Structured Support Vector Machine
Hung-yi Lee
公告

• 因為作業二的 deadline 正好卡到期中考週，為了不要讓大家太辛苦，所以作業二的 deadline 延後一週
  • 作業二的 deadline 延後到 11/20
• 作業三公布的日期和 deadline 不變
  • 作業三公布的日期仍然為 11/13
• 也就是說，作業二和作業三會有一週的重疊
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 $F$
  $$F : X \times Y \to \mathbb{R}$$
- $F(x, y)$: 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://www.vision.ee.ethz.ch/~hpedemo/gallery.php
Problem 1: Evaluation

- $F(x,y)$ is linear

\[ F(x) = w \cdot \phi(x) \]

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

- 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: \[ \{(x^1, \hat{y}^1), (x^2, \hat{y}^2), \ldots, (x^N, \hat{y}^N)\} \]

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

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

1. Structured SVM
2. Regularization
3. Cutting Plane Algorithm for Structured SVM
4. Multi-class and binary SVM
5. Beyond Structured SVM (open question)
Outline

1. Separable case
2. Non-separable case
3. Considering Errors
4. Regularization
5. Structured SVM
6. Cutting Plane Algorithm for Structured SVM
7. Multi-class and binary SVM
8. Beyond Structured SVM (open question)
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 \( \{(x^1, \hat{y}^1), (x^2, \hat{y}^2), \ldots, (x^N, \hat{y}^N)\} \)
- **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 \(w \cdot \phi(x^n, y)\)
      \[
      \tilde{y}^n = \arg\max_{y \in Y} w \cdot \phi(x^n, y) \quad \text{(problem 2)}
      \]
    - 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

\[ w^0 = 0 \rightarrow w^1 \rightarrow w^2 \rightarrow \ldots \ldots \rightarrow w^k \rightarrow w^{k+1} \rightarrow \ldots \ldots \]

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

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

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

\[ \forall n \; \text{(All training examples)} \]
\[ \forall y \in Y - \{\hat{y}^n\} \; \text{(All incorrect label for an example)} \]

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

Assume \(\|\hat{w}\| = 1\) without loss of generality
**Proof of Termination**

w is updated *once it sees a mistake*

\[ w^0 = 0 \rightarrow w^1 \rightarrow w^2 \rightarrow \ldots \rightarrow w^k \rightarrow w^{k+1} \rightarrow \ldots \]

\[ w^k = w^{k-1} + \phi(x^n, \hat{y}^n) - \phi(x^n, \tilde{y}^n) \] (the relation of \( w^k \) 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?) \[
\cos \rho_k = \frac{\hat{w} \cdot w^k}{||\hat{w}|| \cdot ||w^k||}
\]

\[
\hat{w} \cdot w^k = \hat{w} \cdot (w^{k-1} + \phi(x^n, \hat{y}^n) - \phi(x^n, \tilde{y}^n)) \\
= \hat{w} \cdot w^{k-1} + \hat{w} \cdot \phi(x^n, \hat{y}^n) - \hat{w} \cdot \phi(x^n, \tilde{y}^n) \geq \hat{w} \cdot w^{k-1} + \delta \\
\geq \delta \quad \text{(Separable)}
\]
Proof of Termination

w is updated once it sees a mistake

\[ w^0 = 0 \rightarrow w^1 \rightarrow w^2 \rightarrow \ldots \ldots \rightarrow w^k \rightarrow w^{k+1} \rightarrow \ldots \ldots \]

\[ w^k = w^{k-1} + \phi(x^n, \hat{y}^n) - \phi(x^n, \tilde{y}^n) \] (the relation of \( w^k \) 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?) \[ \cos \rho_k = \frac{\hat{w} \cdot w^k}{\|\hat{w}\| \cdot \|w^k\|} \]

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

\[ = 0 \geq \delta \]

\[ \hat{w} \cdot w^1 \geq \hat{w} \cdot w^0 + \delta \]

\[ \hat{w} \cdot w^2 \geq \hat{w} \cdot w^1 + \delta \]

\[ \hat{w} \cdot w^1 \geq \delta \]

\[ \hat{w} \cdot w^2 \geq 2\delta \]

\[ \begin{cases} \hat{w} \cdot w^k \geq k \delta \end{cases} \]

