

Day 30

特徵工程

# 分類型特徵優化 - 葉編碼



# 分類預測的集成 ( 1 / 3 )

由於分類預測的集成，概念上與迴歸預測的集成有所不同，所以在最後做個補充分類的預測結果，意義上是對機率的預估，而不同特徵表示不同的判斷條件

想一想：假如要估計鐵達尼號上的生存機率

已知來自法國的旅客生存機率是 0.8，且年齡 40 到 50 區間的生存機率也是 0.8

那麼同時符合兩種條件的旅客，生存機率應該是多少呢？



# 分類預測的集成 ( 2 / 3 )

假如當作兩個預估模型，迴歸預測要集成兩種預測的做法有兩種：相加或平均

但是相加  $0.8 + 0.8 = 1.6$ ，機率會超過 1，不合理

平均  $(0.8 + 0.8) / 2 = 0.8$ ，法國機率已是 0.8，加上正向的事件居然還更低，也不合理  
應該要比 0.8 更高，但又不能到 1

那麼，該如何集成才合理呢





# 分類預測的集成 ( 3 / 3 )

解法：**邏輯斯迴歸(logistic regression)**與其重組

我們可以將邏輯斯迴歸理解成 「線性迴歸 + Sigmoid 函數」

而 sigmoid 函數理解成 「成功可能性與機率的互換」

這裡的成功可能性正表示更可能，負表示較不可能

