Support Vector Machines (SVM)

Krittameth Teachasrisaksakul

บทนำ

จนถึงบัดนี้ เราได้เรียนเกี่ยวกับ machine learning models ที่ค่อนข้างง่ายที่จะวิเคราะห์ และหาค่าที่เหมาะสมที่สุดได้ (optimal) เมื่อ assumptions (สมมติฐาน) ของมันเป็นจริง

ตัวอย่าง เช่น Gaussian Discriminant Analysis (GDA) มีสมมติฐานว่า conditional distribution (การแจกแจงแบบมีเงื่อนไข) $p(x \mid y)$ เป็นแบบ multivariate Gaussian (การแจกแจงแบบปกติหลายตัวแปร)

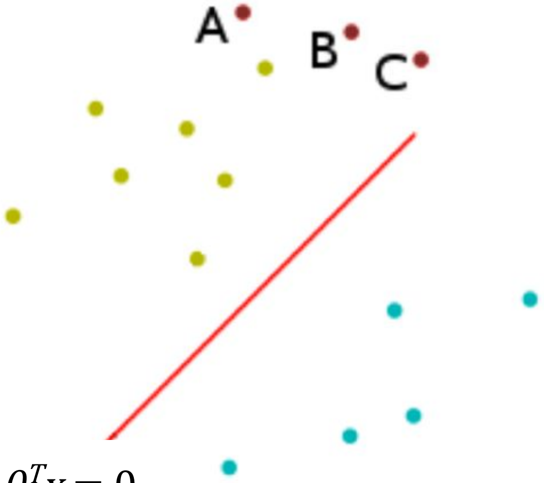
จะเกิดอะไรขึ้นเมื่อ assumptions ถูกละเมิด (ไม่เป็นจริง) ?

เราจะเรียนเกี่ยวกับ support vector machines (SVMs) ซึ่ง flexible มากกว่า และสามารถประยุกต์ใช้ได้อย่างกว้างขวางมากกว่าวิธีที่เรียนไปแล้ว

แม้ว่า deep neural networks ได้รับความสนใจมากที่สุดเมื่อเร็วๆนี้ ผู้คนมากมายยังคงเชื่อว่า SVM เป็น supervised classifiers (ตัวแยกประเภทที่เรียนรู้แบบมีผู้สอน / จากตัวอย่างข้อมูล) ที่หาได้ง่าย / มีพร้อมใช้ (off-the-shelf) ที่ดีที่สุด

บทนำ

SVMs เกิดมาจากแนวคิด maximum margin classification (การแยกประเภทโดยทำให้ margin สูงที่สุด)



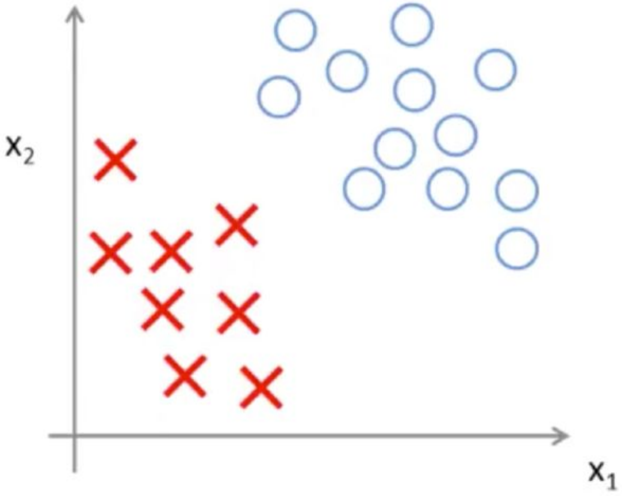
$\theta^T x = 0$

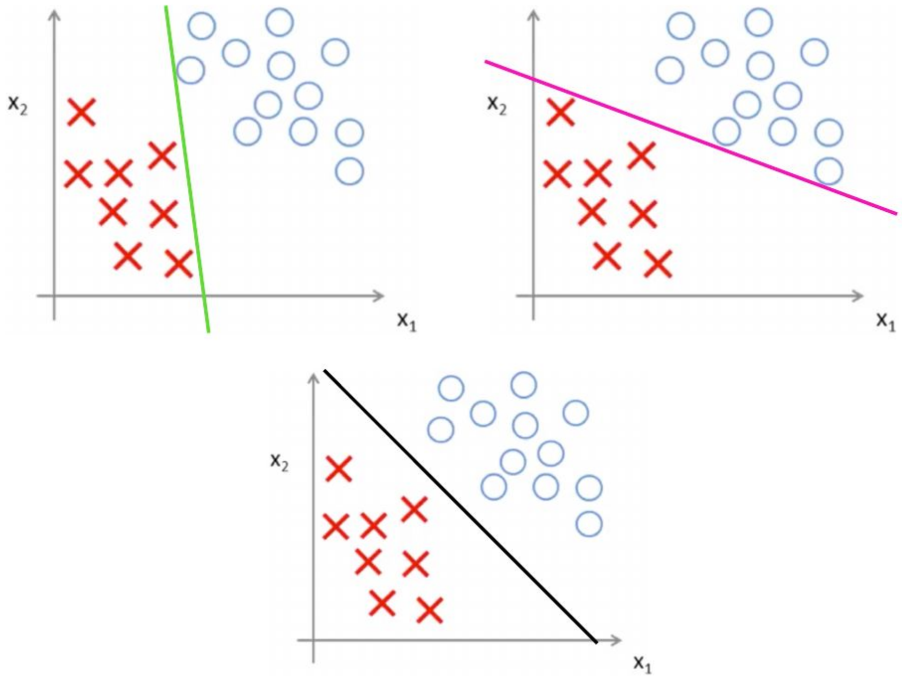
- สมมติ จุดสีเหลืองเป็นข้อมูล training data จาก class 1 และ จุดสีฟ้าเป็นข้อมูล training data จาก class 0
- $\theta^T x = 0$ เป็น ระนาบ hyperplane ที่แบ่งข้อมูลทั้ง 2 class หรือ decision boundary (ขอบเขตตัดสินใจ) ระหว่าง 2 class
- จุด A อยู่ไกลที่สุดจาก decision boundary เราสามารถทำนาย (อย่างมั่นใจ) ว่าจุด A เป็น class 1 แต่จุด C กำกวมมากกว่าว่าจะเป็น class ใด

ข้อสังเกตนี้ ทำให้เกิดหลักการ maximizing the margin (ทำให้ margin สูงที่สุด) !

SVM Decision Boundary: Linearly Separable Case

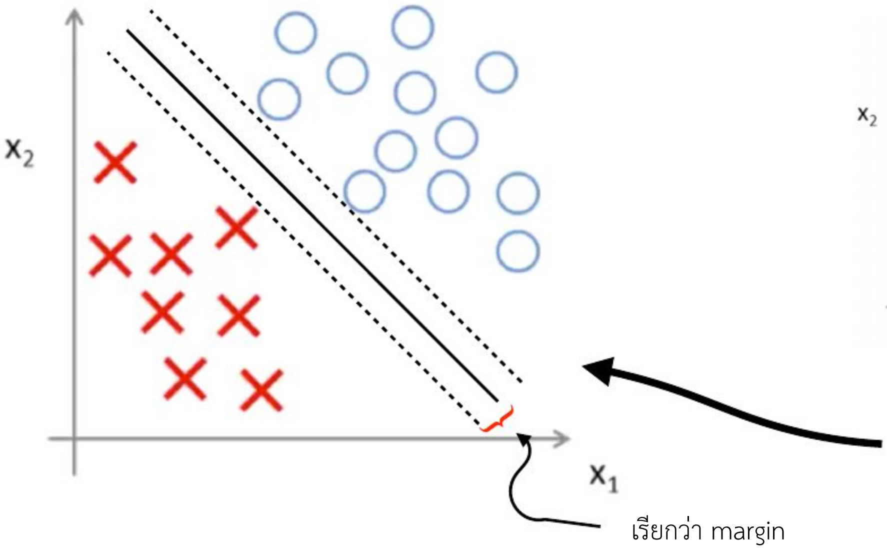
(กรณีที่สามารถแบ่ง class ได้ด้วยขอบเขตตัดสินใจที่เป็นเชิงเส้น)

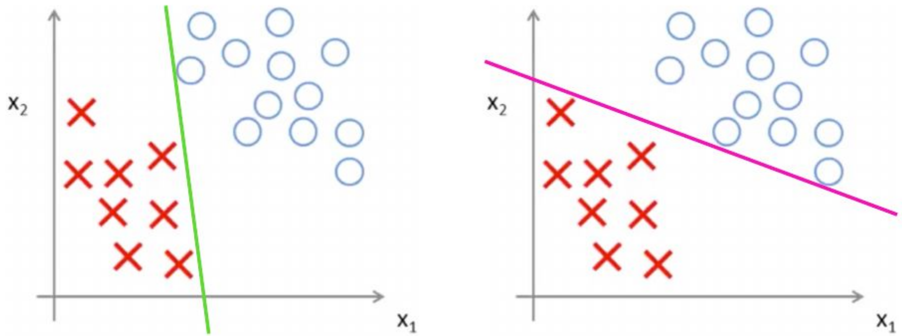


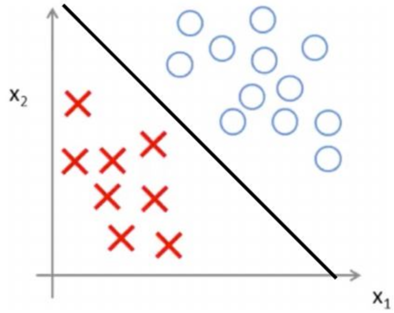


SVM Decision Boundary: Linearly Separable Case

(กรณีที่สามารถแบ่ง class ได้ด้วยขอบเขตตัดสินใจที่เป็นเชิงเส้น)





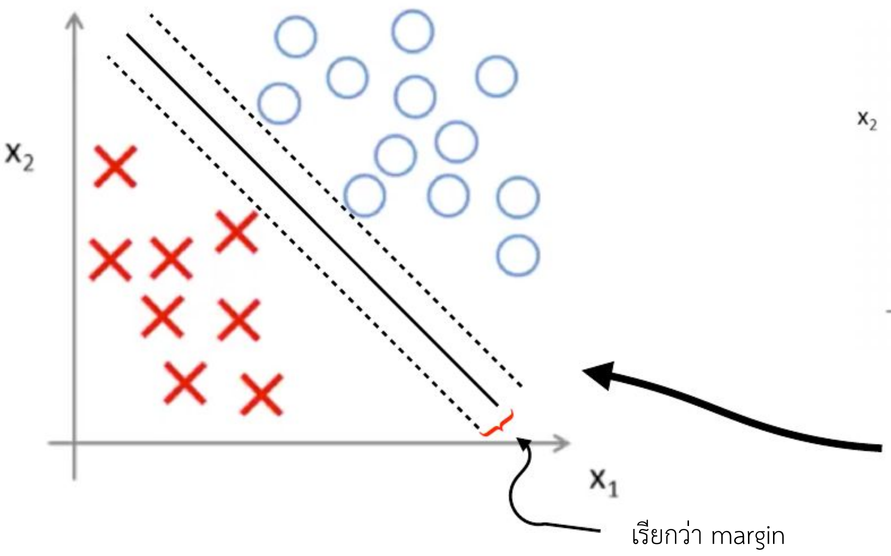


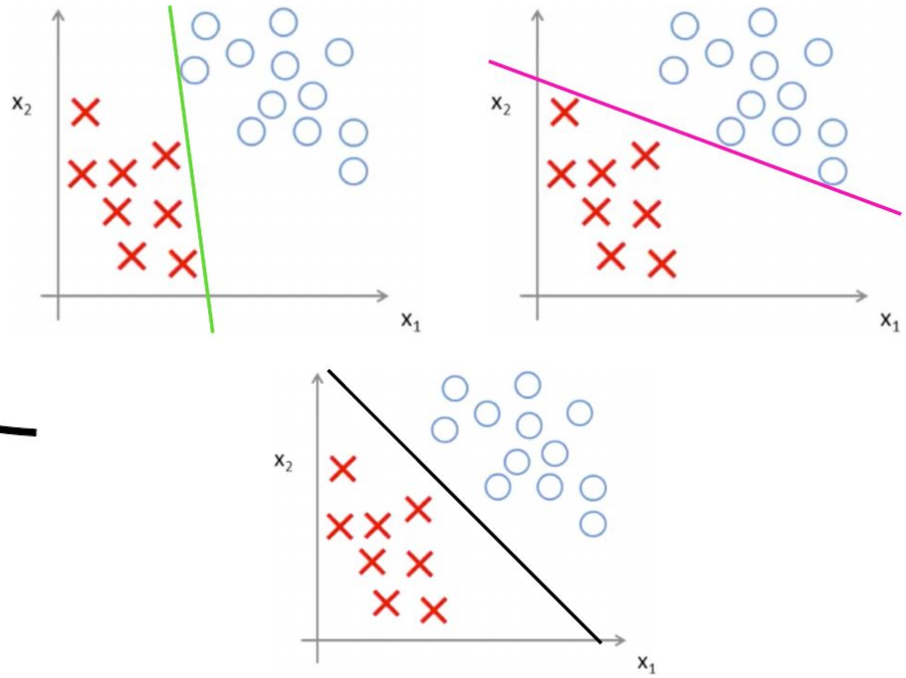
หมายเหตุ: SVM พยายามแบ่งข้อมูลด้วย margin ที่ใหญ่ที่สุดเท่าที่จะเป็นไปได้

(margin = ระยะห่างระหว่าง decision boundary กับจุดข้อมูลที่ใกล้ที่สุด)

ใช้ SVM เป็น Large Margin Classifier

(ตัวแยกประเภทที่มี margin ขนาดใหญ่)





หมายเหตุ: SVM พยายามแบ่งข้อมูลด้วย margin ที่ใหญ่ที่สุดเท่าที่จะเป็นไปได้

(margin = ระยะห่างระหว่าง decision boundary กับจุดข้อมูลที่ใกล้ที่สุด)

Support Vector Machines (SVM)

Optimization Objective

Krittameth Teachasrisaksakul

อีกมุมมองหนึ่งของ Logistic Regression

ทบทวน: Logistic regression เราทำแบบจำลอง (model) ของ $p(y = 1 \mid x; \theta)$ ด้วย

$$h_\theta(x) = g(\boldsymbol{\theta}^T \boldsymbol{x})$$

กฎการแยกประเภทแบบ logistic (logistic classification rule) คือ:

$$y^{\textbf{pred}}(\boldsymbol{x}) = \begin{cases} 1 & \textbf{if } h_\theta(\boldsymbol{x}) \geq 0.5 \\ 0, & \textbf{otherwise} \end{cases}$$

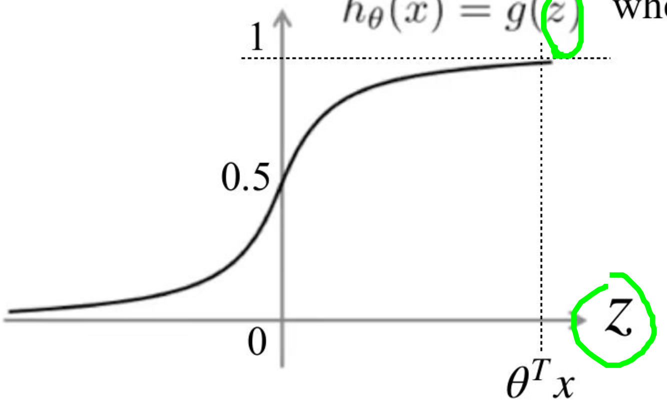
เป้าหมายของเรา ควรเป็น การหา θ ที่ทำให้

- $\theta^T x^{(i)} \gg 0$ สำหรับ i ทุกตัวที่มี $y^{(i)} = 1$ และ
- $\theta^T x^{(i)} \ll 0$ สำหรับ i ทุกตัวที่มี $y^{(i)} = 0$

