# 下世代物聯網應用技術工作坊 - 機器學習

國立陽明交通大學MIPLab – 吳仁傑

# Course Summary

1. Validation
   - K-fold cross validation (*optional)
   - Leave-one-out cross validation
   - Holdout validation
2. Evaluation
   - Confusion matrix

# Course Summary

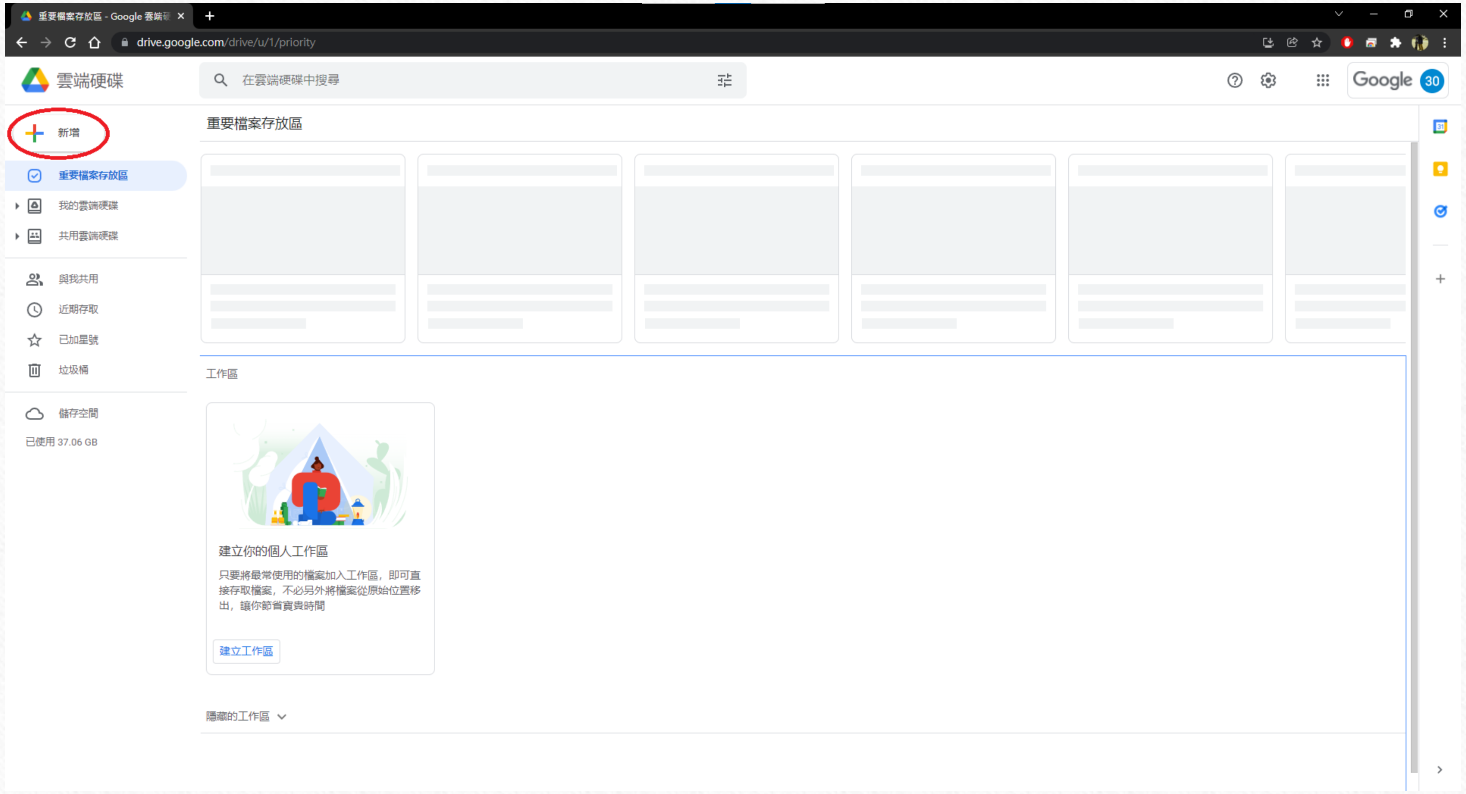
3. Supervised learning algorithm
    - Naive Bayes
    - Decision tree
    - Linear regression
    - Logistic regression
    - SVM

- Unsupervised learning algorithm
    - KNN

# Setting

1. 下載教材檔案 ML_workshop.zip
    - https://drive.google.com/file/d/1GAkqaSCaGXiR1kGKrQZ4BnYFq5gFE8VL/view?usp=sharing
2. 將檔案解壓縮後把整個ML_workshop資料夾上傳到自己的google雲端硬碟

