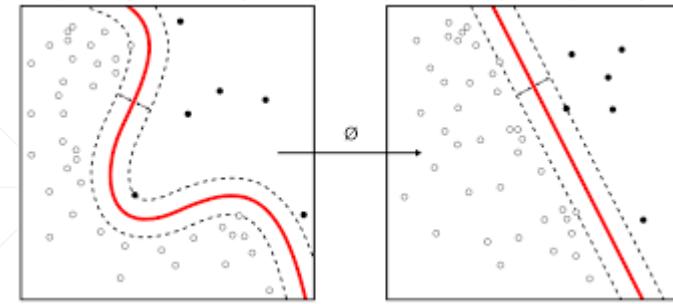


# Support Vector Machine

---

~Abhishek Kumar

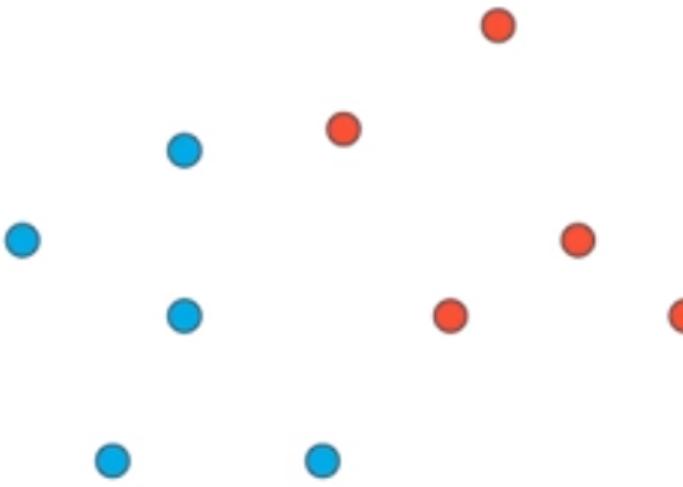


# SVM

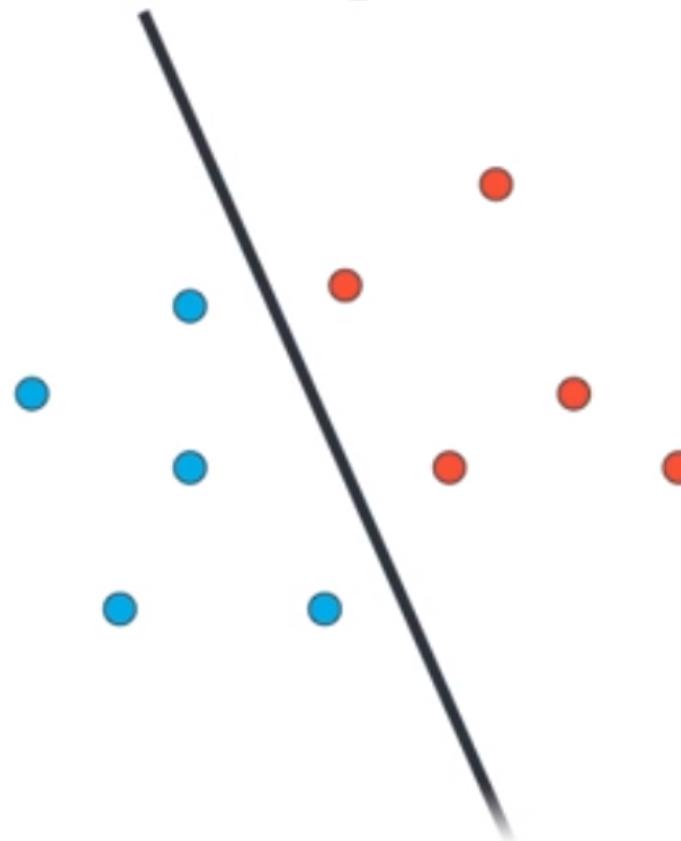
- Supervised
- Classification algorithm



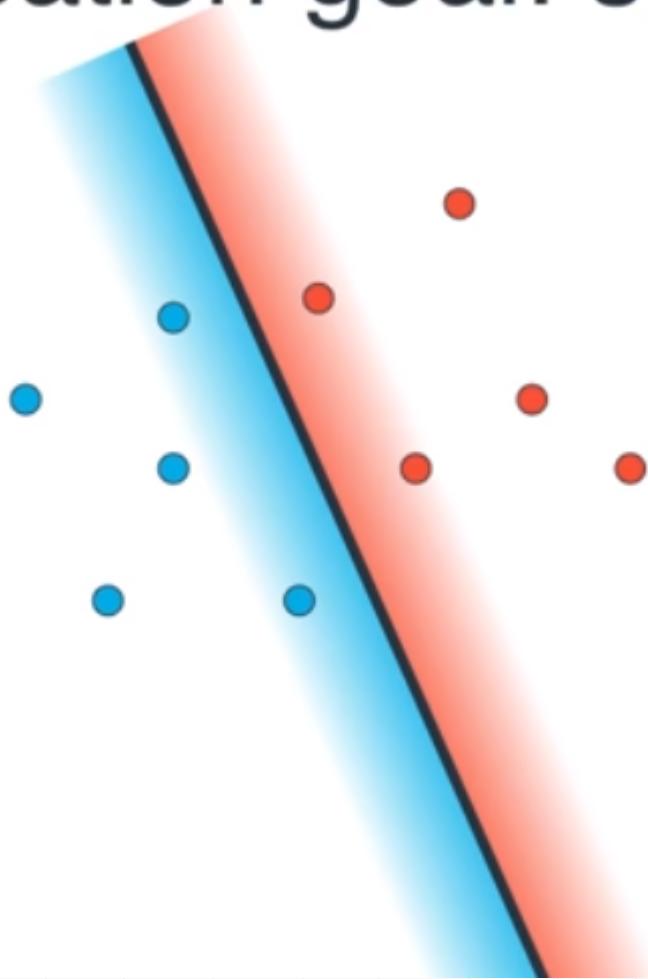
# Classification goal: split data

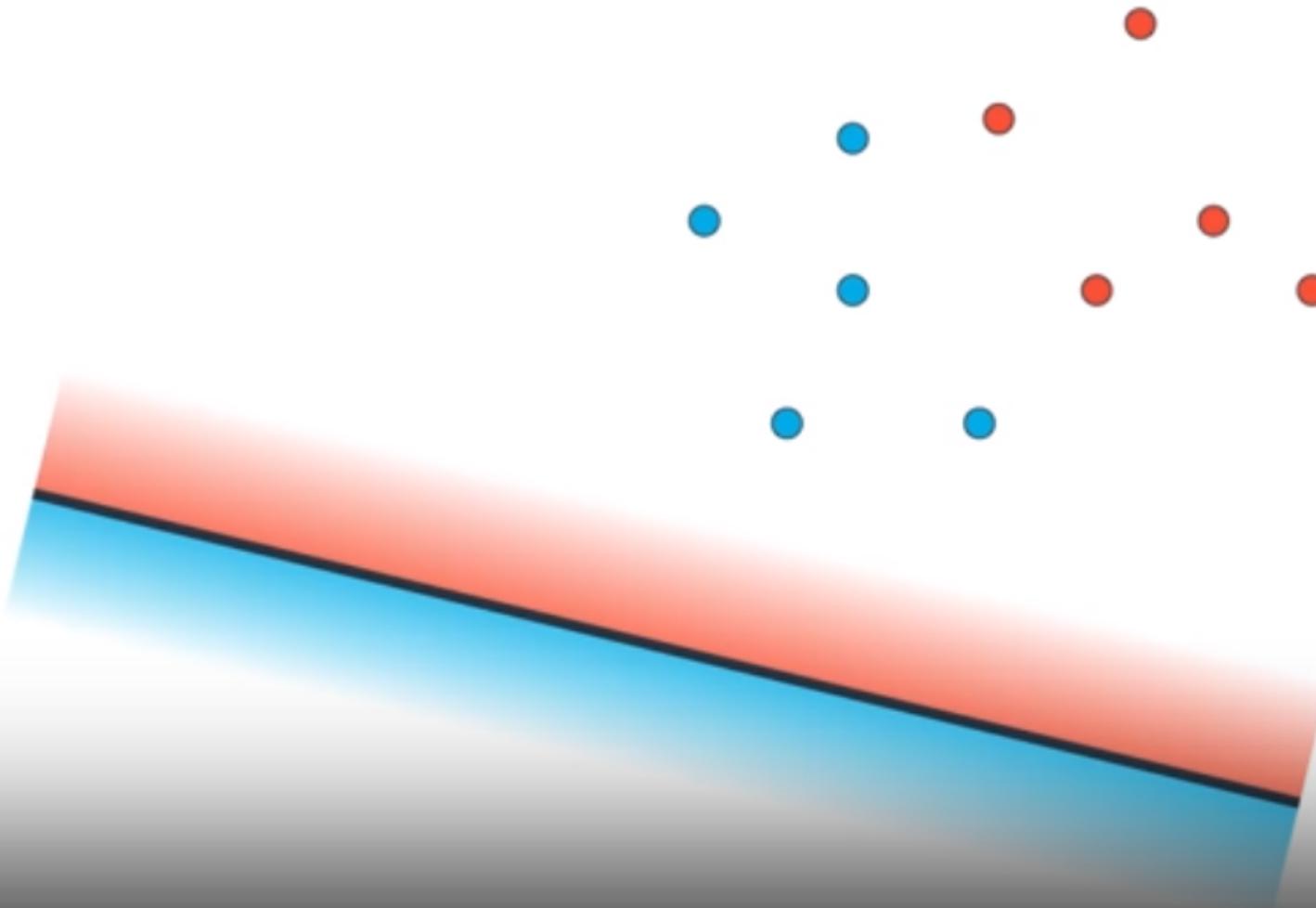


# Classification goal: split data

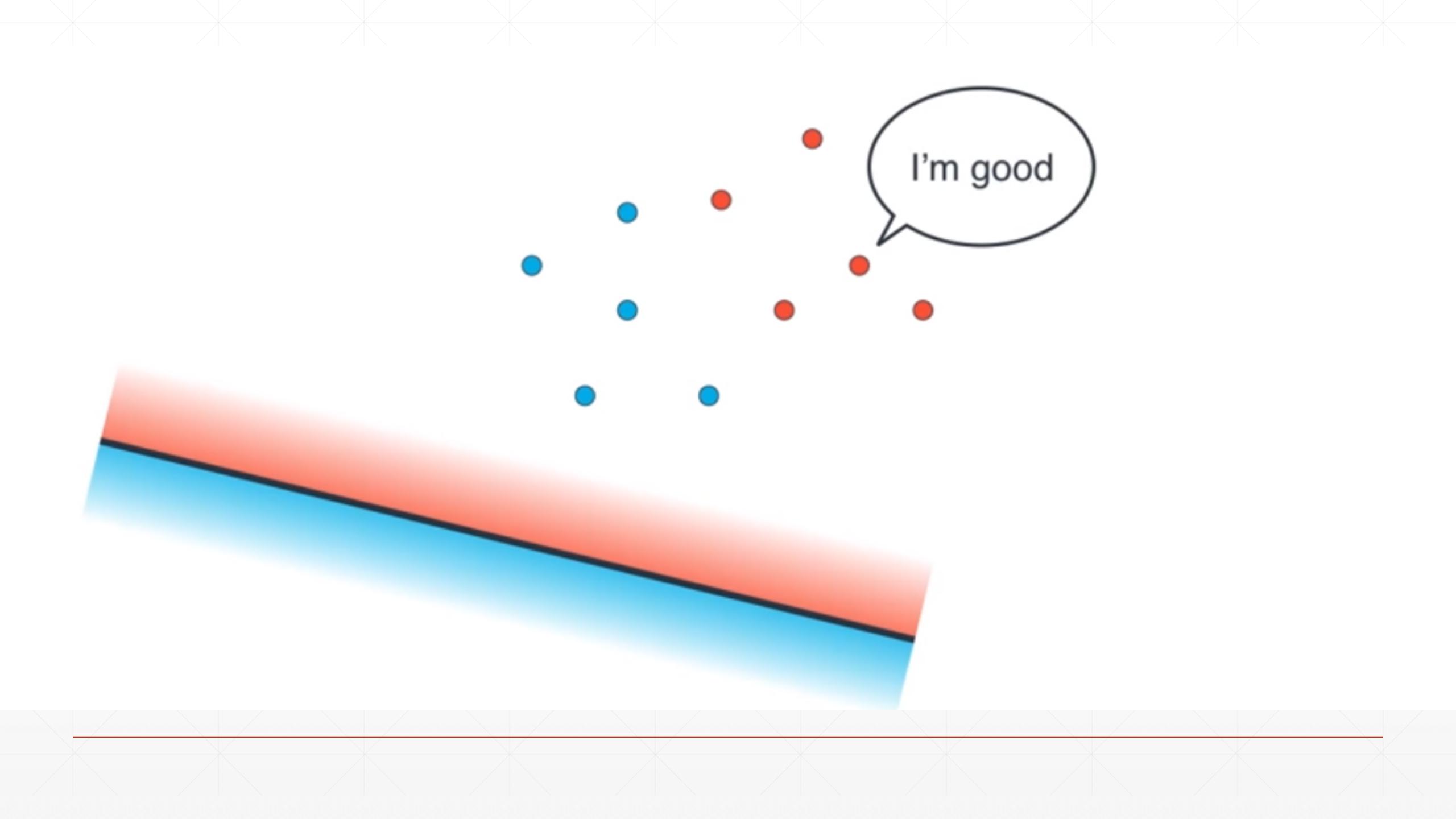


# Classification goal: split data

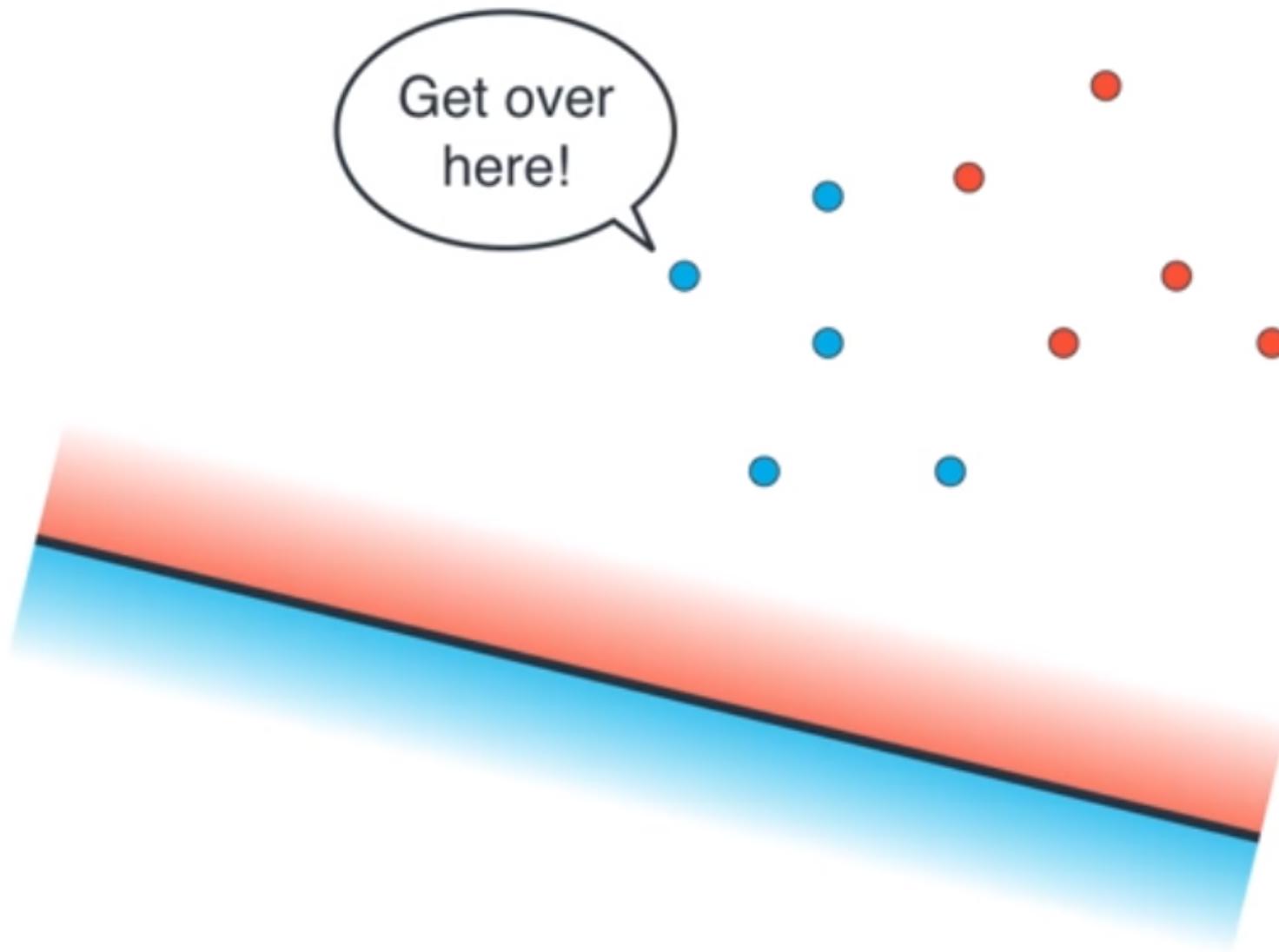


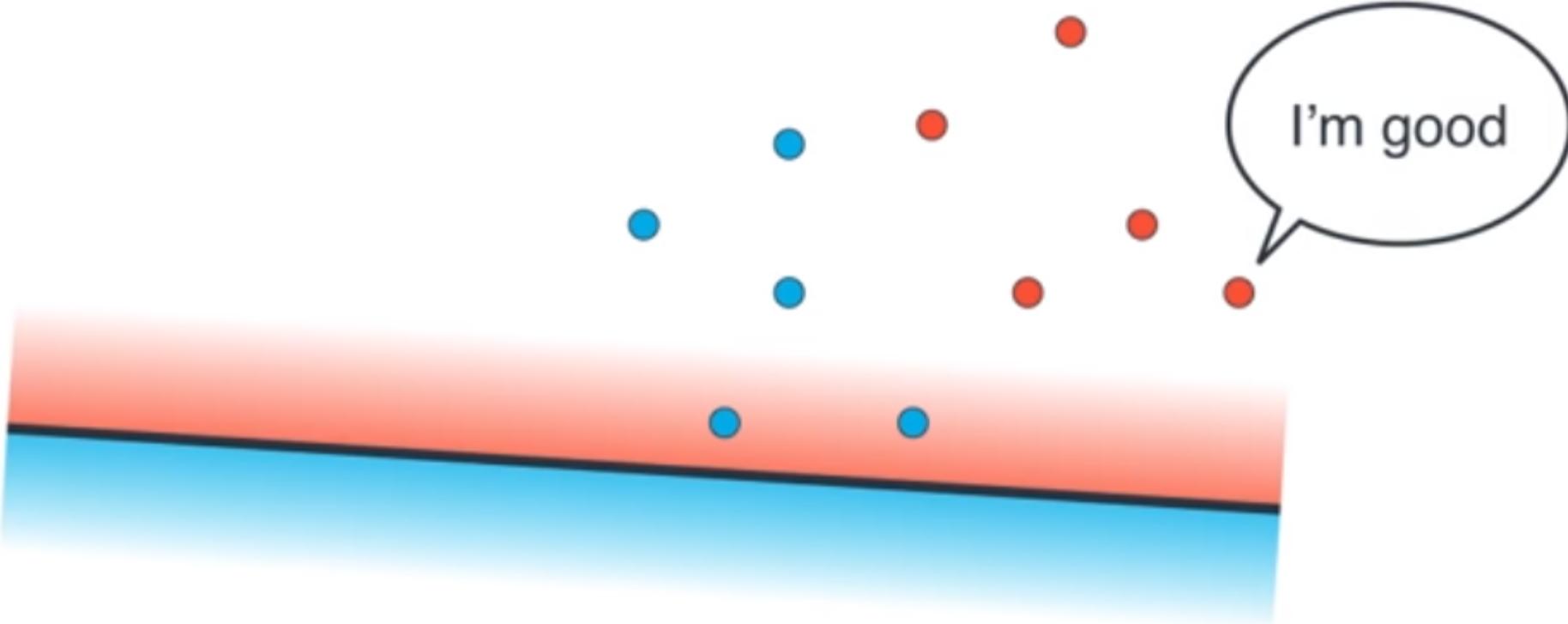


---



I'm good

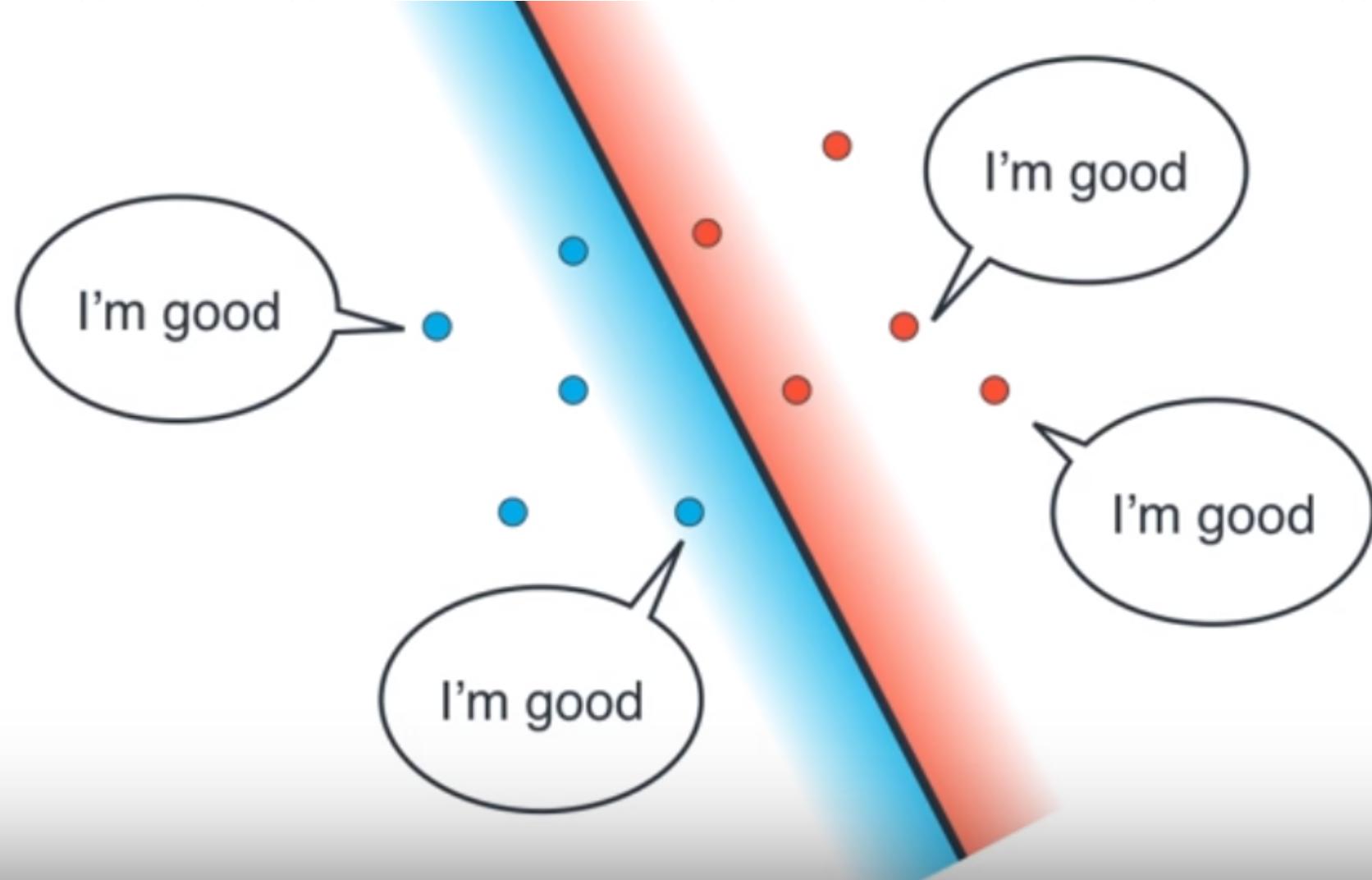




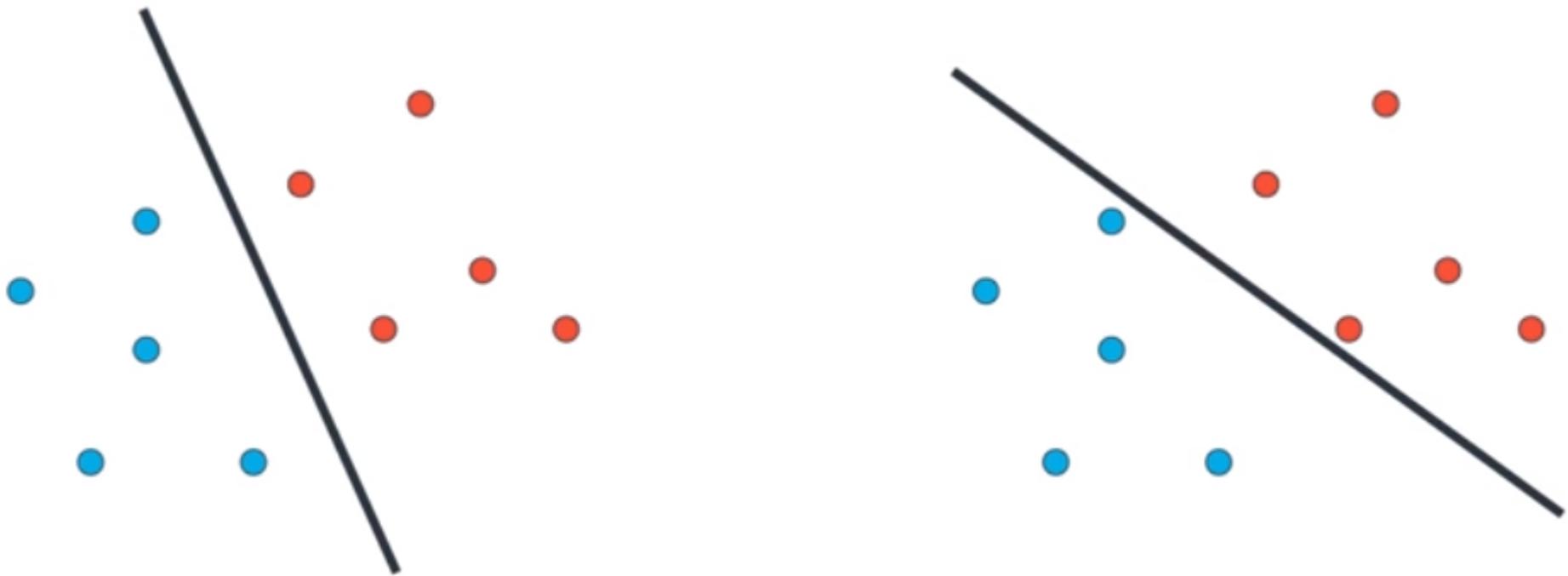
I'm good

A diagram illustrating a social interaction between two groups of people. At the bottom, there is a horizontal bar divided into three segments: a blue segment on the left, an orange segment in the middle, and a black segment on the right. Above this bar, several