อีกมุมมองหนึ่งของ Logistic Regression

Hypothesis function:

$$h_\theta(x) = \frac{1}{1 + e^{-\theta^T x}}$$



ปรับ logistic regression เพื่อให้ได้ SVM

sigmoid activation function

ลองคิดเกี่ยวกับ สิ่งที่เราอยากให้ logistic regression ทำ:

- ถ้า $y = 1$ แล้วเราอยากให้ $h_\theta(x) \approx 1$ ก็คือ $\theta^T x \gg 0$ หรือ $z \gg 0$ (มากกว่า 0 มากๆ)
- ถ้า $y = 0$ แล้วเราอยากให้ $h_\theta(x) \approx 0$ ก็คือ $\theta^T x \ll 0$ หรือ $z \ll 0$

อีกมุมมองหนึ่งของ Logistic Regression

Cost ของ example (ตัวอย่าง) $\rightarrow$ $-(y\log h_\theta(x) + (1-y)\log(1-h_\theta(x)))$

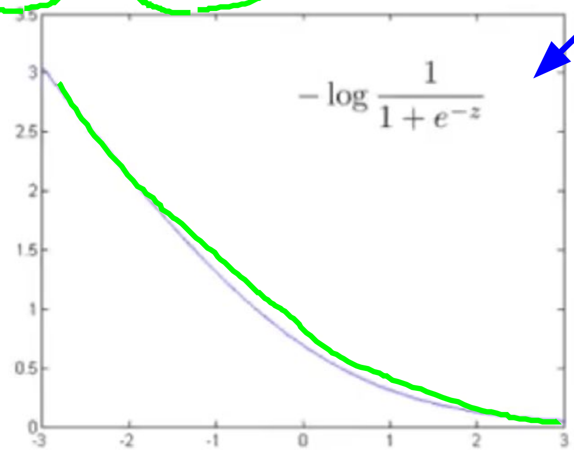
$z = \theta^T x$

$$\Longleftrightarrow \boxed{-y\log\frac{1}{1+e^{-\theta^T x}}} - \cancel{(1-y)\log(1-\frac{1}{1+e^{-\theta^T x}}))}$$

0

กรณี 1 ($y = 1 \Longrightarrow \theta^T x \gg 0$):

แค่พจน์แรกที่มีผล เพราะ พจน์หลัง = 0



in SVM

Plot: function ของ Z vs. Z

ถ้า Z หรือ $\theta^T x$ มาก

$\longrightarrow$ cost function น้อย

เมื่อ logistic regression เจอ positive example (y=1)

มันจะทำให้ $z = \theta^T x$ มีค่าใหญ่ (>> 0) เพราะจะทำให้พจน์นี้ใน cost function มีค่าน้อย

อีกมุมมองหนึ่งของ Logistic Regression

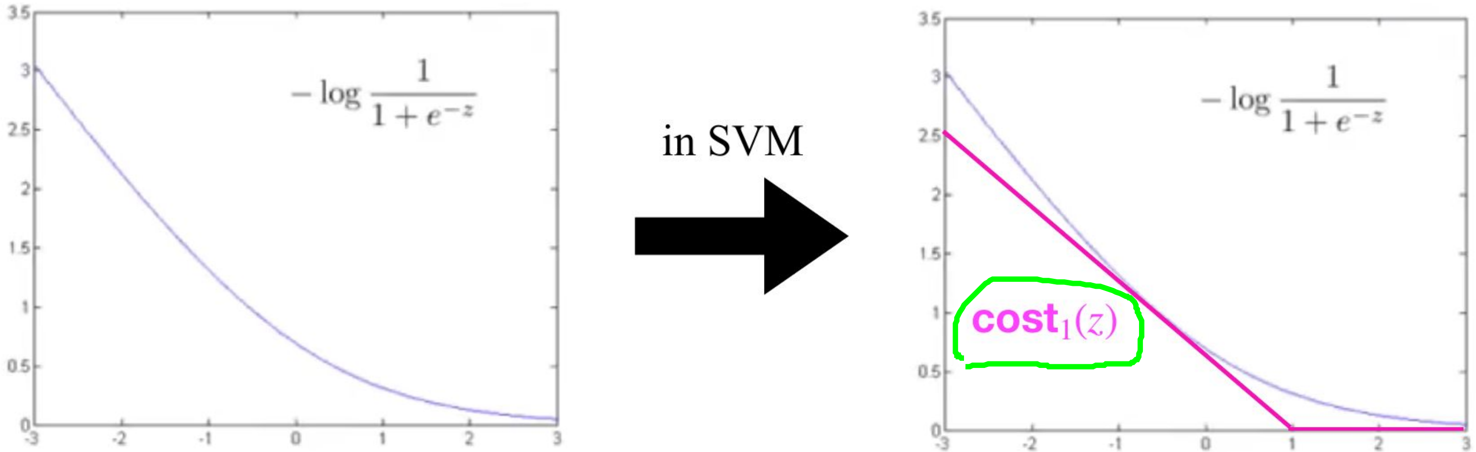
Cost ของ example (ตัวอย่าง)

$$-(y \log h_\theta(x) + (1-y)\log(1-h_\theta(x)))$$

$$z = \theta^T x$$

$$\Longleftrightarrow \boxed{-y \log \frac{1}{1+e^{-\theta^T x}}} - (1-y)\log(1-\frac{1}{1+e^{-\theta^T x}}))$$

กรณี 1 ($y = 1 \Rightarrow \theta^T x \gg 0$):



อีกมุมมองหนึ่งของ Logistic Regression

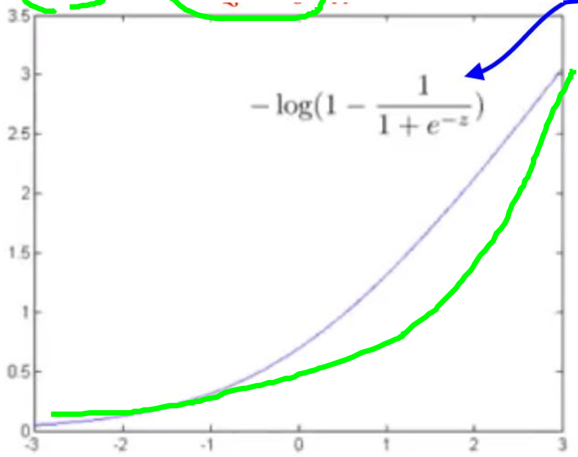
Cost ของ example (ตัวอย่าง)

$$-(y\log h_\theta(x) + (1-y)\log(1-h_\theta(x)))$$

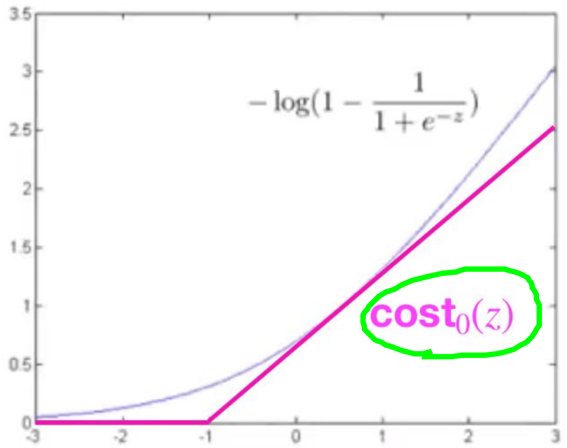
$$\Longleftrightarrow -y\log\frac{1}{1+e^{-\theta^T x}} - \boxed{(1-y)\log(1-\frac{1}{1+e^{-\theta^T x}}))}$$

$z = \theta^T x$

กรณี 2 ($y = 0 \Longrightarrow \theta^T x \ll 0$):



in SVM



Support Vector Machine

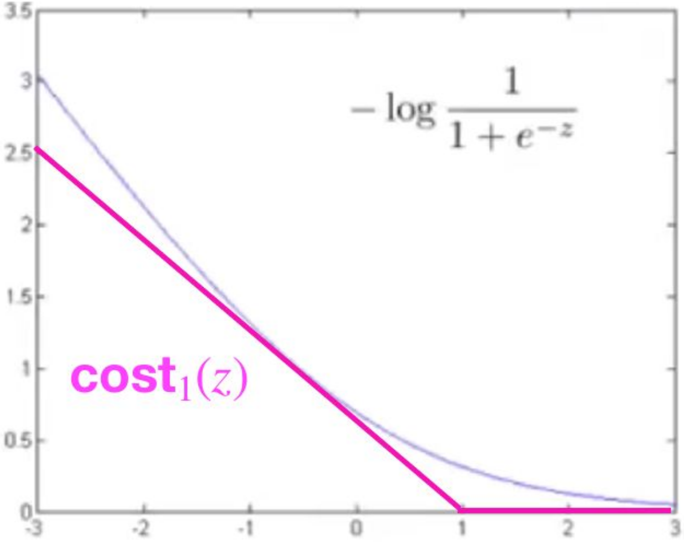
Logistic regression:

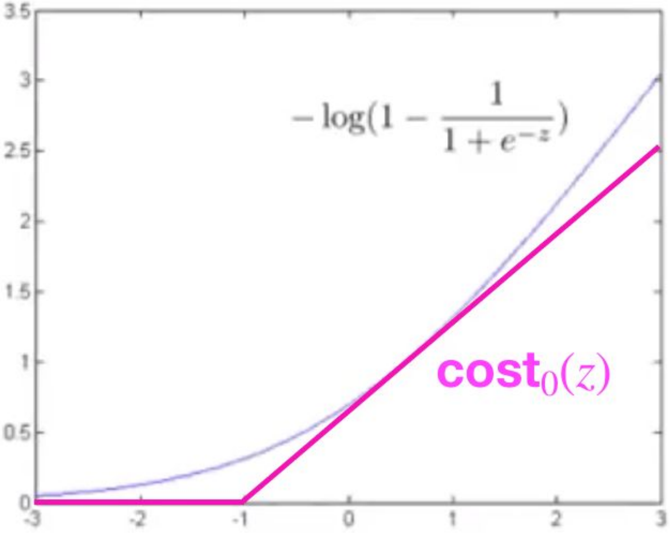
$$J(\theta) = \min_{\theta} \frac{1}{m}\left[\Sigma_{i=1}^{m} y^{(i)}\Big(-\log h_\theta(x^{(i)})\Big) + (1-y^{(i)})\Big(-\log(1-h_\theta(x^{(i)}))\Big)\right] + \frac{\lambda}{2m}\Sigma_{j=1}^{n}\theta_j^2$$

Support Vector Machine

Logistic regression:

$$J(\theta) = \min_{\theta} \frac{1}{m}\left[\Sigma_{i=1}^{m} y^{(i)} \underbrace{\left(-\log h_\theta(x^{(i)})\right)}_{\text{cost}_1(\theta^T x^{(i)})} + (1 - y^{(i)}) \underbrace{\left(-\log(1 - h_\theta(x^{(i)}))\right)}_{\text{cost}_0(\theta^T x^{(i)})}\right] + \frac{\lambda}{2m}\Sigma_{j=1}^{n}\theta_j^2$$





Support Vector Machine

Logistic regression:

$$J(\theta) = \min_{\theta} \frac{1}{m}\left[\Sigma_{i=1}^{m} y^{(i)}\left(-\log h_\theta(x^{(i)})\right) + (1-y^{(i)})\left(-\log(1-h_\theta(x^{(i)}))\right)\right] + \frac{\lambda}{2m}\Sigma_{j=1}^{n}\theta_j^2$$

Support vector machine:

$$J(\theta) = \min_{\theta} \frac{1}{m}\left[\Sigma_{i=1}^{m} y^{(i)}\mathbf{cost}_1(\theta^T x^{(i)}) + (1-y^{(i)})\mathbf{cost}_0(\theta^T x^{(i)})\right] + \frac{\lambda}{2m}\Sigma_{j=1}^{n}\theta_j^2$$

เขียนสมการด้านบนใหม่ เพื่อให้มันสอดคล้องกับ convention ของ SVM

Support Vector Machine

Cost function:

1. เอา 'm' ออกจากสมการ

$$J(\theta) = \min_{\theta} \cancel{\frac{1}{m}} \left[\Sigma_{i=1}^{m} y^{(i)} \textbf{cost}_1(\theta^T x^{(i)}) + (1 - y^{(i)}) \textbf{cost}_0(\theta^T x^{(i)}) \right] + \frac{\lambda}{2\cancel{m}} \Sigma_{j=1}^{n} \theta_j^2$$

ทำไมนี่จึงสมเหตุสมผล ?

$$\frac{dJ}{du} = 0$$

ลองคิดเกี่ยวกับ:

- $\min_u \left((u-5)^2 + 1 \right) \Longrightarrow 2(u-5) \cdot 1 = 0 \Longleftrightarrow u = 5$
- $\min_u 10\left((u-5)^2 + 1 \right) \Longrightarrow 2 \cdot 10 \cdot (u-5) \cdot 1 = 0 \Longrightarrow u = 5$

Support Vector Machine

Cost function:

2. จัดระเบียบสมการเล็กน้อย

$$J(\theta) = \min_{\theta} \cancel{\frac{1}{m}} \underbrace{\left[\Sigma_{i=1}^{m} y^{(i)} \textbf{cost}_1(\theta^T x^{(i)}) + (1 - y^{(i)}) \textbf{cost}_0(\theta^T x^{(i)}) \right]}_{A} + \frac{\lambda}{2\cancel{m}} \underbrace{\Sigma_{j=1}^{n} \theta_j^2}_{B}$$

ก็คือ ใน logistic regression เราเขียน: $A + \lambda B$

ในขณะที่ ใน SVM เราเขียน $CA + B$

สังเกตว่า C ไม่จำเป็นต้องเป็น $1 / \lambda$

Support Vector Machine

Logistic regression:

$$J(\theta) = \min_{\theta} \frac{1}{m}\left[\Sigma_{i=1}^{m} y^{(i)}\Big(-\log h_\theta(x^{(i)})\Big) + (1-y^{(i)})\Big(-\log(1-h_\theta(x^{(i)}))\Big)\right] + \frac{\lambda}{2m}\Sigma_{j=1}^{n}\theta_j^2$$

Support vector machine:

$$J(\theta) = \min_{\theta} C\Sigma_{i=1}^{m}\left[y^{(i)}\textbf{cost}_1(\theta^T x^{(i)}) + (1-y^{(i)})\textbf{cost}_0(\theta^T x^{(i)})\right] + \frac{1}{2}\Sigma_{j=1}^{n}\theta_j^2$$

Question

พิจารณาปัญหา minimization ดังนี้

$$J(\theta) = \min_{\theta} \frac{1}{m}\left[\Sigma_{i=1}^{m} y^{(i)}\mathbf{cost}_1(\theta^T x^{(i)}) + (1 - y^{(i)})\mathbf{cost}_0(\theta^T x^{(i)})\right] + \frac{\lambda}{2m}\Sigma_{j=1}^{n}\theta_j^2$$

$$J(\theta) = \min_{\theta} C\left[\Sigma_{i=1}^{m} y^{(i)}\mathbf{cost}_1(\theta^T x^{(i)}) + (1 - y^{(i)})\mathbf{cost}_0(\theta^T x^{(i)})\right] + \frac{1}{2}\Sigma_{j=1}^{n}\theta_j^2$$

