

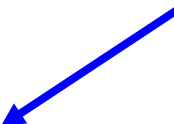
# Structured Support Vector Machine

Hung-yi Lee

# Structured Learning

- We need a more powerful function  $f$ 
  - Input and output are both objects with structures
  - *Object*: sequence, list, tree, bounding box ...

$$f : X \rightarrow Y$$



$X$  is the space of  
one kind of object



$Y$  is the space of  
another kind of object

# Unified Framework

## Step 1: Training

- Find a function  $F$

$$F: X \times Y \rightarrow \mathbb{R}$$

- $F(x,y)$ : evaluate how compatible the objects  $x$  and  $y$  is

## Step 2: Inference (Testing)

- Given an object  $x$

$$\tilde{y} = \arg \max_{y \in Y} F(x, y)$$

# Three Problems

## Problem 1: Evaluation

- What does  $F(x,y)$  look like?

## Problem 2: Inference

- How to solve the “arg max” problem

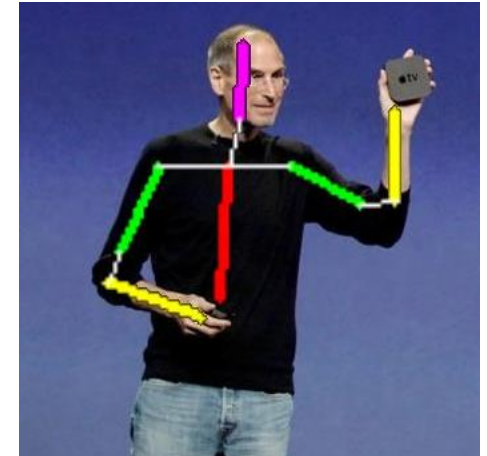
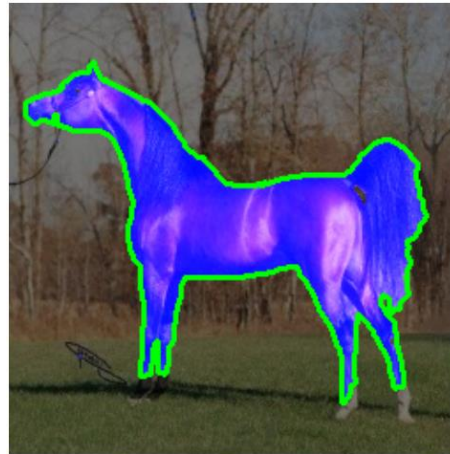
$$y = \arg \max_{y \in Y} F(x, y)$$

## Problem 3: Training

- Given training data, how to find  $F(x,y)$

# Example Task: Object Detection

Example Task



Keep in mind that what you will learn today can be applied to other tasks.

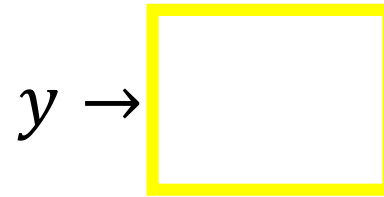
Source of image:

<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.295.6007&rep=rep1&type=pdf>

<http://www.vision.ee.ethz.ch/~hpedemo/gallery.php>

# Problem 1: Evaluation

- $F(x,y)$  is linear

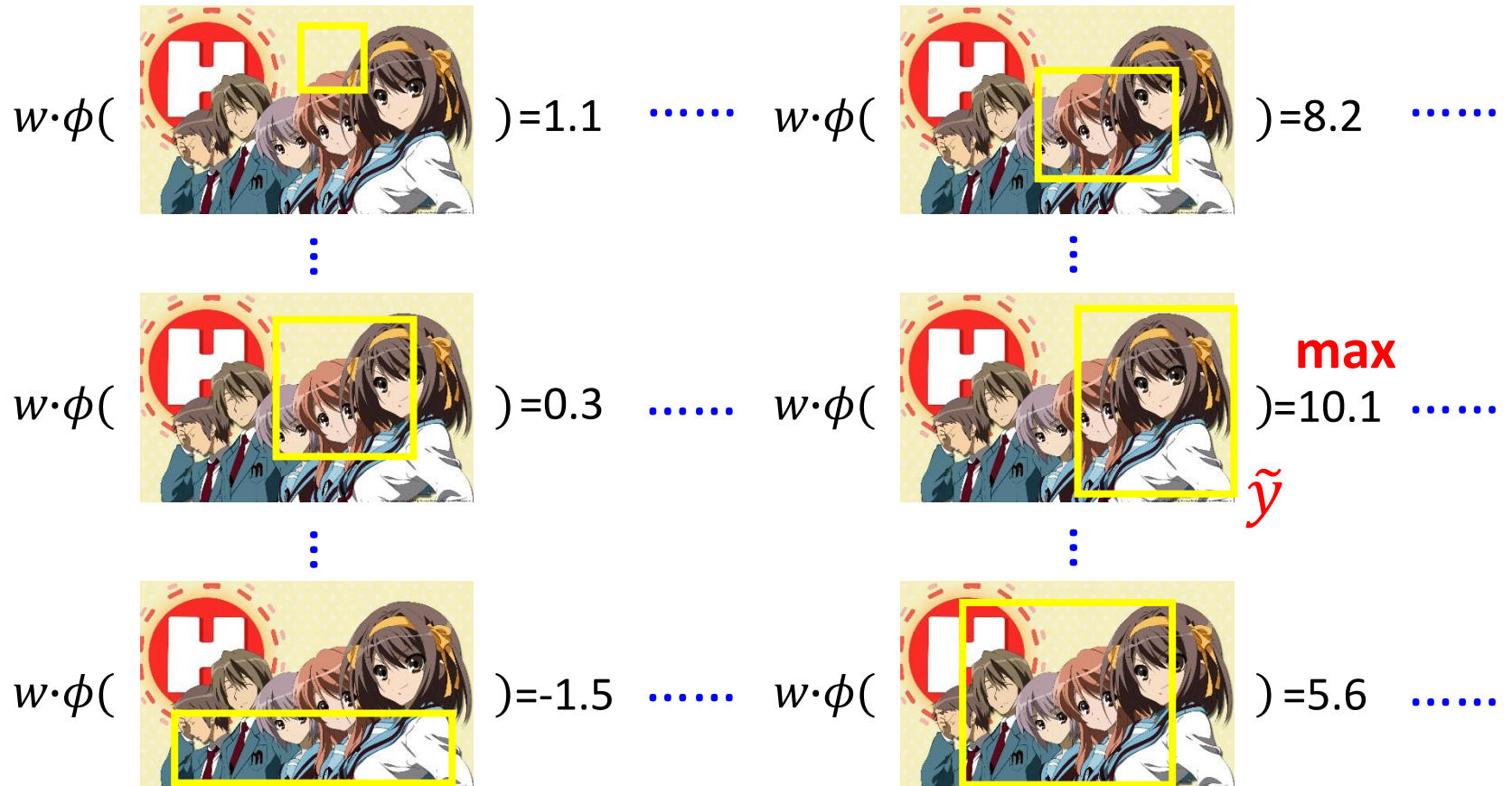


$$F(\text{image with yellow box}) = w \cdot \phi(\text{image with yellow box})$$

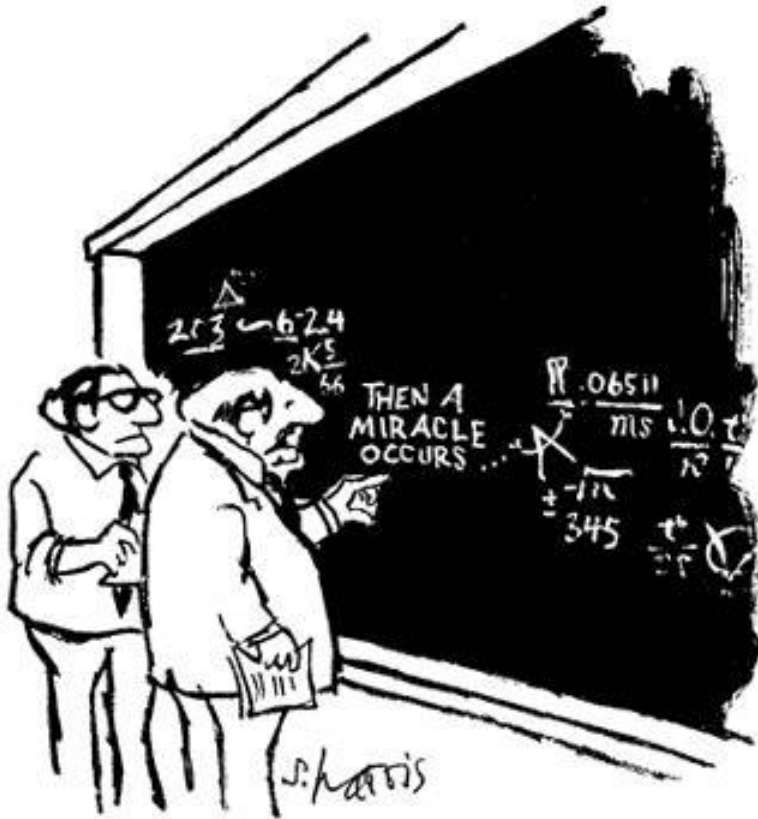
Open question: What if  $F(x,y)$  is not linear?

# Problem 2: Inference

$$\tilde{y} = \arg \max_{y \in \mathbb{Y}} w \cdot \phi(x, y)$$



# Problem 2: Inference



"I think you should be more explicit here in step two."

- Object Detection
  - Branch and Bound algorithm
  - Selective Search
- Sequence Labeling
  - Viterbi Algorithm
- The algorithms can depend on  $\phi(x, y)$
- Genetic Algorithm
- Open question:
  - What happens if the inference is non exact?

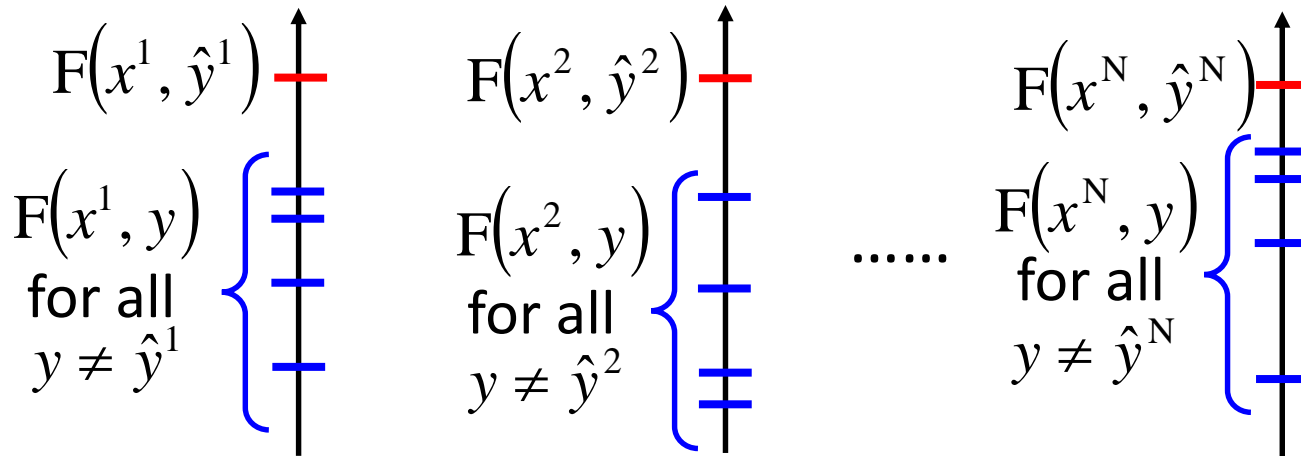


# Problem 3: Training

## Principle

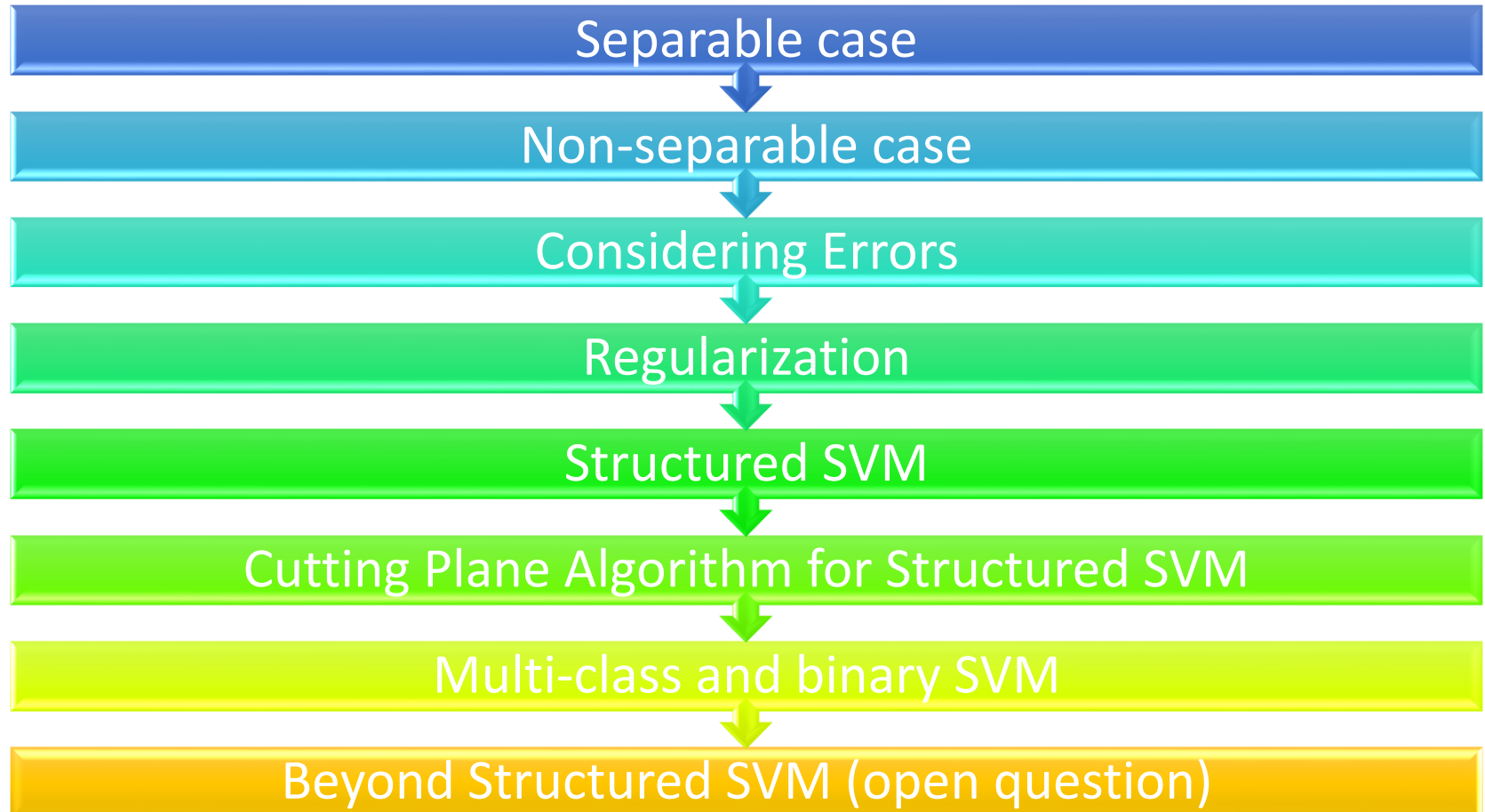
Training data:  $\{(x^1, \hat{y}^1), (x^2, \hat{y}^2), \dots, (x^N, \hat{y}^N)\}$

We should find  $F(x, y)$  such that .....

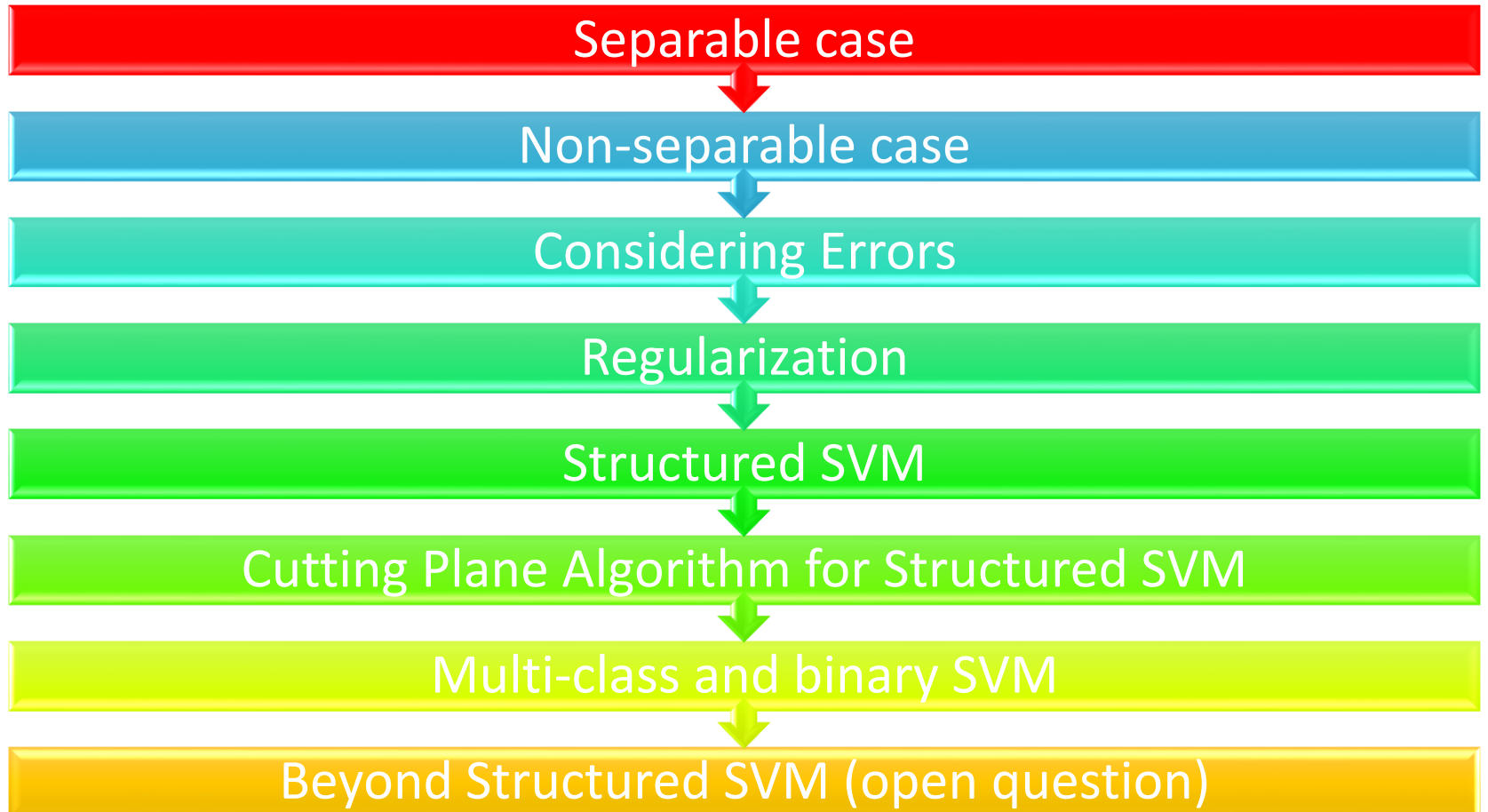


Let's ignore problems 1 and 2 and only focus on problem 3 today.

