

Máy học véc-tơ hỗ trợ (SVM)

Trường hợp không khả tách tuyến tính

13/3/2021

Đối tượng

- Sinh viên Khoa học máy tính

Mục tiêu

- Hiểu được bài toán SVM trường hợp không khả tách tính tuyến tính và đưa nó về bài toán quy hoạch toàn phương
- Áp dụng gói thư viện để huấn luyện mô hình máy học SVM

Tài liệu

- Slide này
- Slides 2 - SVM
- Giáo trình nguyên lý máy học

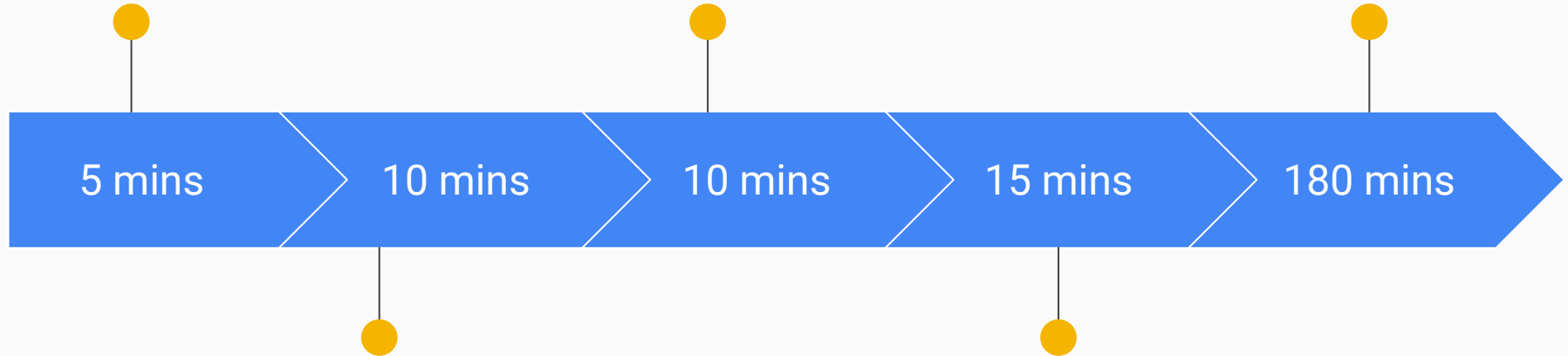
Kế hoạch

Nhắc lại bài toán SVM

Tìm hiểu thư viện tối ưu

Bài tập

(Không khả tách tuyến tính)



Đưa bài toán SVM về
bài toán Quy hoạch
toàn phương

Cài đặt SVM bằng thư
viện tối ưu

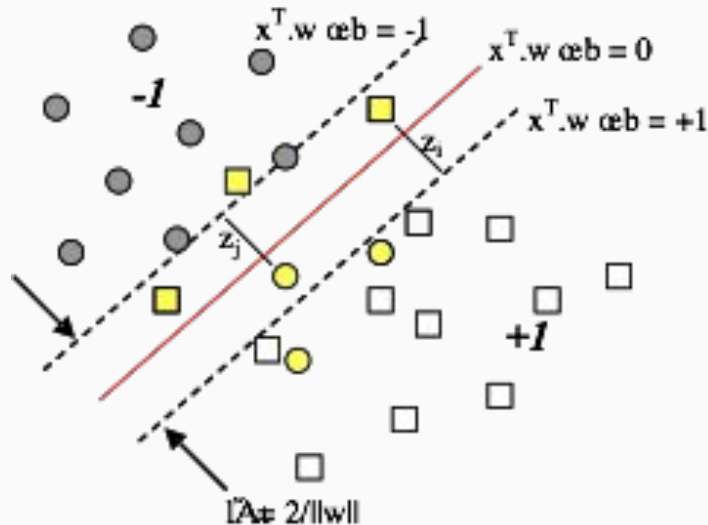
Bài toán SVM

Trường hợp không thể tách được (mỗi lớp nằm về 1 phía)

- Vẫn còn sót lại 1 số phần tử nằm sai phía so với đường thẳng hỗ trợ (các phần tử màu vàng)
- Các ràng buộc không thể thoả mãn

Ràng buộc: $y^{(i)}(w_1 x^{(i)}_1 + w_2 x^{(i)}_2 - b) \geq +1$

Về trái của ràng buộc < 1 .



