

補
第四單元
集成

混合泛化(Blending)



陳明佑

出題教練

知識地圖 機器學習- 參數調整 - 超參數調整與優化

參數調整

監督式學習

Supervised Learning

前處理
Processing

探索式
數據分析
Exploratory Data Analysis

特徵
工程
Feature Engineering

模型
選擇
Model selection

參數調整
Fine-tuning

集成
Ensemble

非監督式學習

Unsupervised Learning

分群
Clustering

降維
Dimension Reduction

參數調整 Fine-tuning

混合泛化
Blending

堆疊泛化
Stacking

本日知識點目標

- 資料工程中的集成，有哪些常見的內容？
- 混合泛化為什麼能提升預測力，使用上要注意什麼問題？

什麼是集成

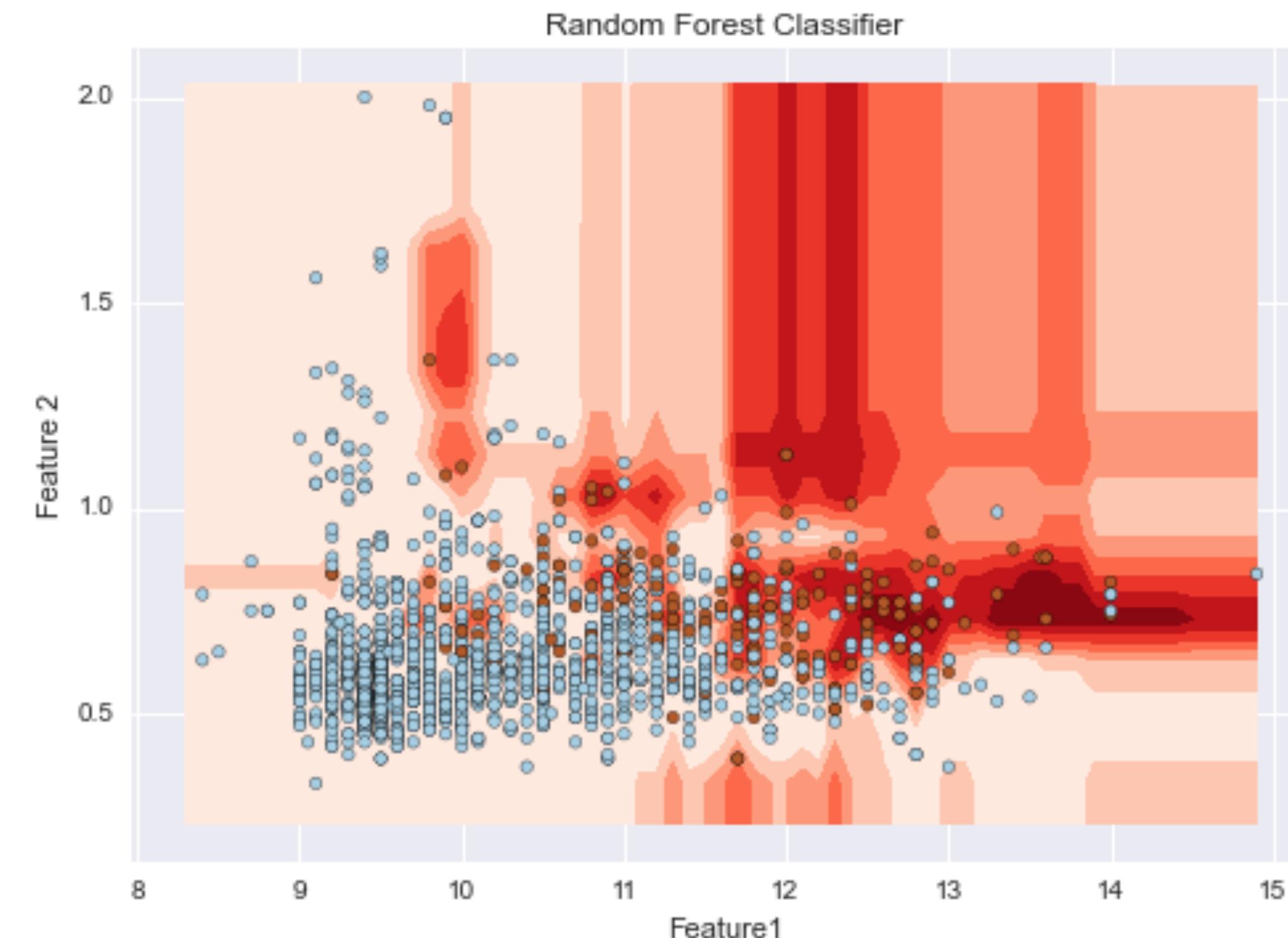
- 集成是使用不同方式，結合多個/多種不同分類器，作為綜合預測的做法統稱
- 將模型截長補短，也可說是機器學習裡的 和議制 / 多數決



