Bài 3-Tiết 1: Mô hình Support Vector Machine

Trình bày: Ths Phạm Việt Anh

Viện Công nghệ HaUI
Trường Đại học Công nghiệp Hà Nội

Nội dung của bài giảng

1. Mô hình Support Vector Classification
2. Hàm Kernel trong SVM
3. Hàm mất mát trong SVM
4. Tổng kết

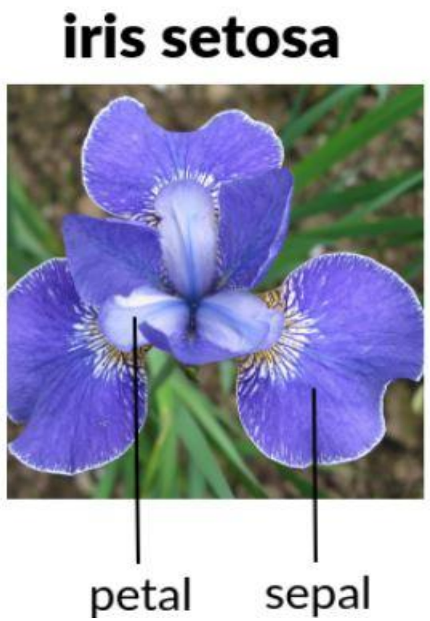
Tổng quan về SVM

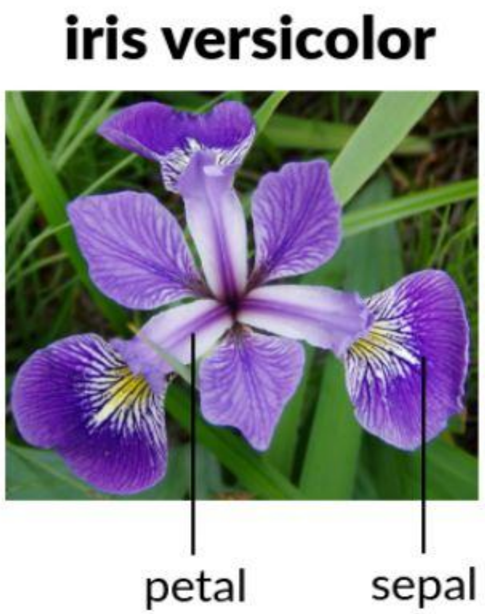
- ❑ Là một thuật toán khá hiệu quả trong lớp các bài toán phân loại nhị phân và dự báo của học có giám sát.
- ❑ SVM thường được sử dụng trong các tác vụ phân loại và dự báo, cũng như được nhiều công ty ứng dụng và triển khai trên môi trường production.
- ❑ Một số ứng dụng của SVM có thể kể tới:
  - ❖ Mô hình chẩn đoán bệnh. Dựa vào biến mục tiêu là những chỉ số xét nghiệm lâm sàng, thuật toán đưa ra dự báo về một số bệnh như tiểu đường, suy thận, máu nhiễm mỡ,…
  - ❖ Mô hình phân loại tin tức. Xác định chủ đề của một đoạn văn bản, phân loại cảm xúc văn bản, phân loại thư rác.
  - ❖ Mô hình phát hiện gian lận.
- ❑ Trước khi thuật toán CNN và Deep Learning bùng nổ thì SVM là lớp mô hình cực kì phổ biến trong phân loại ảnh.

Mô hình Support Vector Classification

❑ Xét bài toán sau: Cho 10 quan sát (đối tượng) với thuộc tính chiều dài của đài hoa. Dựa vào thuộc tính và các quan sát đó cần phân loại hoa Iris.

Objects	Sepal Length	Class
x_1	5.1	0
x_2	4.9	0
x_3	4.7	0
x_4	4.6	0
x_5	5.0	0
x_6	6.4	1
x_7	6.9	1
x_8	6.5	1
x_9	6.7	1
x_{10}	5.7	1

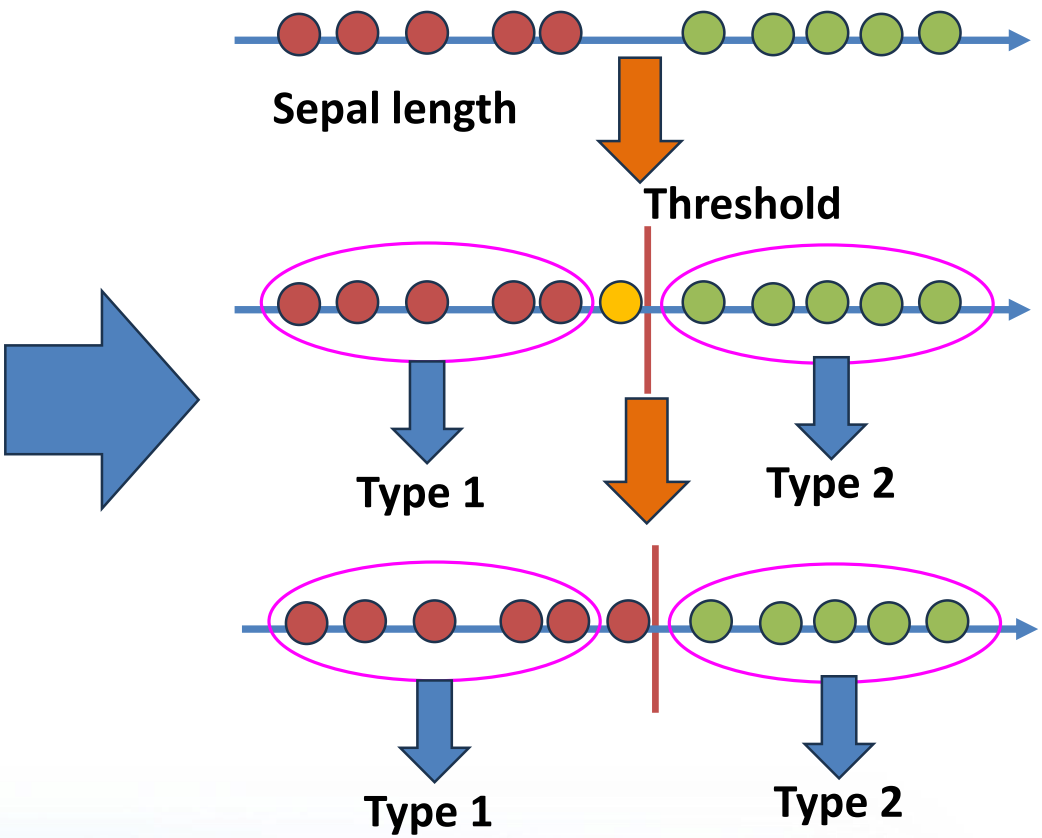




Mô hình Support Vector Classification

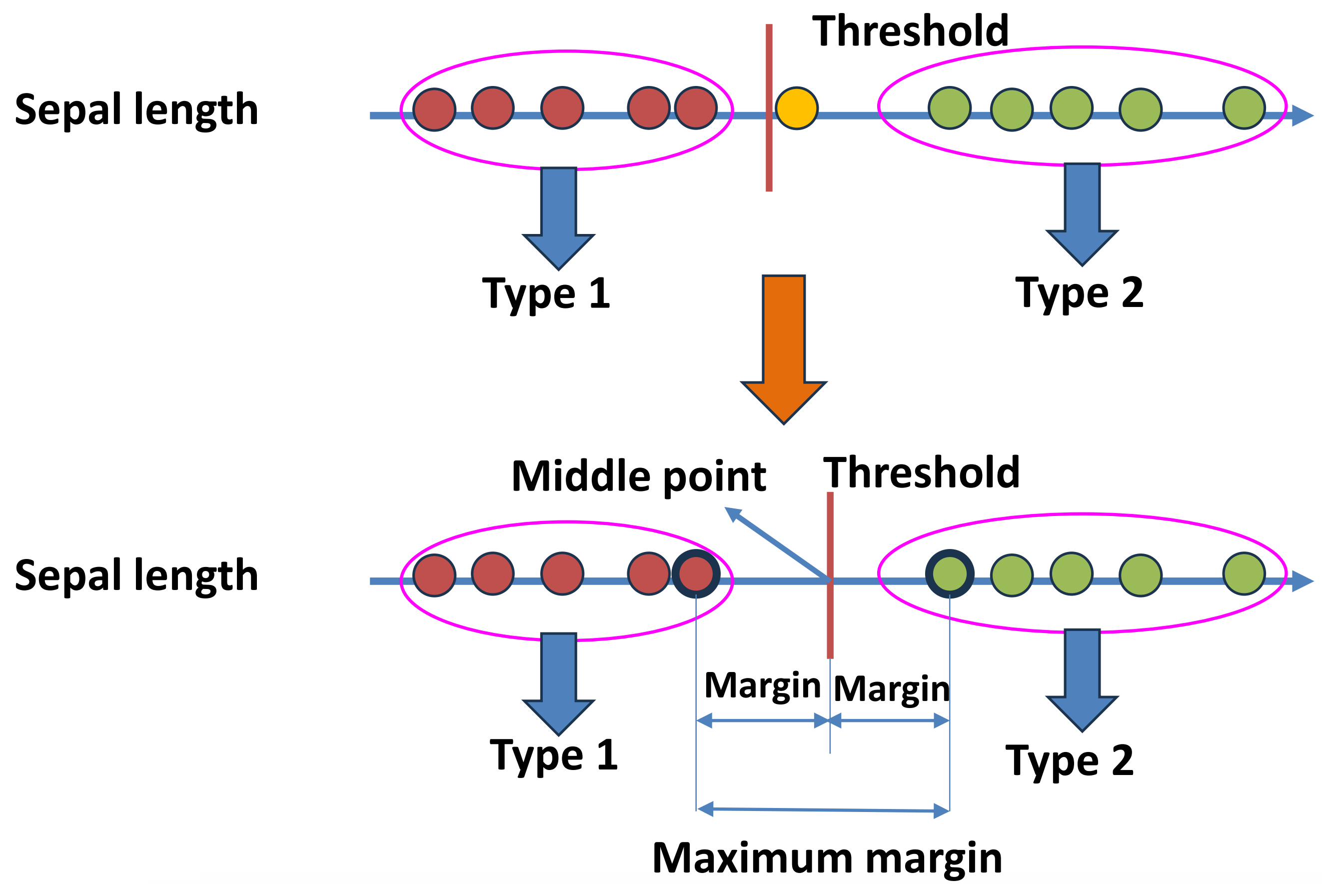
- Xét bài toán sau: Cho 10 quan sát (đối tượng) với thuộc tính chiều dài của đài hoa. Dựa vào thuộc tính và các quan sát đó cần phân loại hoa Iris.