# Outline



# Outline

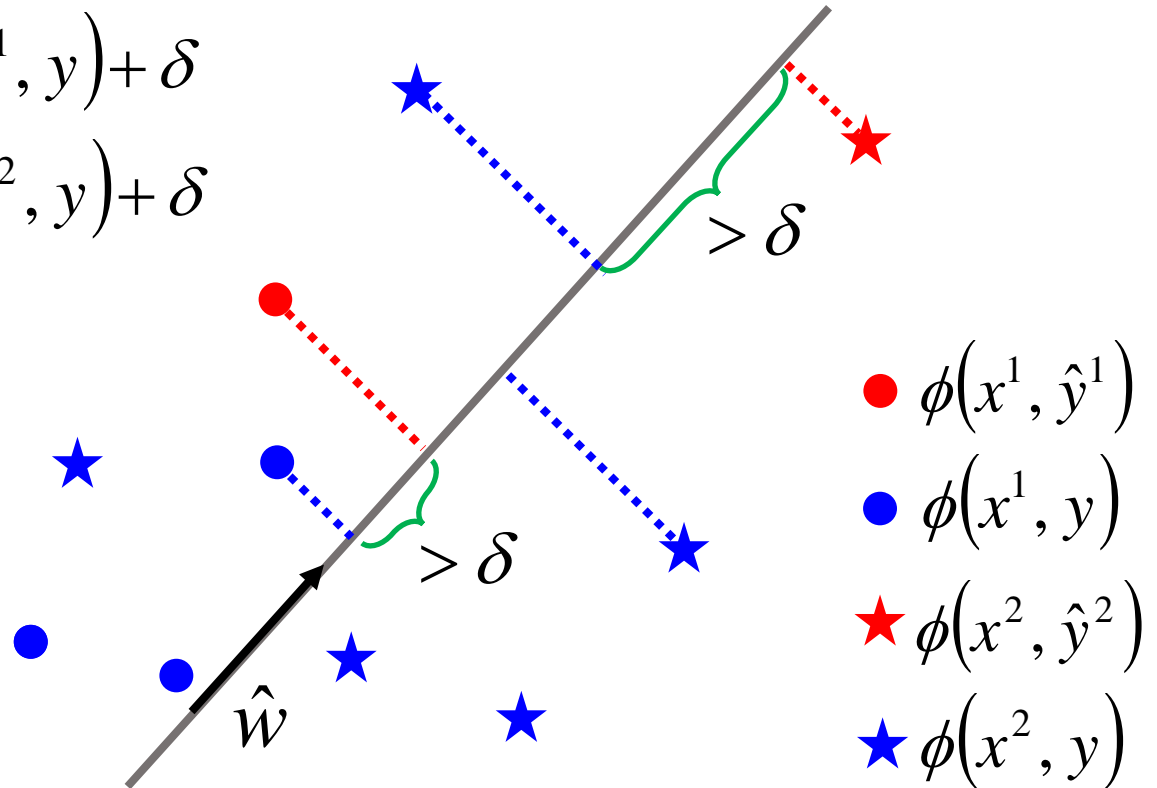


# Assumption: Separable


- There exists a weight vector  $\hat{w}$

$$\hat{w} \cdot \phi(x^1, \hat{y}^1) \geq \hat{w} \cdot \phi(x^1, y) + \delta$$

$$\hat{w} \cdot \phi(x^2, \hat{y}^2) \geq \hat{w} \cdot \phi(x^2, y) + \delta$$



# Structured Perceptron

- **Input**: training data set  $\{(x^1, \hat{y}^1), (x^2, \hat{y}^2), \dots, (x^N, \hat{y}^N)\}$
- **Output**: weight vector  $w$
- **Algorithm**: Initialize  $w = 0$ 
  - do
    - For each pair of training example  $(x^n, \hat{y}^n)$ 
      - Find the label  $\tilde{y}^n$  maximizing  $w \cdot \phi(x^n, y)$ 
$$\tilde{y}^n = \arg \max_{y \in Y} w \cdot \phi(x^n, y) \text{ (problem 2)}$$
      - If  $\tilde{y}^n \neq \hat{y}^n$ , update  $w$ 
$$w \rightarrow w + \phi(x^n, \hat{y}^n) - \phi(x^n, \tilde{y}^n)$$
  - until  $w$  is not updated  We are done!

# Warning of Math

In separable case, to obtain a  $\hat{w}$ , you only have to update at most  $(R/\delta)^2$  times

$\delta$ : margin

$R$ : the largest distance between  $\phi(x, y)$  and  $\phi(x, y')$

Not related to the space of  $y$ !

# Proof of Termination

w is updated **once it sees a mistake**

$$w^0 = 0 \rightarrow w^1 \rightarrow w^2 \rightarrow \dots \rightarrow w^k \rightarrow w^{k+1} \rightarrow \dots$$

$$w^k = w^{k-1} + \phi(x^n, \hat{y}^n) - \phi(x^n, \tilde{y}^n) \text{ (the relation of } w^k \text{ and } w^{k-1})$$

**Remind**: we are considering the separable case

Assume there exists a weight vector  $\hat{w}$  such that

$\forall n$  (All training examples)

$\forall y \in Y - \{\hat{y}^n\}$  (All incorrect label for an example)

$$\hat{w} \cdot \phi(x^n, \hat{y}^n) \geq \hat{w} \cdot \phi(x^n, y) + \delta$$

Assume  $\|\hat{w}\| = 1$  without loss of generality

# Proof of Termination

w is updated **once it sees a mistake**

$$w^0 = 0 \rightarrow w^1 \rightarrow w^2 \rightarrow \dots \rightarrow w^k \rightarrow w^{k+1} \rightarrow \dots$$

$$w^k = w^{k-1} + \phi(x^n, \hat{y}^n) - \phi(x^n, \tilde{y}^n) \text{ (the relation of } w^k \text{ and } w^{k-1})$$

Proof that: The angle  $\rho_k$  between  $\hat{w}$  and  $w^k$  is smaller as k increases

Analysis  $\cos \rho_k$  (larger and larger?)  $\cos \rho_k = \frac{\hat{w} \cdot w^k}{\|\hat{w}\| \cdot \|w^k\|}$

$$\begin{aligned} \hat{w} \cdot w^k &= \hat{w} \cdot (w^{k-1} + \phi(x^n, \hat{y}^n) - \phi(x^n, \tilde{y}^n)) \\ &= \hat{w} \cdot w^{k-1} + \underbrace{\hat{w} \cdot \phi(x^n, \hat{y}^n) - \hat{w} \cdot \phi(x^n, \tilde{y}^n)}_{\geq \delta \text{ (Separable)}} \geq \hat{w} \cdot w^{k-1} + \delta \end{aligned}$$



# Proof of Termination

w is updated **once it sees a mistake**

$$w^0 = 0 \rightarrow w^1 \rightarrow w^2 \rightarrow \dots \rightarrow w^k \rightarrow w^{k+1} \rightarrow \dots$$

$$w^k = w^{k-1} + \phi(x^n, \hat{y}^n) - \phi(x^n, \tilde{y}^n) \quad (\text{the relation of } w^k \text{ and } w^{k-1})$$

Proof that: The angle  $\rho_k$  between  $\hat{w}$  and  $w^k$  is smaller as k increases

Analysis  $\cos \rho_k$  (larger and larger?)  $\cos \rho_k = \frac{\hat{w} \cdot w^k}{\|\hat{w}\| \cdot \|w^k\|}$

$$\hat{w} \cdot w^k \geq \hat{w} \cdot w^{k-1} + \delta$$

$$\left. \begin{array}{ll} \hat{w} \cdot w^1 \geq \hat{w} \cdot w^0 + \delta & \hat{w} \cdot w^2 \geq \hat{w} \cdot w^1 + \delta \dots \dots \\ \hat{w} \cdot w^1 \geq \delta & \hat{w} \cdot w^2 \geq 2\delta \dots \dots \end{array} \right\} \hat{w} \cdot w^k \geq k\delta$$

(so what)

# Proof of Termination

$$\cos \rho_k = \frac{\hat{w}}{\|\hat{w}\|} \cdot \boxed{\|w^k\|} \quad w^k = w^{k-1} + \phi(x^n, \hat{y}^n) - \phi(x^n, \tilde{y}^n)$$

$$\begin{aligned} \|w^k\|^2 &= \|w^{k-1} + \phi(x^n, \hat{y}^n) - \phi(x^n, \tilde{y}^n)\|^2 \\ &= \|w^{k-1}\|^2 + \underbrace{\|\phi(x^n, \hat{y}^n) - \phi(x^n, \tilde{y}^n)\|^2}_{> 0} + \underbrace{2w^{k-1} \cdot (\phi(x^n, \hat{y}^n) - \phi(x^n, \tilde{y}^n))}_{? < 0 \text{ (mistake)}} \end{aligned}$$

Assume the distance  
between any two feature  
vectors is smaller than R

$$\leq \|w^{k-1}\|^2 + R^2$$

$$\begin{aligned} \|w^1\|^2 &\leq \|w^0\|^2 + R^2 = R^2 \\ \|w^2\|^2 &\leq \|w^1\|^2 + R^2 \leq 2R^2 \\ &\dots \\ \|w^k\|^2 &\leq kR^2 \end{aligned}$$

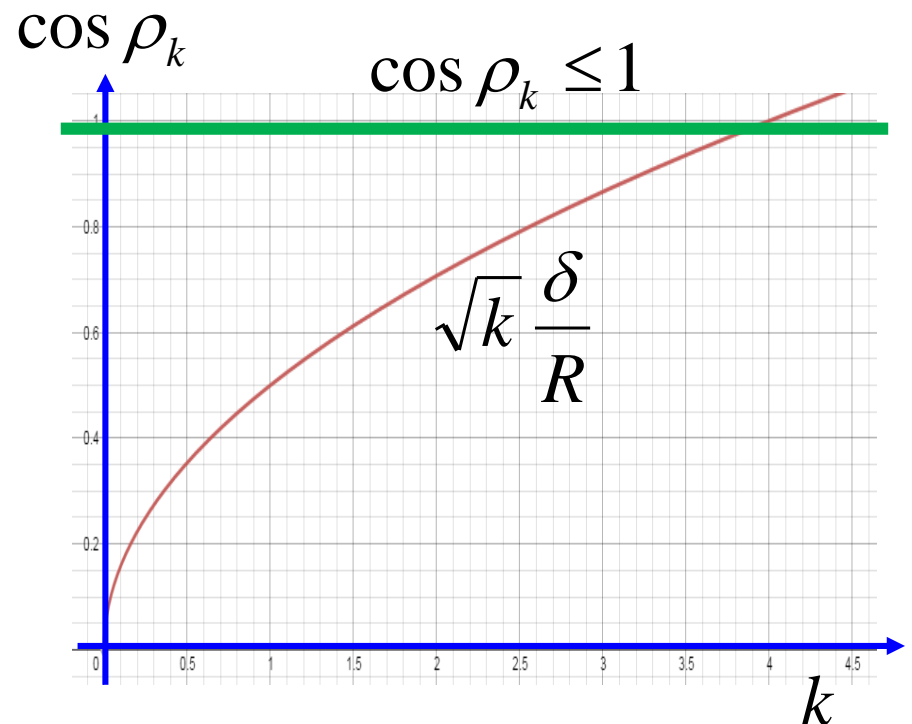
# Proof of Termination

$$\cos \rho_k = \frac{\hat{w}}{\|\hat{w}\|} \cdot \frac{w^k}{\|w^k\|} \quad \hat{w} \cdot w^k \geq k\delta \quad \|w^k\|^2 \leq kR^2$$

$$\geq \frac{k\delta}{\sqrt{kR^2}} = \sqrt{k} \frac{\delta}{R}$$

$$\sqrt{k} \frac{\delta}{R} \leq 1$$

$$k \leq \left(\frac{R}{\delta}\right)^2$$



# End of Warning

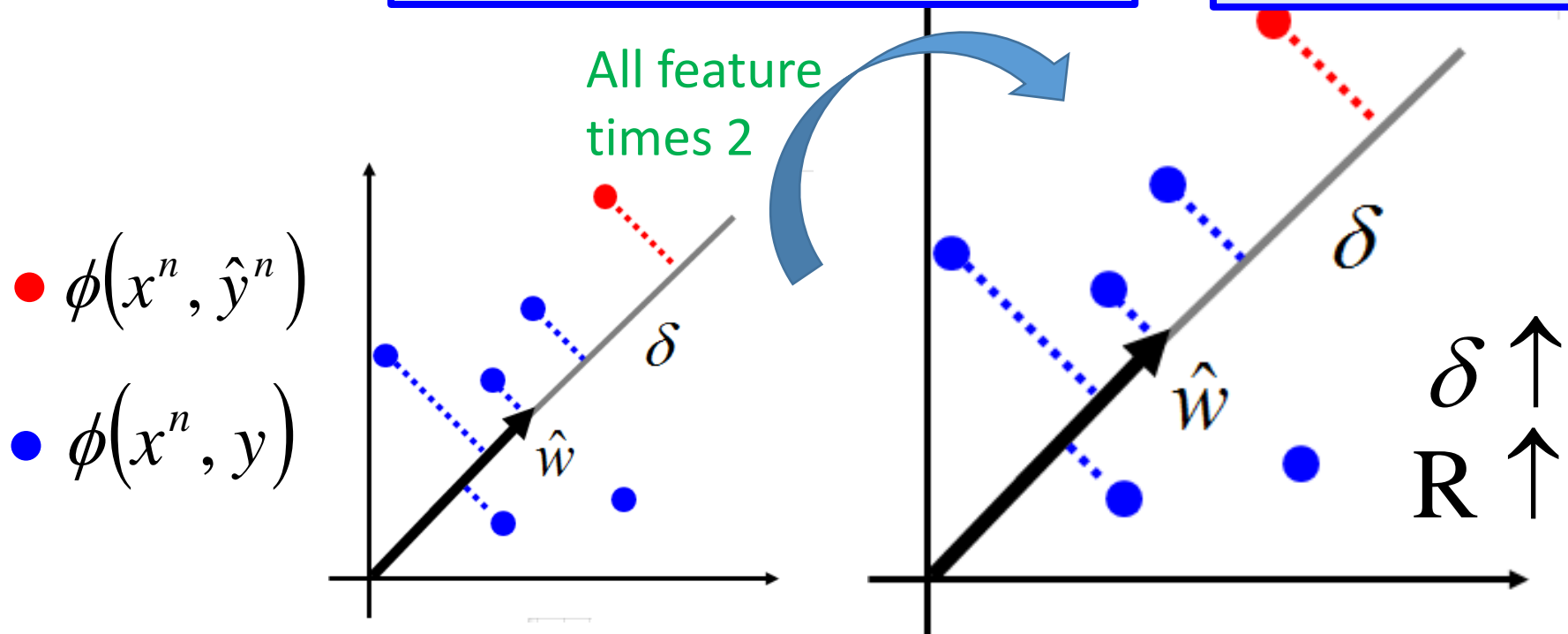
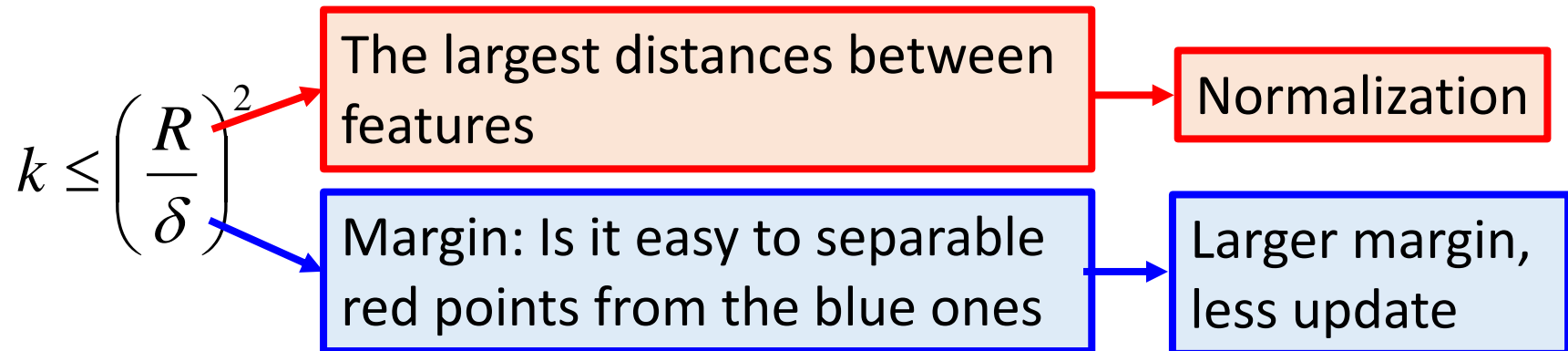
In separable case, to obtain a  $\hat{w}$ , you only have to update at most  $(R/\delta)^2$  times