small circles are scattered across the white space. There are four blue circles clustered near the center-left, and five red circles clustered towards the right side. A speech bubble originates from one of the blue circles, containing the text "Get over here!".

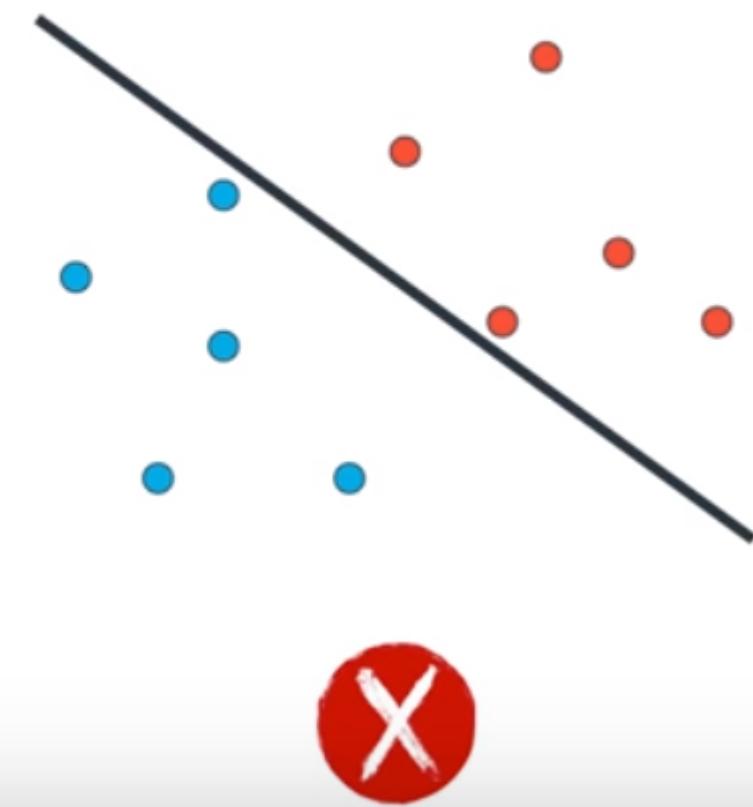
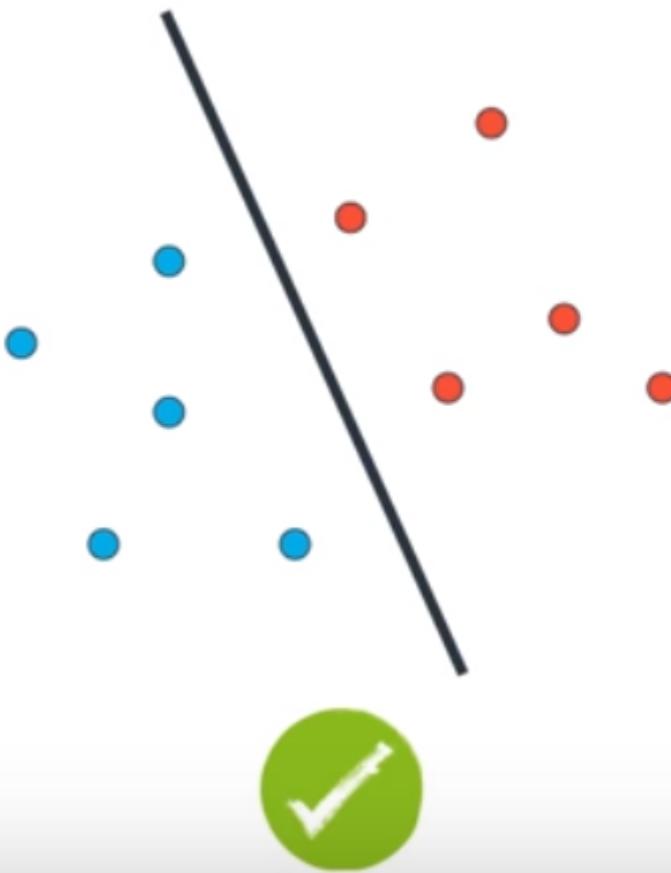
Get over  
here!



# Which line is better?

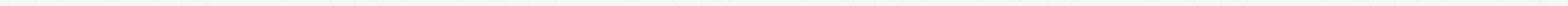


# Which line is better?

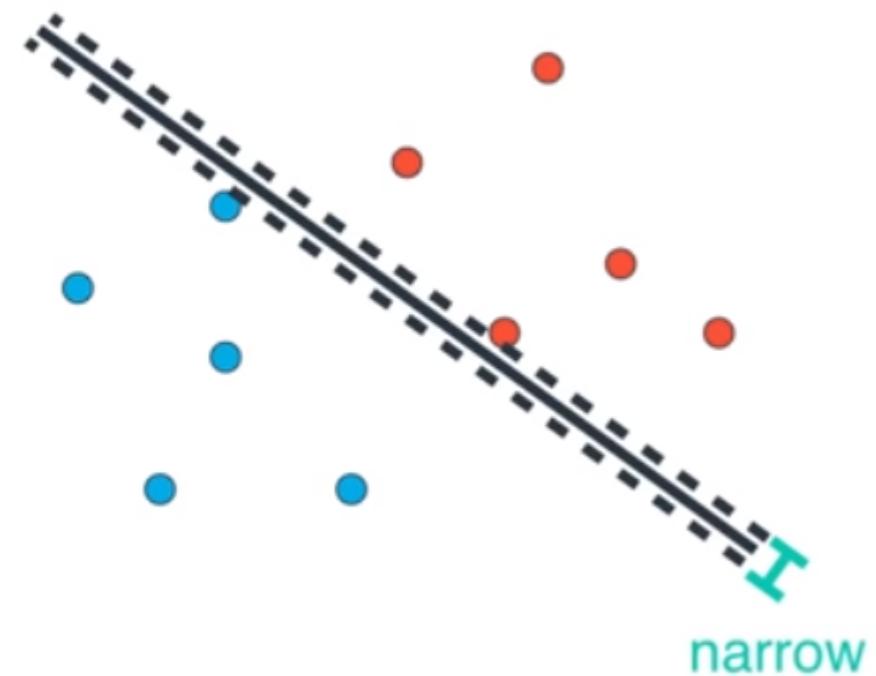
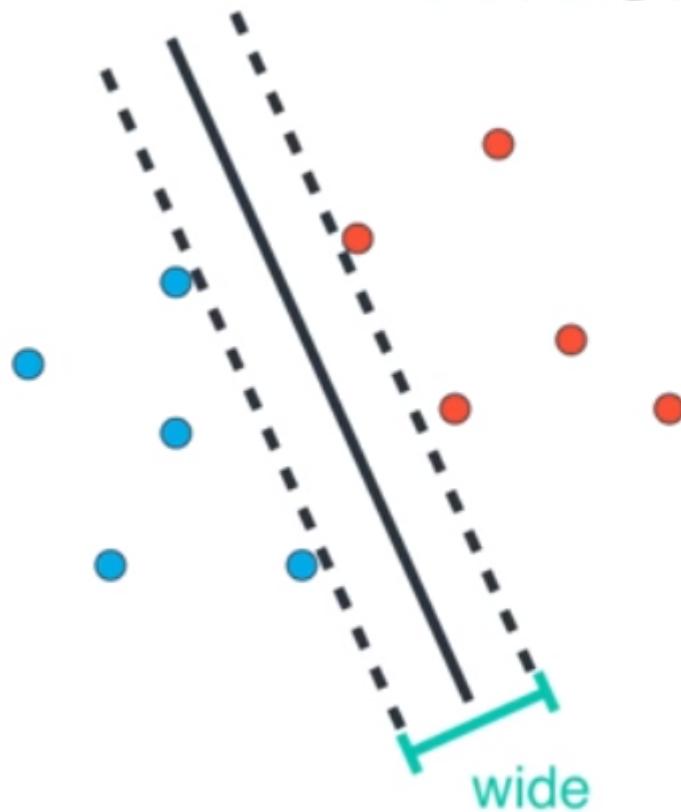


# How do we tell the computer to pick the best line?

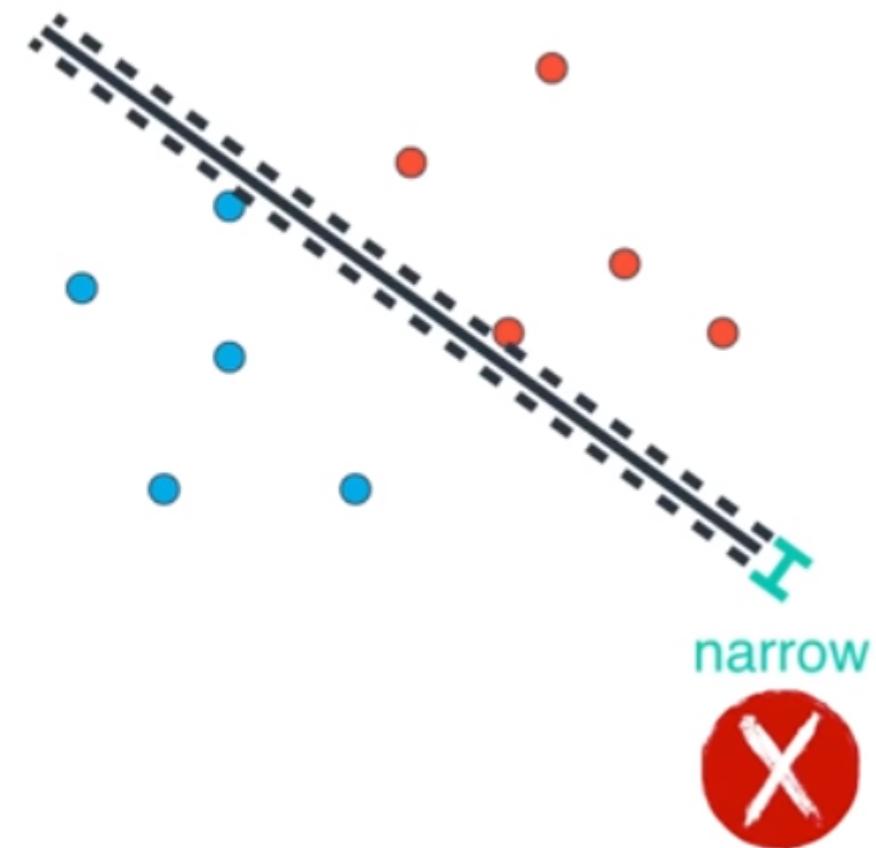
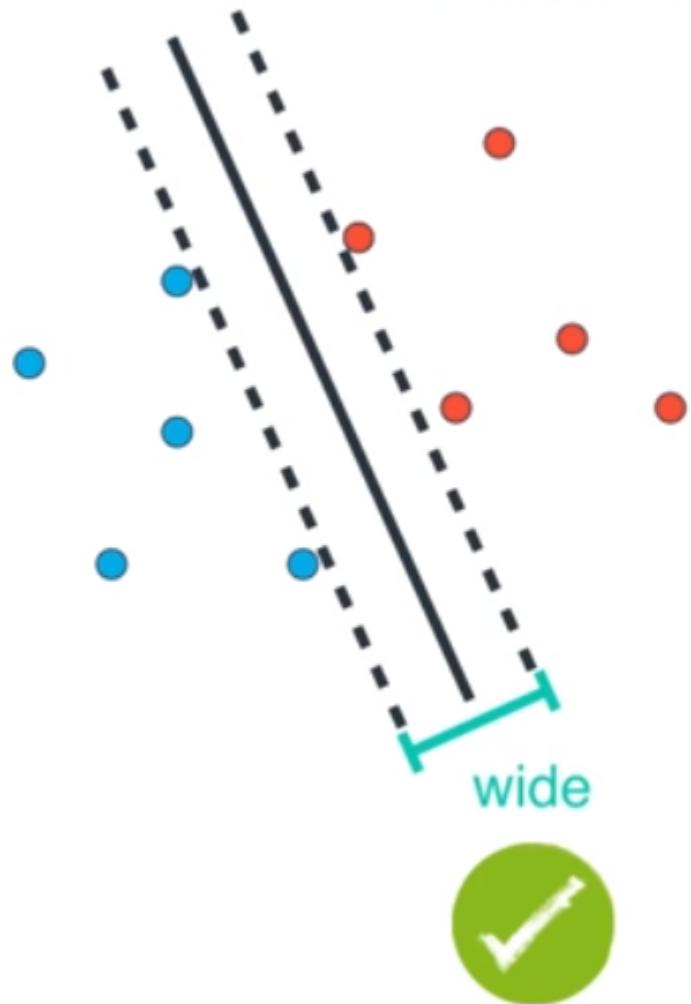
- Can we say that? We should find two lines that space apart as far as possible from each other



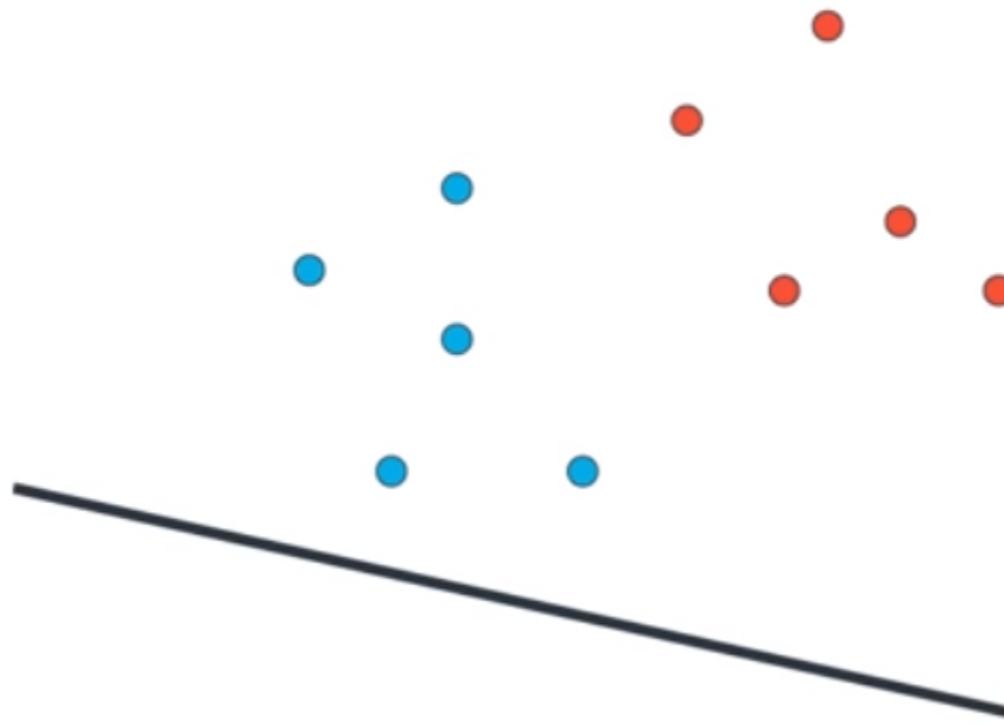
# Which line is better?



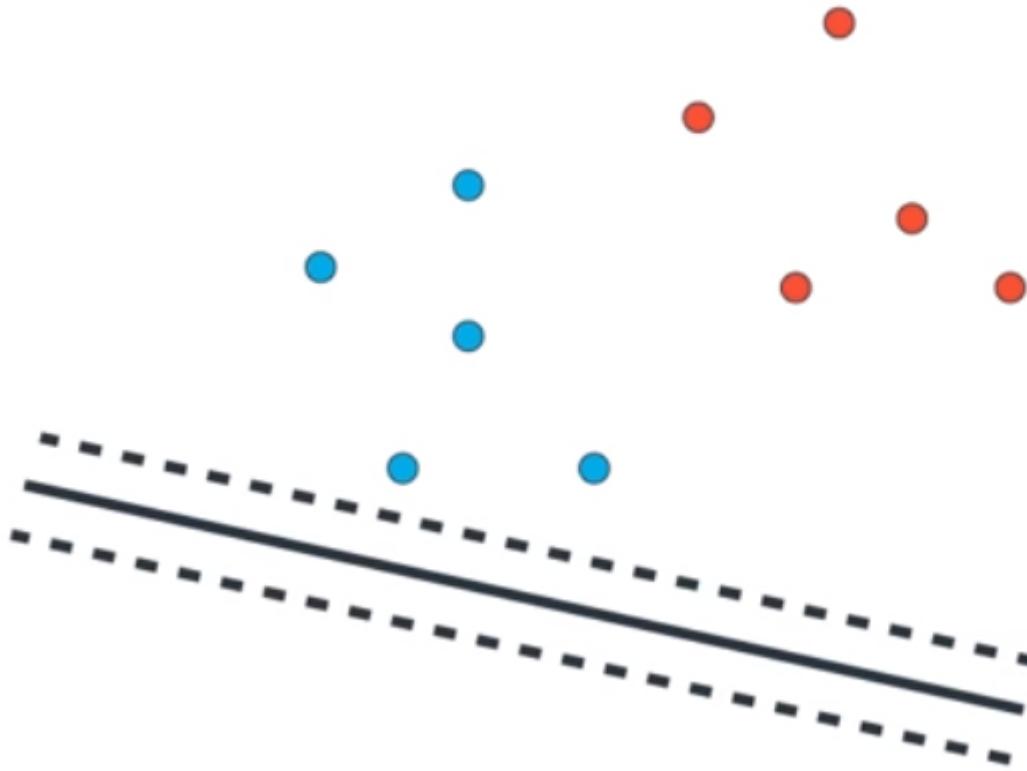
# Which line is better?



