors (Samples)

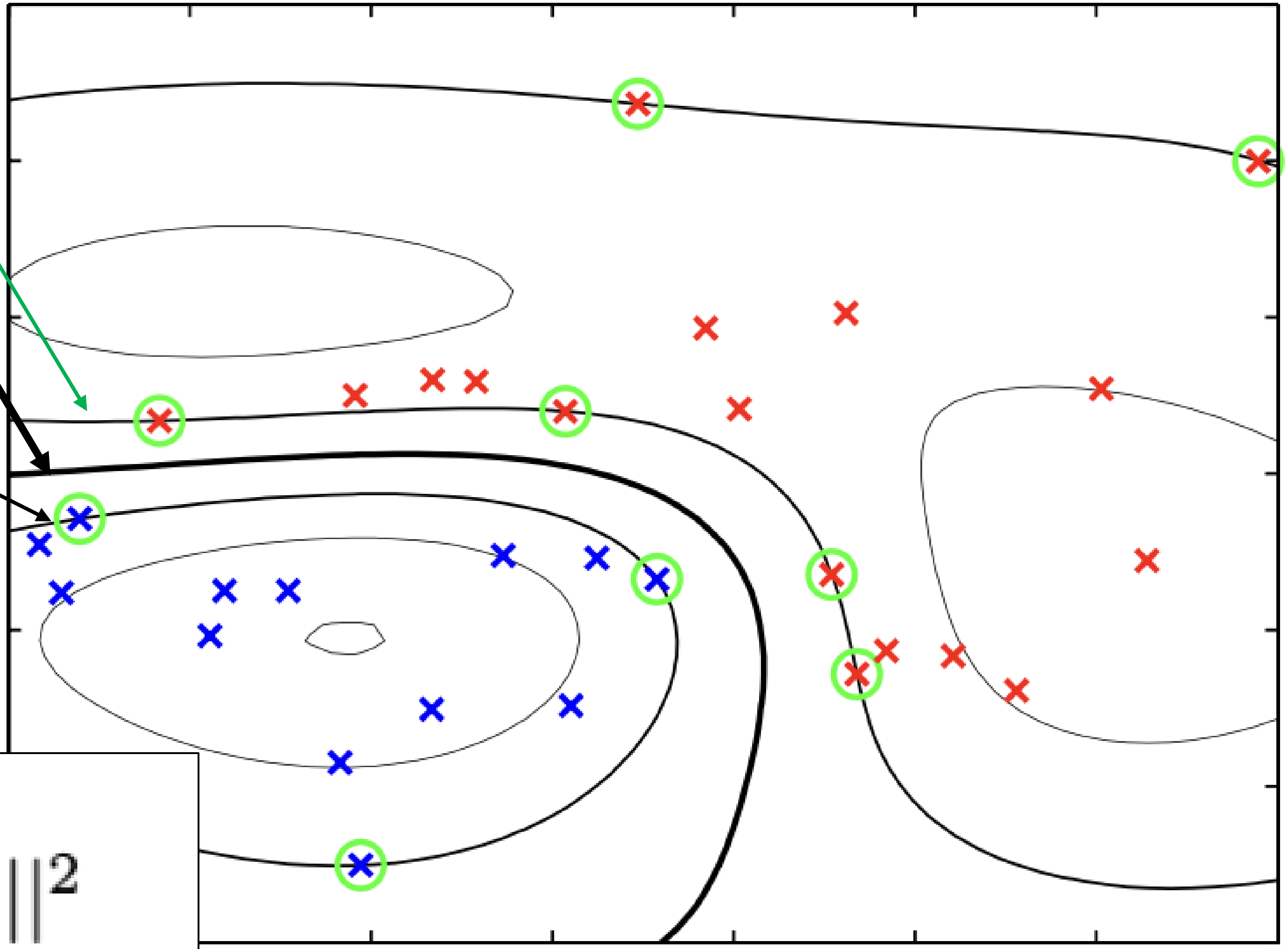
(3) Margins

The Outcomes of Gaussian Kernel SVM

(1) **Decision hyperplane**

(2) Support Vectors (Samples)

(3) Margins



Gaussian Kernel

$$\kappa(x, x') = \exp \frac{-||x - x'||^2}{2\sigma^2}$$

About Gaussian Kernel

Gaussian Kernel

$$\kappa(x, x') = \exp \frac{-\,||x - x'||^2}{2\sigma^2}$$

- Gaussian kernel embeds the infinite dimensional feature space. However, SVM with Gaussian is not sensitive to # data points because it the maximum margin boundary can be defined by two + / – data samples. (performance/complexity)

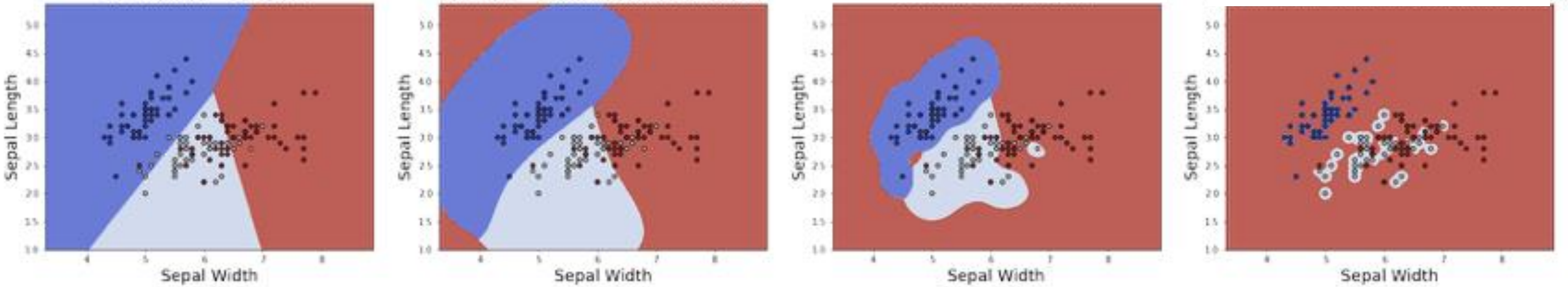
- The model complexity depends on σ.