$\delta$ : margin

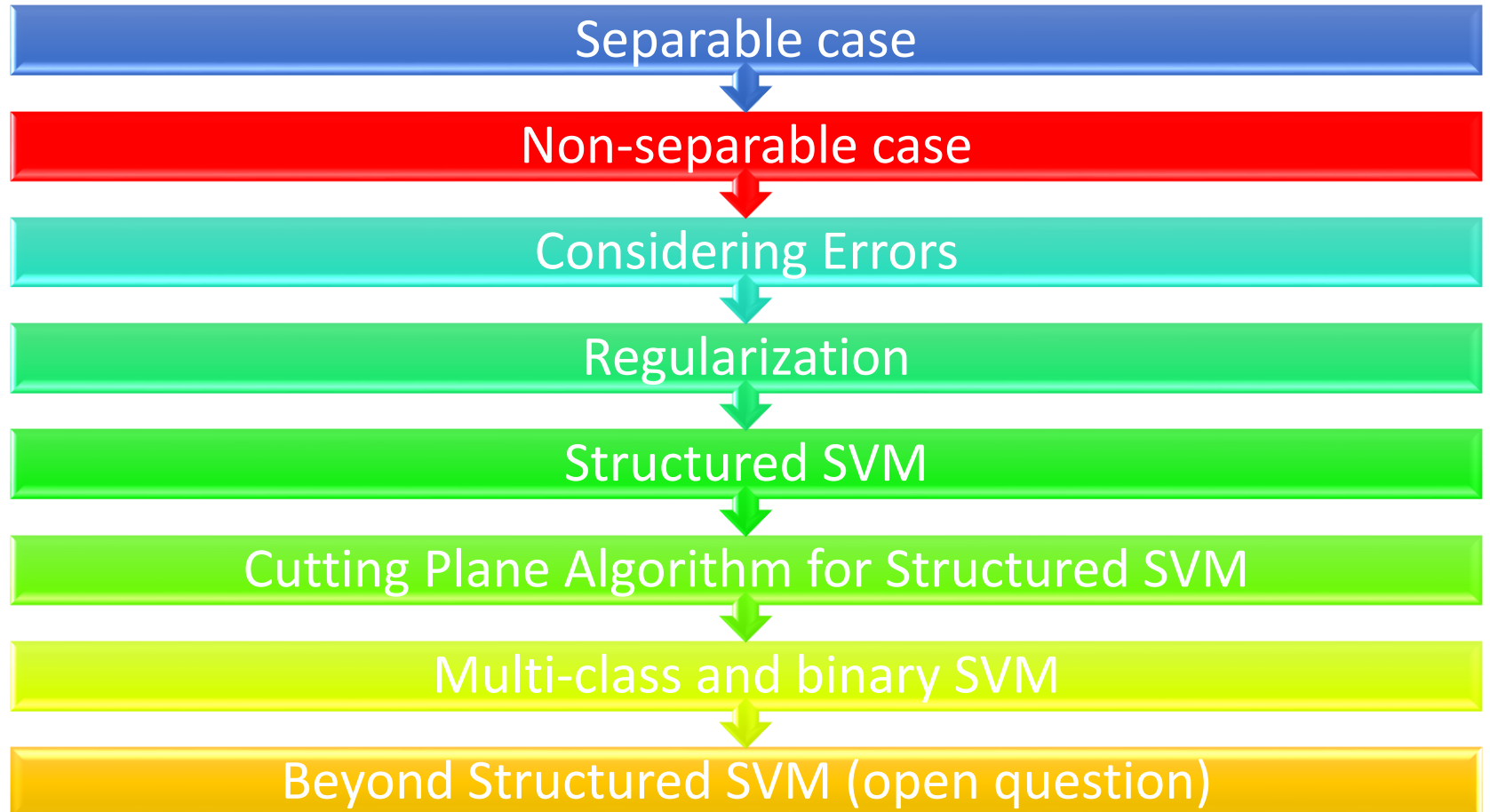
$R$ : the largest distance between  $\phi(x, y)$  and  $\phi(x, y')$

Not related to the space of  $y$ !

# How to make training fast?



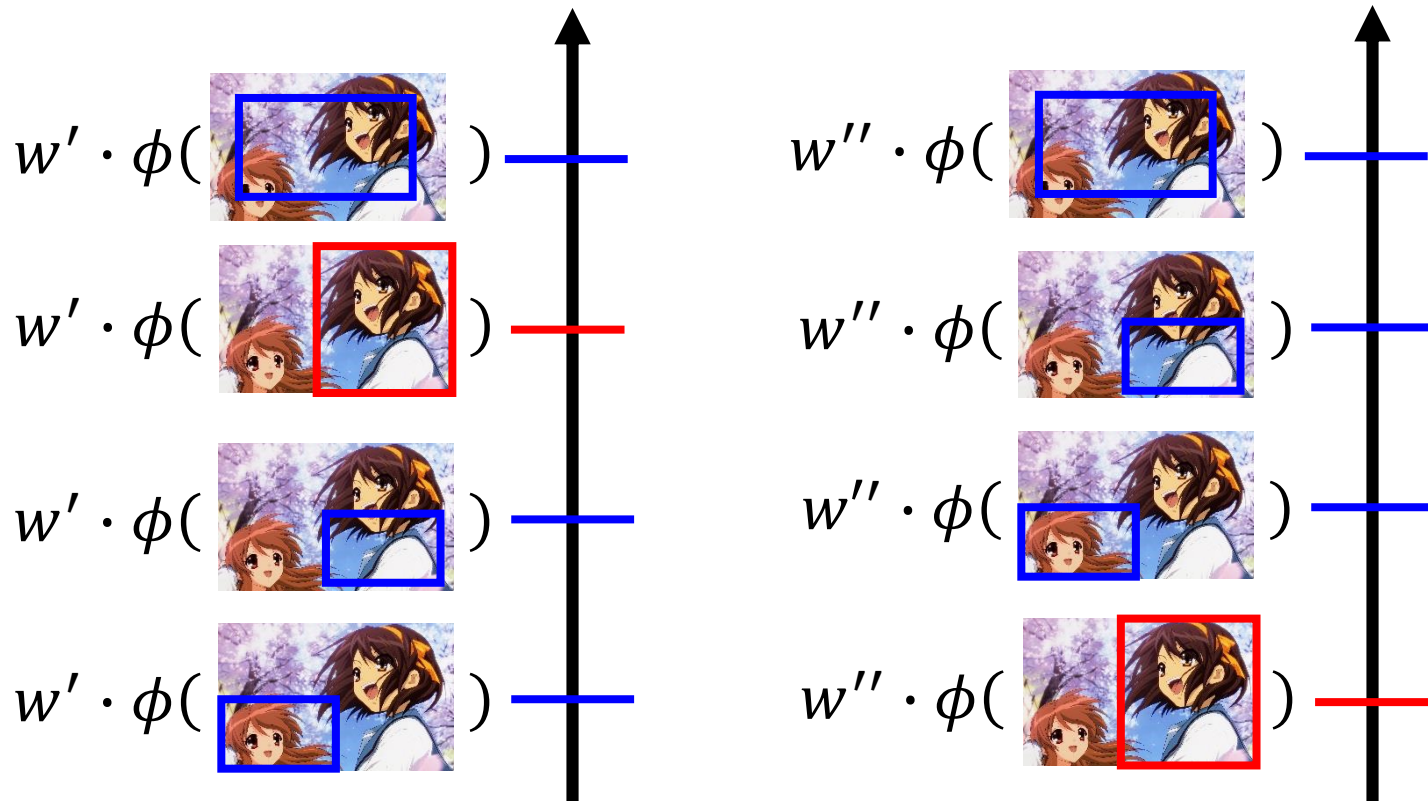
# Outline



# Non-separable Case

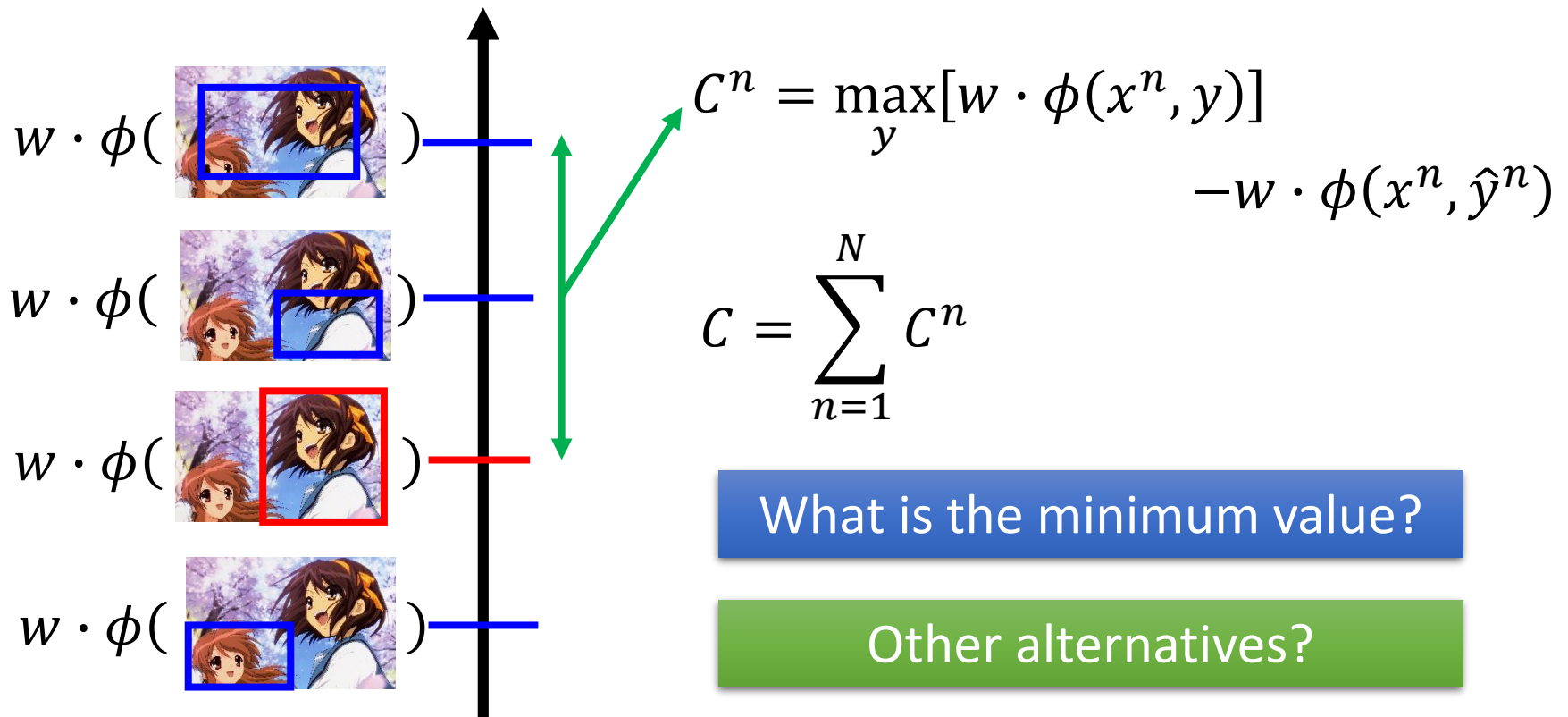
Undoubtedly,  $w'$  is better than  $w''$ .

- When the data is non-separable, some weights are still better than the others.



# Defining Cost Function

- Define a cost  $C$  to evaluate how bad a  $w$  is, and then pick the  $w$  minimizing the cost  $C$





# (Stochastic) Gradient Descent

Find  $w$  minimizing the cost  $\mathcal{C}$

$$\mathcal{C} = \sum_{n=1}^N \mathcal{C}^n$$

$$\mathcal{C}^n = \max_y [w \cdot \phi(x^n, y)] - w \cdot \phi(x^n, \hat{y}^n)$$

**(Stochastic) Gradient descent:**

We only have to know how to compute  $\nabla \mathcal{C}^n$ .

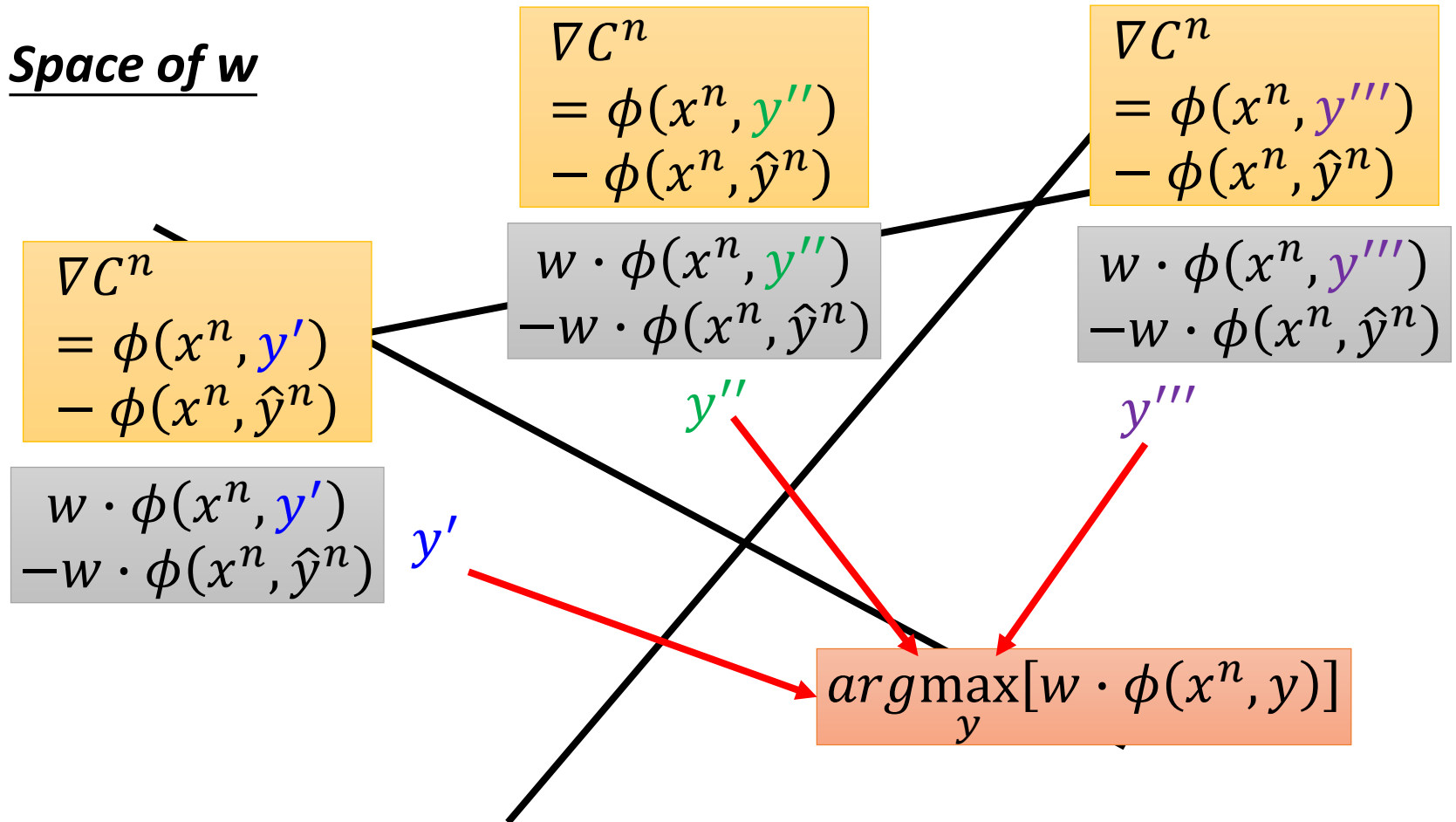
However, there is “max” in  $\mathcal{C}^n$  .....

$$C^n = \max_y [w \cdot \phi(x^n, y)] - w \cdot \phi(x^n, \hat{y}^n)$$

When  $w$  is different,  
the  $y$  can be different.

How to compute  $\nabla C^n$ ?