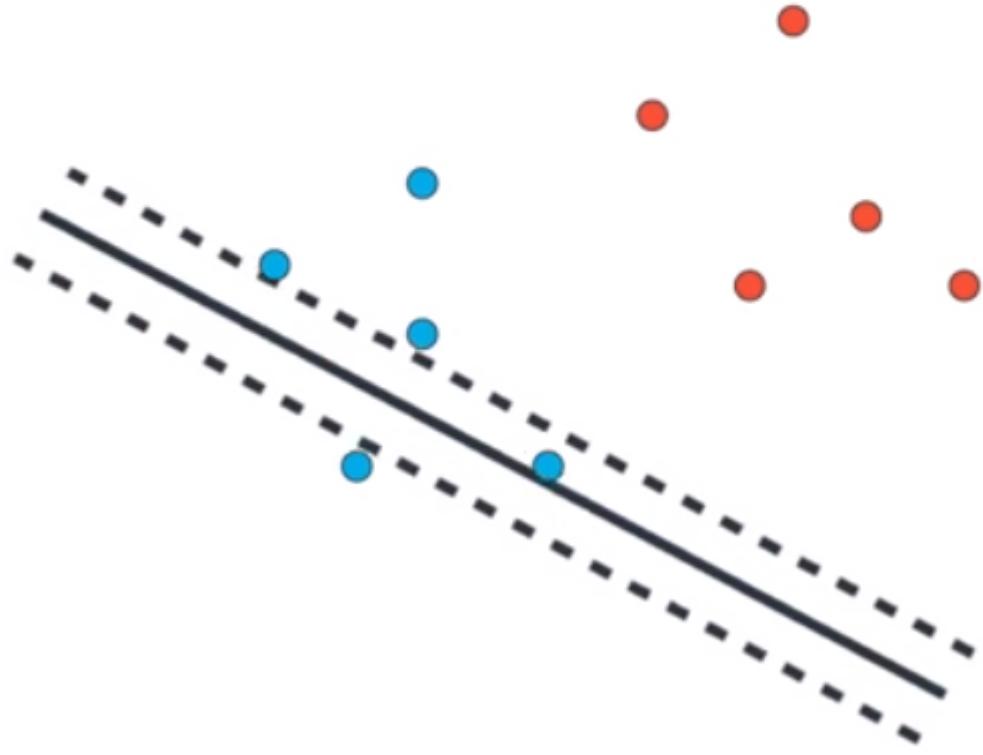
# Split data - separate lines



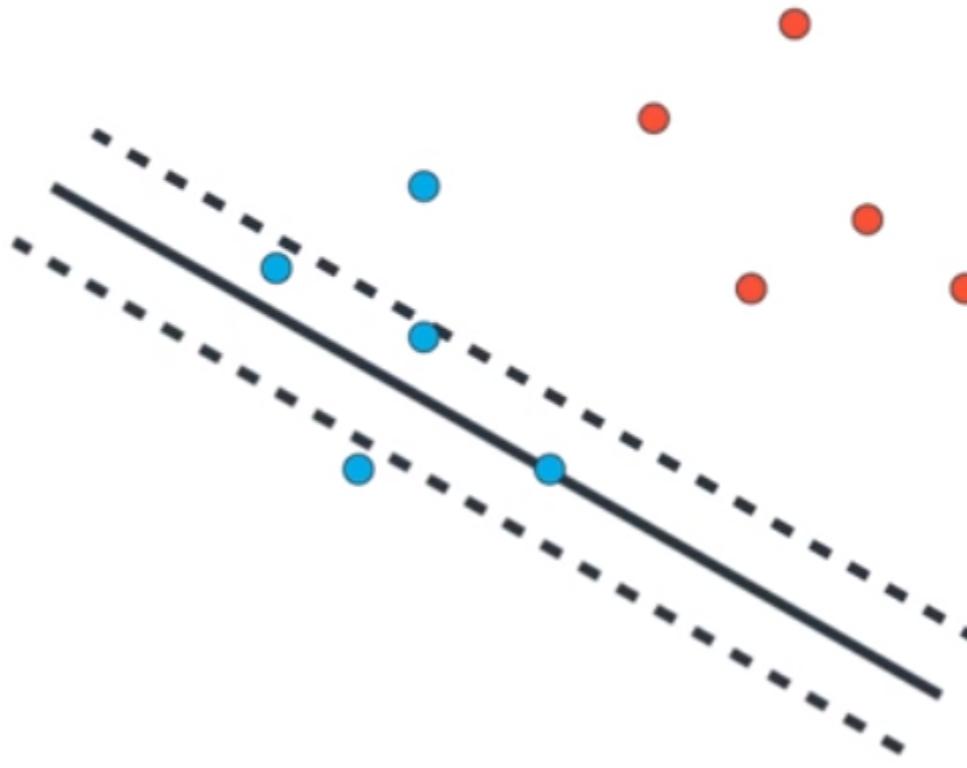
# Split data - separate lines



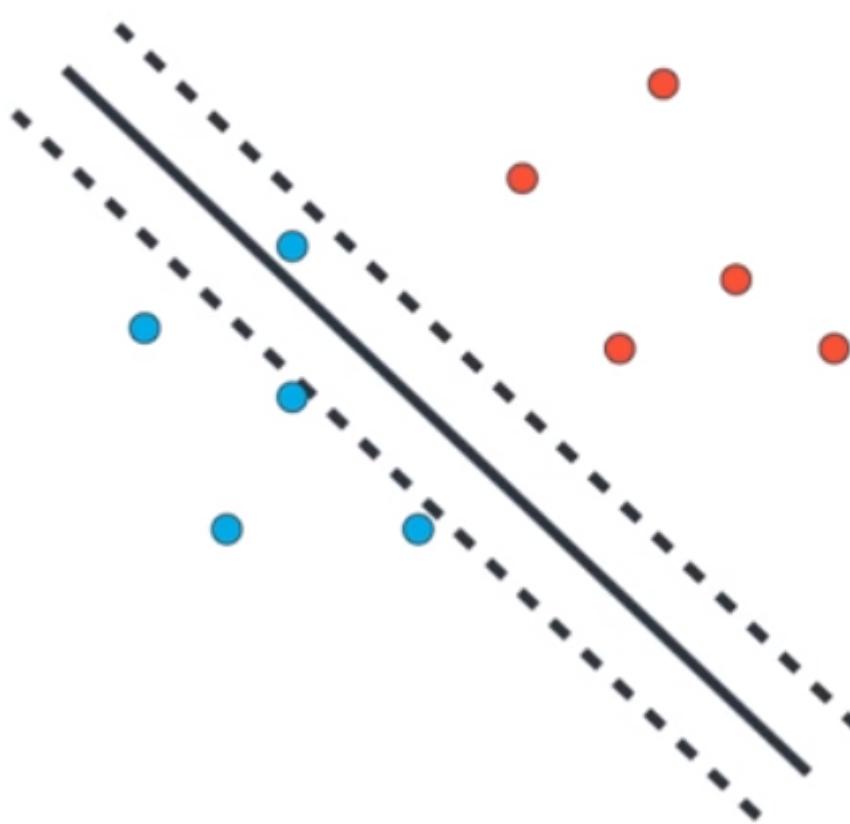
# Split data - separate lines



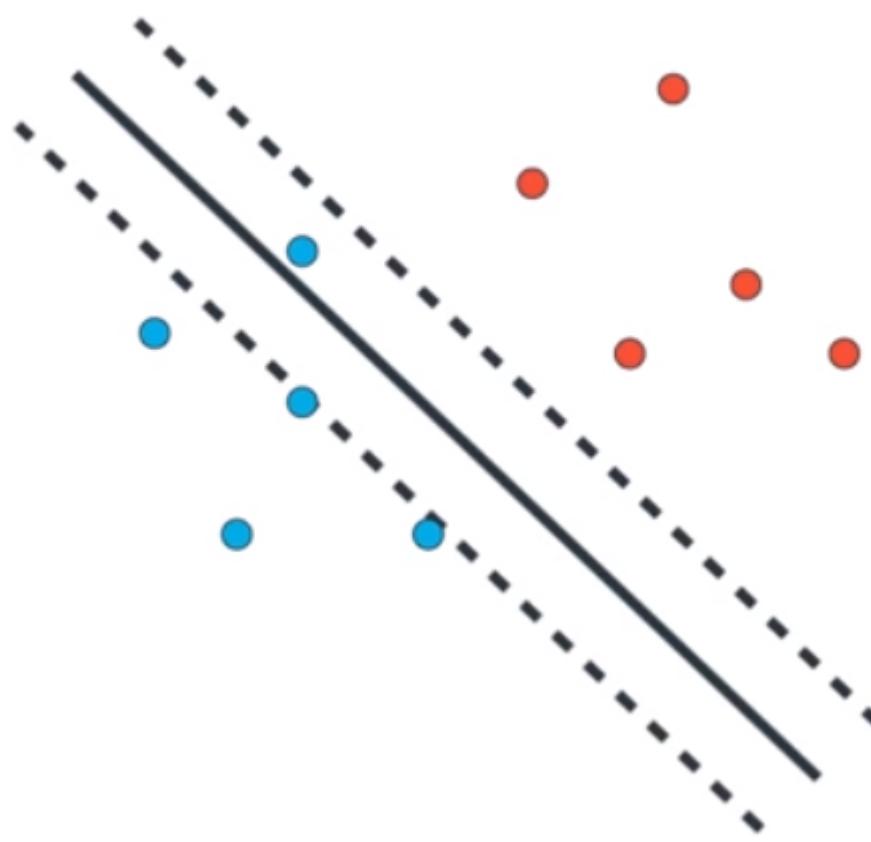
# Split data - separate lines



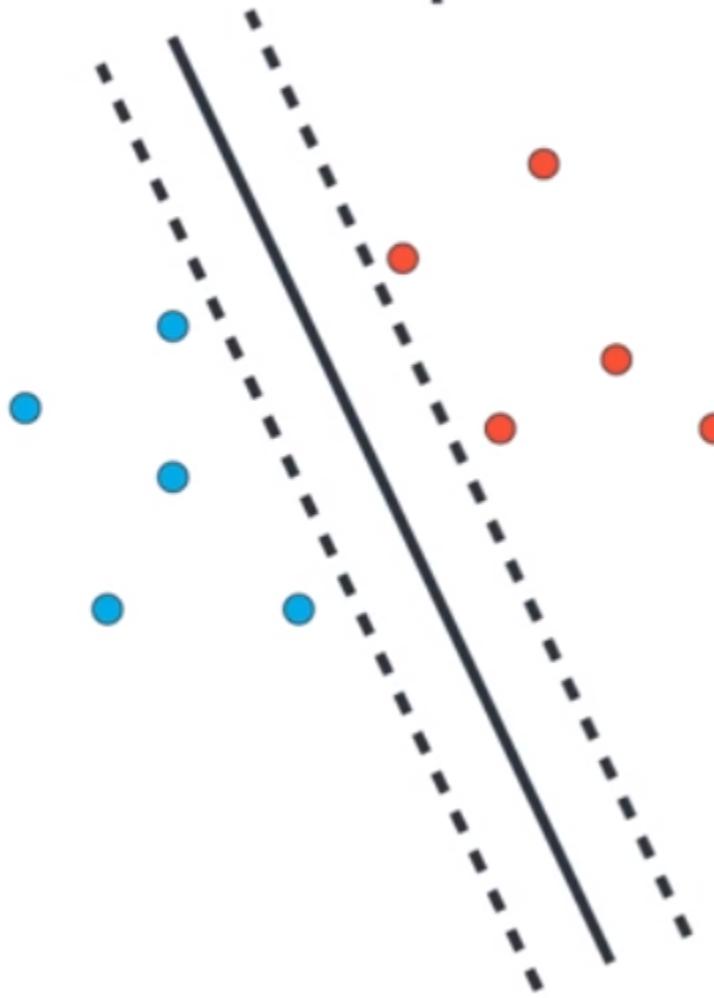
# Split data - separate lines



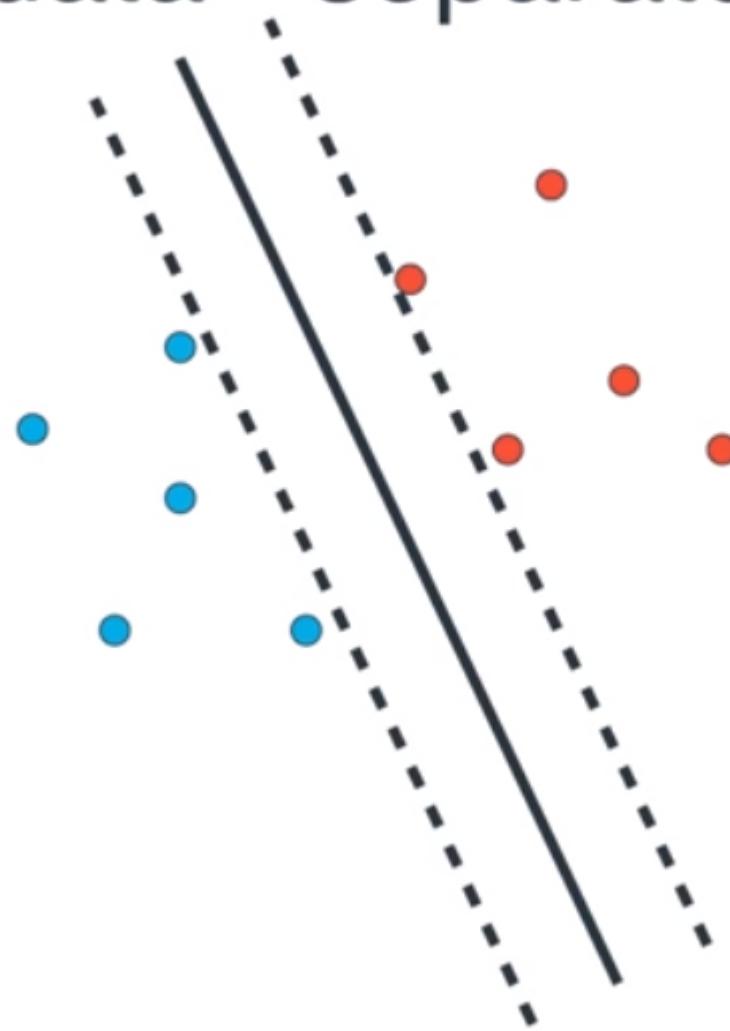
# Split data - separate lines



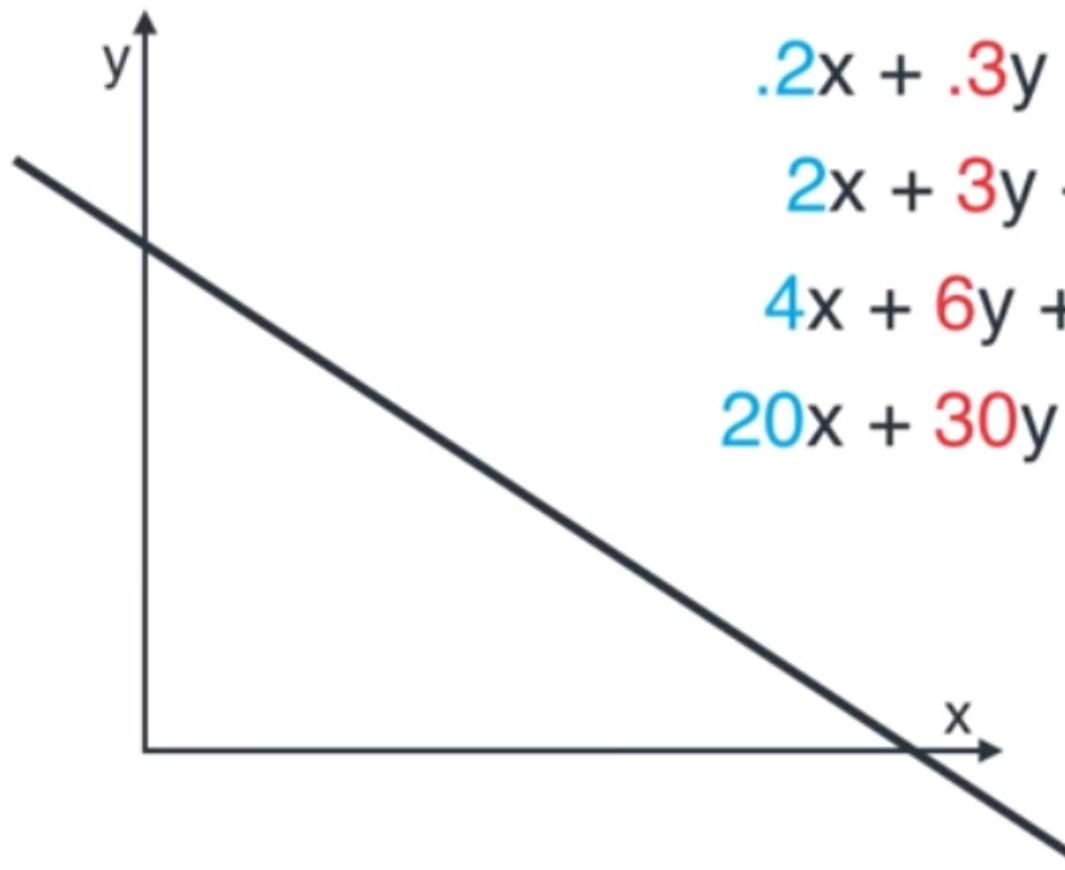
# Split data - separate lines



# Split data - separate lines



# How to separate lines?



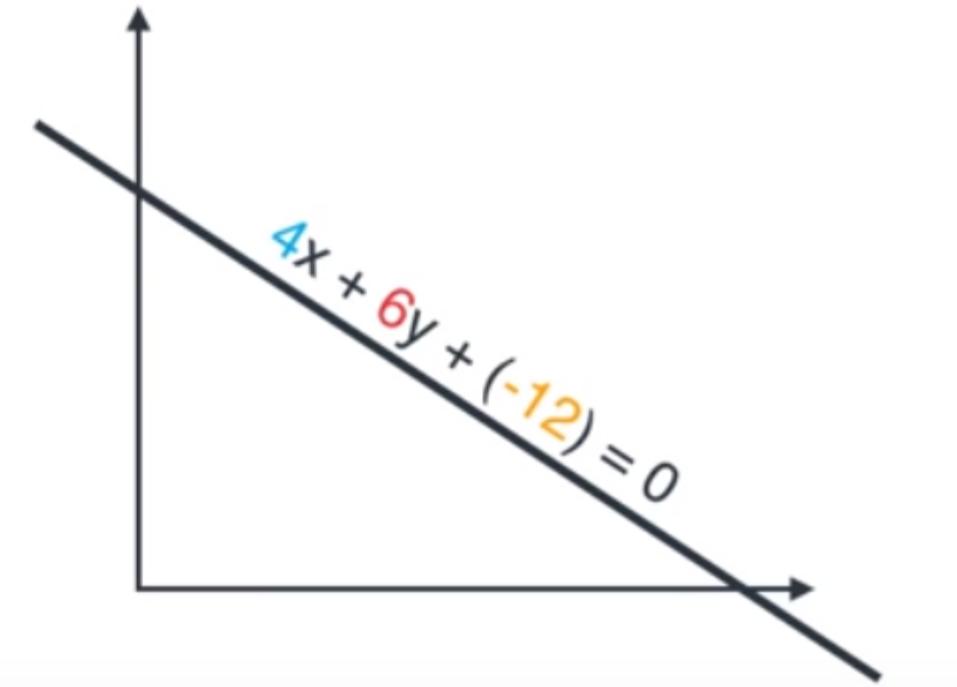
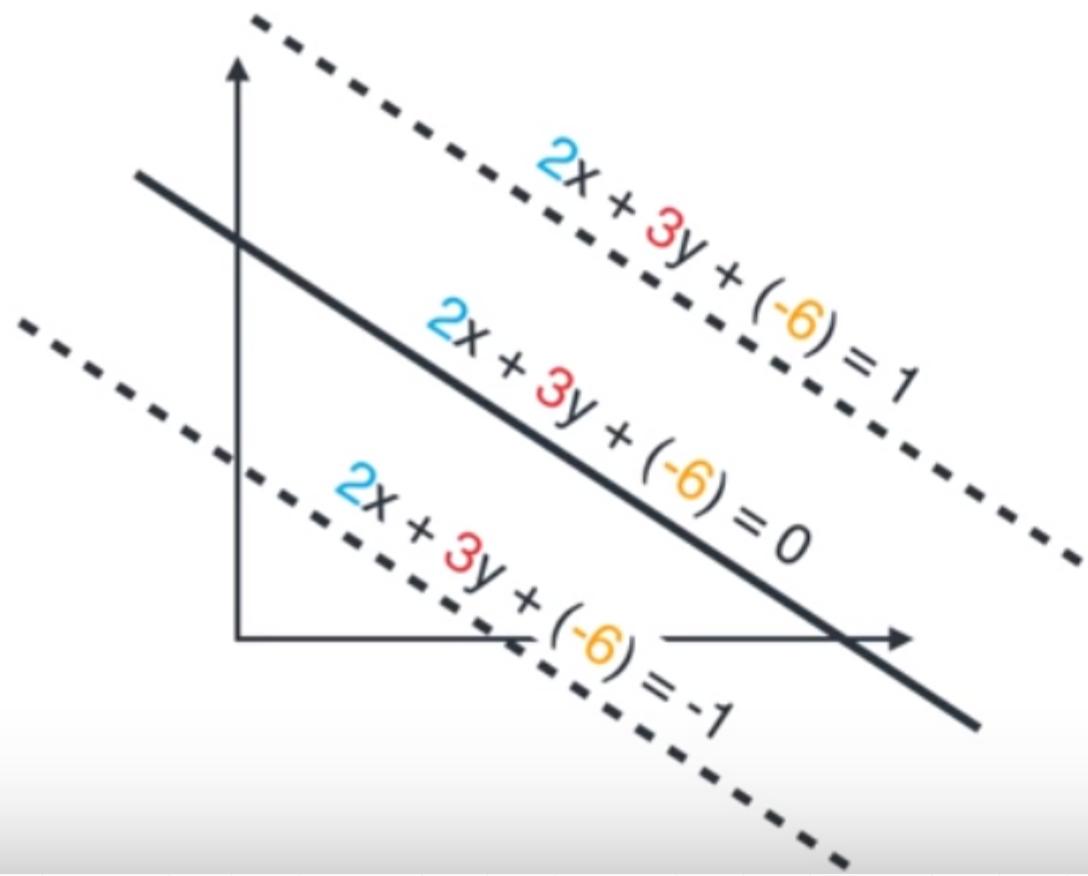
$$.2x + .3y + (-.6) = 0$$