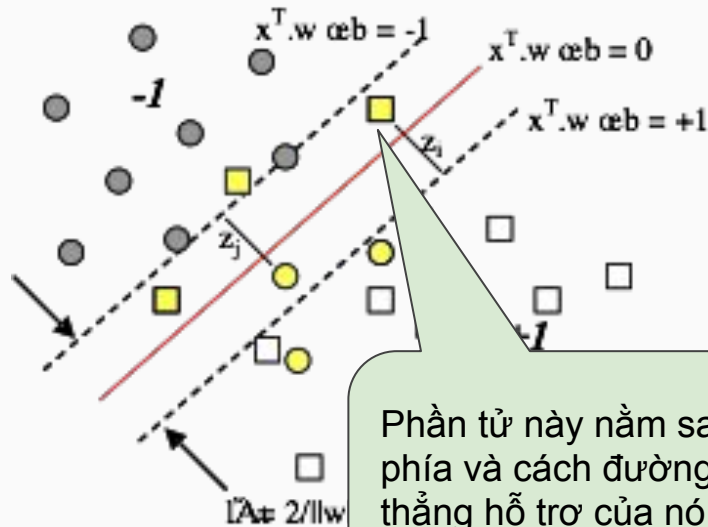
Bài toán SVM

Trường hợp không thể tách được (mỗi lớp nằm về 1 phía)

- Vẫn còn sót lại 1 số phần tử nằm sai phía so với đường thẳng hỗ trợ (các phần tử màu vàng)
- Các ràng buộc không thể thoả mãn

Ràng buộc: $y^{(i)}(w_1 x^{(i)}_1 + w_2 x^{(i)}_2 - b) \geq +1$

Về trái của ràng buộc < 1 .



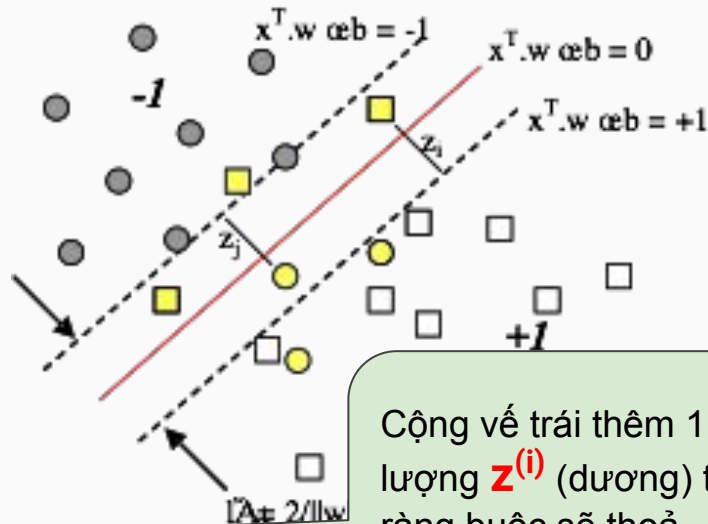
Bài toán SVM

Trường hợp không thể tách được (mỗi lớp nằm về 1 phía)

- Vẫn còn sót lại 1 số phần tử nằm sai phía so với đường thẳng hỗ trợ (các phần tử màu vàng)
- Các ràng buộc không thể thoả mãn

$$y^{(i)}(w_1 x^{(i)}_1 + w_2 x^{(i)}_2 - b) + \mathbf{z}^{(i)} \geq +1$$

Ràng buộc luôn thoả mãn !



Cộng về trái thêm 1 lượng $\mathbf{z}^{(i)}$ (dương) thì ràng buộc sẽ thoả mãn.

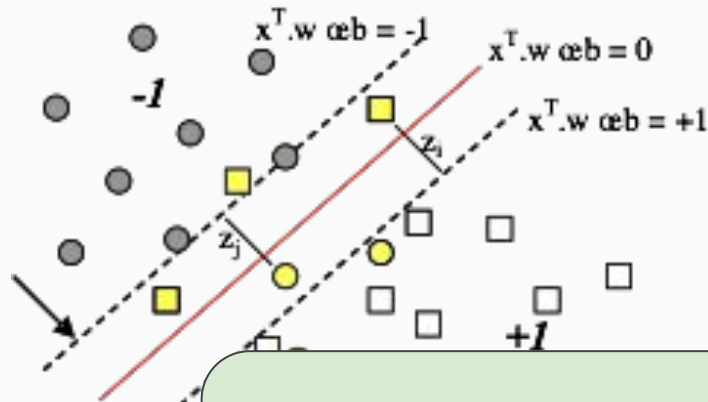
Bài toán SVM

Trường hợp không thể tách được (mỗi lớp nằm về 1 phía)

- Vẫn còn sót lại 1 số phần tử nằm sai phía so với đường thẳng hỗ trợ (các phần tử màu vàng)
- Các ràng buộc không thể thoả mãn

$$y^{(i)}(w_1 x^{(i)}_1 + w_2 x^{(i)}_2 - b) + z^{(i)} \geq +1$$

Ràng buộc luôn thoả mãn !