Space of  $w$




# (Stochastic) Gradient Descent

For  $t = 1$  to  $T$ :  Update the parameters  $T$  times

Randomly pick a training data  $\{x^n, \hat{y}^n\}$   stochastic

$\tilde{y}^n = \underset{y}{\operatorname{argmax}}[w \cdot \phi(x^n, y)]$   Locate the region

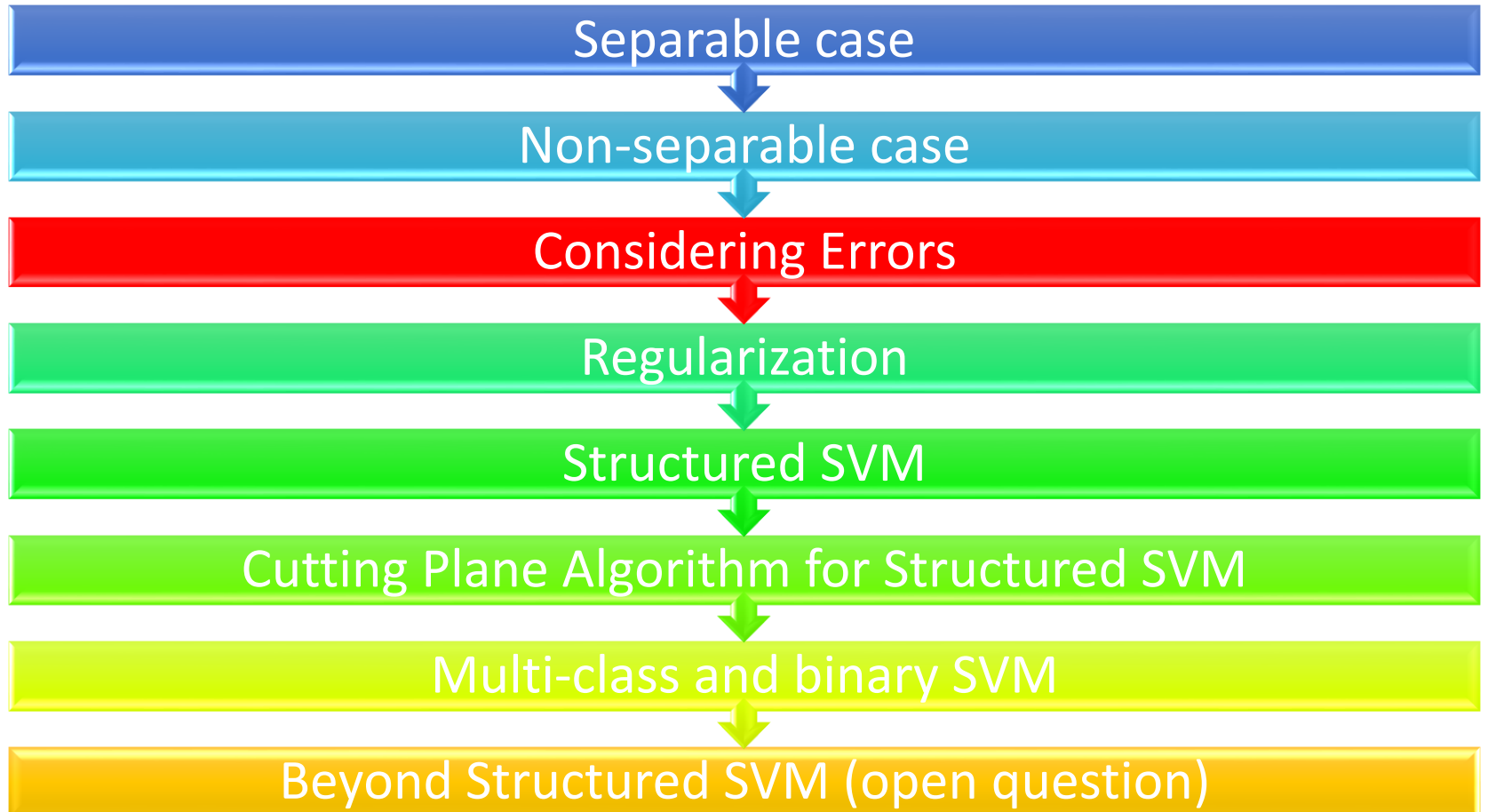
$\nabla C^n = \phi(x^n, \tilde{y}^n) - \phi(x^n, \hat{y}^n)$   simple

$$w \rightarrow w - \eta \nabla C^n$$

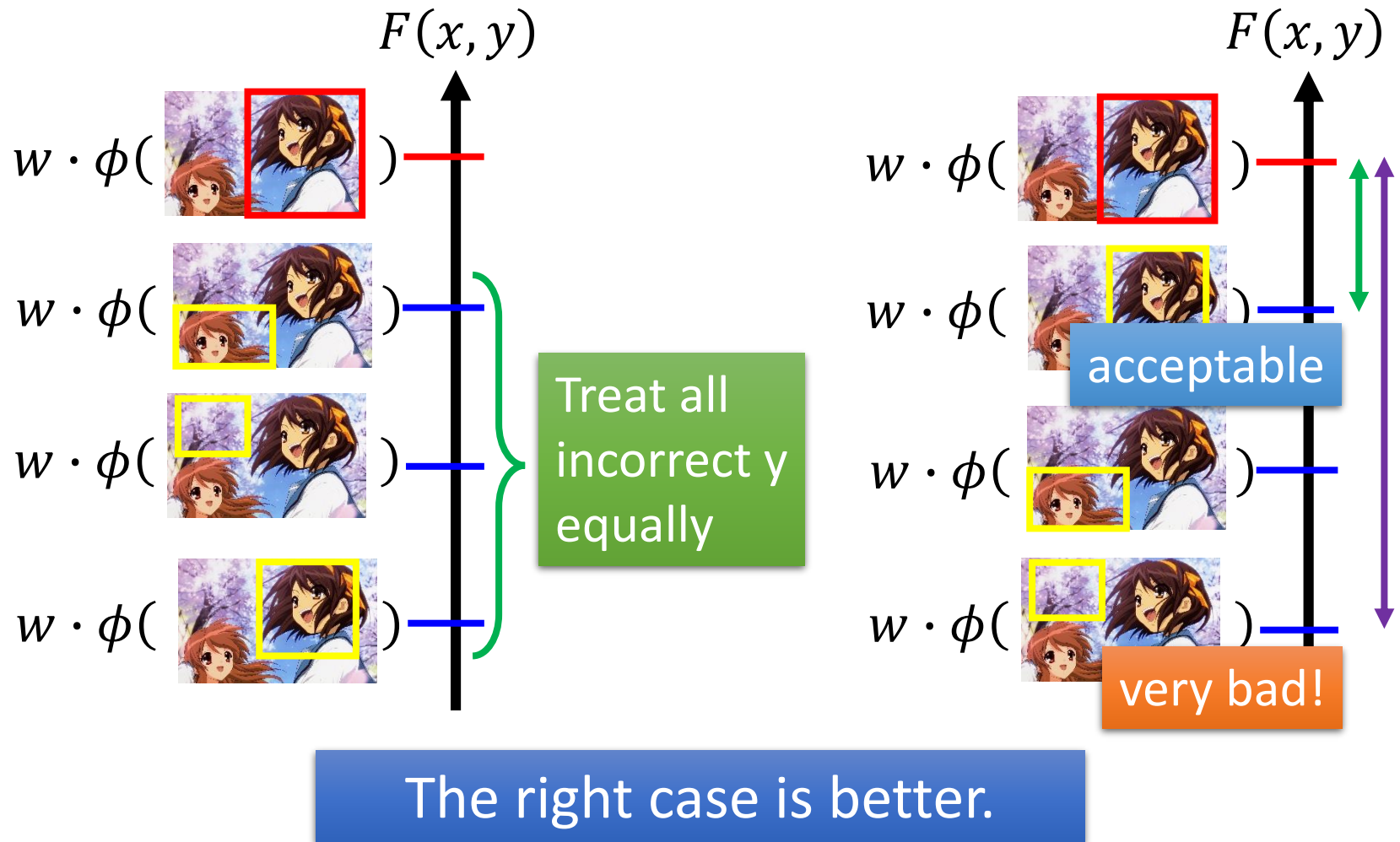
$$= w - \eta [\phi(x^n, \tilde{y}^n) - \phi(x^n, \hat{y}^n)]$$

If we set  $\eta = 1$ , then we are doing structured perceptron.

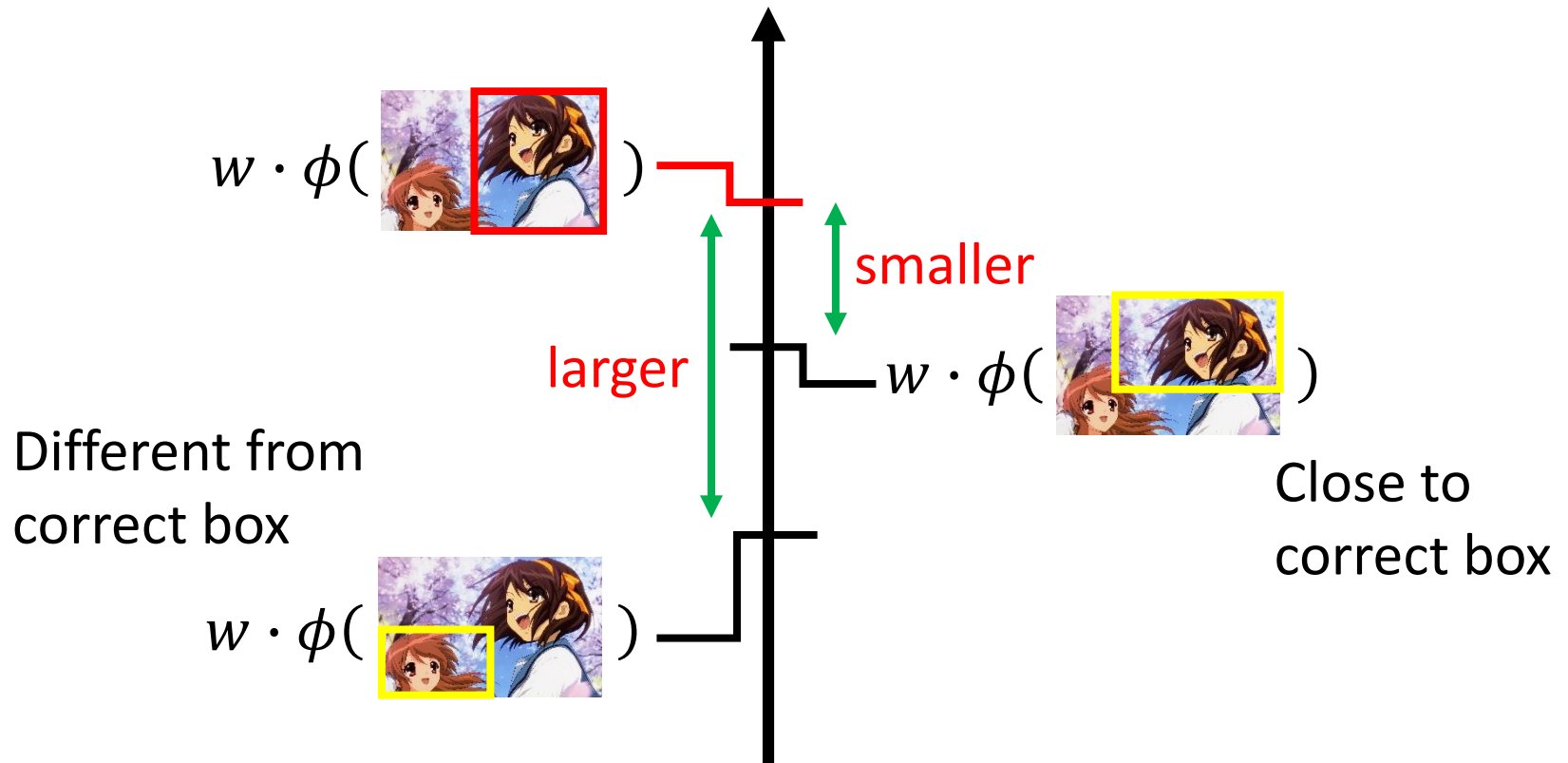
# Outline



Based on what we have considered .....



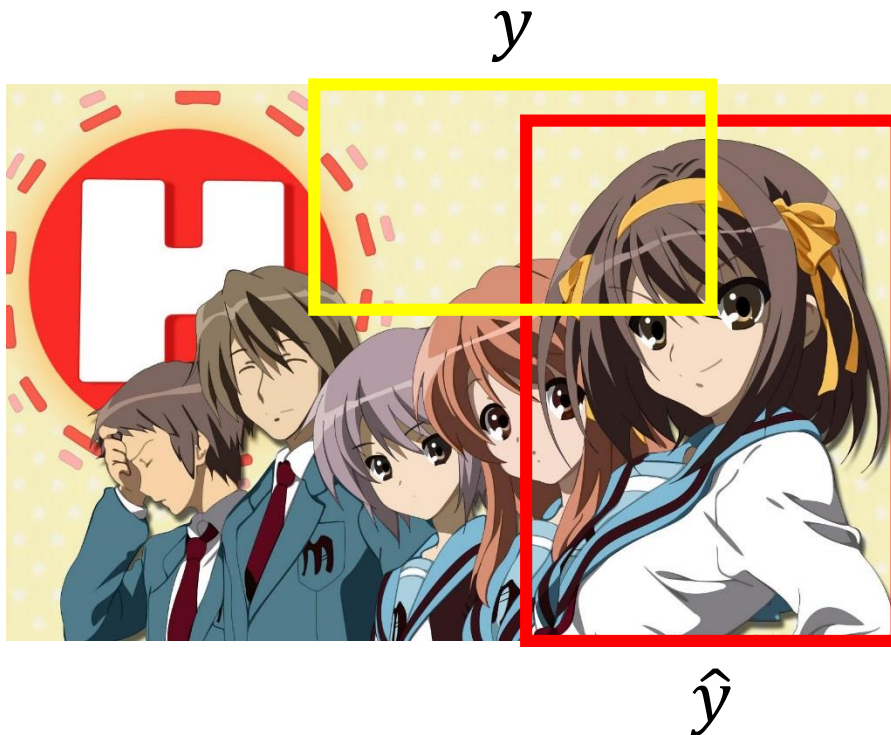
# Considering the incorrect ones



How to measure the difference

# Defining Error Function

- $\Delta(\hat{y}, y)$ : difference between  $\hat{y}$  and  $y$  ( $> 0$ )



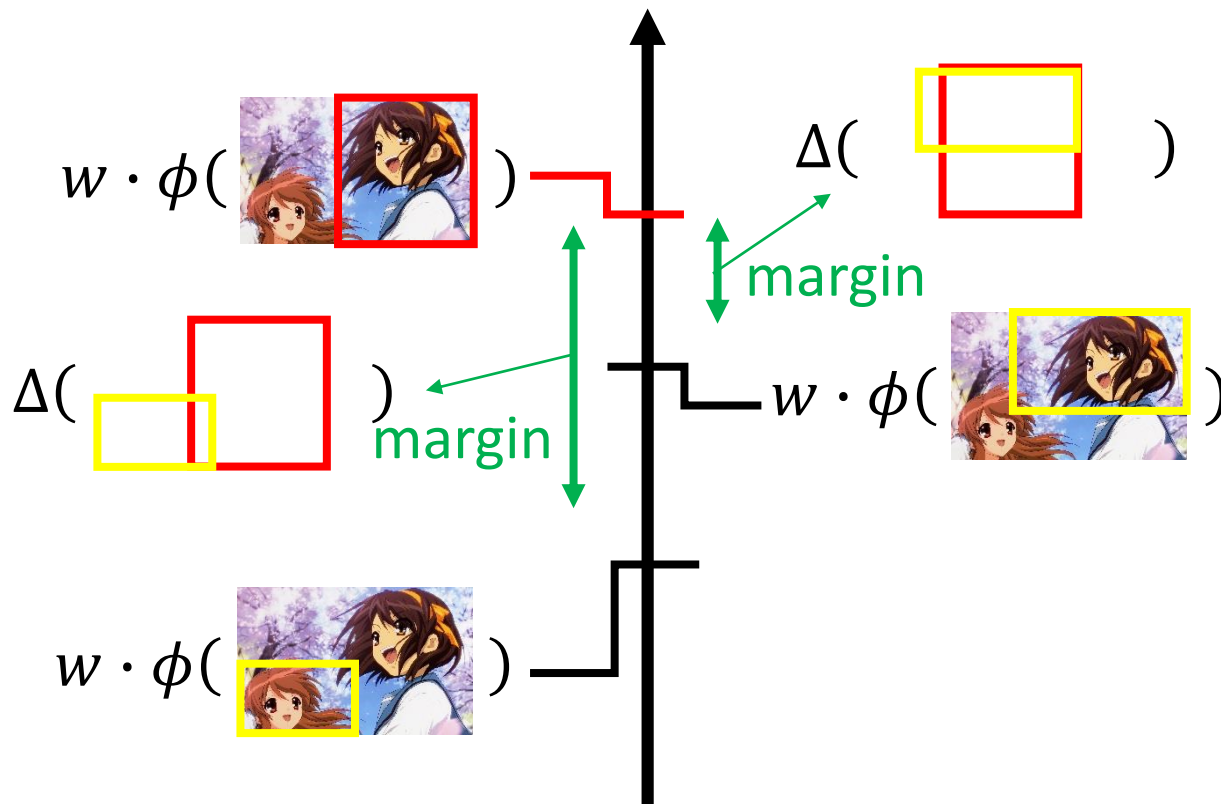
$A(y)$ : area of bounding box  $y$

$$\Delta(\hat{y}, y) = 1 - \frac{A(\hat{y}) \cap A(y)}{A(\hat{y}) \cup A(y)}$$

# Another Cost Function

$$C^n = \max_y [w \cdot \phi(x^n, y)] - w \cdot \phi(x^n, \hat{y}^n)$$

$$C^n = \max_y [\Delta(\hat{y}^n, y) + w \cdot \phi(x^n, y)] - w \cdot \phi(x^n, \hat{y}^n)$$





# Gradient Descent

$$C^n = \max_y [w \cdot \phi(x^n, y)] - w \cdot \phi(x^n, \hat{y}^n)$$

$$C^n = \max_y [\Delta(\hat{y}^n, y) + w \cdot \phi(x^n, y)] - w \cdot \phi(x^n, \hat{y}^n)$$

In each iteration, pick a training data  $\{x^n, \hat{y}^n\}$

$$\tilde{y}^n = \arg\max_y [w \cdot \phi(x^n, y)] \quad \arg\max_y [\Delta(\hat{y}^n, y) + w \cdot \phi(x^n, y)]$$

**Oh no! Problem 2.1**

$$\nabla C^n(w) = \phi(x^n, \tilde{y}^n) - \phi(x^n, \hat{y}^n)$$

$$w \rightarrow w - \eta [\phi(x^n, \tilde{y}^n) - \phi(x^n, \hat{y}^n)]$$

# Another Viewpoint

$$\tilde{y}^n = \arg \max_y w \cdot \phi(x^n, y)$$

- Minimizing the new cost function is minimizing the upper bound of the errors on training set

$$C' = \sum_{n=1}^N \Delta(\hat{y}^n, \tilde{y}^n) \leq C = \sum_{n=1}^N c^n \quad \text{upper bound}$$

We want to find  $w$  minimizing  $C'$  (errors)

It is hard!

Because  $y$  can be any kind of objects,  $\Delta(\cdot, \cdot)$  can be any function .....