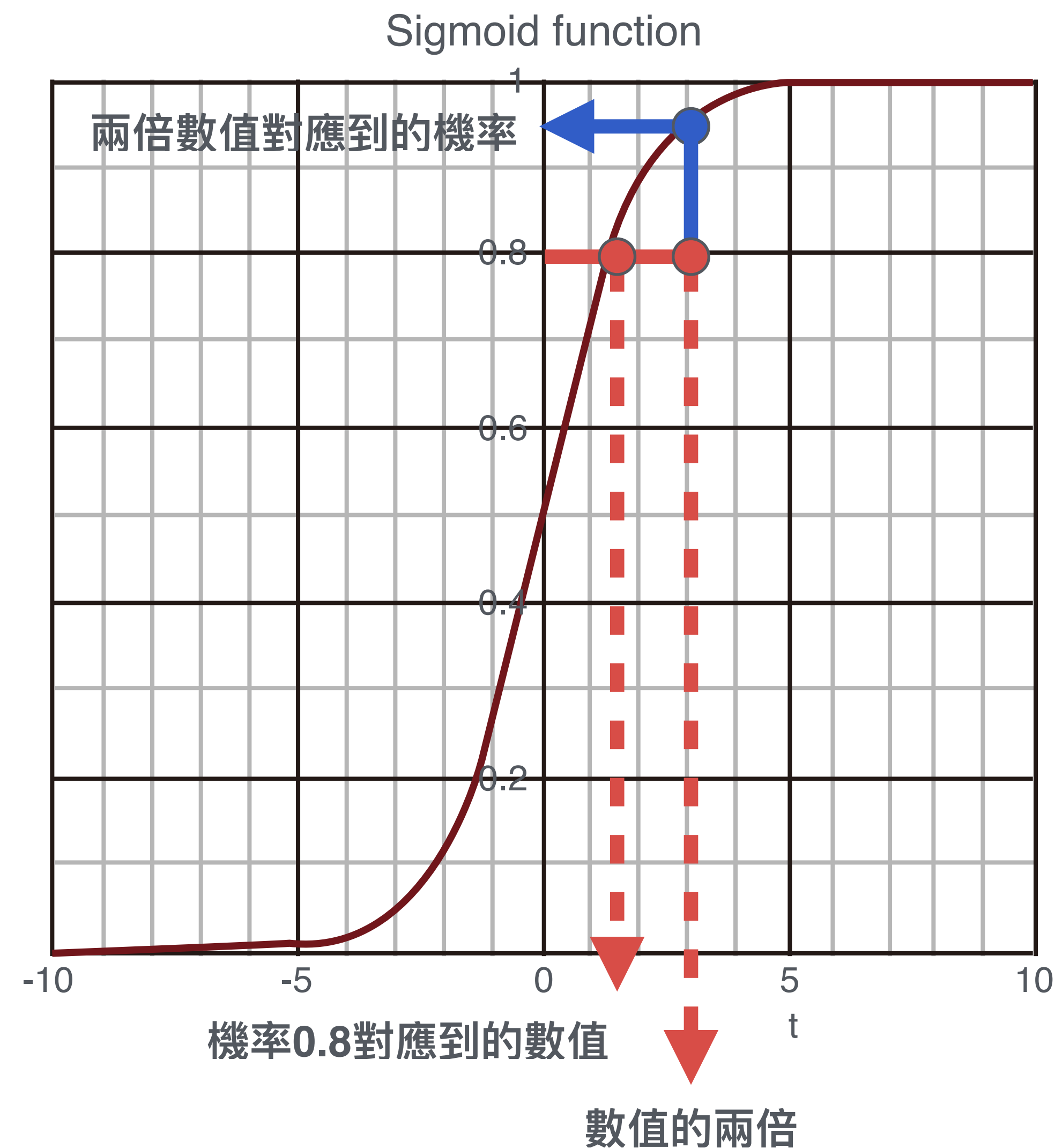
所以當我們使用 sigmoid 的反函數

就可以將機率重新轉為成功可能性

加完後再用 sigmoid 轉回機率

以此例而言，我們可以看到最後加成的結果是一個

介於 0.9 到 1 之間的機率



# 葉編碼 (leaf encoding) 原理 ( 1 / 2 )

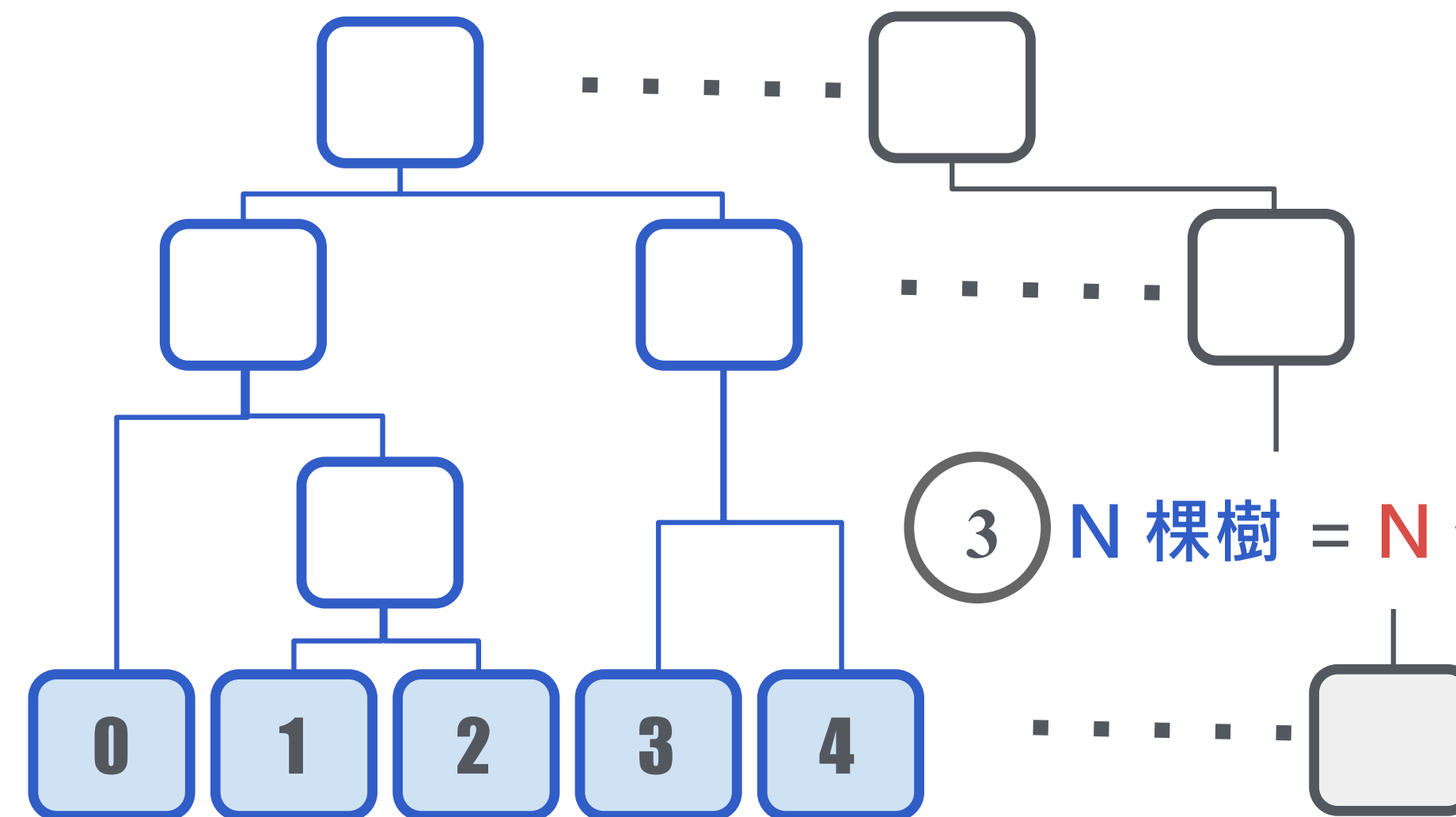
---

- 樹狀模型 (只要樹的數量大於 1) 做出預測時，模型預測時就免不了要用到相加 ( 梯度提升樹 ) 或平均 ( 隨機森林 ) 等集成方式
- 所以在分類預測時，理論上我們也必須將每棵樹對應的機率用 sigmoid 反函數反轉成成功可能性後相加 / 平均，但實際上可以使用葉編碼再加上邏輯斯迴歸，就可以達成上述結果

# 葉編碼 (leaf encoding) 原理 ( 2 / 2 )

- 葉編碼 (leaf encoding) 顧名思義，是採用**決策樹**的葉點作為編碼依據重新編碼
- **每棵樹**視為一個**新特徵**，每個新特徵均為**分類型特徵**，決策樹的葉點與該特徵標籤一一對應
- 最後再以邏輯斯迴歸合併預測

