Machine Learning
(機器學習)

Lecture 2: The Learning Problems

Hsuan-Tien Lin (林軒田)

htlin@csie.ntu.edu.tw

Department of Computer Science
& Information Engineering

National Taiwan University
(國立台灣大學資訊工程系)

Roadmap

1. **When** Can Machines Learn?

Lecture 2: The Learning Problems

- Learning with Different Output Space $\mathcal{Y}$
- Learning with Different Data Label y_n
- Learning with Different Protocol $f \Rightarrow (\mathbf{x}_n, y_n)$
- Learning with Different Input Space $\mathcal{X}$

Credit Approval Problem Revisited

age	23 years
gender	female
annual salary	NTD 1,000,000
year in residence	1 year
year in job	0.5 year
current debt	200,000

credit? $\{$**no**(-1), **yes**$(+1)\}$

unknown target function
$f: \mathcal{X} \to \mathcal{Y}$

(ideal credit approval formula)

training examples
$\mathcal{D}: (\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_N, y_N)$

(historical records in bank)

learning algorithm
$\mathcal{A}$

final hypothesis
$g \approx f$

('learned' formula to be used)

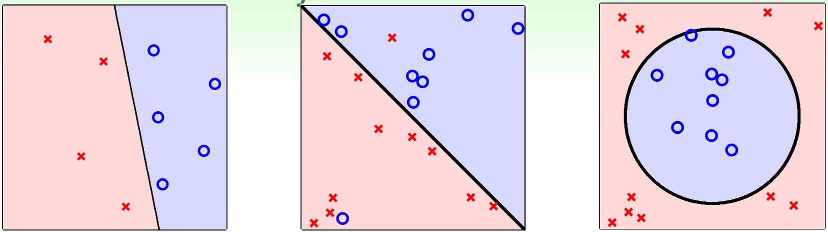
hypothesis set
$\mathcal{H}$

(set of candidate formula)

就是是非題

$\mathcal{Y} = \{-1, +1\}$: **binary classification**

More Binary Classification Problems

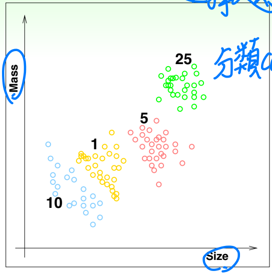


- credit approve/disapprove
- email spam/non-spam
- patient sick/not sick
- ad profitable/not profitable
- answer correct/incorrect (KDDCup 2010)

core and important problem with
many tools as **building block of other tools**

Multiclass Classification: Coin Recognition Problem

选擇題



分類coin

- classify US coins (1c, 5c, 10c, 25c) by (size, mass) *輸出有K種
- $\mathcal{Y} = \{1c, 5c, 10c, 25c\}$, or
  $\mathcal{Y} = \{1, 2, \cdots, K\}$ **(abstractly)**
- binary classification: special case with $K = 2$

Other Multiclass Classification Problems

- written digits $\Rightarrow 0, 1, \cdots, 9$ ← 10 digits
- emails $\Rightarrow$ spam, primary, social, promotion, update (Google)

many applications in practice

Multiclass Classification for Object Recognition: Which Fruit?

image classification

?

(image by Robert-Owen-Wahl from Pixabay)

apple

orange

strawberry

kiwi

(images by Pexels, PublicDomainPictures, 192635, Rob van der Meijden from Pixabay)

$$\mathcal{Y} = \{\text{apple}, \text{orange}, \text{strawberry}, \text{kiwi}\}$$

Which Fruits?

?: $\{$apple, orange, kiwi$\}$

(image by Michal Jarmoluk from Pixabay)

apple

orange

strawberry

kiwi

(images by Pexels, PublicDomainPictures, 192635, Rob van der Meijden from Pixabay)

multilabel classification: 複選題

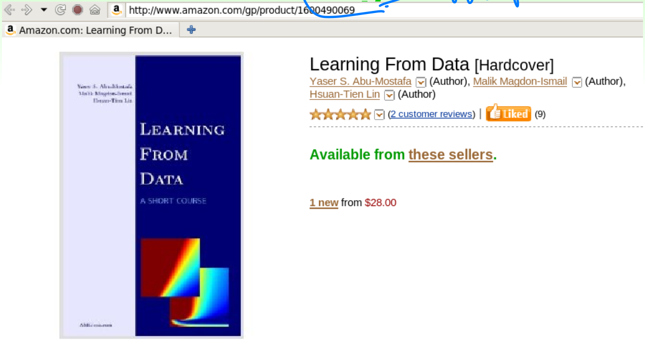
classify input to **multiple (or no)** categories