$C$  serves as the surrogate of  $C'$

Proof that  $\Delta(\hat{y}^n, \tilde{y}^n) \leq c^n$

# Another Viewpoint

$$C^n = \max_y [\Delta(\hat{y}^n, y) + w \cdot \phi(x^n, y)] - w \cdot \phi(x^n, \hat{y}^n)$$

Proof that  $\Delta(\hat{y}^n, \tilde{y}^n) \leq C^n$

$$\Delta(\hat{y}^n, \tilde{y}^n) \leq \Delta(\hat{y}^n, \tilde{y}^n) + \underbrace{[w \cdot \phi(x^n, \tilde{y}^n) - w \cdot \phi(x^n, \hat{y}^n)]}_{\tilde{y}^n = \arg \max_y w \cdot \phi(x^n, y)} \geq 0$$

$$= [\Delta(\hat{y}^n, \tilde{y}^n) + w \cdot \phi(x^n, \tilde{y}^n)] - w \cdot \phi(x^n, \hat{y}^n)$$

$$\leq \max_y [\Delta(\hat{y}^n, y) + w \cdot \phi(x^n, y)] - w \cdot \phi(x^n, \hat{y}^n)$$

$$= C^n$$

# More Cost Functions

$$\Delta(\hat{y}^n, \tilde{y}^n) \leq C^n$$

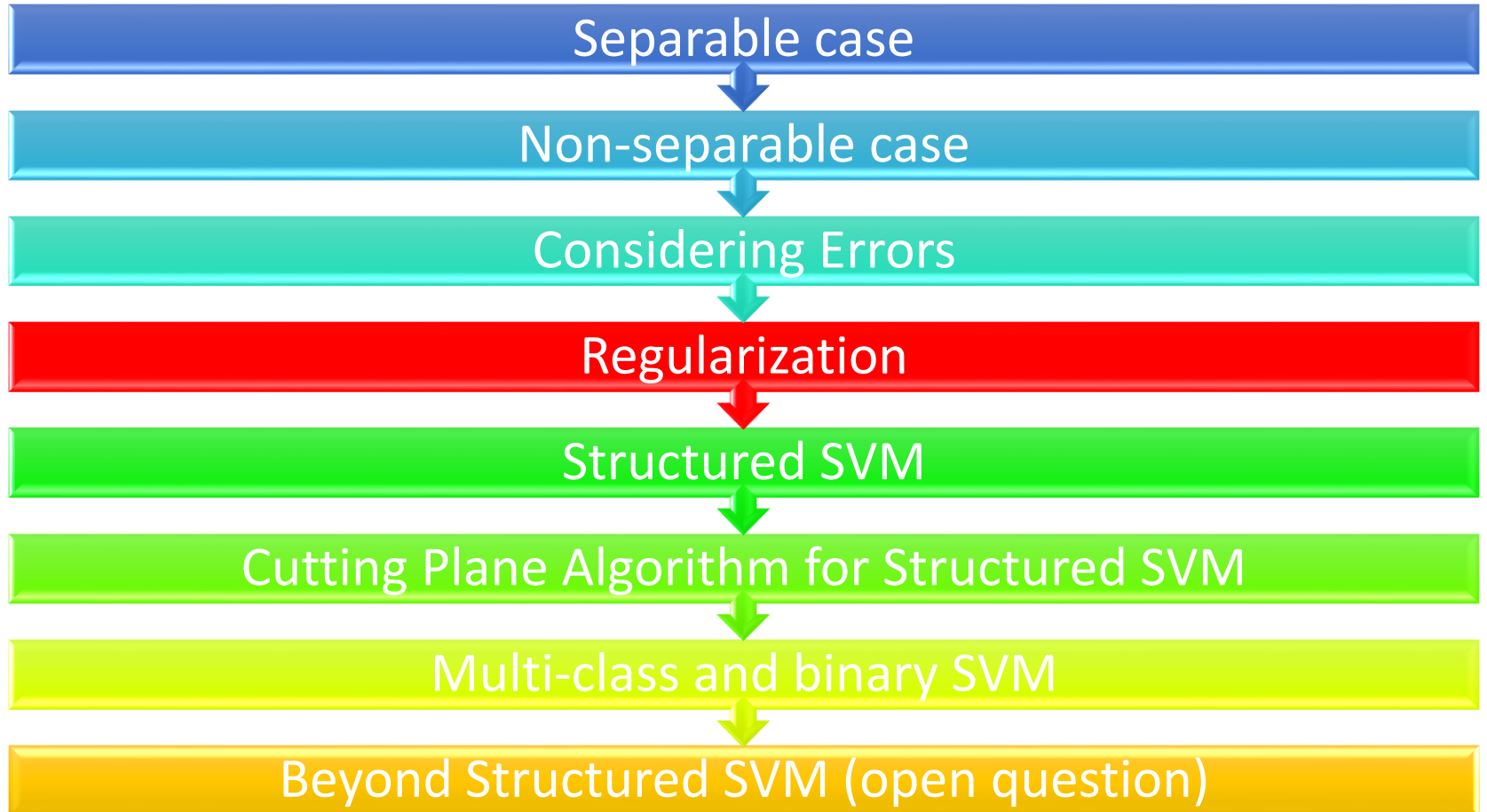
**Margin rescaling:**

$$C^n = \max_y [\Delta(\hat{y}^n, y) + w \cdot \phi(x^n, y)] - w \cdot \phi(x^n, \hat{y}^n)$$

**Slack variable rescaling:**

$$C^n = \max_y \Delta(\hat{y}^n, y) [1 + w \cdot \phi(x^n, y) - w \cdot \phi(x^n, \hat{y}^n)]$$

# Outline



# Regularization

Training data and testing data can have different distribution.

$w$  close to zero can minimize the influence of mismatch.

Keep the incorrect answer from a margin depending on errors

$$C = \sum_{n=1}^N C^n$$

$C^n$

$$= \max_y [\Delta(\hat{y}^n, y) + w \cdot \phi(x^n, y)] - w \cdot \phi(x^n, \hat{y}^n)$$

$$C = \underbrace{\frac{1}{2} \|w\|^2}_{\text{Regularization}} + \lambda \underbrace{\sum_{n=1}^N C^n}_{\text{margin depending on errors}}$$

Regularization:  
Find the  $w$  close to zero

# Regularization

$$C = \sum_{n=1}^N C^n \quad \longrightarrow \quad C = \frac{1}{2} \|w\|^2 + \lambda \sum_{n=1}^N C^n$$

In each iteration, pick a training data  $\{x^n, \hat{y}^n\}$

$$\bar{y}^n = \underset{y}{\operatorname{argmax}} [\Delta(\hat{y}^n, y) + w \cdot \phi(x^n, y)]$$

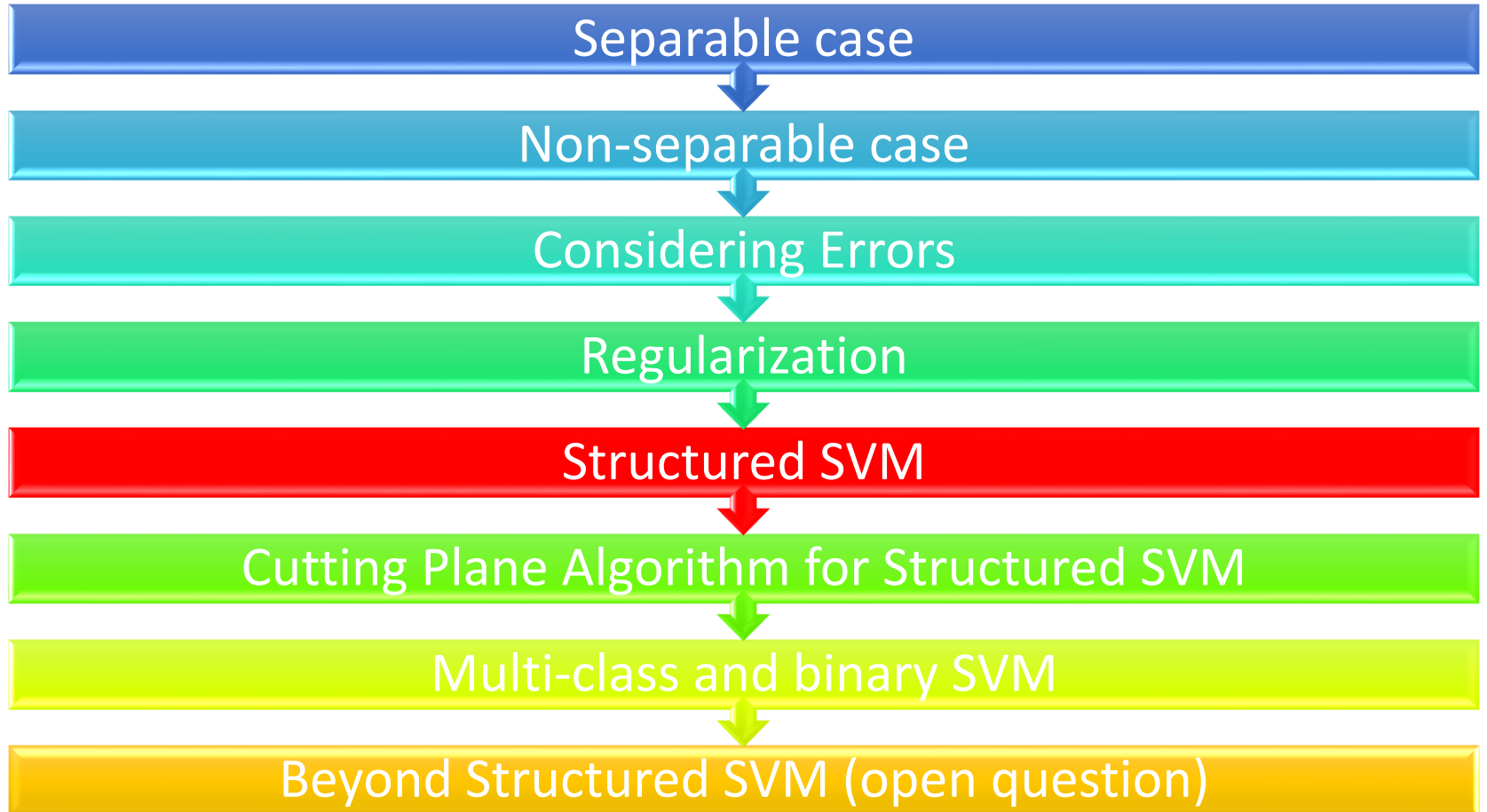
$$\nabla C^n = \phi(x^n, \bar{y}^n) - \phi(x^n, \hat{y}^n) + w$$

$$w \rightarrow w - \eta [\phi(x^n, \bar{y}^n) - \phi(x^n, \hat{y}^n)] - \eta w$$

$$= (1 - \eta)w - \eta [\phi(x^n, \bar{y}^n) - \phi(x^n, \hat{y}^n)]$$

Weight decay as in DNN

# Outline





# Structured SVM

Find  $w$  minimizing  $C$

$$C = \frac{1}{2} \|w\|^2 + \lambda \sum_{n=1}^N C^n$$

$$C^n = \max_y [\Delta(\hat{y}^n, y) + w \cdot \phi(x^n, y)] - w \cdot \phi(x^n, \hat{y}^n)$$

$$C^n + w \cdot \phi(x^n, \hat{y}^n) = \max_y [\Delta(\hat{y}^n, y) + w \cdot \phi(x^n, y)]$$

Are they equivalent?

We want to minimize  $C$

For  $\forall y$ :

$$C^n + w \cdot \phi(x^n, \hat{y}^n) \geq \Delta(\hat{y}^n, y) + w \cdot \phi(x^n, y)$$

$$w \cdot \phi(x^n, \hat{y}^n) - w \cdot \phi(x^n, y) \geq \Delta(\hat{y}^n, y) - C^n$$

# Structured SVM

Find  $w$  minimizing  $C$

$$C = \frac{1}{2} \|w\|^2 + \lambda \sum_{n=1}^N c^n$$

$$c^n = \max_y [\Delta(\hat{y}^n, y) + w \cdot \phi(x^n, y)] - w \cdot \phi(x^n, \hat{y}^n)$$

III

Find  $w, \varepsilon^1, \dots, \varepsilon^N$  minimizing  $C$

$$C = \frac{1}{2} \|w\|^2 + \lambda \sum_{n=1}^N \varepsilon^n$$

For  $\forall n$ :

For  $\forall y$ :

$$w \cdot \phi(x^n, \hat{y}^n) - w \cdot \phi(x^n, y) \geq \Delta(\hat{y}^n, y) - \varepsilon^n$$

*Slack variable*

# Structured SVM

Find  $w, \varepsilon^1, \dots, \varepsilon^N$  minimizing  $\mathcal{C}$

$$\mathcal{C} = \frac{1}{2} \|w\|^2 + \lambda \sum_{n=1}^N \varepsilon^n$$

For  $\forall n$ :

For  $\forall y$ :

$$w \cdot \phi(x^n, \hat{y}^n) - w \cdot \phi(x^n, y) \geq \Delta(\hat{y}^n, y) - \varepsilon^n$$

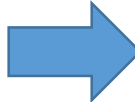
For  $\forall y \neq \hat{y}^n$ :

$$w \cdot (\phi(x^n, \hat{y}^n) - \phi(x^n, y)) \geq \Delta(\hat{y}^n, y) - \varepsilon^n, \quad \varepsilon^n \geq 0$$

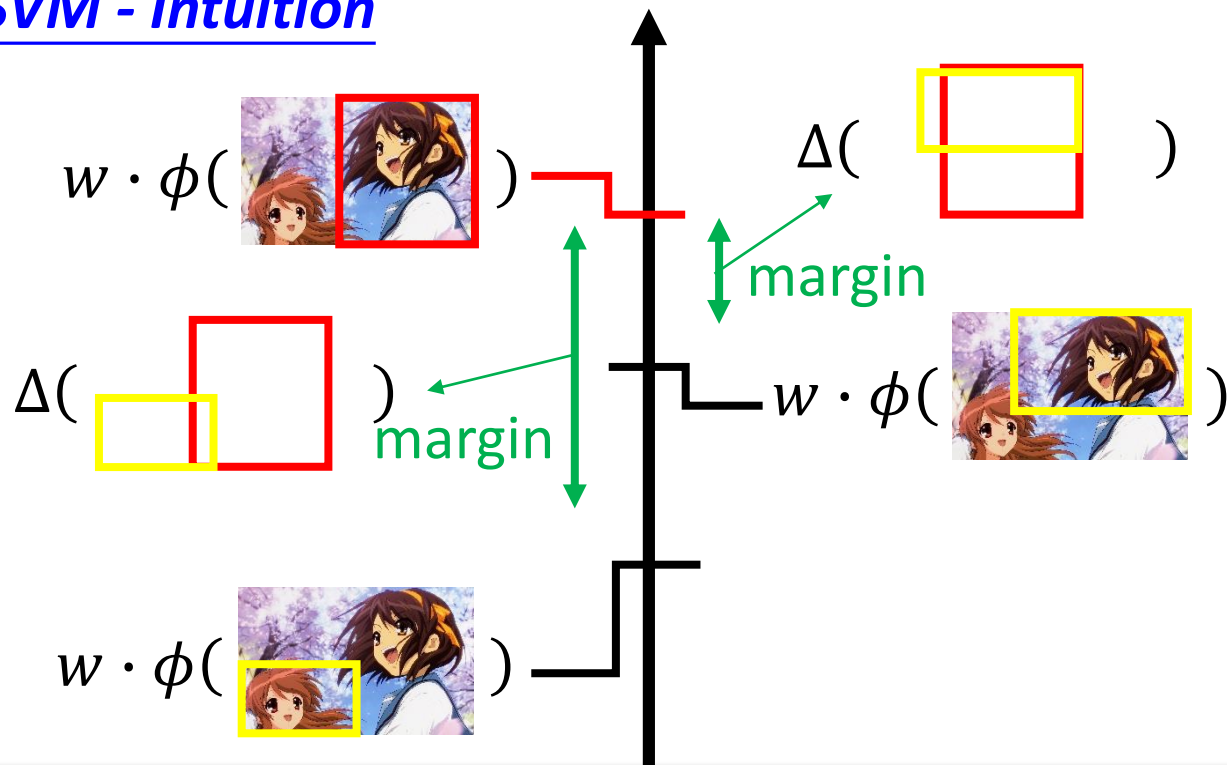
If  $y = \hat{y}^n$ :  $w \cdot \phi(x^n, \hat{y}^n) - w \cdot \phi(x^n, \hat{y}^n)$   $\geq$   $\Delta(\hat{y}^n, \hat{y}^n)$   $- \varepsilon^n$

$=0$

$=0$

  $\varepsilon^n \geq 0$

## Structured SVM - Intuition

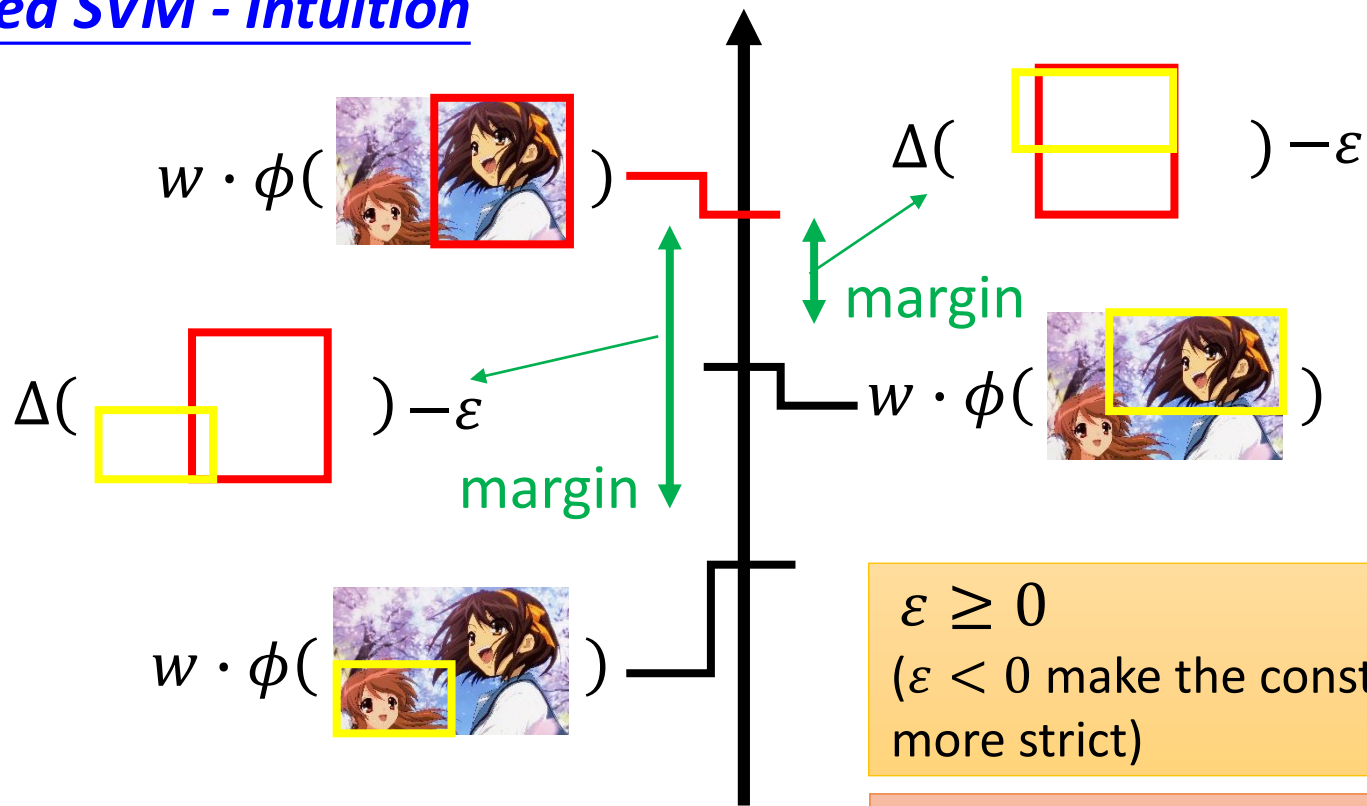


It is possible that no  $w$  can achieve this.

