Lecture 2: Supervised Learning Setup (Mathematical Formalization)

- Given labeled examples (our data), find the right prediction of an unlabel example

- Setup: Given a data set D

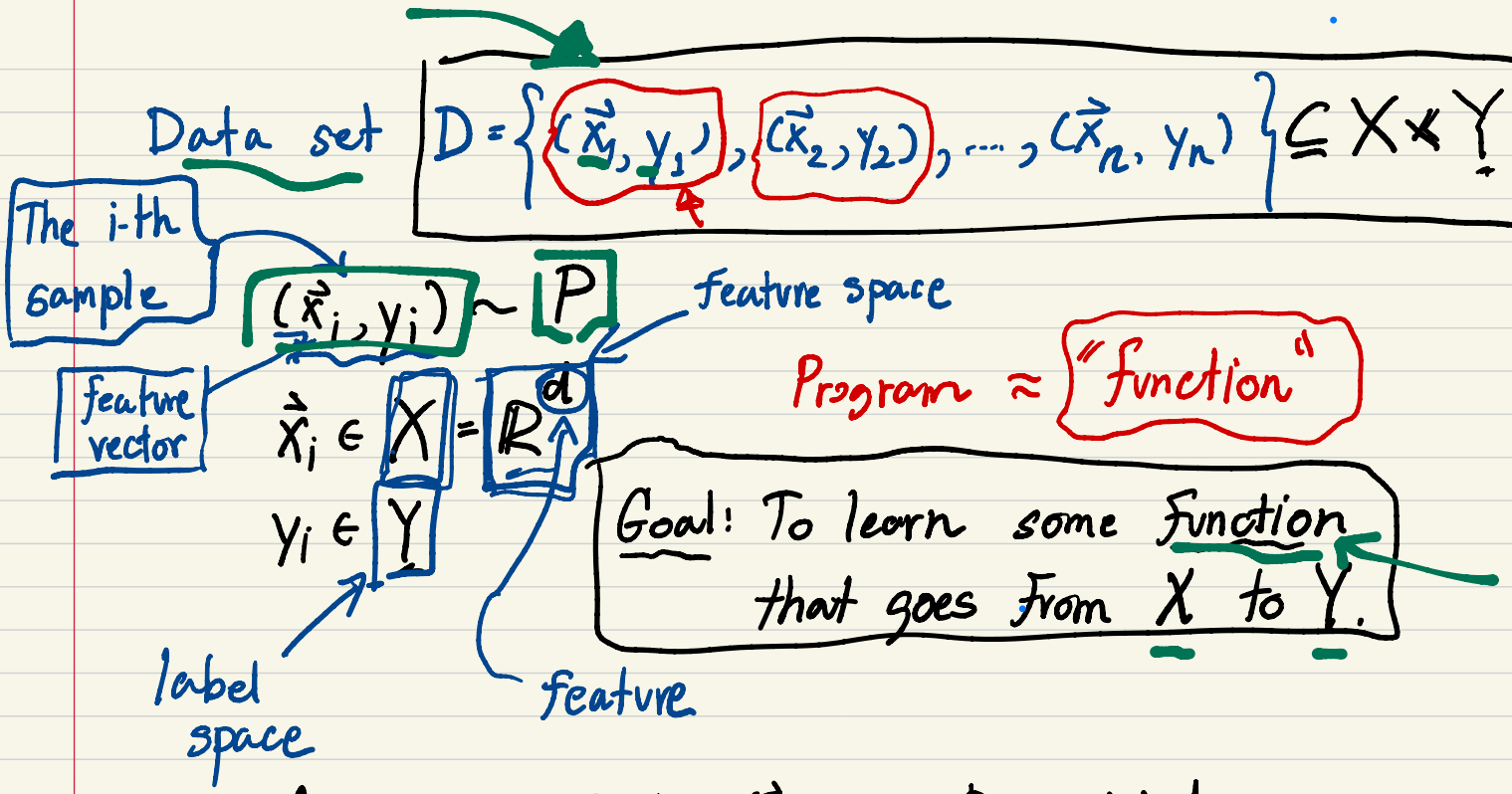
$$D = \{(\vec{x}_1, y_1), (\vec{x}_2, y_2) \ldots (\vec{x}_n, y_n)\}$$