$\mathcal{Y} = 2^{\{\text{apple},\text{orange},\text{strawberry},\text{kiwi}\}}$

O O O O

X X X X

What **Tags**? 標籤



?: {machine learning, ~~data structure~~, data mining, ~~object oriented programming~~, artificial intelligence, ~~compiler~~, ~~architecture~~, ~~chemistry~~, textbook, ~~children book~~, ~~. . . etc.~~ }

another **multilabel** classification problem:
tagging input to multiple categories

Binary Relevance: Multilabel Classification via Yes/No

Tag: → become binary classification

binary classification

{yes, no}

multilabel w/ L classes: L yes/no questions

machine learning (Y), data structure (N), data mining (Y), OOP (N), AI (Y), compiler (N), architecture (N), chemistry (N), textbook (Y), children book (N), *etc.*

- Binary Relevance (BR): reduction (transformation) to **multiple isolated binary classification**
- disadvantages (addressed by more sophisticated models):
  - **isolation**—hidden relations not exploited * 不考慮類別間的关係
    (e.g. ML and DM highly correlated, ML subset of AI, textbook & children book disjoint)
  - **imbalanced**—few yes, many no ← class imbalanced

it's why binary classification is important.

BR for multilabel classification:
uses **binary classification** as a core tool

Regression: Patient Recovery Prediction Problem

- binary classification: patient features $\Rightarrow$ sick or not
- multiclass classification: patient features $\Rightarrow$ which type of cancer
- regression: patient features $\Rightarrow$ **how many days before recovery**
- $\mathcal{Y} = \mathbb{R}$ or $\mathcal{Y} = [\text{lower}, \text{upper}] \subset \mathbb{R}$ (bounded regression)
  —**deeply studied in statistics** *輸出是实数 ⇒ Regression

Other Regression Problems

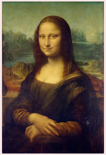
- company data $\Rightarrow$ stock price 预测股票
- climate data $\Rightarrow$ temperature

also core and important with many 'statistical' tools as **building block of other tools**

Sophisticated Output: Image Generation Problems

Style Transfer

Cycle GAN

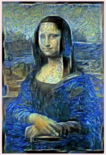


(Leonardo da Vinci, in Public Domain)

$+$

(Van Gogh, in Public Domain)

$\Rightarrow$



(Pjfinlay, with CC0)

all images are downloaded from Wikipedia

Other Image Generation Problems

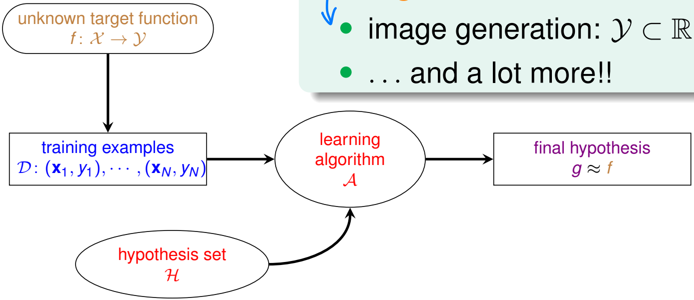
- noisy image $\Rightarrow$ clean image 無中生有
- low-resolution image $\Rightarrow$ high-resolution image

$\mathcal{Y}$: a 'manifold' $\subset \mathbb{R}^{w \times h \times c}$,
arguably **not just multi-pixel regression**

Mini Summary

Learning with Different Output Space $\mathcal{Y}$

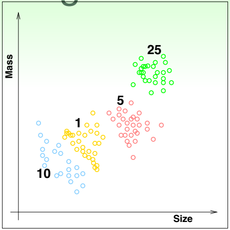
- **binary classification**: $\mathcal{Y} = \{-1, +1\}$
- multiclass classification: $\mathcal{Y} = \{1, 2, \cdots, K\}$
- multilabel classification: $\mathcal{Y} = 2^{\{1,2,\cdots,K\}}$
- **regression**: $\mathcal{Y} = \mathbb{R}$
- image generation: $\mathcal{Y} \subset \mathbb{R}^{w \times h \times c}$
- . . . and a lot more!!



core tools: binary classification and regression

Questions?