- 其中又分為資料面的集成：如裝袋法(Bagging) / 提升法(Boosting)
- 以及模型與特徵的集成：如混合泛化(Blending) / 堆疊泛化(Stacking)

資料面集成：裝袋法 (Bagging)

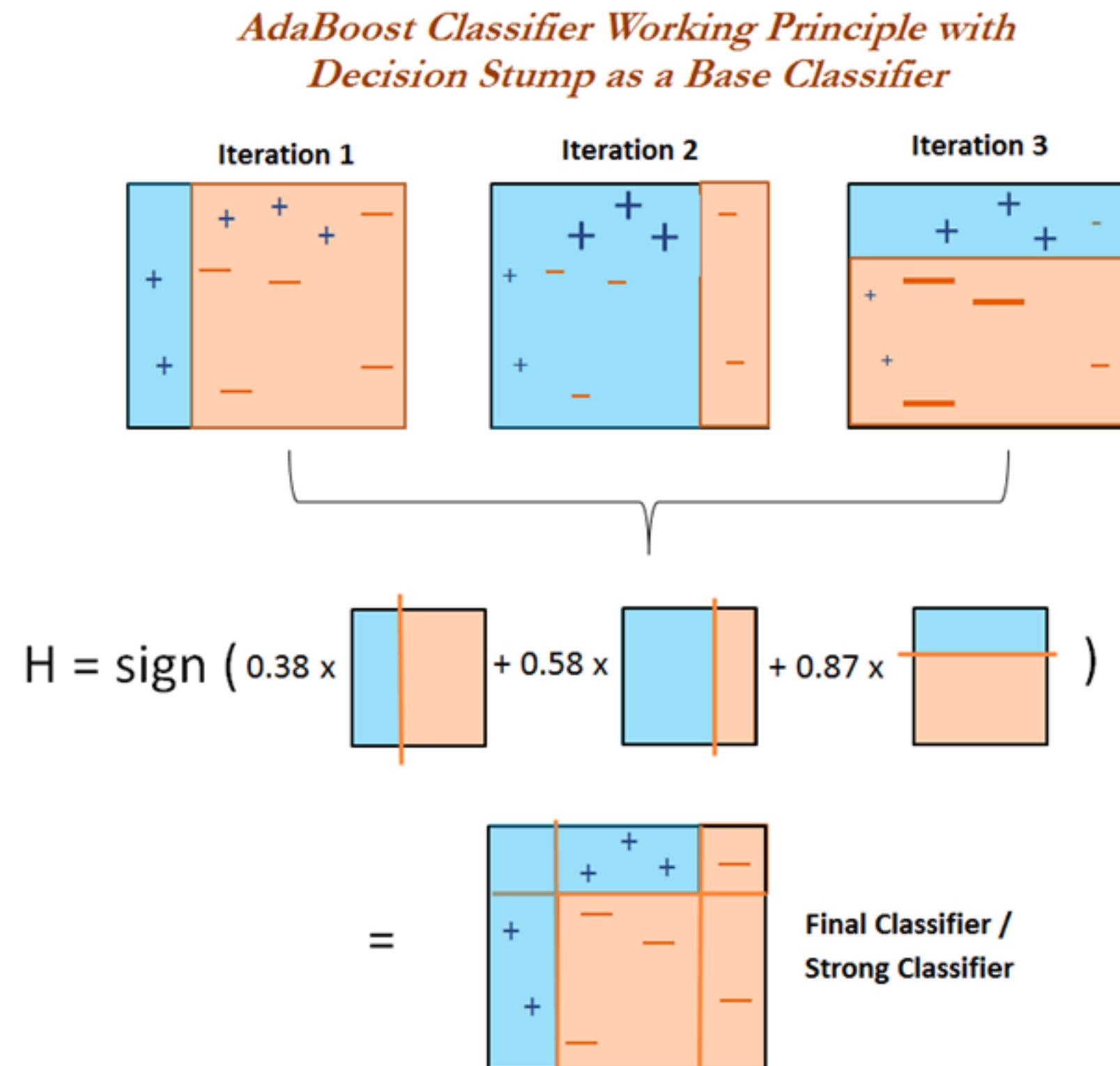
- 裝袋法顧名思義，是將資料放入袋中抽取，每回合結束後全部放回袋中重抽
- 再搭配弱分類器取平均/多數決結果，最有名的就是前面學過的**隨機森林**



