機器學習面試的12個基礎問題，強烈推薦！

Datawhale 今天

選自Medium

作者：JP Tech等

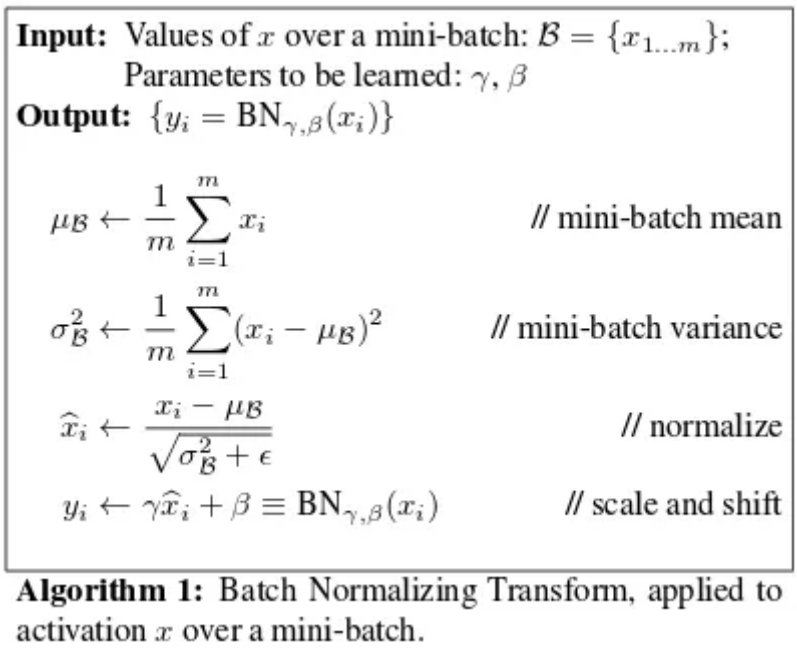
機器之心編譯

> 畢業季找工作了？如果想應聘機器學習工程師崗位，你可能會遇到技術面試，這是面試官掂量你對技術的真正理解的時候，所以還是相當重要的。近日，JP Tech 發表了一篇文章，介紹了他們面試新人時可能會提出的12 個面試問題。問題很基礎，但卻值得一看。

這些問題是我在面試AI 工程師崗位時常問到的問題。事實上，並非所有面試都需要用到所有這些問題，因為這取決於面試者的經驗以及之前做過的項目。經過很多面試（尤其是與學生的面試）之後，我收集了12 個深度學習領域的面試問題。我將在本文中將其分享給你。

問題1：闡述批歸一化的意義

這是一個非常好的問題，因為這涵蓋了面試者在操作神經網絡模型時所需知道的大部分知識。你的回答方式可以不同，但都需要說明以下主要思想：



Algorithm 1: Batch Normalizing Transform, applied to activation x over a mini-batch.

算法1：批歸一化變換，在一個mini-batch上應用於激活x。

批歸一化是一種用於訓練神經網絡模型的有效方法。這種方法的目標是對特徵進行歸一化處理（使每層網絡的輸出都經過激活），得到標準差為1 的零均值狀態。所以其相反的現像是非零均值。這將如何影響模型的訓練：

首先，這可以被理解成非零均值是數據不圍繞0 值分佈的現象，而是數據的大多數值大於0 或小於0。結合高方差問題，數據會變得非常大或非常小。在訓練層數很多的神經網絡時，這個問題很常見。如果特徵不是分佈在穩定的區間（從小到大的值）裡，那麼就會對網絡的優化過程產生影響。我們都知道，優化神經網絡將需要用到導數計算。

假設一個簡單的層計算公式y = (Wx + b)，y 在W 上的導數就是這樣：dy=dWx。因此，x 的值會直接影響導數的值（當然，神經網絡模型的梯度概念不會如此之簡單，但理論上，x 會影響導數）。因此，如果x 引入了不穩定的變化，則這個導數要么過大，要么就過小，最終導致學習到的模型不穩定。而這也意味著當使用批歸一化時，我們可以在訓練中使用更高的學習率。