Objects	Sepal length	Class
x_1	5.1	0
x_2	4.9	0
x_3	4.7	0
x_4	4.6	0
x_5	5.0	0
x_6	6.4	1
x_7	6.9	1
x_8	6.5	1
x_9	6.7	1
x_{10}	5.7	1



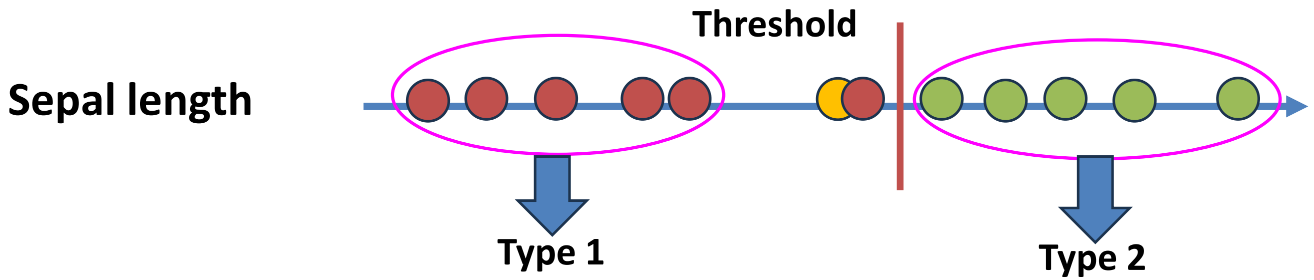
Mô hình Support Vector Classification

❑ Xét một trường hợp sau:

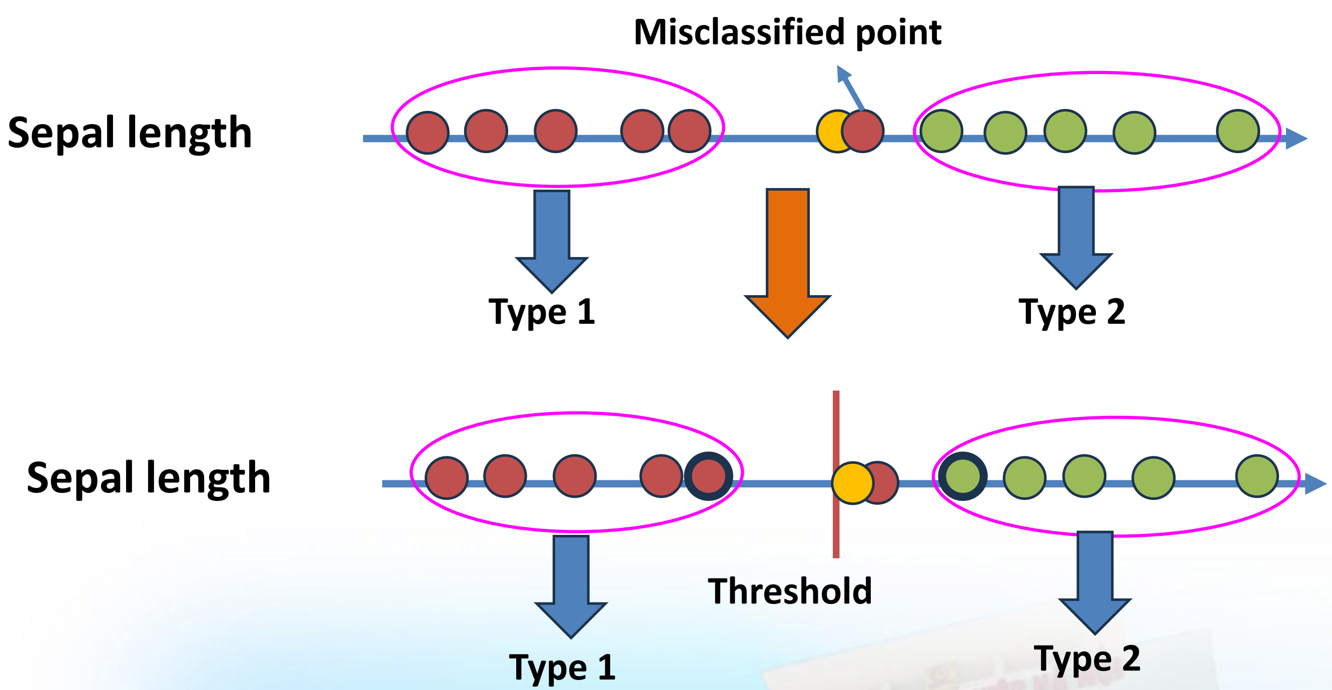


Mô hình Support Vector Classification

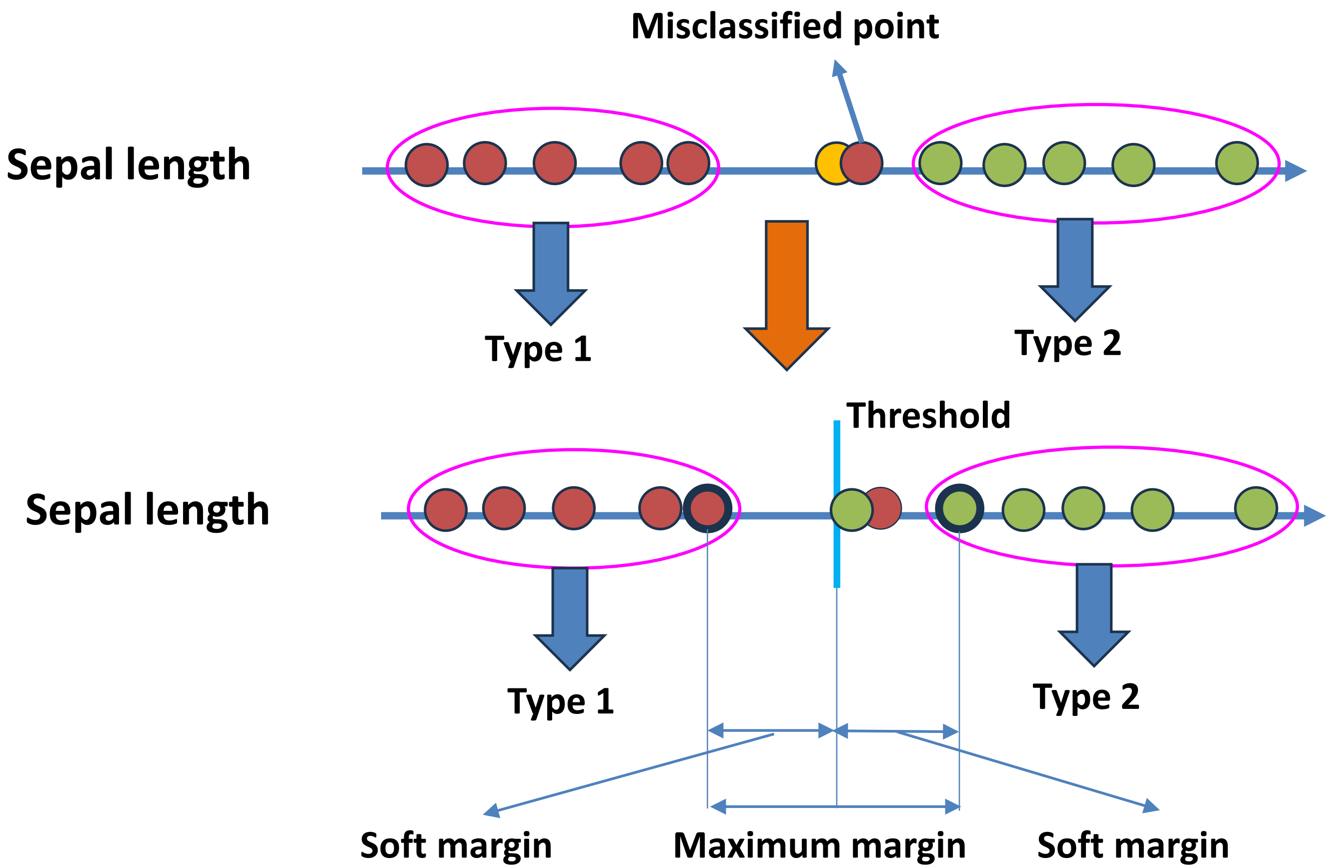
❑ Tính không hiệu quả của phương pháp lề lớn nhất (Maximum margin):