$(\vec{x}_i, y_i) \sim P$ (unknown distribution)

i: number

$\vec{x}_i \in \mathbb{R}^d$ (e.g. spam Filter)

y_i:
- Binary classification: $y_i \in \{0, 1\}$ (e.g. face detection)
- Multi-class classification: $y_i \in \{0, 1, 2, \ldots, K\}$ (e.g. house price prediction)
- Regression: $y_i \in \mathbb{R}$



Assumption: Each $(\vec{x}_i, y_i) \in D$ is i.i.d.
("independent" and "identically distributed")

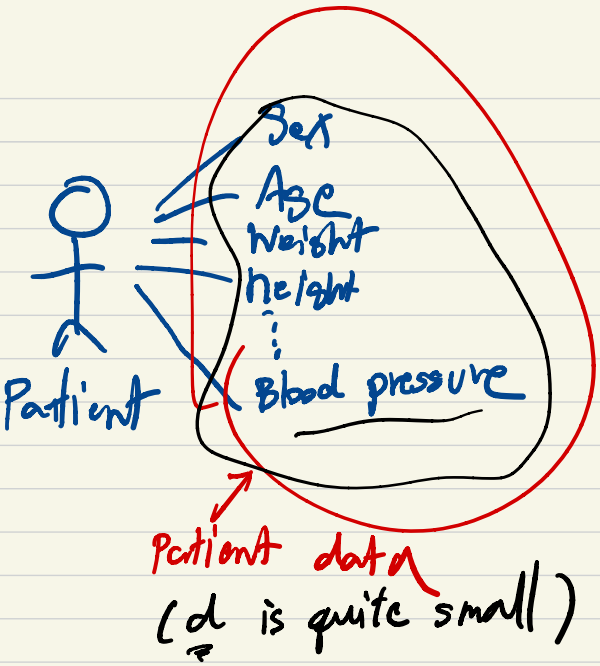
Examples of Feature Vectors

Patient Data

$$\vec{x}_i = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ \vdots \\ x_d \end{bmatrix} = \begin{bmatrix} 0 \\ 25 \\ 64 \\ 170 \\ \vdots \\ 110 \end{bmatrix}$$

- x_1: 0 — male / female
- x_2: 25 — age in years
- x_3: 64 — weight in kg
- x_4: 170 — height in cm
- x_d: 110 — Blood pressure

$$\vec{x}_i \in \mathbb{R}^d$$



(d is quite small)

Text document using Bag-of-Word (B.o.W.) representation

"Zebra Ant Ant"

"Ant Zebra Ant Ant"