$$2x + 3y + (-6) = 0$$

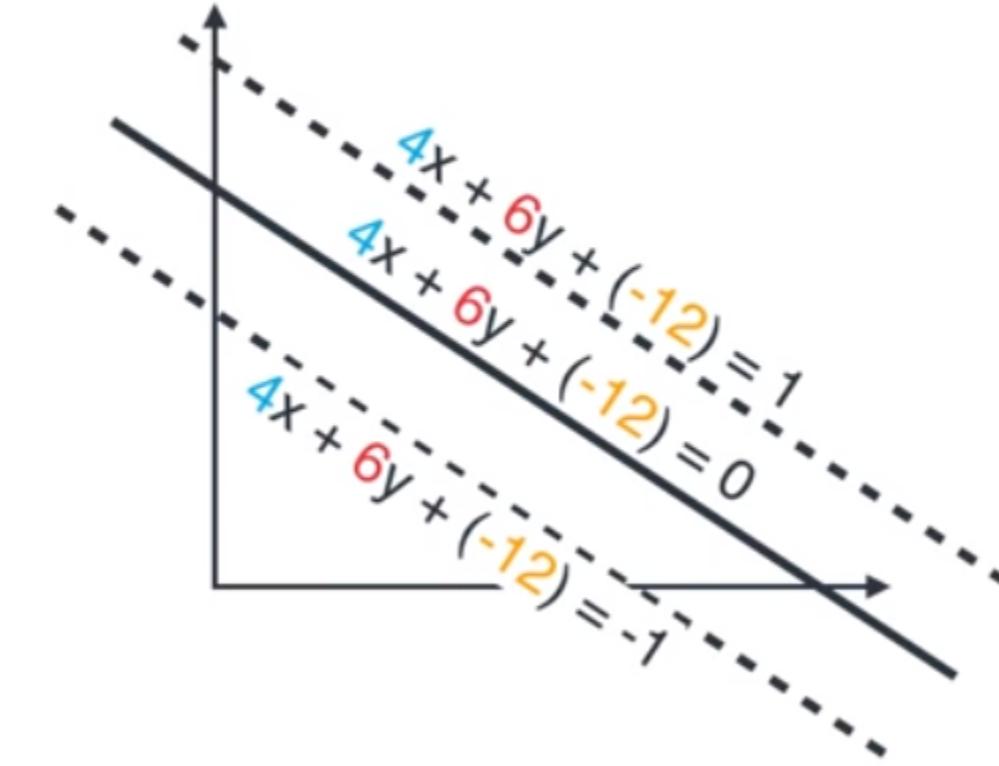
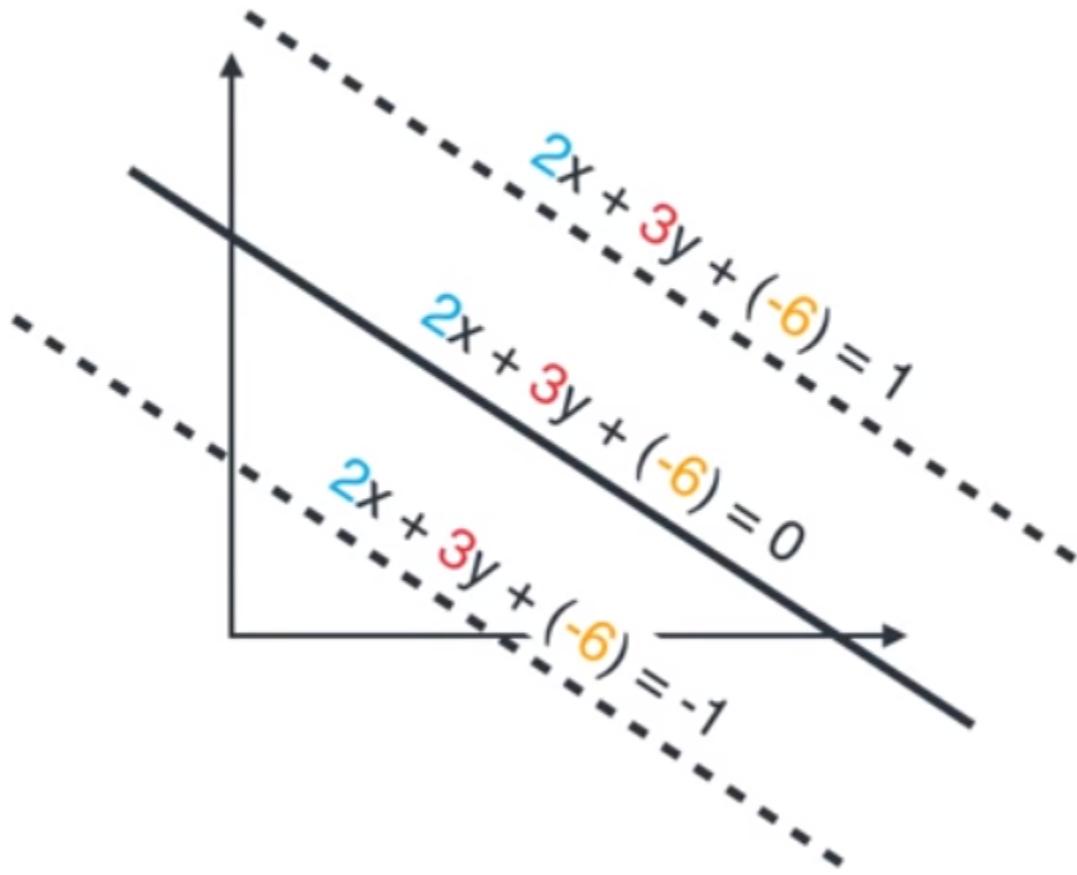
$$4x + 6y + (-12) = 0$$

$$20x + 30y + (-60) = 0$$

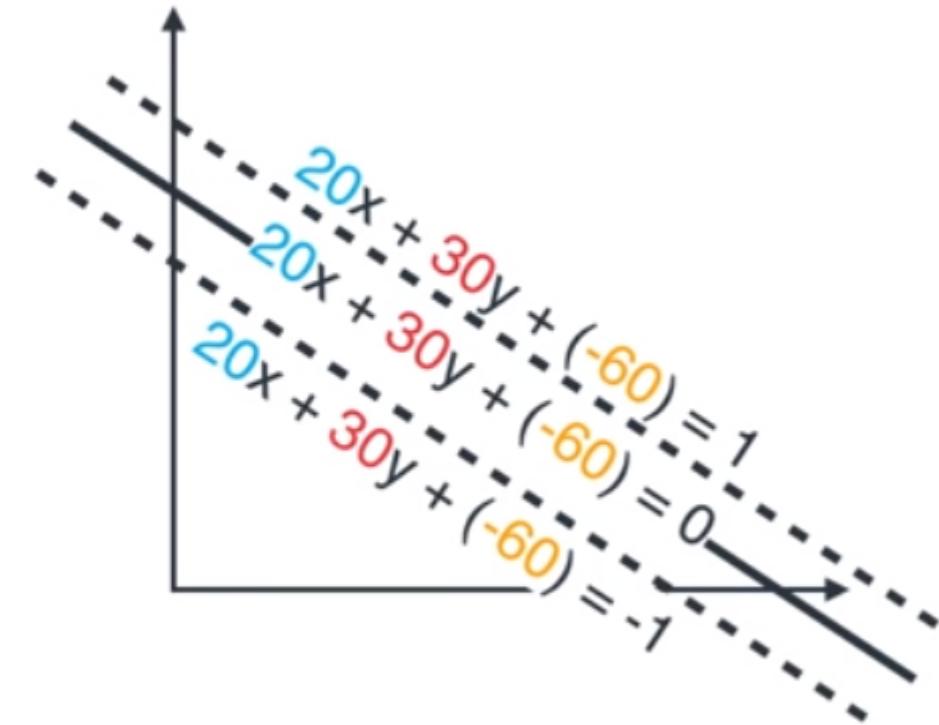
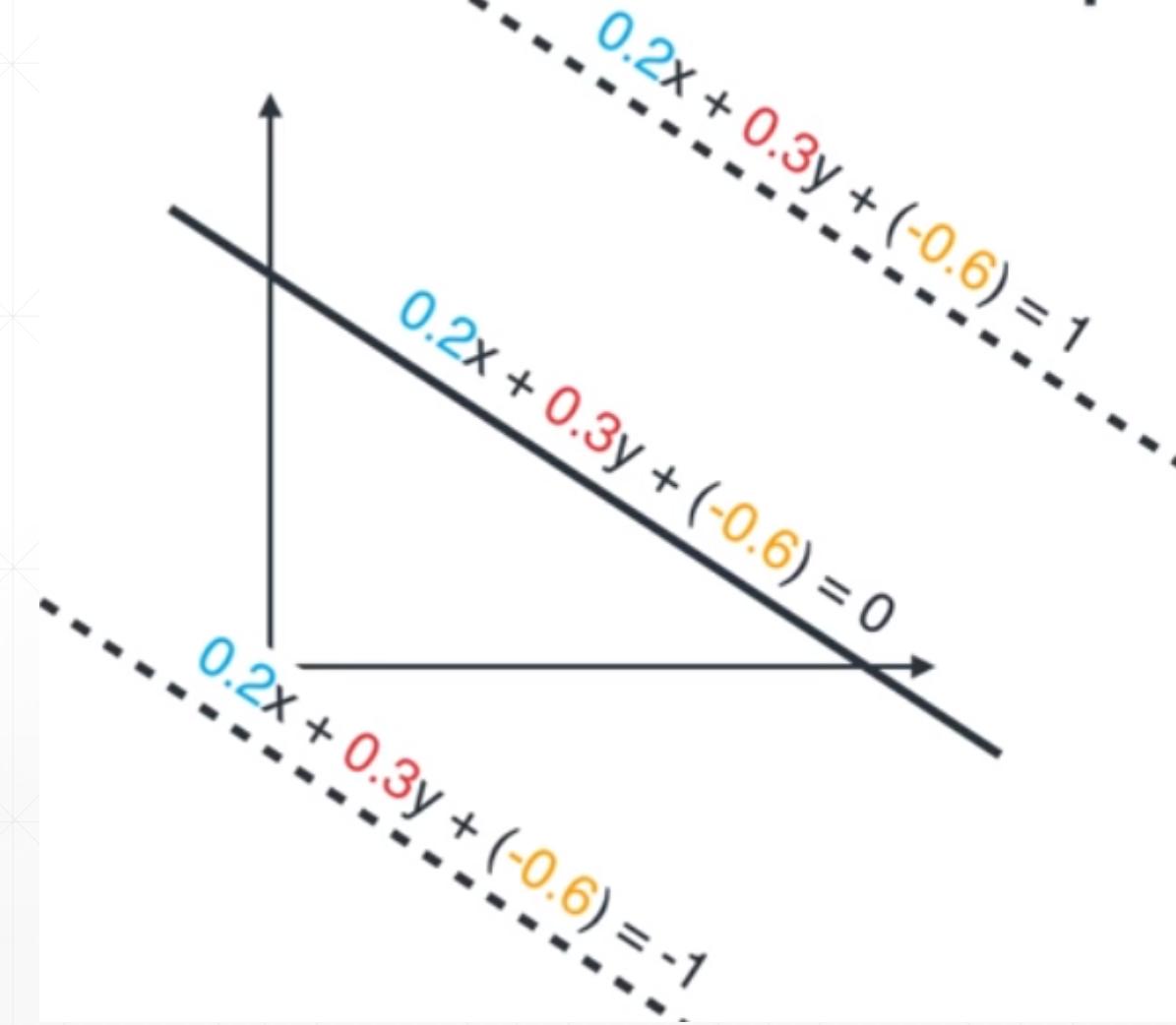
# How to separate lines?



