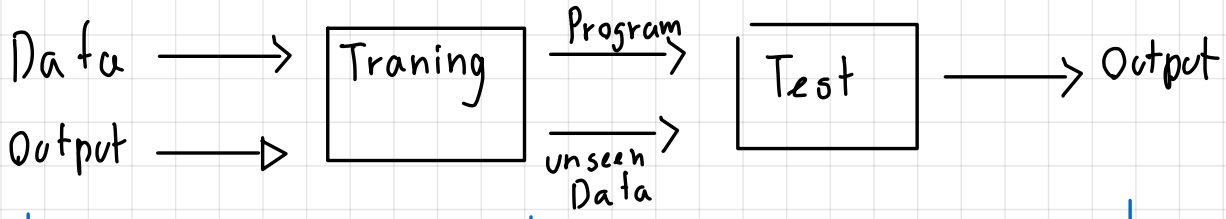
# Supervised learning Setup : Traing ***

## Diagram




traning

- จะใช้ Algorithm ในการฝึก
- ใช้ data และ เฉลยในการ train
- program ที่ได้มาจาก การ train ของ data และ label

testing

- ใช้ program กับ unseen data ในการ ทำนาย ผลเฉลย

Supervised learning

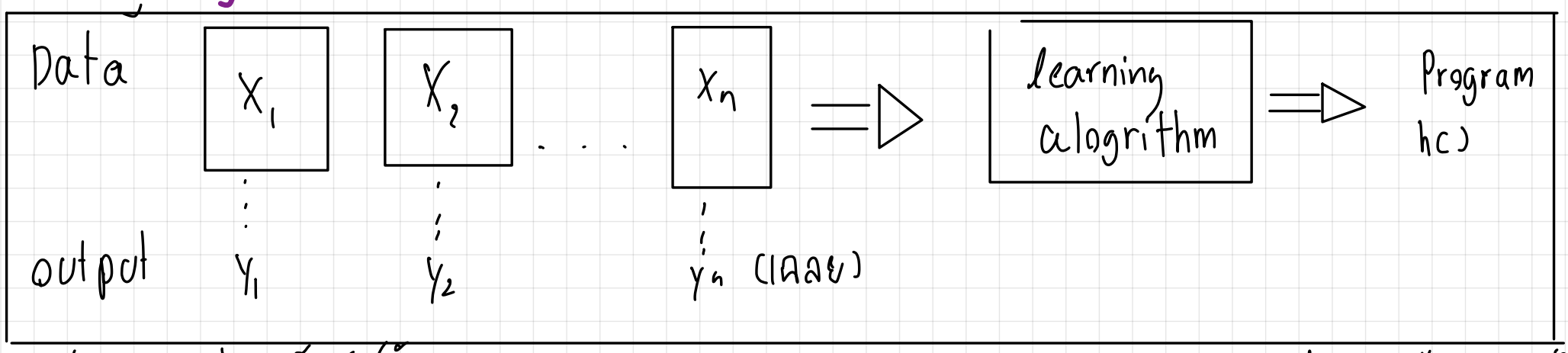
- ML แบบหนึ่ง
- ให้ label ตัวอย่าง และ ทำการทำนาย ของผลลัพธ์ ข้อมูลที่ถูกต้อง โดยข้อมูลที่ไม่เคยเห็น

ประเภท output

- Binary classification
- Multiclass classification
- Regression

} call hc) "classifier"

## training




Note: $x_n, y_n$ มีความสัมพันธ์กัน

การทำนายจากข้อมูล X และ ได้

รูปแบบทางคณิต

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

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 ถูกสมมุติว่า มีความเกี่ยวข้องกับ unaccessable P

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

P เป็นเหมือน การกระจายตัว ที่ไม่สามารถเข้าถึงข้อมูลได้หมด

ก้าขยากเข้าถึงต้องรวบรวมข้อมูลทั้งหมด

Goal: Result คือ program $h: X \to Y$ ผลลัพธ์เป็น Y

Data และ output เป็นสิ่งที่รู้

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

- Y เรียกว่า "label space"
  - Type: Binary classification
    $Y = \{0, 1\}$, $Y = \{-1, +1\}$
    Multiclass Classification
    $Y = \{0, 1, \ldots, k-1\}$
    Regression : $Y = \mathbb{R}$