Supervised: Coin Recognition Revisited



unknown target function
$f\colon \mathcal{X} \to \mathcal{Y}$

training examples
$\mathcal{D}\colon (\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_N, y_N)$

learning algorithm
$\mathcal{A}$

final hypothesis
$g \approx f$

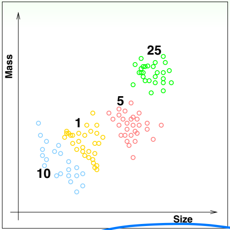
hypothesis set
$\mathcal{H}$

*有給正確答案

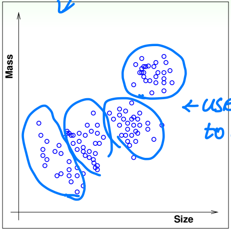
supervised learning:
every $\mathbf{x}_n$ **comes with corresponding** y_n

Unsupervised: Coin Recognition without y_n

no label ⇒ clusting.



supervised multiclass classification



← use distance to classify

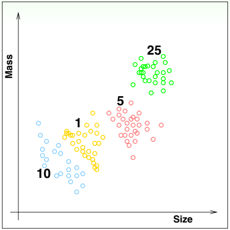
unsupervised multiclass classification $\Longleftrightarrow$ '**clustering**'

Other Clustering Problems

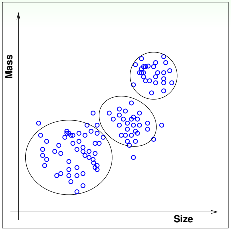
- articles ⇒ topics
- consumer profiles ⇒ consumer groups

clustering: a challenging but useful problem

Unsupervised: Coin Recognition without y_n



supervised multiclass classification



unsupervised multiclass classification
$\Longleftrightarrow$ '**clustering**'

Other Clustering Problems

- articles $\Rightarrow$ topics
- consumer profiles $\Rightarrow$ consumer groups

clustering: a challenging but useful problem

Unsupervised: Learning without y_n

Other Unsupervised Learning Problems

- clustering: $\{\mathbf{x}_n\} \Rightarrow \text{cluster}(\mathbf{x})$
  (≈ 'unsupervised multiclass classification')
  —i.e. articles ⇒ topics
- **density estimation**: $\{\mathbf{x}_n\} \Rightarrow \text{density}(\mathbf{x})$ $\in [0,1]$
  (≈ 'unsupervised bounded regression')
  —i.e. traffic reports with location ⇒ dangerous areas
  事故的密度分析
- **outlier detection**: $\{\mathbf{x}_n\} \Rightarrow \text{unusual}(\mathbf{x})$ abonormal
  (≈ extreme 'unsupervised binary classification')
  —i.e. Internet logs ⇒ intrusion alert
- . . . and a lot more!!

因為沒有給定的label，所以目標很分散

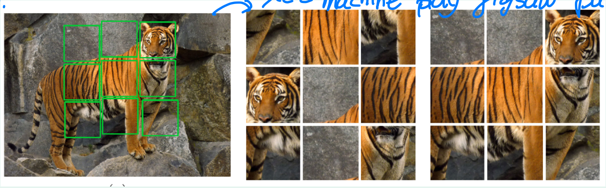
unsupervised learning: diverse, with possibly very different performance goals

Self-supervised: Unsupervised + Self-defined Goal(s)

jigsaw puzzle: pieces → full picture

learn by self.

let machine play jigsaw puzzle



(Figure 1 of Noroozi and Favaro,

Unsupervised Learning of Visual Representations by Solving Jigsaw Puzzles. ECCV 2016)

Other Popular Goals

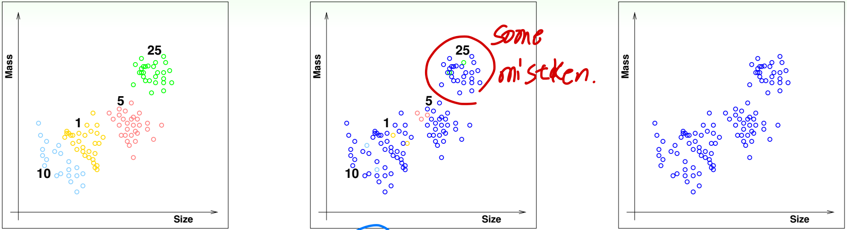
- colorization: grayscale image → colored image (填色)
- center word prediction: chunk of text → center word
- next sentence prediction: sentence A → is sentence B next?