$$\left. \begin{aligned} w \cdot (\phi(\text{img}_1) - \phi(\text{img}_2)) &\geq \Delta(\text{margin}) \\ w \cdot (\phi(\text{img}_1) - \phi(\text{img}_3)) &\geq \Delta(\text{margin}) \end{aligned} \right\} \forall y \neq \hat{y}$$

(lots of inequalities)

# Structured SVM - Intuition



$$w \cdot (\phi(\text{image in red box}) - \phi(\text{image in yellow box})) \geq \Delta(\text{yellow box}) - \varepsilon$$

$$w \cdot (\phi(\text{image in red box}) - \phi(\text{image in yellow box})) \geq \Delta(\text{yellow box}) - \varepsilon$$

(lots of inequalities)

*slack variable*

# Structured SVM - Intuition

$$\text{Minimize } \frac{1}{2} \|w\|^2 + \lambda \sum_{n=1}^2 \varepsilon^n$$

For  $x^1$

$$\left. \begin{aligned} w \cdot (\phi(\text{img}_1, \text{img}_2)) - \phi(\text{img}_1, \text{img}_3) &\geq \Delta(\text{img}_1, \text{img}_3) - \varepsilon^1 \\ w \cdot (\phi(\text{img}_1, \text{img}_2)) - \phi(\text{img}_4, \text{img}_2) &\geq \Delta(\text{img}_4, \text{img}_2) - \varepsilon^1 \end{aligned} \right\} \forall y \neq \hat{y}^1$$

(lots of inequalities)

$\varepsilon^1 \geq 0$

For  $x^2$

$$\left. \begin{aligned} w \cdot (\phi(\text{img}_5, \text{img}_6)) - \phi(\text{img}_5, \text{img}_7) &\geq \Delta(\text{img}_5, \text{img}_7) - \varepsilon^2 \\ w \cdot (\phi(\text{img}_5, \text{img}_6)) - \phi(\text{img}_8, \text{img}_6) &\geq \Delta(\text{img}_8, \text{img}_6) - \varepsilon^2 \end{aligned} \right\} \forall y \neq \hat{y}^2$$

(lots of inequalities)

$\varepsilon^2 \geq 0$

Training data:

$\hat{y}^1$

$x^1$



$x^2$

$\hat{y}^2$



# Structured SVM

Find  $w, \varepsilon^1, \dots, \varepsilon^N$  minimizing  $C$

$$C = \frac{1}{2} \|w\|^2 + \lambda \sum_{n=1}^N \varepsilon^n$$

For  $\forall n$ :

For  $\forall y \neq \hat{y}^n$ :

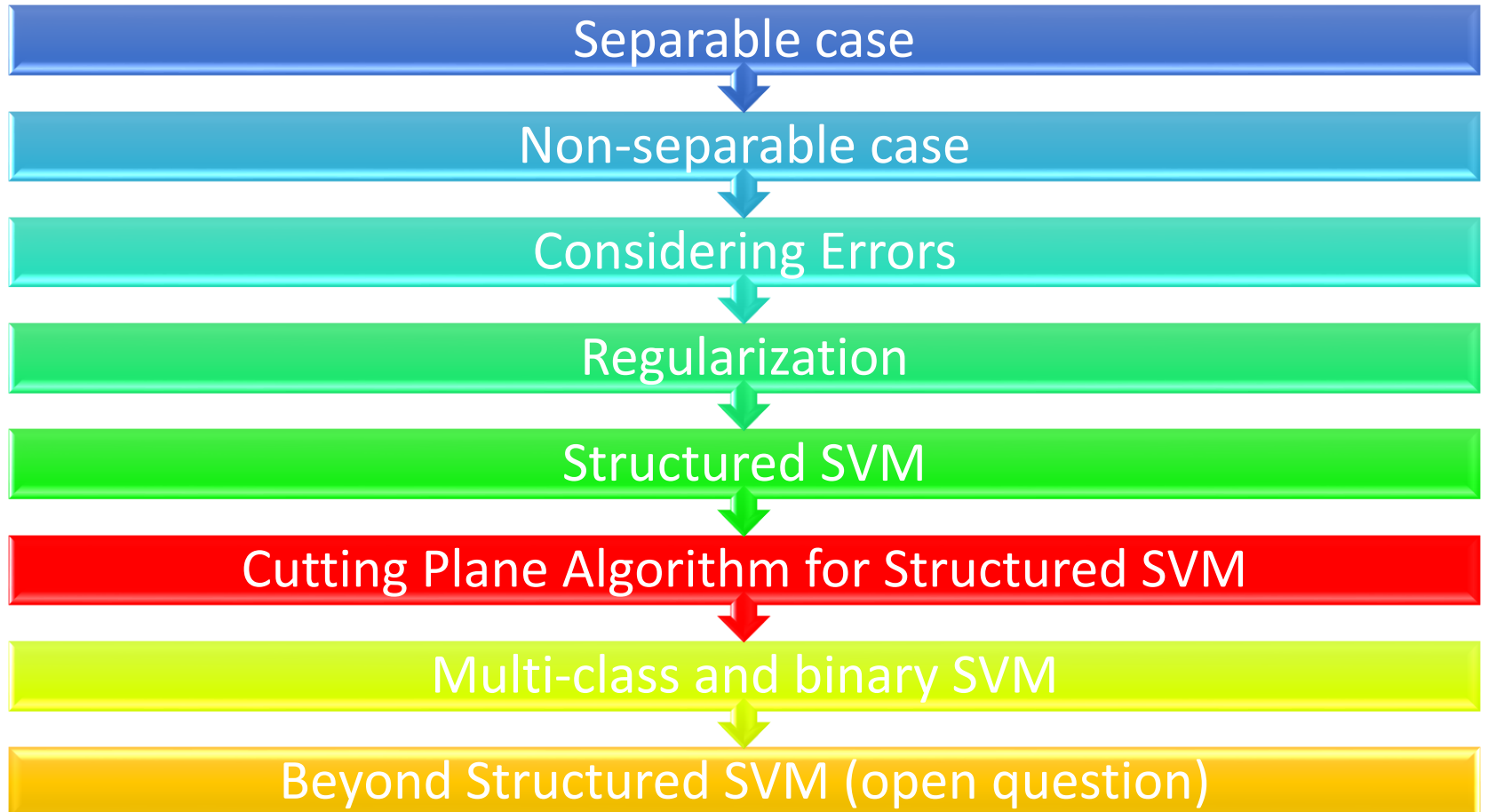
$$w \cdot (\phi(x^n, \hat{y}^n) - \phi(x^n, y)) \geq \Delta(\hat{y}^n, y) - \varepsilon^n, \quad \varepsilon^n \geq 0$$

Solve it by the solver in SVM package

Quadratic Programming (QP) Problem

**Too many constraints .....**

# Outline





Find  $w, \varepsilon^1, \dots, \varepsilon^N$  minimizing  $C$

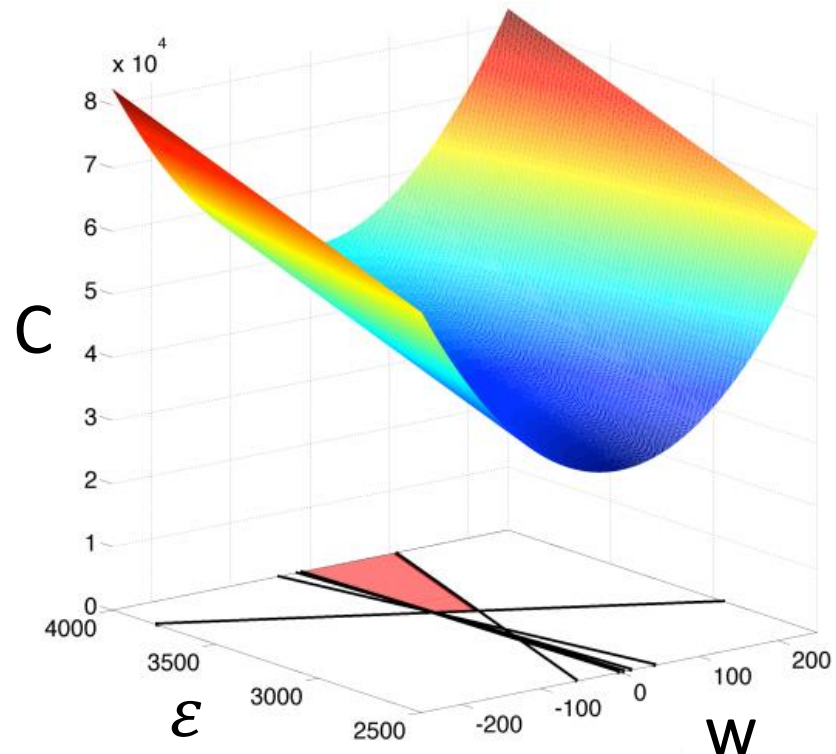
$$C = \frac{1}{2} \|w\|^2 + \lambda \sum_{n=1}^N \varepsilon^n$$

For  $\forall n$ :

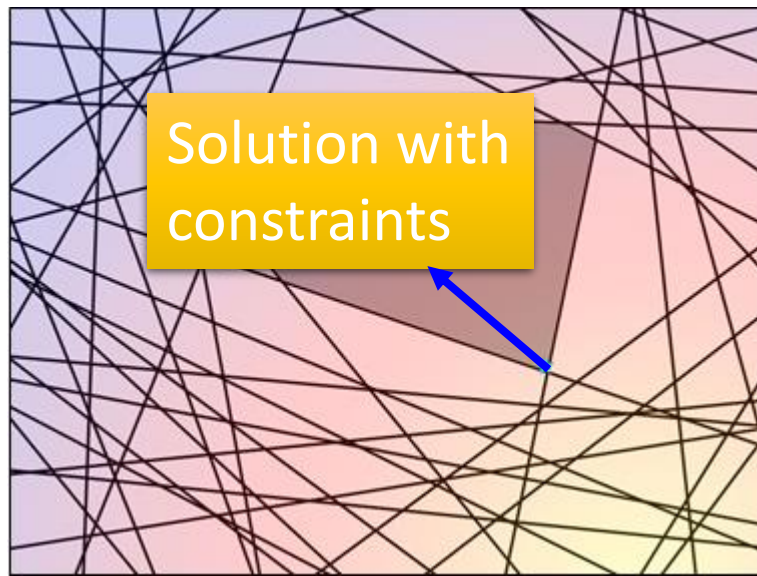
For  $\forall y \neq \hat{y}^n$ :

$$w \cdot (\phi(x^n, \hat{y}^n) - \phi(x^n, y)) \geq \Delta(\hat{y}^n, y) - \varepsilon^n, \quad \varepsilon^n \geq 0$$

Source of image:  
[http://abnerguzman.com/publications/gkb\\_aistats13.pdf](http://abnerguzman.com/publications/gkb_aistats13.pdf)



# Cutting Plane Algorithm



Parameter space  
 $(w, \varepsilon^1, \dots, \varepsilon^N)$

Color is the value of  $C$  which is going to be minimized:

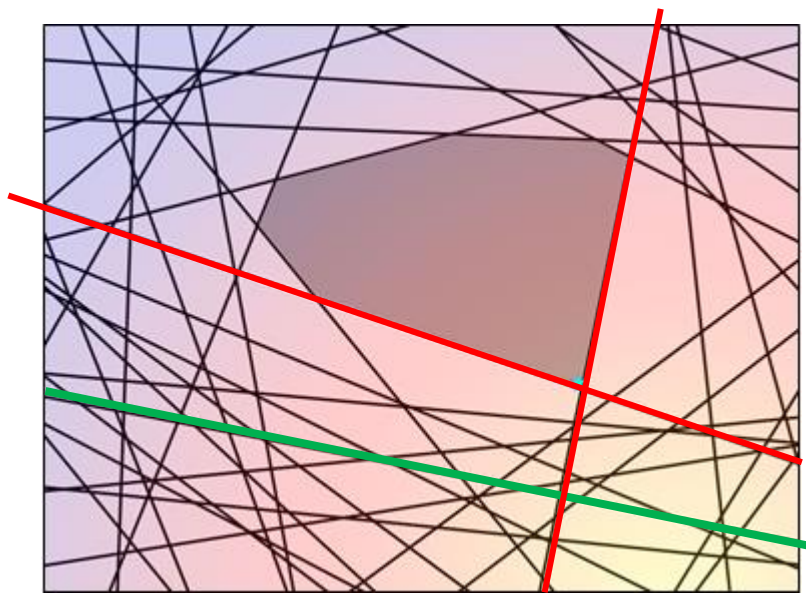
$$C = \frac{1}{2} \|w\|^2 + \lambda \sum_{n=1}^N \varepsilon^n$$

For  $\forall r, \forall y, y \neq \hat{y}^n$ :

- $w \cdot (\phi(x^n, \hat{y}^n) - \phi(x^n, y)) \geq \Delta(\hat{y}^n, y) - \varepsilon^n$
- $\varepsilon^n \geq 0$

# Cutting Plane Algorithm

Although there are lots of constraints, most of them do not influence the solution.



Parameter space  
 $(w, \varepsilon^1, \dots, \varepsilon^N)$

Red lines: determine the solution

Green line: Remove this constraint will not influence the solution

$$y \in \mathbb{A}^n$$

For  $\forall r, \forall y, y \neq \hat{y}^n$ :

- $w \cdot (\phi(x^n, \hat{y}^n) - \phi(x^n, y)) \geq \Delta(\hat{y}^n, y) - \varepsilon^n$
- $\varepsilon^n \geq 0$

$\mathbb{A}^n$ : a very small set of  $y \rightarrow$  **working set**

# Cutting Plane Algorithm

- Elements in **working set**  $\mathbb{A}^n$  is selected iteratively  
**Initialize**  $\mathbb{A}^1 \dots \mathbb{A}^N$

Find  $w, \varepsilon^1 \dots \varepsilon^N$  minimizing  $C$

$$C = \frac{1}{2} \|w\|^2 + \lambda \sum_{n=1}^N \varepsilon^n$$

Solve a QP  
problem

For  $\forall r$ :

For  $\forall y \in \mathbb{A}^n, y \neq \hat{y}^n$ :

$$w \cdot (\phi(x^n, \hat{y}^n) - \phi(x^n, y)) \geq \Delta(\hat{y}^n, y) - \varepsilon^n \quad \varepsilon^n \geq 0$$

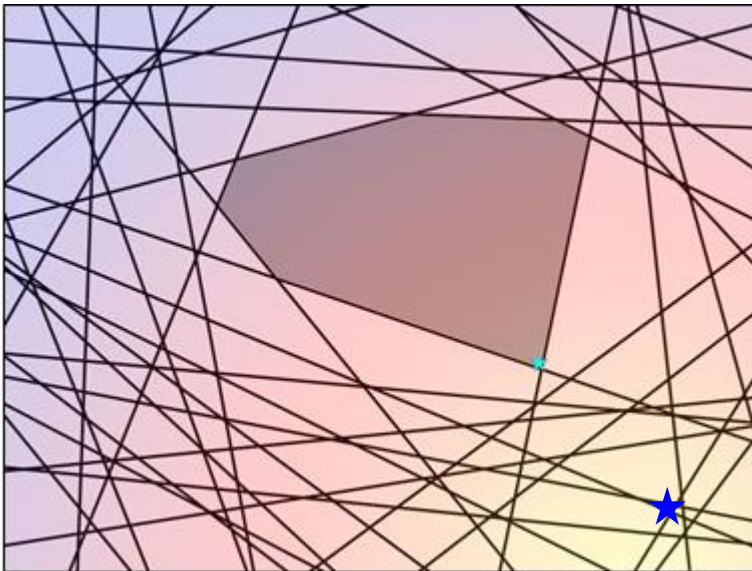
obtain  
solution  $w$

**Repeatedly**

Add elements  
into  $\mathbb{A}^1 \dots \mathbb{A}^N$

# Cutting Plane Algorithm

- Strategies of adding elements into **working set**  $A^n$



Initialize  $A^n = \text{null}$

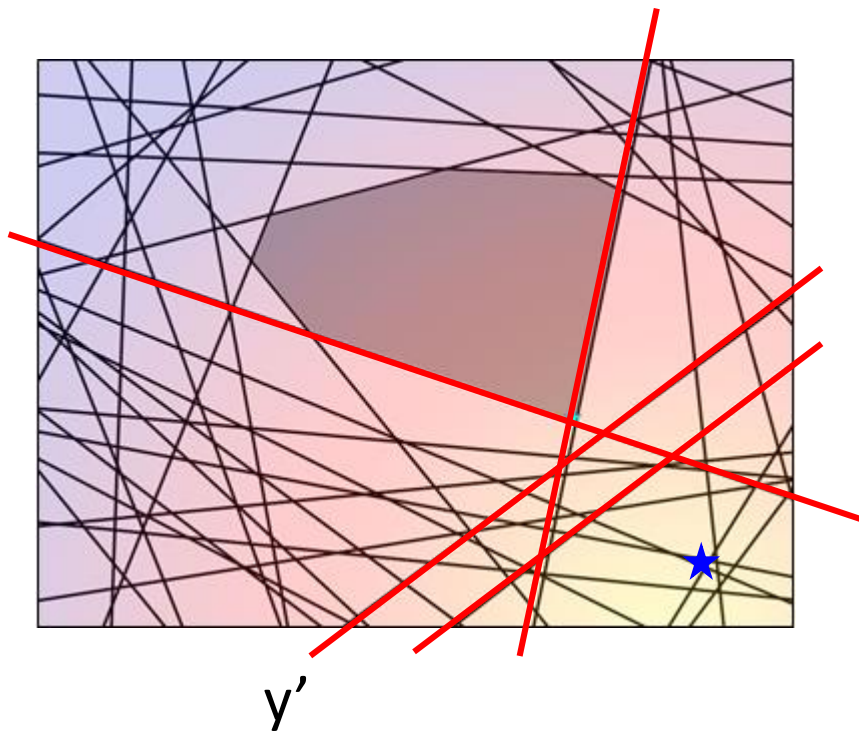
No constraint at all

Solving QP

The solution  $w$  is  
the blue point.

