Logistic Regression

อ. ปรัชญ์ ปิยะวงศ์วิศาล

Pratch Piyawongwisal

Today

- Recap - Linear Regression
  - Minimizing cost function
  - Solutions
    - Normal Equation
    - (Stochastic) Gradient Descent
  - Polynomial Regression
  - Regularization -> solves overfitting
- Logistic Regression
  - Sigmoid Function
  - Class probability
  - Softmax Regression
- Lab: IRIS dataset

Recap: Supervised Learning

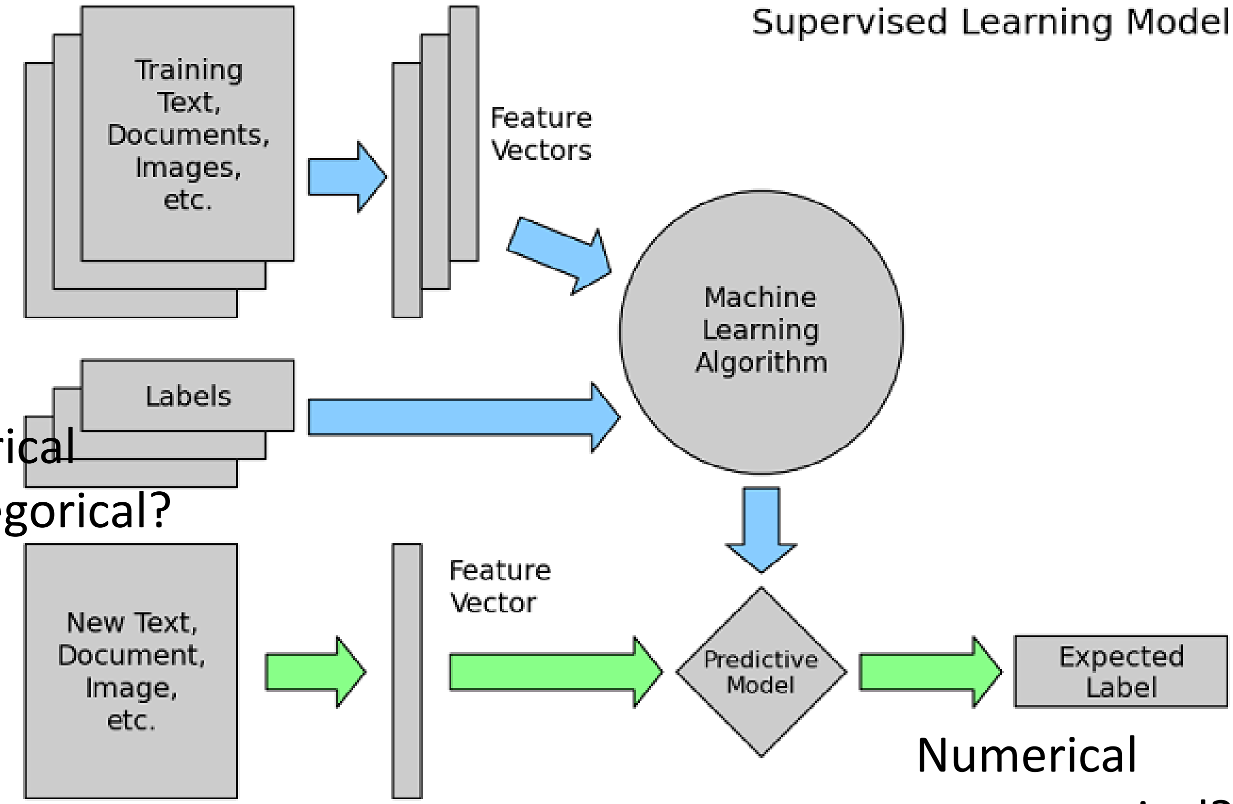
- Classification
  - Predicts class labels/categories
  - ทำนายค่าที่เป็นหมวดหมู่ = จำแนกประเภท
  - อาจมองเป็นการหา **boundary** ที่แบ่งข้อมูลในแต่ละหมวดหมู่ ออกจากกัน

kNN

Logistic Regression

- Regression
  - Predicts continuous values
  - ทำนายค่าที่เป็นจำนวนจริง
  - อาจมองเป็นการหา **hyperplane** ที่ **fit** กับข้อมูลที่มีมากที่สุด