批歸一化可幫助我們避免x 的值在經過非線性激活函數之後陷入飽和的現象。也就是說，批歸一化能夠確保激活都不會過高或過低。這有助於權重學習——如果不使用這一方案，某些權重可能永遠不會學習。這還能幫助我們降低對參數的初始值的依賴。

批歸一化也可用作正則化（regularization）的一種形式，有助於實現過擬合的最小化。使用批歸一化時，我們無需再使用過多的dropout；這是很有助益的，因為我們無需擔心再執行dropout 時丟

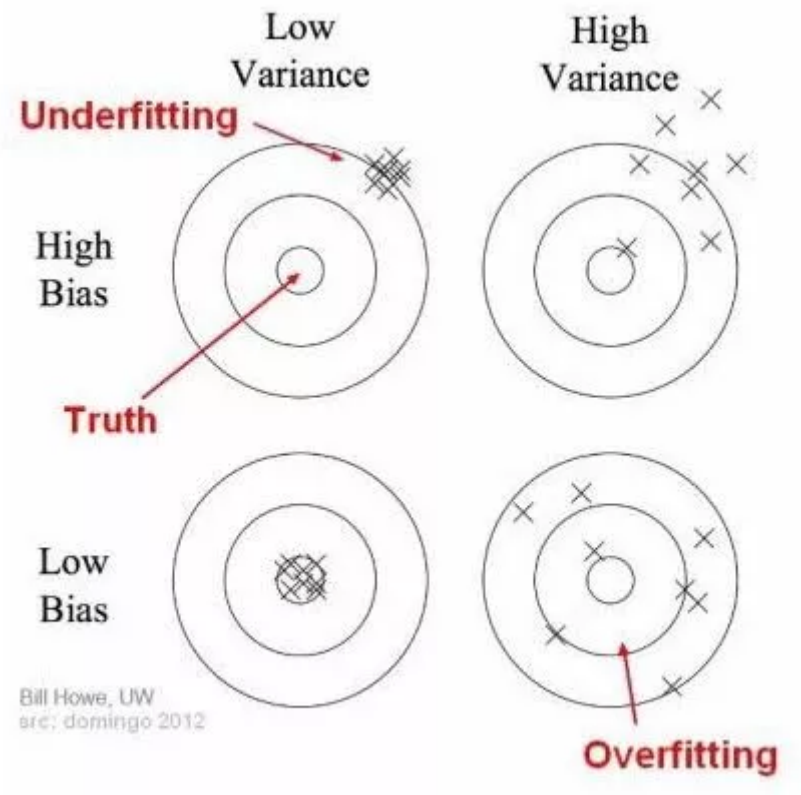
失太多信息。但是，仍然建議組合使用這兩種技術。

問題2：闡述偏置和方差的概念以及它們之間的權衡關係

偏置（bias）是什麼？這很好理解，偏置是當前模型的平均預測結果與我們需要預測的實際結果之間的差異。當模型的偏置較高時，說明其不夠關注訓練數據。這會使得模型過於簡單，無法在訓練和測試上同時實現優良的準確度。這個現像也被稱為「欠擬合」。

方差（variance）可以簡單理解為是模型輸出在一個數據點上的分佈（或聚類）。方差越大，模型越有可能更密切關注訓練數據，而無法提供在從未見過的數據上的泛化能力。由此造成的結果是，模型可在訓練數據集上取得非常好的結果，但在測試數據集上的表現卻非常差。這個現像被稱為過擬合。