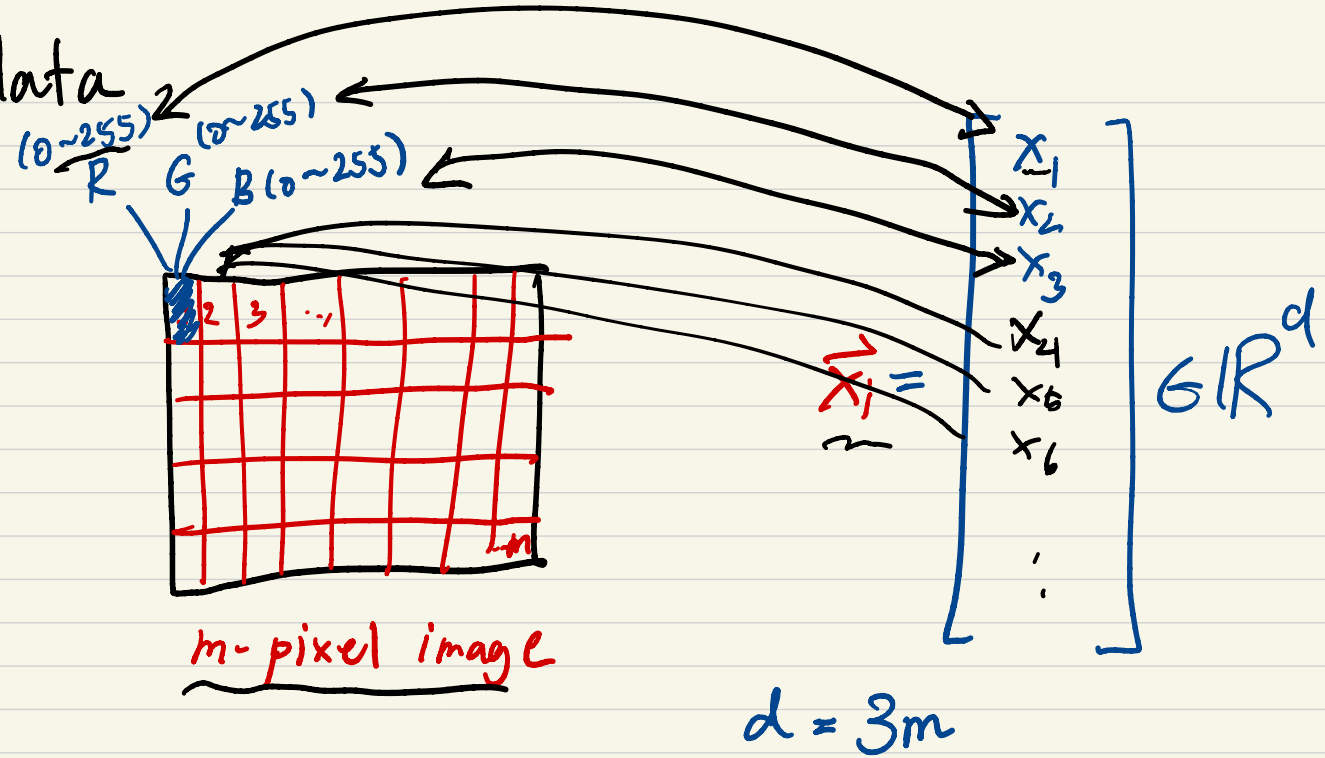
$$\vec{x}_i = \begin{bmatrix} 3 \\ \vdots \\ 1 \end{bmatrix}$$

