

Machine Learning Foundations

(機器學習基石)



Lecture 12: Nonlinear Transformation

Hsuan-Tien Lin (林軒田)

`htlin@csie.ntu.edu.tw`

Department of Computer Science
& Information Engineering

National Taiwan University
(國立台灣大學資訊工程系)



Roadmap

- 1 When Can Machines Learn?
- 2 Why Can Machines Learn?
- 3 **How** Can Machines Learn?

Lecture 11: Linear Models for Classification

binary classification via **(logistic) regression**;
multiclass via **OVA/OVO decomposition**

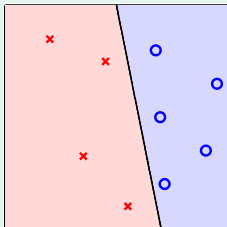
Lecture 12: Nonlinear Transformation

- Quadratic Hypotheses
- Nonlinear Transform
- Price of Nonlinear Transform
- Structured Hypothesis Sets

- 4 How Can Machines Learn Better?

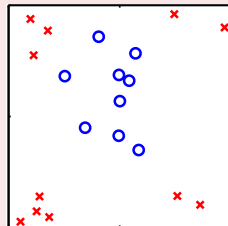
Linear Hypotheses

up to now: linear hypotheses



- visually: **'line'-like** boundary
- mathematically: linear scores $\mathbf{s} = \mathbf{w}^T \mathbf{x}$

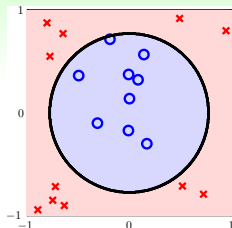
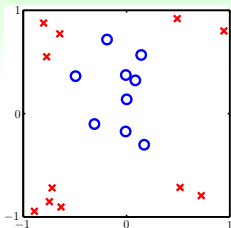
but limited ...



- theoretically: d_{VC} **under control :-)**
- practically: on some \mathcal{D} , **large** E_{in} for every line :-)

how to **break the limit** of linear hypotheses

Circular Separable



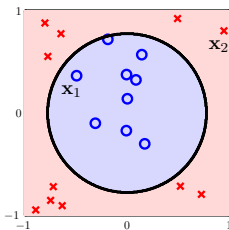
- \mathcal{D} not linear separable
- but **circular separable** by a circle of radius $\sqrt{0.6}$ centered at origin:

$$h_{\text{SEP}}(\mathbf{x}) = \text{sign} \left(-x_1^2 - x_2^2 + 0.6 \right)$$

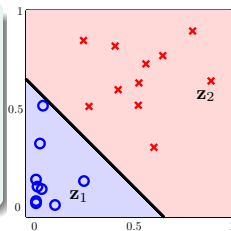
re-derive **Circular**-PLA, **Circular**-Regression,
blahblah ... all over again? :-)

Circular Separable and Linear Separable

$$\begin{aligned}
 h(\mathbf{x}) &= \text{sign} \left(\underbrace{0.6}_{\tilde{w}_0} \cdot \underbrace{1}_{z_0} + \underbrace{(-1)}_{\tilde{w}_1} \cdot \underbrace{x_1^2}_{z_1} + \underbrace{(-1)}_{\tilde{w}_2} \cdot \underbrace{x_2^2}_{z_2} \right) \\
 &= \text{sign} \left(\tilde{\mathbf{w}}^T \mathbf{z} \right)
 \end{aligned}$$



- $\{(\mathbf{x}_n, y_n)\}$ circular separable
 $\implies \{(\mathbf{z}_n, y_n)\}$ linear separable
- $\mathbf{x} \in \mathcal{X} \xrightarrow{\Phi} \mathbf{z} \in \mathcal{Z}$:
 (nonlinear) feature transform Φ



circular separable in $\mathcal{X} \implies$ linear separable in \mathcal{Z}
vice versa?