❑ Để xử lý vấn đề này sử dụng thêm phương pháp Allowing misclassification:

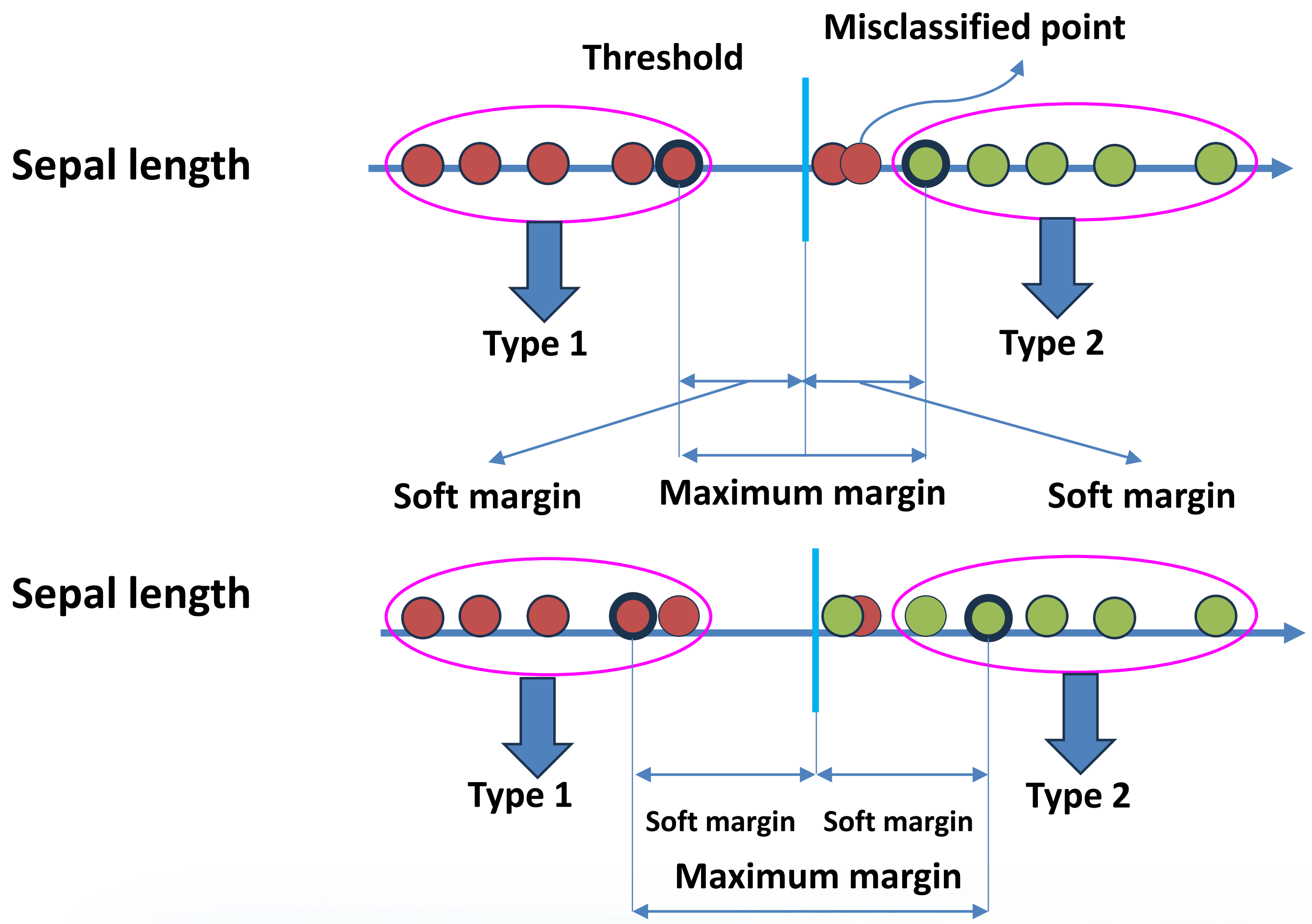


Mô hình Support Vector Classification



Mô hình Support Vector Classification

❑ Cần phải lựa chọn Soft margin thế nào cho hiệu quả:



❖ Sử dụng phương pháp Cross-Validation để lựa chọn Soft margin hiệu quả.

Mô hình Support Vector Classification

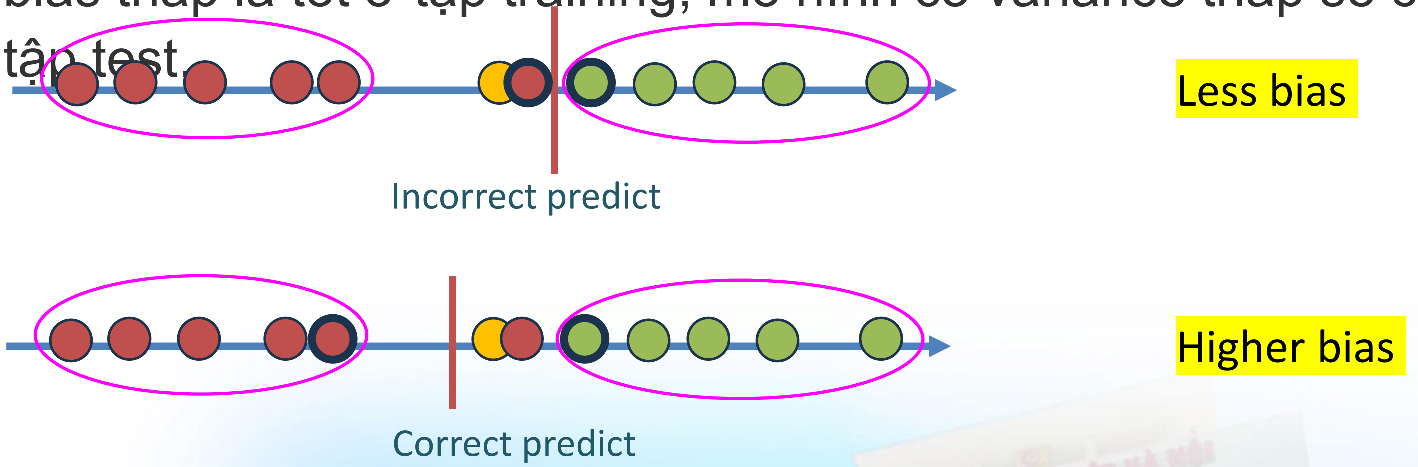
❑ Phương pháp Cross-Validation (Kiểm định chéo)

❖ Thay đổi lề ở các khoảng cách khác nhau để tìm lề nào tốt ở cả training set và test set.

❑ Việc xác định Soft Margin để phân tách 2 nhóm dữ liệu được gọi là SVC

❖ Hai điểm lân cận (không tính điểm phân loại sai) được gọi là các Support Vector.

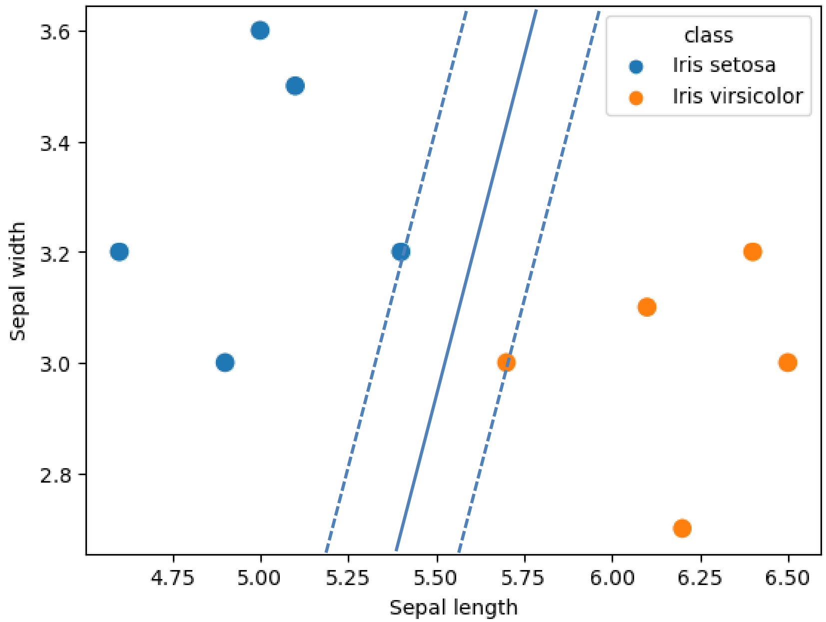
❖ Việc chọn Soft margin cho phép cân bằng sai số bias và variance: Sai số bias thấp là tốt ở tập training, mô hình có variance thấp sẽ có kết quả tốt ở tập test.