The Effect of Gamma on the number of Support vectors & decision Boundary

$$\gamma = \frac{1}{\sigma^2}$$

From https://www.kaggle.com/code/gorkemgunay/understanding-parameters-of-svm

Gamma : 0.01 $Gamma$:1 Gamma :10 Gamma :500



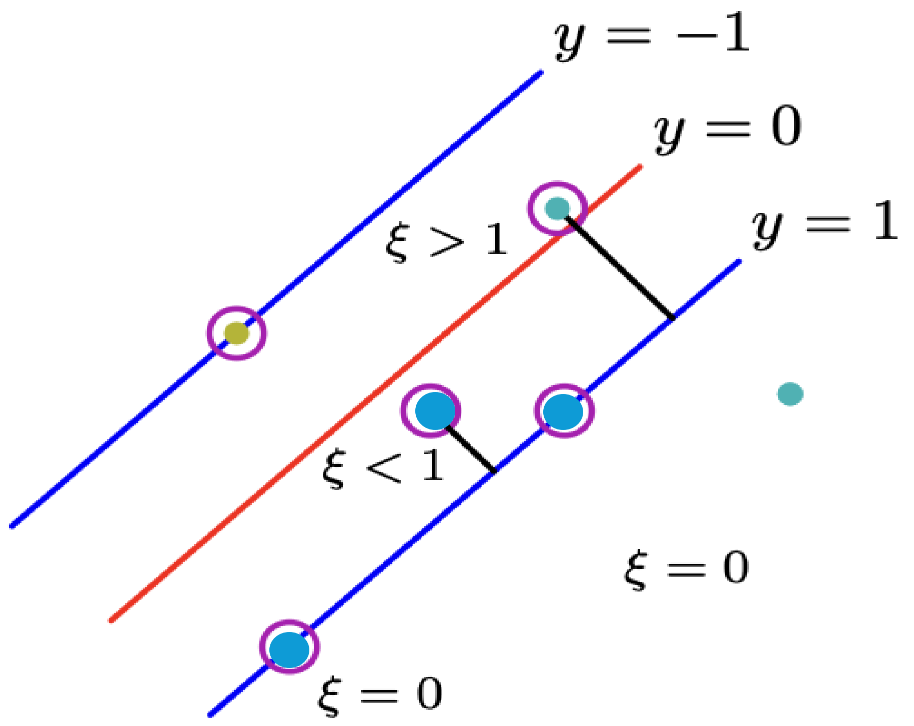
- Small gamma: some representative samples become support vectors.
- Large gamma: every samples become support vectors

What happens for SVM if data samples are not linearly separable?

+ It does not converge, so we use soft margin SVM for the case.

For each data points,
the constraint to define the margin is
relaxed by the slack variable $\xi_n \geq 0$

From Bishop Figure 7.3



- hard margin

$$t_n(w^t x_n + b) \geq 1 \quad \forall n$$

- soft margin

$$t_n(w^t x_n + b) \geq \boxed{1 - \xi n} \text{ and } \quad \xi n \geq 0 \quad \forall n$$

+ margin can be reduced; even it can be a negative value.

- Hard Margin Case (Exact-Separable)

$$w*, b* = \underset{w,b}{\arg\min} \frac{1}{2}||w||^2$$

$$\text{subject to} \quad t_n(w^t x_n + b) \geq 1 \quad \forall n$$

- Soft Margin Case (Non-Separable)

$$w*, b* = \underset{w,b}{\arg\min}\, C \cdot \sum_{n=1}^{N} \xi_n + \frac{1}{2}||w||^2$$

$$\text{subject to} \quad t_n(w^t x_n + b) \geq 1 - \xi_n \quad \forall n$$

$$\text{subject to} \quad \xi_n \geq 0 \quad \forall n$$

Soft Margin SVM Primal and Dual Problem

- Primal

$$w*, b* = \underset{w,b}{\arg\min}\, C \cdot \sum_{n=1}^{N} \xi_n + \frac{1}{2}||w||^2$$

$$\text{subject to} \quad t_n(w^t x_n + b) \geq 1 - \xi_n \quad \forall n$$

$$\text{subject to} \quad \xi_n \geq 0 \quad \forall n$$

- Dual

$$\lambda*_{n=1}^{N} = \underset{\lambda*}{\arg\max} \sum_{n=1}^{N} \lambda *_n - \frac{1}{2} \sum_{n=1}^{N} \sum_{m=1}^{N} \lambda_n^* \cdot t_n \lambda_m^* \cdot t_m \cdot \kappa(x_n, x_m)$$