(so what)
Proof of Termination

\[ \cos \rho_k = \frac{\hat{w}}{\|\hat{w}\|} \cdot \frac{w^k}{\|w^k\|} \]

\[ w^k = w^{k-1} + \phi(x^n, \hat{y}^n) - \phi(x^n, \tilde{y}^n) \]

\[ \|w^k\|^2 = \|w^{k-1} + \phi(x^n, \hat{y}^n) - \phi(x^n, \tilde{y}^n)\|^2 \]

\[ = \|w^{k-1}\|^2 + \|\phi(x^n, \hat{y}^n) - \phi(x^n, \tilde{y}^n)\|^2 + 2w^{k-1} \cdot (\phi(x^n, \hat{y}^n) - \phi(x^n, \tilde{y}^n)) \]

\[ > 0 \]

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

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

\[ \|w^1\|^2 \leq \|w^0\|^2 + R^2 = R^2 \]

\[ \|w^2\|^2 \leq \|w^1\|^2 + R^2 \leq 2R^2 \]

\[ \cdots \]

\[ \|w^k\|^2 \leq kR^2 \]
Proof of Termination

\[ \cos \rho_k = \frac{\hat{w} \cdot w^k}{\|\hat{w}\| \cdot \|w^k\|} \]

\[ \hat{w} \cdot w^k \geq k \delta \]

\[ \|w^k\|^2 \leq k R^2 \]

\[ \geq \frac{k \delta}{\sqrt{k R^2}} = \sqrt{k} \frac{\delta}{R} \]

\[ \sqrt{k} \frac{\delta}{R} \leq 1 \]

\[ k \leq \left( \frac{R}{\delta} \right)^2 \]
End of Warning

In separable case, to obtain a $\widehat{\mathbf{w}}$, 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?

The largest distances between features

Margin: Is it easy to separable red points from the blue ones

Normalization

Larger margin, less update

All feature times 2

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

\[ \phi(x^n, \hat{y}^n) \]

\[ \phi(x^n, y) \]

\[ \delta \]

\[ \hat{w} \]

\[ \vec{R} \]

\[ \delta \uparrow \]
Outline

- Separable case
- Non-separable case
- Considering Errors
- Regularization
- Structured SVM
- Cutting Plane Algorithm for Structured SVM
- Multi-class and binary SVM
- Beyond Structured SVM (open question)
Non-separable Case

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

Undoubtedly, \( w' \) is better than \( w'' \).
Defining Cost Function

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

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

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

What is the minimum value?

Other alternatives?
(Stochastic) Gradient Descent

Find $w$ minimizing the cost $C$

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

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

(Stochastic) Gradient descent:

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

However, there is “max” in $C^n$ ......
\[ C^n = \max_y [w \cdot \phi(x^n, y)] - w \cdot \phi(x^n, \hat{y}^n) \]

How to compute \( \nabla C^n \)?

When \( w \) is different, the \( y \) can be different.

**Space of \( w \)**

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

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

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

\[ \arg\max_y [w \cdot \phi(x^n, y)] \]
(Stochastic) Gradient Descent

For $t = 1$ to $T$: 

Update the parameters $T$ times

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

\[ \hat{y}^n = \underset{y}{\arg\max} [w \cdot \phi(x^n, y)] \]

\[ \nabla C^n = \phi(x^n, \hat{y}^n) - \phi(x^n, \hat{y}^n) \]

\[ w \rightarrow w - \eta \nabla C^n \]

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

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

- Beyond Structured SVM (open question)
- Multi-class and binary SVM
- Cutting Plane Algorithm for Structured SVM
- Structured SVM
- Regularization
- Considering Errors
- Non-separable case
- Separable case
Based on what we have considered ......

\[ F(x, y) \]

\[ w \cdot \phi(\text{something}) \]

\[ w \cdot \phi(\text{something}) \]

\[ w \cdot \phi(\text{something}) \]

\[ w \cdot \phi(\text{something}) \]

The right case is better.