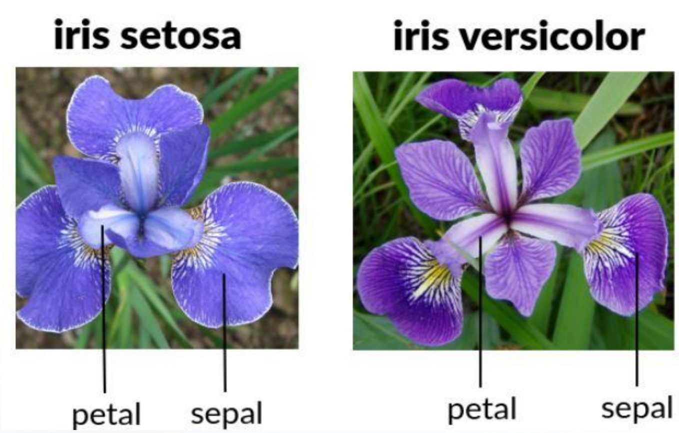


Mô hình Support Vector Classification

❑ Xét một ví dụ với 2 thuộc tính đầu vào

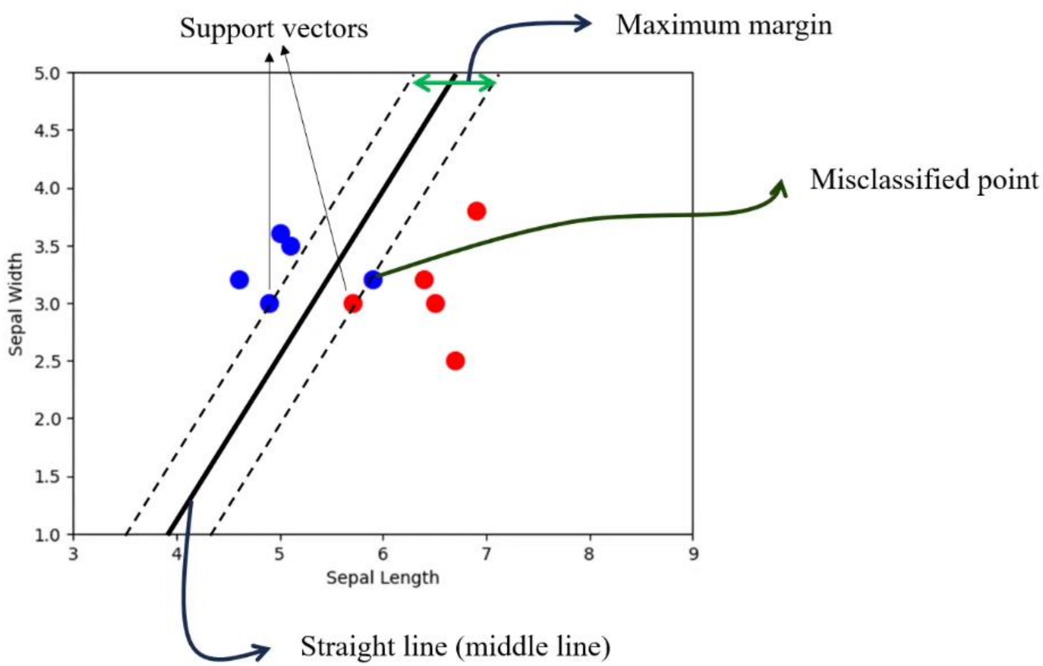
Objects	Sepal length	Sepal width	Class
x_1	5.1	3.5	0
x_2	4.9	3.0	0
x_3	4.7	3.2	0
x_4	4.6	3.2	0
x_5	5.0	3.6	0
x_6	6.4	3.2	1
x_7	6.9	3.8	1
x_8	6.5	3.0	1
x_9	6.7	2.5	1
x_{10}	5.7	3.0	1



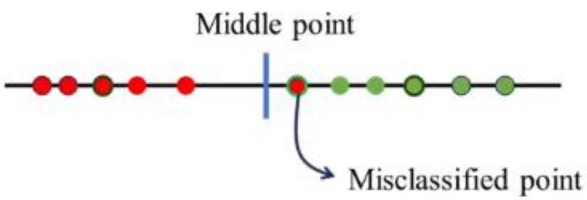


Mô hình Support Vector Classification

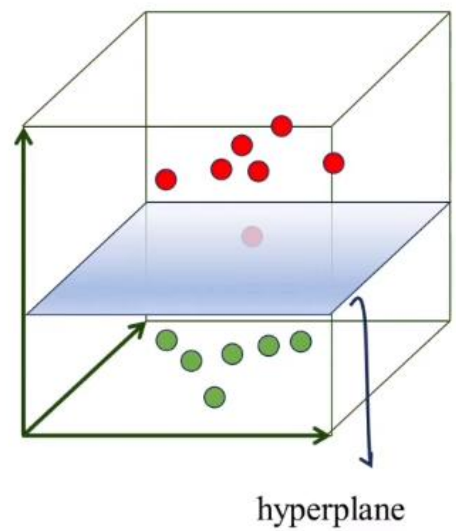
- ❑ Đối với dữ liệu 1 chiều thì chỉ cần một điểm dữ liệu để phân tách, đối với dữ liệu hai chiều thì sử dụng một đường thẳng và từ ba chiều là 1 hyperplane.



SVC là một đường thẳng



SVC là một điểm



SVC là một hyperplane

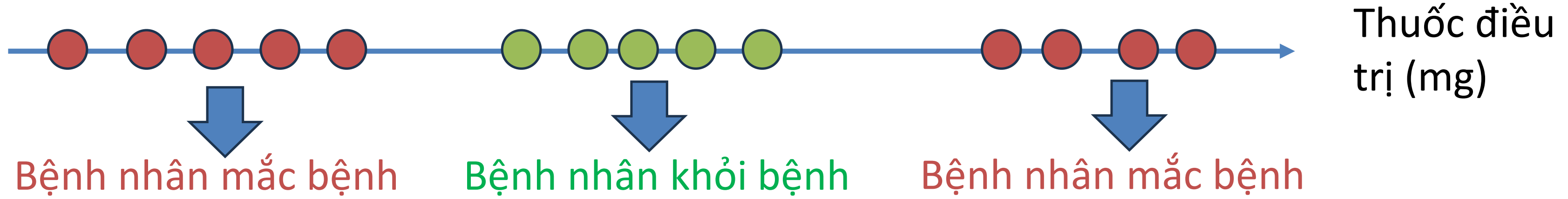
Mô hình Support Vector Classification

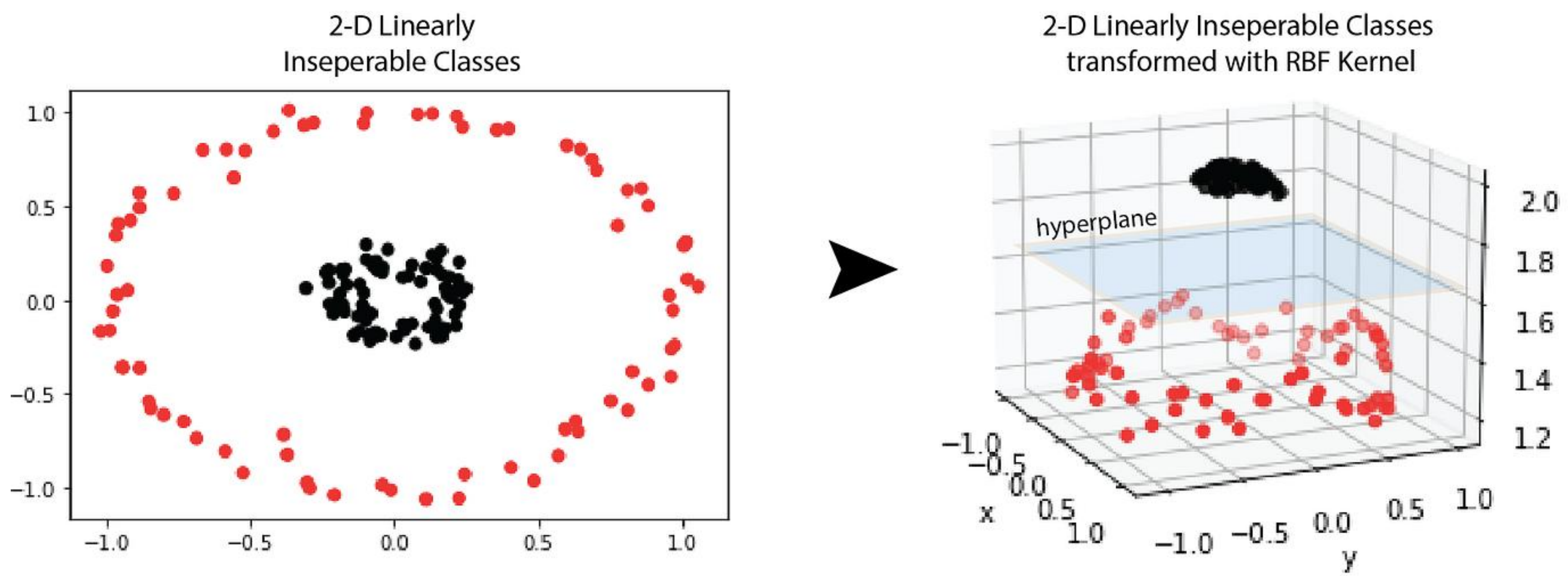
❑ Nhận xét:

- ❖ Phương pháp chỉ xử lý được trên không gian tuyến tính, do đó phương pháp này được gọi riêng là Linear Support Vector Classifier (LSVC).
- ❖ Phương pháp có thể xử lý trên không gian nhiều chiều, số điểm ít và số thuộc tính cao.
- ❖ Sử dụng lề mềm (Soft margin) có thể tránh được hiện tượng overfitting.
- ❖ Hiệu quả với phân loại nhị phân (binary).
- ❖ Dễ dàng tối ưu để tìm 1 hyperplane phù hợp.
- ❖ Tuy nhiên, phương pháp LSVC không thể xử lý được với các bài toán không tuyến tính.