$$\text{subject to} \quad 0 \leq \lambda_n^* \leq C$$

$$\sum_{n=1}^{N} \lambda_n^* \cdot t_n = 0$$

KKT conditions

$$\nabla_w L(w, b, \lambda_{n=1}^N, \xi_{n=1}^N, \mu_{n=1}^M) = \vec{w} - \sum_{n=1}^{N} (\lambda_n \cdot t_n \cdot \vec{x_n})$$

$$\nabla_b L(w, b, \lambda_{n=1}^N, \xi_{n=1}^N, \mu_{n=1}^M) = \sum_{n=1}^{N} \lambda_n \cdot t_n$$

$$\nabla_{\xi_n} L(w, b, \lambda_{n=1}^N, \xi_{n=1}^N, \mu_{n=1}^M) = C - \lambda_n - \mu_n$$

$$\lambda_n \cdot \{t_n(w^t x_n) + b - 1 + \xi_n\} = 0$$

$$\mu_n \xi n = 0$$

Soft Margin SVM Primal and Dual Problem

- Primal

$$w*, b* = \arg\min_{w,b} C \cdot \sum_{n=1}^{N} \xi_n + \frac{1}{2}||w||^2$$

$$\text{subject to} \quad t_n(w^t x_n + b) \geq 1 - \xi_n \quad \forall n$$

$$\text{subject to} \quad \xi_n \geq 0 \quad \forall n$$

- Dual

$$\lambda *_{n=1}^{N} = \arg\max_{\lambda *} \sum_{n=1}^{N} \lambda *_n - \frac{1}{2} \sum_{n=1}^{N} \sum_{m=1}^{N} \lambda_n^* \cdot t_n \lambda_m^* \cdot t_m \cdot \kappa(x_n, x_m)$$

$$\text{subject to} \quad 0 \leq \lambda_n^* \leq C$$

$$\sum_{n=1}^{N} \lambda_n^* \cdot t_n = 0$$

- $\lambda *_n = 0$
- Support vectors: $0 < \lambda *_n < C$
- Support vectors: $\lambda *_n = C$

All data samples must satisfy the KKT condition.

- $\lambda *_n = 0$
- Support vectors: $0 < \lambda *_n < C$
- Support vectors: $\lambda *_n = C$

→

"correctly labeled with a room to spare"

$$y_n(w^t x_n + b) \geq 1$$

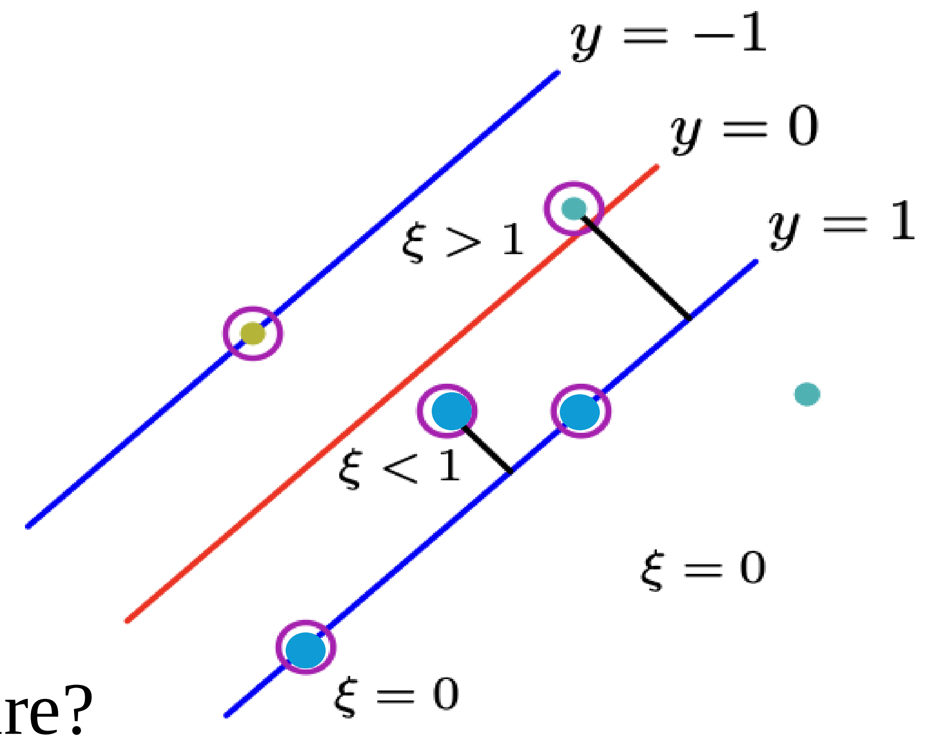
$$y_n(w^t x_n + b) = 1$$ "unbound"

$$y_n(w^t x_n + b) \leq 1$$

"incorrectly labeled or lie within the margin"

From Bishop Figure 7.3

- $\lambda *_n = 0$
- Support vectors: $0 < \lambda *_n < C$
- Support vectors: $\lambda *_n = C$