ปัญหา optimization 2 ปัญหานี้ จะให้ค่า θ เดียวกัน (ก็คือ ค่า θ ค่าเดียวกันเป็นคำตอบที่เหมาะสม (optimal solution) ของปัญหาทั้ง 2 ปัญหา

(i) $C = \lambda$ (ii) $C = -\lambda$

(iii) $C = 1/\lambda$ (iv) $C = 2/\lambda$

Support Vector Machine

ไม่เหมือนกับ logistic regression : SVM ไม่มีการตีความในรูปของความน่าจะเป็น (probablistic interpretation)

Cost function:

$$J(\theta) = \min_{\theta} C\Sigma_{i=1}^{m}\left[y^{(i)}\textbf{cost}_1(\theta^T x^{(i)}) + (1 - y^{(i)})\textbf{cost}_0(\theta^T x^{(i)})\right] + \frac{1}{2}\Sigma_{j=1}^{n}\theta_j^2$$

Hypothesis function:

$$h_\theta(x) = \begin{cases} 1, & \textbf{if } \theta^T x \geq 0 \\ 0, & \textbf{otherwise} \end{cases}$$

Support Vector Machines (SVM)

Large Margin Classifier

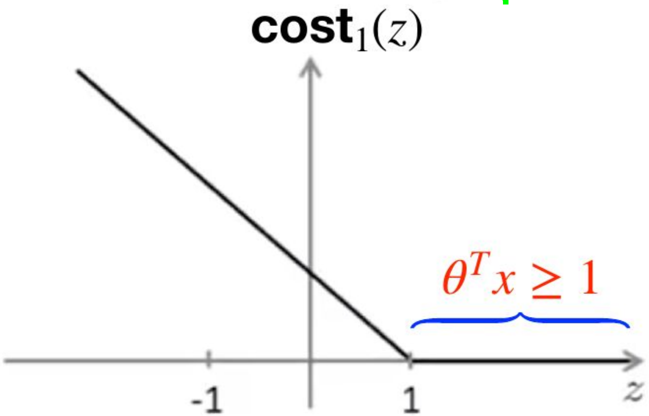
(ตัวแยกประเภทที่มี margin ขนาดใหญ่)

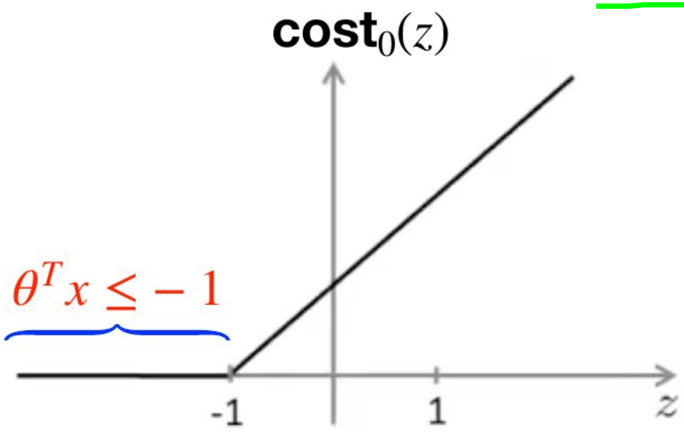
Krittameth Teachasrisaksakul

Figure source: Andrew Ng's Machine Learning course

ความเข้าใจพื้นฐาน

Cost function: $$J(\theta) = \min_{\theta} C\Sigma_{i=1}^{m}\left[y^{(i)}\mathbf{cost}_1(\theta^T x^{(i)}) + (1 - y^{(i)})\mathbf{cost}_0(\theta^T x^{(i)})\right] + \frac{1}{2}\Sigma_{j=1}^{n}\theta_j^2$$



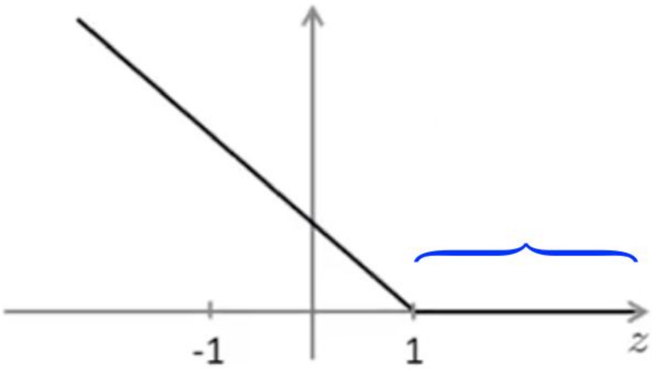


- ถ้า $y = 1$ เราอยากให้ $\theta^T x \geq 1$ (ไม่ใช่แค่ ≥ 0)
- ถ้า $y = 0$ เราอยากให้ $\theta^T x \leq -1$ (ไม่ใช่แค่ < 0)
- **ต่อไป ถ้าเราอยากตั้งค่า C เป็นค่าที่สูงมาก เช่น 100,000 ?**

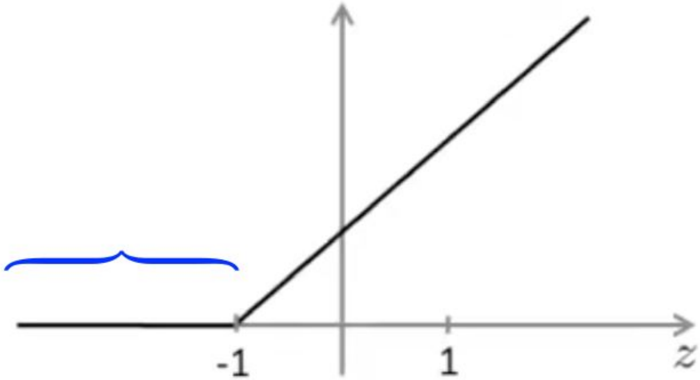
Decision Boundary ของ SVM

$$\min_{\theta} \underbrace{C}_{10^5} \underbrace{\Sigma_{i=1}^{m}\left[y^{(i)}\mathbf{cost}_1(\theta^T x^{(i)}) + (1-y^{(i)})\mathbf{cost}_0(\theta^T x^{(i)})\right]}_{\approx 0} + \frac{1}{2}\Sigma_{j=1}^{n}\theta_j^2$$

เมื่อใดก็ตามที่ $y^{(i)} = 1$ จะได้ $\theta^T x^{(i)} \geq 1$



เมื่อใดก็ตามที่ $y^{(i)} = 0$ จะได้ $\theta^T x^{(i)} \leq -1$



Decision Boundary ของ SVM

$$\underbrace{\min_{\theta} C}_{10^5} \underbrace{\Sigma_{i=1}^{m}\left[y^{(i)}\mathbf{cost}_1(\theta^T x^{(i)}) + (1-y^{(i)})\mathbf{cost}_0(\theta^T x^{(i)})\right]}_{\approx 0} + \frac{1}{2}\Sigma_{j=1}^{n}\theta_j^2$$

เมื่อใดก็ตามที่ $\underline{y^{(i)}} = 1$ จะได้ $\underline{\theta^T x^{(i)} \geq 1}$

เมื่อใดก็ตามที่ $\underline{y^{(i)}} = 0$ จะได้ $\underline{\theta^T x^{(i)} \leq -1}$

$$\min_{\theta} C \cdot 0 + \frac{1}{2}\Sigma_{j=1}^{n}\theta_j^2$$

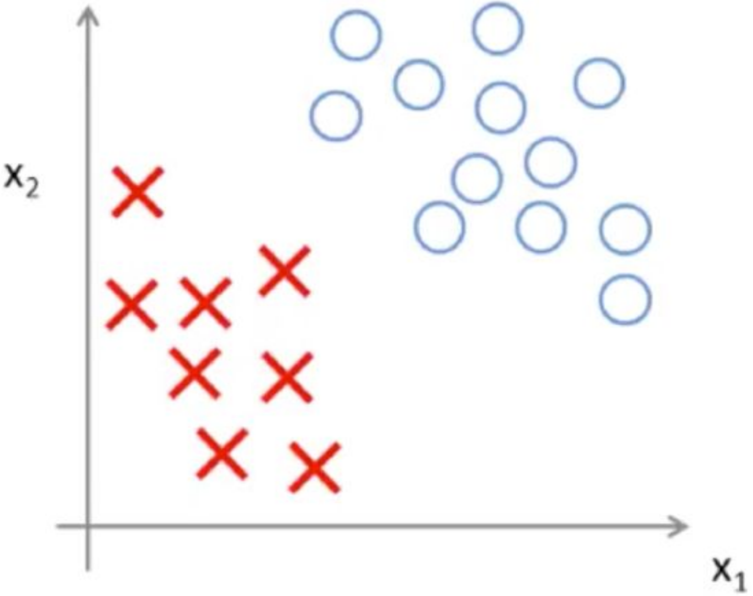
subject to
- $\theta^T x^{(i)} \geq 1$ if $y^{(i)} = 1$
- $\theta^T x^{(i)} \leq -1$ if $y^{(i)} = 0$

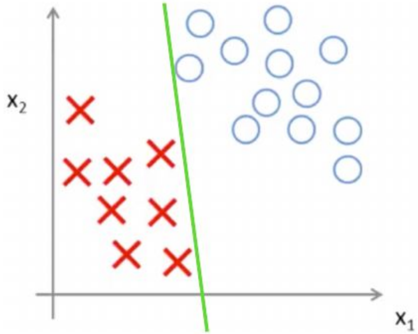
เมื่อแก้ปัญหา optimization นี้
เราจะได้ decision boundary ที่น่าสนใจ !

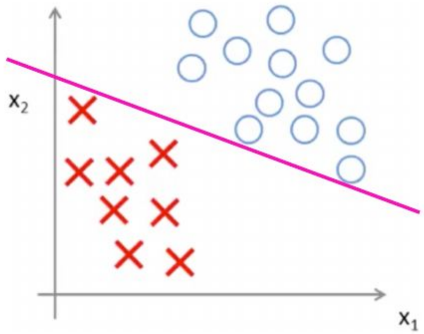
- ก็คือ เมื่อ C สูงมาก $\longrightarrow$ SVM จะอ่อนไหวต่อ outlier (ค่า/ข้อมูลผิดปกติ)
- ลดค่า C จะทำให้ $\longrightarrow$ SVM อ่อนไหว น้อยลง

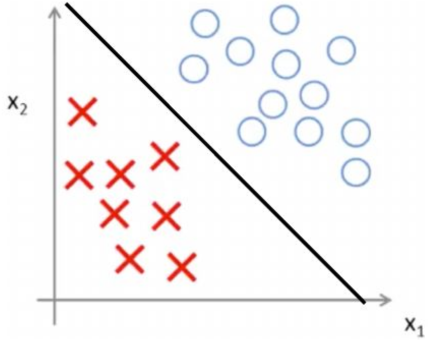
SVM Decision Boundary: Linearly Separable Case

กรณีที่สามารถแบ่ง class ได้ด้วยขอบเขตตัดสินใจที่เป็นเชิงเส้น



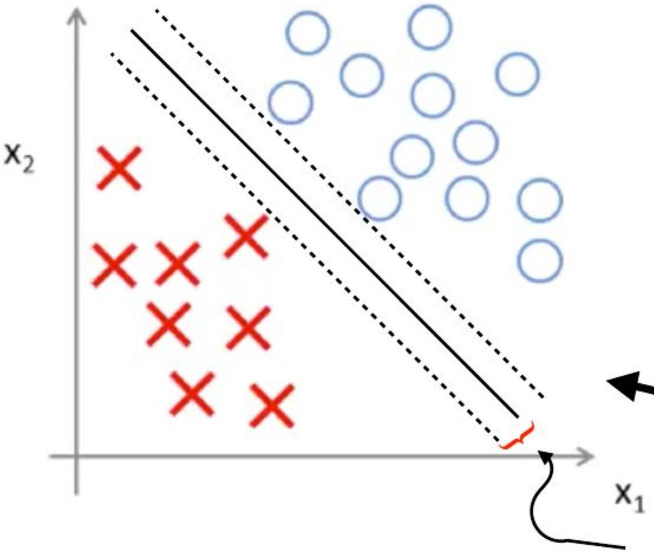


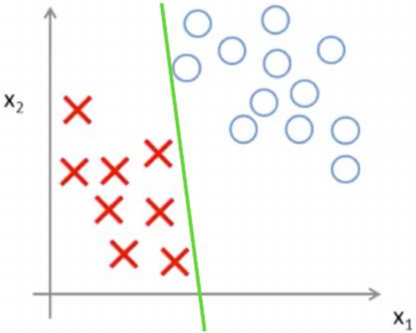


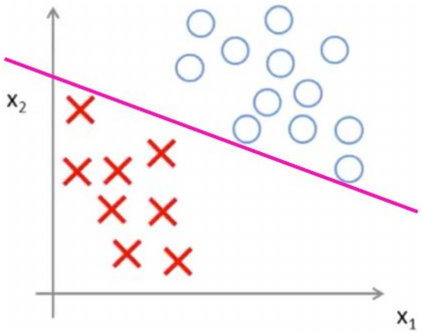


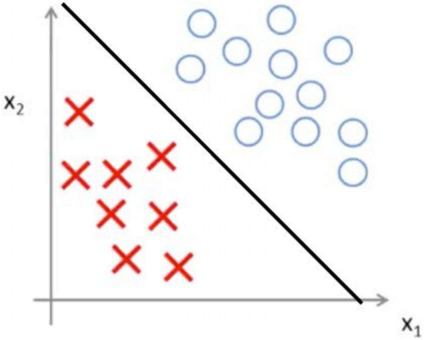
SVM Decision Boundary: Linearly Separable Case

กรณีที่สามารถแบ่ง class ได้ด้วยขอบเขตตัดสินใจที่เป็นเชิงเส้น





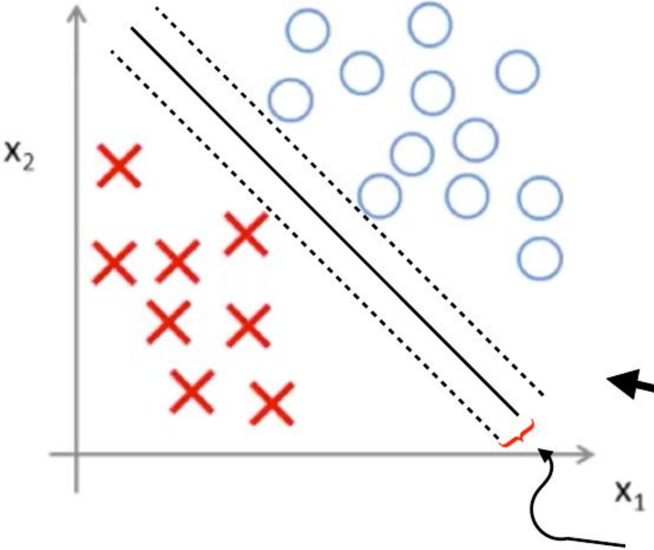


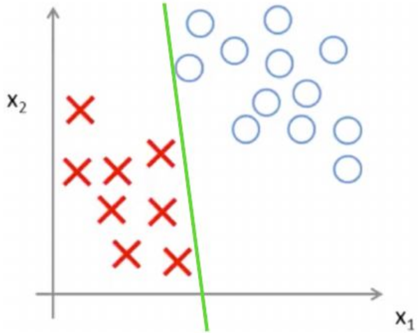


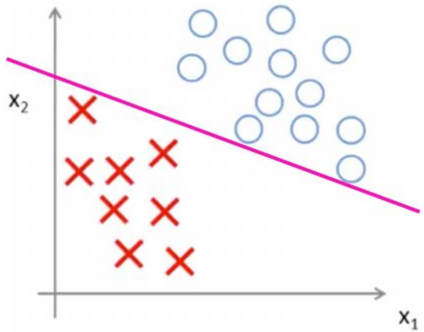
เรียกว่า ‘margin’