Mô hình Support Vector Classification

- ❑ Xét trường hợp không tuyến tính:

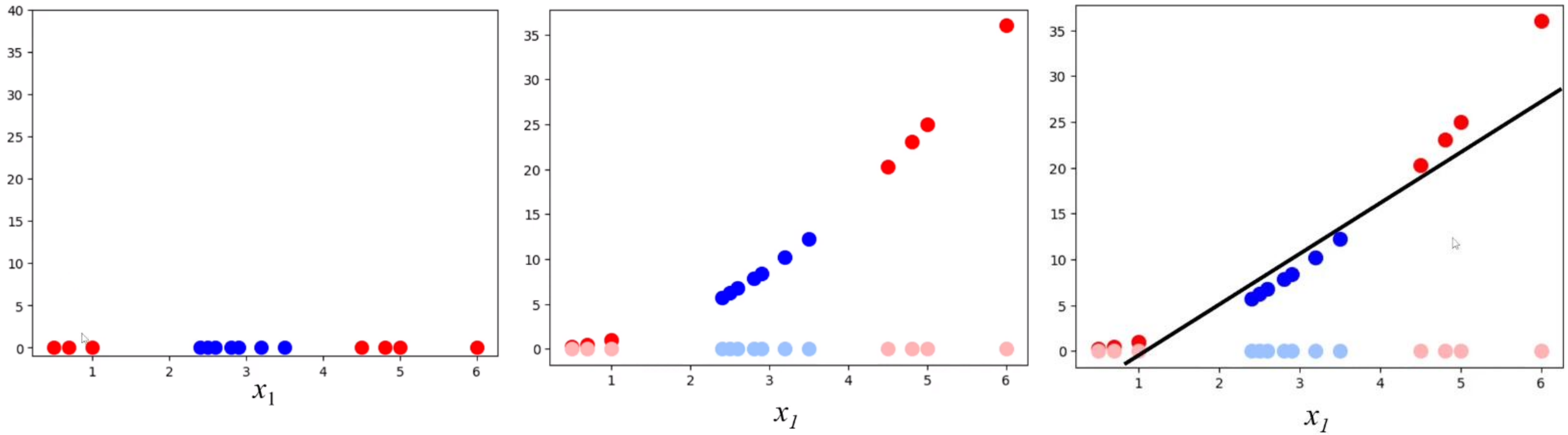




Hàm kernel trong SVM

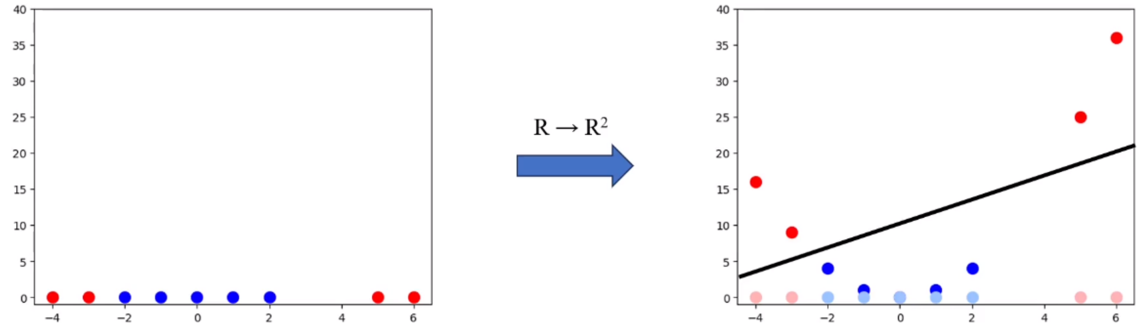
- Để xử lý trong trường hợp dữ liệu không tuyến tính, phương pháp LSVC sử dụng một kỹ thuật được gọi là Kernel trick.
- Kernel trick là 1 kỹ thuật chuyển đổi dữ liệu từ số chiều thấp sang một không gian có số chiều cao hơn.

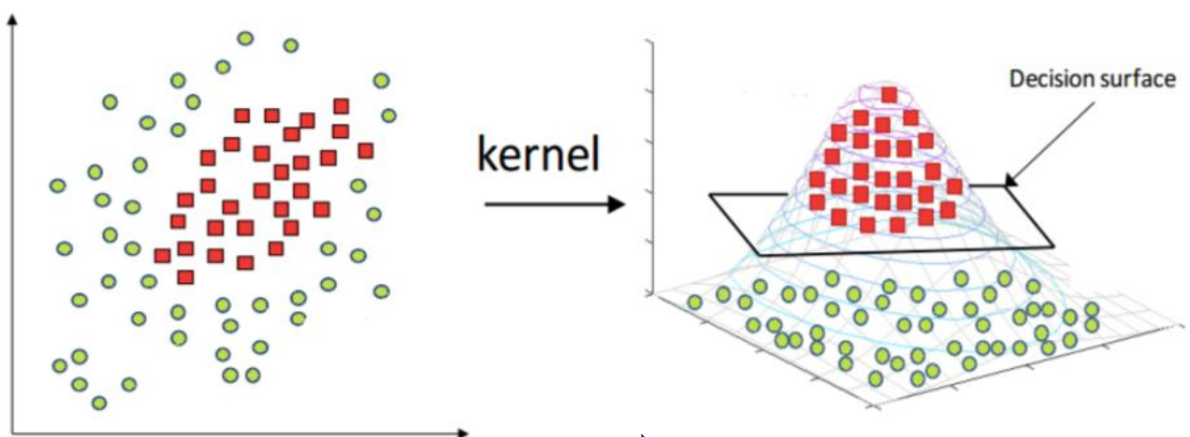
$$x_2 = (x_1)^2$$



Hàm kernel trong SVM

- Giải thuật SVM không sử dụng dữ liệu gốc mà sử dụng trên dữ liệu đã được biến đổi để tìm kiếm hyperplane cho việc phân tách.

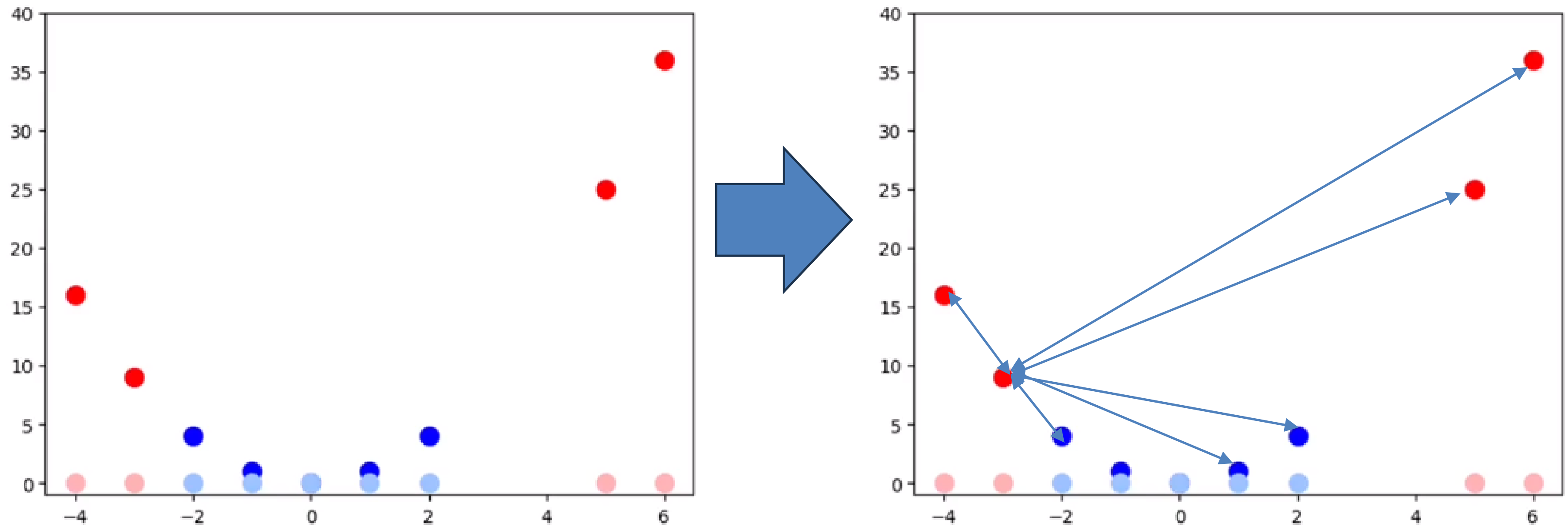




Giá trị đầu vào ban đầu x → Ánh xạ giá trị bởi một hàm $\phi(x)$

Hàm kernel trong SVM

- Để tối ưu mô hình và tìm kiếm được một hyperplane, SVM cần tính toán các quan hệ giữa toàn bộ các cặp điểm dữ liệu (được biến đổi) thông qua tích vô hướng: $\phi(\mathbf{x})^T\,\phi(\mathbf{z})$.



- ❖ Tuy nhiên điều này sẽ rất khó để thực hiện với các bộ dữ liệu nhiều chiều.