Tuy nhiên, ta cũng không muốn có nhiều phần tử lỗi hay là **lượng cộng thêm vào này càng nhỏ càng tốt !**

Bài toán SVM

Trường hợp không thể tách được (mỗi lớp nằm về 1 phía):

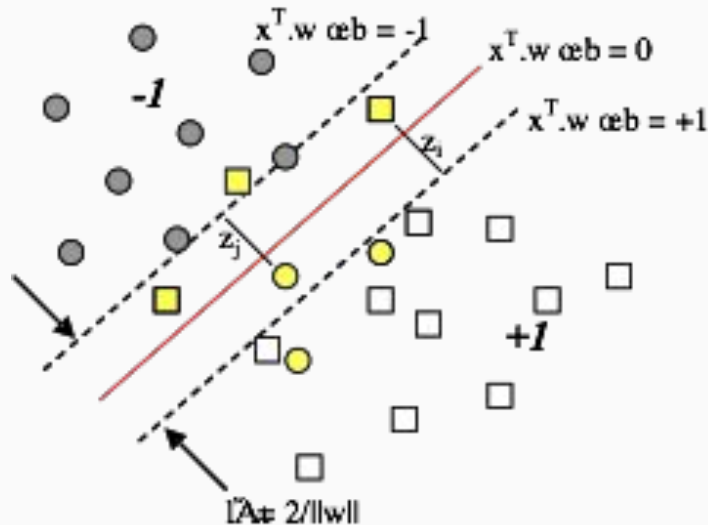
- Hàm mục tiêu:

$$\frac{1}{2}|w|^2 + c \sum_i z^{(i)}$$

- Các ràng buộc:

$$y^{(i)}(w_1 x_1^{(i)} + w_2 x_2^{(i)} - b) + \mathbf{z}^{(i)} \geq +1$$

$$\mathbf{z}^{(i)} \geq 0 \quad (i = 1, 2, \dots, m)$$



Bài toán SVM

Trường hợp không thể tách được (mỗi lớp nằm về 1 phía):

- Hàm mục tiêu:

$$\frac{1}{2}|w|^2 + c \sum_i z^{(i)}$$

- Các ràng buộc:

$$y^{(i)} \left(w_1 x_1^{(i)} + w_2 x_2^{(i)} - b \right) + z^{(i)} \geq +1$$

$$z^{(i)} \geq 0 \quad (i = 1, 2, \dots, m)$$

Hằng số **c** là một siêu tham số (hyperparameter) của mô hình. Ta phải điều chỉnh siêu tham số này khi huấn luyện.

Trong thư viện libSVM, option -c dùng để mô tả tham số này.

c là hằng số dung hoà (trade-off) giữa độ rộng lề và tổng khoảng cách lỗi.

- c càng lớn \Rightarrow LỖI NHỎ
- c càng nhỏ \Rightarrow LỀ LỚN.

Đưa về bài toán quy hoạch toàn phương

Dạng chuẩn:

$$\begin{array}{ll}\text{minimize} & (1/2)x^T Px + q^T x \\ \text{subject to} & Gx \leq h \\ & Ax = b\end{array}$$

Dạng quadprog:

$$\begin{array}{l}\frac{1}{2}x^T Gx - a^T x \\ \\ C^T x \geq b\end{array}$$

Ta sẽ đưa thẳng về dạng **quadprog** luôn cho tiện.