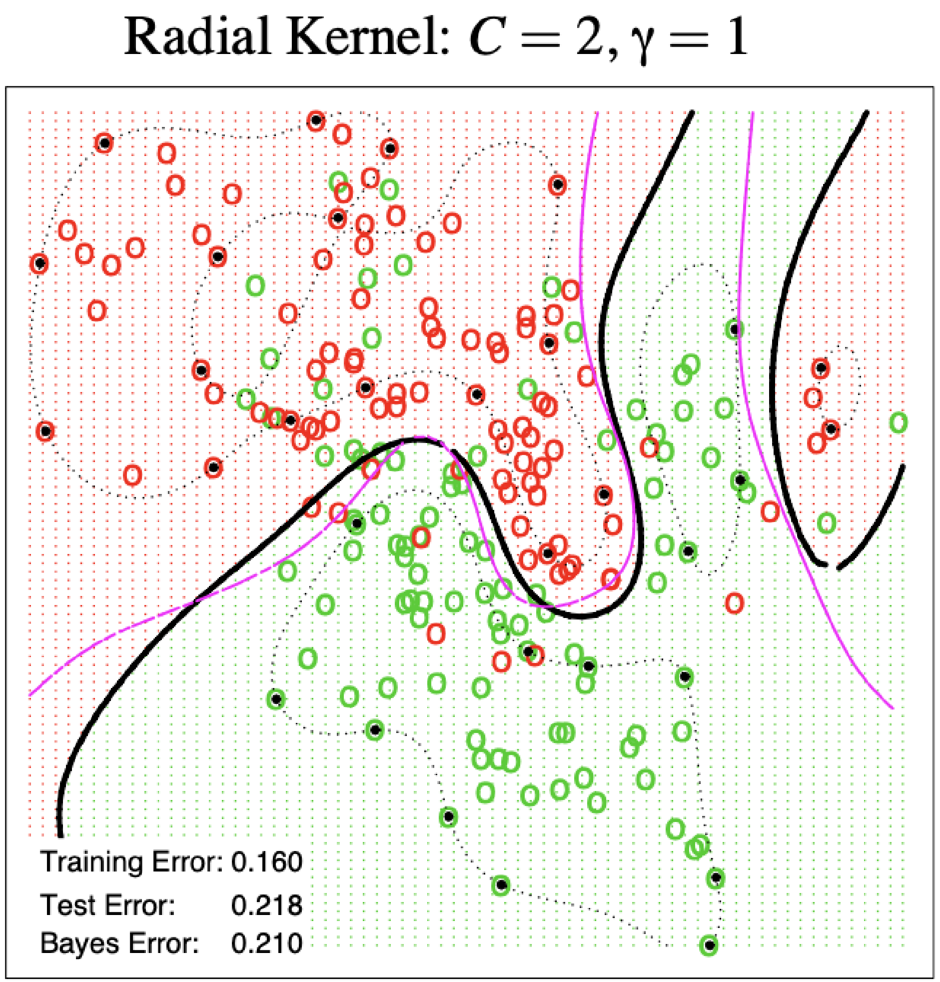


Q: How many support vectors are in this figure?

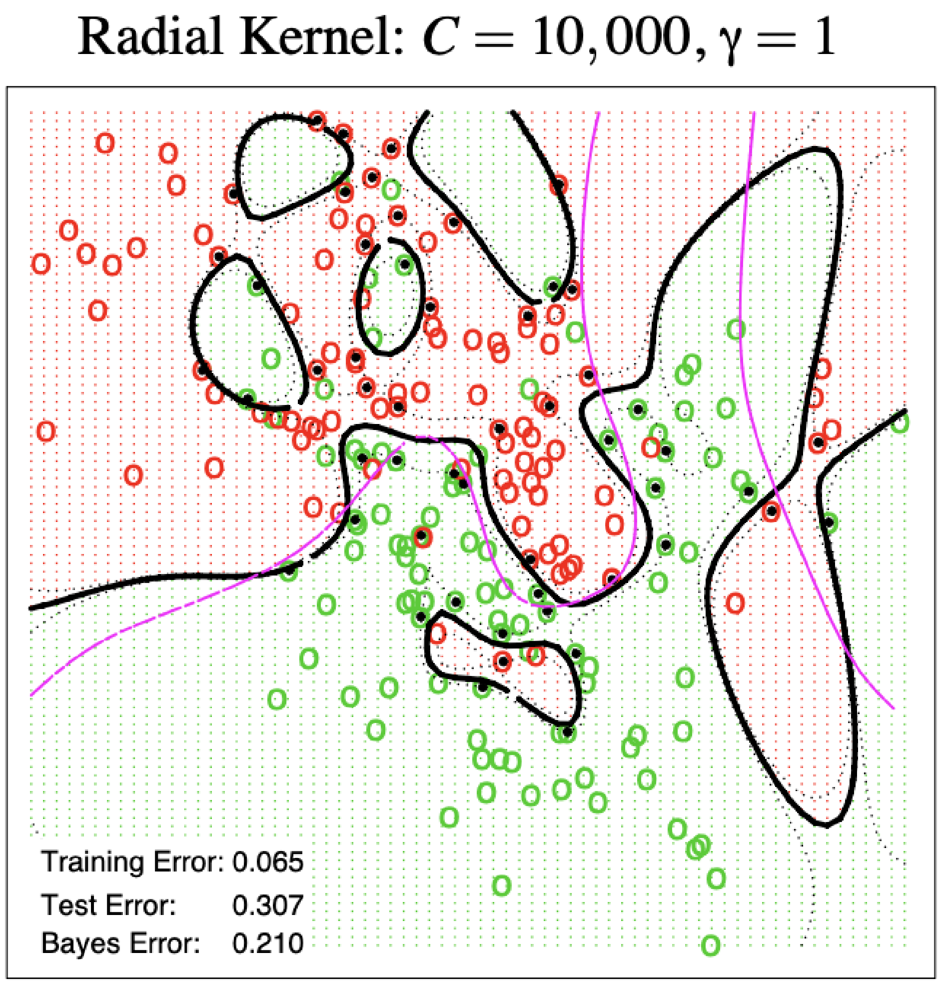
Q: Soft margin SVM can handle non-separable data.
Can soft margin algorithm function as regularization for SVM?