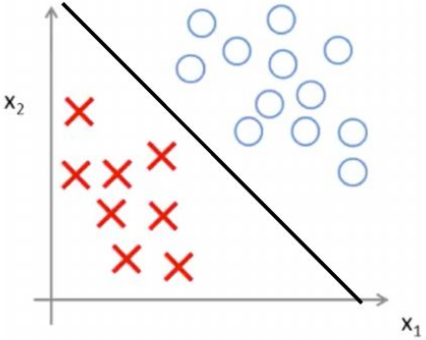
หมายเหตุ: SVM พยายามแบ่งข้อมูลด้วย margin ที่ใหญ่ที่สุดเท่าที่จะเป็นไปได้
margin = ระยะห่างระหว่าง decision boundary กับจุดข้อมูลที่ใกล้ที่สุด

SVM เป็น Large Margin Classifier (ตัวแยกประเภทที่มี margin ขนาดใหญ่)







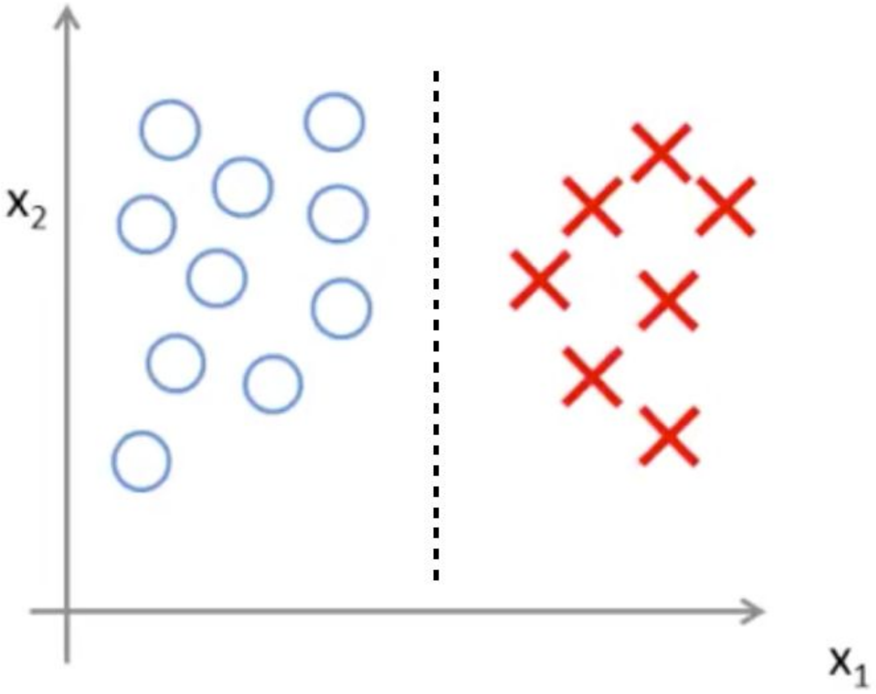


เรียกว่า ‘margin’

หมายเหตุ: SVM พยายามแบ่งข้อมูลด้วย margin ที่ใหญ่ที่สุดเท่าที่จะเป็นไปได้

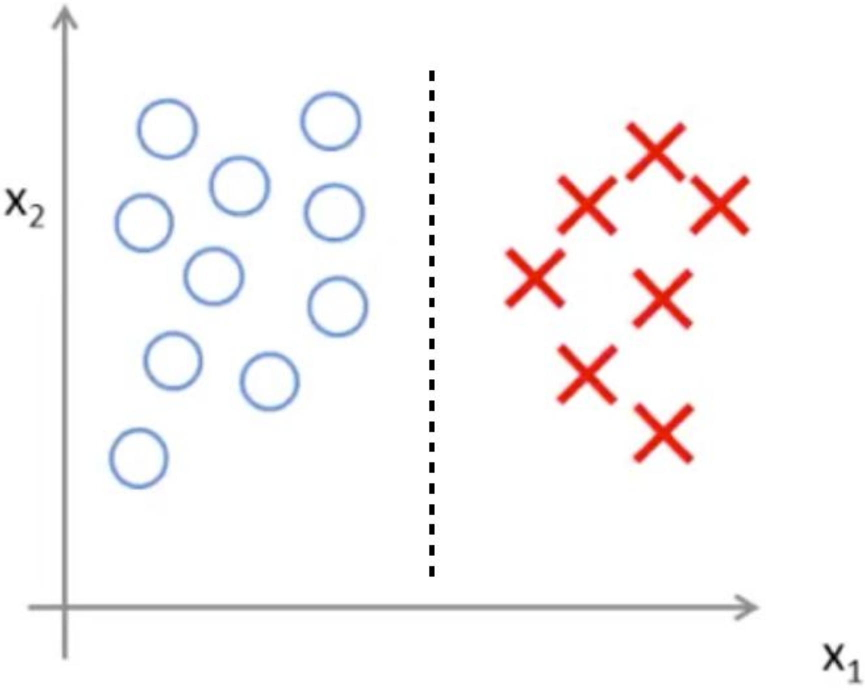
margin = ระยะห่างระหว่าง decision boundary กับจุดข้อมูลที่ใกล้ที่สุด

Large Margin Classifier เมื่อมี Outliers (ค่า/ข้อมูลผิดปกติ)



SVM พยายามหา decision boundary ที่มีระยะห่าง (*margin*) จาก ตัวอย่าง / sample สูงสุด

Large Margin Classifier เมื่อมี Outliers (ค่า/ข้อมูลผิดปกติ)



ลองเพิ่ม outlier 

เมื่อ C สูงมาก : SVM จะอ่อนไหวต่อ outlier

เมื่อลดค่า C จะทำให้ SVM อ่อนไหวน้อยลง

ถ้าข้อมูลไม่ 'linearly sample' (ไม่สามารถแยกชนิด / class ด้วย decision boundary ที่เป็นเส้นตรง)

แล้ว SVM ยังคงทำงานได้ถูกต้อง

Question

พิจารณาชุดข้อมูล training set เมื่อ 'x' แทน ตัวอย่างที่ ($y = 1$) และ 'o' แทน ตัวอย่าง negative ที่ ($y = 0$) สมมติฝึก/สร้าง (train) SVM (ที่ทำนาย 1 เมื่อ $\theta_0 + \theta_1 x_1 + \theta_2 x_2 \geq 0$) SVM อาจจะให้ค่า $\theta_0, \theta_1, \theta_2$?

(i) $\theta_0 = 3,\ \theta_1 = 1,\ \theta_2 = 0$

(ii) $\theta_0 = -3,\ \theta_1 = 1,\ \theta_2 = 0$

(iii) $\theta_0 = 3,\ \theta_1 = 0,\ \theta_2 = 1$

(iv) $\theta_0 = -3,\ \theta_1 = 0,\ \theta_2 = 1$

$x_1 \geq 3$

$-3 + x_1 \geq 0$

Support Vector Machines (SVM)

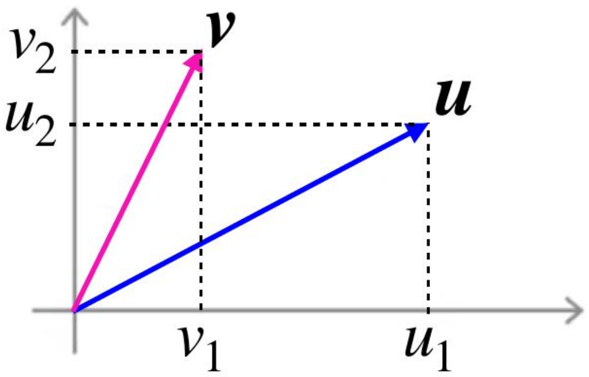
คณิตศาสตร์เบื้องหลัง

Large Margin Classification

Krittameth Teachasrisaksakul

Figure source: Andrew Ng's Machine Learning course

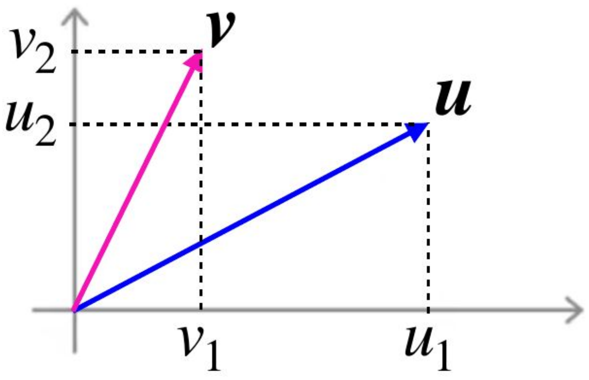
Preliminary: Vector Inner Product



ถ้ามี $\boldsymbol{u} = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$ กับ $\boldsymbol{v} = \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}$ ให้คำนวณ $\boldsymbol{u}^T\boldsymbol{v} = ?$

($\boldsymbol{u}^T\boldsymbol{v}$ เรียกว่า 'vector inner product')

Preliminary: Vector Inner Product



ถ้ามี $\boldsymbol{u} = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$ กับ $\boldsymbol{v} = \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}$ ให้คำนวณ $\boldsymbol{u}^T\boldsymbol{v} = ?$

($\boldsymbol{u}^T\boldsymbol{v}$ เรียกว่า 'vector inner product')

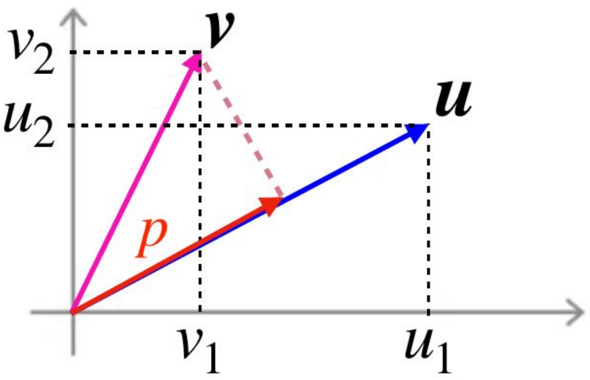
สามารถ quantify (วัดปริมาณ) vector โดยหาค่า euclidean length หรือ norm ของมัน

$$\|\boldsymbol{u}\| = \sqrt{u_1^2 + u_2^2} \quad \text{(by Pythagoras theorem)}$$

$\in$

$\mathbb{R}$

Preliminary: Vector Inner Product



ถ้ามี $\boldsymbol{u} = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$ กับ $\boldsymbol{v} = \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}$ ให้คำนวณ $\boldsymbol{u}^T\boldsymbol{v} = ?$

($\boldsymbol{u}^T\boldsymbol{v}$ เรียกว่า 'vector inner product')

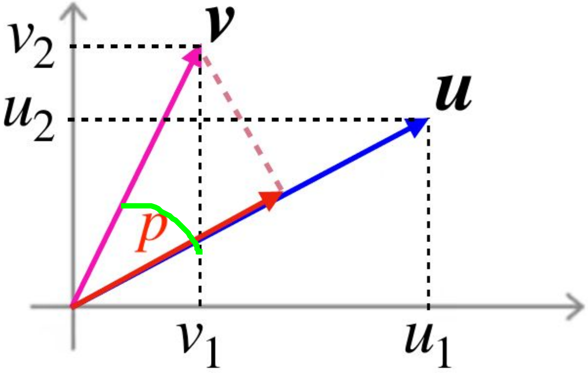
จาก figure :

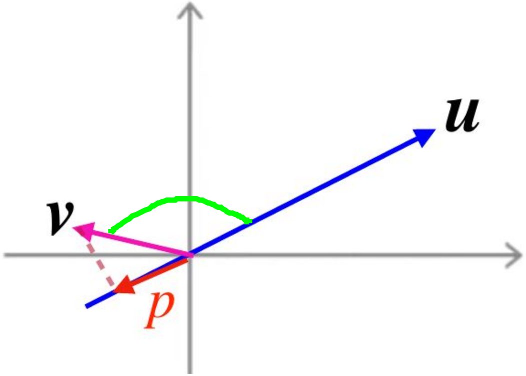
p = ความยาวของ projection (การฉายภาพ) ของ $\boldsymbol{v}$ บน $\boldsymbol{u}$

เป็นไปได้ที่จะแสดงให้เห็นว่า

$$\boldsymbol{u}^T\boldsymbol{v} = p \cdot \|\boldsymbol{u}\| = u_1v_1 + u_2v_2$$
$$= \boldsymbol{v}^T\boldsymbol{u}$$

Preliminary: Vector Inner Product





ถ้ามี $\boldsymbol{u} = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$ กับ $\boldsymbol{v} = \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}$ ให้คำนวณ $\boldsymbol{u}^T\boldsymbol{v} = ?$

($\boldsymbol{u}^T\boldsymbol{v}$ เรียกว่า ‘vector inner product’)

จาก figure :

p = ความยาวของ projection (การฉายภาพ) ของ $\boldsymbol{v}$ บน $\boldsymbol{u}$

เป็นไปได้ที่จะแสดงให้เห็นว่า

$$\boldsymbol{u}^T\boldsymbol{v} = p \cdot \|\boldsymbol{u}\|$$

หมายเหตุ: p เป็นลบ ถ้าขนาดมุมระหว่าง $\boldsymbol{u}$ กับ $\boldsymbol{v} > 90$

Decision Boundary ของ SVM

ทำให้สมการง่ายขึ้น เพื่อที่จะวิเคราะห์มันได้ง่ายขึ้น

- $J(\theta) = \frac{1}{2}\Sigma_{j=1}^{n}\theta_j^2$ s.t. - $\theta^T x^{(i)} \geq 1$ if $y^{(i)} = 1$