Treat all incorrectly equally

\[ F(x, y) \]

\[ w \cdot \phi(\text{something}) \]

\[ w \cdot \phi(\text{something}) \]

\[ w \cdot \phi(\text{something}) \]

\[ w \cdot \phi(\text{something}) \]

acceptable

very bad!
Considering the incorrect ones

\[ w \cdot \phi(\text{different from correct box}) \]

\[ w \cdot \phi(\text{close to correct box}) \]

How to measure the difference
Defining Error Function

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

\[
A(y): \text{area of bounding box } y
\]

\[
\Delta(\hat{y}, y) = 1 - \frac{A(\hat{y}) \cap A(y)}{A(\hat{y}) \cup A(y)}
\]
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 \( \{x^n, \hat{y}^n\} \)

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

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

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

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

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

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

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

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

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

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

\[ = C^n \]
More Cost Functions

\[ \Delta(\hat{y}^n, \check{y}^n) \leq C^n \]

**Margin rescaling:**

\[ C^n = \max_{y} \left[ \Delta(\hat{y}^n, y) + w \cdot \phi(x^n, y) \right] - w \cdot \phi(x^n, \check{y}^n) \]

**Slack variable rescaling:**

\[ C^n = \max_{y} \Delta(\hat{y}^n, y) \left[ 1 + w \cdot \phi(x^n, y) - w \cdot \phi(x^n, \check{y}^n) \right] \]
Regularization

Training data and testing data can have different distribution.

\( w \) close to zero can minimize the influence of mismatch.

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

Regularization:
Find the \( w \) close to zero

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

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

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

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

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

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

Weight decay as in DNN
Outline

Beyond Structured SVM (open question)

Multi-class and binary SVM

Cutting Plane Algorithm for Structured SVM

Structured SVM

Regularization

Considering Errors

Non-separable case

Separable case
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 $C$

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

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

Find $w, \varepsilon^1, \ldots, \varepsilon^N$ 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 \( w, \varepsilon^1, \ldots, \varepsilon^N \) minimizing \( C \)

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

For \( \forall n \):

For \( \forall y \):

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

For \( \forall y \neq \hat{y}^n \):

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

If \( y = \hat{y}^n \):

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

\[
= 0 = 0 \quad \Rightarrow \quad \varepsilon^n \geq 0
\]
Structured SVM - Intuition

It is possible that no \( w \) can achieve this.

\[
\begin{align*}
  w \cdot \phi(\text{image}) - \phi(\text{image}) & \geq \Delta(\text{margin}) \\
  \forall y \neq \hat{y} \quad \text{(lots of inequalities)}
\end{align*}
\]
Structured SVM - Intuition

\[ w \cdot \phi(x) \geq \Delta - \varepsilon \]
\[ w \cdot \phi(y) \leq \Delta - \varepsilon \]

\( \varepsilon \geq 0 \)
\( (\varepsilon < 0 \text{ make the constraints more strict}) \)

\( \varepsilon \) should be minimized

\[ w \cdot (\phi(x) - \phi(y)) \geq \Delta - \varepsilon \]

(lots of inequalities)

slack variable
Structured SVM - Intuition

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

For \( x^1 \)

\[ w \cdot (\phi(x^1)) - \phi(y^1) \geq \Delta(\hat{y}^1) - \varepsilon^1 \]
\[ \forall y \neq \hat{y}^1 \]

(lots of inequalities)

\[ \varepsilon^1 \geq 0 \]

For \( x^2 \)

\[ w \cdot (\phi(x^2)) - \phi(y^2) \geq \Delta(\hat{y}^2) - \varepsilon^2 \]
\[ \forall y \neq \hat{y}^2 \]

(lots of inequalities)

\[ \varepsilon^2 \geq 0 \]
Structured SVM

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

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

For $\forall n$:

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

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

Solve it by the solver in SVM package

Quadratic Programming (QP) Problem

Too many constraints ......
Outline

1. Separable case
2. Non-separable case
3. Considering Errors
4. Regularization
5. Structured SVM
6. Cutting Plane Algorithm for Structured SVM
7. Multi-class and binary SVM
8. Beyond Structured SVM (open question)
Find $w, \epsilon^1, \ldots, \epsilon^N$ minimizing $C$

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

For $\forall n$:

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

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

Cutting Plane Algorithm

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

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.

\[ (w, \varepsilon^1, \ldots, \varepsilon^N) \]

\( y \in A^n \)

