1/22/2019. L-1.

Supervised Learning.

Given a set of "training" examples (S).

e.g., (a) e-mails with labels (spam/not)

(b) images with labels (eg, cat/dog, $\{0, 1, \ldots, 9\}$).

(c) e-transactions (fraud/OK)

TASK :- Given 'S', and a new "unlabeled" example, predict its label.

Unsupervised Learning.

Given a dataset, derive interesting structures/ preprocess the data.

(a) clustering images
(b) summarization (news, google cards)
(c) dimensionality reduction.

Reinforcement Learning.

take step, get reward, continue to learn & play.
- chess, α-go, exploring robots.

SUPERVISED LEARNING.

Q:- How to represent data?

A training example is a pair of (feature, label).

(a) (image1, cat), (image2, dog), ...

(b) ((3 BR, 2BA, 3000 sqft, 14850), 300k), ...

Feature label.

Feature :- a representation/abstraction of the object.

- usually be represented as a vector in 'd' dimensions. $\mathbb{R}^d$

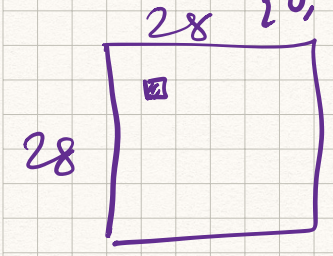
example :- MNIST dataset :- $\{0, \ldots 9\}$.

60,000 hand written digits, each 28×28.

gray scale image.

$\{0, \ldots, 255\}$ → 0 - black

→ 255 - white.



$\longrightarrow \{0, \ldots 255\}^{784}$. $\mathbb{R}^{784}$.

$\{0, \ldots 255\}^{28 \times 28}$.

Label :- $\{0, \ldots, 9\}$. (y)

60 K ⟶ training,

Set-up.

'S' ⟶ training set with 'n' examples.

$S = \{(\bar{x}_1, y_1), \ldots, (\bar{x}_n, y_n)\}$.

using 'S', come up with an algorithm, such that given a "new" unlabeled feature, we can guess its label "well enough".

GOAL :- do well on new points.

'T' - test set.

Learning Algorithm.

A function $h : \mathbb{R}^d \to \mathcal{Y}$.

ERM. Empirical Risk Minimization.

design algorithms that work well on 'S'.
& "hope" that you do well on new test data.

$$h_* : \mathbb{R}^{784} \to \{0, \ldots, 9\}$$

- 60K. → accuracy of h_* on 'S'.

- If S, T are generated by similar processes.
'n' is large enough
performance on 'T' ≈ performance on 'S'.
→ performance on 'S'!

---

overfitting.