# How to separate lines?

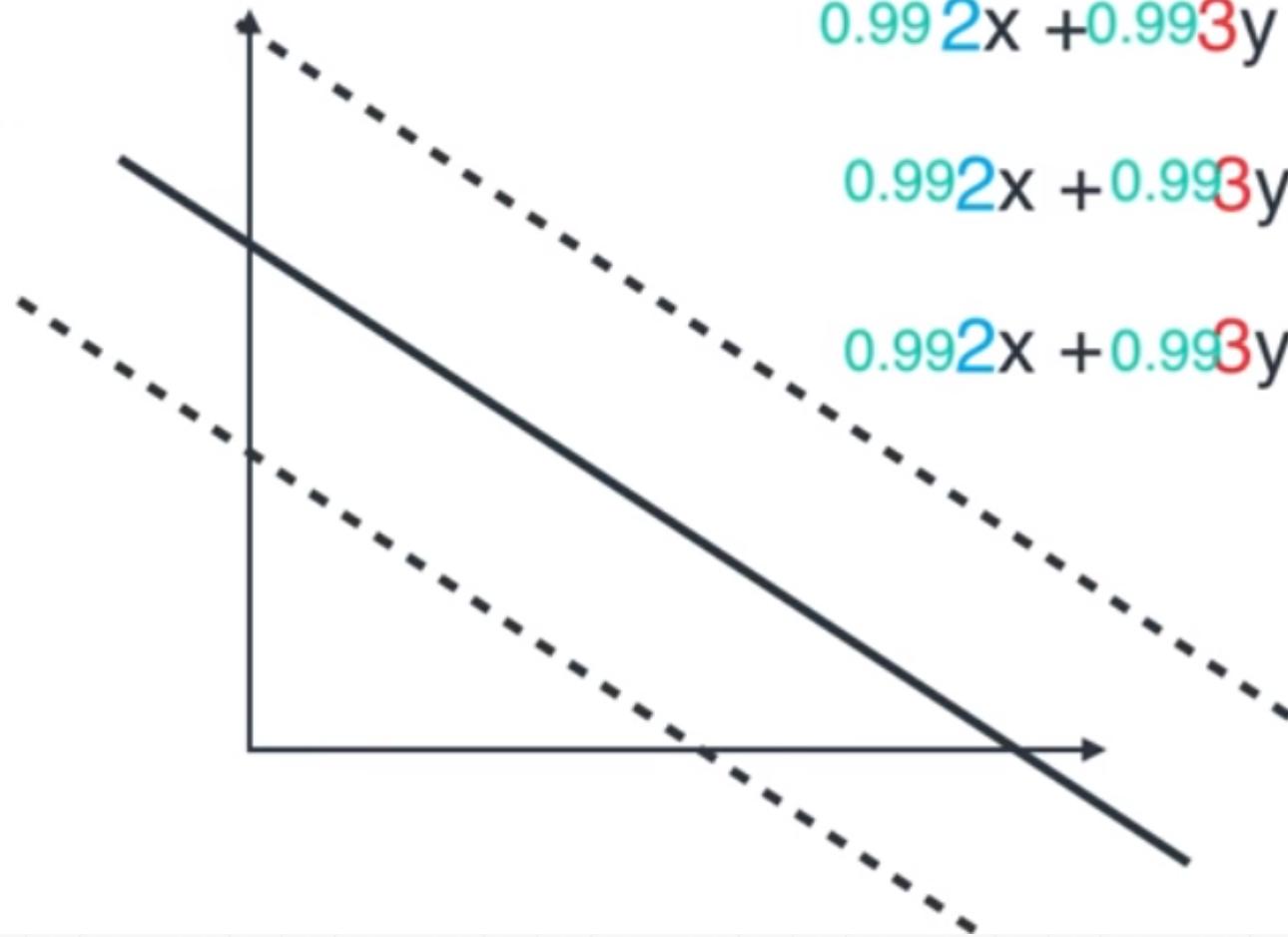


# How to separate lines?



# Expanding rate

Expanding rate



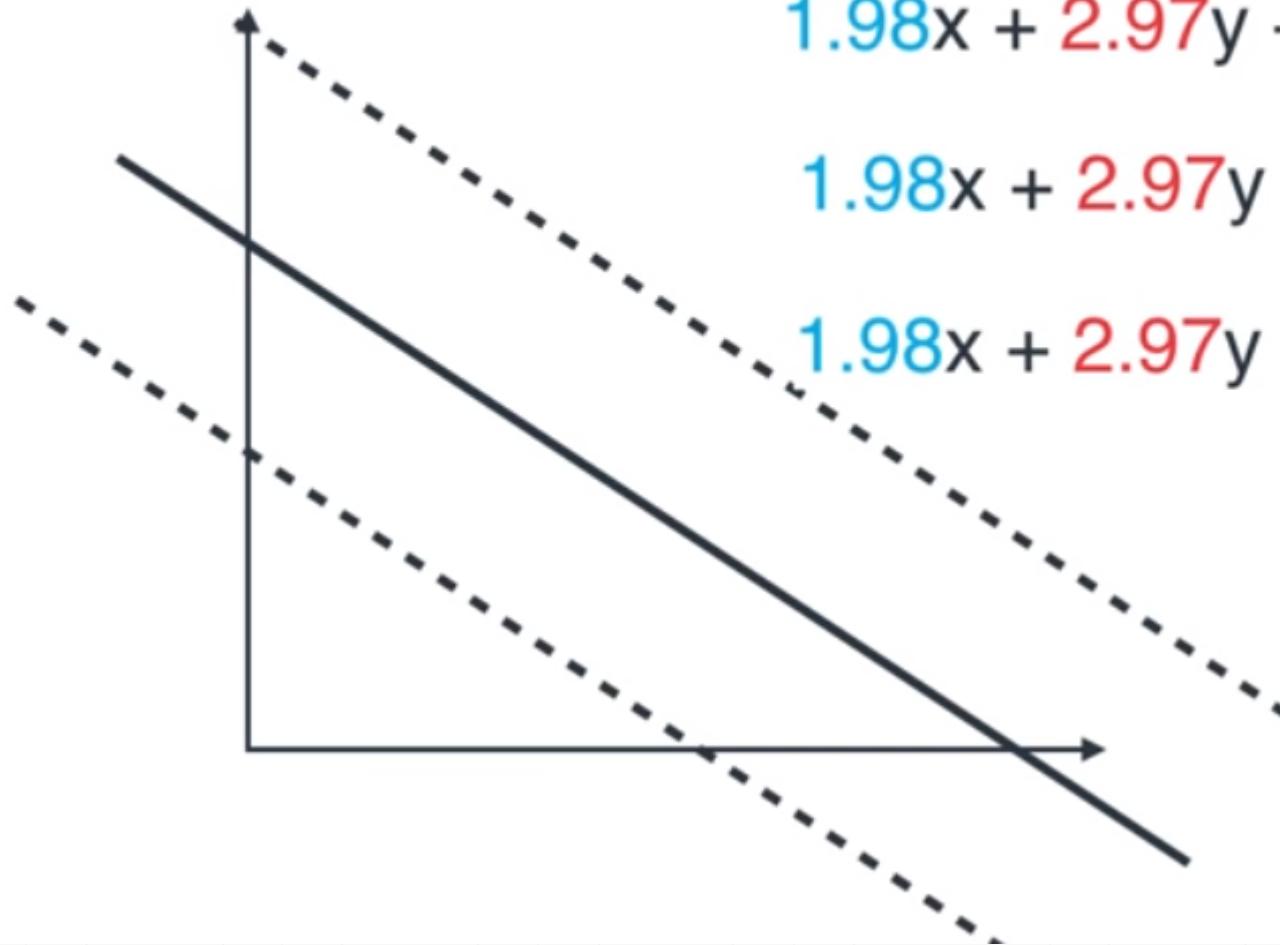
$$0.992x + 0.993y + 0.99(-6) = -1$$

$$0.992x + 0.993y + 0.99(-6) = 0$$

$$0.992x + 0.993y + 0.99(-6) = 1$$

# Expanding rate

Expanding rate

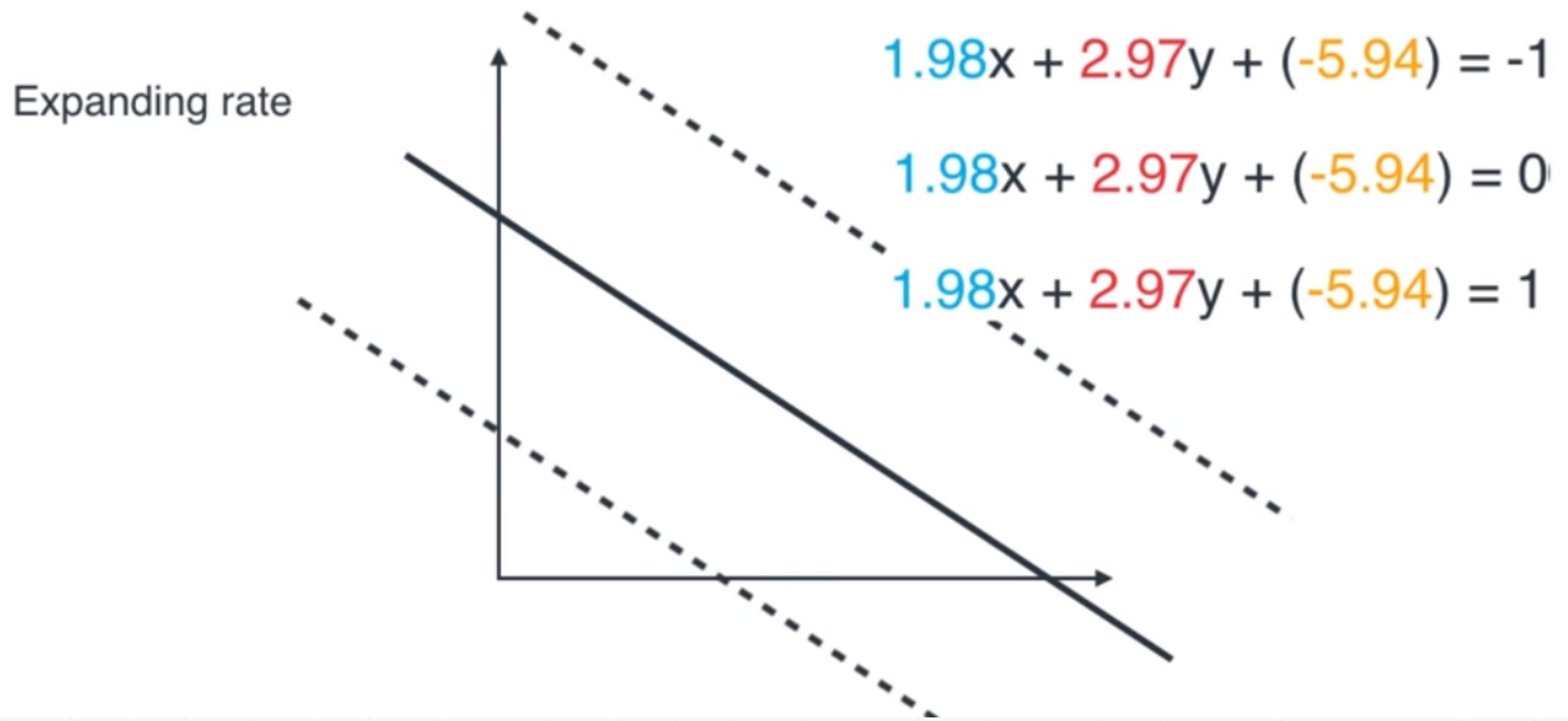


$$1.98x + 2.97y + (-5.94) = -1$$

$$1.98x + 2.97y + (-5.94) = 0$$

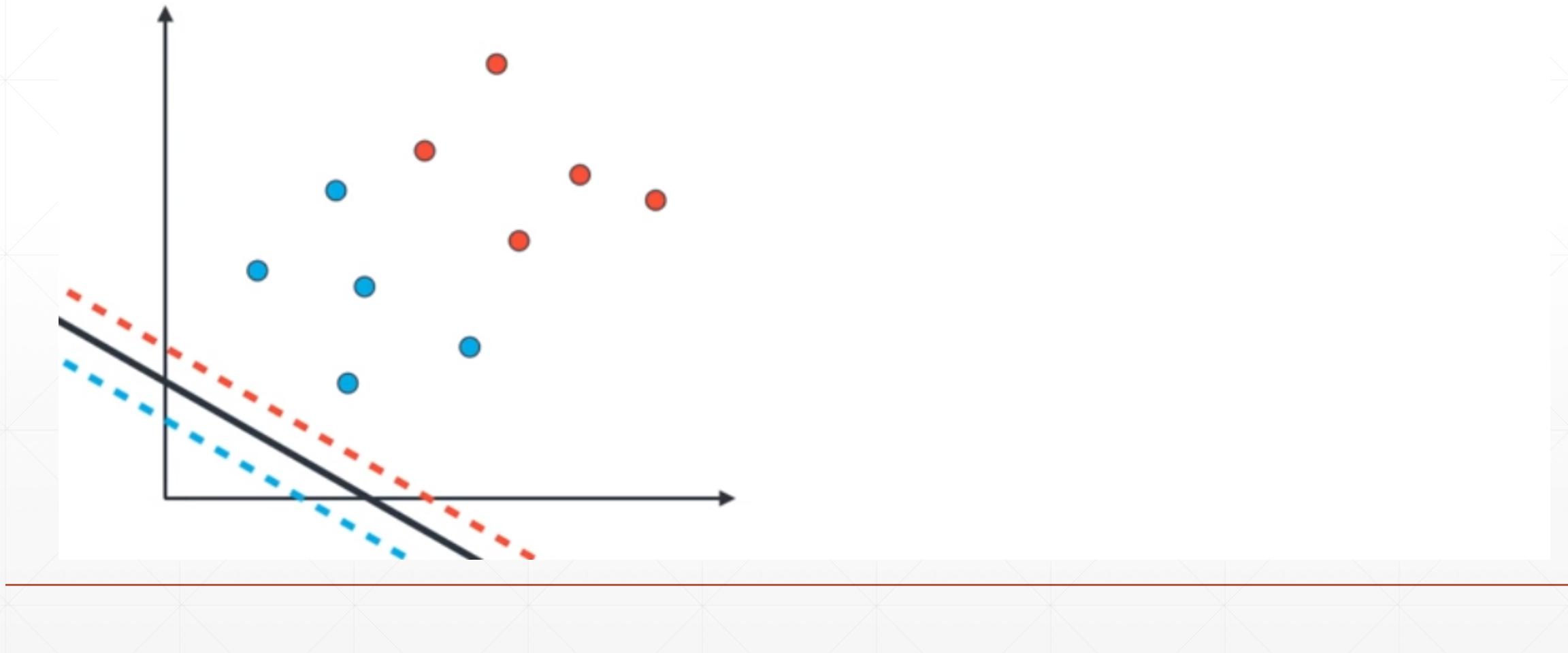
$$1.98x + 2.97y + (-5.94) = 1$$

# Expanding rate



# SVM algorithm

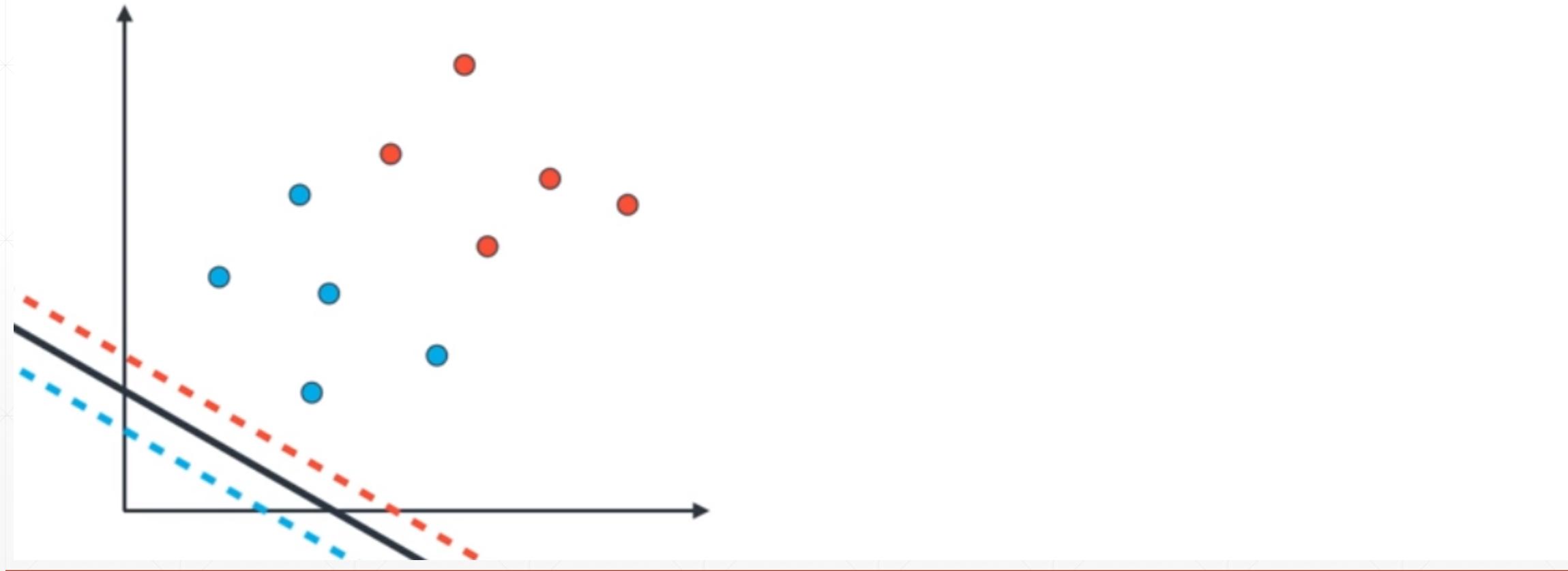
**Step 1:** Start with a line, and two equidistant parallel lines to it.



# SVM algorithm

**Step 1:** Start with a line, and two equidistant parallel lines to it.

**Step 2:** Pick a large number. **1000** (number of repetitions, or epochs)

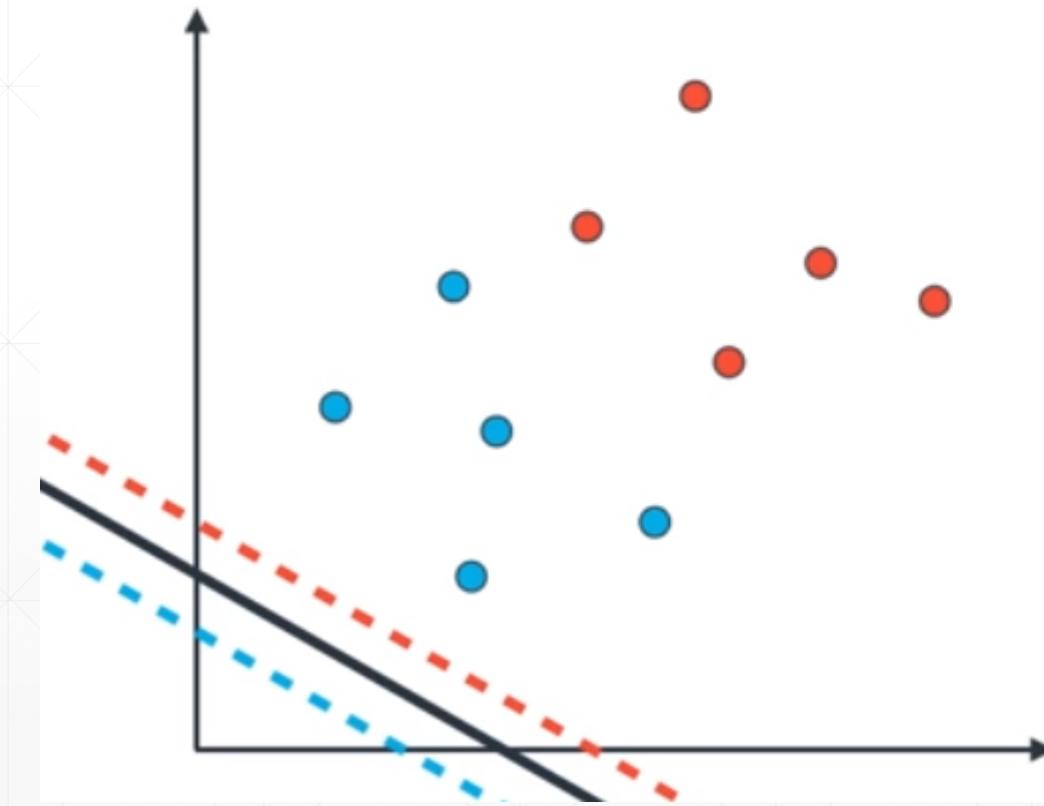


# SVM algorithm

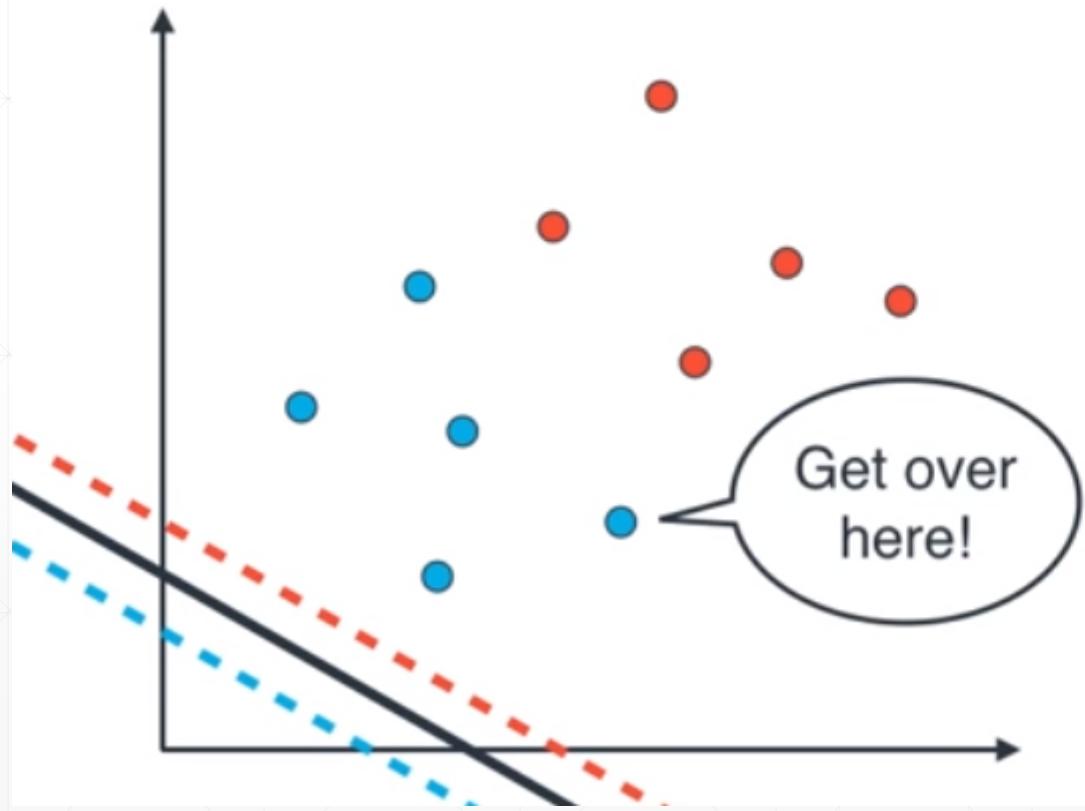
**Step 1:** Start with a line, and two equidistant parallel lines to it.

**Step 2:** Pick a large number. **1000** (number of repetitions, or epochs)

**Step 3:** Pick a number close to 1. (the expanding factor) **0.99**



# SVM algorithm



**Step 1:** Start with a line, and two equidistant parallel lines to it.

**Step 2:** Pick a large number. **1000** (number of repetitions, or epochs)

**Step 3:** Pick a number close to 1. (the expanding factor) **0.99**

**Step 4:** (repeat **1000** times)

- Pick random point
- If point is correctly classified:
  - Do nothing
- If point is incorrectly classified
  - Move line towards point

# SVM algorithm

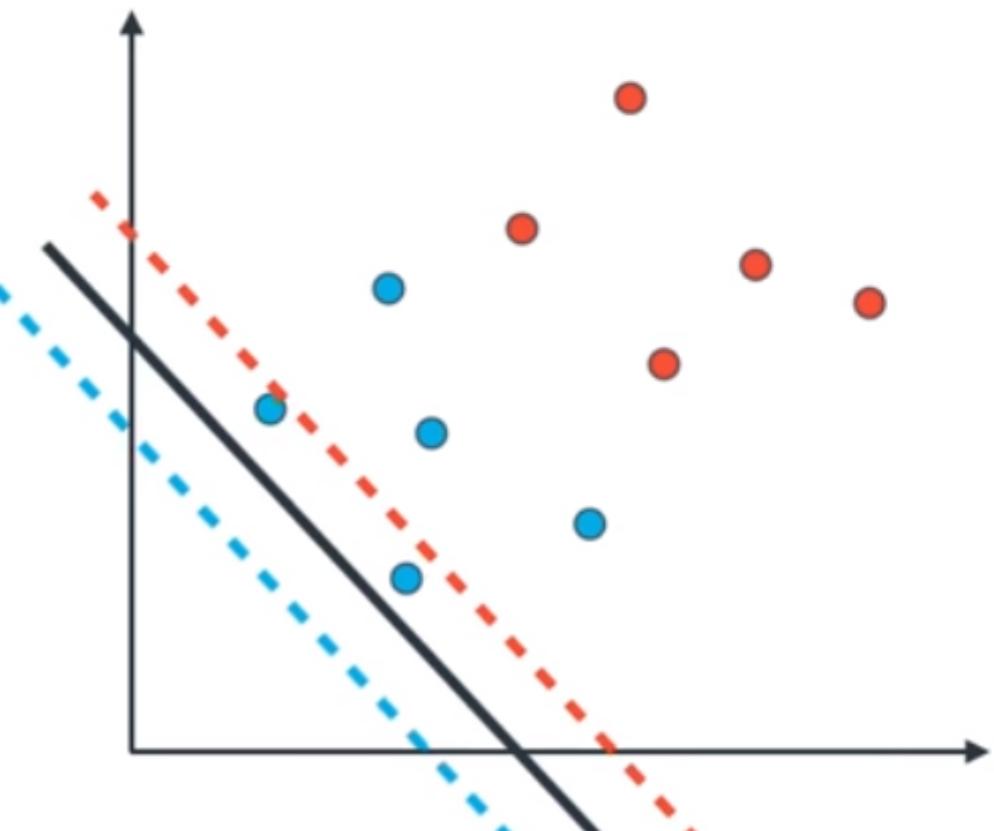
**Step 1:** Start with a line, and two equidistant parallel lines to it.

**Step 2:** Pick a large number. **1000** (number of repetitions, or epochs)

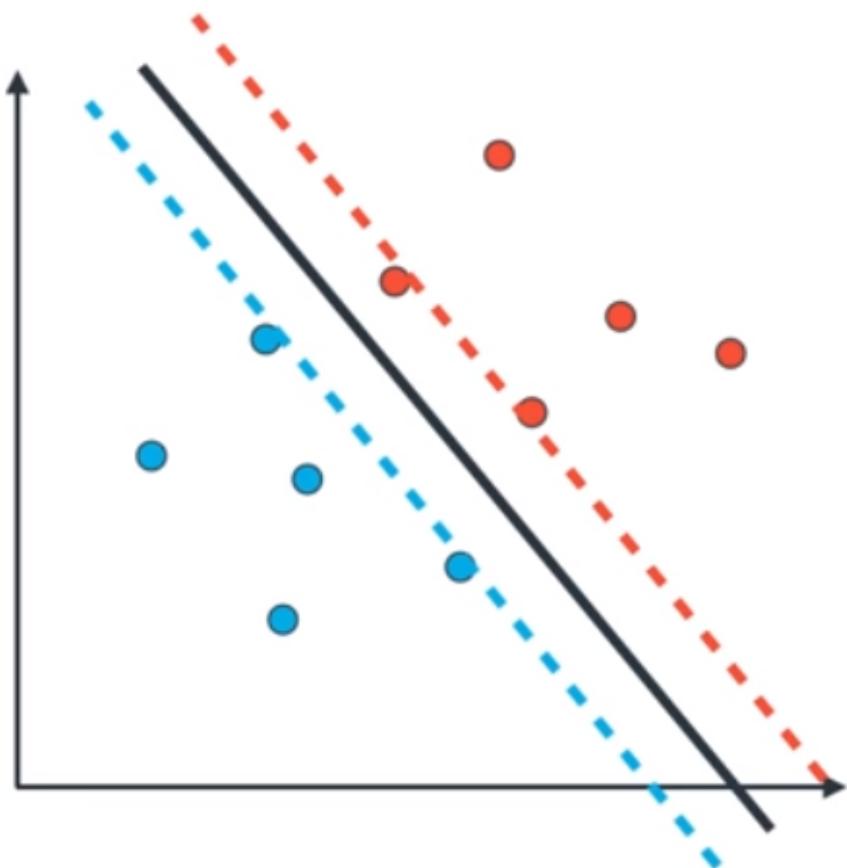
**Step 3:** Pick a number close to 1. (the expanding factor) **0.99**

**Step 4:** (repeat **1000** times)

- Pick random point
- If point is correctly classified:
  - Do nothing
- If point is incorrectly classified
  - Move line towards point
- Separate the lines using the expanding factor



# SVM algorithm



**Step 1:** Start with a line, and two equidistant parallel lines to it.

**Step 2:** Pick a large number. **1000** (number of repetitions, or epochs)

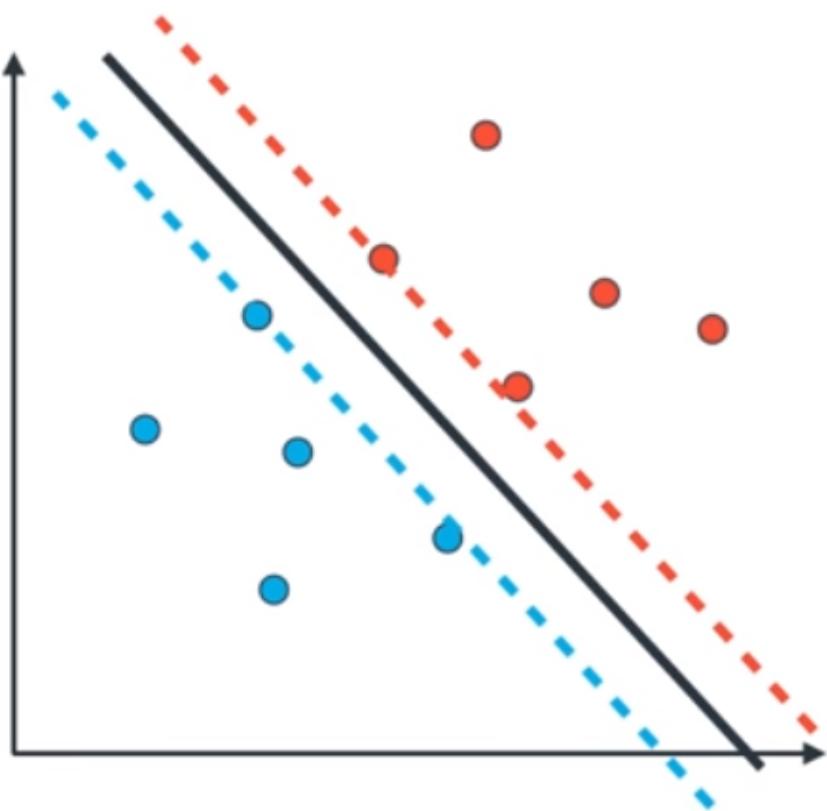
**Step 3:** Pick a number close to 1. (the expanding factor) **0.99**

**Step 4:** (repeat **1000** times)

- Pick random point
- If point is correctly classified:
  - Do nothing
- If point is incorrectly classified
  - Move line towards point
- Separate the lines using the expanding factor

**Step 5:** Enjoy your lines that separate the data!

# SVM algorithm



**Step 1:** Start with a random line of equation  $ax + by + c = 0$ .  
Draw parallel lines with equations:

- $ax + by + c = 1$ , and
- $ax + by + c = -1$

**Step 2:** Pick a large number. **1000** (number of repetitions, or epochs)

**Step 3:** Pick a learning rate. **0.01**

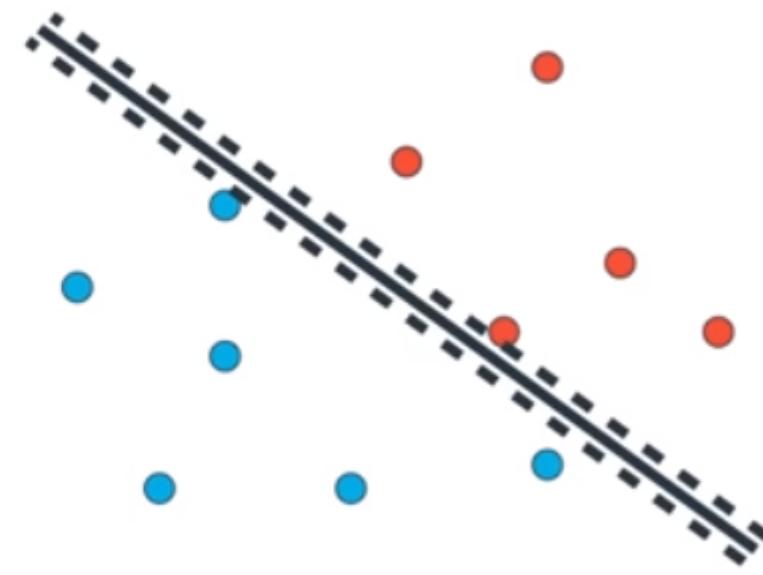
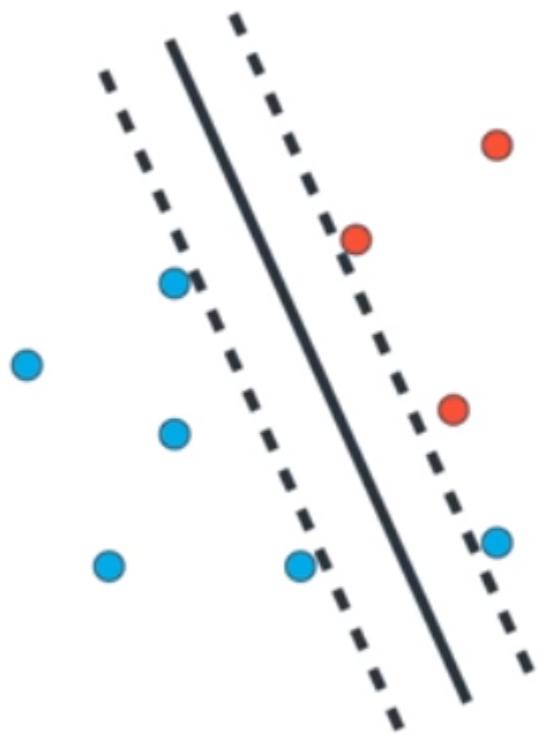
**Step 4:** Pick an expanding rate. **0.99**

**Step 5:** (repeat **1000** times)

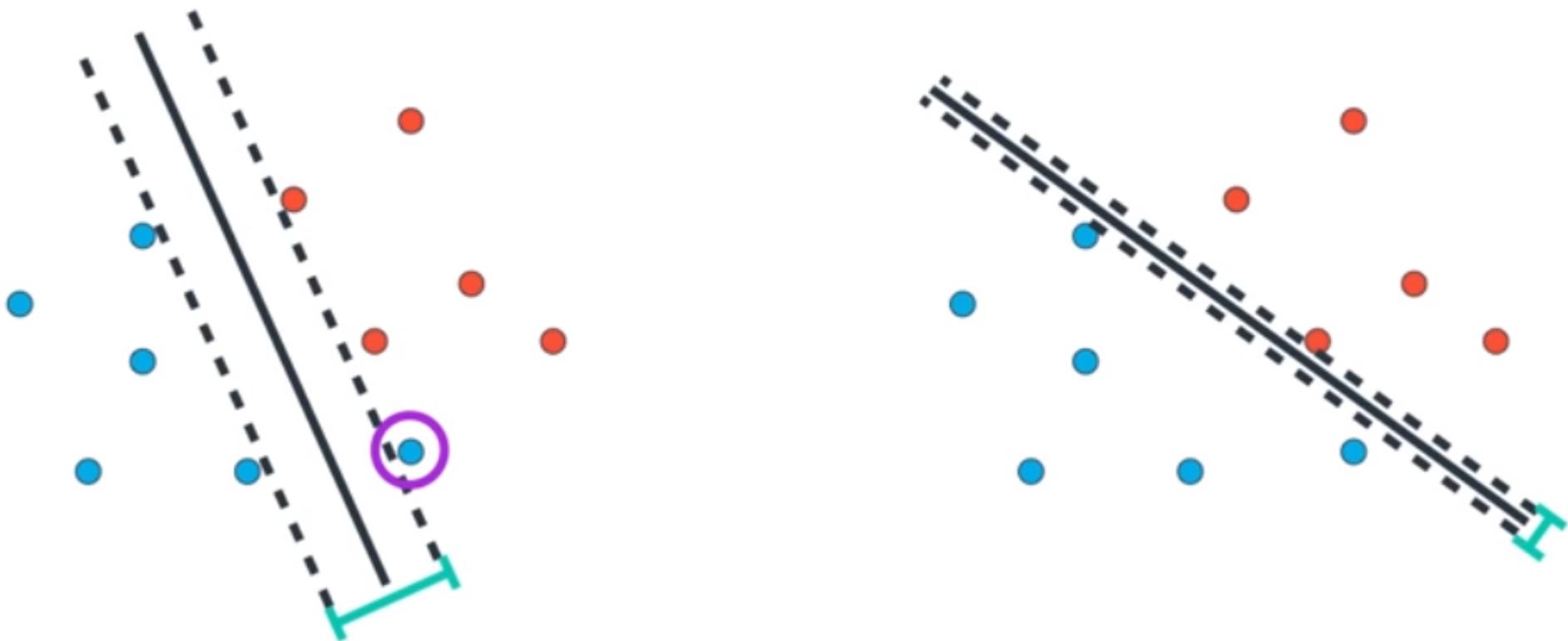
- Pick random point **(p,q)**
- If point is correctly classified
  - Do nothing
- If point is **blue**, and  $ap+bq+c > 0$ 
  - Subtract  $0.01p$  to  $a$
  - Subtract  $0.01q$  to  $b$
  - Subtract  $0.01$  to  $c$
- If point is, **red** and  $ap+bq+c < 0$ 
  - Add  $0.01p$  to  $a$
  - Add  $0.01q$  to  $b$
  - Add  $0.01$  to  $c$