# Cutting Plane Algorithm

- Strategies of adding elements into **working set**  $A^n$



There are lots of constraints  
is violated

Find ***the most violated one***

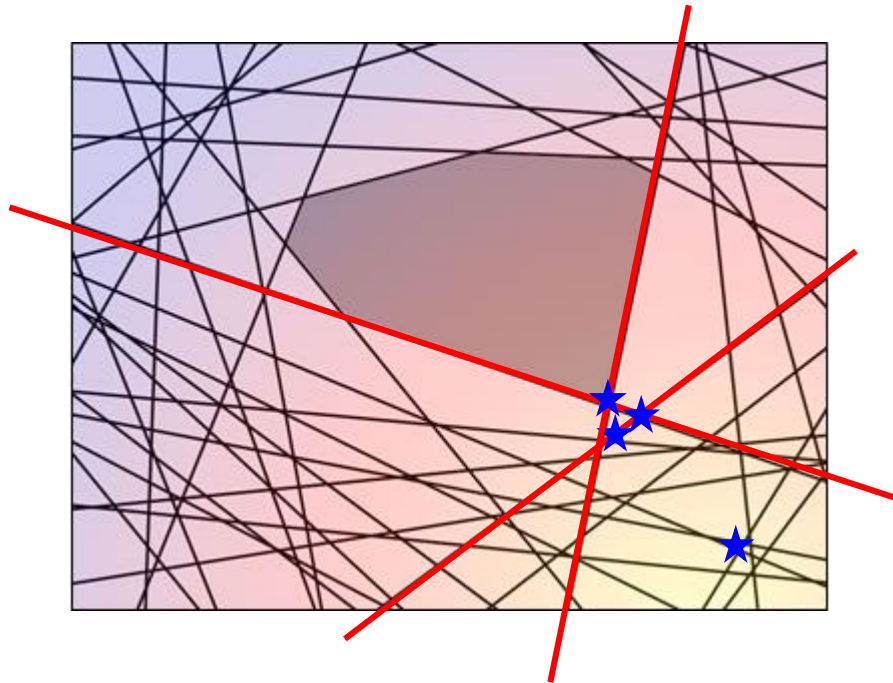
Suppose it is the constraint  
from  $y'$

Extent the working set

$$A^n = A^n \cup \{y'\}$$

# Cutting Plane Algorithm

- Strategies of adding elements into **working set**  $A^n$





# Find the most violated one

- Given  $w'$  and  $\varepsilon'$  from working sets at hand, which constraint is the most violated one?

**Constraint:**  $w \cdot (\phi(x, \hat{y}) - \phi(x, y)) \geq \Delta(\hat{y}, y) - \varepsilon$

**Violate a Constraint:**

$$w' \cdot (\phi(x, \hat{y}) - \phi(x, y)) < \Delta(\hat{y}, y) - \varepsilon'$$

**Degree of Violation**

$$\Delta(\hat{y}, y) - \varepsilon' - w' \cdot (\phi(x, \hat{y}) - \phi(x, y))$$

$$\longrightarrow \Delta(\hat{y}, y) + w' \cdot \phi(x, y)$$

**The most violated one:**

$$\arg \max_y [\Delta(\hat{y}, y) + w \cdot \phi(x, y)]$$



# Cutting Plane Algorithm

Given training data:  $\{(x^1, \hat{y}^1), (x^2, \hat{y}^2), \dots, (x^N, \hat{y}^N)\}$

Working Set  $\mathbb{A}^1 \leftarrow null, \mathbb{A}^2 \leftarrow null, \dots, \mathbb{A}^N \leftarrow null$

**Repeat**

$w \leftarrow$  Solve a **QP** with Working Set  $\mathbb{A}^1, \mathbb{A}^2, \dots, \mathbb{A}^N$

**QP:** Find  $w, \varepsilon^1 \dots \varepsilon^N$  minimizing  $\frac{1}{2} \|w\|^2 + \lambda \sum_{n=1}^N \varepsilon^n$

For  $\forall n$ :

For  $\forall y \in \mathbb{A}^n$ :

$$w \cdot (\phi(x^n, \hat{y}^n) - \phi(x^n, y)) \geq \Delta(\hat{y}^n, y) - \varepsilon^n, \varepsilon^n \geq 0$$

# Cutting Plane Algorithm

Given training data:  $\{(x^1, \hat{y}^1), (x^2, \hat{y}^2), \dots, (x^N, \hat{y}^N)\}$

Working Set  $\mathbb{A}^1 \leftarrow null, \mathbb{A}^2 \leftarrow null, \dots, \mathbb{A}^N \leftarrow null$

**Repeat**

$w \leftarrow$  Solve a **QP** with Working Set  $\mathbb{A}^1, \mathbb{A}^2, \dots, \mathbb{A}^N$

**For** each training data  $(x^n, \hat{y}^n)$ :

$$\bar{y}^n = \arg \max_y [\Delta(\hat{y}^n, y) + w \cdot \phi(x^n, y)]$$

find the most violated constraints

Update working set  $\mathbb{A}^n \leftarrow \mathbb{A}^n \cup \{\bar{y}^n\}$

**Until**  $\mathbb{A}^1, \mathbb{A}^2, \dots, \mathbb{A}^N$  doesn't change any more

**Return**  $w$

Training data:



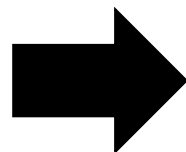
$$\mathbb{A}^1 = \{\}$$

$$\mathbb{A}^2 = \{\}$$

$$w = 0$$

**QP:** Find  $w, \varepsilon^1, \varepsilon^2$  minimizing  $\frac{1}{2} \|w\|^2 + \lambda \sum_{n=1}^2 \varepsilon^n$

There is no constraint



Solution:  $w = 0$

Training data:



$$\mathbb{A}^1 = \{\} \longrightarrow \mathbb{A}^1 = \{ \text{yellow box} \}$$

$$\mathbb{A}^2 = \{\} \longrightarrow \mathbb{A}^2 = \{ \text{yellow box} \}$$

$$w = 0$$

$$\bar{y}^1 = \arg \max_y [\Delta(\hat{y}^1, y) + 0 \cdot \phi(x^1, y)]$$

$$\Delta(\text{yellow box, red box}) + \cancel{w \cdot \phi(x^1, y)} = 0.90$$

$$\Delta(\text{yellow box, red box}) + \cancel{w \cdot \phi(x^1, y)} = 0.88$$

$$\Delta(\text{yellow box, red box}) + \cancel{w \cdot \phi(x^1, y)} = 0.25$$

$$\Delta(\text{yellow box, red box}) + \cancel{w \cdot \phi(x^1, y)} = 1.0$$

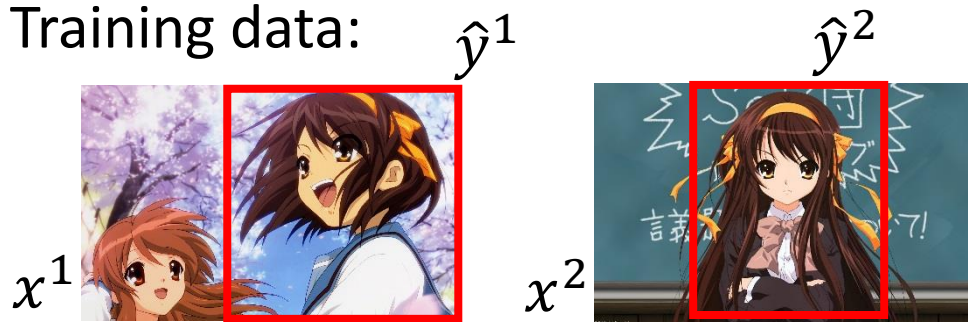
$$\Delta(\text{yellow box, red box}) + \cancel{w \cdot \phi(x^1, y)} = 1.0$$

$$\Delta(\text{yellow box, red box}) + \cancel{w \cdot \phi(x^1, y)} = 1.0$$

$\bar{y}^1$

$$\bar{y}^2 = \arg \max_y [\Delta(\hat{y}^2, y) + 0 \cdot \phi(x^2, y)]$$

Training data:



$$\mathbb{A}^1 = \{ \begin{bmatrix} 1 & 0 \end{bmatrix} \}$$

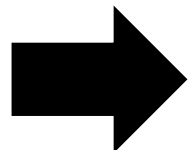
$$\mathbb{A}^2 = \{ \begin{bmatrix} 0 & 1 \end{bmatrix} \}$$

$$w = w^1$$

**QP:** Find  $w, \varepsilon^1, \varepsilon^2$  minimizing  $\frac{1}{2} \|w\|^2 + \lambda \sum_{n=1}^2 \varepsilon^n$

$$w \cdot (\phi(\text{img1\_red\_box}) - \phi(\text{img1\_yellow\_box})) \geq \Delta(\text{yellow\_box}, \text{red\_box}) - \varepsilon^1$$

$$w \cdot (\phi(\text{img2\_red\_box}) - \phi(\text{img2\_yellow\_box})) \geq \Delta(\text{red\_box}, \text{yellow\_box}) - \varepsilon^2$$



Solution:  $w = w^1$

Training data:

$\hat{y}^1$



$x^1$

$\hat{y}^2$



$x^2$

$$\mathbb{A}^1 = \{ \text{small square} \}$$

$$\mathbb{A}^2 = \{ \text{large square} \}$$

$$\mathbb{A}^2 = \{ \text{vertical rectangle} \}$$

$$\mathbb{A}^2 = \{ \text{vertical rectangle} \}$$

$$w = w^1$$

$$\bar{y}^1 = \arg \max_y [\Delta(\hat{y}^1, y) + w^1 \cdot \phi(x^1, y)]$$

$$\Delta(\text{small square}, \text{large square}) + w \cdot \phi(\text{image with small yellow box}, \text{image with large red box}) = 0.97$$

$$\Delta(\text{small square}, \text{large square}) + w \cdot \phi(\text{image with small yellow box}, \text{image with large red box}) = 1.55$$

$\bar{y}^1$

$$\Delta(\text{large square}, \text{large square}) + w \cdot \phi(\text{image with large yellow box}, \text{image with large red box}) = 1.25$$

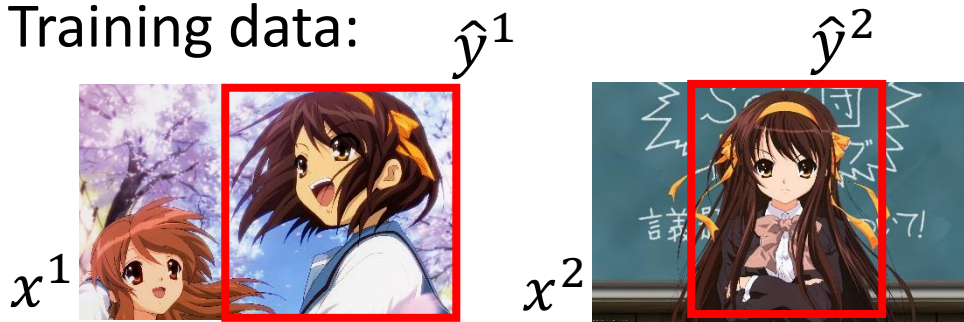
$$\Delta(\text{small square}, \text{large square}) + w \cdot \phi(\text{image with small yellow box}, \text{image with large red box}) = 1.01$$

$$\Delta(\text{vertical rectangle}, \text{large square}) + w \cdot \phi(\text{image with vertical yellow box}, \text{image with large red box}) = -0.99$$

$$\Delta(\text{small square}, \text{large square}) + w \cdot \phi(\text{image with small yellow box}, \text{image with large red box}) = -1.10$$

$$\bar{y}^2 = \arg \max_y [\Delta(\hat{y}^2, y) + w^1 \cdot \phi(x^2, y)]$$

Training data:



$$\mathbb{A}^1 = \{ \text{[red box]}, \text{[red box]} \}$$

$$\mathbb{A}^2 = \{ \text{[red box]}, \text{[red box]} \}$$

**QP:** Find  $w, \varepsilon^1, \varepsilon^2$  minimizing  $\frac{1}{2} \|w\|^2 + \lambda \sum_{r=1}^2 \varepsilon^r$

The process repeats iteratively

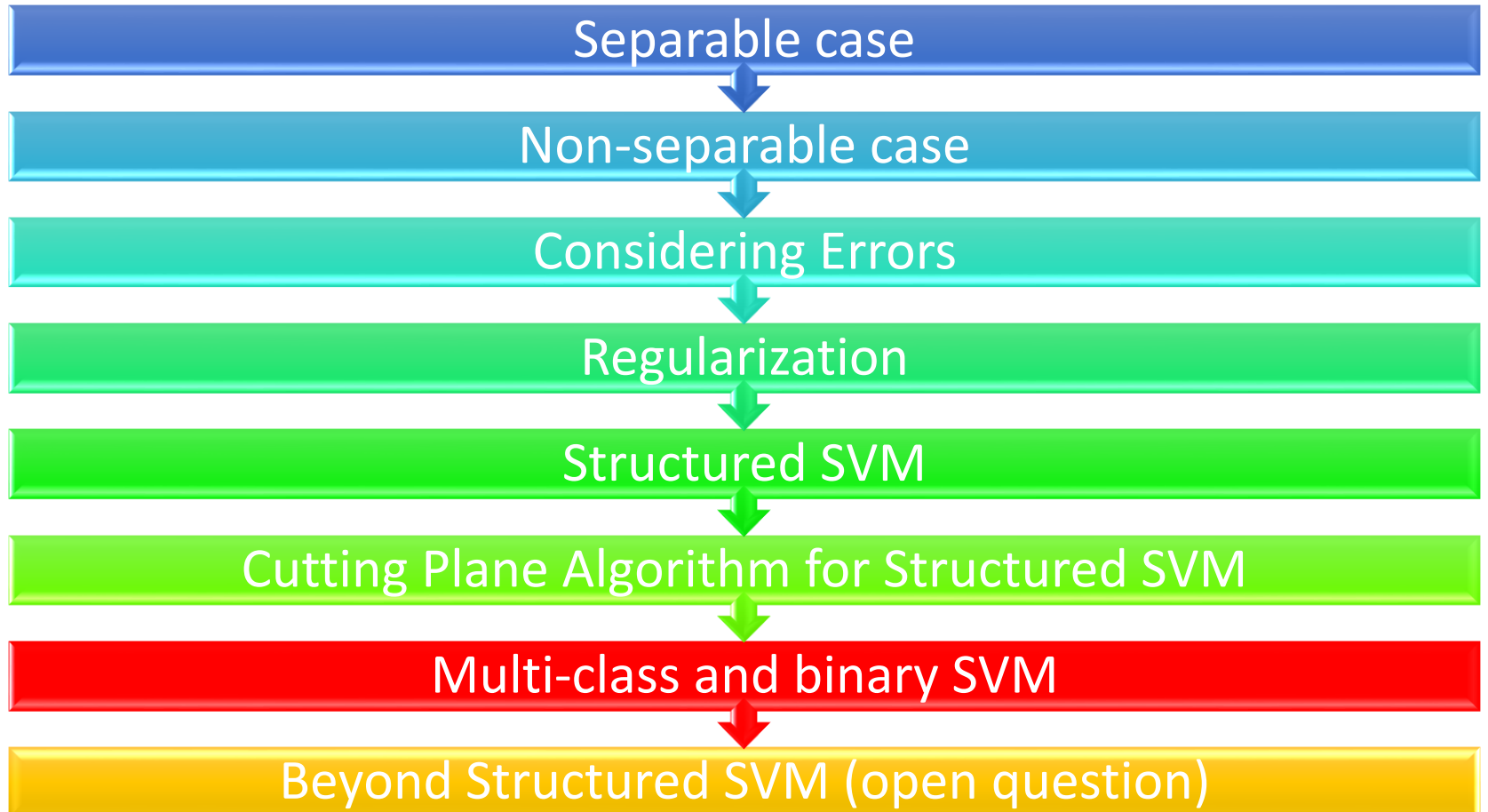
$$w \cdot (\phi(\text{[red box]} \text{ [red box]}) - \phi(\text{[red box]} \text{ [red box]})) \geq \Delta(\text{[red box]} \text{ [red box]}) - \varepsilon^1$$

$$w \cdot (\phi(\text{[red box]} \text{ [red box]}) - \phi(\text{[red box]} \text{ [red box]})) \geq \Delta(\text{[red box]} \text{ [red box]}) - \varepsilon^1$$

$$w \cdot (\phi(\text{[red box]} \text{ [red box]}) - \phi(\text{[red box]} \text{ [red box]})) \geq \Delta(\text{[red box]} \text{ [red box]}) - \varepsilon^2$$

$$w \cdot (\phi(\text{[red box]} \text{ [red box]}) - \phi(\text{[red box]} \text{ [red box]})) \geq \Delta(\text{[red box]} \text{ [red box]}) - \varepsilon^2$$

# Concluding Remarks





# Multi-class SVM

$$F(x, y) = w \cdot \phi(x, y)$$

- Problem 1: Evaluation
  - If there are K classes, then we have K weight vectors  $\{w^1, w^2, \dots, w^K\}$

$$y \in \{1, 2, \dots, k, \dots, K\}$$

$$F(x, y) = w^y \cdot \vec{x}$$

$\vec{x}$ : vector

representation of  $x$

$$w = \begin{bmatrix} w^1 \\ w^2 \\ \vdots \\ w^k \\ \vdots \\ w^K \end{bmatrix} \quad \phi(x, y) = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ \vec{x} \\ \vdots \\ 0 \end{bmatrix}$$

# Multi-class SVM

- Problem 2: Inference

$$F(x, y) = w^y \cdot \vec{x}$$

$$\hat{y} = \arg \max_{y \in \{1, 2, \dots, k, \dots, K\}} F(x, y)$$

$$= \arg \max_{y \in \{1, 2, \dots, k, \dots, K\}} w^y \cdot \vec{x}$$

The number of classes are usually small,  
so we can just enumerate them.