  - $\theta^T x^{(i)} \leq -1$ if $y^{(i)} = 0$
- $\theta_0 = 0$
- $n = 2$

Decision Boundary ของ SVM

ทำให้สมการง่ายขึ้น เพื่อที่จะวิเคราะห์มันได้ง่ายขึ้น

- $J(\theta) = \frac{1}{2}\Sigma_{j=1}^{n}\theta_j^2 = \frac{1}{2}(\theta_1^2 + \theta_2^2) = \frac{1}{2}(\underbrace{\sqrt{\theta_1^2 + \theta_2^2}}_{= \|\boldsymbol{\theta}\|})^2$ where $\boldsymbol{\theta} = \begin{bmatrix} \theta_0 \\ \theta_1 \\ \theta_2 \end{bmatrix}$

  s.t. - $\theta^T x^{(i)} \geq 1$ if $y^{(i)} = 1$

  - $\theta^T x^{(i)} \leq -1$ if $y^{(i)} = 0$
- $\theta_0 = 0$
- $n = 2$

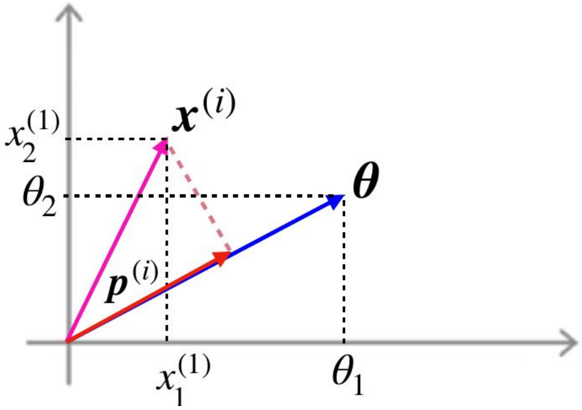
Decision Boundary ของ SVM

- $J(\theta) = \frac{1}{2}\Sigma_{j=1}^{n}\theta_j^2 = \frac{1}{2}(\theta_1^2 + \theta_2^2) = \frac{1}{2}(\sqrt{\theta_1^2 + \theta_2^2})^2 = \frac{1}{2}\|\theta\|^2$

  s.t. $\boldsymbol{\theta}^T\boldsymbol{x}^{(i)} \geq 1$ if $y^{(i)} = 1$

  $\boldsymbol{\theta}^T\boldsymbol{x}^{(i)} \leq -1$ if $y^{(i)} = 0$

- $\theta_0 = 0$
- $n = 2$



$u^T v$

ลองคำนวณ $\boldsymbol{\theta}^T\boldsymbol{x}^{(i)} = ?$

$$\boldsymbol{\theta}^T\boldsymbol{x}^{(i)} = \boldsymbol{p}^{(i)} \cdot \|\boldsymbol{\theta}\|$$
$$= \theta_1 x_1^{(i)} + \theta_2 x_2^{(i)}$$

Decision Boundary ของ SVM

Objective function

ของ SVM จะกลายเป็น

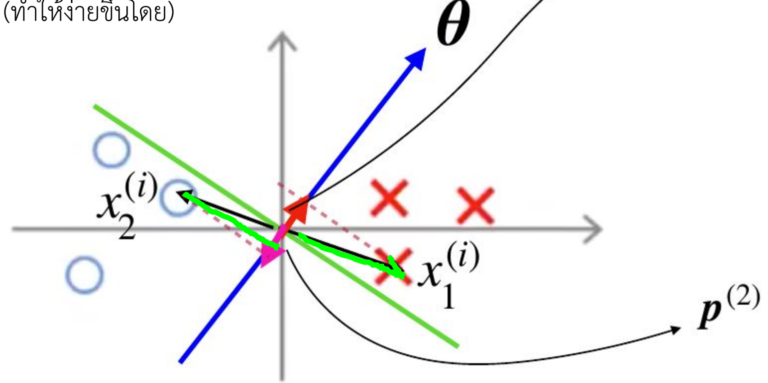
$$\min_{\theta} \frac{1}{2}\Sigma_{j=1}^{n}\theta_j^2 \Longleftrightarrow \min_{\theta} \Sigma_{j=1}^{n}\frac{1}{2}\|\boldsymbol{\theta}\|^2$$

s.t. - $\boldsymbol{p}^{(i)} \cdot \|\boldsymbol{\theta}\| \geq 1 \text{ if } y^{(i)} = 1$

- $\boldsymbol{p}^{(i)} \cdot \|\boldsymbol{\theta}\| \leq -1 \text{ if } y^{(i)} = 0$

Simplification: $\theta_0 = 0$

(ทำให้ง่ายขึ้นโดย)



เมื่อ $p^{(i)}$ เป็น projection ของ $x^{(i)}$ บน vector θ

เพราะ $p^{(1)}$ น้อยมาก

ดังนั้น $||\theta||$ ต้องเยอะมาก

(ทำไม นี่จึงสมเหตุสมผล ?)

เพราะ $p^{(2)}$ น้อยมาก

ดังนั้น $||\theta||$ ต้องเยอะมาก

(ทำไม นี่จึงสมเหตุสมผล ?)

Decision Boundary ของ SVM

Objective function

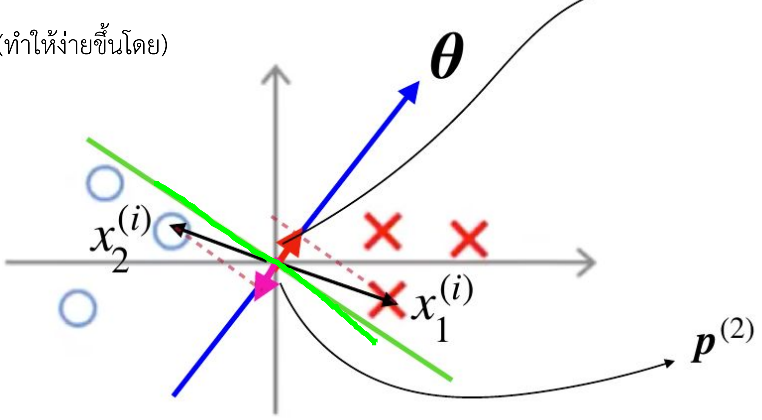
ของ SVM จะกลายเป็น

$$\min_{\theta} \frac{1}{2}\Sigma_{j=1}^{n}\theta_j^2 \iff \min_{\theta} \Sigma_{j=1}^{n}\frac{1}{2}\|\boldsymbol{\theta}\|^2 \quad \text{s.t.} \quad \begin{array}{l} - \ \boldsymbol{p}^{(i)} \cdot \|\boldsymbol{\theta}\| \geq 1 \quad \text{if} \ \ y^{(i)} = 1 \\ - \ \boldsymbol{p}^{(i)} \cdot \|\boldsymbol{\theta}\| \leq -1 \quad \text{if} \ \ y^{(i)} = 0 \end{array}$$

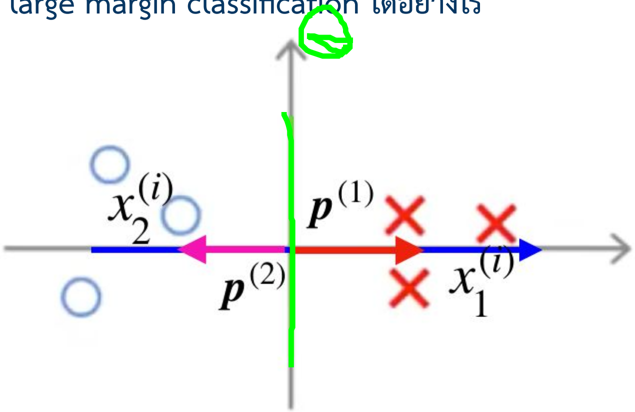
เมื่อ $p^{(i)}$ เป็น projection ของ $x^{(i)}$ บน vector θ

Simplification: $\theta_0 = 0$

(ทำให้ง่ายขึ้นโดย)



นี่อธิบายว่า SVM ทำให้เกิด large margin classification ได้อย่างไร !



Decision Boundary ของ SVM

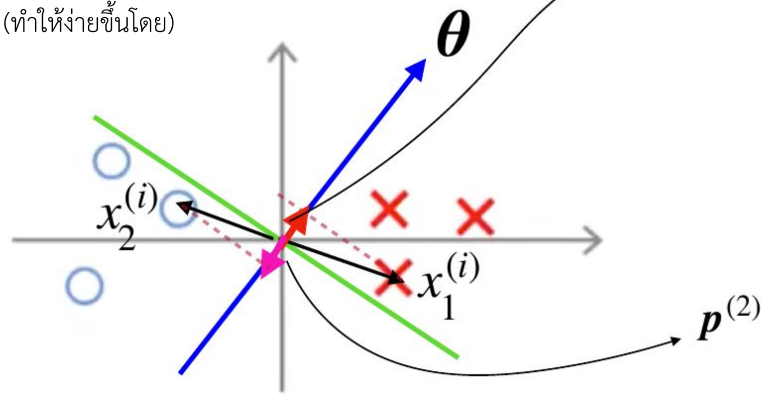
Objective function
ของ SVM จะกลายเป็น

$$\min_{\theta} \frac{1}{2}\Sigma_{j=1}^{n}\theta_j^2 \Longleftrightarrow \min_{\theta} \Sigma_{j=1}^{n}\frac{1}{2}\|\boldsymbol{\theta}\|^2 \quad \text{s.t.} \quad \begin{array}{l} -\ \boldsymbol{p}^{(i)} \cdot \|\boldsymbol{\theta}\| \geq 1 \ \text{ if } \ y^{(i)} = 1 \\ -\ \boldsymbol{p}^{(i)} \cdot \|\boldsymbol{\theta}\| \leq -1 \ \text{ if } \ y^{(i)} = 0 \end{array}$$

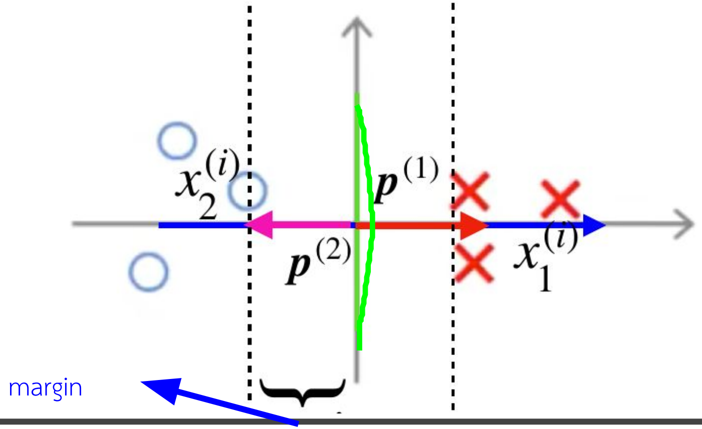
เมื่อ $p^{(i)}$ เป็น projection ของ $x^{(i)}$ บน vector θ

Simplification: $\theta_0 = 0$

(ทำให้ง่ายขึ้นโดย)



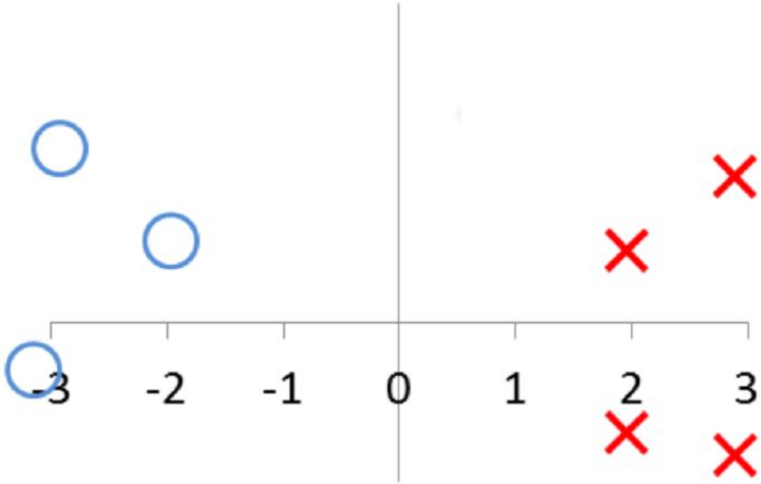
นี่อธิบายว่า SVM ทำให้เกิด large margin classification ได้อย่างไร !



Question

ปัญหา SVM optimization ที่เราใช้ คือ:

$$\min_{\theta} \frac{1}{2}\Sigma_{j=1}^{n}\theta_j^2 \text{ s.t. } \quad \begin{array}{l} - \ \boldsymbol{p}^{(i)} \cdot \|\boldsymbol{\theta}\| \geq 1 \quad \text{if } y^{(i)} = 1 \\ - \ \boldsymbol{p}^{(i)} \cdot \|\boldsymbol{\theta}\| \leq -1 \quad \text{if } y^{(i)} = 0 \end{array}$$



เมื่อ $\boldsymbol{p}^{(i)}$ เป็น projection (signed / ที่มีเครื่องหมาย) ของ $\boldsymbol{x}^{(i)}$ ลงบน $\boldsymbol{\theta}$ พิจารณาชุดข้อมูล training set ด้านขวา ที่ค่าที่เหมาะสม (optimal value) ของ $\boldsymbol{\theta}$ ค่าของ $\|\boldsymbol{\theta}\|$ จะเป็นเท่าไร?

(i) 1 / 4 (ii) 1 / 2
(iii) 1 (iv) 2

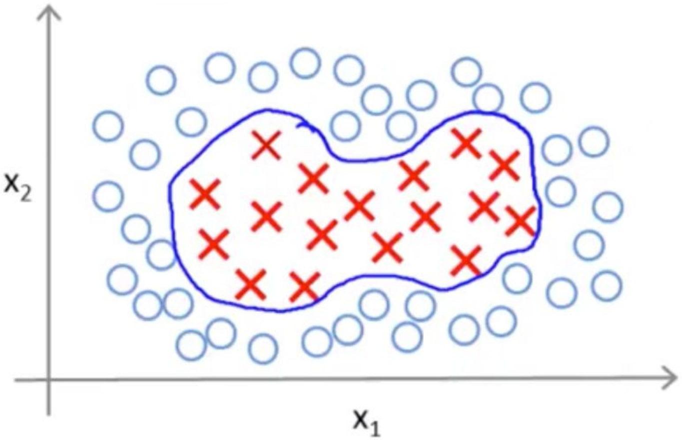
Support Vector Machines (SVM)

Kernels (Part 1)

Krittameth Teachasrisaksakul

ความเข้าใจพื้นฐาน

ขอบเขตตัดสินใจที่ไม่เป็นเชิงเส้น (Non-linear decision boundary)



ทำนาย $y = 1$ ถ้า

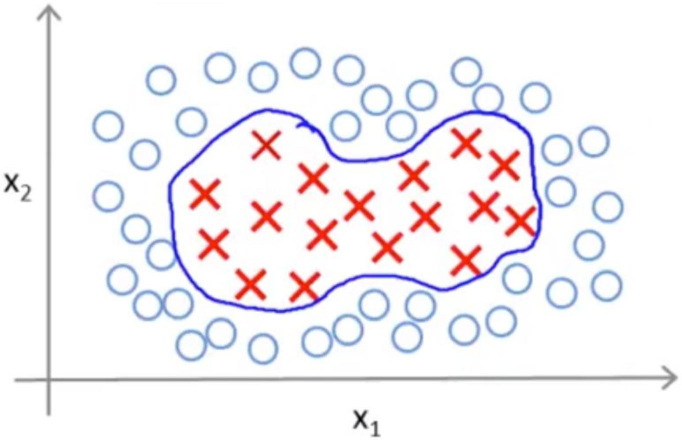
$$\theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_1 x_2 + \theta_4 x_1^2 + \theta_5 x_2^2 + \ldots \geq 0$$

ก็คือ

$$h_\theta(x) = \begin{cases} 1, & \textbf{if } \theta_0 + \theta_1 x_1 + \ldots \geq 0 \\ 0, & \textbf{otherwise} \end{cases}$$

ความเข้าใจพื้นฐาน

ขอบเขตตัดสินใจที่ไม่เป็นเชิงเส้น (Non-linear decision boundary)



ทำนาย $y = 1$ ถ้า

$$\theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_1 x_2 + \theta_4 x_1^2 + \theta_5 x_2^2 + \ldots \geq 0$$

ก็คือ

$$h_\theta(x) = \begin{cases} 1, & \textbf{if } \theta_0 + \theta_1 x_1 + \ldots \geq 0 \\ 0, & \textbf{otherwise} \end{cases}$$

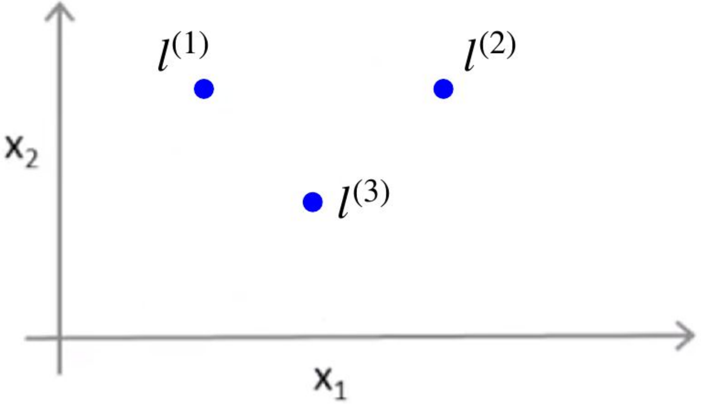
สัญลักษณ์ใหม่ (New notation): $\theta_0 + \theta_1 f_1 + \theta_2 f_2 + \ldots$ ature

เช่น $f_1 = x_1, f_2 = x_2, f_3 = x_1 x_2, f_4 = x_1^2, f_5 = x_2^2, \ldots$

คำถาม: มีตัวเลือก features f_1, f_2, f_3 ที่ต่างไป หรือ ดีกว่าหรือไม่?

