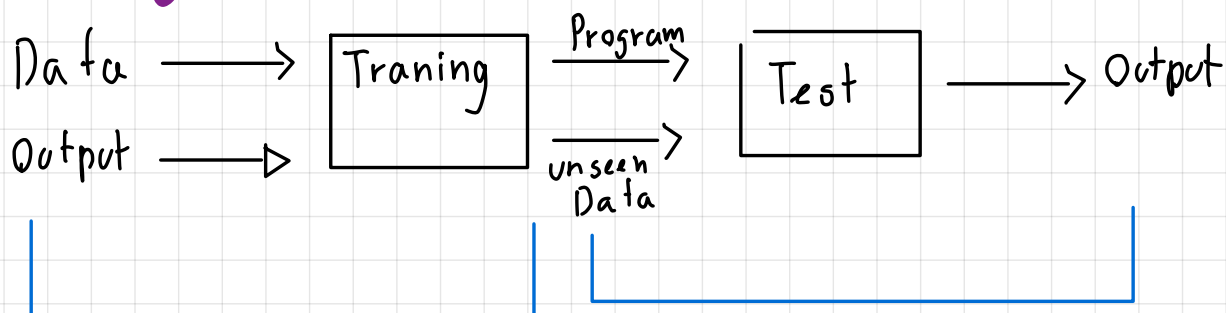


# Supervised learning Setup: Training \*\*\*

## Diagram



training

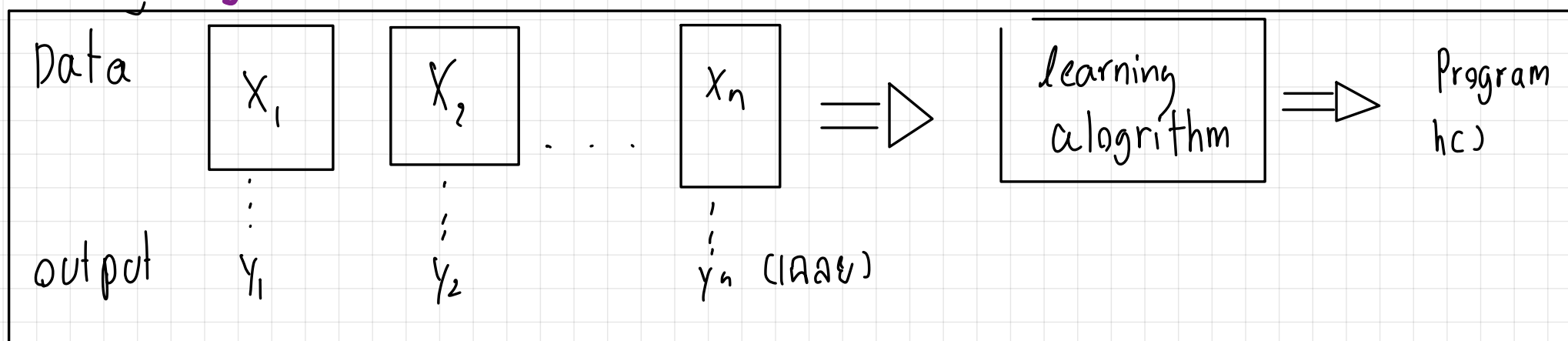
testing

- ใช้ Algorithm ในการฝึก
- ใช้ data และ label ในการ train

- ใช้ program กับ unseen data ในการ ทำนาย ผลลัพธ์

- program ที่ได้มาจากการ train ของ data และ label

## training



Note:  $x_n, y_n$  คือค่าสมมุติ

การทำนายจากข้อมูล  $X$  และ ได้

Goal: Result คือ program  $h: X \rightarrow Y$  ผลลัพธ์เป็น  $Y$   
Data และ output เป็นสิ่งที่รู้

- $X$  เรียกว่า "feature space"
- $X = \mathbb{R}^d$  โดย  $d$  เป็นจำนวนเต็ม
- $d$  เรียกว่า dimension/feature คือ มิติของข้อมูล