# Multi-class SVM

$$y \in \{dog, cat, bus, car\}$$

$$\Delta(\hat{y}^n = dog, y = cat) = 1$$

$$\Delta(\hat{y}^n = dog, y = bus) = 100$$

(defined as your wish)

- Problem 3: Training

Find  $w, \varepsilon^1, \dots, \varepsilon^N$  minimizing  $\mathcal{C}$

$$\mathcal{C} = \frac{1}{2} \|w\|^2 + \lambda \sum_{n=1}^N \varepsilon^n$$

For  $\forall n$ :

For  $\forall y \neq \hat{y}^n$ :

**There are only  $N(K-1)$  constraints.**

$$(w^{\hat{y}^n} - w^y) \cdot \vec{x} \geq \underline{\Delta(\hat{y}^n, y)} - \varepsilon^n, \quad \varepsilon^n \geq 0$$

$$w \cdot \phi(x^n, \hat{y}^n) = w^{\hat{y}^n} \cdot \vec{x}$$

$$w \cdot \phi(x^n, y) = w^y \cdot \vec{x}$$

Some types of misclassifications may be worse than others.

# Binary SVM

- Set  $K = 2$       $y \in \{1,2\}$

For  $\forall y \neq \hat{y}^n$ :

$$(w^{\hat{y}^n} - w^y) \cdot \vec{x} \geq \overset{=1}{\Delta(\hat{y}^n, y)} - \varepsilon^n, \quad \varepsilon^n \geq 0$$

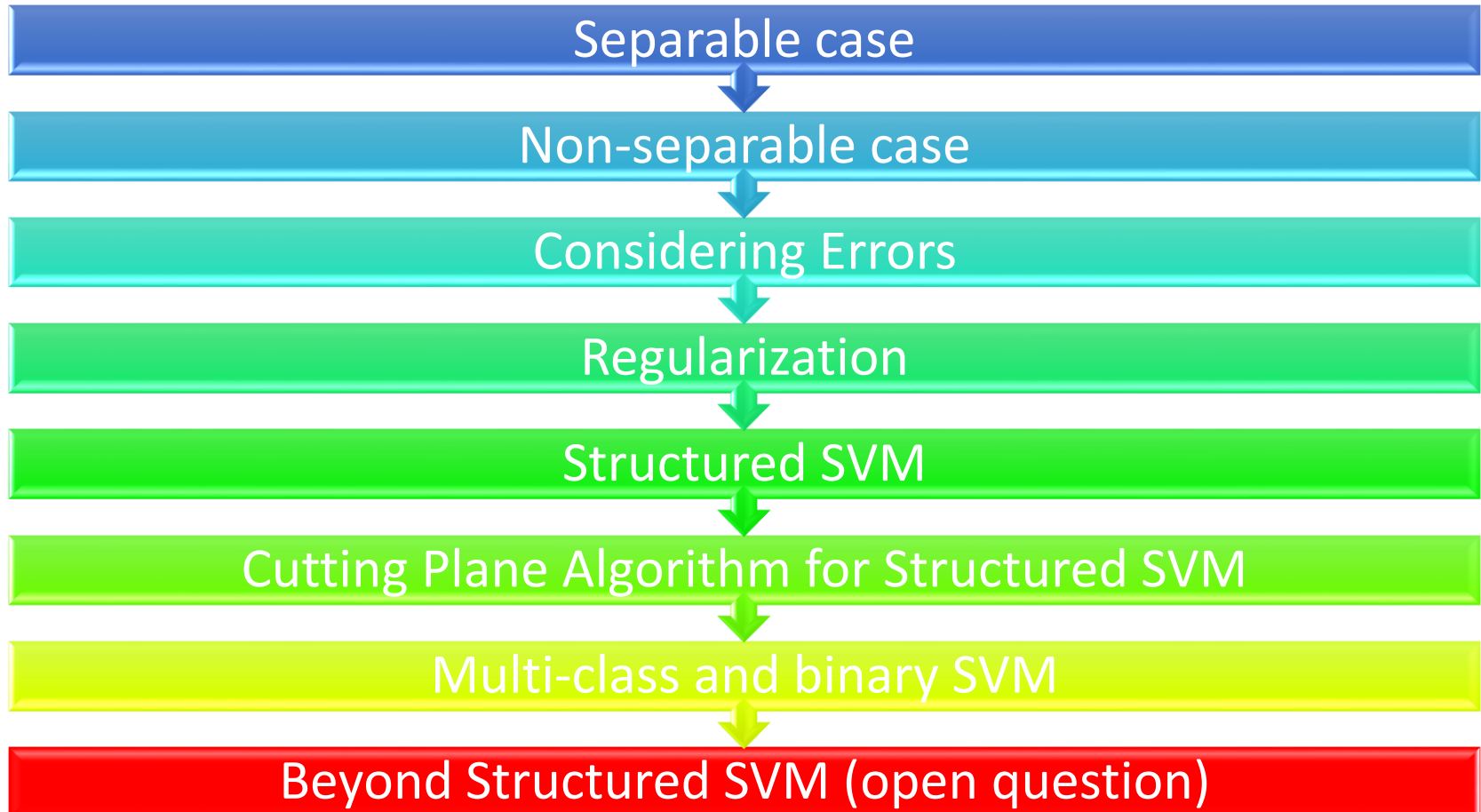
If  $y=1$ :  $(w^1 - w^2) \cdot \vec{x} \geq 1 - \varepsilon^n$   $\Rightarrow$   $w \cdot \vec{x} \geq 1 - \varepsilon^n$

$w$

If  $y=2$ :  $(w^2 - w^1) \cdot \vec{x} \geq 1 - \varepsilon^n$   $\Rightarrow$   $-w \cdot \vec{x} \geq 1 - \varepsilon^n$

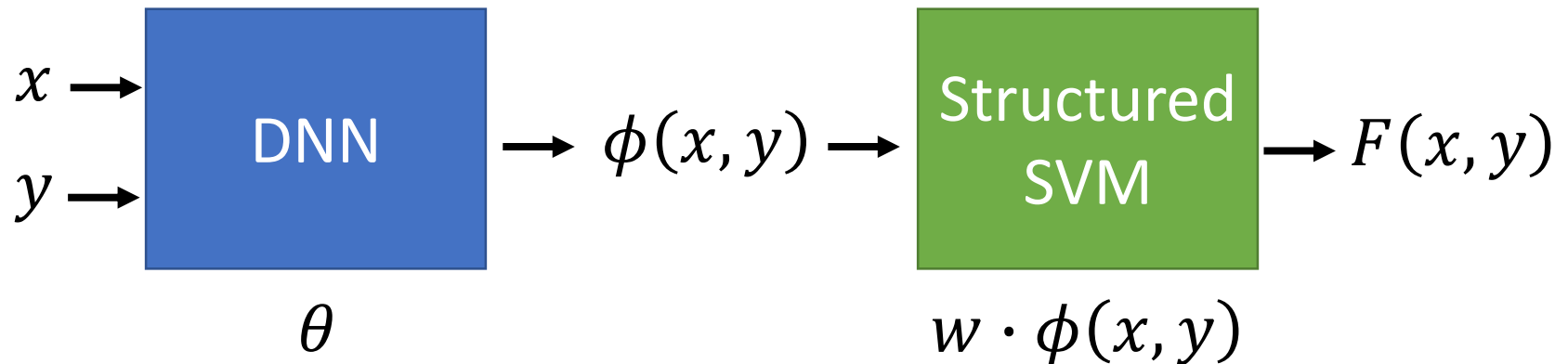
$-w$

# Concluding Remarks



# Beyond Structured SVM

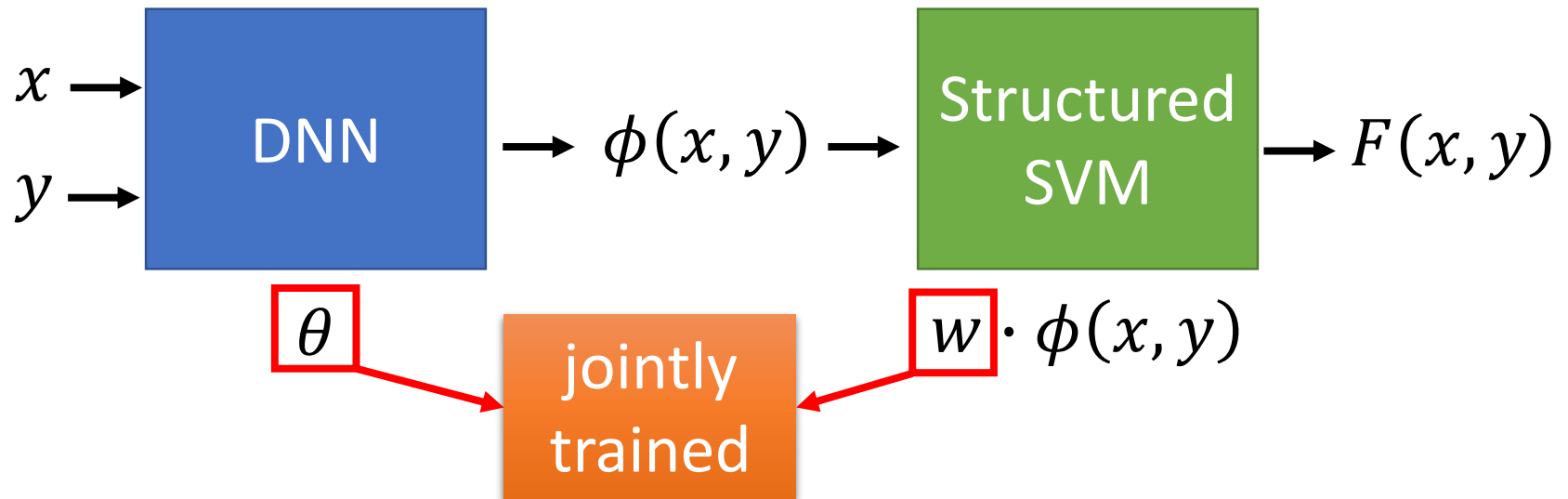
- Involving DNN when generating  $\phi(x, y)$



Ref: Hao Tang, Chao-hong Meng, Lin-shan Lee, "An initial attempt for phoneme recognition using Structured Support Vector Machine (SVM)," ICASSP, 2010  
Shi-Xiong Zhang, Gales, M.J.F., "Structured SVMs for Automatic Speech Recognition," in Audio, Speech, and Language Processing, IEEE Transactions on, vol.21, no.3, pp.544-555, March 2013

# Beyond Structured SVM

- Jointly training structured SVM and DNN

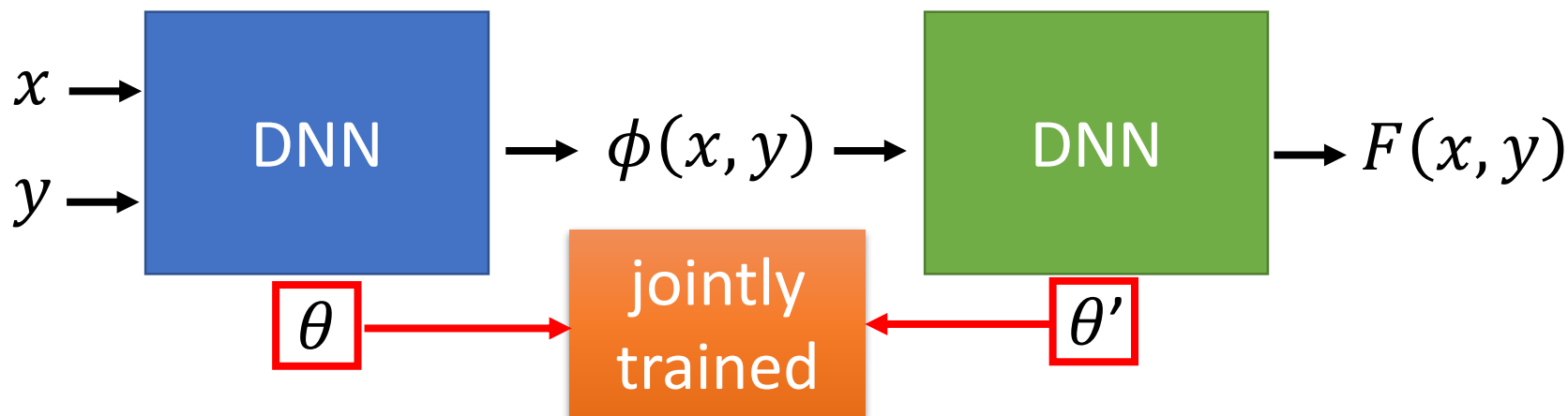


Ref: Shi-Xiong Zhang, Chaojun Liu, Kaisheng Yao, and Yifan Gong, "DEEP NEURAL SUPPORT VECTOR MACHINES FOR SPEECH RECOGNITION", Interspeech 2015

# Beyond Structured SVM

- Replacing Structured SVM with DNN

A DNN with  $x$  and  $y$  as input and  $F(x, y)$  (a scalar) as output



$$C = \frac{1}{2} \|\theta\|^2 + \frac{1}{2} \|\theta'\|^2 + \lambda \sum_{n=1}^N C^n$$

$$C^n = \max_y [\Delta(\hat{y}^n, y) + F(x^n, y)] - F(x^n, \hat{y}^n)$$

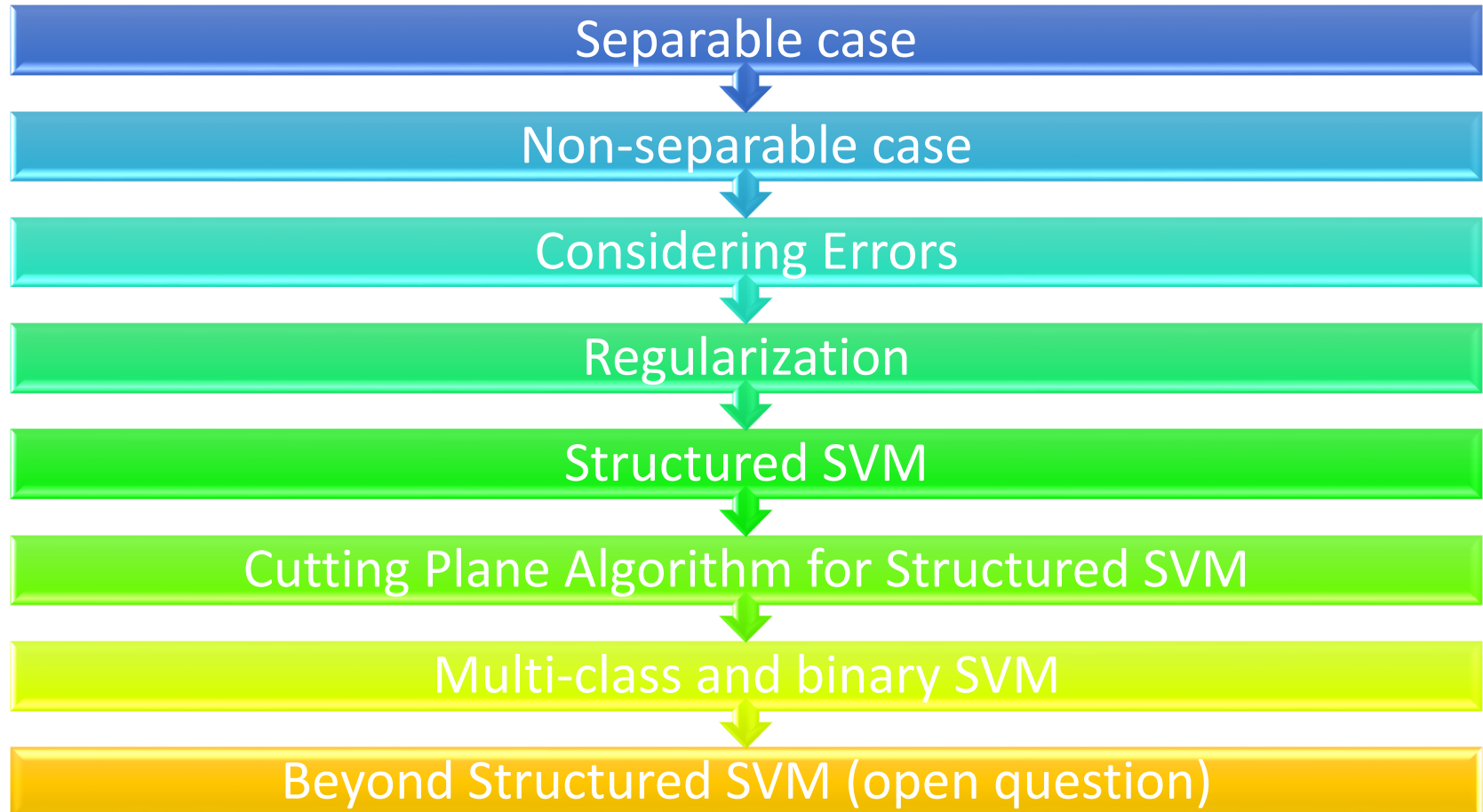
Ref: Yi-Hsiu Liao, Hung-yi Lee, Lin-shan Lee, "Towards Structured Deep Neural Network for Automatic Speech

Recognition"

[http://speech.ee.ntu.edu.tw/~tlkagk/paper/DNN\\_ASRU15.pdf](http://speech.ee.ntu.edu.tw/~tlkagk/paper/DNN_ASRU15.pdf)



# Concluding Remarks



# Acknowledgement

- 感謝 盧柏儒 同學於上課時發現投影片上的錯誤
- 感謝 徐翊祥 同學於上課時發現投影片上的錯誤