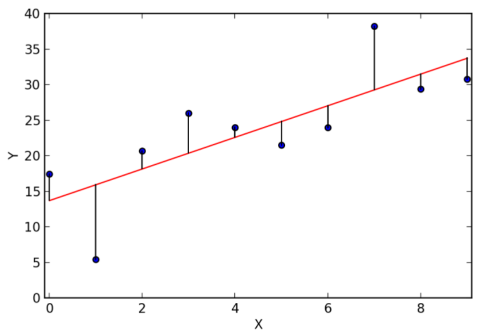
Linear Regression

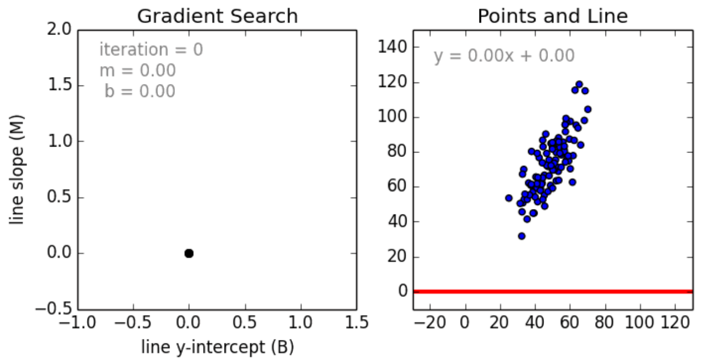


Numerical or categorical?

Numerical or categorical?

Linear Regression - Summary



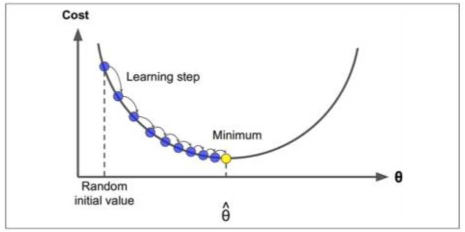


- Regression: หาโมเดลที่ fit กับข้อมูลได้ดีที่สุดได้อย่างไร?
- โมเดล: $\hat{y} = h_\theta(x) = \boldsymbol{\theta}^T \boldsymbol{x}$
- cost function ของโมเดล: mean-square error (MSE)

$$J(\theta) = MSE(\theta) = \frac{1}{M}\sum_{i=1}^{M}(\theta^T x^{(i)} - y^{(i)})^2$$

- การ train โมเดล คือ การหาค่าของ $\hat{\theta}_{MSE} = \underset{\theta}{\operatorname{argmin}}\, J(\theta)$
- Solution 1: แก้สมการตรง ๆ จะได้ Normal equation: $\hat{\theta}_{MSE} = (\mathbf{X}^{\mathbf{T}}.\mathbf{X})^{-\mathbf{1}}.\mathbf{X}^{\mathbf{T}}.\mathbf{y}$
- Solution 2: หรือใช้วิธี Gradient Descend โดยค่อยๆ update θ:

$$\theta^{(\textbf{next step})} = \theta - \eta\nabla_\theta MSE(\theta)$$



ภูเขา MSE

Exercise: Housing Price Prediction

- ข้อมูล training

ขนาด (Sq. ft.)	จำนวนห้องนอน	ระยะทางไปห้างสรรพสินค้า	ราคา (ล้านบาท)
1000	2	5	9.5
1500	3	30	8.0
2000	5	40	12.5
1700	1	5	9.0
1200	2	30	5.5

โมเดล Linear Regression (ราคาสำหรับ 1 หลัง) คือ

$$h_\theta(x) = \theta_0 + \theta_1 \mathrm{x}_1 + \theta_2 \mathrm{x}_2 + \theta_3 \mathrm{x}_3$$