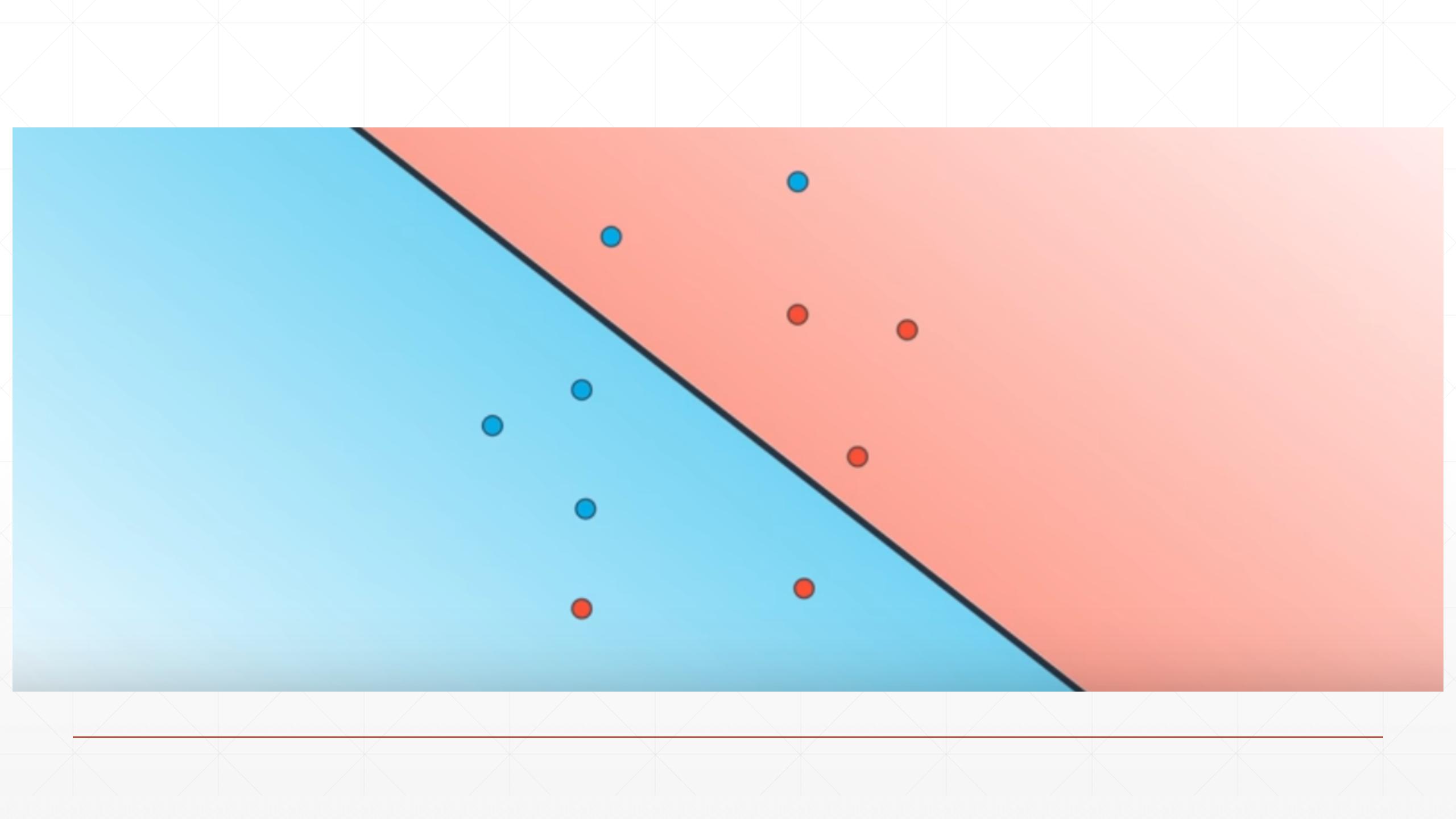
- **Multiply  $a$ ,  $b$ ,  $c$ , by 0.99**

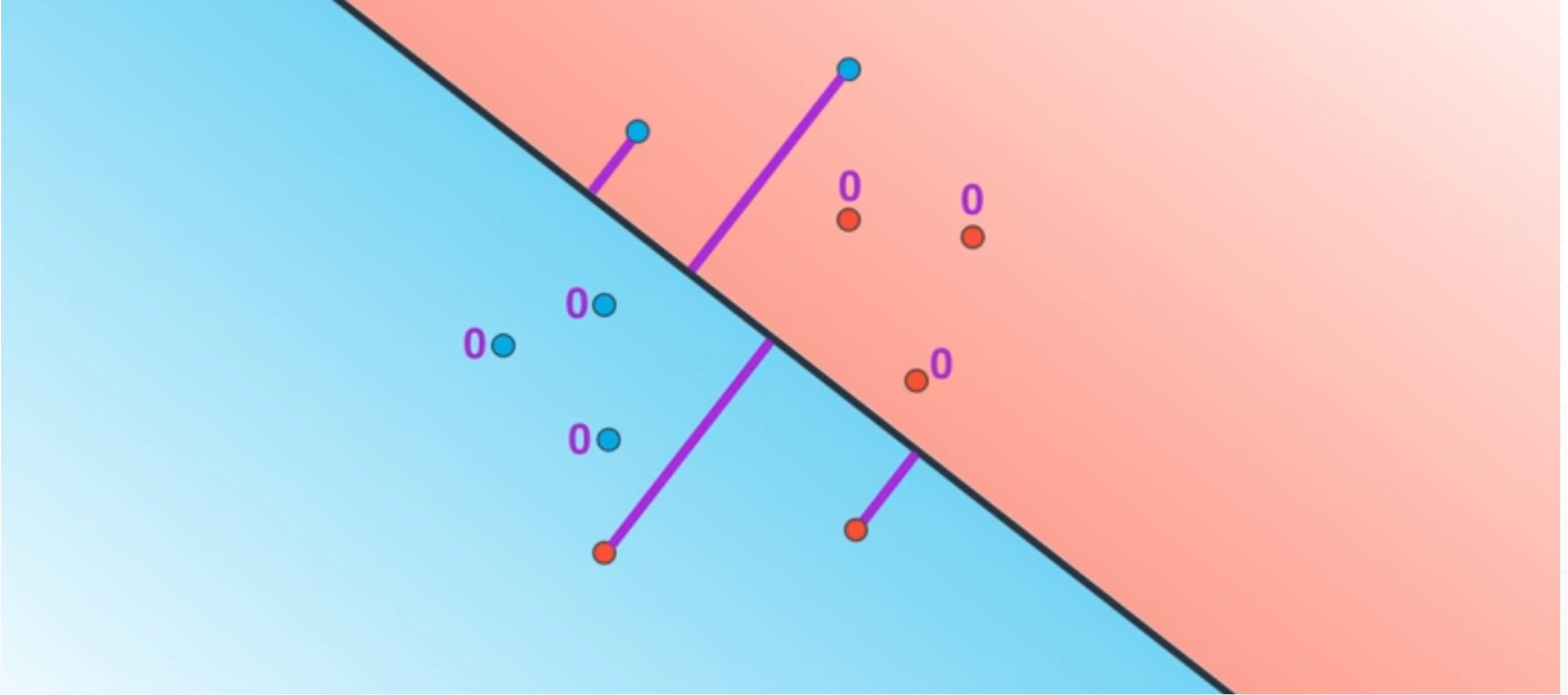
# Which line is better?



# Which line is better?







$$|2(1) + 3(1) - 6| = |-1|$$

$$(1,1)$$

Error = 1

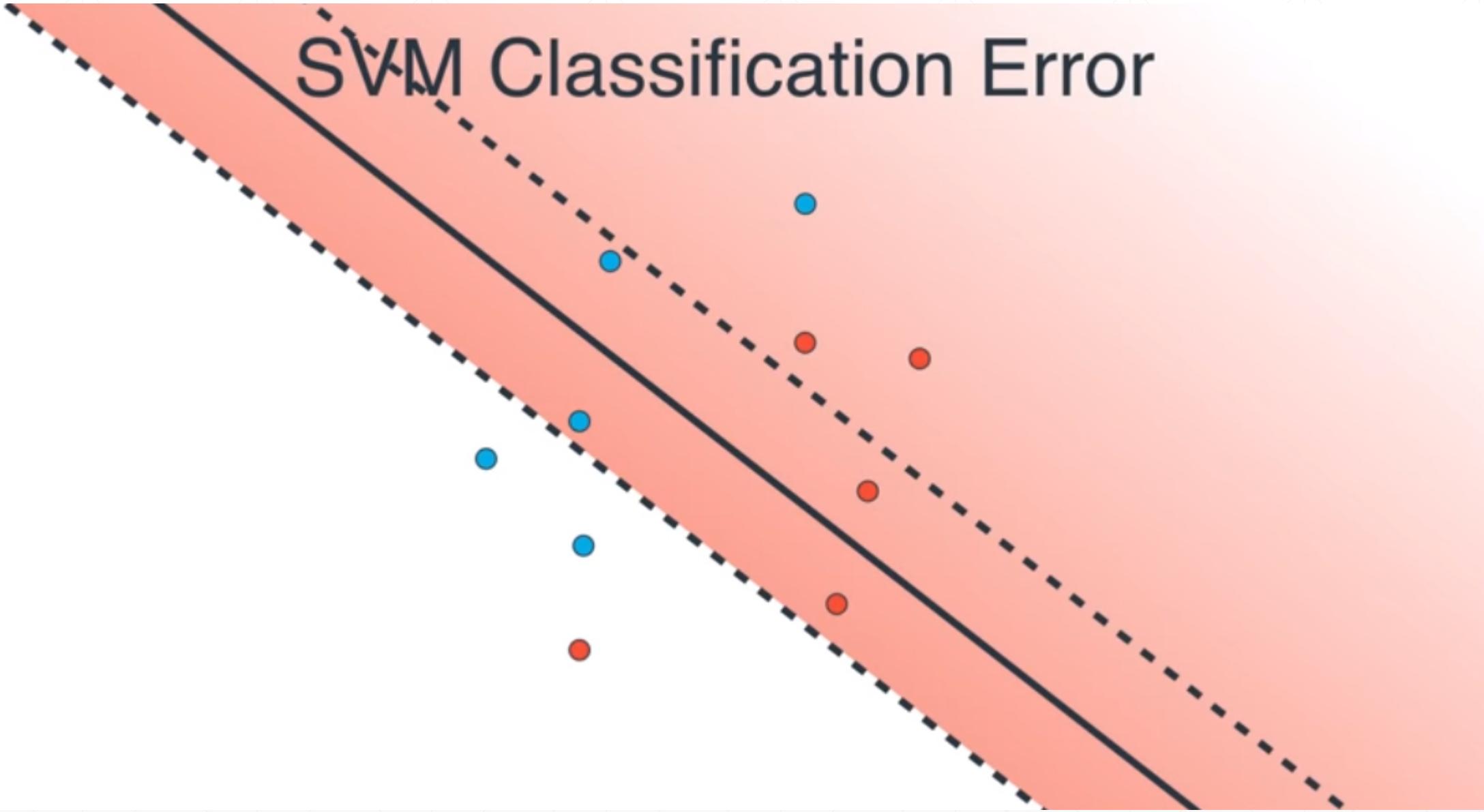
$$2x + 3y - 6 = 0$$

(4,5)