Đưa về bài toán quy hoạch toàn phương

Dạng quadprog:

$$\frac{1}{2}x^T Gx - a^T x$$

$$C^T x \geq b$$

Hàm mục tiêu của SVM

$$\frac{1}{2}|w|^2 + c \sum_i z^{(i)}$$

Viết lại hàm mục tiêu:

$$\frac{1}{2} (w^T w + 0.b.b + 0.z^T z) + c \sum_i z^{(i)}$$

Hãy tìm cách biểu diễn phần trong ngoặc dưới dạng $x^T Gx$

Đưa về bài toán quy hoạch toàn phương

$$\frac{1}{2} (w^T w + 0.b.b + 0.z^T z) + c \sum_i z^{(i)}$$

Biểu diễn phần trong ngoặc
dưới dạng:

$$x^T G x$$

Đặt $x =$

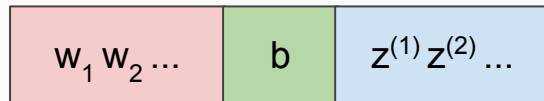
w_1 w_2 ...
b
$z^{(1)}$ $z^{(2)}$...

Đưa về bài toán quy hoạch toàn phương

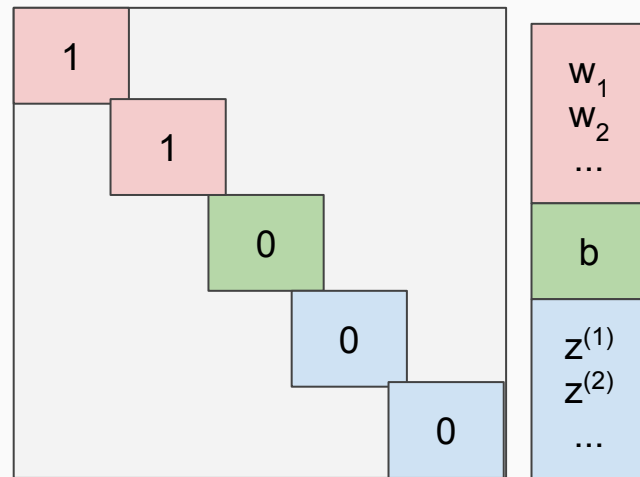
$$\frac{1}{2} (w^T w + 0.b.b + 0.z^T z) + c \sum_i z^{(i)}$$

Biểu diễn phần trong ngoặc
dưới dạng:

$$x^T G x$$



x^T



G

x

Đưa về bài toán quy hoạch toàn phương

$$\frac{1}{2} (w^T w + 0.b.b + 0.z^T z) + c \sum_i z^{(i)}$$

Biểu diễn phần tổng các z_i dưới dạng:

$a^T x$

0 0 ... 0	0	-c -c ... -c
-----------	---	--------------

a^T

-c

0 0 ... 0	0	1 1 ... 1
-----------	---	-----------

Các phần tương ứng với w và b đều = 0, chỉ có phần tương ứng với z là = 1.

Vì thế $a^T x =$ tổng các z_i .

w_1 w_2 ...
b
$z^{(1)}$ $z^{(2)}$...

x

Đưa về bài toán quy hoạch toàn phương

Các ràng buộc:

$$y^{(i)}(w_1 x_1^{(i)} + w_2 x_2^{(i)} - b) + z^{(i)} \geq +1$$

$$z^{(i)} \geq 0 \quad (i = 1, 2, \dots, m)$$

Cần phải đưa về dạng:

$$C^T x \geq b$$

$$y^{(i)}x_1^{(i)}w_1 + y^{(i)}x_2^{(i)}w_2 - y^{(i)}b + 1.z^{(i)} \geq +1$$

$C^T =$

$y^{(1)}x^{(1)}_1$	\dots	$y^{(1)}x^{(1)}_n$	$-y^{(1)}$	1	0	\dots	0
\dots			$-y^{(2)}$	0	1	\dots	0
$y^{(m)}x^{(m)}_1$	\dots		\dots	\dots	\dots		
$y^{(m)}x^{(m)}_n$			$-y^{(m)}$	0	0	\dots	1
0	0		0	1	0	0	
0	0		0	0	1	0	
0	0		0	0	0	1	

x

w_1	1
w_2	1
\dots	\dots
b	1
$z^{(1)}$	0
$z^{(2)}$	0
\dots	0

b

Ví dụ 1

Giả sử, ta có tập huấn luyện:

x1	x2	y
2	2	+1
3	1	+1
1	1	-1

Hãy viết ma trận C^T !

$$y^{(i)}x_1^{(i)}w_1 + y^{(i)}x_2^{(i)}w_2 - y^{(i)}b + 1.z^{(i)} \geq +1$$

$C^T =$

$y^{(1)}x^{(1)}_1 \dots y^{(1)}x^{(1)}_n$	$-y^{(1)}$	1	0	...	0
...	$-y^{(2)}$	0	1	...	0
$y^{(m)}x^{(m)}_1 \dots$
$y^{(m)}x^{(m)}_n$	$-y^{(m)}$	0	0	...	1
0	0	0	1	0	0
0	0	0	0	1	0
0	0	0	0	0	1

x

w_1	1
w_2	1
...	...
b	1
$z^{(1)}$	0
$z^{(2)}$	0
...	0

b

Dừng lại chút và làm bài !