For \( \forall r, \forall \gamma, y \neq \gamma^n \):
- \( w \cdot (\phi(x^n, \gamma^n) - \phi(x^n, y)) \geq \Delta(\gamma^n, y) - \varepsilon^n \)
- \( \varepsilon^n \geq 0 \)

\( A^n \): a very small set of \( y \to \text{working set} \)
Cutting Plane Algorithm

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

Initialize $\mathbb{A}^1 \ldots \mathbb{A}^N$

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

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

For $\forall r$:

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

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

$\varepsilon^n \geq 0$

Obtain solution $w$

Repeatedly

Add elements into $\mathbb{A}^1 \ldots \mathbb{A}^N$
Cutting Plane Algorithm

- Strategies of adding elements into **working set** $\mathbb{A}^n$

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

No constraint at all

Solving QP

The solution $w$ is the blue point.

Image credit: Yisong Yue
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

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

[Image credit: Yisong Yue]
Cutting Plane Algorithm

• Strategies of adding elements into *working set* $\mathbb{A}^n$
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*

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

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

*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), \ldots, (x^N, \hat{y}^N) \}

Working Set \( A^1 \leftarrow \text{null} \), \( A^2 \leftarrow \text{null} \), \ldots, \( A^N \leftarrow \text{null} \)

Repeat

\( w \leftarrow \text{Solve a QP with Working Set } A^1, A^2, \ldots, A^N \)

\[ \text{QP: Find } w, \varepsilon^1 \ldots \varepsilon^N \text{ minimizing } \frac{1}{2} ||w||^2 + \lambda \sum_{n=1}^{N} \varepsilon^n \]

For \( \forall n \):

For \( \forall y \in A^n \):

\[ w \cdot (\phi(x^n, \hat{y}^n) - \phi(x^n, y)) \geq \Delta(\hat{y}^n, y) - \varepsilon^n, \varepsilon^n \geq 0 \]
Cutting Plane Algorithm

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

Working Set \( A^1 \leftarrow \text{null}, A^2 \leftarrow \text{null}, \ldots, A^N \leftarrow \text{null} \)

Repeat

1. \( w \leftarrow \text{Solve a QP with Working Set } A^1, A^2, \ldots, A^N \)

2. 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

3. Update working set \( A^n \leftarrow A^n \cup \{\bar{y}^n\} \)

Until \( A^1, A^2, \ldots, A^N \) doesn’t change any more

Return \( w \)
Training data:

\[ x^1 \quad y^1 \quad x^2 \quad y^2 \]

\[ w = 0 \]

\[ \mathbb{A}^1 = \emptyset \]

\[ \mathbb{A}^2 = \emptyset \]

**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:

\[ x^1 \quad y^1 \quad \hat{y}^1 \quad x^2 \quad y^2 \quad \hat{y}^2 \]

\[ \mathbb{A}^1 = \{ \} \quad \rightarrow \quad \mathbb{A}^1 = \{ \square \} \]

\[ \mathbb{A}^2 = \{ \} \quad \rightarrow \quad \mathbb{A}^2 = \{ \square \} \]

\[ w = 0 \]

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

\[ \Delta(\quad ) + w \cdot \phi(\quad ) = 0.90 \]

\[ \Delta(\quad ) + w \cdot \phi(\quad ) = 0.25 \]

\[ \Delta(\quad ) + w \cdot \phi(\quad ) = 1.0 \]

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

\[ \Delta(\quad ) + w \cdot \phi(\quad ) = 1.0 \]

\[ y^1 = 1.0 \]
Training data:

\[ x_1, y_1 \quad x_2, y_2 \]

\[ \hat{y}_1, \hat{y}_2 \]

\[ A^1 = \{ \square \} \]

\[ A^2 = \{ \Box \} \]

\[ w = w^1 \]

**QP:** Find \( w, \varepsilon^1, \varepsilon^2 \) minimizing

\[
\frac{1}{2} \|w\|^2 + \lambda \sum_{n=1}^{2} \varepsilon^n
\]

\[
w \cdot (\phi(x_1) - \phi(x_2)) \geq \Delta(\square) - \varepsilon^1
\]

\[
w \cdot (\phi(x_2) - \phi(x_1)) \geq \Delta(\Box) - \varepsilon^2
\]