เราสามารถเขียนข้อมูลในรูป matrix (สำหรับ m หลัง) ได้ดังนี้

$$\boldsymbol{X} = \begin{bmatrix} 1 & 1000 & 2 & 5 \\ 1 & 1500 & 3 & 30 \\ 1 & 2000 & 5 & 40 \\ 1 & 1700 & 1 & 5 \\ 1 & 1200 & 30 & 5.5 \end{bmatrix} \qquad \boldsymbol{y} = \begin{bmatrix} 9.5 \\ 8.0 \\ 12.5 \\ 9.0 \\ 5.5 \end{bmatrix}$$

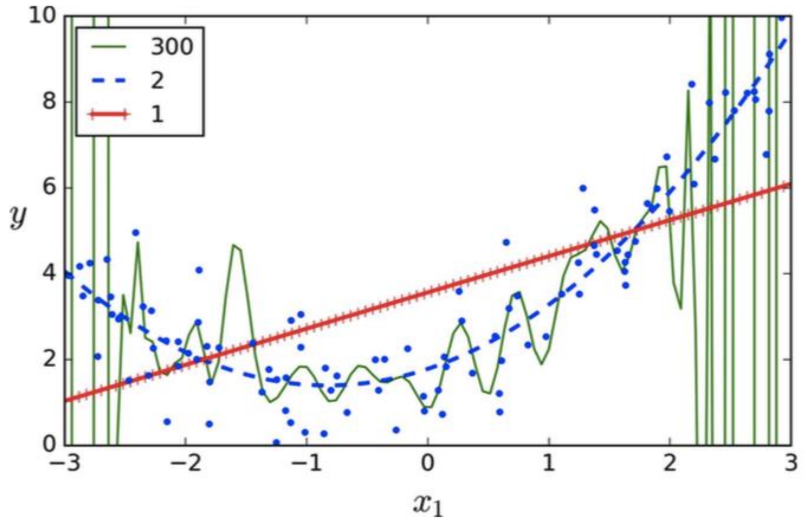
โมเดลในรูป matrix: $\mathbf{y} = \boldsymbol{X\theta}$

จากนั้นคำนวณหา $\hat{\theta}_{MSE}$ โดยใช้ normal equation:

$$\hat{\theta}_{MSE} = (\mathbf{X}^{\mathbf{T}}.\mathbf{X})^{-1}.\mathbf{X}^{\mathbf{T}}.\mathbf{y}$$

Polynomial Regression, Regularization

- สามารถทำ Polynomial Regression โดยเพิ่มมิติข้อมูล training เช่น
  - x_i เดิม = [น้ำหนัก,ส่วนสูง,อายุ] = [70, 150, 30]
  - x_i ใหม่ = [70, 150, 30, 70^2, 150^2, 30^2]
- แต่ถ้า poly degree สูงไปอาจทำให้ overfit ☹
- Solution: เพิ่มพจน์ regularization ใน cost function



$$J(\theta) = \mathrm{MSE}(\theta) + \boxed{\alpha \frac{1}{2} \sum_{i=1}^{n} \theta_i^2}$$

Ridge (L2) regularization

ช่วยลดอิทธิพลของ polynomial degree สูงๆ อย่าง x^4, x^5 = ลด overfitting

Probabilistic Interpretation of Linear Regression

- (whiteboard)
  - ทำไม MSE จึงเป็น cost ที่ reasonable ?
  - การตีความ Linear Regression จากมุมมองทางสถิติ
  - การมอง training data เป็นตัวแปรสุ่ม X, y
  - การประมาณ (estimate) parameter θ จากข้อมูล
    - Maximum Likelihood Estimation (MLE)

Logistic Regression

How to apply?