Kernels : ความเข้าใจพื้นฐาน

นี่คือ แนวคิดใหม่ในการนิยาม feature ใหม่ f_1, f_2, f_3

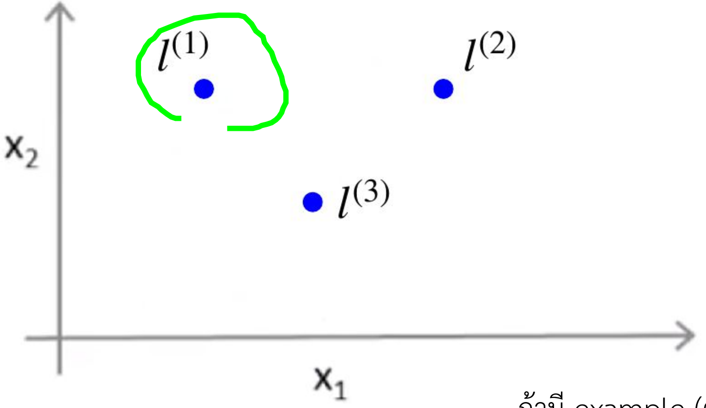


ถ้ามี x คำนวณ feature ใหม่ ที่ขึ้นอยู่กับ proximity (ความใกล้ชิด) ของ landmarks $l^{(1)}, l^{(2)}, l^{(3)}$

Figure source: Andrew Ng's Machine Learning course

Kernels : ความเข้าใจพื้นฐาน

นี่คือ แนวคิดใหม่ในการนิยาม feature ใหม่ f_1, f_2, f_3



ถ้ามี x คำนวณ feature ใหม่ ที่ขึ้นอยู่กับ proximity (ความใกล้ชิด) ของ landmarks $l^{(1)}, l^{(2)}, l^{(3)}$

e

ถ้ามี example (ตัวอย่าง) x $\longrightarrow$

$$f_1 := \textbf{similarity}(x, l^{(1)}) = \exp\left(-\frac{\|x - l^{(1)}\|^2}{2\sigma^2}\right)$$

$$f_2 := \textbf{similarity}(x, l^{(2)}) = \exp\left(-\frac{\|x - l^{(2)}\|^2}{2\sigma^2}\right)$$

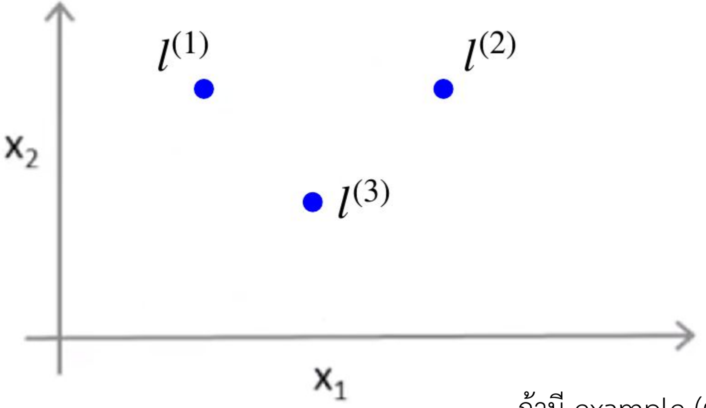
$$\vdots$$

kernels

(Gaussian kernels)

Kernels : ความเข้าใจพื้นฐาน

นี่คือ แนวคิดใหม่ในการนิยาม feature ใหม่ f_1, f_2, f_3



ถ้ามี x คำนวณ feature ใหม่ ที่ขึ้นอยู่กับ proximity (ความใกล้ชิด) ของ landmarks $l^{(1)}, l^{(2)}, l^{(3)}$

ถ้ามี example (ตัวอย่าง) x ⟶

$$f_1 := \textbf{similarity}(x, l^{(1)}) = \exp\left(-\frac{\|x - l^{(1)}\|^2}{2\sigma^2}\right)$$

$k(x, l^{(i)})$

$$f_2 := \underline{\textbf{similarity}}(x, l^{(2)}) = \exp\left(-\frac{\|x - l^{(2)}\|^2}{2\sigma^2}\right)$$

$$\vdots$$

kernels

(Gaussian kernels)

Kernels และ Similarity

$$f_1 := \textbf{similarity}(x, l^{(1)}) = \exp(-\frac{\|x - l^{(1)}\|^2}{2\sigma^2}),$$

$$\text{where } \|x - l\|^2 = \Sigma_{j=1}^{n}(x_j - l_j)^2$$

ถ้า $x \approx l^{(1)}$ แล้ว

ถ้า x อยู่ไกลจาก $l^{(1)}$ แล้ว

Kernels และ Similarity

$$f_1 := \textbf{similarity}(x, l^{(1)}) = \exp\left(-\frac{\|x - l^{(1)}\|^2}{2\sigma^2}\right),$$

$$\text{where } \|x - l\|^2 = \Sigma_{j=1}^{n}(x_j - l_j)^2$$

ถ้า $x \approx l^{(1)}$ แล้ว

$$f_1 \approx \exp\left(-\frac{0^2}{2\sigma^2}\right) \approx 1$$

ถ้า x อยู่ไกลจาก $l^{(1)}$ แล้ว

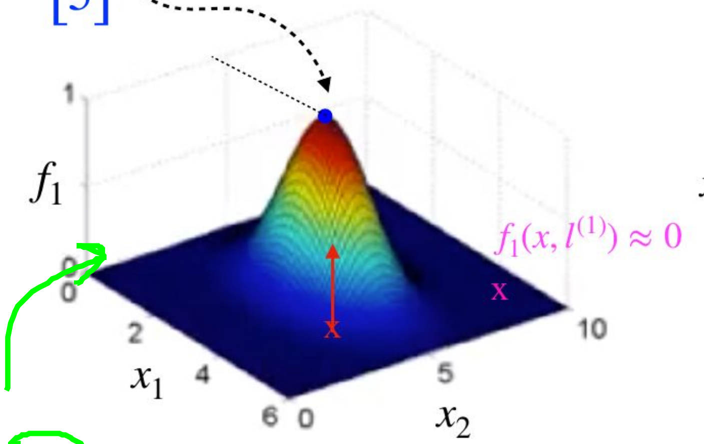
$$f_1 \approx \exp\left(-\frac{(\textbf{large number})^2}{2\sigma^2}\right) \approx 0$$

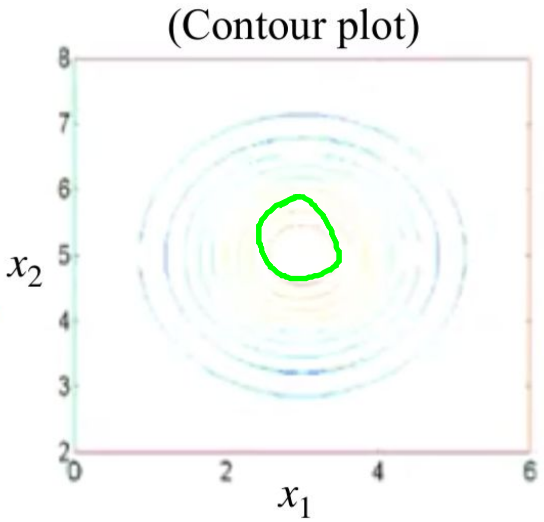
สังเกตว่า landmark แต่ละจุด ทำให้เกิด (นิยาม) feature ใหม่

Kernels และ Similarity

ตัวอย่าง: ถ้ามี $l^{(1)} = \begin{bmatrix} 3 \\ 5 \end{bmatrix}$ $f_1 = \exp\left(- \frac{\|x - l^{(1)}\|^2}{2\sigma^2} \right)$

$x = \begin{bmatrix} 3 \\ 5 \end{bmatrix}$ $\sigma^2 = 1$





ต่อไป ลองดูผลของ σ^2 (ซึ่งเป็น parameter ของ Gaussian kernel)

Figure source: Andrew Ng's Machine Learning course

ตัวอย่าง: ถ้ามี $l^{(1)} = \begin{bmatrix} 3 \\ 5 \end{bmatrix}$ $f_1 = \exp\left(- \frac{\|x - l^{(1)}\|^2}{2\sigma^2} \right)$

Kernels และ Similarity

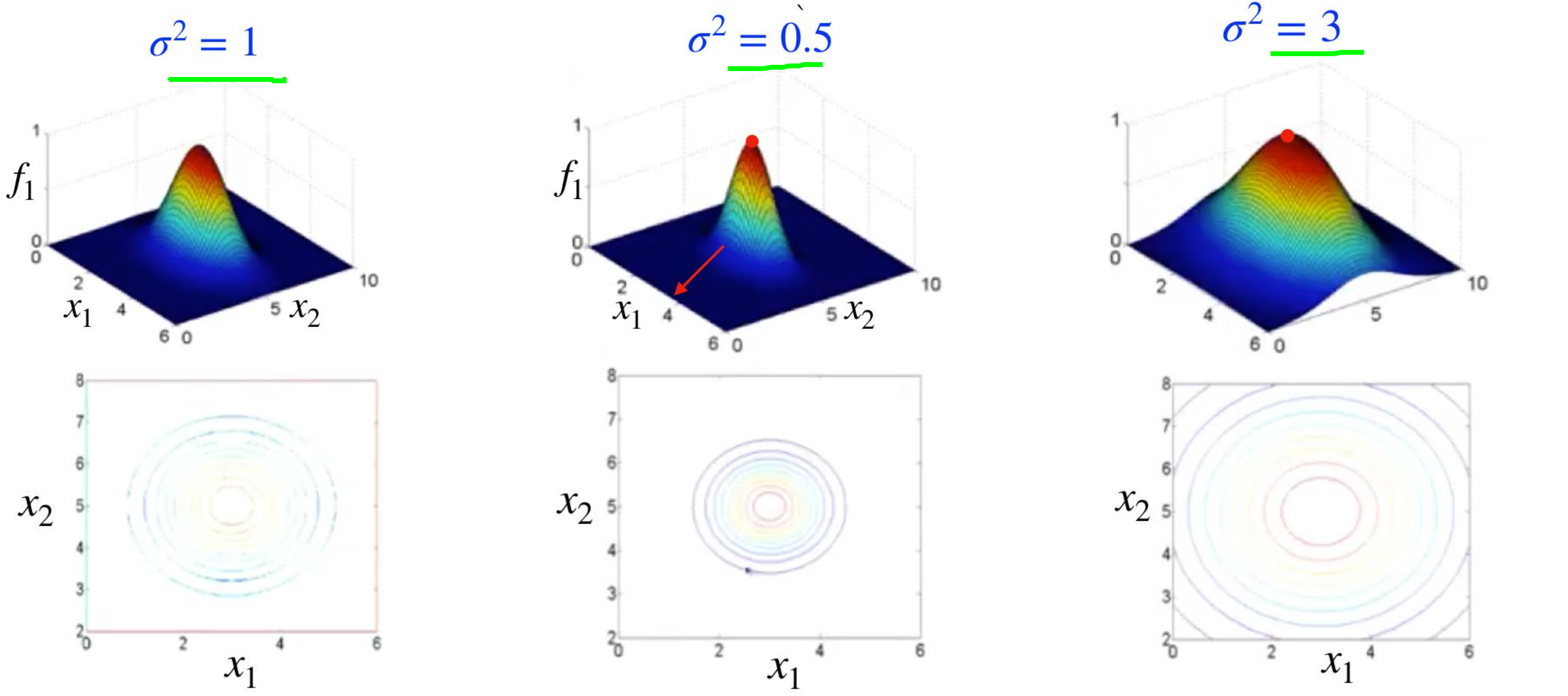
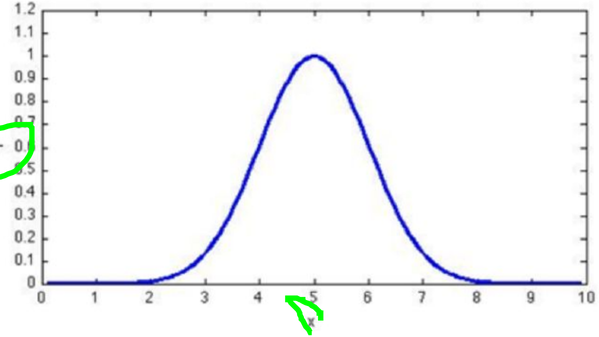


Figure source: Andrew Ng's Machine Learning course



Question

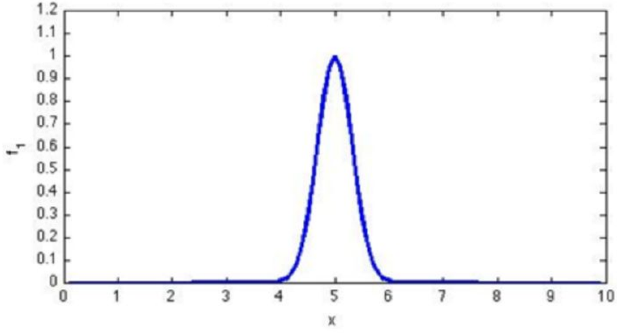
พิจารณาตัวอย่าง 1 มิติ (1-D example) ที่มี 1 feature x_1 และสมมติ $l^{(1)} = 5$

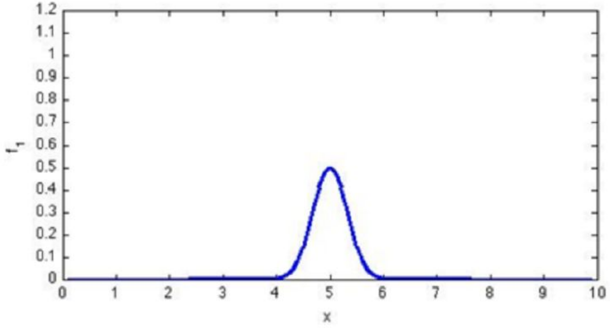
ภาพด้านบน คือ plot ของ

$$f_1 = \exp\left(-\frac{\|x_1 - l^{(1)}\|^2}{2\sigma^2}\right)^2 = 1$$

$\sigma^2 = 1$

สมมติ เราเปลี่ยนให้ $\sigma^2 = 4$ ภาพใดที่เป็น plot ของ f_1 ที่มีค่า σ^2 ค่าใหม่ ?