圖片來源：stackexchange.

資料面集成：提升法 (Boosting)

- 提升法則是由之前模型的預測結果，去改變資料被抽到的權重或目標值
- 將錯判資料被抽中的機率放大，正確的縮小，就是**自適應提升** (AdaBoost, Adaptive Boosting)
- 如果是依照估計誤差的殘差項調整新目標值，則就是**梯度提升機** (Gradient Boosting Machine) 的作法，只是梯度提升機還加上用梯度來選擇決策樹分支

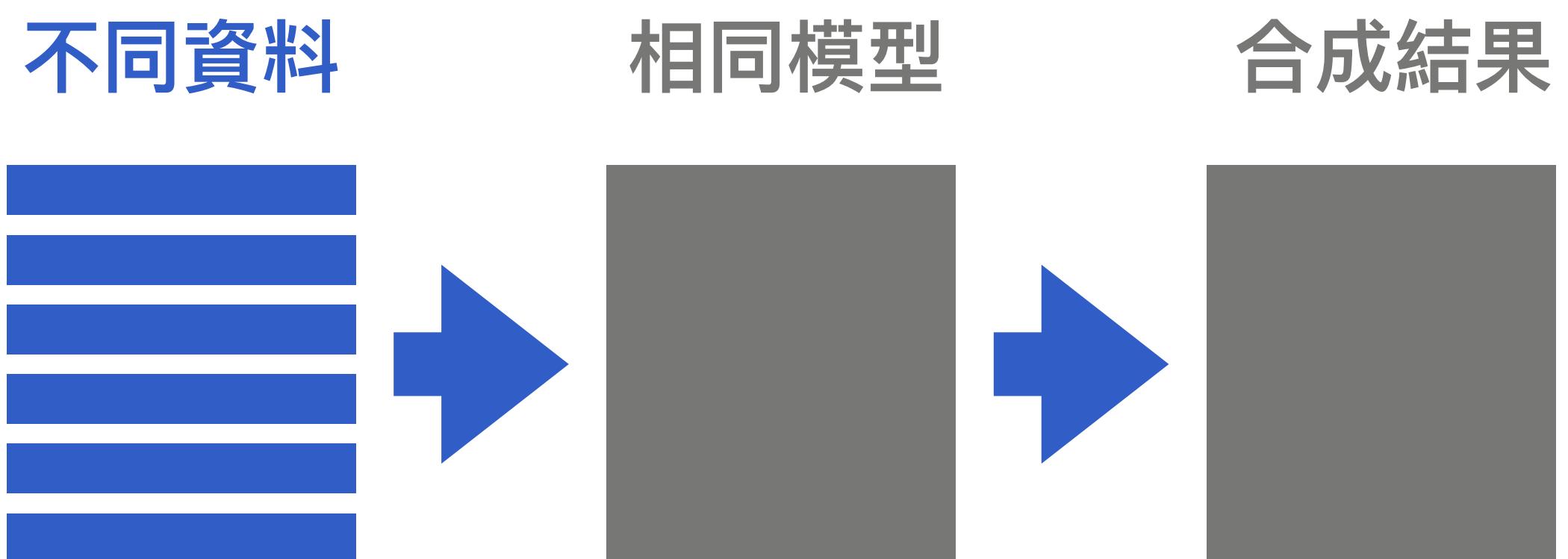


資料集成 v.s. 模型與特徵集成

- 兩者雖然都稱為集成，其實適用範圍差異很大，通常不會一起提及
- 這裡為了避免同學混淆，在這邊將兩者做個對比
- 資料集成**

Bagging / Boosting

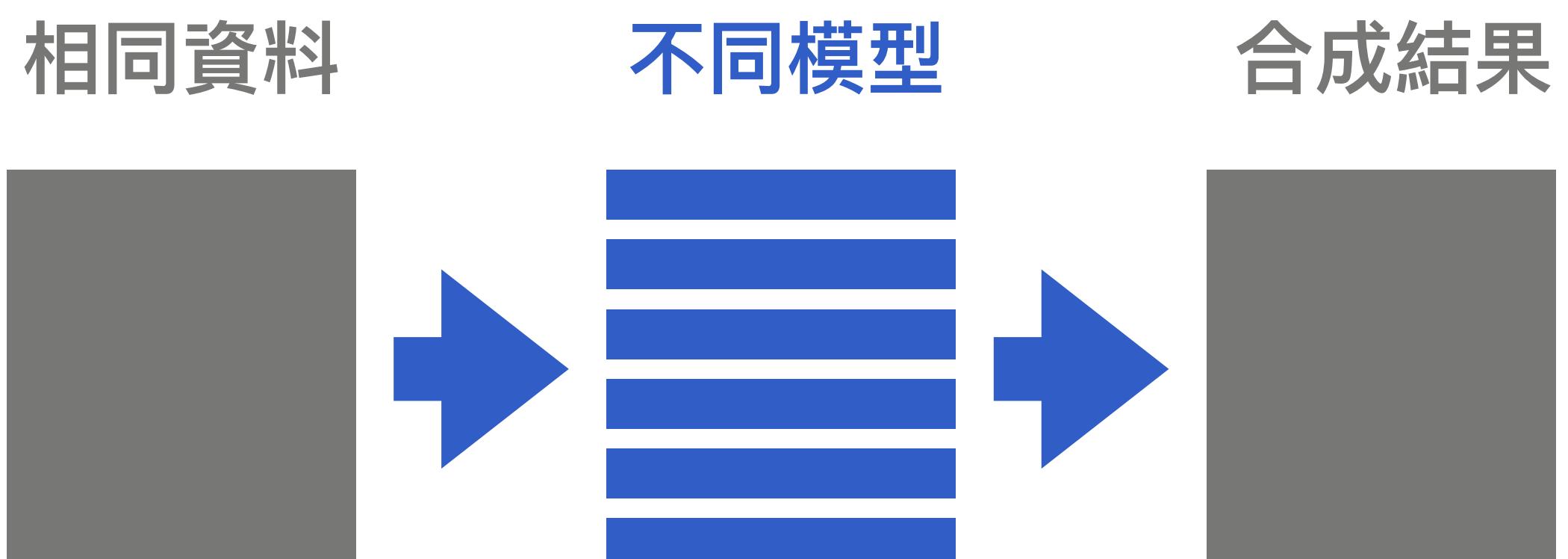
- 使用不同訓練資料 + 同一種模型，多次估計的結果合成最終預測