Linear Hypotheses in \mathcal{Z} -Space

$$(z_0, z_1, z_2) = \mathbf{z} = \Phi(\mathbf{x}) = (1, x_1^2, x_2^2)$$

$$h(\mathbf{x}) = \tilde{h}(\mathbf{z}) = \text{sign} \left(\tilde{\mathbf{w}}^T \Phi(\mathbf{x}) \right) = \text{sign} \left(\tilde{w}_0 + \tilde{w}_1 x_1^2 + \tilde{w}_2 x_2^2 \right)$$

$$\tilde{\mathbf{w}} = (\tilde{w}_0, \tilde{w}_1, \tilde{w}_2)$$

- $(0.6, -1, -1)$: circle (○ inside)
- $(-0.6, +1, +1)$: circle (○ outside)
- $(0.6, -1, -2)$: ellipse
- $(0.6, -1, +2)$: hyperbola
- $(0.6, +1, +2)$: **constant** ○ :-)

lines in \mathcal{Z} -space
 \iff **special** quadratic curves in \mathcal{X} -space

General Quadratic Hypothesis Set

a 'bigger' \mathcal{Z} -space with $\Phi_2(\mathbf{x}) = (1, x_1, x_2, x_1^2, x_1x_2, x_2^2)$

perceptrons in \mathcal{Z} -space \iff quadratic hypotheses in \mathcal{X} -space

$$\mathcal{H}_{\Phi_2} = \left\{ h(\mathbf{x}) : h(\mathbf{x}) = \tilde{h}(\Phi_2(\mathbf{x})) \text{ for some linear } \tilde{h} \text{ on } \mathcal{Z} \right\}$$

- can **implement all possible quadratic curve boundaries**: circle, ellipse, **rotated** ellipse, hyperbola, parabola, ...

$$\text{ellipse } 2(x_1 + x_2 - 3)^2 + (x_1 - x_2 - 4)^2 = 1$$

$$\iff \tilde{\mathbf{w}}^T = [33, -20, -4, 3, 2, 3]$$

- include **lines and constants as degenerate cases**

next: **learn** a good quadratic hypothesis g

Fun Time

Using the transform $\Phi_2(\mathbf{x}) = (1, x_1, x_2, x_1^2, x_1 x_2, x_2^2)$, which of the following weights $\tilde{\mathbf{w}}^T$ in the \mathcal{Z} -space implements the parabola $2x_1^2 + x_2 = 1$?

- 1 $[-1, 2, 1, 0, 0, 0]$
- 2 $[0, 2, 1, 0, -1, 0]$
- 3 $[-1, 0, 1, 2, 0, 0]$
- 4 $[-1, 2, 0, 0, 0, 1]$

Fun Time

Using the transform $\Phi_2(\mathbf{x}) = (1, x_1, x_2, x_1^2, x_1 x_2, x_2^2)$, which of the following weights $\tilde{\mathbf{w}}^T$ in the \mathcal{Z} -space implements the parabola $2x_1^2 + x_2 = 1$?

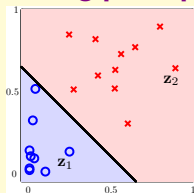
- ① $[-1, 2, 1, 0, 0, 0]$
- ② $[0, 2, 1, 0, -1, 0]$
- ③ $[-1, 0, 1, 2, 0, 0]$
- ④ $[-1, 2, 0, 0, 0, 1]$

Reference Answer: ③

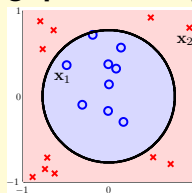
Too simple, uh? :-) Flexibility to implement arbitrary quadratic curves opens new possibilities for minimizing E_{in} !