+ yes, by using the slack variable, the soft margin SVM can set a larger margin (than then tight margin without slack) and allow some data points are within the margin area. This results in having more support vectors to build the final SVM machine. More support vectors around boundary area create the smoother boundary.

From the paper: https://jmlr.csail.mit.edu/papers/volume5/hastie04a/hastie04a.pdf



For small C, Large Slack



For large C, Small Slack

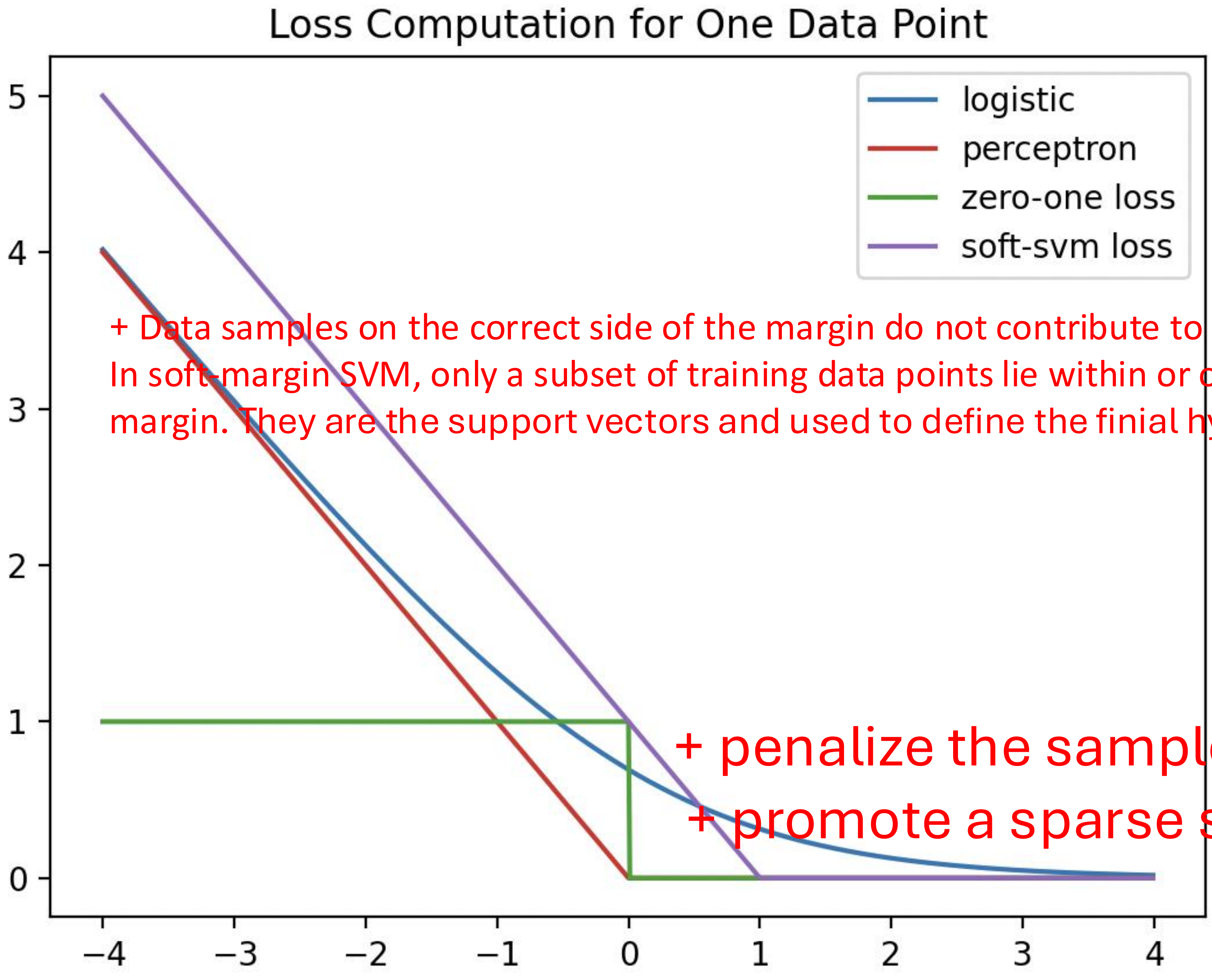
- Small C: Smooth boundaries for Soft-Margin so better generalization

We need to conduct cross validation to select proper σ^2 and C.

The Loss for One Data Point of soft Margin SVM

$$\text{Soft Margin SVM Loss }(x,y) = \begin{cases} \xi_n = 0 \quad \text{for} \quad yw^t x \geq 1 \\ \\ \xi_n = 1 - yw^t x \quad \text{for} \quad yw^t x < 1 \end{cases}$$

Loss Comparison for One Data Point (Objective Function)



+ Data samples on the correct side of the margin do not contribute to the loss.
In soft-margin SVM, only a subset of training data points lie within or on the incorrect side of the margin. They are the support vectors and used to define the finial hyperplane.