Hàm kernel trong SVM

- Hàm kernel là một trong các phương pháp để giải quyết vấn đề nêu trên khi giúp tính toán được những giá trị ở không gian có số chiều lớn hơn bằng cách chỉ sử dụng input ở không gian có số chiều thấp hơn.

$$\mathbf{x} \rightarrow \phi(\mathbf{x})$$
$$\mathbf{z} \rightarrow \phi(\mathbf{z})$$

Chuyển đổi

$$\phi(\mathbf{x})^T \phi(\mathbf{z})$$

Phương pháp ban đầu

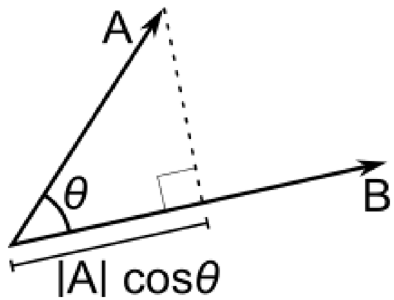
$$Kernel\ function\ K: K(\mathbf{x}, \mathbf{z}) = \phi(\mathbf{x})^T\ \phi(\mathbf{z})$$

- Hàm kernel được định nghĩa từ 2 biến $\mathbf{x}$ và $\mathbf{z}$ ban đầu nhưng tính được giá trị của không gian cao hơn, nghĩa là nó sẽ đi tìm $\phi(\mathbf{x})^T \phi(\mathbf{z})$ mà không cần qua bước chuyển đổi không gian.

Hàm kernel trong SVM

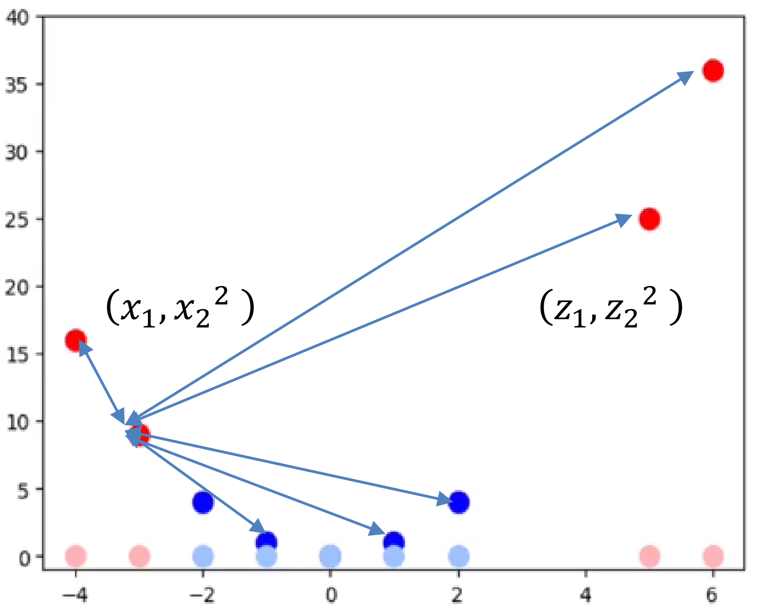
- Xét không gian hai chiều với 2 vector $\mathbf{x} = (x_1, x_2)$ và $\mathbf{z} = (z_1, z_2)$.

Tích vô hướng của hai vector $\mathbf{x}$ và $\mathbf{z}$ là:



$\mathbf{x}^T\mathbf{z} = x_1 * z_1 + x_2 * z_2.$ $\quad\quad$ $\mathbf{x}^T\mathbf{z} = \|\mathbf{x}\| * \|\mathbf{z}\| * \cos\theta$

$$\phi(\mathbf{x})^T \phi(\mathbf{z}) = x_1 * z_1 + x_2^2 * z_2^2$$



Polinomial kernel: $(\Upsilon \mathbf{x}^T\mathbf{z} + r)^d$

Cho $\Upsilon = 1$, $r = 1/2$, $d = 2$ thì:

$$K(\mathbf{x}, \mathbf{z}) = \left(x_1 z_1 + x_2 z_2 + \frac{1}{2}\right)^2$$

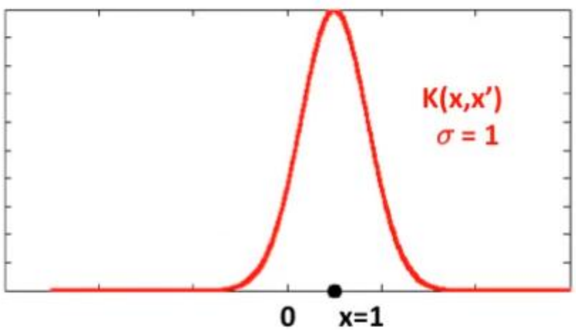
$$= (x_1 z_1)^2 + 2x_1 z_1 x_2 z_2 + x_1 z_1 + x_2 z_2 + (x_2 z_2)^2 + \frac{1}{4}$$

$$= \left(x_1^2, \sqrt{2}x_1 x_2, x_1, x_2, \frac{1}{2}\right)^T \left(z_1^2, \sqrt{2}z_1 z_2, z_1, z_2, \frac{1}{2}\right)$$

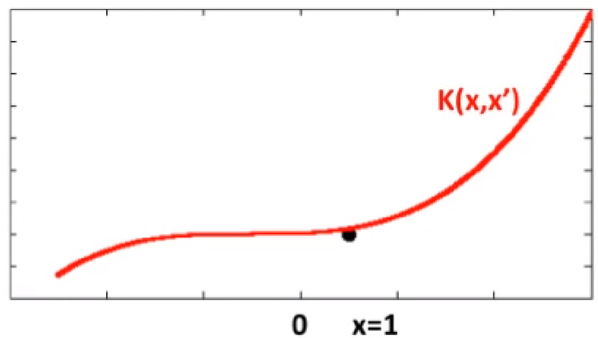
Hàm kernel trong SVM

- Một số hàm kernel thông dụng:

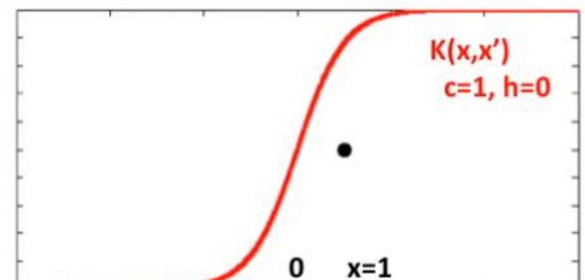
Tên	Công thức	`kernel`	Thiết lập hệ số
linear	$\mathbf{x}^T\mathbf{z}$	`'linear'`	không có hệ số
polynomial	$(r + \gamma\mathbf{x}^T\mathbf{z})^d$	`'poly'`	d: `degree`, γ: `gamma`, r: `coef0`
sigmoid	$\tanh(\gamma\mathbf{x}^T\mathbf{z} + r)$	`'sigmoid'`	γ: `gamma`, r: `coef0`
rbf	$\exp(-\gamma\|\|\mathbf{x} - \mathbf{z}\|\|_2^2)$	`'rbf'`	$\gamma > 0$: `gamma`