**Solution:** \( w = w^1 \)
Training data: \( \hat{y}^1 \) \( \hat{y}^2 \) 

\[ \begin{align*}
\hat{y}^1 &= \arg\max_y [\Delta(\hat{y}^1, y) + w^1 \cdot \phi(x^1, y)] \\
\Delta(\hat{y}^1, y) + w \cdot \phi(x^1, y) &= 0.97 \\
\Delta(\hat{y}^1, y) + w \cdot \phi(x^1, y) &= 1.25 \\
\Delta(\hat{y}^1, y) + w \cdot \phi(x^1, y) &= 0.97 \\
\Delta(\hat{y}^2, y) + w^1 \cdot \phi(x^2, y) &= -0.99 \\
\hat{y}^2 &= \arg\max_y [\Delta(\hat{y}^2, y) + w^1 \cdot \phi(x^2, y)] \\
\end{align*} \]
Training data: $\hat{y}^1$, $\hat{y}^2$

$x^1$, $x^2$

$A^1 = \{ \square, \square \}$

$A^2 = \{ \square, \square \}$

**QP:** Find $w, \varepsilon^1, \varepsilon^2$ minimizing

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

The process repeats iteratively.
Concluding Remarks

- Separable case
- Non-separable case
- Considering Errors
- Regularization
- Structured SVM
- Cutting Plane Algorithm for Structured SVM
- Multi-class and binary SVM
- Beyond Structured SVM (open question)
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 \( \{w^1, w^2, \ldots, w^K\} \)

\( y \in \{1, 2, \ldots, k, \ldots, K\} \)

\[ F(x, y) = w^y \cdot \tilde{x} \]

\( \tilde{x} \): vector representation of \( x \)
Multi-class SVM

- Problem 2: Inference

\[ F(x, y) = w^y \cdot \hat{x} \]

\[ \hat{y} = \arg \max_{y \in \{1, 2, \ldots, k, \ldots, K\}} F(x, y) \]

\[ = \arg \max_{y \in \{1, 2, \ldots, k, \ldots, K\}} w^y \cdot \hat{x} \]

The number of classes are usually small, so we can just enumerate them.
Multi-class SVM

• Problem 3: Training

Find \( w, \varepsilon^1, \cdots, \varepsilon^N \) minimizing \( C \)

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

For \( \forall n \):

For \( \forall y \neq \hat{y}^n \):

\[
(w^{\hat{y}^n} - wy) \cdot \tilde{x} \geq \Delta(\hat{y}^n, y) - \varepsilon^n, \: \varepsilon^n \geq 0
\]

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

\[
w \cdot \phi(x^n, y) = wy \cdot \tilde{x}
\]

Some types of misclassifications may be worse than others.

\( y \in \{\text{dog, cat, bus, car}\} \)

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

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

(defined as your wish)

There are only \( N(K-1) \) constraints.
Binary SVM

- Set $K = 2$ \( y \in \{1,2\} \)

For \( \forall y \neq \hat{y}^n \):

\[
(w^{\hat{y}^n} - w^y) \cdot \hat{x} \geq \Delta(\hat{y}^n, y) - \varepsilon^n, \; \varepsilon^n \geq 0
\]

If \( y = 1 \):

\[
(w^1 - w^2) \cdot \hat{x} \geq 1 - \varepsilon^n
\]

\( w \)

\[
w \cdot \hat{x} \geq 1 - \varepsilon^n
\]

If \( y = 2 \):

\[
(w^2 - w^1) \cdot \hat{x} \geq 1 - \varepsilon^n
\]

\( -w \)

\[
-w \cdot \hat{x} \geq 1 - \varepsilon^n
\]
Concluding Remarks

- Separable case
- Non-separable case
- Considering Errors
- Regularization
- Structured SVM
- Cutting Plane Algorithm for Structured SVM
- Multi-class and binary SVM
- Beyond Structured SVM (open question)
Beyond Structured SVM

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

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

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

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


Concluding Remarks

Separable case

Non-separable case

Considering Errors

Regularization

Structured SVM

Cutting Plane Algorithm for Structured SVM

Multi-class and binary SVM

Beyond Structured SVM (open question)
Acknowledgement

- 感謝 劉柏儒 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