First, do pre-train text, than fine-tune.

self-supervised learning: recipe to **learn 'physical knowledge'** before actual task

Semi-supervised: Coin Recognition with Some y_n

給你少數的真的label,去找cluster



supervised — **semi-supervised** — unsupervised (clustering)

Other Semi-supervised Learning Problems

- face images with a few labeled $\Rightarrow$ face identifier (Facebook)
- medicine data with a few labeled $\Rightarrow$ medicine effect predictor

半監督式

semi-supervised learning: leverage unlabeled data to avoid 'expensive' labeling

Weakly-supervised: Learning without True y_n

complementary label: $\bar{y}_n$ ('not' label) instead of y_n

← hard to get

用資訊比較少的label來train

easy to get. ⇒ semi-supervision.

(Figure 1 of Yu et al., Learning with Biased Complementary Labels, ECCV 2018)

Other Weak Supervisions

- partial label: a set Y_n that contains true y_n
- noisy label: y'_n, a noisy version of true y_n
- proportion label: aggregated statistics of a set of y_n

complementary label

weakly-supervised learning: another **realistic** family to reduce labeling burden

Reinforcement Learning

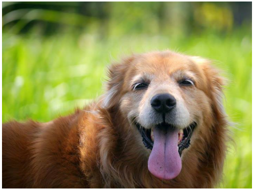
a 'very different' but natural way of learning

Teach Your Dog: Say 'Sit Down'

The dog pees on the ground. ← panelty

BAD DOG. THAT'S A VERY WRONG ACTION.

- cannot easily show the dog that y_n = sit when $\mathbf{x}_n$ = 'sit down'
- but can 'punish' to say $\tilde{y}_n$ = pee is wrong



Other Reinforcement Learning Problems Using $(\mathbf{x}, \tilde{y}, \text{goodness})$

- (customer, ad choice, ad click earning) ⇒ ad system ⇒ don't click ⇒ ☹
- (cards, strategy, winning amount) ⇒ black jack agent

reinforcement: learn with **'partial/implicit information'** (often sequentially) ⇒ a little bit like weekly supervisor.

Reinforcement Learning

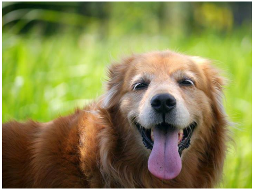
a 'very different' but natural way of learning

Teach Your Dog: Say 'Sit Down'

The dog sits down.

Good Dog. Let me give you some cookies.

- still cannot show y_n = sit when $\mathbf{x}_n$ = 'sit down'
- but can 'reward' to say $\tilde{y}_n$ = sit is good



Other Reinforcement Learning Problems Using $(\mathbf{x}, \tilde{y}, \text{goodness})$

- (customer, ad choice, ad click earning) $\Rightarrow$ ad system
- (cards, strategy, winning amount) $\Rightarrow$ black jack agent

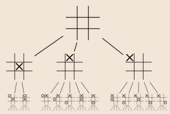
reinforcement: learn with **'partial/implicit information'** (often sequentially)

THE Most Well-known Reinforcement Learning Agent

AlphaGo ← can play game itself
self-training.

Non-ML Techniques

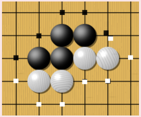
Monte C. Tree Search
≈ **move simulation** in brain Traditional.



(CC-BY-SA 3.0 by Stannered on Wikipedia)

ML Techniques

Deep Learning
≈ **board analysis** in human brain



(CC-BY-SA 2.0 by Frej Bjon on Wikipedia)

Reinforcement Learn.
≈ **(self)-practice** in human training

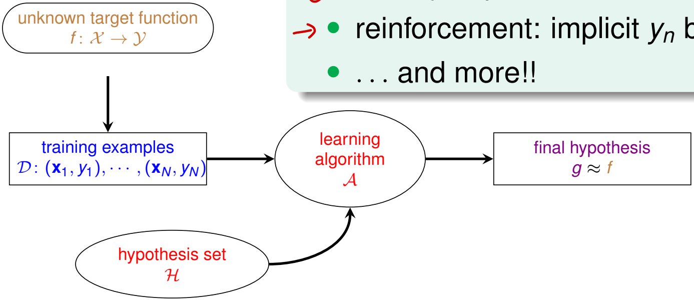
(Public Domain, from Wikipedia)

good AI: important to use the **right** techniques—ML **& others, including human**

Mini Summary

Learning with Different Data Label y_n

- **supervised**: all y_n
- unsupervised: no y_n
- self-supervised: self-defined y'_n from $\mathbf{x}_n$
- semi-supervised: some y_n
- weakly-supervised: no true y_n
- reinforcement: implicit y_n by goodness($\tilde{y}_n$)
- . . . and more!!