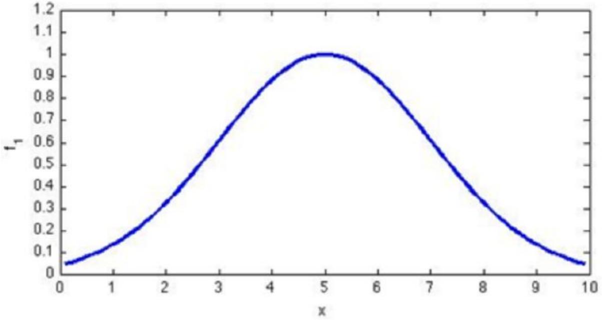
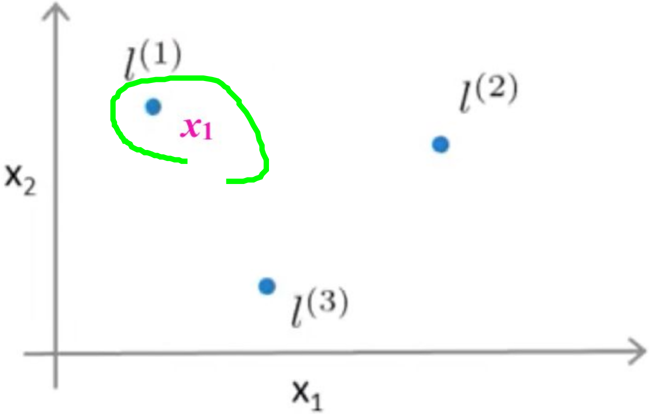


Figure source: Andrew Ng's Machine Learning course

Kernels และ Similarity

ถ้ามี features (คุณลักษณะ) เหล่านี้ ลองดูว่าจะเรียนรู้ hypothesis function อะไรได้



Hypothesis function:

ทำนาย '1' เมื่อ $\theta_0 + \theta_1 f_1 + \theta_2 f_2 + \theta_3 f_3 \geq 0$

ถ้ามี example x_1

เราจะคำนวณ: f_1, f_2, f_3

สมมติ เราทำนาย

$$\theta_0 = -0.5,\ \theta_1 = 1,\ \theta_2 = 1,\ \theta_3 = 0$$

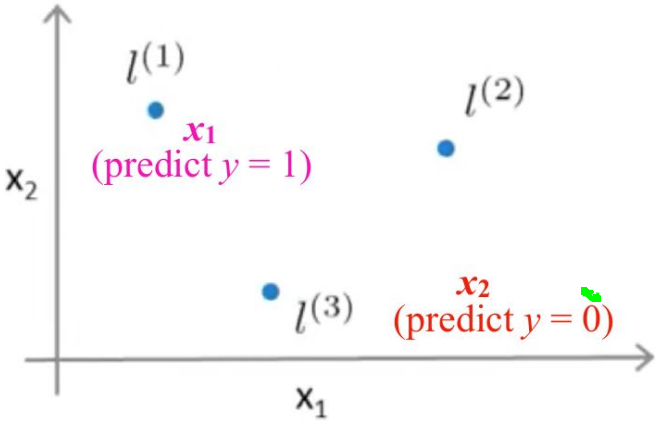
$\because f_1 \approx 1,\ f_2 \approx 0,\ f_3 \approx 0$

ก็คือ $\theta_0 + \theta_1 \cdot 1 + \theta_2 \cdot 0 + \theta_3 \cdot 0 = -0.5 + 1 = 0.5 \geq 0$

$\therefore$ **ทำนาย $y = 1$ สำหรับ x_1**

Kernels และ Similarity

ถ้ามี features (คุณลักษณะ) เหล่านี้ ลองดูว่าจะเรียนรู้ hypothesis function อะไรได้



Hypothesis function:

ทำนาย '1' เมื่อ $\theta_0 + \theta_1 f_1 + \theta_2 f_2 + \theta_3 f_3 \geq 0$

ถ้ามี example $\boldsymbol{x_2}$

เราจะคำนวณ: f_1, f_2, f_3

สมมติ เราทำนาย

$$\theta_0 = -0.5,\ \theta_1 = 1,\ \theta_2 = 1,\ \theta_3 = 0$$

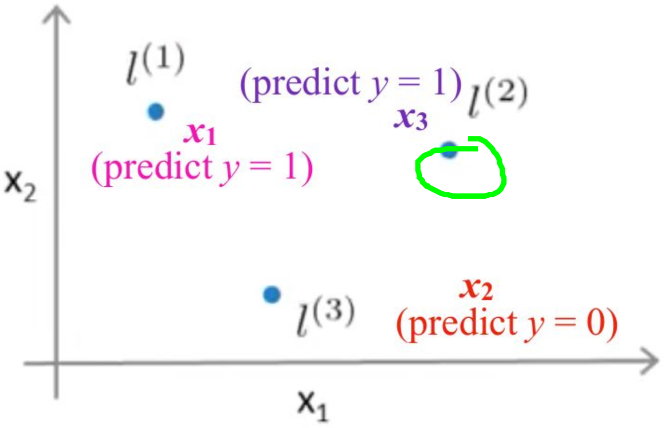
$\because f_1 \approx 0,\ f_2 \approx 0,\ f_3 \approx 0$

ก็คือ $\theta_0 + \theta_1 \cdot 1 + \theta_2 \cdot 0 + \theta_3 \cdot 0 = -0.5 < 0$

$\therefore$ **ทำนาย $\boldsymbol{y = 0}$ สำหรับ $\boldsymbol{x_2}$**

Kernels และ Similarity

ถ้ามี features (คุณลักษณะ) เหล่านี้ ลองดูว่าจะเรียนรู้ hypothesis function อะไรได้



$$\because f_1 \approx 0,\ f_2 \approx 0,\ f_3 \approx 0$$

ก็คือ $\theta_0 + \theta_1 \cdot 1 + \theta_2 \cdot 0 + \theta_3 \cdot 0 = -0.5 < 0$

$\therefore$ **ทำนาย $y = 0$ สำหรับ x_2**

Hypothesis function:

ทำนาย '1' เมื่อ $\theta_0 + \theta_1 f_1 + \theta_2 f_2 + \theta_3 f_3 \geq 0$

ถ้ามี example x_2

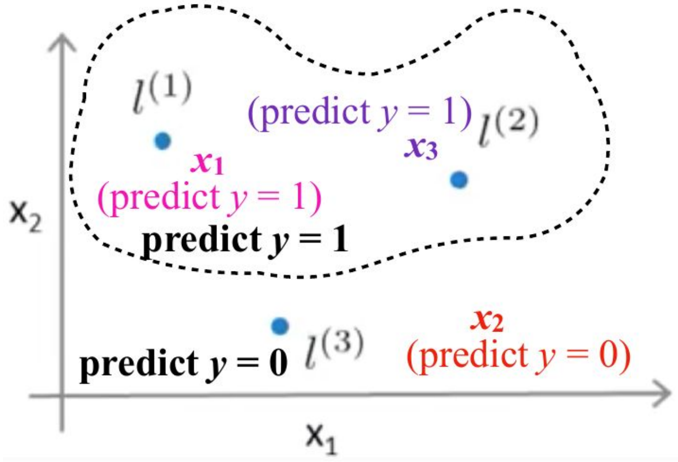
เราจะคำนวณ: f_1, f_2, f_3

สมมติ เราทำนาย

$$\theta_0 = -0.5,\ \theta_1 = 1,\ \theta_2 = 1,\ \theta_3 = 0$$

Kernels และ Similarity

ถ้ามี features (คุณลักษณะ) เหล่านี้ ลองดูว่าจะเรียนรู้ hypothesis function อะไรได้



Hypothesis function:

ทำนาย '1' เมื่อ $\theta_0 + \theta_1 f_1 + \theta_2 f_2 + \theta_3 f_3 \geq 0$

ถ้ามี example x_2

เราจะคำนวณ: f_1, f_2, f_3

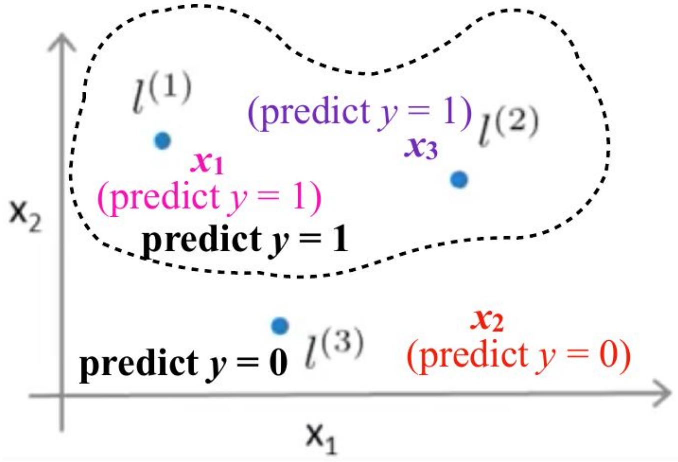
สมมติ เราทำนาย

$$\theta_0 = -0.5,\ \theta_1 = 1,\ \theta_2 = 1,\ \theta_3 = 0$$

นี่ให้ความเข้าใจพื้นฐานเกี่ยวกับว่า นิยามของ landmarks และ kernel function ทำให้เรียนรู้ non-linear decision boundary ที่ค่อนข้างซับซ้อนได้อย่างไร !

Kernels และ Similarity

ถ้ามี features (คุณลักษณะ) เหล่านี้ ลองดูว่าจะเรียนรู้ hypothesis function อะไรได้



Hypothesis function:

ทำนาย '1' เมื่อ $\theta_0 + \theta_1 f_1 + \theta_2 f_2 + \theta_3 f_3 \geq 0$

ถ้ามี example x_2

เราจะคำนวณ: f_1, f_2, f_3

สมมติ เราทำนาย

$$\theta_0 = -0.5,\ \theta_1 = 1,\ \theta_2 = 1,\ \theta_3 = 0$$

คำถาม: หา landmarks เหล่านี้ได้อย่างไร ? เลือก landmarks เหล่านี้ได้อย่างไร ? Similarity function อื่นๆ มีอะไรบ้าง? เป็นต้น

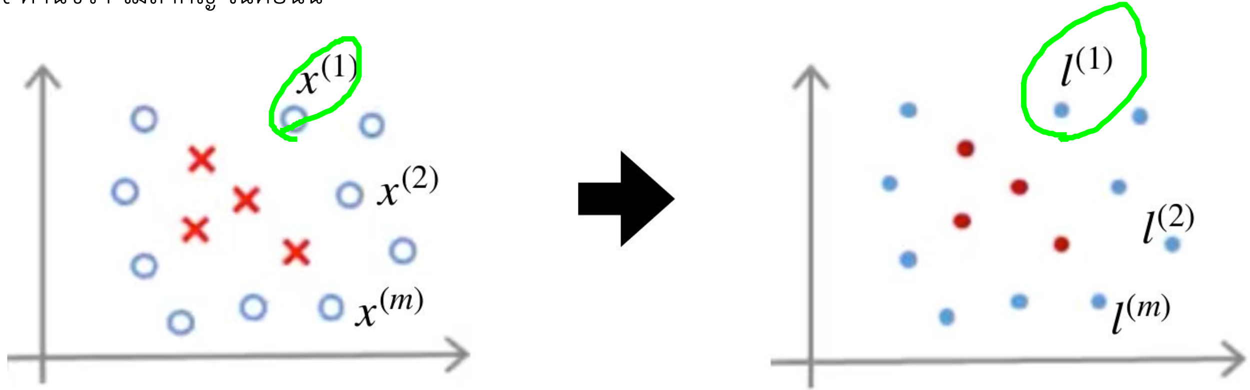
Support Vector Machines (SVM)

Kernels (Part 2)

Krittameth Teachasrisaksakul

การเลือก landmarks

สำหรับ X ทุกตัว ถ้า X อยู่ในชุดข้อมูล เรากำหนดให้ X เป็น landmarks
สีแต่ละสีของจุดใน plot ด้านขวา ไม่สำคัญ ในตอนนี้



นิยาม: ถ้ามีตัวอย่าง example m ตัว

$$(x^{(1)}, y^{(1)}),\ (x^{(2)}, y^{(2)}),\ \ldots,\ (x^{(m)}, y^{(m)})$$

เลือก / กำหนดให้ $l^{(1)} := x^{(1)}, l^{(2)} := x^{(2)}, \ldots, l^{(m)} := x^{(m)}$

SVM ที่ใช้ Kernels

ถ้ามีตัวอย่าง $\boldsymbol{X}$ (จากชุดข้อมูล training / cross validation / testing):

$$\left.\begin{aligned} f_1 &= \textbf{similarity}(x, l^{(1)}) \\ f_2 &= \textbf{similarity}(x, l^{(2)}) \\ &\vdots \end{aligned}\right\} \quad \boldsymbol{f} = \begin{bmatrix} f_0 \\ f_1 \\ f_2 \\ \vdots \\ f_m \end{bmatrix}$$

เมื่อ $f_0 = 1$ (interceptor)

SVM ที่ใช้ Kernels

ถ้ามีตัวอย่าง $\boldsymbol{X}$ (จากชุดข้อมูล training / cross validation / testing):

$$\left.\begin{array}{l} f_1 = \textbf{similarity}(x, l^{(1)}) \\ f_2 = \textbf{similarity}(x, l^{(2)}) \\ \vdots \end{array}\right\} \quad \boldsymbol{f} = \begin{bmatrix} f_0 \\ f_1 \\ f_2 \\ \vdots \\ f_m \end{bmatrix}$$

เมื่อ $f_0 = 1$ (interceptor)

ตัวอย่าง: สำหรับตัวอย่างจากชุดข้อมูล training ($x^{(i)}$, $y^{(i)}$) เราสามารถสร้าง vector

$$x^{(i)} \rightarrow \begin{array}{l} f_1^{(i)} = \textbf{similarity}(x^{(i)}, l^{(1)}) \\ f_2^{(i)} = \textbf{similarity}(x^{(i)}, l^{(2)}) \\ \vdots \\ f_m^{(i)} = \textbf{similarity}(x^{(i)}, l^{(m)}) \end{array}$$

$x^{(i)}$

$$f_i^{(i)} = \textbf{similarity}(x^{(i)}, l^{(i)}) = \exp\left(-\frac{0}{2\sigma^2}\right) = 1$$

SVM ที่ใช้ Kernels

ถ้ามีตัวอย่าง $\boldsymbol{X}$ (จากชุดข้อมูล training / cross validation / testing):

$$\left.\begin{aligned} f_1 &= \textbf{similarity}(x, l^{(1)}) \\ f_2 &= \textbf{similarity}(x, l^{(2)}) \\ &\vdots \end{aligned}\right\} \quad \boldsymbol{f} = \begin{bmatrix} f_0 \\ f_1 \\ f_2 \\ \vdots \\ f_m \end{bmatrix}$$

เมื่อ $f_0 = 1$ (interceptor)

ตัวอย่าง: สำหรับตัวอย่างจากชุดข้อมูล training $(x^{(i)}, y^{(i)})$ เราสามารถสร้าง vector

$$x^{(i)} \rightarrow \begin{aligned} f_1^{(i)} &= \textbf{similarity}(x^{(i)}, l^{(1)}) \\ f_2^{(i)} &= \textbf{similarity}(x^{(i)}, l^{(2)}) \\ &\vdots \\ f_m^{(i)} &= \textbf{similarity}(x^{(i)}, l^{(m)}) \end{aligned}$$

แทนที่จะใช้ $X^{(i)} \in \mathbb{R}^{n+1}$

เราจะเขียนแทน $X^{(i)}$ แต่ละตัวด้วย feature vector

$$f^{(i)} = \begin{bmatrix} f_0^{(i)} \\ f_1^{(i)} \\ \vdots \\ f_m^{(i)} \end{bmatrix}$$

SVM ที่ใช้ Kernels

สมมติ เรามี parameter ที่เรียนรู้แล้ว $\boldsymbol{\theta}$ $(\boldsymbol{\theta} \in \mathbb{R}^{m+1})$

$\theta^T x$

Hypothesis: ถ้ามี $\boldsymbol{x}$ เราคำนวณ features $\boldsymbol{f} \in \mathbb{R}^{m+1}$

แล้ว เราทำนาย $'y=1'$ ถ้า

$$\boldsymbol{\theta}^T \boldsymbol{f} \geq 0 \Longleftrightarrow \theta_0 f_0 + \theta_1 f_1 + \ldots + \theta_m f_m \geq 0$$

จะหาค่า parameter $\boldsymbol{\theta}$ ได้อย่างไร?