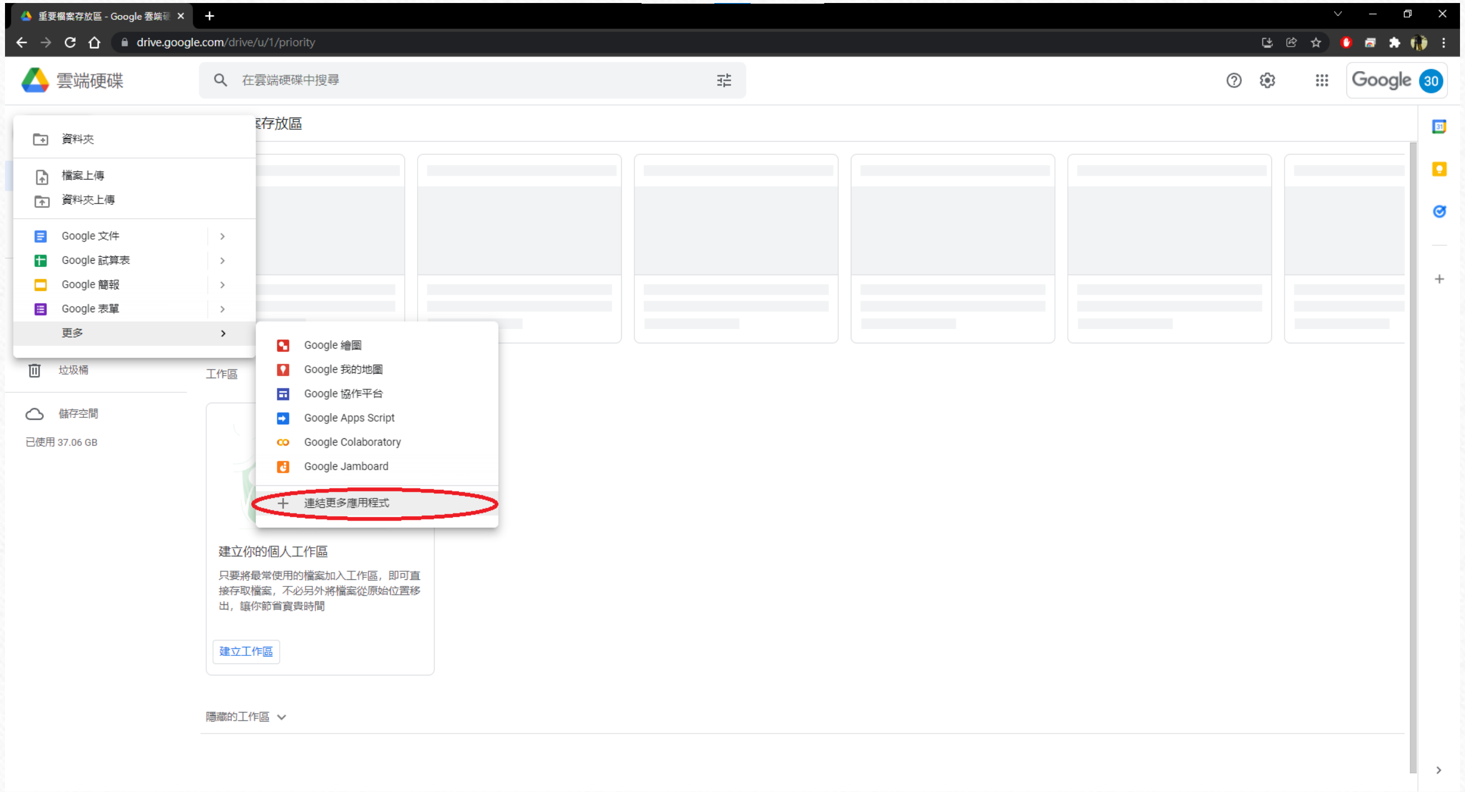
# Setting (安裝Google Colaboratory)



