Logit、XGBoost 及 Shap 概念及應用解析

編撰：屏東大學 周國華老師 (與 Google Gemini 共筆) 2025/11/30

這是一個非常經典且強大的組合！三個名詞，正好代表了**現代資料科學專案（特別是金融、醫學或社會科學領域）的三個重要階段**：

1. **Logit**：傳統統計學的基準（Baseline）與解釋性基礎。
2. **XGBoost**：追求最高預測準確度的機器學習模型（黑盒子）。
3. **SHAP**：用來解釋複雜模型（如 XGBoost）的工具，打開黑盒子。

以下我為你詳細拆解這三個方法的概念與應用場景，並說明它們通常如何協同工作。

---

1. Logit (Logistic Regression, 羅吉斯迴歸)

概念

雖然名字裡有「迴歸」，但它其實是用來做分類的。它是傳統統計學中最基礎的方法。

簡單來說，它透過一個 S 型函數（Sigmoid Function），將輸入的數據轉換成 0 到 1 之間的機率值。

$$P(Y=1|X) = \frac{1}{1+e^{-(\beta_0+\beta_1 x_1+...)}}$$

(註：請參閱文後附錄解說此公式)

- **白箱模型**：它的優點是結構非常透明。我們可以直接看到係數（Coefficient），比如「年齡每增加 1 歲，患病機率增加多少」。

應用方式

- **信用評分**：銀行傳統的信用評分卡（Scorecard）大多基於 Logit，因為法規要求必須能解釋為何拒絕客戶貸款。
- **醫學研究**：分析吸菸與肺癌的關聯（Odds Ratio）。
- **基準測試（Baseline）**：在跑複雜模型前，通常先跑一個 Logit，看看最基本的準確率是多少，用來當作比較的低標。

---

2. XGBoost (eXtreme Gradient Boosting, 極限梯度提升)

概念

這是目前機器學習競賽（如 Kaggle）和業界實務中的王者。它屬於「集成學習」（Ensemble Learning）的一種。

想像你要做一個決策，Logit 像是一個專家做決定；而 XGBoost 像是成千上萬個小決策樹（Decision Trees）組成的委員會。

1. **梯度提升（Gradient Boosting）**：它不是一次建好所有樹，而是第一棵樹預測錯的部分，第二棵樹會專注去修正，第三棵再修正前兩棵的錯誤，以此類推。
2. **極致優化**：它在運算速度和正規化（防止過擬合）上做了極致的優化，所以效能極高。

應用方式

- **精準行銷**：預測這個用戶會不會點擊廣告（CTR 預測）。
- **詐欺偵測**：從數百萬筆交易中瞬間找出異常刷卡行為。
- **風險預測**：違約風險預測（通常準確率會比 Logit 高很多）。

缺點：它是**黑盒子（Black Box）**。雖然預測很準，但你很難直觀地看出「為什麼」模型會判定這個人會違約。這就是為什麼我們需要下一個工具——SHAP。

---

3. SHAP (SHapley Additive exPlanations)

概念

SHAP 的核心概念來自**賽局理論（Game Theory）**中的 Shapley Value。

想像一個團隊拿了獎金（模型的預測結果），我們要怎麼公平地把功勞分給每個成員（特徵）？

- SHAP 會去計算：當某個特徵（例如「年齡」）加入模型時，對預測結果的**邊際貢獻**是多少。
- 它解決了 XGBoost 等複雜模型「看不懂」的問題，將黑盒子模型轉化為人類可理解的解釋。

應用方式

SHAP 提供了兩個層次的解釋：

1. **全域解釋 (Global Importance)**：告訴你整體而言，哪些特徵最重要？（例如：對於違約預測，最重要的前三名是「收入」、「負債比」、「過去遲繳次數」）。
2. **局部解釋 (Local Explanation)**：針對**單一案例**進行解釋。

- *情境*：為什麼「張三」的貸款被拒絕了？
- *SHAP 回答*：雖然張三收入高（加分），但因為他過去遲繳 3 次（大幅扣分），且負債比過高（扣分），導致分數低於門檻。