Good Quadratic Hypothesis

\mathcal{Z} -space
 perceptrons
good perceptron
 separating perceptron



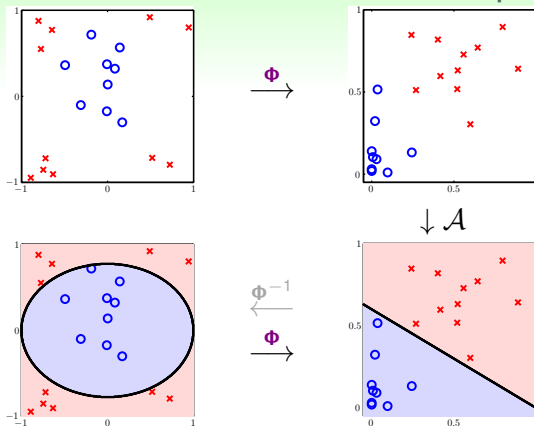
\mathcal{X} -space
 quadratic hypotheses
good quadratic hypothesis
 separating quadratic hypothesis



- want: get **good perceptron** in \mathcal{Z} -space
- known: get **good perceptron** in \mathcal{X} -space with data $\{(\mathbf{x}_n, y_n)\}$

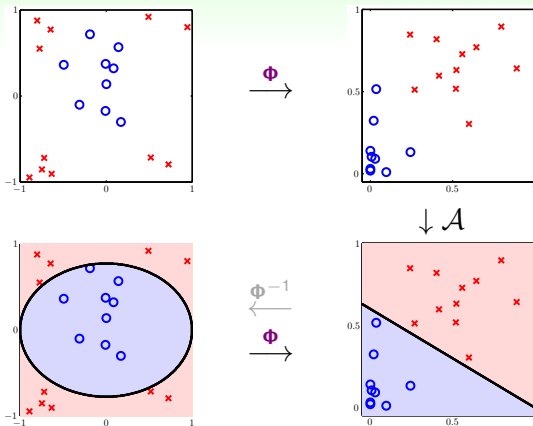
todo: get **good perceptron** in \mathcal{Z} -space with data $\{(\mathbf{z}_n = \Phi_2(\mathbf{x}_n), y_n)\}$

The Nonlinear Transform Steps



- 1 transform original data $\{(\mathbf{x}_n, y_n)\}$ to $\{(\mathbf{z}_n = \Phi(\mathbf{x}_n), y_n)\}$ by Φ
- 2 get a good perceptron $\tilde{\mathbf{w}}$ using $\{(\mathbf{z}_n, y_n)\}$ and your favorite linear classification algorithm \mathcal{A}
- 3 return $g(\mathbf{x}) = \text{sign}(\tilde{\mathbf{w}}^T \Phi(\mathbf{x}))$

Nonlinear Model via Nonlinear Φ + Linear Models

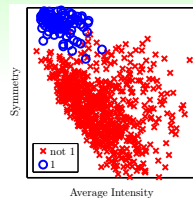
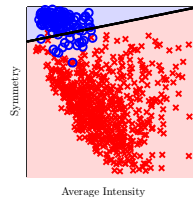


two choices:

- feature transform Φ
- linear model \mathcal{A} ,
not just binary classification

Pandora's box :-):

can now freely do **quadratic PLA, quadratic regression, cubic regression, ..., polynomial regression**

Feature Transform Φ 
 $\downarrow \mathcal{A}$


not new, not just polynomial:

raw (pixels) $\xrightarrow{\text{domain knowledge}}$ concrete (intensity, symmetry)

the force, too good to be true? :-)

Fun Time

Consider the quadratic transform $\Phi_2(\mathbf{x})$ for $\mathbf{x} \in \mathbb{R}^d$ instead of in \mathbb{R}^2 . The transform should include all different quadratic, linear, and constant terms formed by (x_1, x_2, \dots, x_d) . What is the number of dimensions of $\mathbf{z} = \Phi_2(\mathbf{x})$?

- 1 d
- 2 $\frac{d^2}{2} + \frac{3d}{2} + 1$
- 3 $d^2 + d + 1$
- 4 2^d

Fun Time

Consider the quadratic transform $\Phi_2(\mathbf{x})$ for $\mathbf{x} \in \mathbb{R}^d$ instead of in \mathbb{R}^2 . The transform should include all different quadratic, linear, and constant terms formed by (x_1, x_2, \dots, x_d) . What is the number of dimensions of $\mathbf{z} = \Phi_2(\mathbf{x})$?