Đừng vội lật slide !!!

Ví dụ 1

$$y^{(i)}x_1^{(i)}w_1 + y^{(i)}x_2^{(i)}w_2 - y^{(i)}b + 1.z^{(i)} \geq +1$$

Giả sử, ta có tập huấn luyện:

x1	x2	y
2	2	+1
3	1	+1
1	1	-1

Hãy viết ma trận C^T !

$C^T =$

2	2	-1	1	0	0
3	1	-1	0	1	0
-1	-1	1	0	0	1
0	0	0	1	0	0
0	0	0	0	1	0
0	0	0	0	0	1

w_1	1
w_2	1
...	1
b	
$z^{(1)}$	0
$z^{(2)}$	0
$z^{(3)}$	0

x

Dừng lại chút và
cài đặt thử đi !

Đừng vội lật slide !!!

Cài đặt ví dụ 1

```
6 G = np.matrix([[1.0, 0.0, 0.0, 0.0, 0.0, 0.0],
7                [0.0, 1.0, 0.0, 0.0, 0.0, 0.0],
8                [0.0, 0.0, 0.0001, 0.0, 0.0, 0.0],
9                [0.0, 0.0, 0.0, 0.0001, 0.0, 0.0],
10               [0.0, 0.0, 0.0, 0.0, 0.0001, 0.0],
11               [0.0, 0.0, 0.0, 0.0, 0.0, 0.0001]
12               ])
13
14 c = 100
15 a = np.array([0.0, 0.0, 0.0, -c, -c, -c])
```

Cài đặt ví dụ 1

```
17 C = np.matrix([[2.0, 2.0, -1.0, 1.0, 0.0, 0.0],
18                [3.0, 1.0, -1.0, 0.0, 1.0, 0.0],
19                [-1.0, -1.0, 1.0, 0.0, 0.0, 1.0],
20
21                [0.0, 0.0, 0.0, 1.0, 0.0, 0.0],
22                [0.0, 0.0, 0.0, 0.0, 1.0, 0.0],
23                [0.0, 0.0, 0.0, 0.0, 0.0, 1.0],
24                ]).T
25
26 b = np.array([1.0, 1.0, 1.0, 0.0, 0.0, 0.0])
```


Cài đặt ví dụ 1

```
28 sol = qp.solve_qp(G, a, C, b)
29
30 wb = sol[0]
31
32 print(wb)
```

```
[1. 1. 3. 0. 0. 0.]
```

Kết quả:

$w_1 = 1, w_2 = 1$

$b = 3$

$z = [0 \ 0 \ 0]$

Ví dụ 2

Giả sử, ta có tập huấn luyện:

x1	x2	y
2	2	+1
3	1	+1
1	1	-1
2.5	1.4	-1

Hãy viết ma trận C^T !

$$y^{(i)}x_1^{(i)}w_1 + y^{(i)}x_2^{(i)}w_2 - y^{(i)}b + 1.z^{(i)} \geq +1$$

$C^T =$

$y^{(1)}x^{(1)}_1 \dots y^{(1)}x^{(1)}_n$	$-y^{(1)}$	1 0 ... 0
...	$-y^{(2)}$	0 1 ... 0
$y^{(m)}x^{(m)}_1 \dots$
$y^{(m)}x^{(m)}_n$	$-y^{(m)}$	0 0 ... 1
0 0 .. 0	0	1 0 ... 0
0 0 .. 0	0	0 1 ... 0
...
0 0 .. 0	0	0 0 ... 1

x

w_1	1
w_2	1
...	...
b	1
$z^{(1)}$	0
$z^{(2)}$	0
...	...
	0

b

Dừng lại chút và
làm bài !

Đừng vội lật slide !!!