Radial basic function



Polynomial

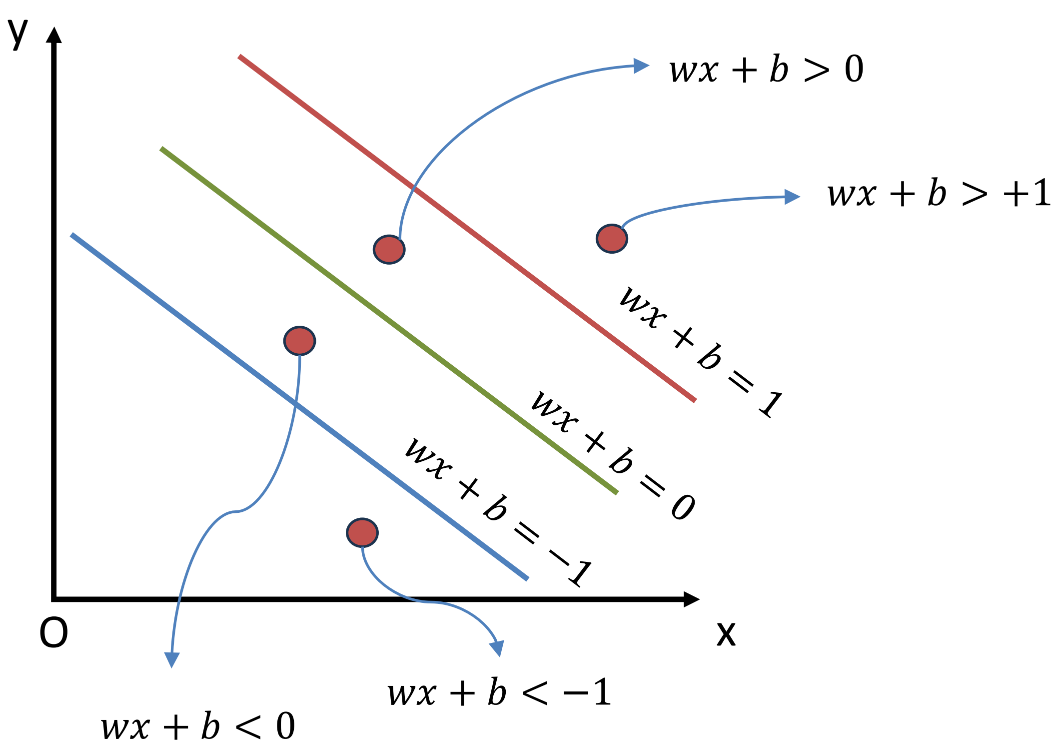


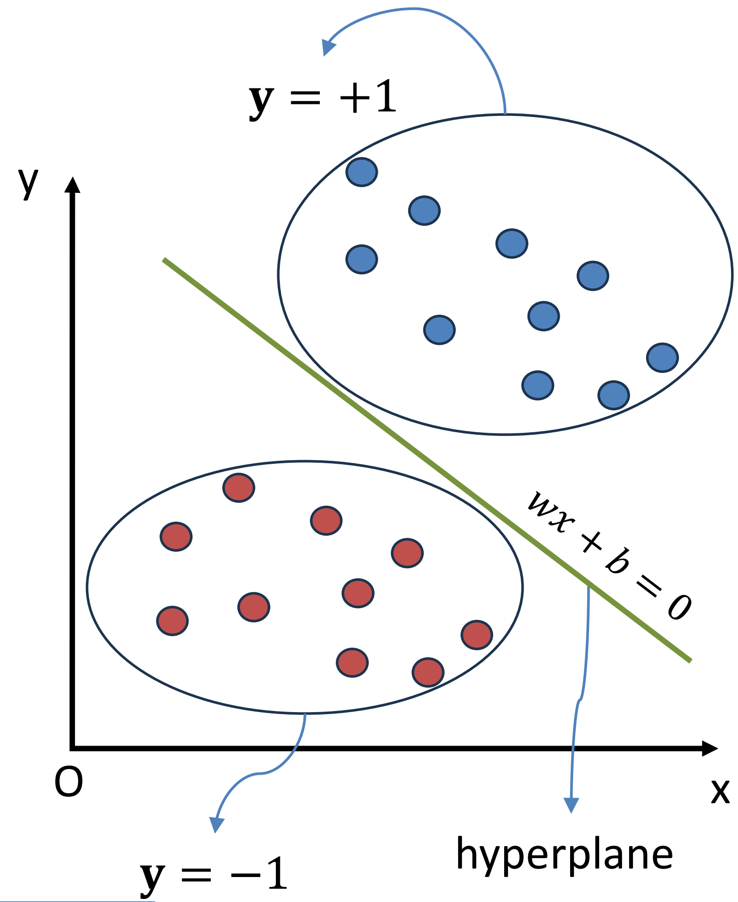
Sigmoid

$$tanh(x) = \frac{(e^x - e^{-x})}{(e^x + e^{-x})}$$

Hàm mất mát trong SVM

$$\mathbf{w}^T\mathbf{x} + b = 0$$
$$\mathbf{w} = (w_1, w_2)$$

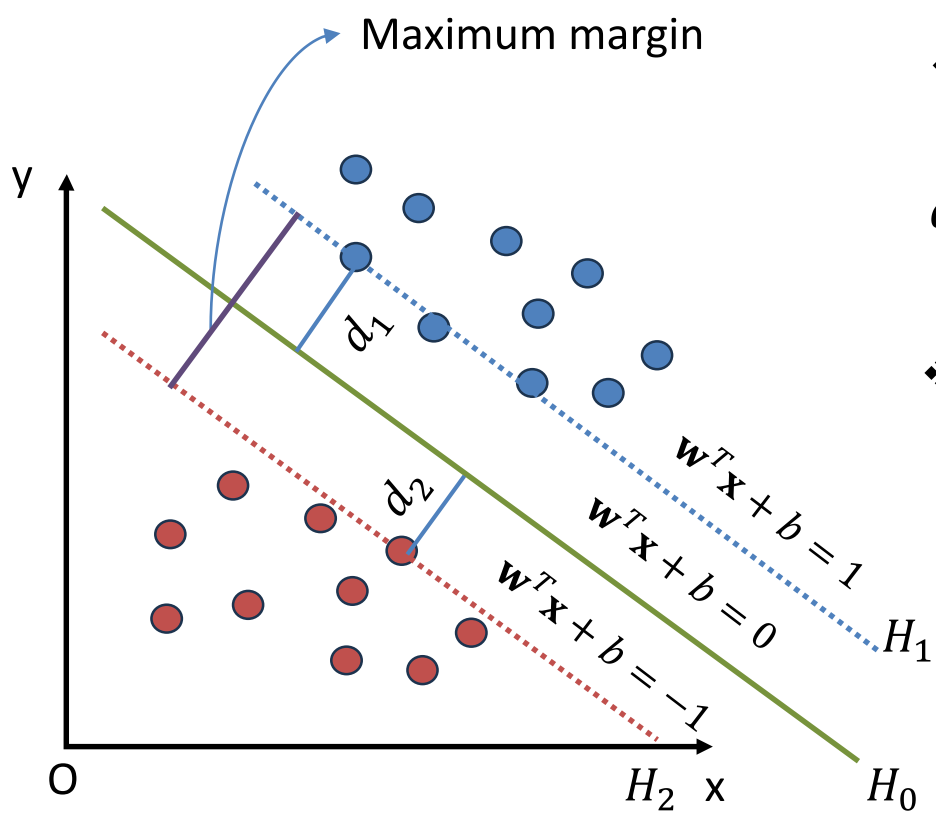




Satisfy:

- ✓ $\mathbf{w}^T\mathbf{x}_i + b \geq 1 \; for \; y_i = +1$
- ✓ $\mathbf{w}^T\mathbf{x}_i + b \leq -1 \; for \; y_i = -1$

Hàm mất mát trong SVM



❖ Khoảng cách từ H_0 đến H_1:

$$d_1 = \frac{|\mathbf{w}^T\mathbf{x}+b|}{\|\mathbf{w}\|} = \frac{1}{\|\mathbf{w}\|} = d_2$$

❖ Từ điều kiện đưa ra, mong muốn đặt ra là không tồn tại 1 điểm nằm giữa hai đường phân cách

- ✓ $\mathbf{w}^T\mathbf{x} + b \geq 1 \; for \; \mathbf{y} = +1$
- ✓ $\mathbf{w}^T\mathbf{x} + b \leq -1 \; for \; \mathbf{y} = -1$

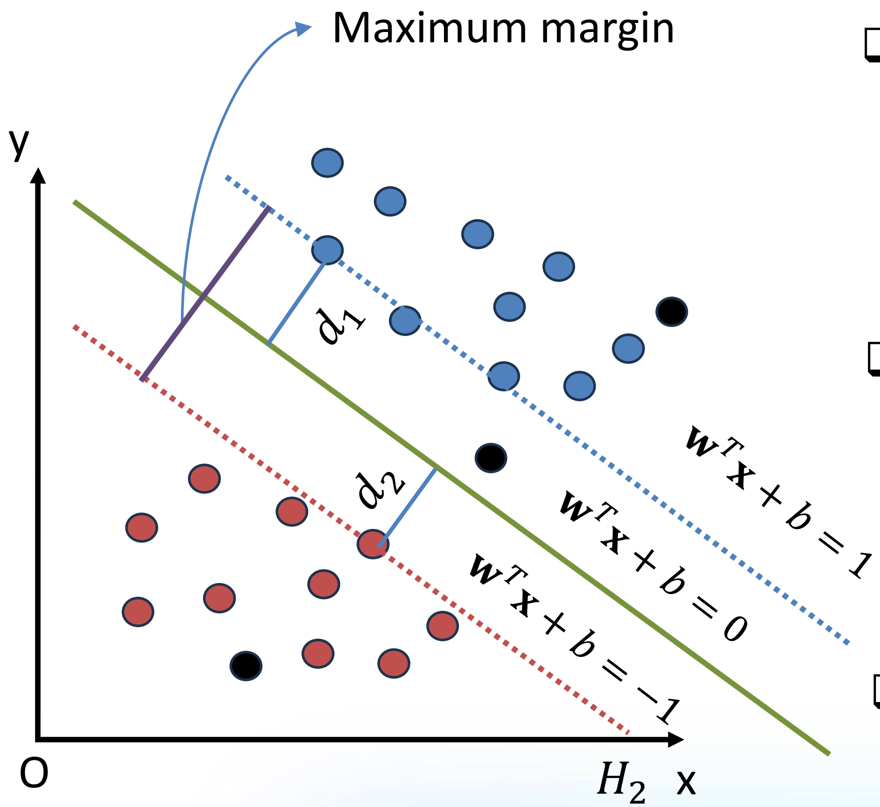
$$\boxed{\Longrightarrow \mathbf{y}(\mathbf{w}^T\mathbf{x} + b) \geq 1}$$

❖ Do đó, cần cực đại hóa giá trị maximum margin $\frac{2}{\|\mathbf{w}\|}$ theo điều kiện $\mathbf{y}(\mathbf{w}^T\mathbf{x} + b) \geq 1$.

Hàm mất mát trong SVM

- Để mô hình học cách phân loại nhãn sao cho đúng

$$\mathcal{L}(\mathbf{w}, b)=\sum_{i=1}^{n}\left(1 - y_i(\mathbf{w}^T\mathbf{x}_i + b)\right)$$



- Để đảm bảo cả yếu tố độ rộng cho maximum margin cần bổ sung thành phần:

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

- Để đảm bảo sự cân bằng giữa hai thành phần sử dụng thêm đại lượng phạt:

$$\mathcal{L}(\mathbf{w}, b, \boldsymbol{\lambda})= \frac{\|\mathbf{w}\|^2}{2} + \sum_{i=1}^{n}\lambda_i\left(1 - y_i(\mathbf{w}^T\mathbf{x}_i + b)\right)$$

- Khi λ lớn thì SVM tập trung làm nhỏ tổn thất ở thành phần thứ 2 và ngược lại.