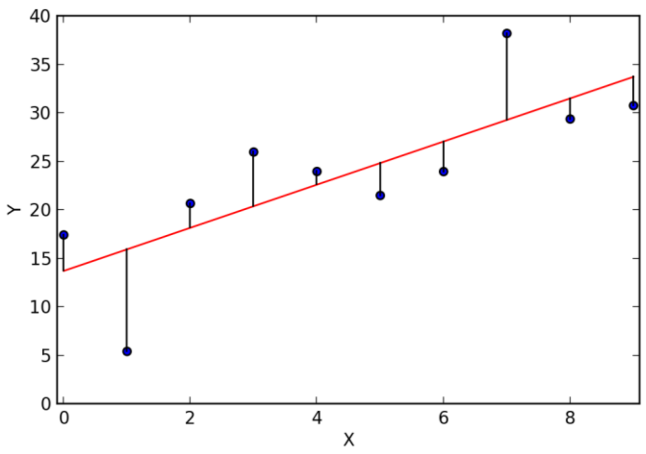
Linear Regression ➔ (Binary) Classification

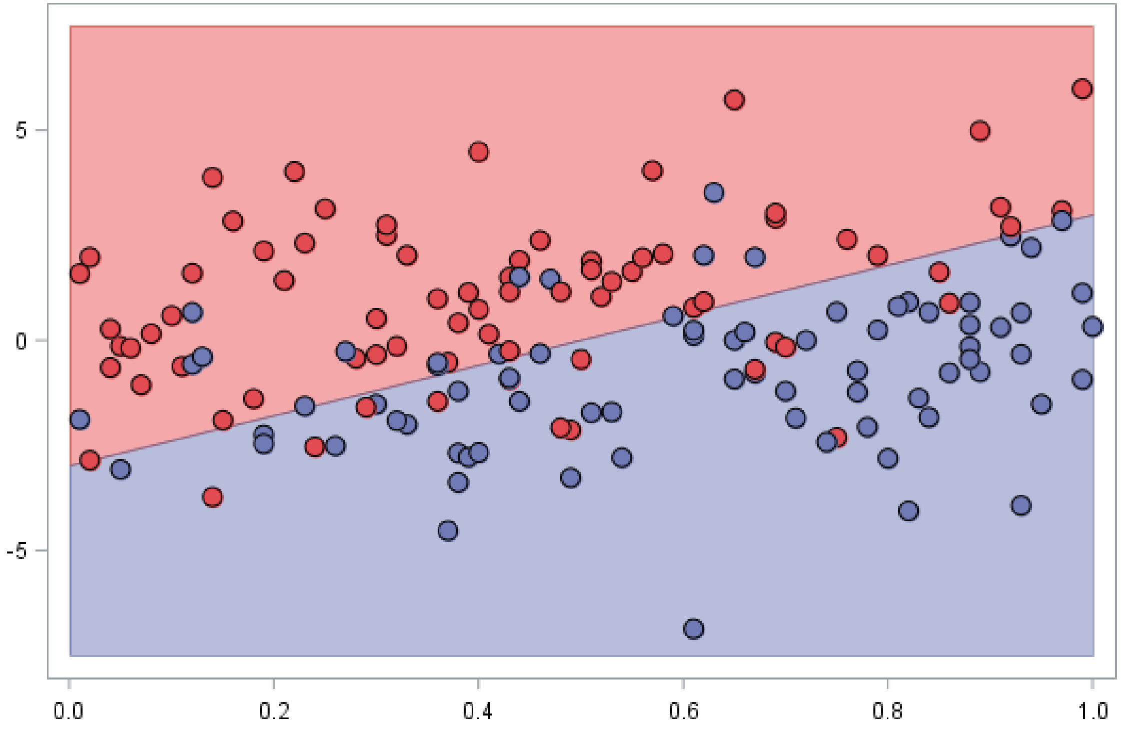
Linear Regression

- Input: $\boldsymbol{x}$ คือ input features
- Output: $\hat{y}$ เป็นจำนวนจริง

โมเดล: $\hat{y} = h_\theta(x) = \boldsymbol{\theta}^T\boldsymbol{x}$

- ความสัมพันธ์แบบ linear





พิจารณาปัญหา binary classification เช่น

- อีเมลเป็น spam (label=1) หรือไม่เป็นสแปม (label=0)
- คนไข้เป็นมะเร็ง (label=1) หรือไม่เป็นมะเร็ง (label=0)