- 3 — Ant
- 1 — Zebra

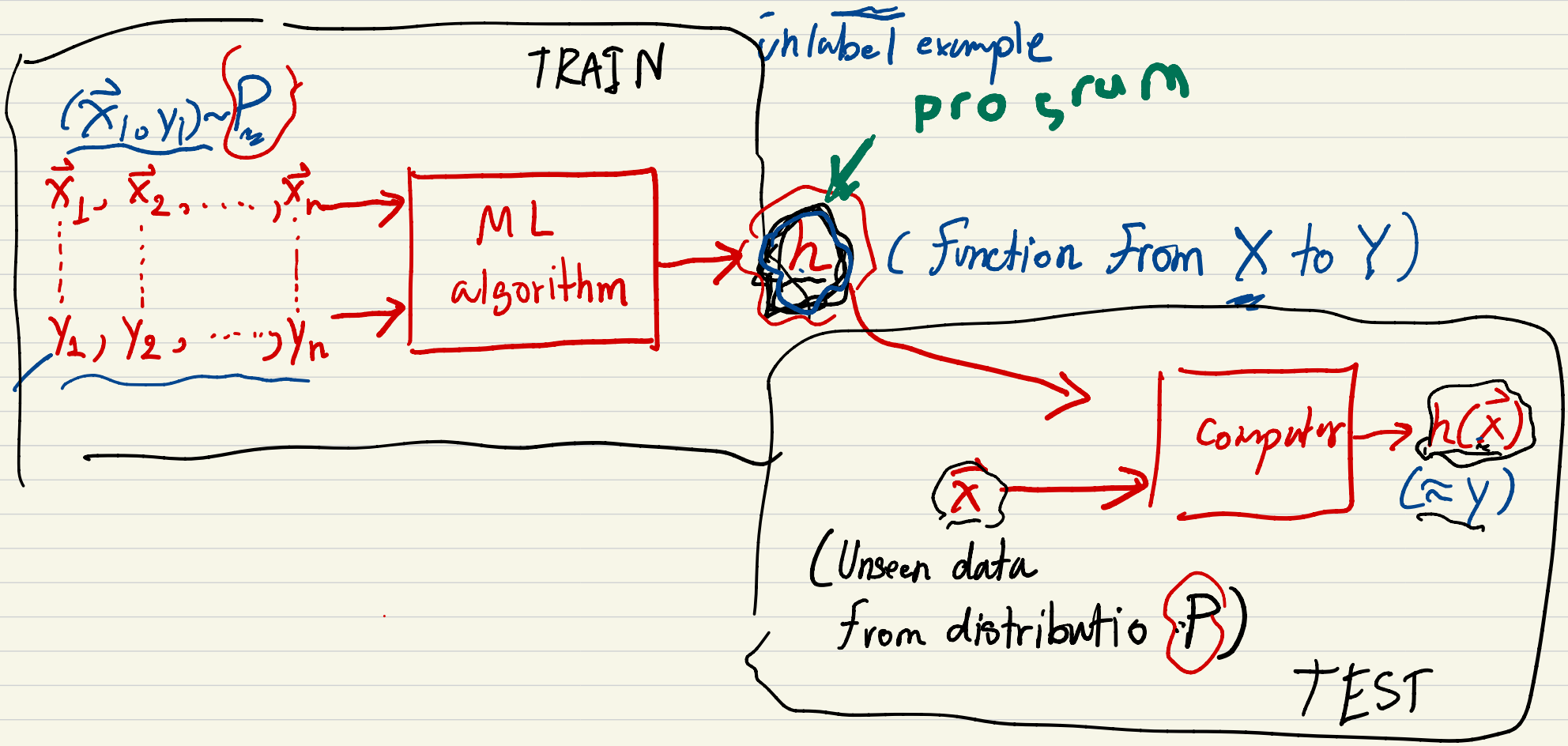
$$\vec{x}_i \in \mathbb{R}^d$$

(In English dictionary $d \approx 170k$).

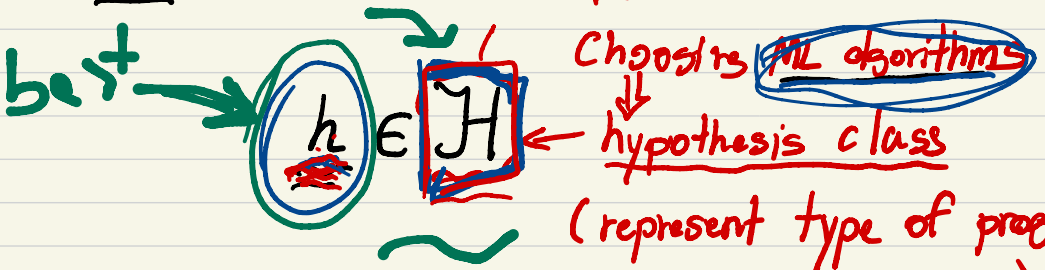
Image data



7MP Image $\Rightarrow$ $d = 3 \times 7M = \underline{21M}$


TRAIN
$(\vec{x}_1, y_1) \sim P$
$\vec{x}_1, \vec{x}_2, \ldots, \vec{x}_n$
$y_1, y_2, \ldots, y_n$
ML algorithm
h
(h/label example)
program
(Function from X to Y)
$\vec{x}$
Computer
$h(\vec{x})$
$(\approx y)$
(Unseen data from distribution P)
TEST