總結：它們在研究報告中是如何協作的？

你在研討會看到的報告，通常是按照以下這個**黃金流程**進行的：

1. **建立基準 (Logit)**：先用 Logit 跑一次數據，確認變數的顯著性，並建立一個準確率的基準線。
2. **提升效能 (XGBoost)**：使用 XGBoost 訓練模型，目的是為了獲得比 Logit **更高的預測準確率**（通常能提升 5%~15% 不等）。
3. **事後解釋 (SHAP)**：因為 XGBoost 太複雜，研究者最後使用 SHAP 來畫圖，解釋模型發現了什麼規律，證明模型不是亂猜，並挖掘出數據背後的洞察（Insight）。

快速對照表

方法	角色定位	優點	缺點
Logit	傳統基礎、白箱	解釋性高、簡單、計算快	處理非線性關係能力差，準確率通常較低
XGBoost	效能引擎、黑箱	**準確率極高**、可處理複雜非線性數據	模型複雜，人類無法直接理解其邏輯
SHAP	解釋器、翻譯官	**能解釋任何模型**（包含 XGBoost），視覺化效果好	計算量大，運算時間較久

附錄：

羅吉斯迴歸公式解讀

$$P(Y=1|X)=\frac{1}{1+e^{-(\beta_0+\beta_1x_1+\beta_2x_2+...)}}$$

這個公式是我們前面在解說 **Logit (Logistic Regression，羅吉斯迴歸)** 時所提到的核心數學表達式。它雖然看起來複雜，但概念非常簡單：**將任何線性的輸入結果，轉換成介於 0 到 1 之間的機率值**。以下我為您詳細解說公式中的各個符號的意義：

1. 公式目標：機率 (Probability)

符號	中文意義	實際應用情境
P	**條件機率**	
Y	**目標變數 (Target Variable)**	只能是二元值，通常為 1 (成功/發生/舞弊) 或 0 (失敗/未發生/正常)。
X	**特徵變數集合 (Features)**	輸入模型的所有自變數集合 ($x_1, x_2, \ldots$)。

2. 公式核心：Sigmoid 函數（S 形函數）

右側的 $\frac{1}{1+e^{-z}}$ 整個結構被稱為 **Sigmoid Function**。它的作用是將一個範圍從 $-\infty$ 到 $+\infty$ 的數字（即括號內的部分），壓縮到 **0 到 1 之間**。

3. 公式中的參數與特徵 (Features and Coefficients)

括號內的 $-(\beta_0+\beta_1 x_1+\beta_2 x_2+\ldots)$ 是一個線性組合，通常被稱為 **Log-Odds (對數勝算)**，它是我們模型的核心預測能力。

符號	中文意義	說明
e	**自然對數的底數**	一個數學常數，約等於 2.718。
β_0	**截距 (Intercept)**	當所有特徵變數 (x_i) 都為 0 時的基礎對數勝算。
$\beta_1, \beta_2, \ldots$	**係數 (Coefficients)**	模型的**核心權重**。它代表了對應的特徵變數 (x_1, x_2) 對於 $Y=1$ 發生機率的影響程度和方向。
$x_1, x_2, \ldots$	**單一特徵值**	您的輸入數據中的具體數值（例如 x_1 = 報銷金額為 5,000 元，x_2 = 是否週末為 1）。

總結與應用

在您的 **審計舞弊偵測** 專案中，這個公式就是用來計算：

「根據這筆報銷單的金額、部門、日期等特徵，它是舞弊的機率有多高？」

- 如果算出來 $P(Y=1 \mid X)=0.01$，代表機率只有 1%，模型認為它是正常的。

- 如果算出來 $P(Y=1 \mid X)=0.95$，代表機率有 95%，模型認為它是高度舞弊。

Logit 模型之所以在會計、審計等領域常被用作基準，正是因為它的核心 β 係數是**可解釋的**，能讓稽核人員知道：是「金額太高」還是「部門和類別組合異常」導致風險增加。