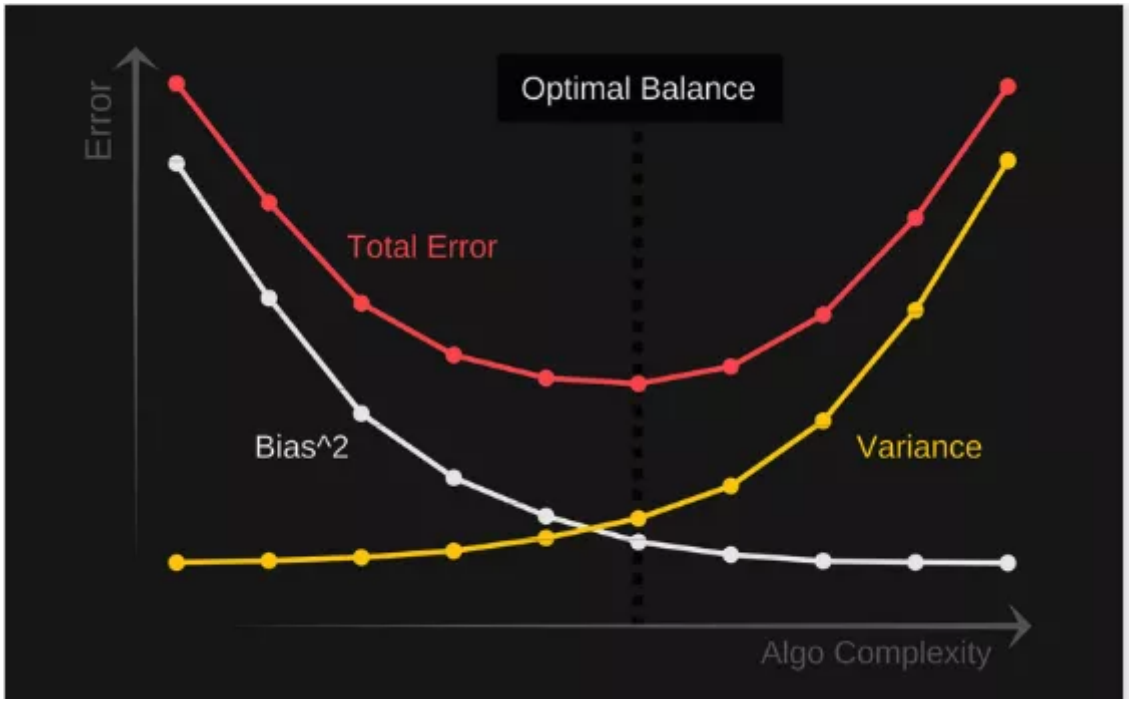
這兩個概念的關係可通過下圖說明：



上圖中，圓圈中心是能夠完美預測精準值的模型。事實上，你永遠無法找到這樣好的模型。隨著我們離圓圈中心越來越遠，模型的預測也越來越差。

我們可以改變模型，使得我們可以增大模型猜測的數量，使其盡可能多地落在圓圈中心。偏置和方差之間需要保持平衡。如果我們的模型過於簡單，有非常少的參數，那麼它就可能有較高的偏置和較低的方差。