SVM ที่ใช้ Kernels

สมมติ เรามี parameter ที่เรียนรู้แล้ว $\boldsymbol{\theta}$ ($\boldsymbol{\theta} \in \mathbb{R}^{m+1}$)

Hypothesis: ถ้ามี x เราคำนวณ features $f \in \mathbb{R}^{m+1}$

แล้ว เราทำนาย '$y=1$' ถ้า $\boldsymbol{\theta}^T \boldsymbol{f} \geq 0 \iff \theta_0 f_0 + \theta_1 f_1 + \ldots + \theta_m f_m \geq 0$

Objective function:

$$\min_{\boldsymbol{\theta}} C \Sigma_{i=1}^{m} y^{(i)} \textbf{cost}_1(\boldsymbol{\theta}^T \boldsymbol{f}^{(i)}) + (1 - \boldsymbol{y}^{(i)}) \textbf{cost}_0(\boldsymbol{\theta}^T \boldsymbol{f}^{(i)}) + \frac{1}{2} \Sigma_{j=1}^{n} \theta_j^2$$

สังเกตว่า สำหรับปัญหา optimization นี้ เรามี $n = m$

SVM ที่ใช้ Kernels

สมมติ เรามี parameter ที่เรียนรู้แล้ว $\boldsymbol{\theta}$ ($\boldsymbol{\theta} \in \mathbb{R}^{m+1}$)

Hypothesis: ถ้ามี $\boldsymbol{x}$ เราคำนวณ features $\boldsymbol{f} \in \mathbb{R}^{m+1}$

แล้ว เราทำนาย '$y=1$' ถ้า $\boldsymbol{\theta}^T\boldsymbol{f} \geq 0 \iff \theta_0 f_0 + \theta_1 f_1 + \ldots + \theta_m f_m \geq 0$

Objective function:

$$\min_{\boldsymbol{\theta}} C\Sigma_{i=1}^{m} y^{(i)}\textbf{cost}_1(\boldsymbol{\theta}^T\boldsymbol{f}^{(i)}) + (1-\boldsymbol{y}^{(i)})\textbf{cost}_0(\boldsymbol{\theta}^T\boldsymbol{f}^{(i)}) + \frac{1}{2}\Sigma_{j=1}^{n}\theta_j^2$$

$$\because \frac{1}{2}\Sigma_{j=1}^{n}\theta_j^2 = \frac{1}{2}(\theta_1^2 + \theta_2^2) = \frac{1}{2}\left(\sqrt{\theta_1^2 + \theta_2^2}\right)^2 = \frac{1}{2}\|\boldsymbol{\theta}\|^2$$

$$\because \Sigma_j \theta_j^2 = \boldsymbol{\theta}^T\boldsymbol{\theta}$$

เพิ่มมาเพื่อ rescaling !

ในบาง implementation : คำนวณ $\boldsymbol{\theta}^T \boldsymbol{M} \boldsymbol{\theta}$ เพื่อ computational efficiency (ประสิทธิภาพในการคำนวณ)

การเลือก Parameter ของ SVM

C (เช่น $\frac{1}{\lambda}$)

C มาก : bias ต่ำลง, variance สูง(≈ λ น้อย) มีแนวโน้ม overfitting

C น้อย : bias สูงขึ้น, variance ต่ำ (≈ λ มาก) มีแนวโน้ม underfitting

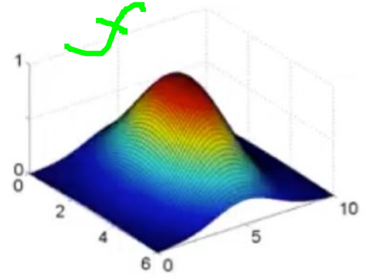
---

σ^2

σ^2 มาก : features f_i เปลี่ยนอย่าง smooth มีแนวโน้ม underfitting

bias สูงขึ้น, variance ต่ำลง

$\sigma^2 = 3$



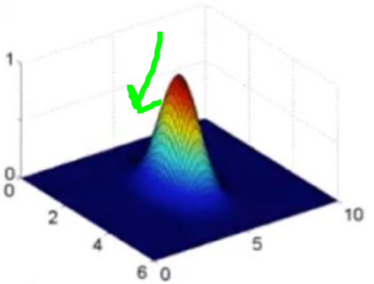
ทบทวน:

$$f_i = \exp\left(-\frac{\|x - l^{(i)}\|^2}{2\sigma^2}\right)$$

σ^2 น้อย : features f_i เปลี่ยนอย่าง smooth น้อยลง มีแนวโน้ม overfitting

bias ต่ำลง, variance สูงขึ้น

$\sigma^2 = 0.5$



Question

สมมติ เรา train SVM และพบว่ามัน overfit ชุดข้อมูล training

ข้อใดต่อไปนี้จะเป็นขั้นตอนต่อไปที่เหมาะสม ? วงทุกข้อที่ถูกต้อง

(i) เพิ่ม C

(ii) ลด C

(iii) เพิ่ม σ^2

(iv) ลด σ^2

Support Vector Machines (SVM)

การใช้ SVM

Krittameth Teachasrisaksakul

การประยุกต์ใช้

1. ใช้ SVM software package (เช่น scikit-learn, libsvm, ...) เพื่อแก้หาค่า parameter
2. จำเป็น ต้องกำหนดค่า:
   i. การเลือกค่า parameter C
   ii. การเลือก kernel ก็คือ similarity function เช่น
   - ไม่มี kernel (หรือ 'linear kernel')

     ทำนาย $y = 1$ ถ้า $\boldsymbol{\theta}^T \boldsymbol{x} \geq 0$ $(\theta_0 + \theta_1 x_1 + \ldots + \theta_n x_n \geq 0)$

Linear kernel บอกเป็นนัยว่าเราใช้ 'standard linear classifier' (ตัวแยกประเภทที่ใช้ฟังก์ชันเชิงเส้นแบบมาตรฐาน)

คำถาม: เมื่อใดที่เราจำเป็นต้องใช้มัน?

เมื่อ n มาก และ m น้อย ก็คือ $x_i \in \mathbb{R}^{n+1}$

ในกรณีนี้ เรามีข้อมูลไม่เพียงพอ และอยากหลีกเลี่ยง overfitting !

การประยุกต์ใช้

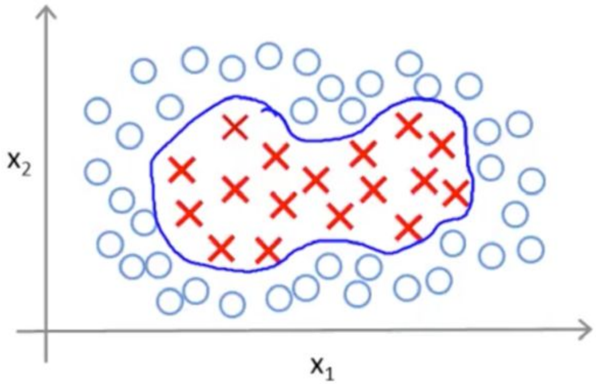
1. ใช้ SVM software package (เช่น scikit-learn, libsvm, ...) เพื่อแก้หาค่า parameter
2. จำเป็น ต้องกำหนดค่า:
   i. การเลือกค่า parameter C
   ii. การเลือก kernel ก็คือ similarity function เช่น

- ไม่มี kernel (หรือ 'linear kernel')

  ทำนาย $y = 1$ ถ้า $\boldsymbol{\theta}^T \boldsymbol{x} \geq 0$ $(\theta_0 + \theta_1 x_1 + \ldots + \theta_n x_n \geq 0)$

- Gaussian kernel (ต้องเลือกค่า σ^2):

$$f_i = \exp\left(- \frac{\|x - l^{(i)}\|^2}{2\sigma^2} \right)$$ เมื่อ $$l^{(i)} = x^{(i)}$$



คำถาม: เมื่อใดที่เราจะใช้ Gaussian kernel ?

เมื่อ $x \in \mathbb{R}^n$, n น้อย, และ m มาก

การประยุกต์ใช้

เตือนความจำเกี่ยวกับ implementation

ต้องทำ feature scaling ก่อนใช้ Gaussian kernel

เพราะอะไร ? $\because \|x - l\|^2 = (x_1 - l_1)^2 + (x_2 - l_2)^2 + \ldots + (x_n - l_n)^2 \quad (x \in \mathbb{R}^{n+1})$

ตัวอย่าง (Housing domain): $x_1 \in [0,\ 1000]\quad \text{feet}^2$

$x_2 \in \{1,\ 2,\ 3,\ 4,\ 5\}$

ถ้าในกรณีนี้ : ขนาดพื้นที่บ้าน (size) จะมีอิทธิพลมากกับ ระยะห่าง (distance) เหล่านี้

Kernel แบบอื่นๆ

- ไม่ใช่ similarity function ทุกอัน $\textbf{similarity}(x, l)$ ที่จะเป็น kernel ที่ถูกต้อง / ใช้ได้ (valid)
- เพื่อยอมรับว่าเป็น kernel ที่ถูกต้อง ต้องสอดคล้องกับ Mercer's theorem
  - เพื่อทำให้แน่ใจว่า optimization ของ SVM package ทำงานอย่างถูกต้อง และ
  - ไม่ diverge !
- ตัวเลือกอื่นของ off-the-shelf kernels (ที่ไม่ต้องพัฒนาเอง)
  - Polynomial kernel: $\textbf{similarity}(x, l) := (x^T l + \varepsilon)^d$ เช่น

$$(x^T l)^2 \qquad (x^T l)^3 \qquad (x^T l + 1)^3 \qquad (x^T l + 5)^4$$

  - Kernel ที่เข้าใจยากขึ้น: string kernel, chi-square kernel, histogram, intersection kernel, ...

Question

สมมติ เราพยายามตัดสินใจเลือกระหว่าง kernel ไม่กี่ตัว และเลือกค่า parameter เช่น C, σ^2 เป็นต้น เราควรเลือกอย่างไร?

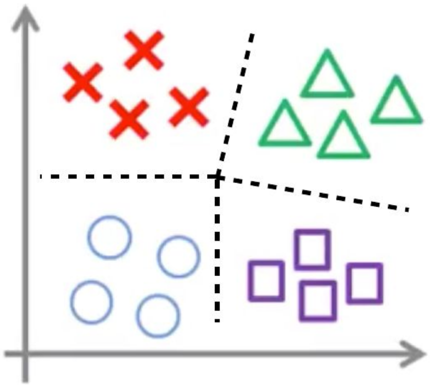
(i) เลือกอะไรก็ตามที่ ทำงานมีประสิทธิภาพสูงสุดกับข้อมูล training

(ii) เลือกอะไรก็ตามที่ ทำงานมีประสิทธิภาพสูงสุดกับข้อมูล cross-validation

(iii) เลือกอะไรก็ตามที่ ทำงานมีประสิทธิภาพสูงสุดกับข้อมูล test

(iv) เลือกอะไรก็ตามที่ ทำให้มี SVM margin มากที่สุด

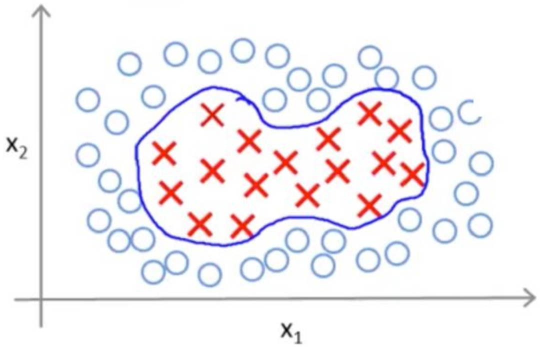
Multi-class Classification (การแยกประเภท มากกว่า 2 ประเภท)



$y \in \{1, 2, 3, ..., K\}$

- SVM package หลายๆอัน มี multi-class classification functionality (ที่ built-in / มีพร้อมใช้ได้)
- ไม่อย่างนั้น ใช้วิธี one-vs-all (แยก 1 class ออกจาก class ที่เหลือทั้งหมด) ก็คือ
  - Train SVM K ตัว แต่ละตัวใช้เพื่อแยก $y = i$ ออกจาก class ที่เหลือทั้งหมด เมื่อ $i = 1, 2, ..., K$
  - หาค่า $\theta^{(1)}, \theta^{(2)}, ..., \theta^{(K)}$
  - เลือก class i ที่มีค่า $(\theta^{(i)})^T x$ มากที่สุด

Logistic Regression *vs.* SVM



เมื่อใดควรใช้ algorithm แต่ละอัน (เทียบกับอีกอัน) ?

สมมติ n = จำนวน features ($x \in \mathbb{R}^{n+1}$), m = จำนวน training examples

1. ถ้า n มาก (เทียบกับ m) (เช่น $n \geq m, n = 10{,}000, 10 \leq m \leq 10{,}000$)
   ใช้ logistic regression หรือ SVM ที่ไม่มี kernel (ก็คือ ‘linear kernel’)
2. ถ้า n น้อย, m มีค่าปานกลาง (intermediate) (เช่น $1 \leq n \leq 10{,}000, 10 \leq m \leq 50{,}000$)
   ใช้ SVM ที่ใช้ Gaussian kernel
3. ถ้า n น้อย, m มาก (เช่น $1 \leq n \leq 1{,}000, m > 50{,}000$)
   สร้าง / เพิ่ม features แล้วใช้ logistic regression หรือ SVM ที่ไม่มี kernel
4. **Neural network (NN) มีแนวโน้มที่จะทำงานได้ดี ใน setting ส่วนมากที่พูดถึง แต่อาจ train ได้ช้ากว่า**
5. **SVM เป็น convex optimization problem** ในทางปฏิบัติ ค่า local optima ไม่ใช่ปัญหาใหญ่ เมื่อใช้ neural network แต่เราไม่ต้องกังวลเรื่องนี้เมื่อใช้ SVM

References

1. Andrew Ng, Machine Learning, Coursera.
2. Teeradaj Racharak, AI Practical Development Bootcamp.
3. What is Machine Learning?, https://www.digitalskill.org/contents/5