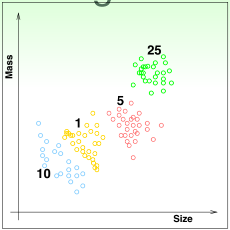


core tool: supervised learning

Questions?

Batch Learning: Coin Recognition Revisited

一整批送進去
data



unknown target function
$f\colon \mathcal{X} \to \mathcal{Y}$

training examples
$\mathcal{D}\colon (\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_N, y_N)$

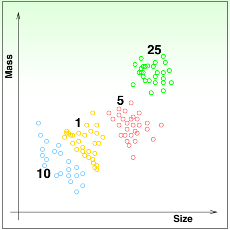
learning algorithm
$\mathcal{A}$

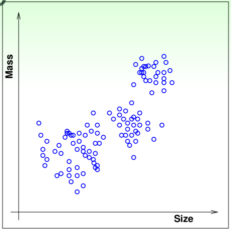
final hypothesis
$g \approx f$

hypothesis set
$\mathcal{H}$

batch supervised multiclass classification:
learn from **all known** data

More Batch Learning Problems





- batch of (email, spam?) $\Rightarrow$ spam filter
- batch of (patient, cancer) $\Rightarrow$ cancer classifier
- batch of patient data $\Rightarrow$ group of patients

✓ batch learning: **a very common protocol**

餵data給它，最一般的方式

Online: Spam Filter that 'Improves'

online learning.

- batch spam filter:
  learn with known (email, spam?) pairs, and predict with fixed g
- **online** spam filter, which **sequentially**: 一封一封送給ML
  1. observe an email $\mathbf{x}_t$
  2. predict spam status with current $g_t(\mathbf{x}_t)$
  3. receive 'desired label' y_t from user, and then update g_t with $(\mathbf{x}_t, y_t)$

Connection to What We Have Learned

- PLA can be easily adapted to online protocol (how?) ← use single mistake to update
- reinforcement learning is often done online (why?) → action → env → reward

online: hypothesis 'improves' through receiving data instances **sequentially**

rather than batch.

Online + Batch for Real-World Applications

model re-trained with historical daily batch data, incrementally or completely → **online** prediction for each request (request is in sequential) ↓ **batch** labelled data collected daily → model re-trained with historical daily batch data

purely online

- incremental update costly online ← very hard
- delayed labels hard to handle properly

purely batch

- cannot capture drifts/trends well.
- complete re-training possibly costly.

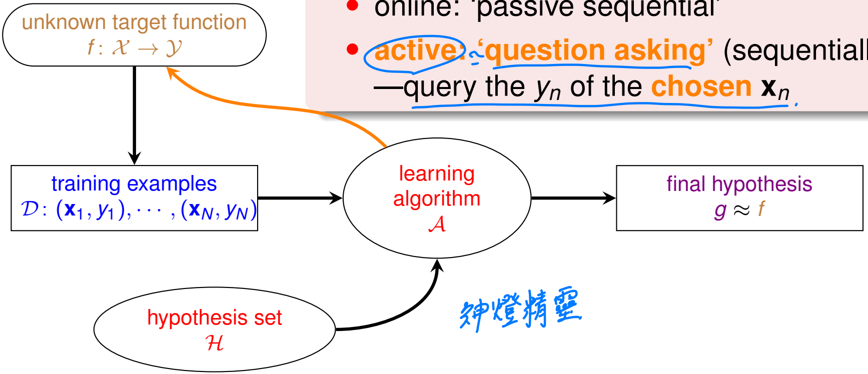
combine

real-world ML system different from textbook settings.