+ penalize the samples within the margi

+ promote a sparse solution

Solving Minimal Optimization (SMO)

- Suppose given $\mathcal{D} : \{(x, y) : x \in \mathbb{R}^M \; and \; t \in \{-1, 1\}\}$ and $\kappa(x, x')$.
  A soft SVM dual function is defined below.
  How could we find the optimal $\lambda_1, \lambda_2 \; \dots, \lambda_n$?

$$\lambda *_{n=1}^{N} = \arg\max_{\lambda *} \sum_{n=1}^{N} \lambda *_n - \frac{1}{2} \sum_{n=1}^{N} \sum_{m=1}^{N} \lambda_n^* \cdot t_n \lambda_m^* \cdot t_m \cdot \kappa(x_n, x_m)$$

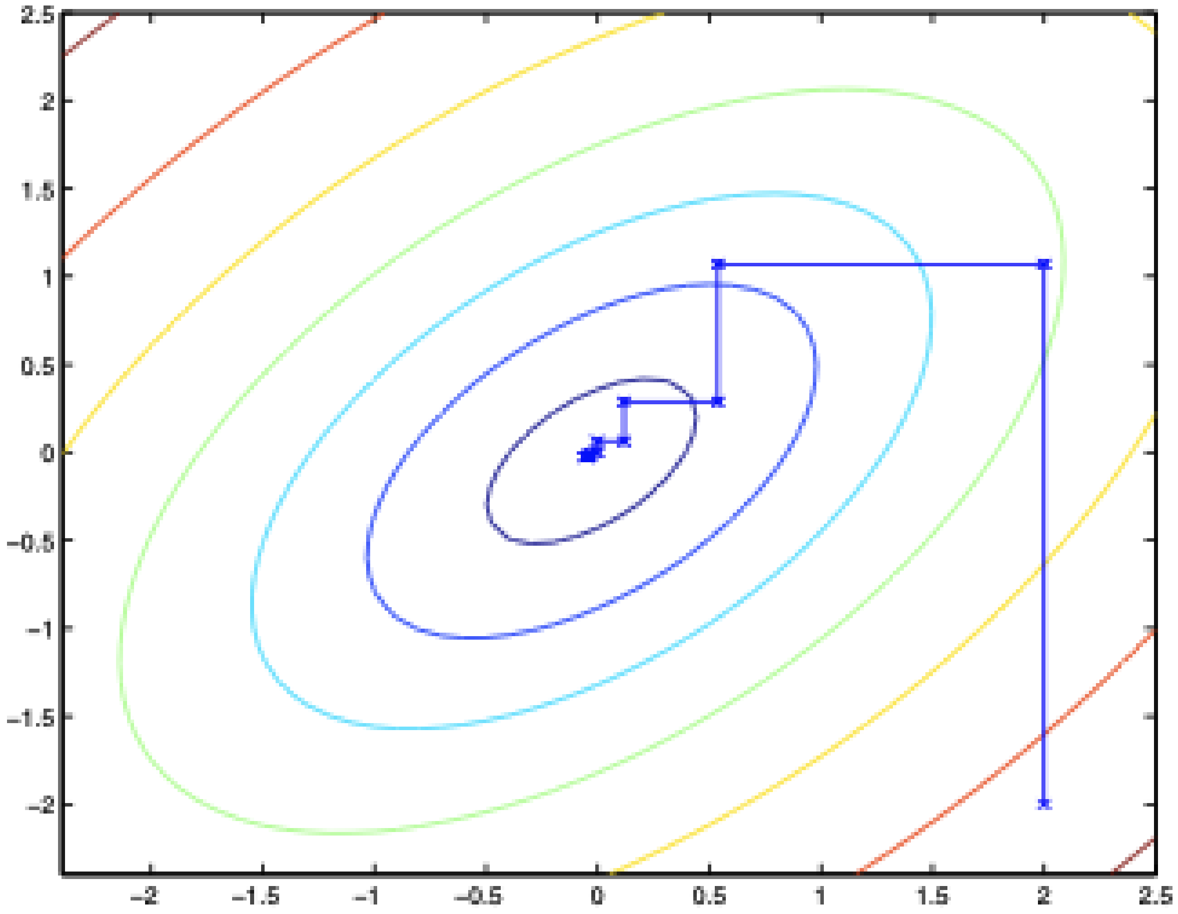
$$\text{subject to} \quad 0 \leq \lambda_n^* \leq C$$

$$\sum_{n=1}^{N} \lambda_n^* \cdot t_n = 0$$

Q: Can we solve this problem by using gradient descent?

Coordinate Ascent Algorithm: One Coordinate at One Time.

https://see.stanford.edu/materials/aimlcs229/cs229-notes3.pdf



At each step, the algorithm finds a minimum along the axis, we selected.

- Suppose given $\mathcal{D} : \{(x, y): x \in \mathbb{R}^M \ and \ t \in \{-1, 1\}\}$ and $\kappa(x, x')$ we defined a soft margin SVM dual function below.
  How could we find the optimal $\lambda_1, \lambda_2 \ \ldots, \lambda_n$?

