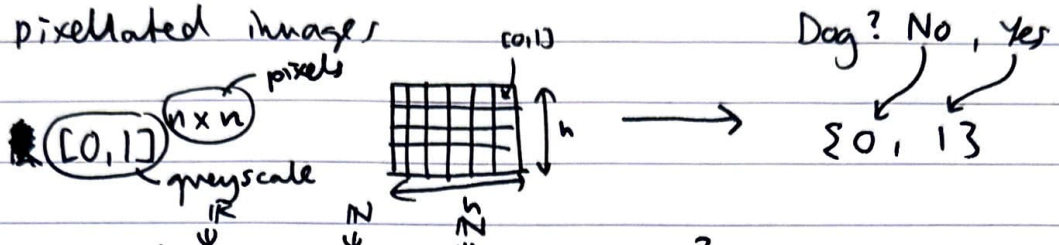
MARC TRUTER
HEFIN LAMBLEY

Machine learning reading group, talk 2 (Supervised learning)

X = set of inputs Y = set of outputs

① pixellated images



② {area of house, city, rooms} $\in \mathbb{R} \times \mathbb{N}^2 \longrightarrow$ Sale price $\in \mathbb{R}$

③ {height, weight, blood pressure} $\in \mathbb{R}^3 \longrightarrow$ Average steps a day $\in \mathbb{R}$

~~Deterministic or not? How are X and Y related?~~

Deterministic or not? How are X and Y related?

① is deterministic

$\exists f^* : X \to Y$, "Given an image, the label (Dog) No or Yes is determined (must be one or other)

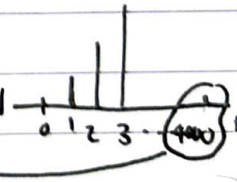
* No ↑ 0.5 dog or 0.6 dog.

②, ③ are not deterministic

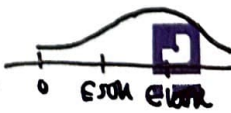
$\exists$ prob dist p on $X \times Y$ (think of conditional $Y|X$)

② "$\exists$ prob dist on X"

eg house very unlikely to have 4000 rooms



"$\exists$ prob dist on $Y|x$" = given fixed features x of house, what is prob of house selling at prob y



We just learnt : X, Y, p

Rmk: If deterministic, we can get a p from f^*.
Take $y|x$ as $f^*(x)$ with probability 1
(so can always work with p).

<u>However! Data sets in practice don't know p.</u>

"no correlation and from same dist"

But, we have n independent samples from p. — independant and identically distributed samples from p

Theory: $D_n(p) = \{\underbrace{(x_1, y_1), \ldots, (x_n, y_n)}_{\text{random set that depends on } p}\}$

Example ②: $D_n(p)$ = set of n random ← data samples from land registry data of house info against sale price

---

We write $\mathcal{A}$ = <u>algorithm</u>, which is a function

$$\mathcal{A} : (X \times Y)^{(n)} \longrightarrow \{f : X \to Y\}$$

($(X \times Y)^{(n)}$: $\downarrow D_n(p)$; n = # data points)

$\mathcal{A}$ trained on $D_n(p)$

$\Rightarrow$ algorithm returns a <u>deterministic</u> rule

① Yes or no diag

<u>exact answer</u> or appeared to

<u>prob of answer</u>

② prob dist of outcomes

● Example ②: House prices are non-deterministic, but still useful to have algo that spits out deterministic function ('good enough for most cases').

no choice / choice

Fix p, D_n, A, we define a loss function

choice → l: $Y \times Y \longrightarrow \mathbb{R}_{\geq 0}$

$(y, z) \longmapsto$ "error"

y: true label from data set; z: $f(x) := A(D_n(p))(x)$ (function for data set D_n; predicted value of y from x)

independent of A, "evaluate every algorithm A in same way"

Example ②: take $x = \{100\,m^2, \text{Coventry}, 3\}$ (area, city, rooms)

$\Rightarrow z = f(x) = A(D_n(p))(x)$ ← predicted value for given x by A

take y = true value from data set

Define loss function $l(y,z) = \|y - z\|$, e.g.

$l(y,z) = \|y - z\| = 10K$ ← decide if good or not

$\|$ €100K $\|$ £80K

Finally we can define risk

$$R(f) := \mathbb{E}_{(x,y)\sim P}[l(y, f(x))] = \text{"average loss over all data"}$$

($f = A(D_n(p))$; P dist. over $X \times Y$)

$$\left(= \int_{X \times Y} l(y, f(x))\, dp(x,y)\right)$$

Risk cannot be computed in practice as we do not have access to p, instead we use empirical risk

$$\hat{R}_n(f) := \frac{1}{n} \sum_{i=1}^{n} l(y_i, f(x_i))$$

↑ just taking over data $D_n(p)$ (what we know)

Result: $n \to \infty$, converge to risk.

Gist of supervised learning:

In practice, fix suitable l, we want to choose an f s.t $\hat{R}_n(f)$ is minimized.

over next four weeks : learn different f's and in each case how to minimise $\hat{R}_n(f)$