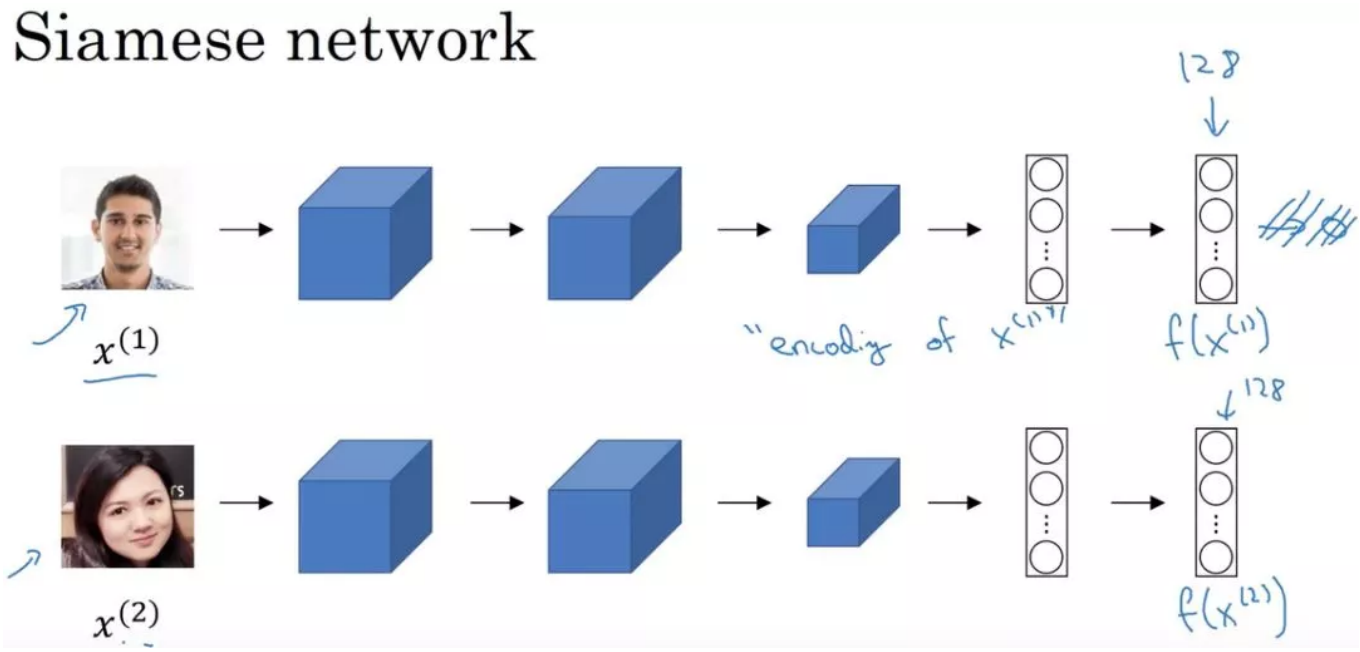
另一方面，如果我們的模型有大量參數，則其將有較高的方差和較低的偏置。這是我們在設計算法時計算模型複雜度的基礎。



問題3：假設深度學習模型已經找到了1000萬個人臉向量，如何通過查詢以最快速度找到一張新人臉?