รูปแบบทางคณิตศาสตร์

$$D = \{(\vec{x}_1, y_1), (\vec{x}_2, y_2), \dots, (\vec{x}_n, y_n)\} \subseteq X \times Y$$

Given: Data ที่มีข้อมูล  $(\vec{x}, y)$  คู่กัน

โดยที่  $\vec{x}$  เรียกว่า "feature vector" ( $\vec{x} \in X$ )

และ  $y$  เรียกว่า "label" ( $y \in Y$ )

- แต่ละ data item จะถูกสมมุติว่า

มีค่าตามเงื่อนไขของ  $p$  uncorrelated

$$\text{Each } (\vec{x}_i, y_i) \sim p$$

$p$  เป็นเหมือน การกระจายตัว

ที่ไม่สามารถดึงข้อมูลได้หมด

ถ้าขนาดเข้าถึงต้องรอบรวมข้อมูลทั้งหมด

## Supervised learning

- ML แบบหนึ่ง
- ใช้ label ที่จ้องหา และหาการทำนายของผลลัพธ์ ข้อมูลที่ถูกต้อง โดยข้อมูลที่ให้โดยนั้น

ประเภท output

- Binary classification (call hc)
- Multiclass classification "classifier"
- Regression

# Example of Feature Vector

- Patient data

$$\vec{X} = \begin{bmatrix} x^1 \\ x^2 \\ \vdots \\ x^d \end{bmatrix} = \begin{bmatrix} 0 \\ 22 \\ \vdots \\ 101 \end{bmatrix} \begin{matrix} \rightarrow \text{sex} \\ \rightarrow \text{Age} \\ \\ \rightarrow \text{blood pressure} \end{matrix}$$

- ในที่นี้  $d$  = คำนวณจำนวนมิติในกรณีนี้

## Text data

"an ant and a zebra"

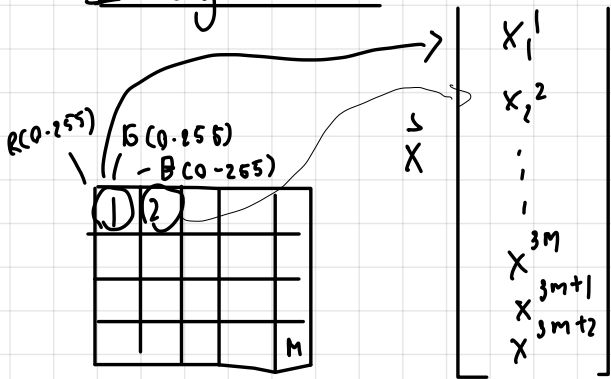
$$\vec{X}_i \in X = \mathbb{R}^d \rightarrow \text{bug!}$$

$$\vec{X} = \begin{bmatrix} x^1 \\ \vdots \\ x^d \end{bmatrix} = \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix}$$

Bag-of-words: เป็นการจัดเก็บคำศัพท์, สูญเสีย detail ลำดับของ word

Note: อาจมี 0 หรือ 1 เกิด waste data ที่ส่งผลต่อการ predict

## Image data



- ไม่เกิด waste data เพราะใช้ทุก coordinate

$$\vec{X}_i \in X = \mathbb{R}^{3m}$$

$$\text{iphone 12 : 12 m pixel} \Rightarrow \vec{X}_i \in \mathbb{R}^{3(12m)} \quad d = 3(12 \times 10^6)$$

# Supervised learning Setup: Testing \*\*\*

- The function  $h$  learn from training is called  
"hypothesis"

- For testing, we hope to apply  $h$  on any data coming from the distribution  $P$ .