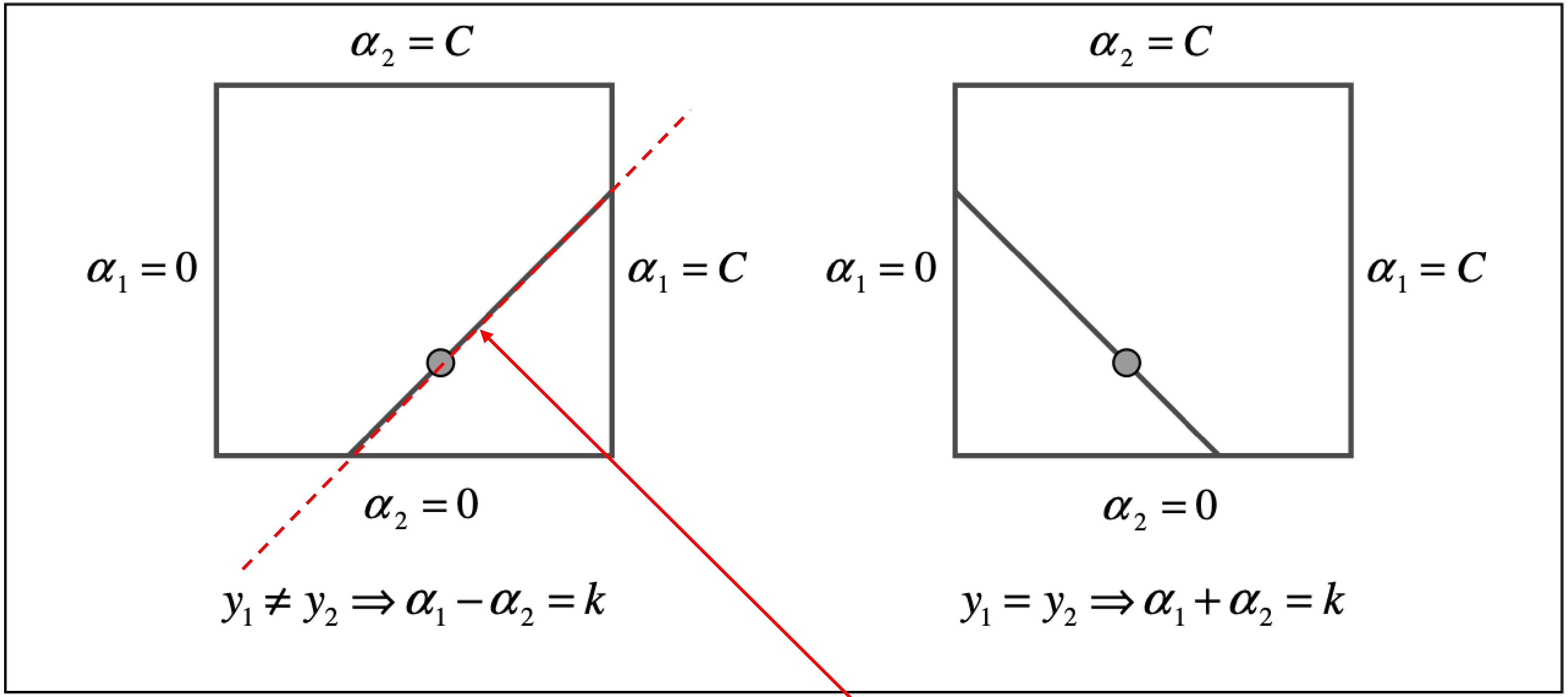
$$\lambda*_{n=1}^{N} = \arg\max_{\lambda*} \sum_{n=1}^{N} \lambda *_n - \frac{1}{2} \sum_{n=1}^{N} \sum_{m=1}^{N} \lambda_n^* \cdot t_n \lambda_m^* \cdot t_m \cdot \kappa(x_n, x_m)$$

$$\text{subject to} \quad 0 \leq \lambda_n^* \leq C$$

$$\sum_{n=1}^{N} \lambda_n^* \cdot t_n = 0$$

Let apply the concept of coordinate ascent to solve dual problem, but we have the two constraints.

In stead of updating one coordinate,
we will update a pair of two coordinates: (α_1, α_2) but others will be fixed.



$$y_1\alpha_1 + y_2\alpha_2 = -\sum_{i=3}^{N} y_i\alpha_i$$

$$y_1\alpha_1 + y_2\alpha_2 = k$$

$$L = \max(0, \alpha_2 - \alpha_1), \qquad H = \min(C, C + \alpha_2 - \alpha_1)$$

- we are going to find a minimum along the line.

Finding a minimum $\alpha_{2(new)}$ along the line : $y_1\alpha_1 - y_2\alpha_2 = k$