建立你的個人工作區

只要將最常使用的檔案加入工作區，即可直接存取檔案，不必另外將檔案從原始位置移出，讓你節省寶貴時間

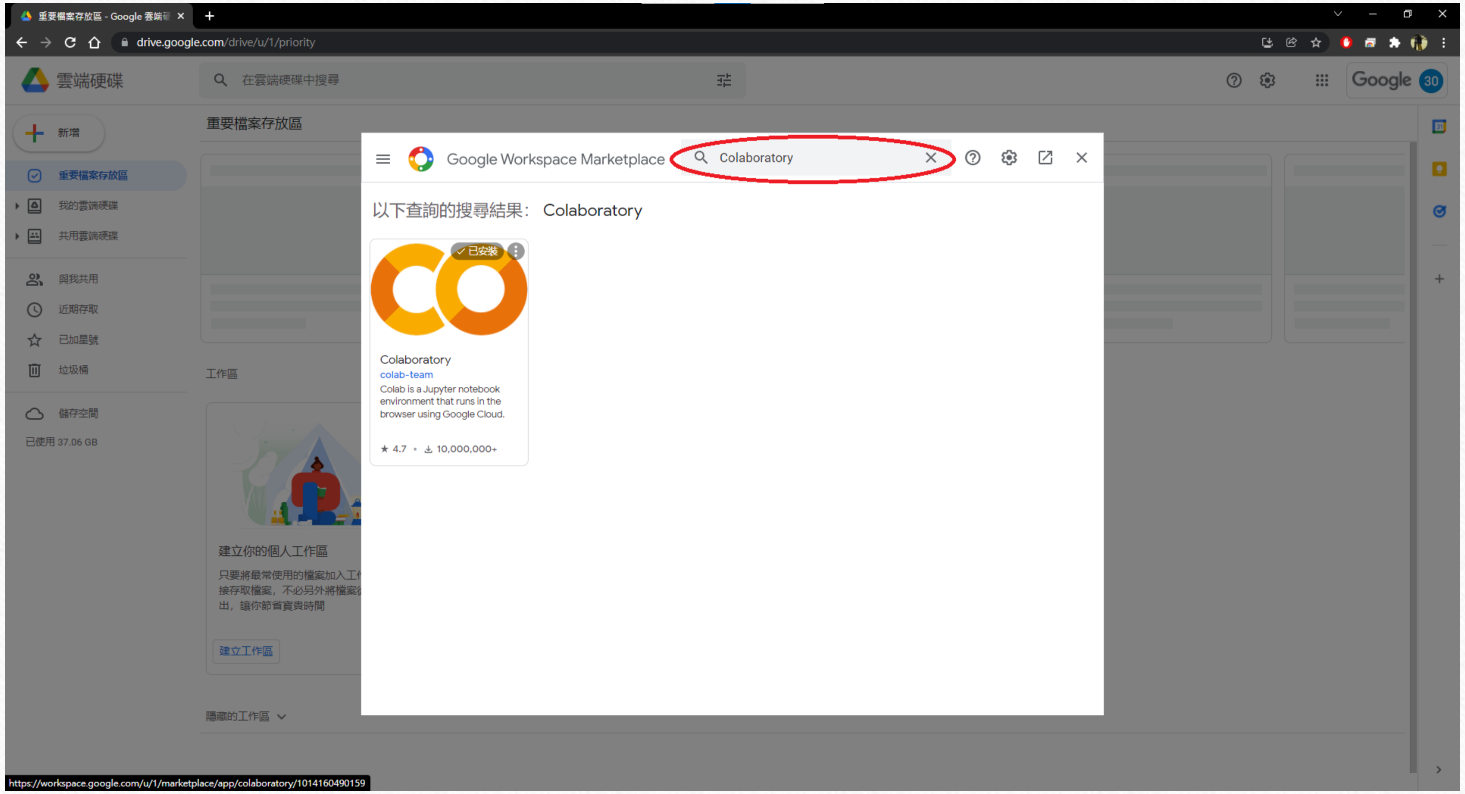
# Setting (安裝Google Colaboratory)



建立你的個人工作區

只要將最常使用的檔案加入工作區，即可直接存取檔案，不必另外將檔案從原始位置移出，讓你節省寶貴時間

# Setting (安裝Google Colaboratory)



drive.google.com/drive/u/1/priority

在雲端硬碟中搜尋

Google Workspace Marketplace

Colaboratory

以下查詢的搜尋結果： Colaboratory

Colaboratory
colab-team
Colab is a Jupyter notebook environment that runs in the browser using Google Cloud.
★ 4.7 • 10,000,000+

建立你的個人工作區

只要將最常使用的檔案加入工作 接存取檔案，不必另外將檔案 出，讓你節省寶貴時間

已使用 37.06 GB

https://workspace.google.com/u/1/marketplace/app/colaboratory/1014160490159

# Linear regression

1. Input : $X = [x_1, \cdots, x_n]$ $(\in \mathbb{R}^n)$
   - Every $x_k$ is a quantitative variable
2. Output : $\widehat{\mathrm{Y}} \in \mathbb{R}$
3. Parameters : $W = [w_0, w_1, \cdots, w_n]$ $(\in \mathbb{R}^{n+1})$

# Linear regression

4. Formula : $\hat{Y} = w_0 + w_1 x_1 + \cdots + w_n x_n$

5. Loss(Error) function: $\mathrm{L} = \left(Y - \hat{Y}\right)^2$

6. Optimization : $\operatorname{argmin}_w \frac{1}{m} \sum \left(Y - \hat{Y}\right)^2$

    - $m$ pieces of data
    - Find $W$ such that we have minimal average loss.