- ① d
- ② $\frac{d^2}{2} + \frac{3d}{2} + 1$
- ③ $d^2 + d + 1$
- ④ 2^d

Reference Answer: ②

Number of different quadratic terms is $\binom{d}{2} + d$;
number of different linear terms is d ;
number of different constant term is 1.

Computation/Storage Price

Q -th order polynomial transform: $\Phi_Q(\mathbf{x}) = \left(\begin{array}{l} 1, \\ x_1, x_2, \dots, x_d, \\ x_1^2, x_1 x_2, \dots, x_d^2, \\ \dots, \\ x_1^Q, x_1^{Q-1} x_2, \dots, x_d^Q \end{array} \right)$

$\underbrace{1}_{\tilde{w}_0} + \underbrace{\tilde{d}}_{\text{others}}$ dimensions
 = # ways of $\leq Q$ -combination from d kinds with repetitions
 = $\binom{Q+d}{Q} = \binom{Q+d}{d} = \mathcal{O}(Q^d)$
 = efforts needed for computing/storing $\mathbf{z} = \Phi_Q(\mathbf{x})$ and $\tilde{\mathbf{w}}$

Q large \implies **difficult to compute/store**

Model Complexity Price

Q -th order polynomial transform: $\Phi_Q(\mathbf{x}) = \left(\begin{array}{l} 1, \\ x_1, x_2, \dots, x_d, \\ x_1^2, x_1 x_2, \dots, x_d^2, \\ \dots, \\ x_1^Q, x_1^{Q-1} x_2, \dots, x_d^Q \end{array} \right)$

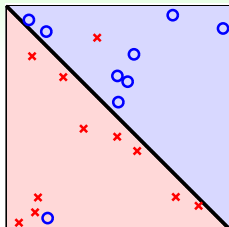
$\underbrace{1}_{\tilde{w}_0} + \underbrace{\tilde{d}}_{\text{others}} \text{ dimensions} = O(Q^d)$

- number of free parameters $\tilde{w}_i = \tilde{d} + 1 \approx d_{VC}(\mathcal{H}_{\Phi_Q})$
- $d_{VC}(\mathcal{H}_{\Phi_Q}) \leq \tilde{d} + 1$, why?

any $\tilde{d} + 2$ inputs not shattered in \mathcal{Z}
 \implies any $\tilde{d} + 2$ inputs not shattered in \mathcal{X}

Q large \implies **large** d_{VC}

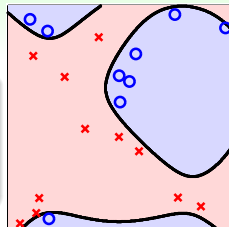
Generalization Issue



Φ_1 (original \mathbf{x})

which one do you prefer? :-)

- Φ_1 'visually' preferred
- Φ_4 : $E_{\text{in}}(g) = 0$ but overkill



Φ_4

- 1 can we make sure that $E_{\text{out}}(g)$ is close enough to $E_{\text{in}}(g)$?
- 2 can we make $E_{\text{in}}(g)$ small enough?

	$\tilde{d}(Q)$	1	2
trade-off:	higher	:- (:- D
	lower	:- D	:- (

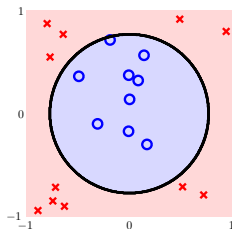
how to pick Q ? **visually**, maybe?