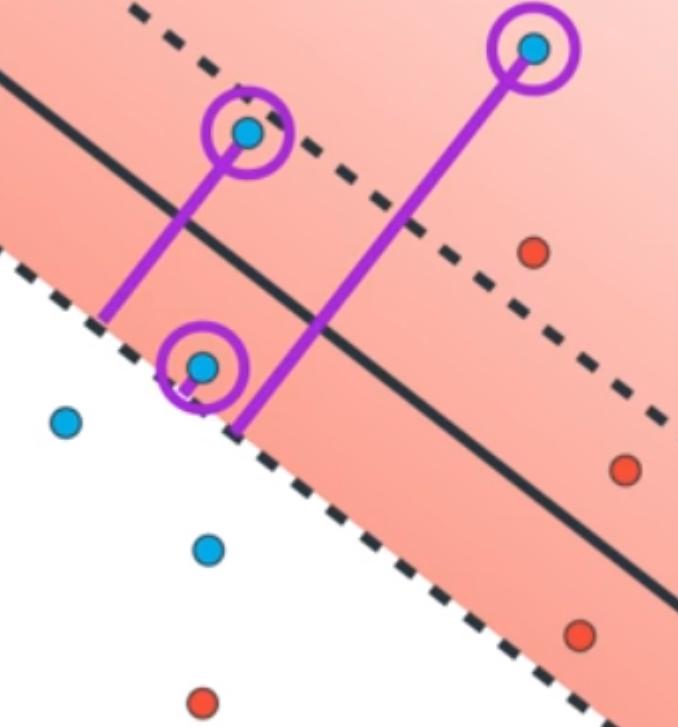
Error = 17

$$2(4) + 3(5) - 6 = 17$$

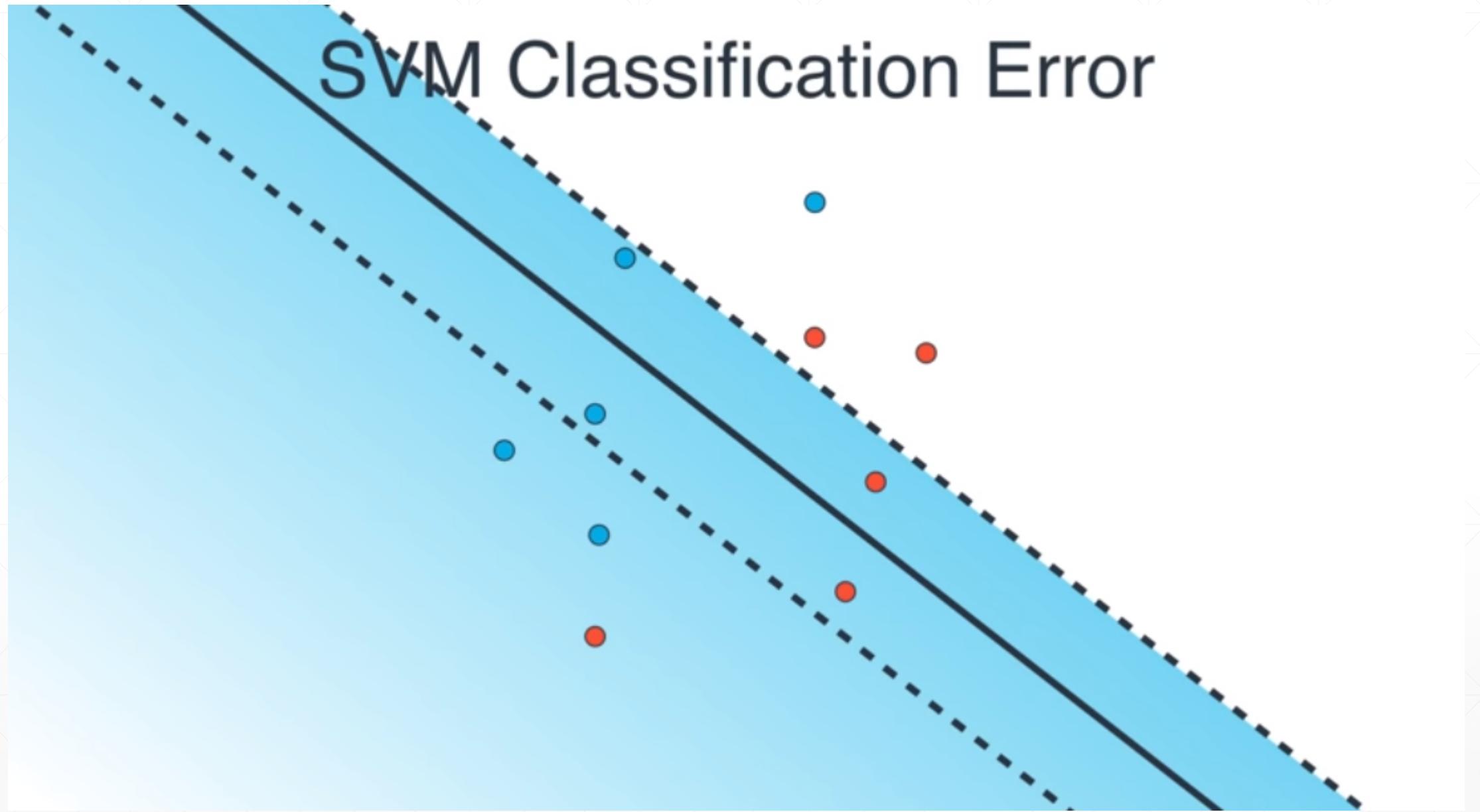
# SVM Classification Error



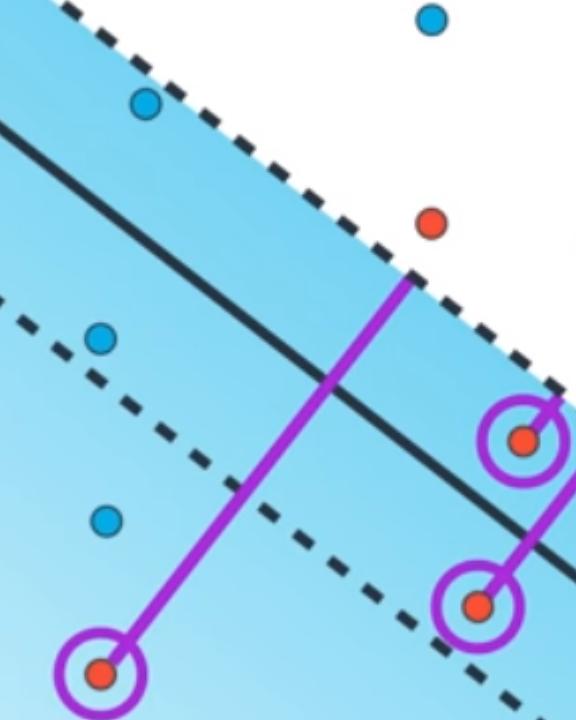
# SVM Classification Error



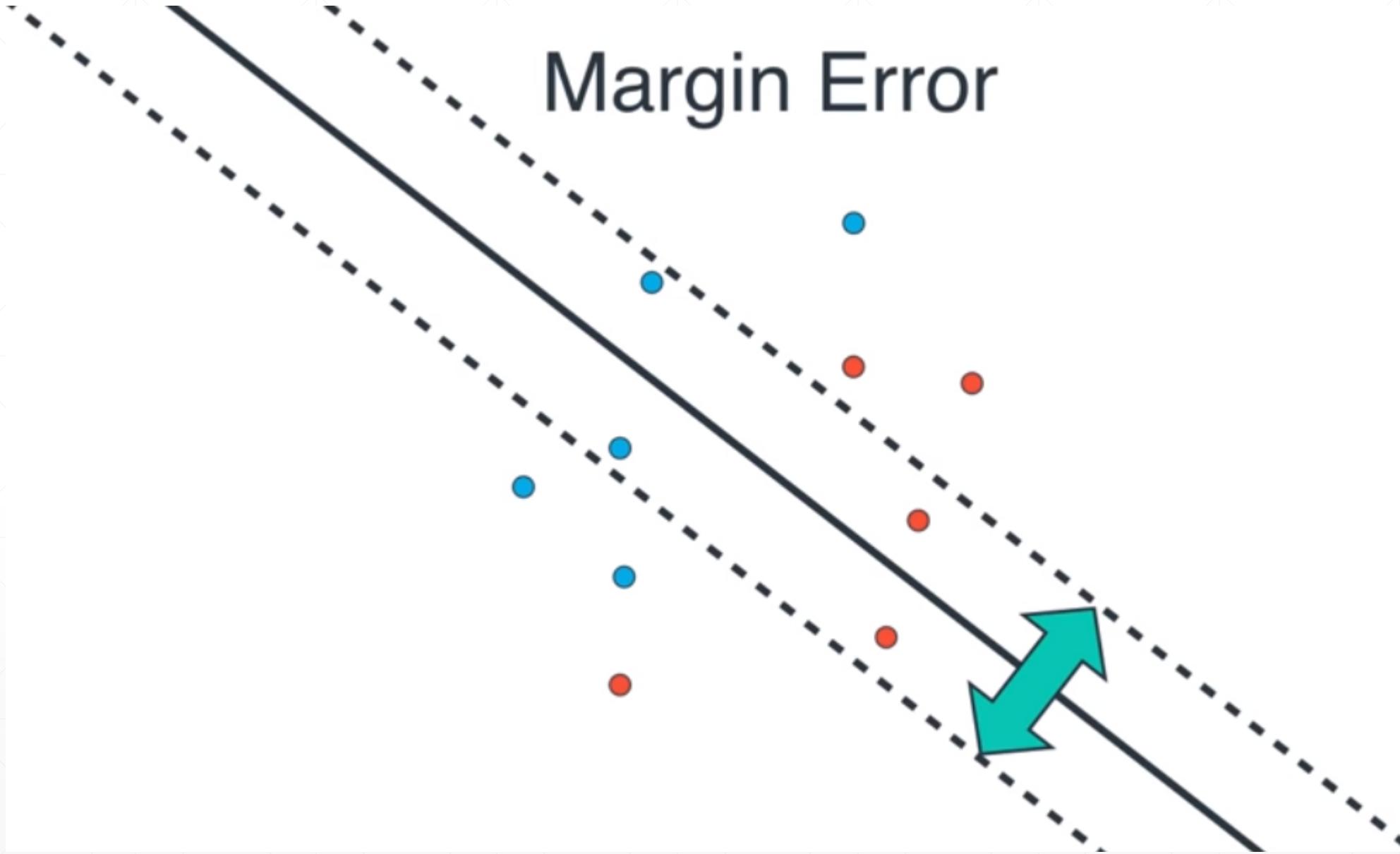
# SVM Classification Error



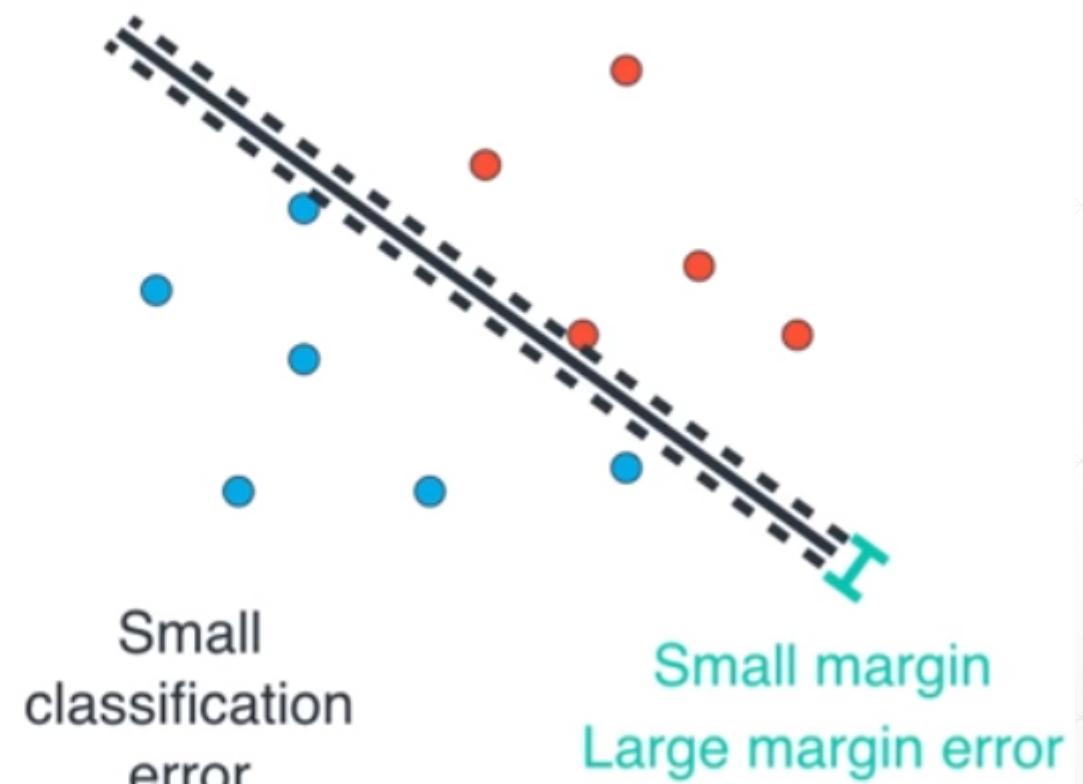
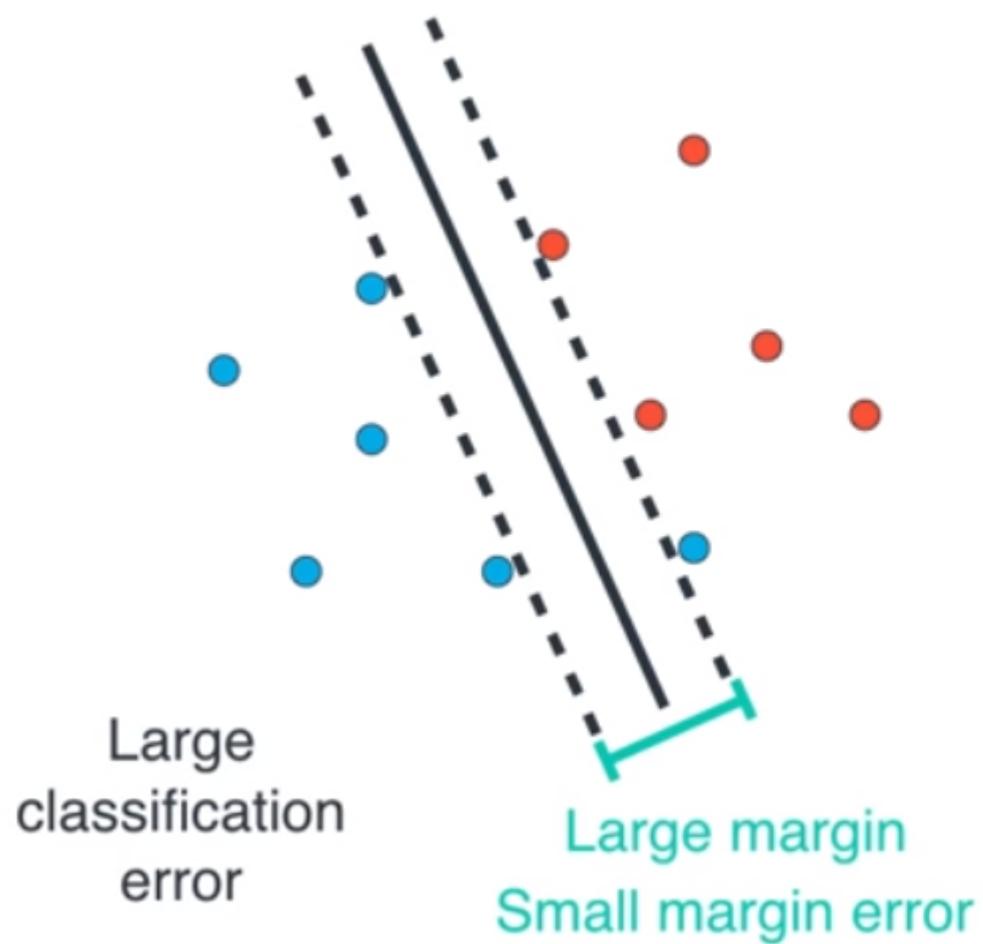
# SVM Classification Error



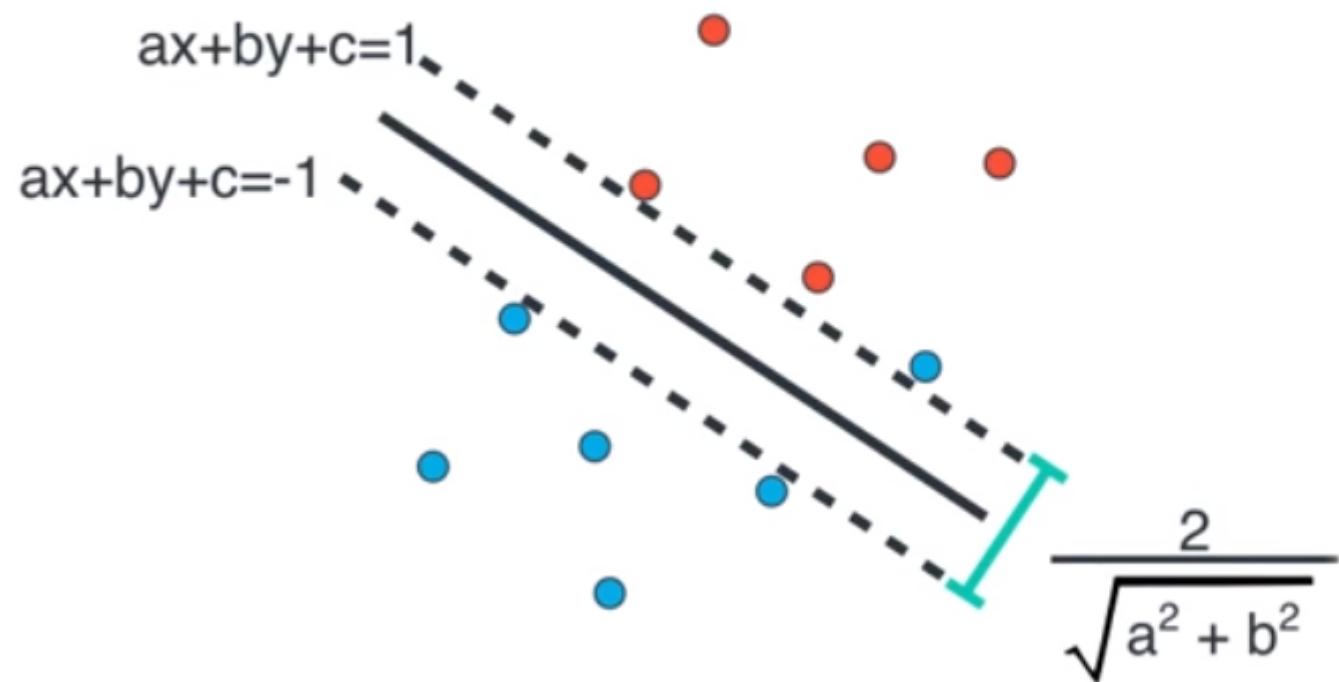
# Margin Error



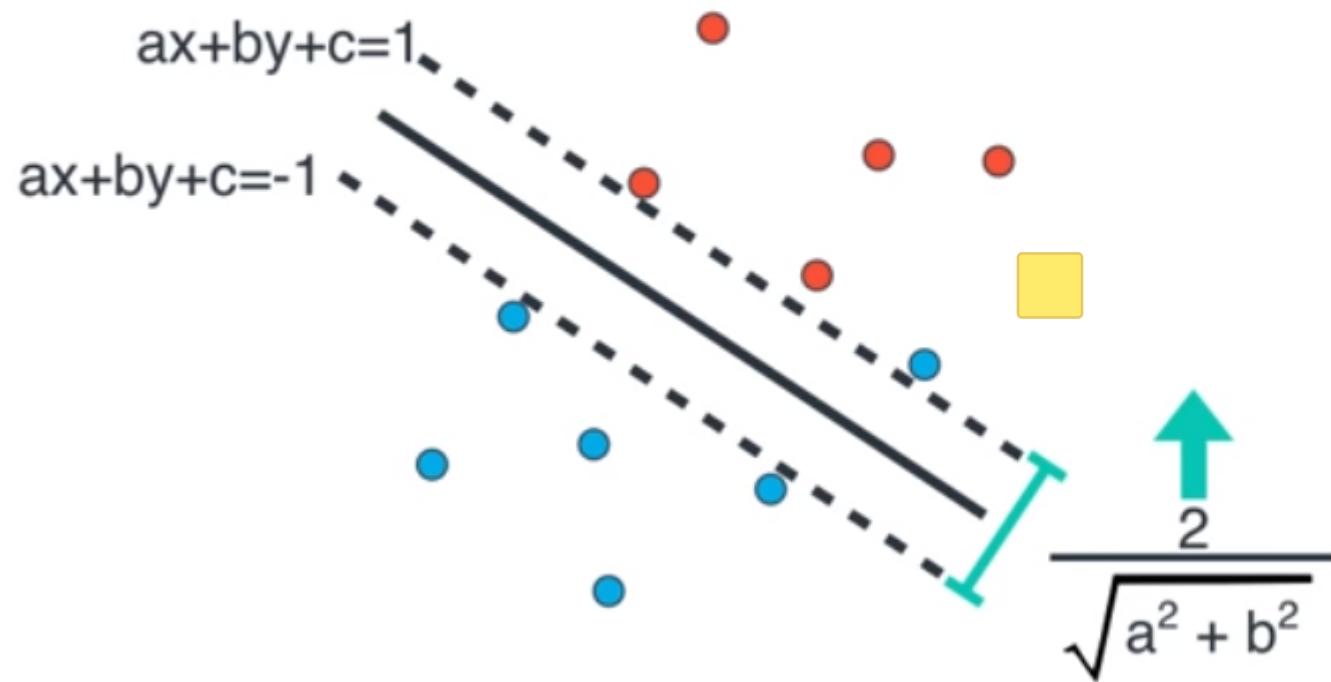
# Margin Error



# Margin Error



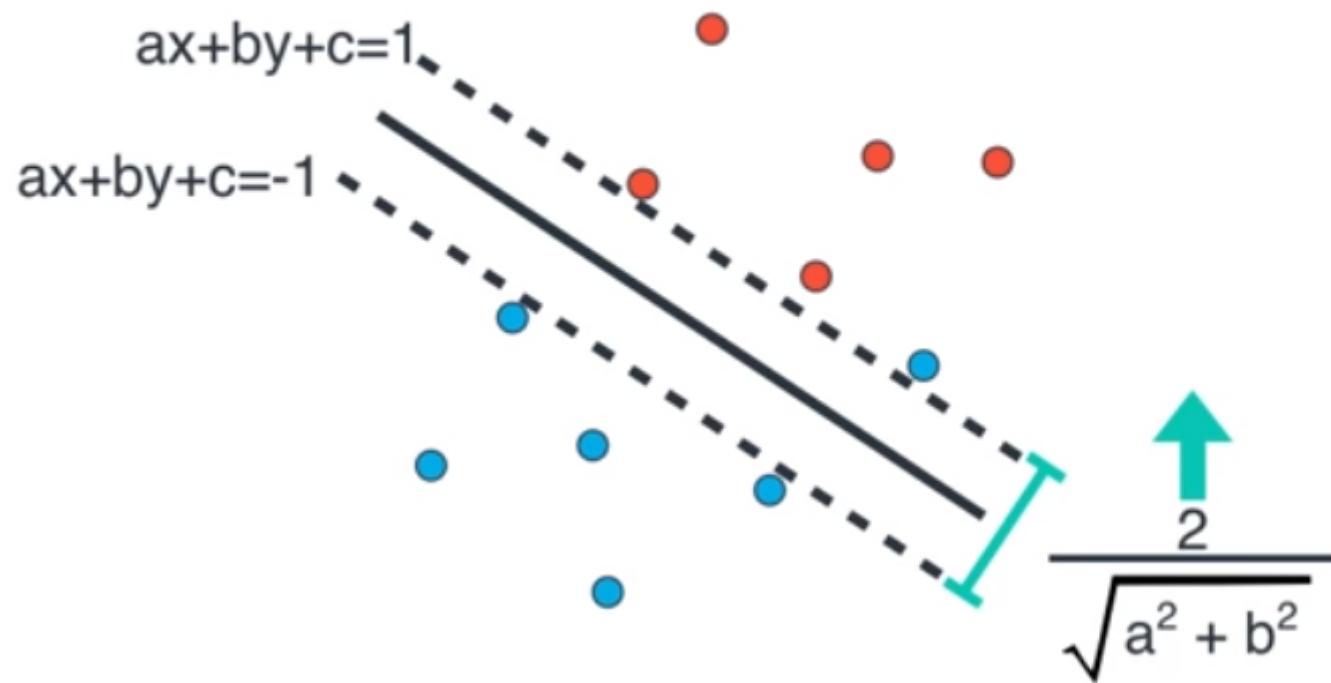
# Margin Error



$$\text{Margin error} = \frac{2}{\sqrt{a^2 + b^2}}$$



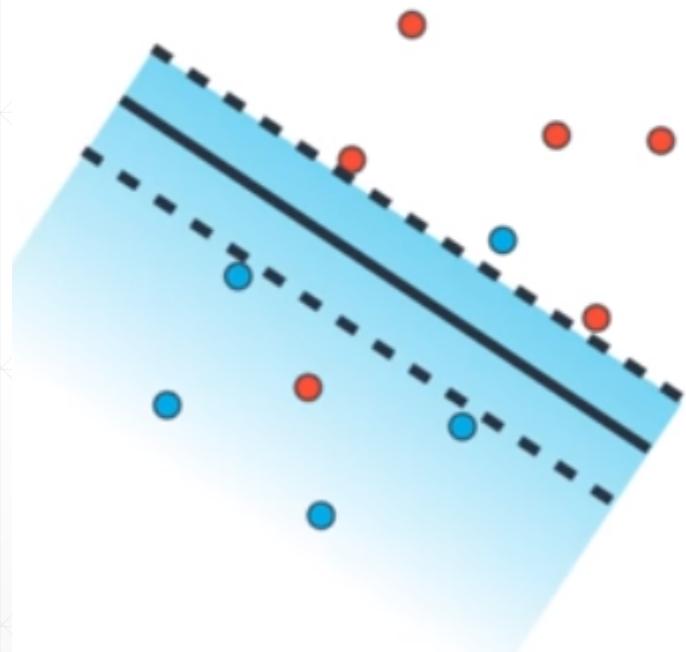
# Margin Error



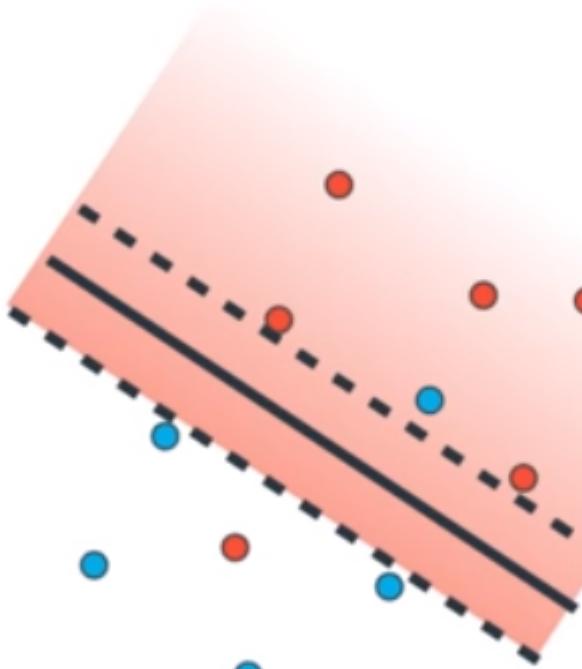
Regularization term!

$$\text{Margin error} = a^2 + b^2$$

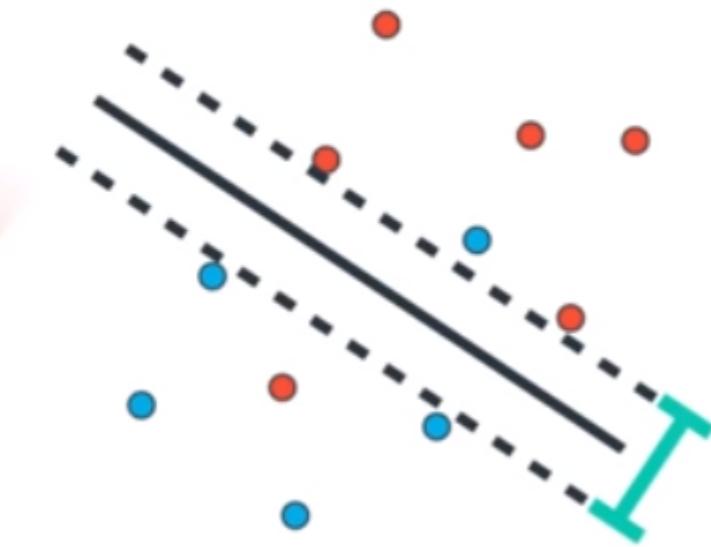
# SVM Error



Blue Classification Error

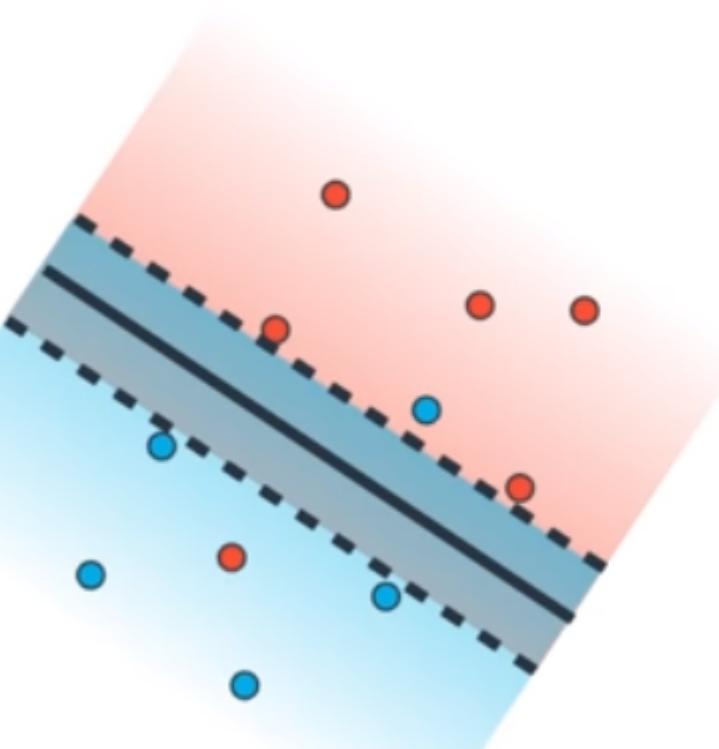


Red Classification Error

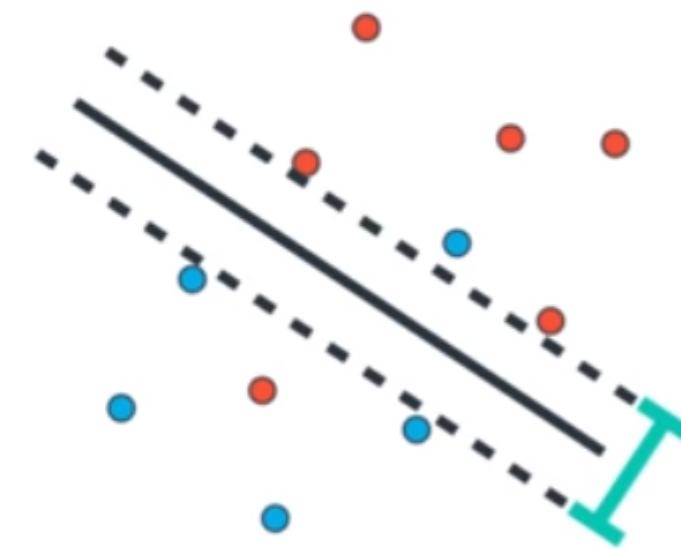


Margin Error

# SVM Error

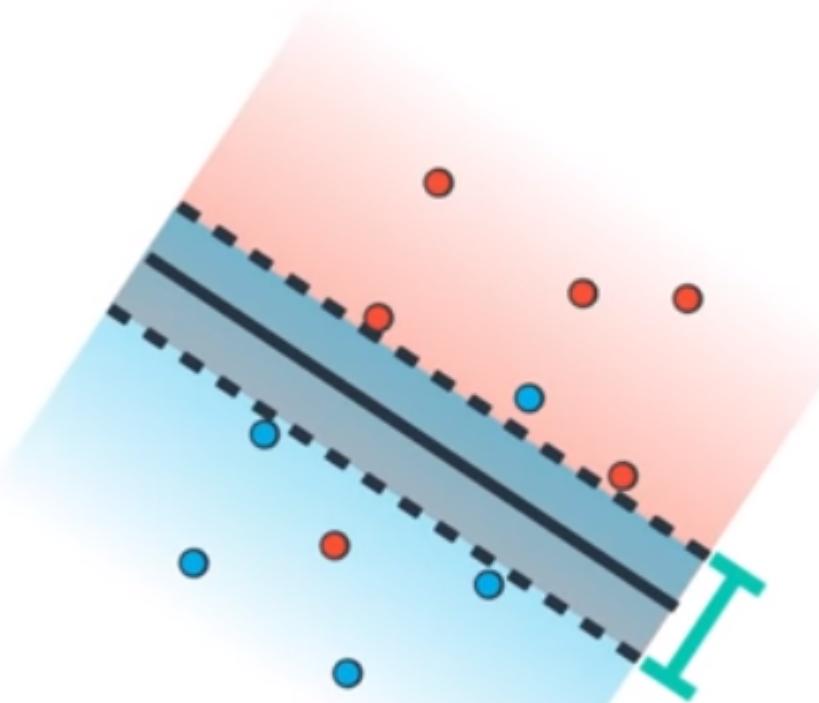


Classification Error



Margin Error

# SVM Error



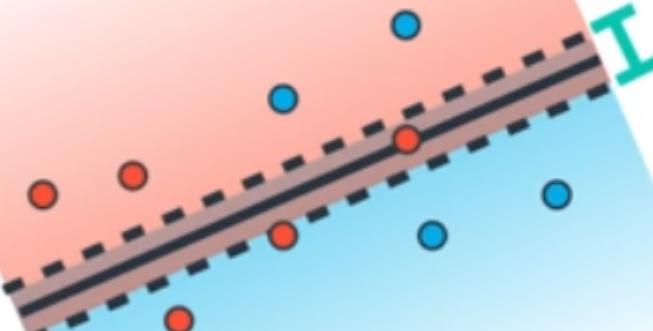
Error

# Gradient Descent

Same as the SVM trick!