- 模型與特徵集成**

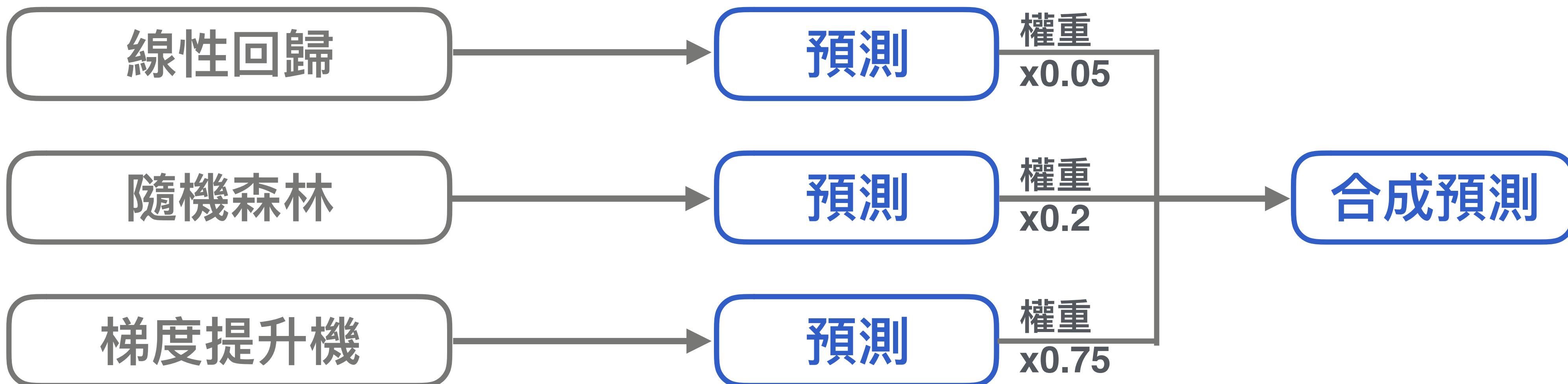
Voting / Blending / Stacking

- 使用同一資料 + 不同模型，合成出不同預測結果



混合泛化 (Blending) (1 / 3)

- 其實混合泛化非常單純，就是將不同模型的預測值加權合成，權重和為 1
如果取預測的平均 or 一人一票多數決(每個模型權重相同)，則又稱為 **投票泛化(Voting)**



- 雖然單純，但因為最容易使用且有效，至今仍然是競賽中常見的作法

混合泛化 (Blending) (2 / 3)

容易使用

- 不只在一般機器學習中有用，影像處理或自然語言處理等深度學習，也一樣可以使用
- 因為只要有預測值(Submit 檔案)就可以使用，許多跨國隊伍就是靠這個方式合作
- 另一方面也因為只要用預測值就能計算，在競賽中可以快速合成多種比例的答案，妥善消耗掉每一天剩餘的 Submit 次數

混合泛化 (Blending) (3 / 3)