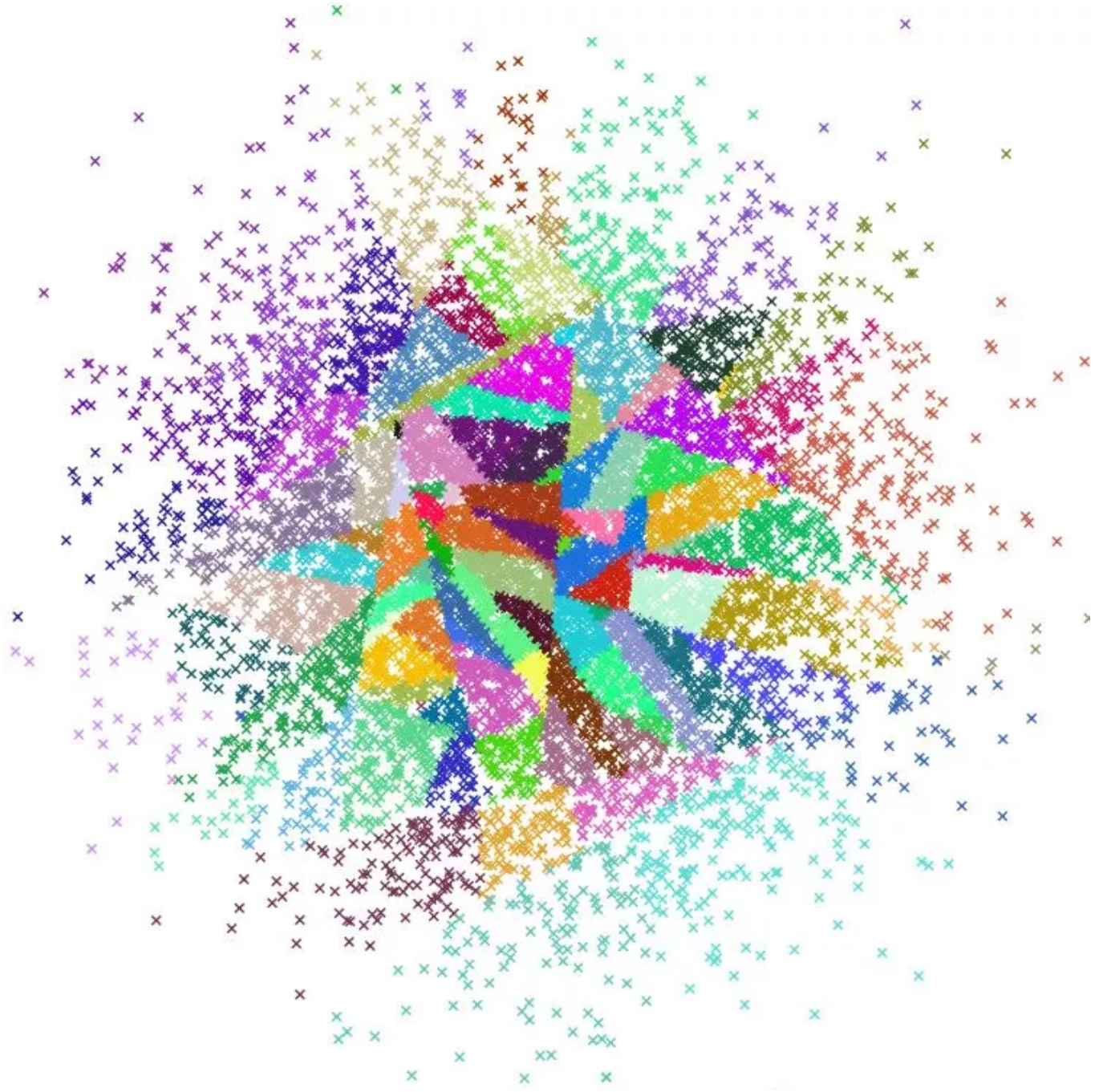
這個問題涉及到深度學習算法的實際應用，關鍵點在於**索引數據**的方法。這是將One Shot Learning應用於人臉識別的最後一步，但這也是最重要的步驟，讓該應用易於實際部署。

基本上來說，對於這個問題，你首先應該通過One Shot Learning 給出人臉識別方法的整體概況。這可以簡單地理解成將每張臉轉換成一個向量，然後識別新的人臉是尋找最接近（最相似）於輸入人臉的向量。通常來說，人們會使用有三元組損失（triplet loss）的定制損失函數的深度學習模型來完成這一任務。



但是，如果有文章開頭那樣的圖像數量增長，那麼在每次識別中都計算與1000 萬個向量的距離可不是個聰明的解決方案，這會使得系統的速度非常慢。我們需要思考在真實向量空間上索引數據的方法，以便讓查詢更加便捷。