Hàm mất mát trong SVM

❑ Một số chú ý:

❖ Khi chấp nhận việc phân loại dựa trên nhãn đúng thì sẽ kéo theo phần margin bị nhỏ đi

❖ Khi chấp nhận việc margin lớn thì đánh đổi việc phân loại sai.

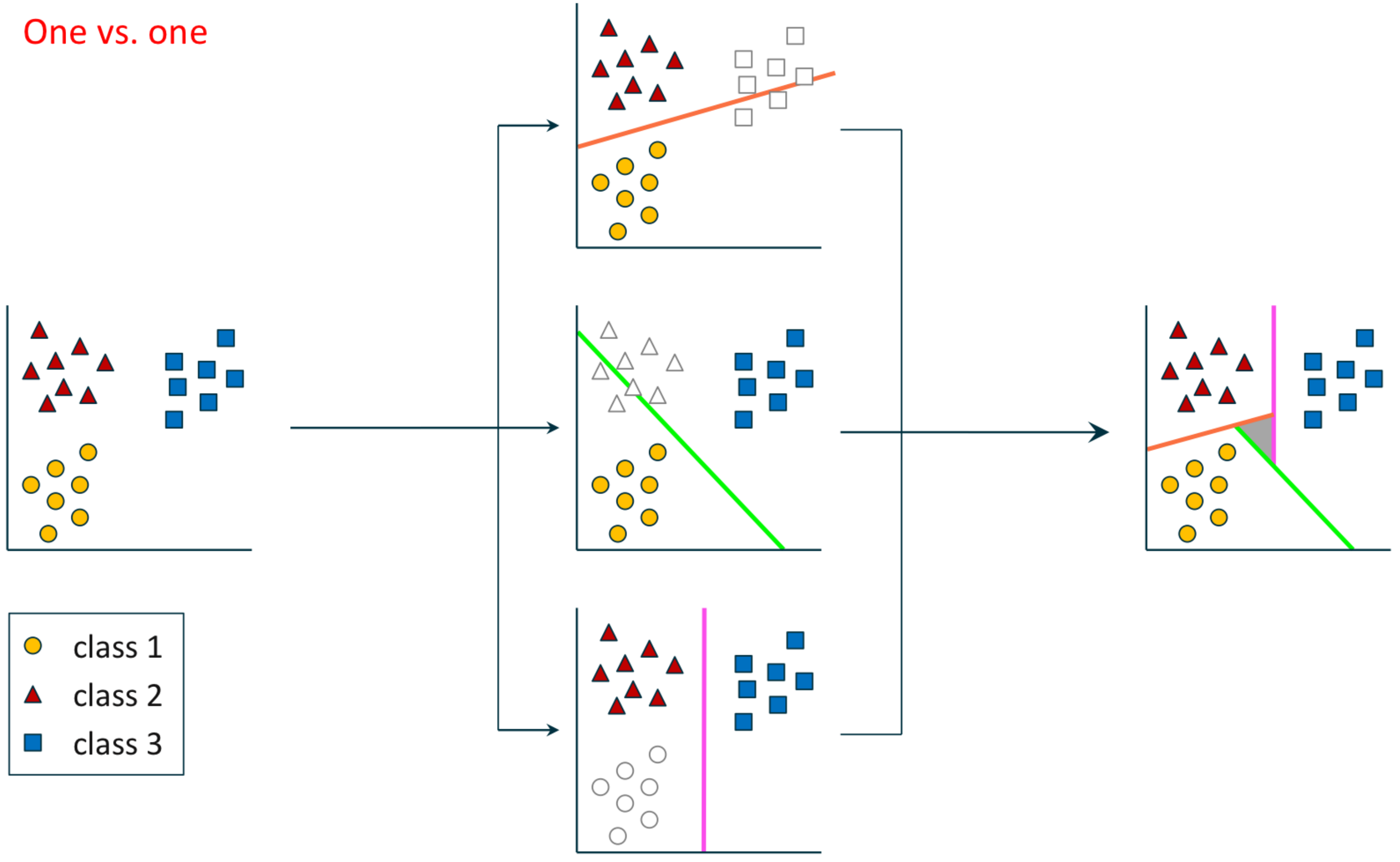
→ Do đó, cần phải cân bằng giữa hai thành phần này.

❖ Phần bình phương trong thành phần trọng số giúp tăng vai trò của việc học trong hàm mất mát nếu muốn tang margin lớn.

Ưu điểm	Nhược điểm
Sử dụng rất tốt với các không gian nhiều chiều	Không phù hợp với các dữ liệu có nhiều bản ghi
Làm việc trong cả trường hợp dữ liệu là tuyến tính hoặc không tuyến tính.	Thời gian tính toán phức tạp
Tránh overfitting khi sử dụng đại lượng phạt dựa trên giá trị điều chuẩn hoặc λ	Hạn chế trên các bộ dữ liệu đa lớp
	Dữ liệu cần phải tiền xử lý

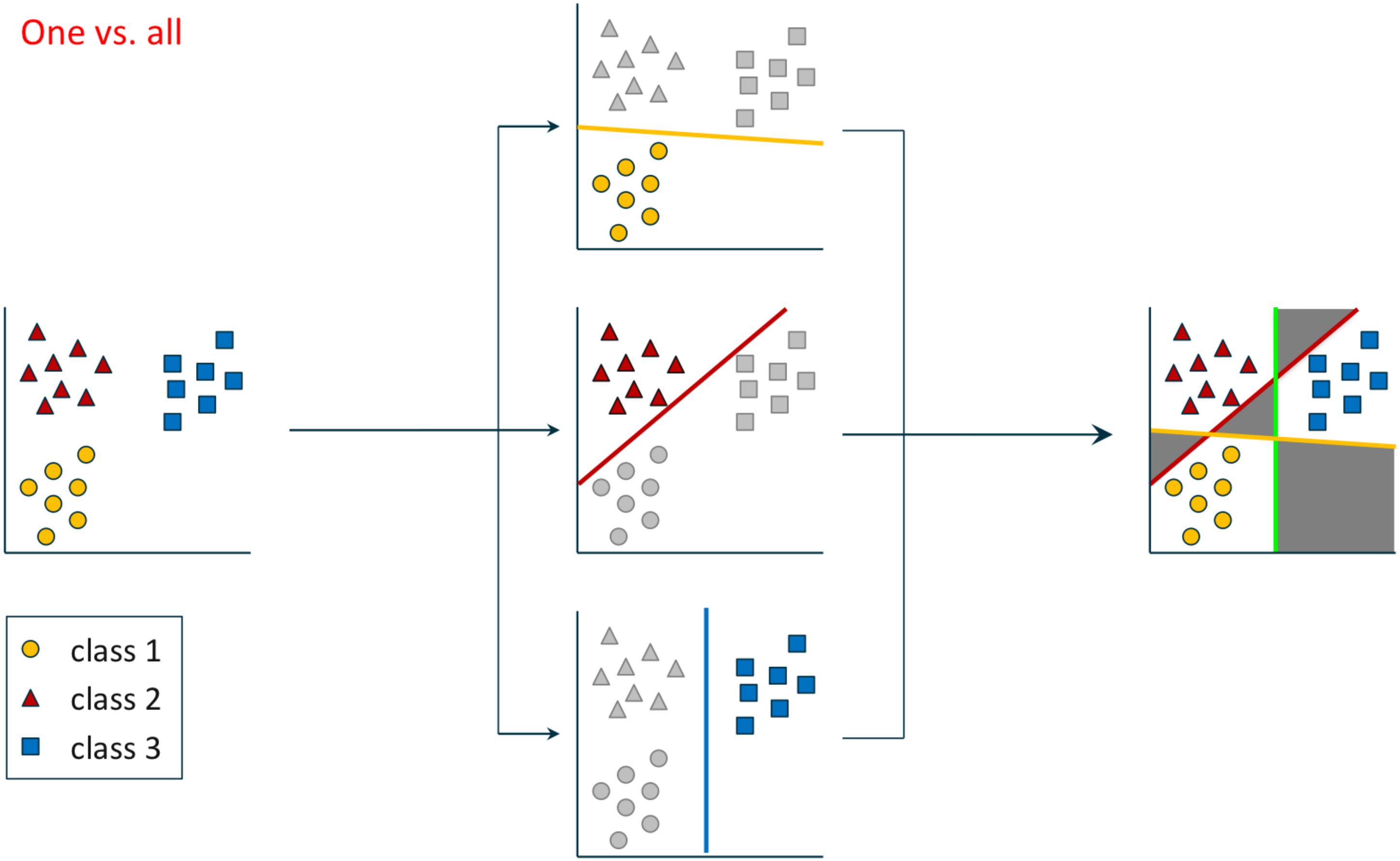
SVM trong bài toán đa lớp

- One vs One



SVM trong bài toán đa lớp

❑ One vs All



Bài tập về nhà

- ❑ Xem lại toàn bộ bài giảng trên lớp, chú ý các kiến thức cơ bản về mô hình SVM.
- ❑ Tìm hiểu quá trình chạy mô hình SVM thông qua thư viện Sklearn trên một bộ dữ liệu mẫu.
- ❑ Đọc trước về một số độ đo sử dụng để đánh giá độ chính xác trên một mô hình phân lớp.
- ❑ Yêu cầu mang máy tính để phục vụ quá trình thực hành vào buổi sau.


thank
you