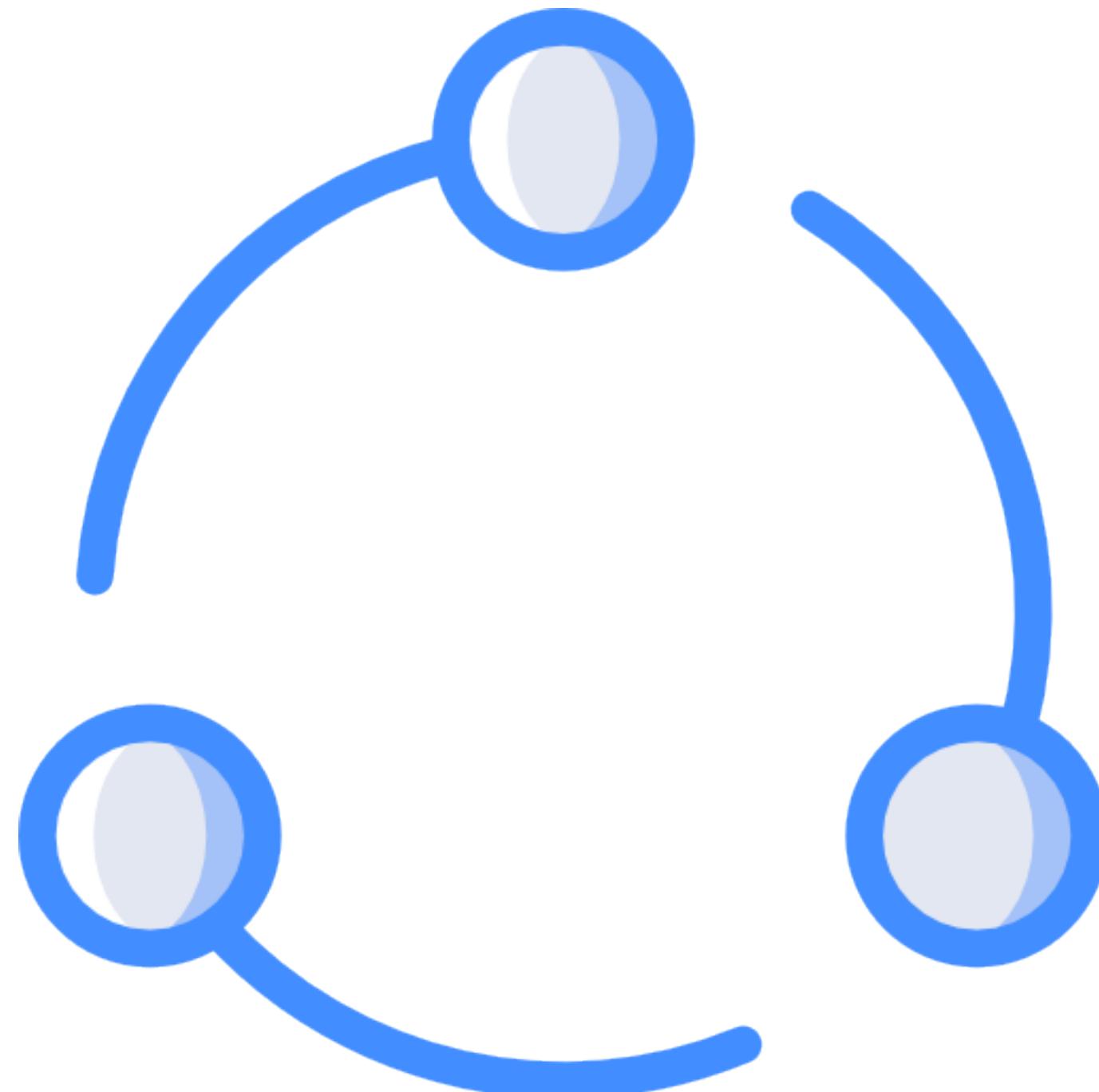
效果顯著

- Kaggle 競賽截止日前的 Kernel，有許多只是對其他人的輸出結果做 Blending，但是因為分數較高，因此也有許多人樂於推薦與發表
- 在2015年前的大賽中，Blending 仍是主流，例如林軒田老師也曾在課程中提及：有競賽的送出結果，是上百個模型的 Blending

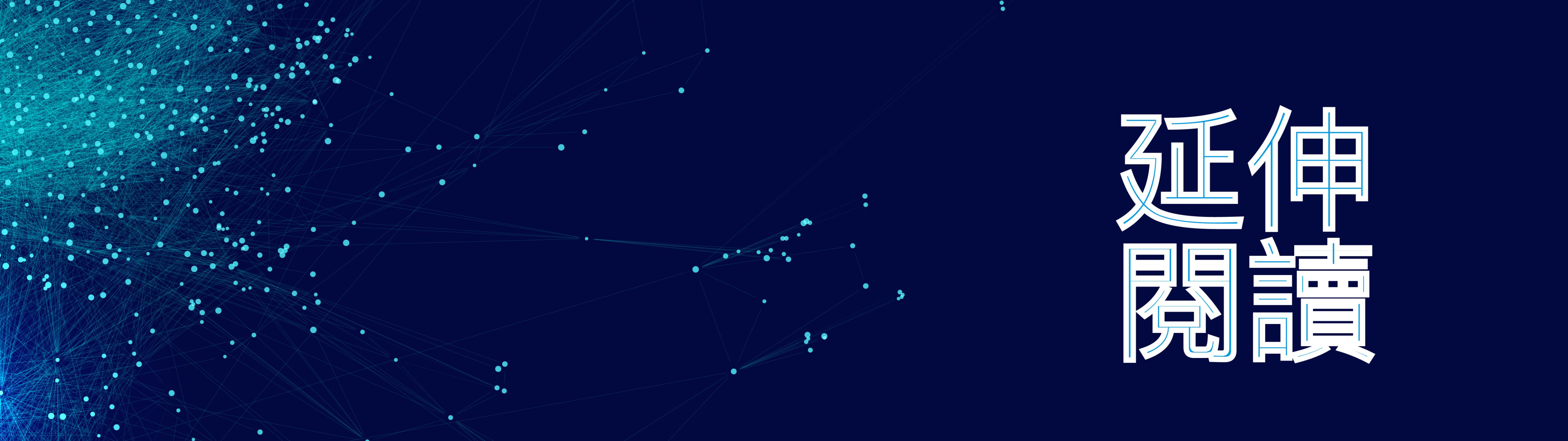
注意事項

- Blending 的前提是：個別**單模效果都很好**(有調參)並且**模型差異大**，單模要好尤其重要，如果單模效果差異太大，Blending 的效果提升就相當有限

重要知識點複習



- 資料工程中的集成，包含了資料面的集成 - **裝袋法(Bagging) / 提升法(Boosting)**，以及模型與特徵的集成 - **混合泛化(Blending) / 堆疊泛化(Stacking)**
- 混合泛化提升預測力的原因是基於**模型差異度大**，在預測細節上能互補，因此預測模型只要各自調參優化過且原理不同，通常都能使用混合泛化集成