Active Learning: Learning by 'Asking'

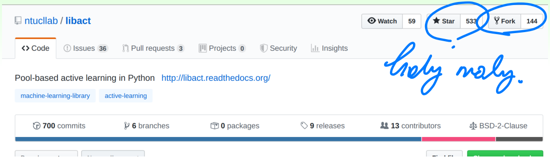
Protocol ⇔ Learning Philosophy

- batch: 'duck feeding'
- online: 'passive sequential'
- **active: 'question asking'** (sequentially) —query the y_n of the **chosen** $\mathbf{x}_n$



active: improve hypothesis with fewer labels (hopefully) by asking questions **strategically**

Making Active Learning More Realistic



holy moly.

open-source tool `libact` developed by NTU CLLab (Yang, 2017)

`https://github.com/ntucllab/libact`

- including many popular strategies
- received $>$ 500 **stars** and continuous **issues**

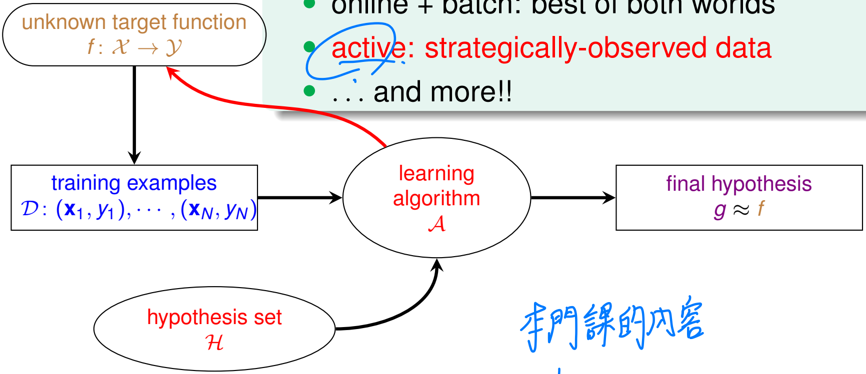
> "`libact` is a Python package designed to **make active learning easier** for real-world users"

Mini Summary

oneline: { data come in sequentially
incremental update.

Learning with Different Protocol $f \Rightarrow (\mathbf{x}_n, y_n)$

- **batch**: all known data
- online: sequential (passive) data
- online + batch: best of both worlds
- active: strategically-observed data
- . . . and more!!



本門課的內容
↓

core protocol: batch

Questions?

Credit Approval Problem Revisited

age	23 years
gender	female
annual salary	NTD 1,000,000
year in residence	1 year
year in job	0.5 year
current debt	200,000

unknown target function
$f: \mathcal{X} \to \mathcal{Y}$

(ideal credit approval formula)

training examples
$\mathcal{D}: (\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_N, y_N)$

(historical records in bank)

learning algorithm
$\mathcal{A}$

final hypothesis
$g \approx f$

('learned' formula to be used)

hypothesis set
$\mathcal{H}$

(set of candidate formula)

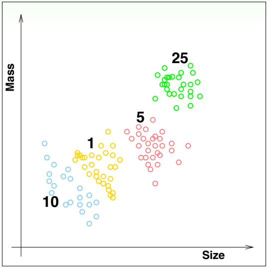
具体的feature

concrete features: each dimension of $\mathcal{X} \subseteq \mathbb{R}^d$
represents 'sophisticated physical meaning'

More on Concrete Features

use some concrete feature to do classify

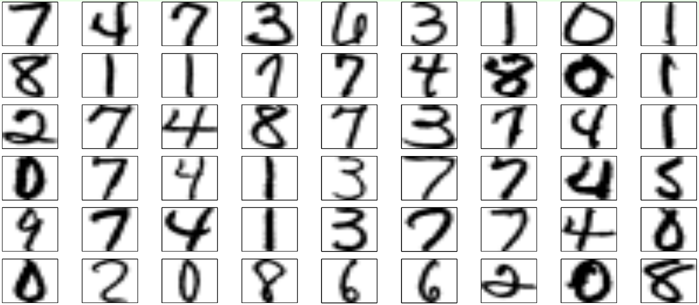
- **(size, mass)** for coin classification
- **customer info** for credit approval
- **patient info** for cancer diagnosis
- often including 'human intelligence' on the learning task



concrete features: the 'easy' ones for ML

因為feature都很 relliable

Raw Features: Digit Recognition Problem (1/2)



MIST
果蠅

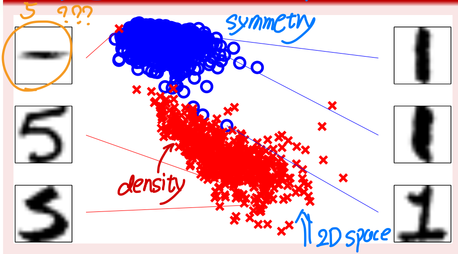
- digit recognition problem: features $\Rightarrow$ meaning of digit
- a typical supervised multiclass classification problem

