Intro-Classical-ML
Monday, 3 October 2022 9:48 PM

WELCOME!!!

Agenda:

1. Summary - learnt → not for exam
2. Overview - to learn
   - ML? MLE Vs SDE
   - ML Task?
   - Types of learnings.
3. Applications. (You can relate)
4. Ex211 Linear Regression (Next) [optional]

Already Learnt

1. Python & SKML libraries
2. Prob & stats.
3. Coordinate Geom / Linear Algebra.
4. Calculus & optimisation. [1-D]

AV → 1 Hour (random) CR
→ 1 cut (Sept).

graphs
→ stem
→ box
→ violin

Prob & stats.

Marginal Probability.

→ $P(\text{diabetic})$ ← ground

weight

$P(\text{diabetic} \mid w = 120\text{kg}) = 0.6$

$P(\text{diabetic} \mid \text{walking} \ \& \ h \geq 7\text{ft}) = 0.31$

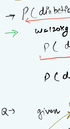
Q→ given $x^{(1)} \Rightarrow [x_1^{(1)}, x_2^{(1)}, \ldots x_d^{(1)}]$ → overview.

$P(\text{diabetic} \mid x^{(1)})$
→ "feature representation."

Linear Algebra.

"Email" → "Vector" x_1, x_2, x_3, x_4, x_5 → "Image"
→ d-dim ✓
→ 2-dim ✓
All levels

"Matrix"



$w \rightarrow \begin{bmatrix} 34 \\ 99 \end{bmatrix}$
$h \rightarrow$ $(2,)$

ML Vs. Classical Programming

ML: ↳ ability of a program to learn (AI Term) without being explicitly programed.

Given → 8, 14, 22, 16 . . .
odd → 1, 5, 13, 7, 33 . . .

Given → $M \cdot V \cdot 2 = 20$

future → 67

(Age, Strength)
Twist → Strong Programme → Price of house.
Rope → Weak
Rope →

ML Pipeline.

Training Data
↓
ML Algorithm / ML Programme
↓
Input → Trained Model → output
sign/Image (Strength) Supervised learning

$$\begin{bmatrix} {}^\circ C \rightarrow {}^\circ F \\ 31^\circ C \rightarrow {}^\circ F \\ [{}^\circ F = \frac{9}{5} \times C + 32] \end{bmatrix}$$

MLP?
logic is unknown.

Example.

"Data-Text"

E_1	E_2	E_3	....	E_n
↓	↓	↓		
S	NS	S		NS

"Keywords"
buy, lottery, here, send money, click here, free → Spam / Non-Spam

Drawbacks

→ Rigid Rules → Overcomes
→ Hard coded works.

A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E.

Task (T) → classify S/NS
Performance (P) (accuracy, Data size)
Experience (E) (Training Data) → Performance (P) (accuracy) → How much features do need emails

T	E	P
Number of classes to have fruit	Row data, Numerical data	Accuracy = Error
classifying fruits O → 0 K → 1 M → 2	P K M Attrib + label	P(O) → bad P(K) → high P(M) → low

Machine Learning

Types of learning

- Supervised
- Unsupervised (ML-2)
- Reinforcement learning

Types of Tasks

- Regression
- Classification
- Clustering
- Recommender Sys
- Time Series forecasting

Classification

→ Spam / Non-Spam.

$P(\text{Output} \mid T_1, T_2 \rightarrow \text{Binary classification}$
$P(\text{Output} \mid T_1, T_2, T_3, T_4 \rightarrow \text{Multi-class classification}$

$\mathcal{D} = \{(x^{(1)}, y^{(1)})_n ; x^{(i)} \in \mathbb{R}^d, y^{(i)} \in \{0,1\}\}$
features, Size, Binary classification

Regression

$\mathcal{D} = \{(x^{(1)}, y^{(1)})_n ; x^{(i)} \in \mathbb{R}^d, y^{(i)} \in \mathbb{R}\}$
features, Size, Price, Continuous

Training Data $(x^{(i)}, y^{(i)})$
Labelled data
"Supervised learning" → $y^{(i)}$ is available for training the model
"classification" → Regression
Image classification
Input data
(Images)
Label data
(Cat, Dog, Bird)
Training Data
ML Algo
Regression Tree
Regression
(Prediction)
Unseen Data
(Test Data)

① linear Regression → Used for regression.
② Logistic Regression → "classification"
↓
imp.

Unsupervised.

→ $\mathcal{D} = \{x^{(1)}, \ldots, x^{(n)} ; x^{(i)} \in \mathbb{R}^d\}$
→ "clustering" → to create groups.
→ similar objects.

[Problem Customers segmentation]
→ 1. Income
→ 2. Age
→ Behavior.
→ Loyalty.
→ Type of Service customer looks?

Casual - Students