延伸 閱讀

除了每日知識點的基礎之外，推薦的延伸閱讀能補足學員們對該知識點的了解程度，建議您解完每日題目後，若有
多餘時間，可再補充延伸閱讀文章內容。

推薦延伸閱讀

機器學習技法 Lecture 7: Blending and Bagging

林軒田老師公開課程 [影片連結](#)

- 當我們在網路上自己搜尋 Blending 時，往往搜尋到的都是林軒田老師的課程筆記，因此我們推薦同學如果對於 Blending 或 Bagging 的理論想要一探更完整內容的話，不妨來這邊尋找研讀的資料，相信絕對不會讓您失望 (如果太困難，也可以參考網路上眾多的閱讀筆記)

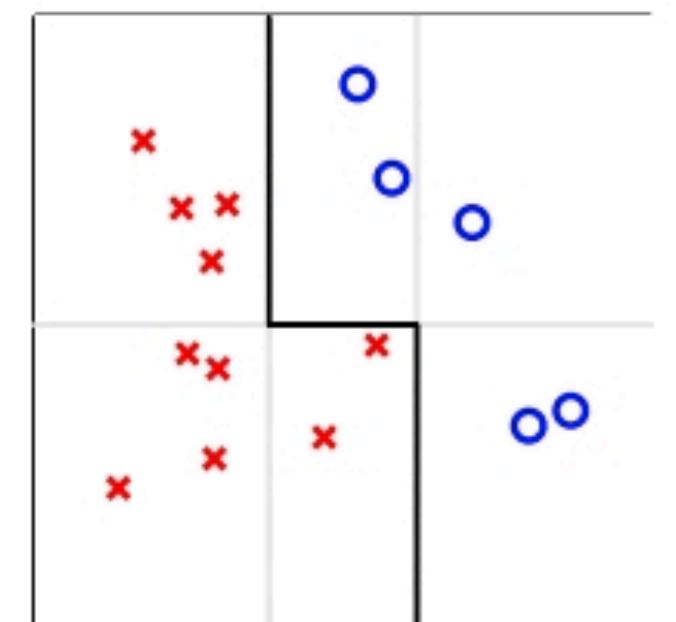
Uniform Blending (Voting) for Classification

uniform blending: known g_t , each with 1 ballot

$$G(\mathbf{x}) = \text{sign} \left(\sum_{t=1}^T 1 \cdot g_t(\mathbf{x}) \right)$$

- same g_t (autocracy): as good as one single g_t
- very different g_t (diversity + democracy): majority can correct minority
- similar results with uniform voting for multiclass

$$G(\mathbf{x}) = \underset{1 \leq k \leq K}{\operatorname{argmax}} \sum_{t=1}^T [g_t(\mathbf{x}) = k]$$



how about regression?

推薦延伸閱讀

Superblend

Kaggle 競賽網站-Kernel 範例 網頁連結

- 這邊就是我們所謂競賽中的 Blending Kernel，只是決定一個權重，將兩個其他的 Kernel 合併成答案檔，就是這場競賽中的最高分 Kernel，我們並不是要鼓勵大家也去這樣去賺分數，而是在告訴大家：Blending 的簡單，以及 Blending 的具有威力。

Code

This kernel has been released under the [Apache 2.0](#) open source license.

[Download Code](#)

```
1 # This Python 3 environment comes with many helpful analytics libraries installed
2 # It is defined by the kaggle/python docker image: https://github.com/kaggle/docker-python
3 # For example, here's several helpful packages to load in
4
5 import numpy as np # linear algebra
6 import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)
7
8 categories = ['toxic', 'severe_toxic', 'obscene', 'threat', 'insult', 'identity_hate']
9
10 blend_all = pd.read_csv("../input/fork-of-blend-it-al-ensemble/lazy_ensemble_submission_on_blend_sources.csv")
11 glove_and_fasttext = pd.read_csv("../input/glove-and-fasttext-blender/blend.csv")
12
13 blend = blend_all.copy()
14
15 blend[categories] = 0.67*blend_all[categories].values + 0.33*glove_and_fasttext[categories].values
16
17 blend.to_csv("superblend.csv", index=False)
```

圖片來源：Kaggle



解題時間

It's Your Turn

請跳出PDF至官網Sample Code & 作業
開始解題