* $\hat{p}$ คือค่าเป็นความน่าจะเป็น

Logistic Regression

- **Key Idea:** เราต้องการให้โมเดล $h_\theta(x)$ ทำนายค่า
  - $\hat{p}(y = 1|x;\theta)$ **ความน่าจะเป็นที่ label จะเป็น 1**
  - $\hat{p}(y = 0|x;\theta)$ **ความน่าจะเป็นที่ label จะเป็น 0**
- จากนั้นเราสามารถใช้ค่าของ $\hat{p}$ ในการ classify ข้อมูลด้วยกฎง่ายๆ นี้ได้

$$\hat{y} = \begin{cases} 0 & \text{if } \hat{p} < 0.5, \\ 1 & \text{if } \hat{p} \geq 0.5. \end{cases}$$

Logistic Regression

- **Key Idea:** เราต้องการให้โมเดล $h_\theta(x)$ ทำนายค่า
  - $\hat{p}(y = 1|x; \theta)$ **ความน่าจะเป็นที่ label จะเป็น 1**
  - $\hat{p}(y = 0|x; \theta)$ **ความน่าจะเป็นที่ label จะเป็น 0**

- Q: ใช้ $h_\theta(x)$ เดียวกันกับ Linear Regression เลยได้ไหม?

$$\hat{p}(y = 1|x; \theta) = h_\theta(x) = \boldsymbol{\theta}^T \boldsymbol{x}$$

Logistic Regression

- **Key Idea:** เราต้องการให้โมเดล $h_\theta(x)$ ทำนายค่า
  - $\hat{p}(y = 1|x; \theta)$ **ความน่าจะเป็นที่ label จะเป็น 1**
  - $\hat{p}(y = 0|x; \theta)$ **ความน่าจะเป็นที่ label จะเป็น 0**

- Q: ใช้ $h_\theta(x)$ เดียวกันกับ Linear Regression เลยได้ไหม?

$$\hat{p}(y = 1|x; \theta) = h_\theta(x) = \boldsymbol{\theta}^T \boldsymbol{x}$$

- **ปัญหา:** <u>ความน่าจะเป็น</u>ต้องมีค่าในช่วง 0-1 เท่านั้น ในขณะที่ $\boldsymbol{\theta}^T \boldsymbol{x}$ มีค่าเป็นเท่าใดก็ได้ ☹