# Naive Bayes classifier

- Bayes' theorem : $P(Y|x_1, \cdots, x_n) = \frac{P(Y)P(x_1, \cdots, x_n|Y)}{P(x_1,\cdots,x_n)}$
- Naïve assumption (conditional independence) :
  $\mathrm{P}(x_i|y, x_1, \cdots, x_{i-1}, x_{i+1}, \cdots, x_n) = P(x_i|y)$
- New formula : $P(Y|x_1, \cdots, x_n) = \frac{P(Y)\Pi_{i=1}^{n}(x_i|Y)}{P(x_1,\cdots,x_n)} \propto P(Y)\Pi_{i=1}^{n}P(x_i|Y)$

# Naive Bayes classifier

- Making prediction according to the new formula :

$$\text{argmax}_Y \; P(Y|x_1, \cdots, x_n) \propto P(Y)\Pi_{i=1}^{n} P(x_i|Y)$$

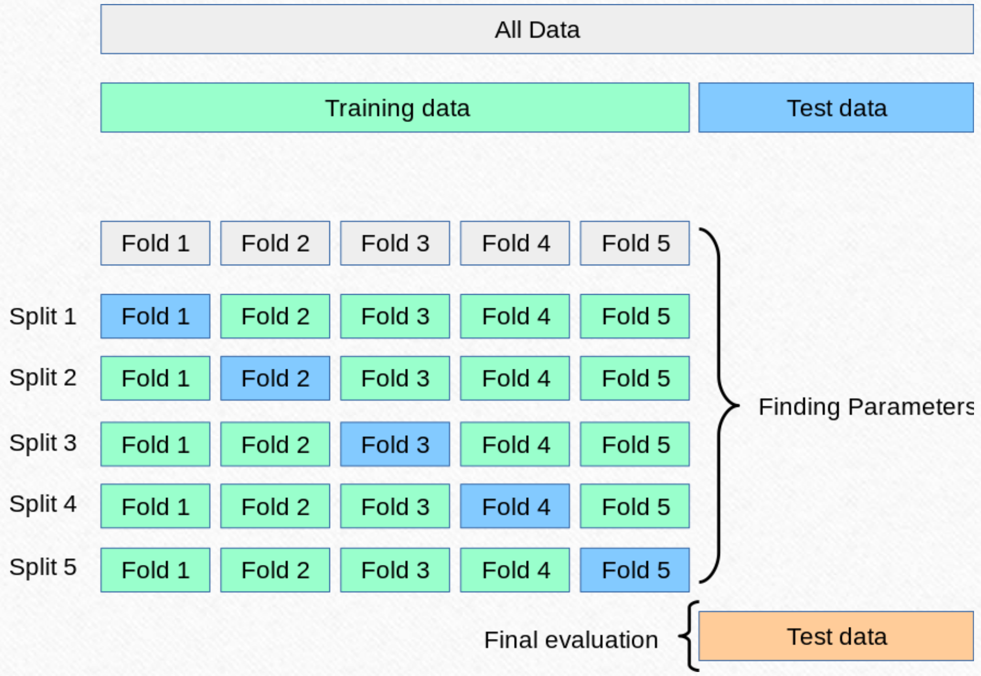
1. $Y \in \{y_1, \cdots, y_k\}$ : a categorical variable
2. $x_i$ : an input feature (either categorical or quantitative)
3. $P(Y)$ : estimated by training dataset
4. $P(x_i|Y)$ : make some assumption of distribution (ex: Gaussian distribution, multinomial distribution, …)

# K-fold cross validation

- 3 roles of dataset
  - Training set : for train model
  - Validation set : for hyperparameters tuning
  - Testing set : for model performance evaluation

# K-fold cross validation

- Hyperparameters is a parameter whose value is used to control the learning process
- Find hyperparameter settings that have best average performance of each split.



All Data

Training data | Test data

|  |  |  |  |  |  |
|---|---|---|---|---|---|
|  | Fold 1 | Fold 2 | Fold 3 | Fold 4 | Fold 5 |
| Split 1 | Fold 1 | Fold 2 | Fold 3 | Fold 4 | Fold 5 |
| Split 2 | Fold 1 | Fold 2 | Fold 3 | Fold 4 | Fold 5 |
| Split 3 | Fold 1 | Fold 2 | Fold 3 | Fold 4 | Fold 5 |
| Split 4 | Fold 1 | Fold 2 | Fold 3 | Fold 4 | Fold 5 |
| Split 5 | Fold 1 | Fold 2 | Fold 3 | Fold 4 | Fold 5 |

Finding Parameters

Final evaluation | Test data