# Example of Feature Vector

- Patient data

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

- ปกติ d จะค่อนข้างเล็กในกรณีนี้

Text data

"an ant and a zebra"

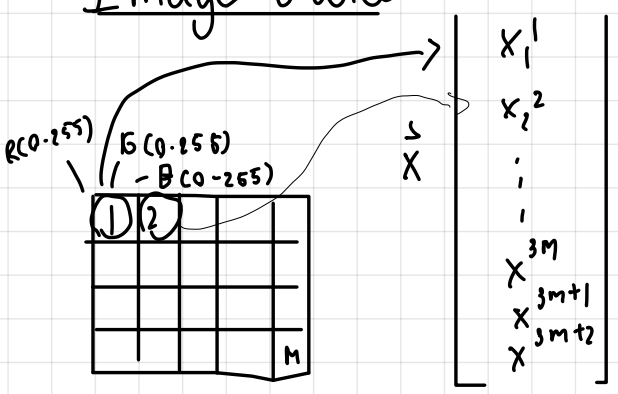
$\vec{x}_i \in X = \mathbb{R}^{d \rightarrow \text{huge}}$

$$\vec{x} = \begin{bmatrix} x^1 \\ \vdots \\ x^3 \\ \vdots \\ x^d \end{bmatrix} = \begin{bmatrix} 1 \\ \vdots \\ 1 \\ \vdots \\ 1 \end{bmatrix}$$

Bag-of-word: เป็นการเก็บความถี่, สูญเสีย detail ลำดับของ word

Note: อาจมีค่า 0 เยอะ=>เกิด waste data ที่ส่งผลต่อการ predict

# Image data




- ไม่เกิด waste data เพราะใช้ทุก coordinate
- $\vec{x}_i \in X = \mathbb{R}^{3m}$
- iphone 12 : 12 mpixel $\Rightarrow \vec{x}_i \in \mathbb{R}^{3(12m)}$ $d = 3(12 \times 10^6)$

# Supervised learning Setup : Testing ***

- The function h learn from traning is called "hypothesis"
- For testing, we hope to apply h on any data coming from the distribution P.

∴ "hypothesis" การสร้างสมมุติฐาน ที่เป็น function จะสามารถ ใช้ได้กับ data ที่มาจากการสุ่มตัวอย่าง population

IDEAL: $\forall (\vec{x}, y) \sim P, h(\vec{x}) = y$ > y is true label of $\vec{x}$

prediction term

## Generating hypothesis h:

① เลือก learning algorithm ที่เข้ากับปัญหา เรียกขั้นตอนนี้ว่า "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
เลือกจากการใช้ loss function

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

** Lower loss => better hypothesis

** gen average loss

# Example of loss functions

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

Note : $L_{0/1}(h, D)$ ; h → hypothesis
D → Data

$\sum_{(\vec{x}, y) \in D} d(h(\vec{x}), y)$ → ผลรวมทั้งหมดของ $(\vec{x}, y) \in D$
โดยที่จะไป hypothesis feature $\vec{x}$ และเทียบกับ y
ถ้าตรงจะเป็น 0 ถ้าไม่ตรงจะมีค่า 1 (คือเป็นข้อผิดพลาด)

โดยที่ $\frac{1}{|D|}$ คือ ส่วนของจำนวน Data ทั้งหมดเพื่อ
แปลงเป็น average loss

- 0/1 loss is commonly used in Classification problems

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

- 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

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

โดยที่ $h(\vec{x})$ ที่ outside data D จะมีค่าเป็น 0

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

generalization loss: $\varepsilon = \mathbb{E}_{(x,y) \sim P}[L(h, \{(\vec{x}, y)\})]$

Note : it's impossible to calculate $\varepsilon$, because
we can't asses P

# Estimating the generalization loss:

- split data

  $D_{TR}$ → train

  $D_{TE}$ → test

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

- learning from the training data set

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

- Evaluating loss via the testing data set

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

- claim As $|D_{TE}| \to +\infty$, $\varepsilon_{TE} \approx \varepsilon$

  (by the weak law of large number)

---

Supervised learning in pratice

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