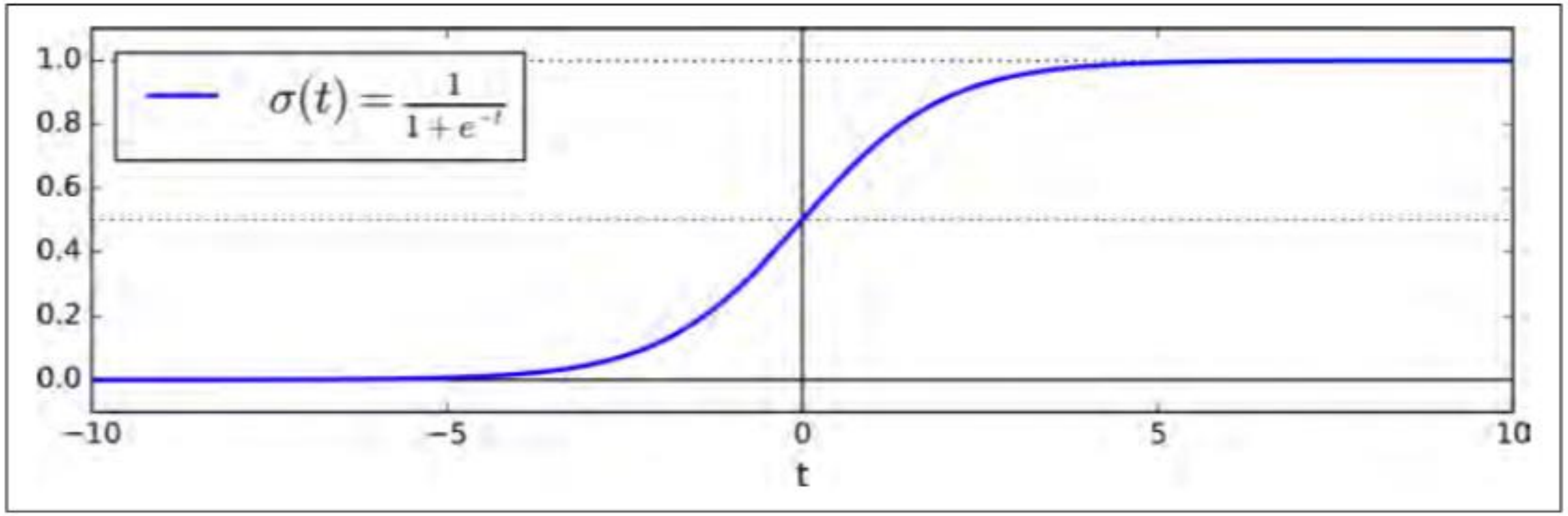
Sigmoid Function

- Solution: เปลี่ยน model เป็น

$$\hat{p} = h_\theta(x) = \sigma(\theta^T x)$$

- โดยที่ $\sigma(t)$ คือ logistic (sigmoid) function

$$\sigma(t) = \frac{1}{1 + \exp(-t)}$$

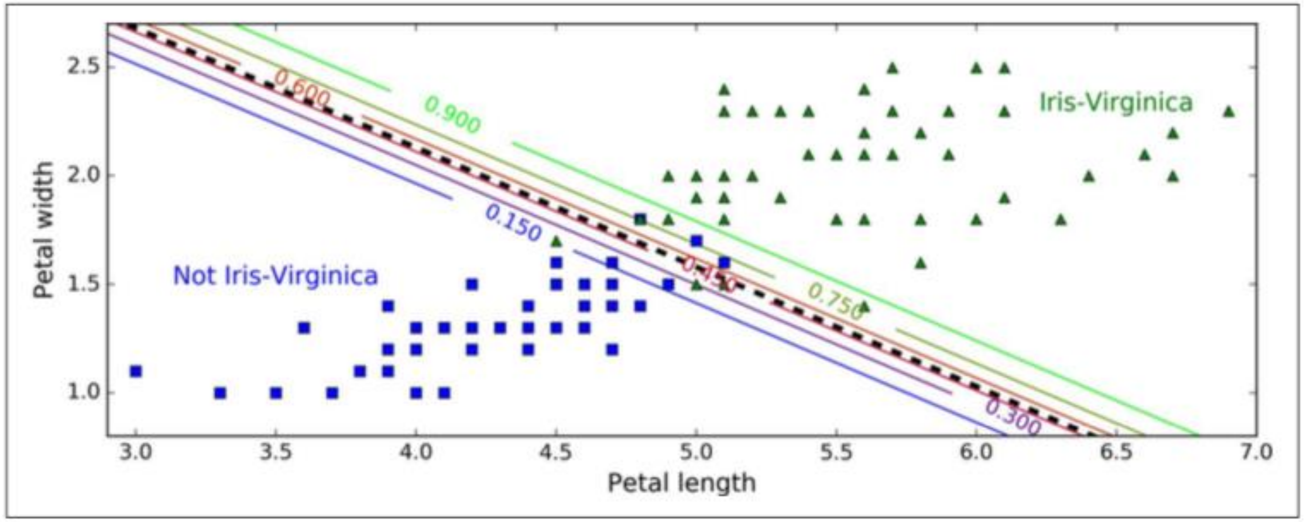


Logistic Regression Model

- โมเดล: $\hat{p}(y = 1|x; \theta) = h_\theta\ (x) = \sigma(\theta^T x)$
- กฎการตัดสินใจ

$$\hat{y} = \begin{cases} 0 & \text{if } \hat{p} < 0.5, \\ 1 & \text{if } \hat{p} \geq 0.5. \end{cases}$$

- โดยที่
  - Input: $\boldsymbol{x}$ คือ input features
  - Output: $\hat{y}$ เป็น binary (0/1)