這些方法的主要思想是將數據劃分成簡單的結構，以便查詢新數據（可能類似於樹結構）。當有新數據時，在樹中查詢有助於快速找到距離最近的向量。



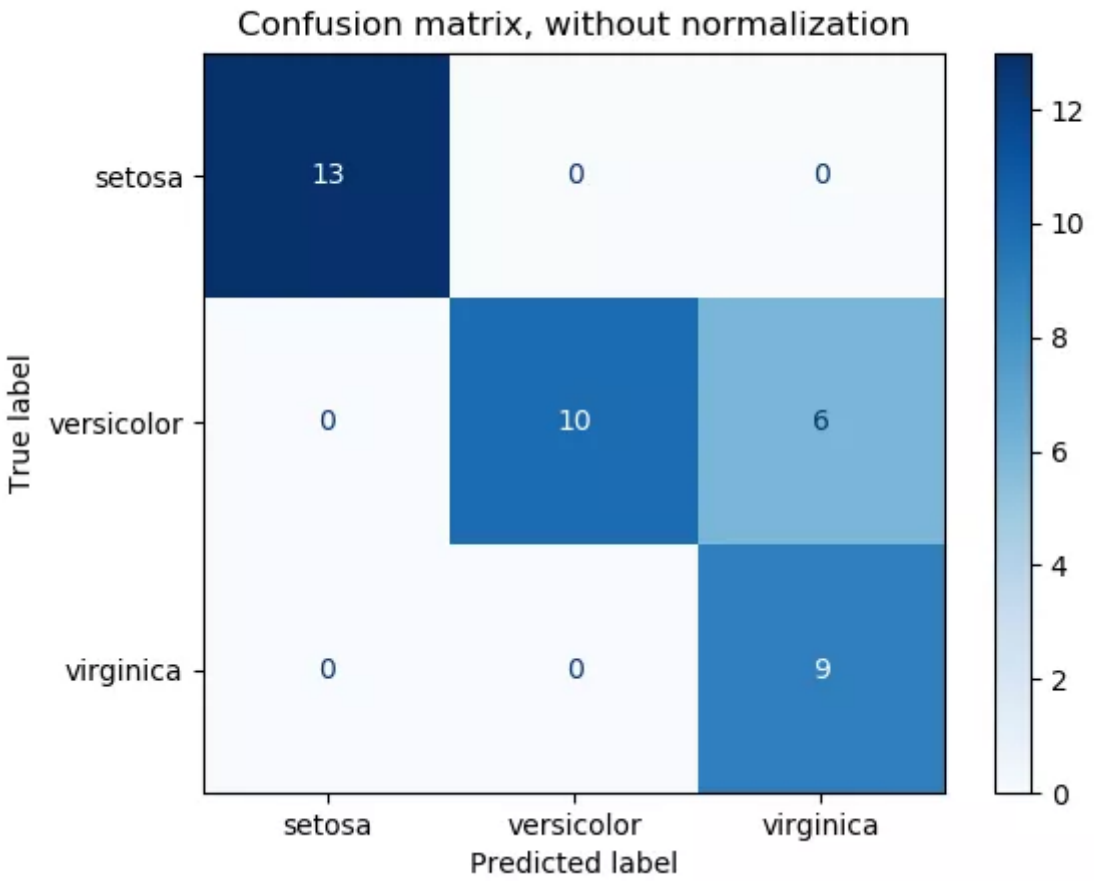
有一些可以用於這一目的的方法，比如局部敏感哈希（LSH）、Approximate Nearest Neighbors Oh Yeah——Annoy Indexing、Faiss等。

問題4：對於分類問題，準確度指數完全可靠嗎？你通常使用哪些指標來評估你的模型？

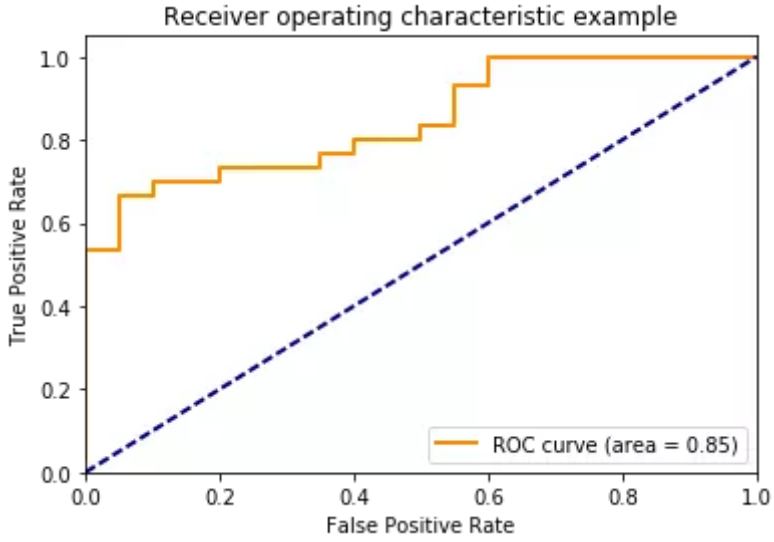
針對分類問題的評估方法有很多。準確度是一種很簡單的指標，也就是用正確的預測數據除以總的數據。這聽起來很合理，但現實情況是，這種度量方式對不平衡的數據問題來說並不夠顯著。假設我們正在構建用於預測網絡攻擊的預測模型（假設攻擊請求大約佔請求總數的1/100000）。