Danger of Visual Choices

first of all, can you really ‘visualize’ when $\mathcal{X} = \mathbb{R}^{10}$? (**well, I can’t :-)**)

Visualize $\mathcal{X} = \mathbb{R}^2$

- full Φ_2 : $\mathbf{z} = (1, x_1, x_2, x_1^2, x_1 x_2, x_2^2)$, $d_{\text{VC}} = 6$
 - or $\mathbf{z} = (1, x_1^2, x_2^2)$, $d_{\text{VC}} = 3$, **after visualizing?**
 - or better $\mathbf{z} = (1, x_1^2 + x_2^2)$, $d_{\text{VC}} = 2$?
 - or even better $\mathbf{z} = (\text{sign}(0.6 - x_1^2 - x_2^2))$?
- careful about **your brain’s ‘model complexity’**



for VC-safety, Φ shall be
decided **without ‘peeking’** data

Fun Time

Consider the Q -th order polynomial transform $\Phi_Q(\mathbf{x})$ for $\mathbf{x} \in \mathbb{R}^2$. Recall that $\tilde{d} = \binom{Q+2}{2} - 1$. When $Q = 50$, what is the value of \tilde{d} ?

- ① 1126
- ② 1325
- ③ 2651
- ④ 6211

Fun Time

Consider the Q -th order polynomial transform $\Phi_Q(\mathbf{x})$ for $\mathbf{x} \in \mathbb{R}^2$. Recall that $\tilde{d} = \binom{Q+2}{2} - 1$. When $Q = 50$, what is the value of \tilde{d} ?

- ① 1126
- ② 1325
- ③ 2651
- ④ 6211

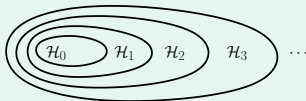
Reference Answer: ②

It's just a simple calculation, but shows you how \tilde{d} becomes hundreds of times of $d = 2$ after the transform.

Polynomial Transform Revisited

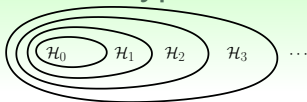
$$\begin{aligned}
 \Phi_0(\mathbf{x}) &= (1), \Phi_1(\mathbf{x}) = (\Phi_0(\mathbf{x}), & x_1, x_2, \dots, x_d) \\
 \Phi_2(\mathbf{x}) &= (\Phi_1(\mathbf{x}), & x_1^2, x_1 x_2, \dots, x_d^2) \\
 \Phi_3(\mathbf{x}) &= (\Phi_2(\mathbf{x}), & x_1^3, x_1^2 x_2, \dots, x_d^3) \\
 &\dots & \dots \\
 \Phi_Q(\mathbf{x}) &= (\Phi_{Q-1}(\mathbf{x}), & x_1^Q, x_1^{Q-1} x_2, \dots, x_d^Q)
 \end{aligned}$$

$$\begin{array}{ccccccccc}
 \mathcal{H}_{\Phi_0} & \subset & \mathcal{H}_{\Phi_1} & \subset & \mathcal{H}_{\Phi_2} & \subset & \mathcal{H}_{\Phi_3} & \subset & \dots & \subset & \mathcal{H}_{\Phi_Q} \\
 \parallel & & \parallel & & \parallel & & \parallel & & & & \parallel \\
 \mathcal{H}_0 & & \mathcal{H}_1 & & \mathcal{H}_2 & & \mathcal{H}_3 & & \dots & & \mathcal{H}_Q
 \end{array}$$



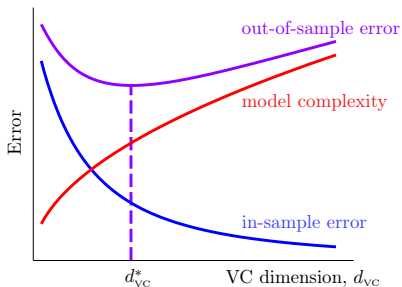
structure: **nested** \mathcal{H}_i

Structured Hypothesis Sets



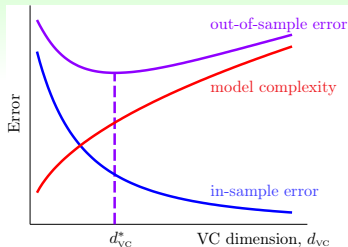
Let $g_i = \operatorname{argmin}_{h \in \mathcal{H}_i} E_{\text{in}}(h)$:

$$\begin{array}{ccccccc}
 \mathcal{H}_0 & \subset & \mathcal{H}_1 & \subset & \mathcal{H}_2 & \subset & \mathcal{H}_3 & \subset & \dots \\
 d_{\text{VC}}(\mathcal{H}_0) & \leq & d_{\text{VC}}(\mathcal{H}_1) & \leq & d_{\text{VC}}(\mathcal{H}_2) & \leq & d_{\text{VC}}(\mathcal{H}_3) & \leq & \dots \\
 E_{\text{in}}(g_0) & \geq & E_{\text{in}}(g_1) & \geq & E_{\text{in}}(g_2) & \geq & E_{\text{in}}(g_3) & \geq & \dots
 \end{array}$$



use \mathcal{H}_{1126} won't be good! :-)

Linear Model First



- tempting sin: use \mathcal{H}_{1126} , low $E_{in}(g_{1126})$ to fool your boss
— **really? :- (a dangerous path of no return**
- safe route: \mathcal{H}_1 first
 - if $E_{in}(g_1)$ good enough, **live happily thereafter :-)**
 - otherwise, move right of the curve
with nothing lost except ‘wasted’ computation

linear model first:
simple, efficient, **safe**, and **workable!**

Fun Time

Consider two hypothesis sets, \mathcal{H}_1 and \mathcal{H}_{1126} , where $\mathcal{H}_1 \subset \mathcal{H}_{1126}$. Which of the following relationship between $d_{\text{VC}}(\mathcal{H}_1)$ and $d_{\text{VC}}(\mathcal{H}_{1126})$ is not possible?

- ① $d_{\text{VC}}(\mathcal{H}_1) = d_{\text{VC}}(\mathcal{H}_{1126})$
- ② $d_{\text{VC}}(\mathcal{H}_1) \neq d_{\text{VC}}(\mathcal{H}_{1126})$
- ③ $d_{\text{VC}}(\mathcal{H}_1) < d_{\text{VC}}(\mathcal{H}_{1126})$
- ④ $d_{\text{VC}}(\mathcal{H}_1) > d_{\text{VC}}(\mathcal{H}_{1126})$

Fun Time

Consider two hypothesis sets, \mathcal{H}_1 and \mathcal{H}_{1126} , where $\mathcal{H}_1 \subset \mathcal{H}_{1126}$. Which of the following relationship between $d_{VC}(\mathcal{H}_1)$ and $d_{VC}(\mathcal{H}_{1126})$ is not possible?

- ① $d_{VC}(\mathcal{H}_1) = d_{VC}(\mathcal{H}_{1126})$
- ② $d_{VC}(\mathcal{H}_1) \neq d_{VC}(\mathcal{H}_{1126})$
- ③ $d_{VC}(\mathcal{H}_1) < d_{VC}(\mathcal{H}_{1126})$
- ④ $d_{VC}(\mathcal{H}_1) > d_{VC}(\mathcal{H}_{1126})$

Reference Answer: ④

Every input combination that \mathcal{H}_1 shatters can be shattered by \mathcal{H}_{1126} , so d_{VC} cannot decrease.

Summary

- 1 When Can Machines Learn?
- 2 Why Can Machines Learn?
- 3 **How** Can Machines Learn?

Lecture 11: Linear Models for Classification

Lecture 12: Nonlinear Transformation

- Quadratic Hypotheses

linear hypotheses on quadratic-transformed data

- Nonlinear Transform

happy linear modeling after $\mathcal{Z} = \Phi(\mathcal{X})$

- Price of Nonlinear Transform

computation/storage/[model complexity]

- Structured Hypothesis Sets

linear/simpler model first

- **next: dark side of the force :-)**

- 4 How Can Machines Learn Better?