$$\boxed{\alpha_{2new} = \alpha_{2old} + \frac{y_2(E_1 - E_2)}{\eta}}$$

$$\eta = \kappa(x_1, x_1) + \kappa(x_2, x_2) - 2\kappa(x_1, x_2)$$

$$E_1 = (\sum_{n=1}^{N} \lambda_n^* t_n \kappa(x_n, x_1) + b) - y_1$$

$$E_2 = (\sum_{n=1}^{N} \lambda_n^* t_n \kappa(x_n, x_2) + b) - y_2$$

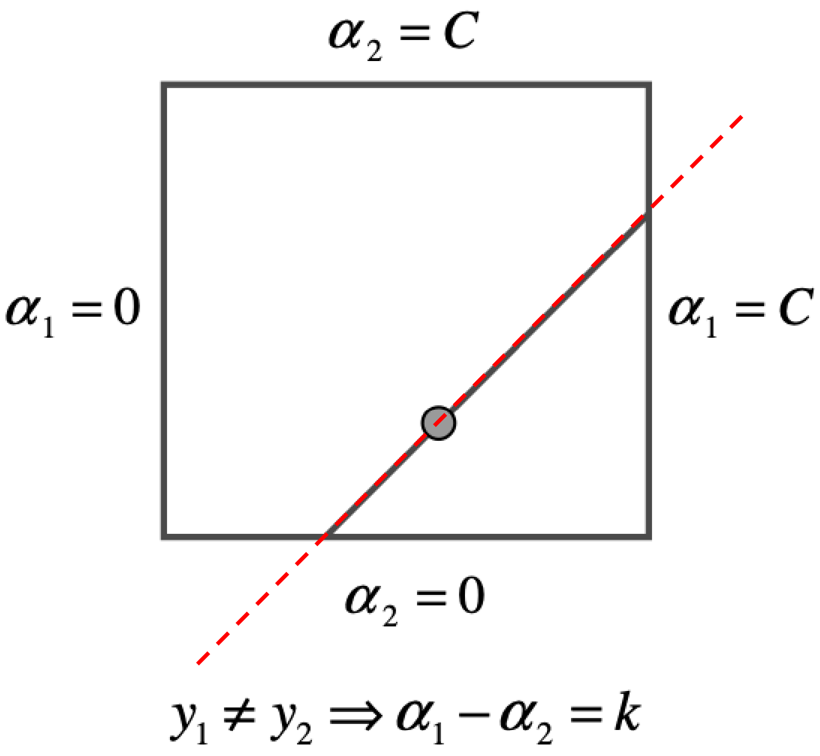
This is derived by $\frac{\mathrm{d}J'(\alpha_2)}{\mathrm{d}\alpha_2} = 0$

The new alpha point is the minimum solution along the line.

For derivations:

https://dsmilab.github.io/Yuh-Jye-Lee/assets/file/teaching/2017_machine_learning/SMO_algorithm.pdf

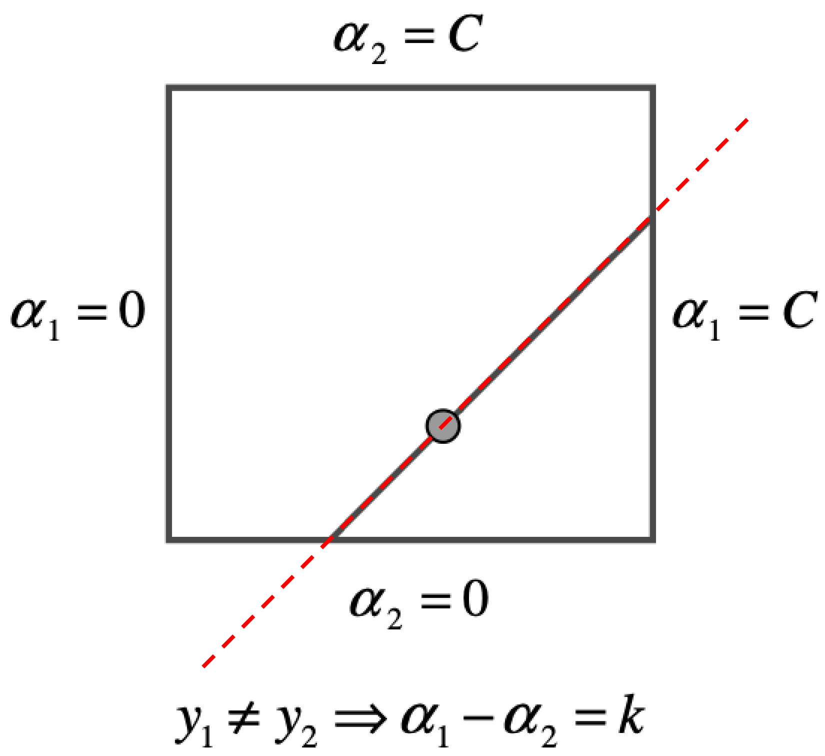
Clipping α_2 :
the minimum solution did not consider the bounds.
Hence, we need clipping



$$\alpha_2^{\text{new,clipped}} = \begin{cases} H & \text{if} \quad \alpha_2^{\text{new}} \geq H; \\ \alpha_2^{\text{new}} & \text{if} \quad L < \alpha_2^{\text{new}} < H; \\ \mathrm{L} & \text{if} \quad \alpha_2^{\text{new}} \leq L. \end{cases}$$

Once we find α_2, then we can update α_1. How?

$+\ \alpha_1$- $\alpha_2 = k$ we know this constraint.



SMO Algorithm

Repeat until KKT conditions are satisfied for all N training samples within certain tolerance (preset usually $10^{-3} \sim 10^{-2}$)

1. Pick two alphas (α_1 *and* α_2). + the KKT conditions are presented in the next slide.

2. Define the range L and H for α_2.

3. Compute $$\alpha_{2new} = \alpha_{2old} + \frac{y_2(E_1 - E_2)}{\eta}$$

4. Clipping by L and H.

5. Update α_1. And α_2 by $\alpha_1 - \alpha_2 = k$

All data samples must satisfy the KKT condition.

- $\lambda*_n = 0$
- Support vectors: $0 < \lambda*_n < C$
- Support vectors: $\lambda*_n = C$

$\longrightarrow$

"correctly labeled with a room to spare"

$$y_n(w^t x_n + b) \geq 1$$

$$y_n(w^t x_n + b) = 1$$

"unbound"

$$y_n(w^t x_n + b) \leq 1$$

"incorrectly labeled or lie within the margin"