Ví dụ 2

$$y^{(i)}x_1^{(i)}w_1 + y^{(i)}x_2^{(i)}w_2 - y^{(i)}b + 1.z^{(i)} \geq +1$$

Giả sử, ta có tập huấn luyện:

x1	x2	y
2	2	+1
3	1	+1
1	1	-1
2.5	1.4	-1

Hãy viết ma trận C^T !

$C^T =$

2	2	-1	1	0	0	0
3	1	-1	0	1	0	0
-1	-1	1	0	0	1	0
-2.5	-1.4	1	0	0	0	1
0	0	0	1	0	0	0
0	0	0	0	1	0	0
0	0	0	0	0	1	0
0	0	0	0	0	0	1

w_1	1
w_2	1
...	1
b	
$z^{(1)}$	0
$z^{(2)}$	0
$z^{(3)}$	0

x

Dừng lại chút và
cài đặt thử đi !

Đừng vội lật slide !!!

Cài đặt ví dụ 2

```
6 G = np.matrix([[1.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0],
7                [0.0, 1.0, 0.0, 0.0, 0.0, 0.0, 0.0],
8                [0.0, 0.0, 0.00000001, 0.0, 0.0, 0.0, 0.0],
9                [0.0, 0.0, 0.0, 0.00000001, 0.0, 0.0, 0.0],
10               [0.0, 0.0, 0.0, 0.0, 0.00000001, 0.0, 0.0],
11               [0.0, 0.0, 0.0, 0.0, 0.0, 0.00000001, 0.0],
12               [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.00000001]
13              ])
14 c = 100
15 a = np.array([0.0, 0.0, 0.0, -c, -c, -c, -c])
```

Đã cập nhật mới.

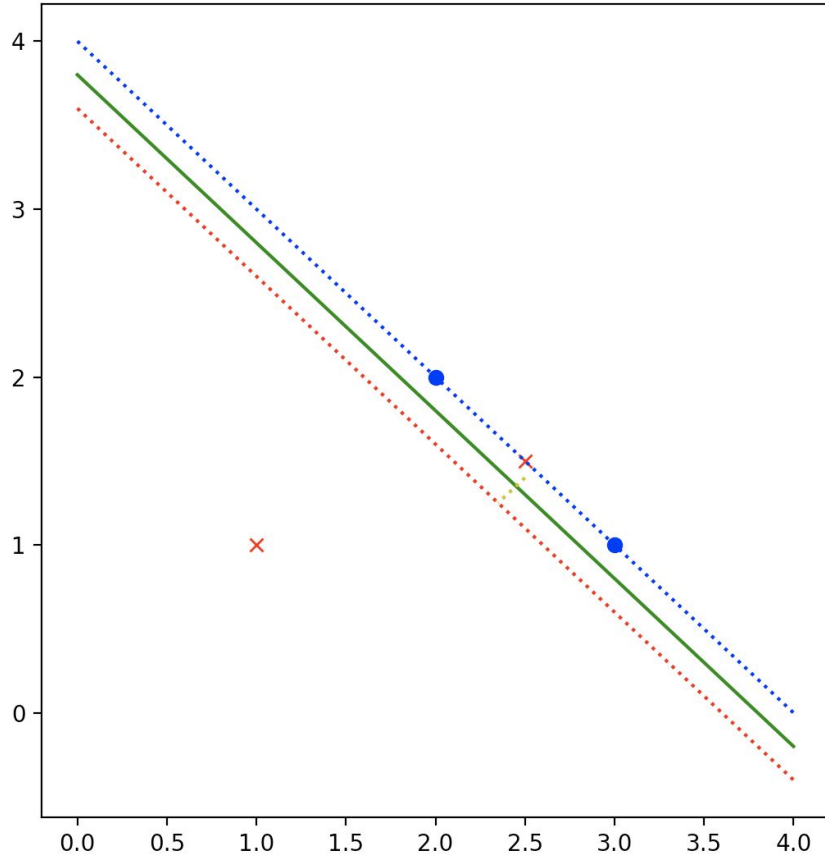
Cài đặt ví dụ 2

```
17 C = np.matrix([[2.0, 2.0, -1.0, 1.0, 0.0, 0.0, 0.0],
18                [3.0, 1.0, -1.0, 0.0, 1.0, 0.0, 0.0],
19                [-1.0, -1.0, 1.0, 0.0, 0.0, 1.0, 0.0],
20                [-2.5, -1.4, 1.0, 0.0, 0.0, 0.0, 1.0],
21
22                [0.0, 0.0, 0.0, 1.0, 0.0, 0.0, 0.0],
23                [0.0, 0.0, 0.0, 0.0, 1.0, 0.0, 0.0],
24                [0.0, 0.0, 0.0, 0.0, 0.0, 1.0, 0.0],
25                [0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 1.0]
26                ]).T
27
28 b = np.array([1.0, 1.0, 1.0, 1.0, 0.0, 0.0, 0.0, 0.0])
```