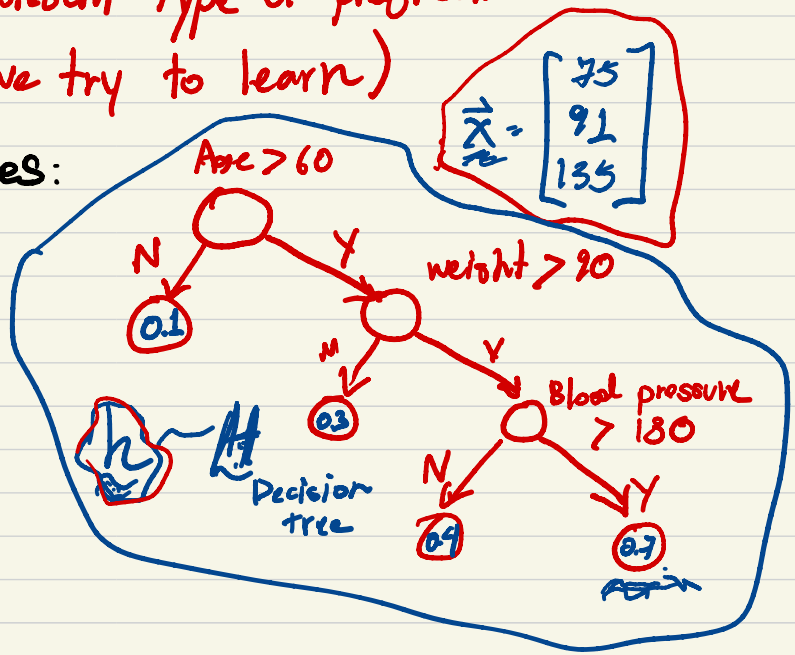
Goal: To learn function h from X to Y.



Choosing ML algorithms ⇓ hypothesis class (represent type of program we try to learn)

Automated job for defining $\mathcal{H}$. (Unnessary)

- Types of hypothesis classes:
  - Decision tree
  - Neural networks
  - Linear classifier
  - Etc.



Learning the hypothesis

① Defying $\mathcal{H}$: Choosing appropriate ML algorithm.
The set $\mathcal{H}$ defines the set of functions we could learn. (non-automated job)

② Learning the best $h \in \mathcal{H}$:

The ML algorithm will find the best hypothesis h within the class (automated job)

Ideally, we wish to find $h \in \mathcal{H}$ that makes the fewest mistakes within our data!!!

Loss Functions:

- Loss Functions are the Funtions that evaluate if one function is better then another.

- We use loss Functions to evaluate how bad any $\hat{h} \in H$ is on the data
  - The higer loss → the worst hypothesis
  - Normally, we normalize the loss by the number of examples (average loss**)

$L(h, D)$ bad ? good ?

zero

$h(x) = y$

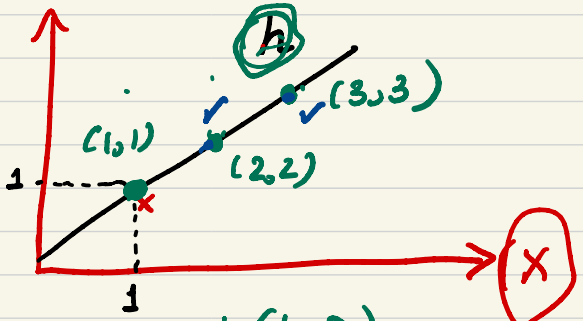
$\forall (x, y) \in D$

Common Loss Functions:

- 0/1-loss: $L(h, D) = \frac{1}{|D|} \sum_{\forall (\vec{x}, y) \in D} d(h(\vec{x}), y)$

$$d(h(\vec{x}), y) = \begin{cases} 0, & \text{if } h(\vec{x}) = y \\ 1, & \text{if } h(\vec{x}) \neq y \end{cases}$$

$h(\vec{x}) = x$



$L(h, D) = \frac{1}{3}(1+0+0)$

$= 0.33$

$L(h, D) = 0$