**特徵工程 Feature Engineering**

當產品經理表示：“我們需要預測特定客戶是否會繼續使用我們的服務。這裡有客戶過去五年與我們產品互動的日誌”，您不能僅僅將這些數據載入軟件庫中並期望得到一個預測。首先，您需要構建一個數據集。

回想第一章中提到的，一個數據集包含一系列標籤示例 (𝑥𝑖,𝑦𝑖)𝑖=1𝑁(*xi*​,*yi*​)*i*=1*N*​。這個𝑁*N*個示例中的每個元素𝑥𝑖*xi*​被稱為特徵向量。特徵向量是一個多維向量，其中每個維度𝑗=1,…,𝐷*j*=1,…,*D*包含以某種方式描述示例的值。這個值被稱為特徵，表示為 𝑥(𝑗)*x*(*j*)。

將原始數據轉化為可用數據集的過程稱為特徵工程。這通常是一項勞動密集型任務，需要數據分析師具有豐富的創造力和理想情況下的特定領域知識。

例如，要轉換計算機系統用戶互動的日誌，可以開發包括用戶特定信息和從日誌中提取的各種統計數據的特徵。對於每個用戶，一個特徵可以捕捉訂閱價格；其他特徵可能衡量每天、每週和每年的連接頻率。其他特徵可能包括會話平均時長（秒）或每個請求的平均響應時間等。任何可量化的元素都可以作為特徵。數據分析師的角色是設計有信息量的特徵——這些特徵使學習算法能夠構建有效預測訓練數據標籤的模型。具有高預測力的高度信息特徵也稱為具有高預測力。例如，用戶會話的平均持續時間可能是預測用戶未來是否繼續使用應用程序的強有力預測因素。

如果模型能準確預測訓練數據，我們說該模型偏差低，即在預測用來構建模型的示例的標籤時，模型犯錯很少。

**獨熱編碼 One-Hot Encoding**

某些學習演算法只能處理數值型特徵向量。當數據集中包含如“顏色”或“星期幾”等類別特徵時，可以將這些類別特徵轉換為多個二進位向量。

考慮一個例子，其中類別特徵“顏色”可以取三個可能的值：“紅色”，“黃色”或“綠色”。您可以將此特徵編碼為包含三個二進位值的數值向量：

* 紅色 = [1, 0, 0]
* 黃色 = [0, 1, 0]
* 綠色 = [0, 0, 1]

此方法會增加您的特徵向量的維度。重要的是不要將類別數字化，如將紅色編為1、黃色為2、綠色為3，以避免暗示類別之間存在固有順序。這種順序可能會誤導學習演算法感知不存在的規律，從而可能導致過度擬合。只有當數據的序位性質與學習過程相關時，才應使用數值編碼，以確保機器學習模型正確解讀數據，無需不必要的複雜性。

**分箱 Binning**

在某些情況下，這在實踐中相對罕見，您可能會遇到一個數值特徵，但您希望將其轉換為類別特徵。分箱（也稱為分桶）涉及將連續特徵轉換為稱為箱或桶的多個二進制特徵，通常基於預定義的值範圍。例如，不是將年齡表示為單一實數值特徵，分析師可能會將年齡範圍劃分為离散的箱：0至5歲的年齡可以分配到一個箱中，6至10歲到第二個箱中，以及11至15歲到第三個箱中，等等。

例如，假設特徵 𝑗=18*j*=18 表示年齡。通過應用分箱，我們用相應的類別箱替換這個連續特徵。假設引入三個新的箱，標籤為“age\_bin1”、“age\_bin2”和“age\_bin3”，分別索引 𝑗=123*j*=123、𝑗=124*j*=124 和 𝑗=125*j*=125。如果對於某個示例 𝑥𝑖*xi*​，𝑥𝑖(18)=7*xi*(18)​=7，那麼我們設置 𝑥𝑖(124)=1*xi*(124)​=1；如果 𝑥𝑖(18)=13*xi*(18)​=13，我們設置 𝑥𝑖(125)=1*xi*(125)​=1，依此類推。

精心設計的分箱可以通過減少對大量示例的需求來提高算法的學習效率。這種效率提升是因為分箱為學習算法提供了一個“提示”：在特定範圍內，特徵的精確值無關緊要，從而簡化了學習任務。當某個類別變量的值的順序重要時，例如表示有序類別的變量（例如 {poor, decent, good, excellent}），我們可以通過用數字替換這些類別來保留其序位性質（例如，{1, 2, 3, 4}）。