Cài đặt ví dụ 2

```
28     b = np.array([1.0, 1.0, 1.0, 1.0, 0.0, 0.0, 0.0, 0.0])
29
30     sol = qp.solve_qp(G, a, C, b)
31
32     wb = sol[0]
33
34     print(wb)
```


Cài đặt ví dụ 2



c = 100

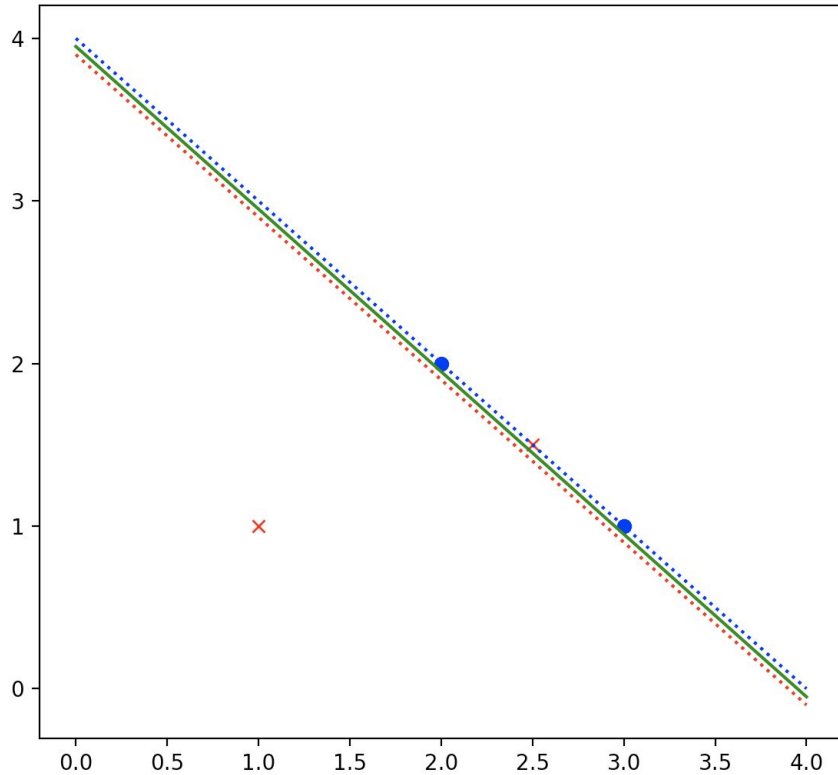
Kết quả:

$w_1 = 4.99621053$, $w_2 = 4.99621053$

$b = 18.98484213$

$z = [0 \ 0 \ 0 \ 1.50037895]$, có nghĩa là $\mathbf{z}^{(4)} = 1.50037895$

Cài đặt ví dụ 2



Tăng $c = 1000$ (Không lỗi)

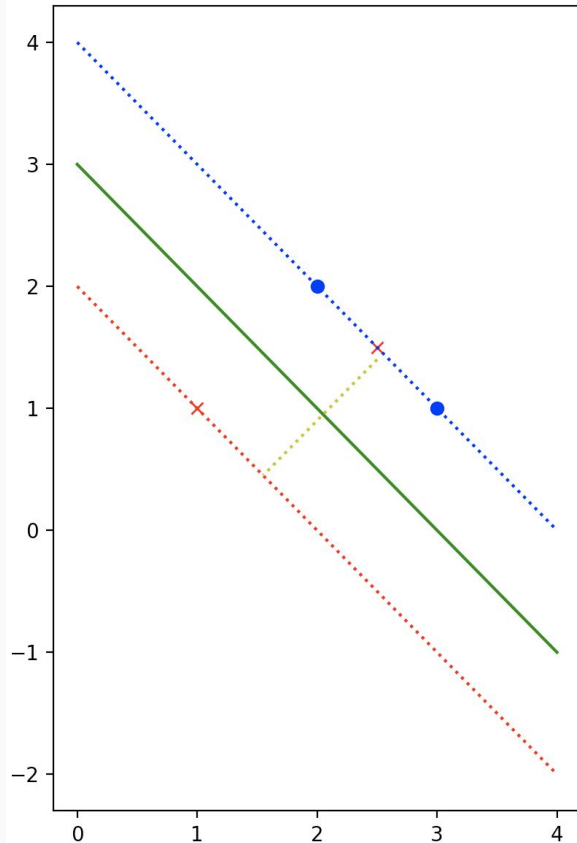
Kết quả:

$w_1 = 20, w_2 = 20$

$b = 79$

$z = [0 \ 0 \ 0 \ 1.4e-14]$, Lỗi quá nhỏ, không đáng kể.