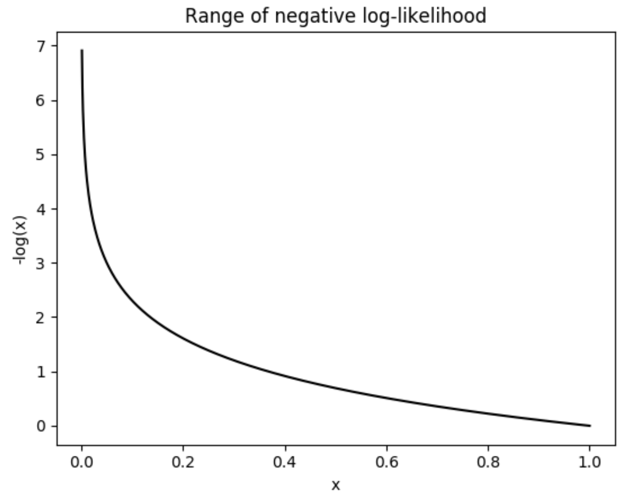


ตัวอย่าง decision boundary จากการใช้วิธี logistic regression

Cost Function

- cost function จะต้องเปลี่ยนไปด้วย
- cost สำหรับแต่ละ instance ข้อมูลคือ

$$c(\theta) = \begin{cases} -\log(\hat{p}) & \text{if } y = 1, \\ -\log(1-\hat{p}) & \text{if } y = 0. \end{cases}$$



ไอเดียคือ ถ้าเฉลยเป็นมะเร็ง (y=1)
แต่ $\hat{p}$ ทำนายว่าใกล้ 0 ค่า cost = -log(~0) จะใหญ่มาก
ถ้า $\hat{p}$ ทำนายว่าใกล้ 1 ค่า cost จะเป็น 0

- cost สำหรับทั้งชุดข้อมูลจึงเป็นดังสมการ

$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m}\left[y^{(i)}log\left(\hat{p}^{(i)}\right) + \left(1 - y^{(i)}\right)log\left(1 - \hat{p}^{(i)}\right)\right]$$

How to train?

- จะ train โมเดลนี้อย่างไร? (minimize cost อย่างไร?)

$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m}\left[y^{(i)}log\left(\hat{p}^{(i)}\right) + \left(1 - y^{(i)}\right)log\left(1 - \hat{p}^{(i)}\right)\right]$$

- ไม่มี closed-form solution แบบ Normal Eq ☹

How to train?

- จะ train โมเดลนี้อย่างไร? (minimize cost อย่างไร?)

$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m}\left[y^{(i)}log\left(\hat{p}^{(i)}\right) + \left(1-y^{(i)}\right)log\left(1-\hat{p}^{(i)}\right)\right]$$

- ไม่มี closed-form solution แบบ Normal Eq ☹
- แต่สามารถหา gradient/partial derivative ได้ ☺

$$\frac{\partial}{\partial\theta_j}J(\theta) = \frac{1}{m}\sum_{i=1}^{m}\left(\sigma\left(\theta^T\cdot\mathbf{x}^{(i)}\right) - y^{(i)}\right)x_j^{(i)}$$

- ดังนั้น ใช้วิธี Stochastic Gradient Descent แบบเดิมได้

From Binary to Multi-class classification

- ข้อจำกัดของ Logistic Regression คือ สามารถใช้กับข้อมูลที่มี label เพียง 2 คลาสเท่านั้น
- หากต้องการ generalize ไปใช้กับ multi-class สามารถเปลี่ยนไปใช้ softmax function

$$\hat{p}_k = \sigma(\mathbf{s}(\mathbf{x}))_k = \frac{\exp\left(s_k(\mathbf{x})\right)}{\sum_{j=1}^{K} \exp\left(s_j(\mathbf{x})\right)}$$