Normalization

**正規化**

正規化是將數值特徵可以取的實際值範圍轉換為標準值範圍的過程，通常是 [−1, 1] 或 [0, 1]。例如，如果某個特徵的自然範圍是 350 到 1450，您可以通過首先從每個值中減去 350，然後將結果除以 1100，將這些值正規化到 [0, 1] 範圍內。

正規化的一般公式為：

𝑥ˉ(𝑗)=𝑥(𝑗)−min(𝑗)max(𝑗)−min(𝑗),*x*ˉ(*j*)=max(*j*)−min(*j*)*x*(*j*)−min(*j*)​,

其中 min(𝑗)min(*j*) 和 max(𝑗)max(*j*) 分別代表數據集中特徵 𝑗*j* 的最小值和最大值。

為什麼要正規化？雖然正規化不是強制性的，但它通常可以增加學習速度。考慮上一章討論的梯度下降方法。假設您有一個二維特徵向量。在更新參數 𝑤1*w*1​ 和 𝑤2*w*2​ 時，使用平均平方誤差相對於 𝑤1*w*1​ 和 𝑤2*w*2​ 的偏導數。如果 𝑥1*x*1​ 的範圍是 [0, 1000] 而 𝑥2*x*2​ 的範圍是 [0, 0.0001]，則相對於較大特徵的導數將不成比例地影響更新過程。

正規化還有助於確保輸入在相似的相對較小範圍內。這一預防措施有助於避免與非常小或非常大的數字相關的計算問題，如數值溢出，從而確保更穩定、更有效的計算。

**Standardization**

**標準化**

**標準化（或 z 分數正規化）是一種過程，其中特徵值被重新調整，使其具有標準正態分佈的特性，平均值（μ）為 0，標準差（σ）為 1。平均值是數據集中所有示例的特徵平均值，σ 是從這個平均值的偏差。**

**特徵的標準分數或 z 分數使用以下公式計算：**

**𝑥^(𝑗)=𝑥(𝑗)−𝜇(𝑗)𝜎(𝑗).*x*^(*j*)=*σ*(*j*)*x*(*j*)−*μ*(*j*)​.**

**在決定使用正規化還是標準化時，沒有一個適用於所有情況的答案。通常，如果數據集不太大且有時間，建議嘗試兩種方法以確定哪一種為您的特定任務提供更好的性能。**

**如果時間限制無法進行多次實驗，請考慮以下指南：**

* **非監督學習：實際上，標準化通常比正規化更有利於非監督學習算法。**
* **正態分佈：對於分佈接近正態分佈（通常稱為鐘形曲線）的特徵，標準化更為適宜。**
* **處理異常值：對於偶爾可能表現出極高或極低值（異常值）的特徵，標準化有優勢，因為正規化可能會將典型值壓縮到非常窄的範圍內。**
* **一般情況：在大多數其他情況下，正規化更可取。**

**現代學習算法，可在流行的庫中找到，通常對於範圍不同的特徵具有很強的魯棒性。儘管特徵重新調整對大多數學習算法都有益，但在許多情況下，使用原始特徵訓練的模型仍然表現良好。**

**處理缺失特徵**

**在某些情況下，數據以已定義特徵的數據集形式提供給分析師。在一些例子中，某些特徵的值可能會缺失。這種情況通常發生在數據集是手工製作的時候，製作它的人忘記填寫某些值或根本就沒有測量到。**

**處理特徵缺失值的典型方法包括：**

* **移除缺失特徵的示例：如果您的數據集足夠大，可以犧牲一些訓練樣本，則可以採用此方法。**
* **使用能夠處理缺失特徵值的學習算法：這可能涉及忽略這些值或基於數據集中的其他數據點推斷它們。**
* **填補缺失值：這意味著基於其他可用數據估計這些值。常見的填補方法包括使用數據集中特徵的平均值、中位數或眾數，或更複雜的技術，如回歸、k最近鄰或多重填補。**

**選擇與數據性質和機器學習任務的具體要求相符的方法至關重要。每種方法都有其優點和缺點：**

* **移除示例：這種方法簡單直接，但可能導致數據大量減少，特別是如果缺失與結果系統相關，可能會丟失寶貴信息。**
* **處理缺失數據的學習算法：某些模型，如某些決策樹，可以內在地處理缺失數據。這種方法保留了數據集的大小，但可能使模型訓練和解釋變得更加複雜。**
* **填補：這種方法可以幫助維持數據集的完整性，不會失去數據。然而，它引入了關於替代缺失值的假設，如果選擇不當，可能會使模型產生偏差。**