Raw Features: Digit Recognition Problem (2/2)

有物理意義的

就是pixel值

by Concrete Features



$\mathbf{x} = $(symmetry, density)

↑ computer MSE of flipped images

by Raw Features

- 16 by 16 gray image $\mathbf{x} \equiv (0, 0, 0.9, 0.6, \cdots) \in \mathbb{R}^{256}$

  gray scale. $16 \times 16 = 256$
- '**simple** physical meaning'; thus more difficult for ML than concrete features

Other Problems with Raw Features

- image pixels, speech signal, etc.

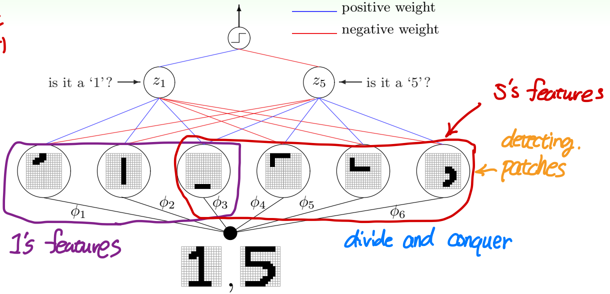
raw input $\xrightarrow{\text{feature engineering.}}$ concrete feature

raw features: often need human ('feature engineering') or machines to **convert to concrete ones**

Deep Learning: 'Automatic' Conversion from Raw to Concrete

Deep Learning也是一種抽取特徵的方法

raw feature → concrete feature



- layered extraction: **simple** to **complex** features
- natural for **difficult** learning task with **raw features**, like **vision**

deep learning: currently popular in **vision/speech**/...

Abstract Features: Rating Prediction Problem

Rating Prediction Problem (KDDCup 2011)

- given previous (userid, itemid, rating) tuples, predict the rating that some userid would give to itemid?
- a regression problem with $\mathcal{Y} \subseteq \mathbb{R}$ as rating and $\mathcal{X} \subseteq \mathbb{N} \times \mathbb{N}$ **as (userid, itemid)**
- '**no** physical meaning'; thus even more difficult for ML

Other Problems with Abstract Features

- student ID in online tutoring system (KDDCup 2010)
- advertisement ID in online ad system

只是一些編號
讓ML自己找意義

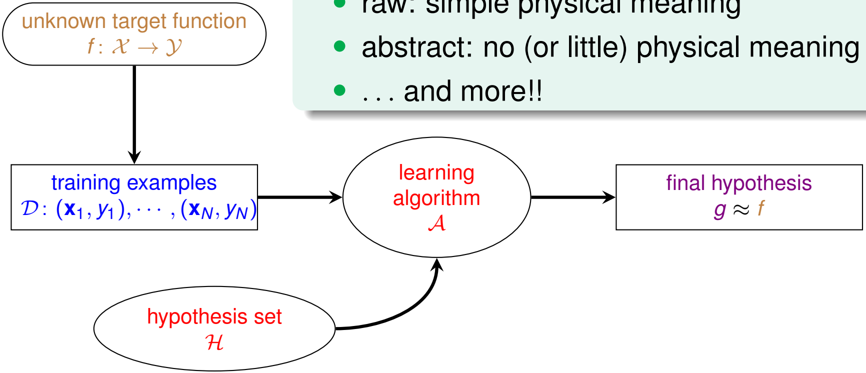
no physical meaning, e.g. student ID

abstract: again need '**feature conversion**/extraction/construction'

Mini Summary

Learning with Different Input Space $\mathcal{X}$

- **concrete**: sophisticated (and related) physical meaning
- raw: simple physical meaning
- abstract: no (or little) physical meaning
- . . . and more!!



'easy' input: concrete

Questions?

Summary

1. **When** Can Machines Learn?

Lecture 1: Basics of Machine Learning

Lecture 2: The Learning Problems

- Learning with Different Output Space $\mathcal{Y}$
  [classification], [regression], others
- Learning with Different Data Label y_n
  [supervised], un/semi/weakly-sup., reinforcement
- Learning with Different Protocol $f \Rightarrow (\mathbf{x}_n, y_n)$
  [batch], online, active
- Learning with Different Input Space $\mathcal{X}$
  [concrete], raw, abstract

- **next: learning is impossible?!**