① 原本第 1 棵樹 = 第 1 個新特徵



② 樹有 5 個葉點 = 特徵有 5 種值

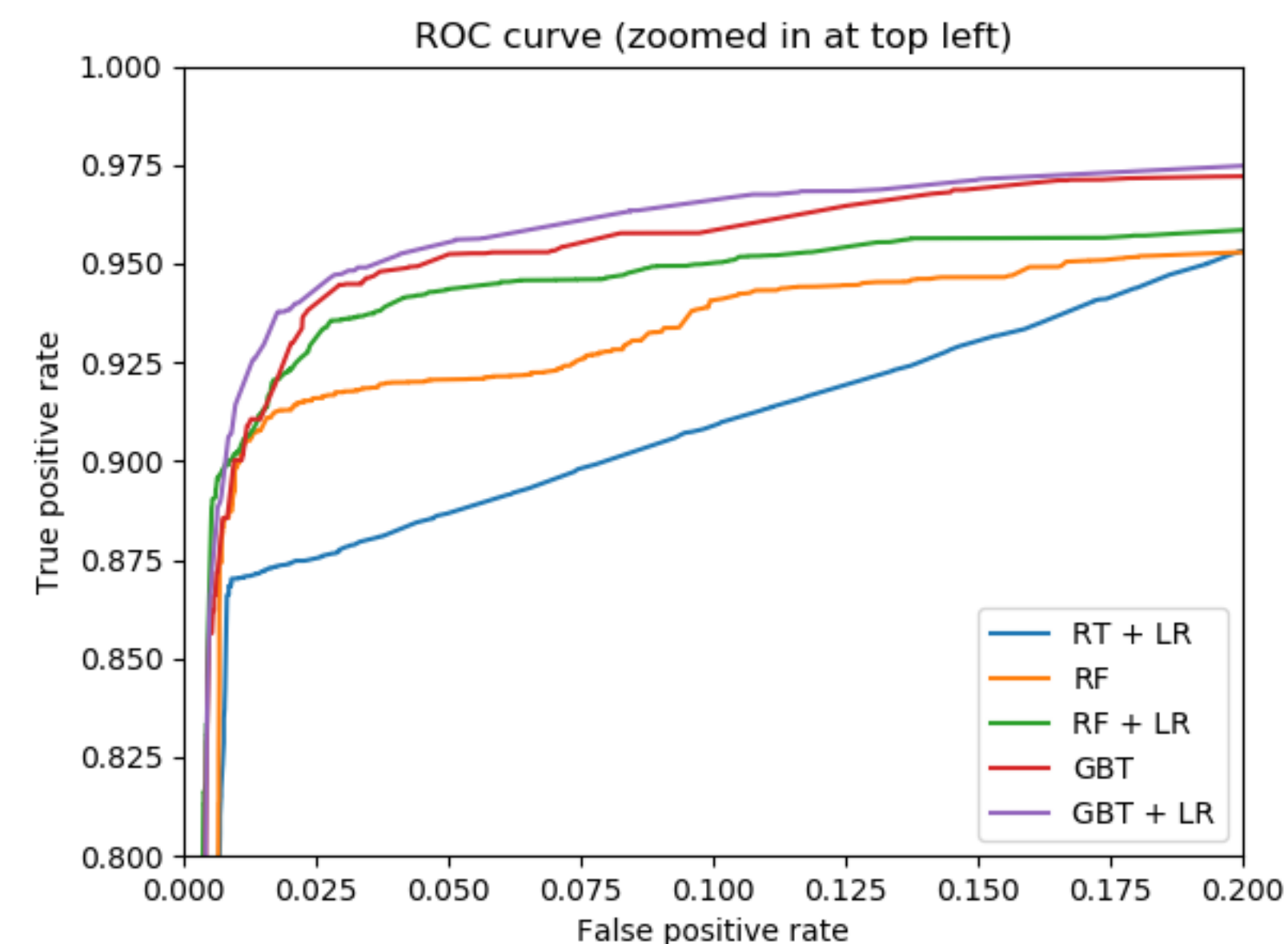
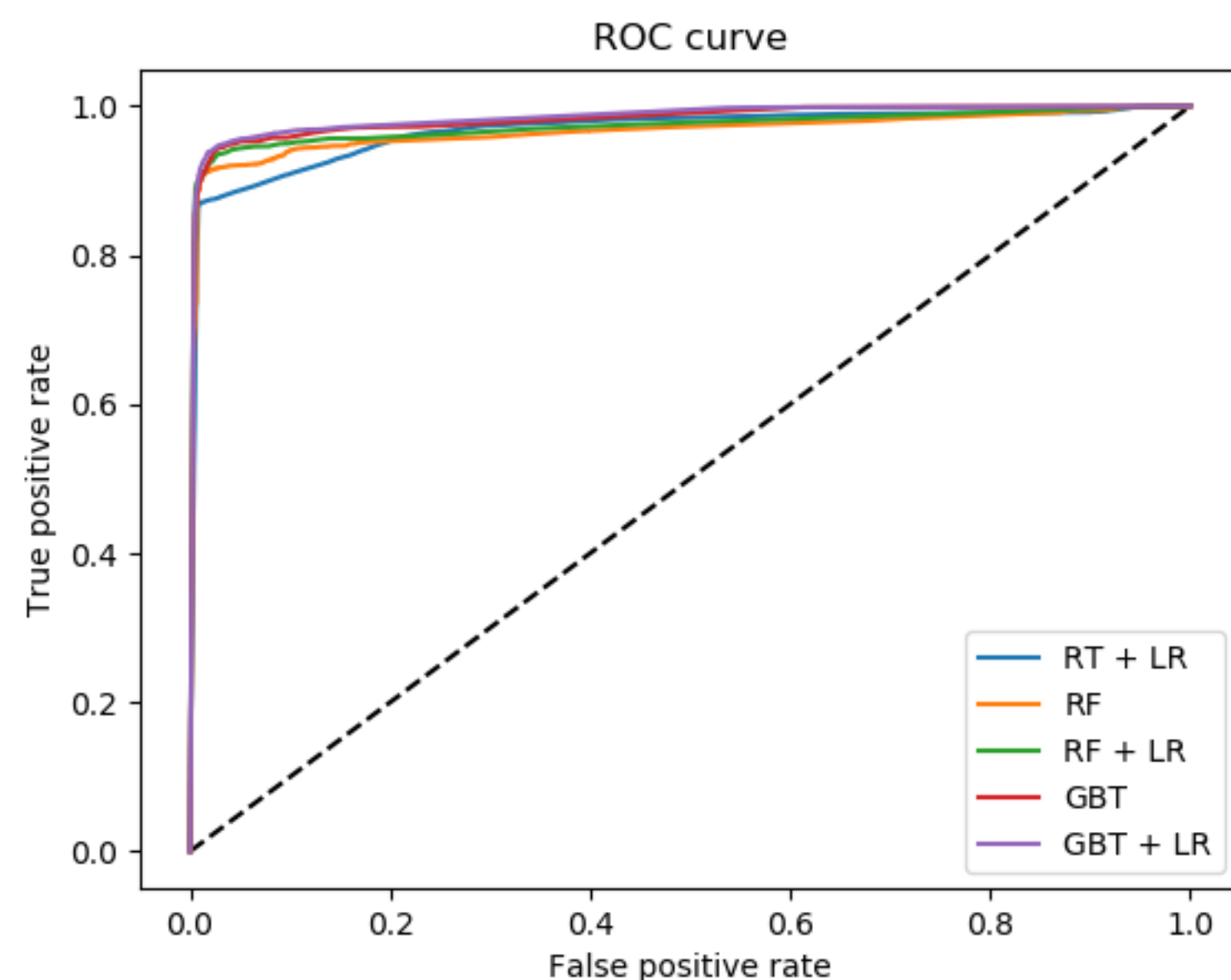