Cài đặt ví dụ 2



Thay đổi $c = 10$ (Tìm lẻ lớn, không quá quan trọng lỗi)

Kết quả:

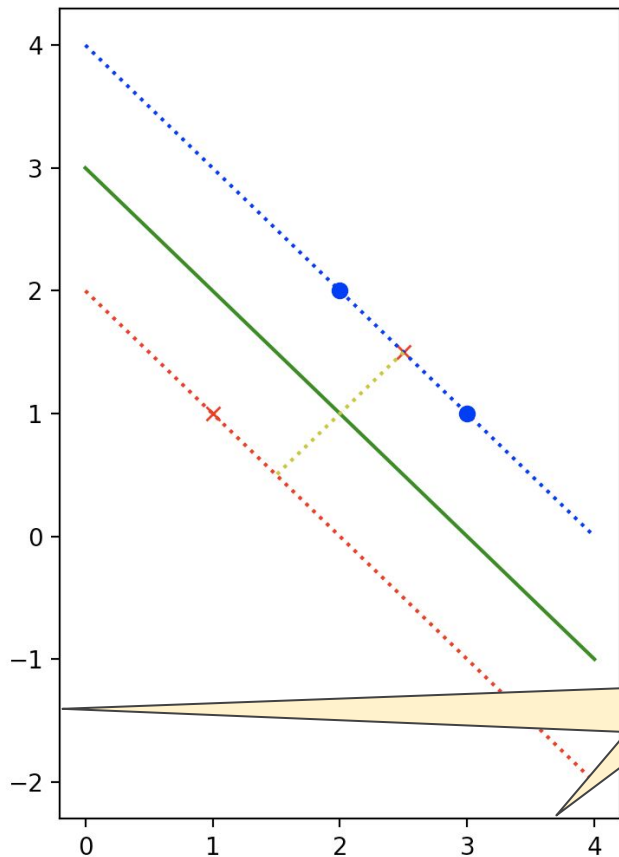
$$w_1 = 1, w_2 = 1$$

$$b = 3$$

$z = [0 \ 0 \ 0 \ 1.9]$, có nghĩa là $\mathbf{z}^{(4)} = 1.9$

Dừng lại chút và
lập trình chạy thử đi !

Ý nghĩa của khoảng cách lỗi $z^{(i)}$



Thay đổi $c = 2$ (Tìm lẻ lớn, không quá quan trọng lỗi)

Kết quả:

$$w_1 = 1, w_2 = 1$$

$$b = 3$$

$z = [0 \ 0 \ 1.9]$, có nghĩa là $z^{(3)} = 1.9$

Đơn vị tính trên các trục: gọi là **đơn vị chính**

Đơn vị tính cho khoảng cách lỗi là đơn vị tính theo pháp véc-tơ w , gọi tắt là **đơn vị w** .

Nếu tính theo **đơn vị w** , thì khoảng cách lẻ là **2**. Còn nếu tính theo **đơn vị chính** thì sẽ là **$2/|w|$** .

Vì thế, nếu lỗi là **$z^{(i)} = 1.9$** thì theo đơn vị chính sẽ là **$1.9/|w|$** .

1 đơn vị $w = 1/|w|$ đv chính

Homework

Bài tập (**hạn cuối: 17h00 chủ nhật ngày 5/4/2020**)

- Xem trên google classroom:

<https://classroom.google.com/u/0/c/NzE2NDAYMjAyOVpa/a/NjU5NzE4ODMzOTBa/details>

Tham khảo

Nộp bài vào đây (nhớ đúng deadline): <https://forms.gle/S4fmK9m8oeAKLGn97>

Vẽ đồ thị với Mathplot: <https://matplotlib.org/tutorials/introductory/pyplot.html>

Thư viện Quadprog: <https://pypi.org/project/quadprog/>