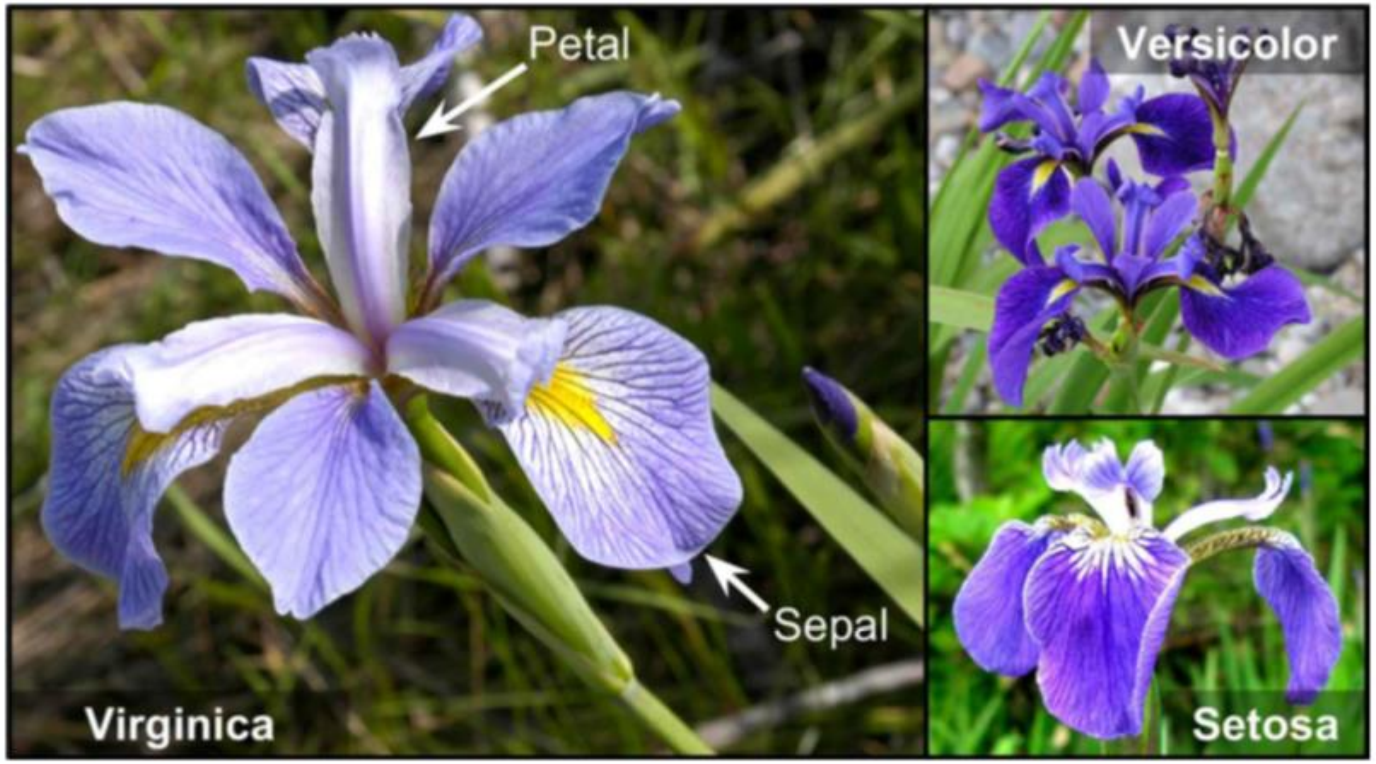
- K is the number of classes.
- $\mathbf{s}(\mathbf{x})$ is a vector containing the scores of each class for the instance $\mathbf{x}$.
- $\sigma(\mathbf{s}(\mathbf{x}))_k$ is the estimated probability that the instance $\mathbf{x}$ belongs to class k given the scores of each class for that instance.

- และ cost function จะเปลี่ยนเป็น Cross-Entropy ซึ่งจะเหมือน LR cost หาก k=2

$$J(\Theta) = -\frac{1}{m}\sum_{i=1}^{m}\sum_{k=1}^{K} y_k^{(i)} \log\left(\hat{p}_k^{(i)}\right)$$

Lab: ใช้ Logistic Regression จำแนกพันธุ์ดอกกล้วยไม้

```
from sklearn.linear_model import LogisticRegression
```



เป้าหมาย: จำแนก Iris-Virginica ออกจากชนิดอื่น โดยใช้ขนาดของกลีบ Sepal/Petal - width/length

Next week

- Support Vector Machine (SVM)