

Day 34

機器學習

訓練/測試集切分



# 本日知識點目標

---



目標  
知識點

了解機器學習中資料的切分



獲得  
知識點

完成今日課程後你應該可以了解

- 為何要進行訓練/測試集切分
- 不同的切分方法以及意義

# 為何需要切分訓練/測試集

---

- 機器學習模型需要資料才能訓練
- 若將手上所有資料都送進模型訓練，這樣就沒有額外資料來評估模型訓練情形！
- 機器學習模型可能會有過擬合 (Over-fitting) 的情形發生，需透過**驗證/測試集**評估模型是否過擬合



圖片來源：[slowlydownward](#)



# 使用 Python Scikit-learn 進行資料切分

Python 中的機器學習套件 Scikit-learn 提供了一個 train-test split 函數，可以幫助快速對資料進行切分

## `sklearn.model_selection.train_test_split`

```
sklearn.model_selection. train_test_split (*arrays, **options)
```

[\[source\]](#)

Split arrays or matrices into random train and test subsets

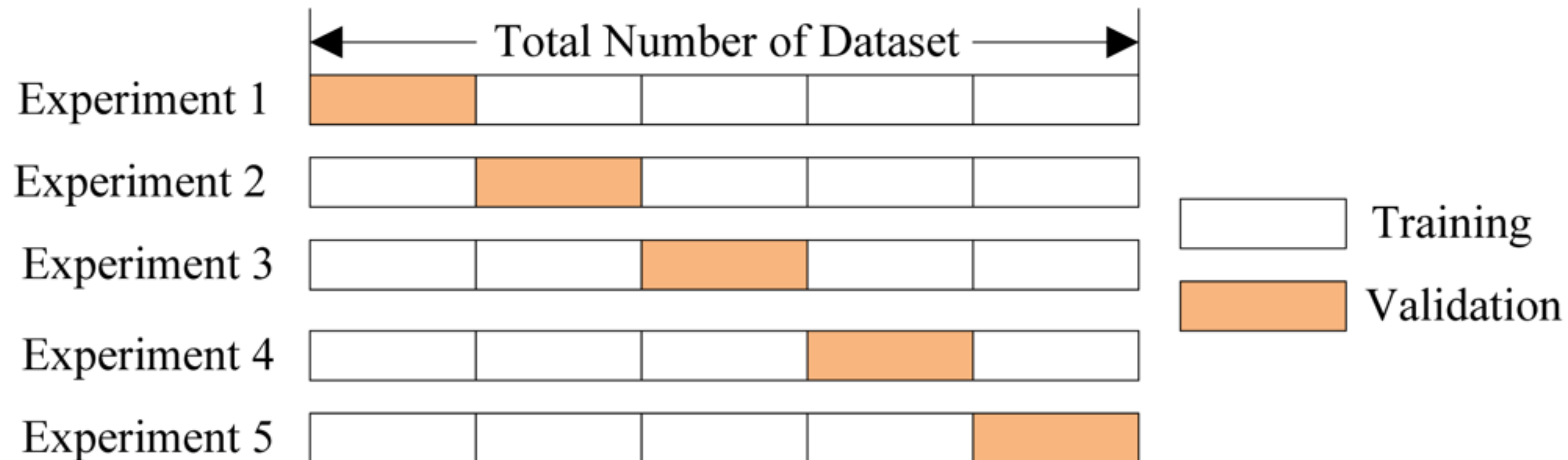
Quick utility that wraps input validation and `next(ShuffleSplit().split(X, y))` and application to input data into a single call for splitting (and optionally subsampling) data in a oneliner.

Read more in the [User Guide](#).

# K-fold Cross-validation

- 若僅做一次訓練/測試集切分，有些資料會沒有被拿來訓練過，因此後續就有 cross-validation 的方法，可以讓結果更為穩定，K 為 fold 數量
- 每筆資料都曾經當過一次驗證集，再取平均得到最終結果。

下圖為 5-fold cross-validation



# 使用 Python Scikit-learn 進行 Cross-validation

Python 中的機器學習套件 Scikit-learn 提供了一個 KFold 函數，可以幫助你快速運用 Cross-validation

**`sklearn.model_selection.KFold`**

```
class sklearn.model_selection. KFold (n_splits='warn', shuffle=False, random_state=None)
```

[\[source\]](#)

K-Folds cross-validator

Provides train/test indices to split data in train/test sets. Split dataset into k consecutive folds (without shuffling by default).

Each fold is then used once as a validation while the k - 1 remaining folds form the training set.

Read more in the [User Guide](#).

# 常見問題

---

Q: 驗證集 (validation set) 與測試集 (testing set) 有甚麼差異？

A: 驗證集常用來評估不同超參數或不同模型的結果。而測試集則是在機器學習專案開始前先保留一小部分資料，專案進行中都不能使用，最終再拿來做測試，Kaggle 競賽的最終排名也是根據測試集的分數來評定。

# 解題時間 It's Your Turn

請跳出PDF至官網Sample Code & 作業  
開始解題