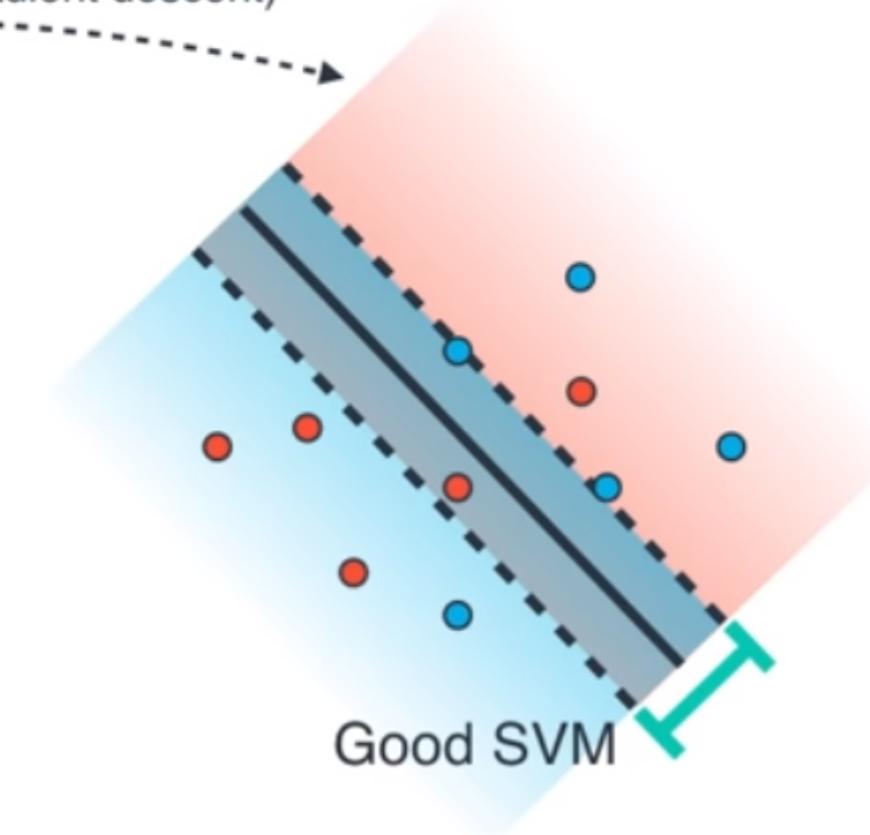
Minimize using calculus (gradient descent)

Large error

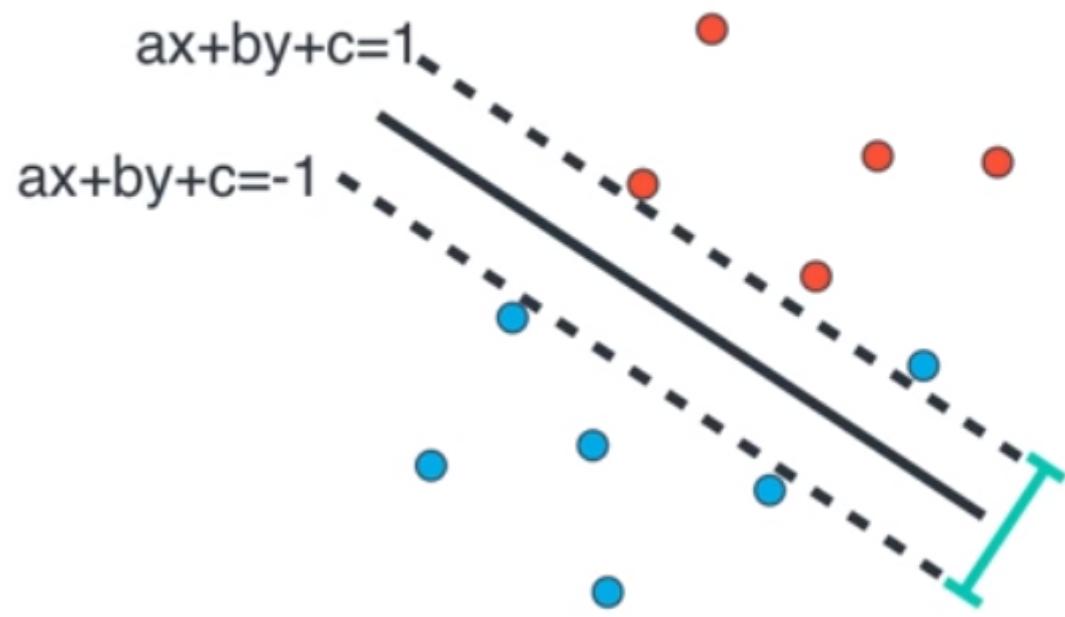


Bad SVM

Good SVM



# Challenge - Gradient Descent



$$\text{Margin error} = a^2 + b^2$$

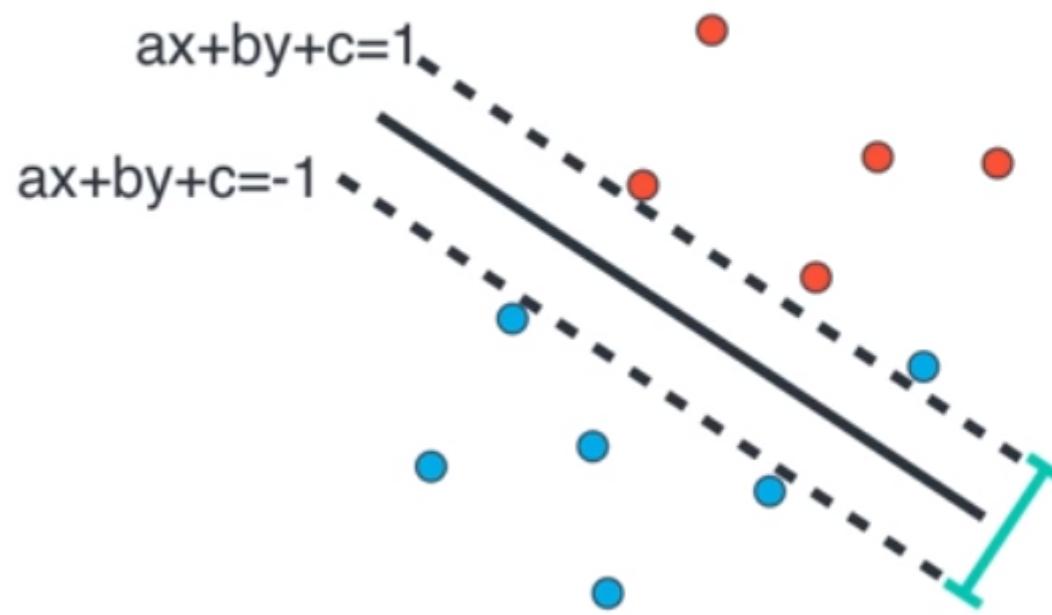
$$d\text{Error}/da = 2a$$

$$d\text{Error}/db = 2b$$

$$a \rightarrow a - \eta \cdot 2a = a(1 - 2\eta)$$

$$b \rightarrow b - \eta \cdot 2b = b(1 - 2\eta)$$

# Challenge - Gradient Descent



$$\text{Margin error} = a^2 + b^2$$

$$d\text{Error}/da = 2a$$

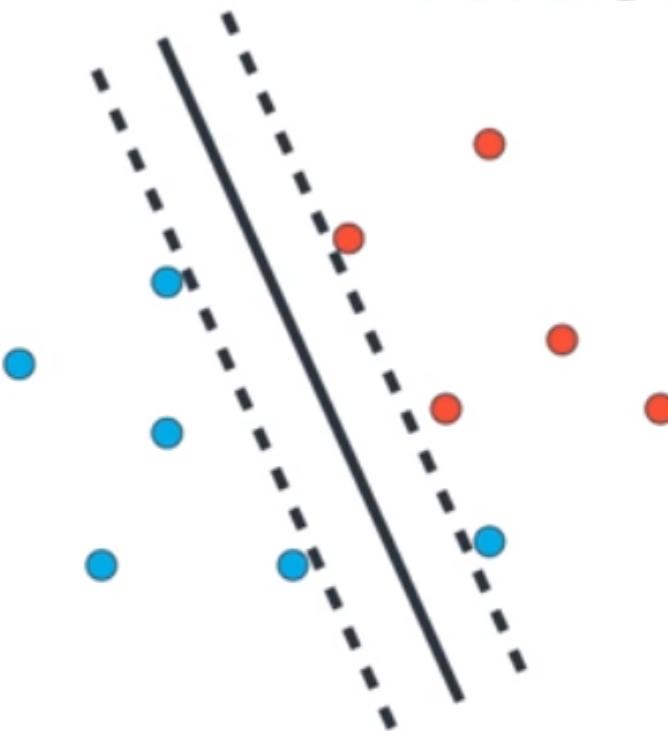
$$d\text{Error}/db = 2b$$

expanding factor!

$$a \longrightarrow a - \eta \quad 2a = a(1 - 2\eta)$$

$$b \longrightarrow b - \eta \quad 2b = b(1 - 2\eta)$$

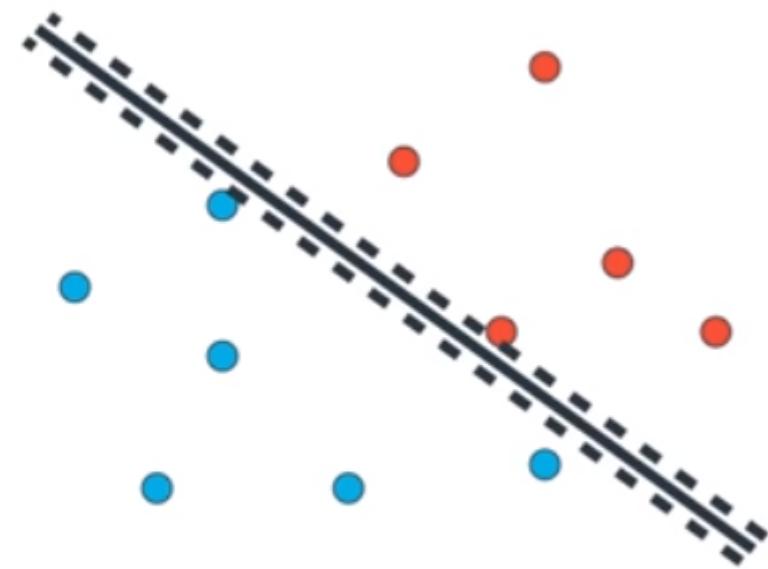
# Which line is better?



Classification  
Error

+

Margin  
Error

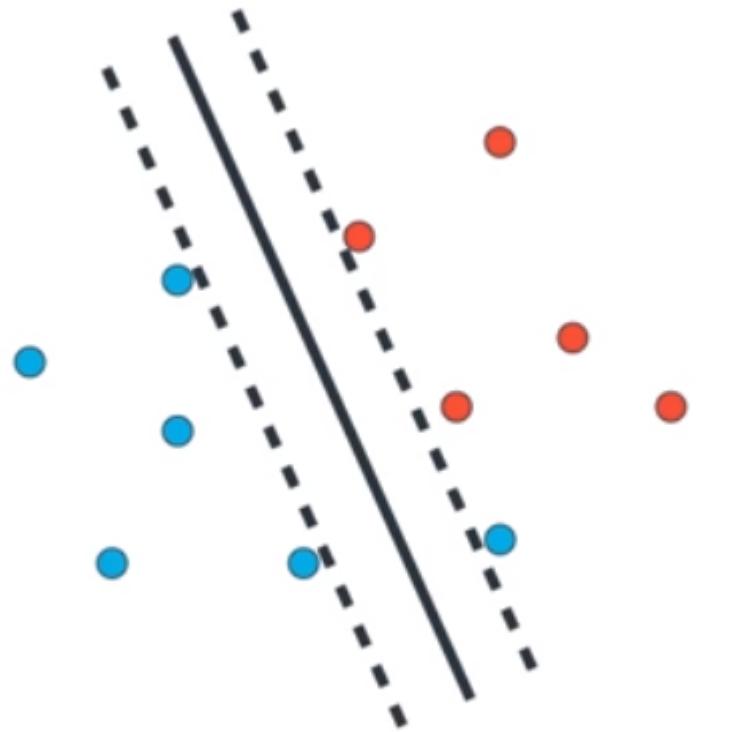


Classification  
Error

+

Margin  
Error

# The C parameter



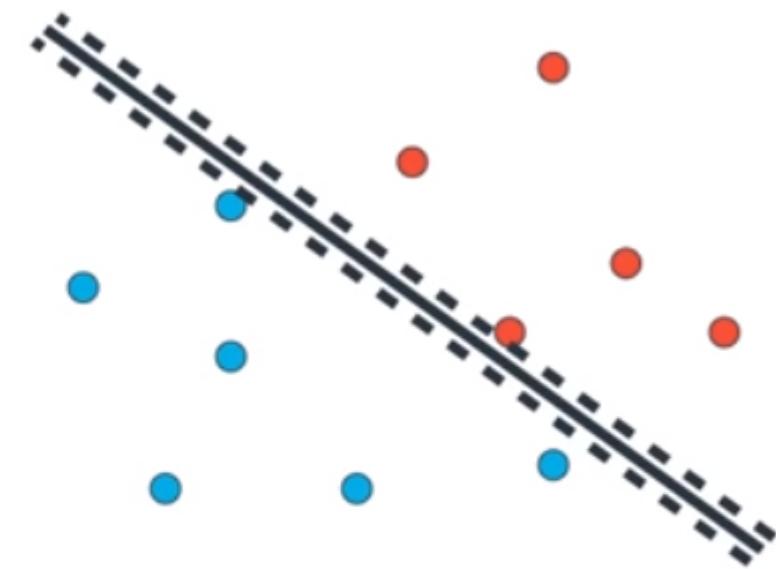
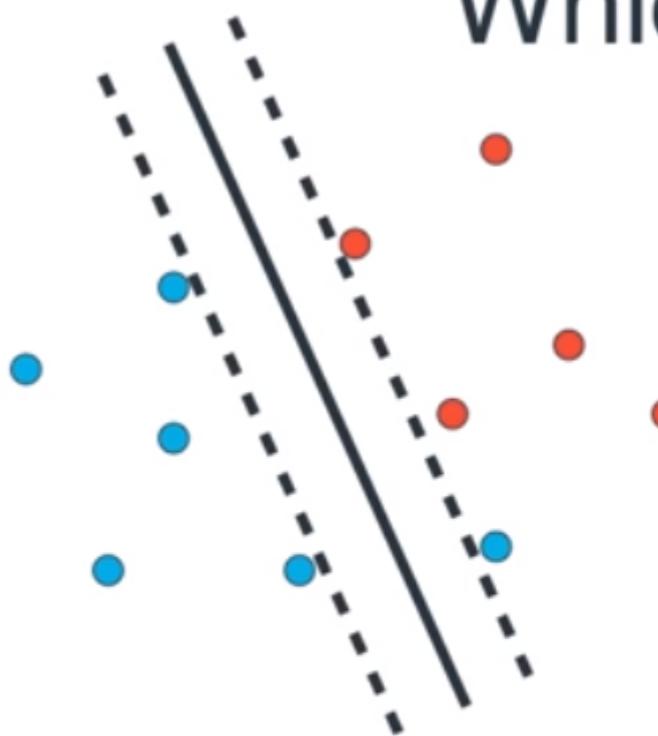
C

Classification  
Error

+

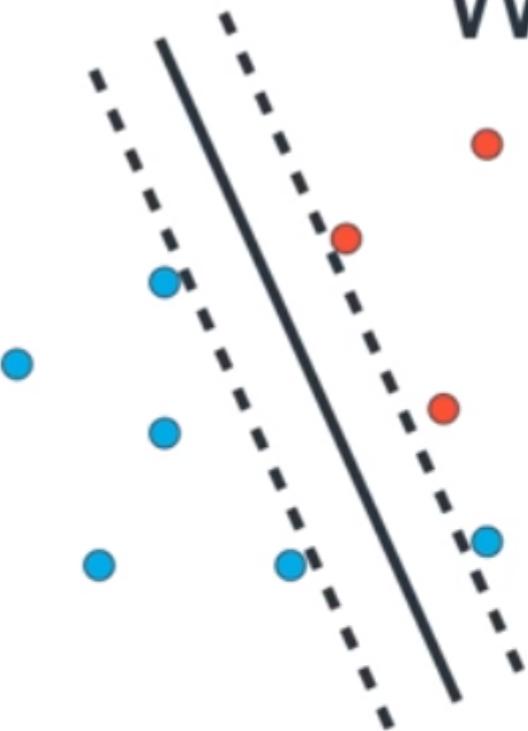
Margin  
Error

# Which line is better?

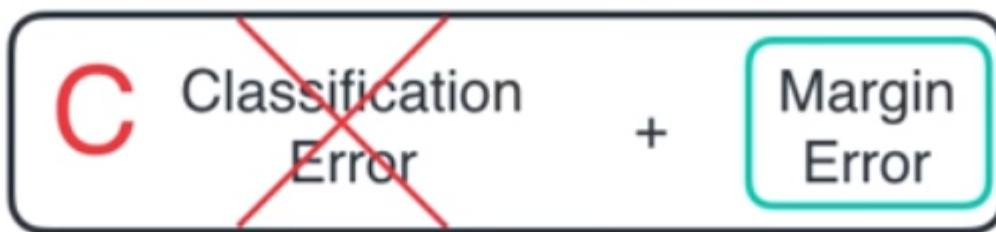
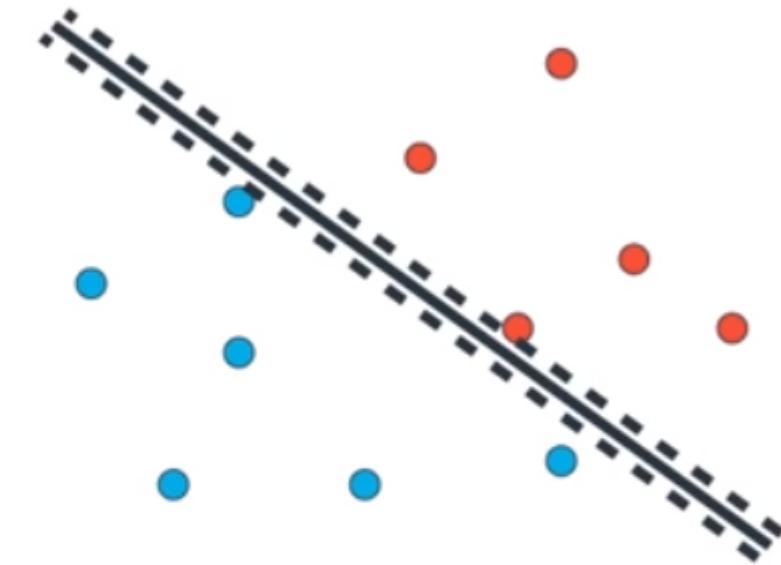


**C** Classification  
Error + Margin  
Error

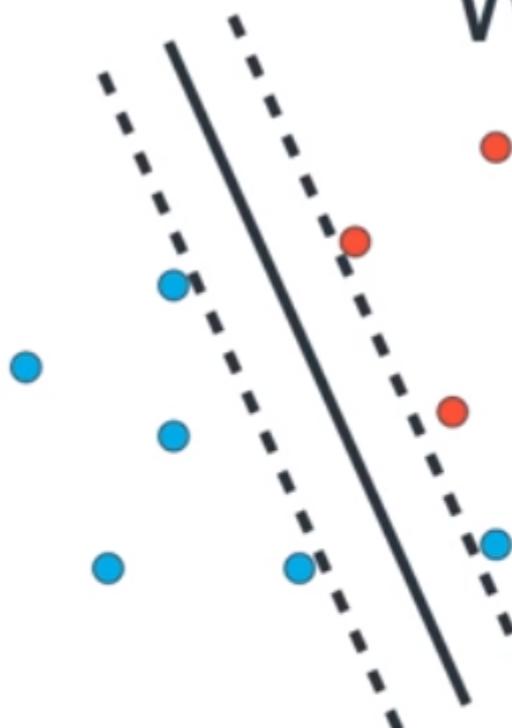
# Which line is better?



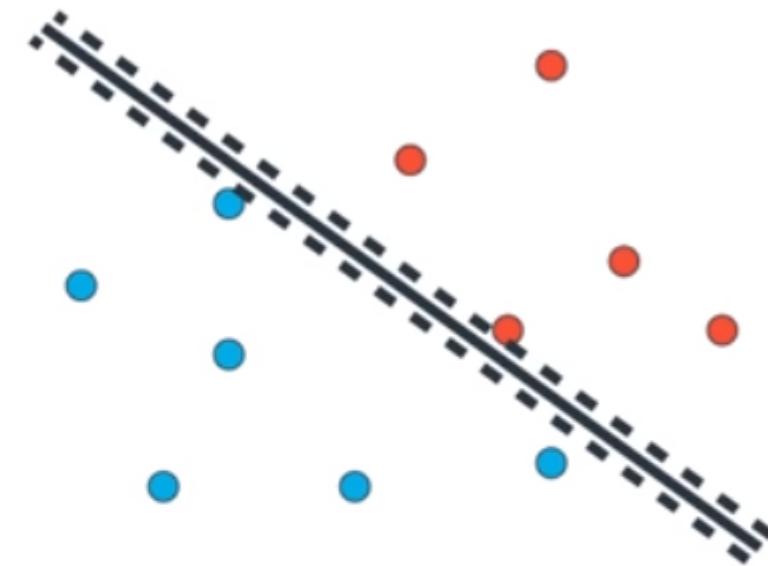
Small C  
Focus on margin



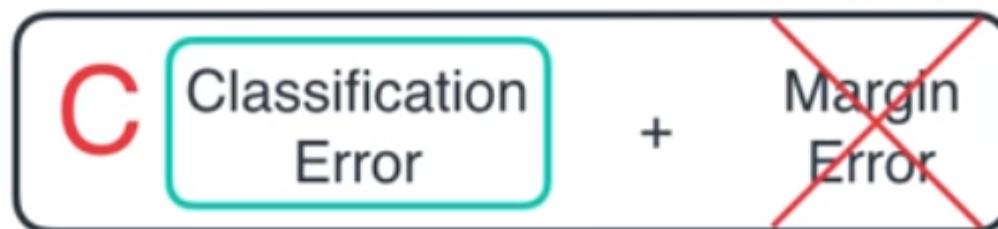
# Which line is better?



Small C  
Focus on margin



Large C  
Focus on classification





# Discussion

---



Thank you!

---