∴ "hypothesis" มาสร้างสมมุติฐาน ที่เป็น function จะสามารถ  
ใช้กับ data ที่มาจากกรณีทั่วไปของ population

$$\boxed{\text{IDEAL: } \forall (\tilde{x}, y) \sim P, h(\tilde{x}) = y} \rightarrow y \text{ is true label of } \tilde{x}$$

prediction term

## Generating hypothesis $h$ :

① เลือก learning algorithm ที่เข้ากันได้กับ  $P$  กับขั้นตอนที่เรียกว่า "hypothesis class"  $H$

( $H$  is the set of all possible hypothesis that can be generated by the algorithm)

② เลือก  $h \in H$  ที่ works best on the data

ใช้การวัด loss function

- Loss function: Any function used to evaluate if one hypothesis is worse than another

\*\*\* Lower loss  $\Rightarrow$  better hypothesis

\*\*\* ต้องการ average loss

## Example of loss functions

① 0/1 loss: 
$$L_{0/1}(h, D) = \frac{1}{|D|} \sum_{(\tilde{x}, y) \in D} d(h(\tilde{x}), y) \text{ where } d(h(\tilde{x}), y) = \begin{cases} 0 & \text{if } h(\tilde{x}) = y \\ 1 & \text{otherwise} \end{cases}$$

Note:  $L_{0/1}(h, D)$ ;  $h \rightarrow \text{hypothesis}$   
 $D \rightarrow \text{Data}$   $\left| \sum_{(\tilde{x}, y) \in D} d(h(\tilde{x}), y) \rightarrow \text{หาผลรวมของ } L(\tilde{x}, y) \in D$   
โดยที่  $\tilde{x}$  คือ hypothesis feature  $\tilde{x}$  และ  $y$  คือ label  
ถ้า  $h(\tilde{x}) = y$  เป็น 0 ถ้าไม่ตรง  $y$  เป็น 1 (ถ้าเป็นข้อผิดพลาด)

โดยที่  $\frac{1}{|D|}$  คือ จำนวนข้อมูล Data ทั้งหมด  
หรือเป็น average loss

- 0/1 loss is commonly used in Classification problems

② Square loss: 
$$L_{sq}(h, D) = \frac{1}{|D|} \sum_{(\tilde{x}, y) \in D} (h(\tilde{x}) - y)^2$$

- square loss is commonly used in regression problems

## Learning (Step 2 in generating hypothesis):

- pick  $h \in H$  that minimize loss on the data

$$h^* = \underset{h \in H}{\operatorname{argmin}} L(h, D)$$

- overfitting: The situation in which  $h$  fits too well on data set  $D$ , but fails at making prediction on data outside  $D$

### Example of overfitting hypothesis

-  $h(\tilde{x}) = \begin{cases} y_i & \text{if } \exists (\tilde{x}_i, y_i) \in D \text{ such that } \tilde{x} = \tilde{x}_i \\ 0 & \text{otherwise} \end{cases}$

หรือว่า  $h(\tilde{x})$  ที่ outside data  $D$  จะเท่ากับ 0

true goal of learning: we wish to learn  $h$  that minimizes the generalization loss (global loss)

$$\text{generalization loss: } \mathcal{E} = \mathbb{E} \left[ L(h, (\tilde{x}, y)) \right]_{(\tilde{x}, y) \sim p}$$

Note: it's impossible to calculate  $\mathcal{E}$ , because we can't access  $p$

# Estimating the generalization loss:

- split data

$D_{TR} \rightarrow \text{train}$

$D_{TE} \rightarrow \text{test}$

- The split ratio is usually 80:20 / 70:30 (training: testing)

- learning from the training data set

$$h^* = \operatorname{argmin} L(h, D_{TR})$$

- Evaluating loss via the testing data set

$$\epsilon_{TE} = L(h, D_{TE})$$

- claim As  $|D_{TE}| \rightarrow +\infty$ ,  $\epsilon_{TE} \approx \epsilon$   
(by the weak law of large number)

---

## Supervised learning in practice

1) Define setup

2) Collect data

3) Training the hypothesis from training data  
(must choose the proper learning algorithm)

4) testing the hypothesis on testing data

5) The error from (4) will indicate the actual accuracy