③ N 棵樹 = N 個新特徵

④ N 個新特徵做邏輯斯迴歸

# 葉編碼 (leaf encoding) + 邏輯斯迴歸

- 葉編碼需要先對樹狀模型擬合後才能生成，如果這步驟挑選了較佳的參數，後續處理效果也會較好，這點與特徵重要性類似
- 實際結果也證明，在分類預測中使用樹狀模型，再對這些擬合完的樹狀模型進行葉編碼+邏輯斯迴歸，通常會將預測效果再進一步提升



# 衍伸討論：有關樹狀模型與模型可解釋性

---

- 經由課程我們知道：樹狀模型有幾個重要的應用
  - **特徵重要性(feature importance)**：目前是特徵選擇的最主流作法
  - **葉編碼**：將特徵打散，完全依照樹狀模型的葉點重新編碼，再加上邏輯斯迴歸，可以再進一步提升分類預測能力
- 上述樹狀模型的獨特應用，都是基於人們對決策樹的理解與**可解釋性(explainable)**而有的設計
- 但目前深度學習的基礎：類神經網路，最缺乏的就是可以解釋性，若類神經網路能在可解釋性上更進一步，則可以想見也可以有更多的衍伸應用例如：capsule 模型)



# 解題時間 It's Your Turn

請跳出PDF至官網Sample Code & 作業  
開始解題