如果該模型預測所有請求都是正常的，那麼其準確率也高達99.9999%，但在這個分類模型中，這個數字通常是不可靠的。上面的準確度計算得到的結果通常是被正確預測的數據的百分比，但沒有詳細

說明每個類別的分類細節。相反，我們可以使用混淆矩陣。基本上來說，混淆矩陣展示了數據點實際屬於的類別，以及模型預測的類別。其形式如下：

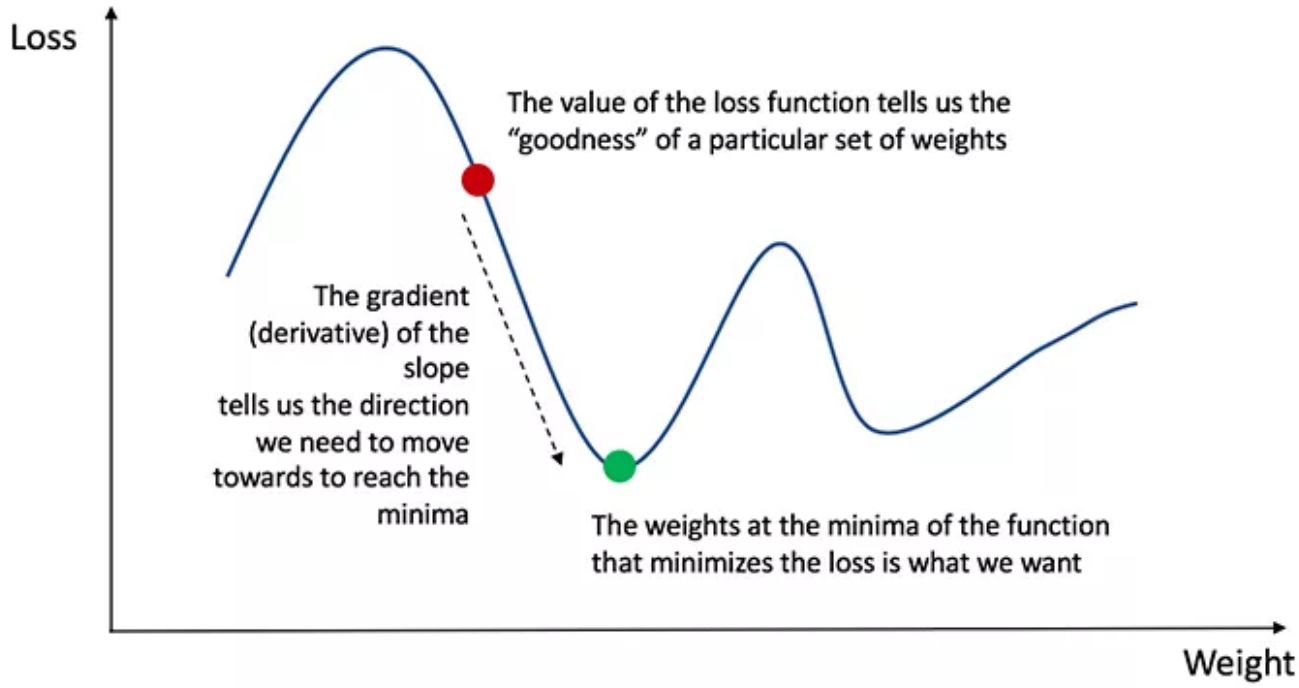


除了表達真正例和假正例指標對應於定義了該分類的每個閾值的變化之外，我們還有名為受試者工作特徵（ROC）的圖表。基於ROC，我們可以知道該模型是否有效。



理想的ROC 越接近左上角的橙色線（即真正例較高，假正例較低），結果就越好。

問題5：你怎麼理解反向傳播？請解釋動作（action）的機制。



這個問題的目標是測試參加面試的人是否理解神經網絡的工作方式。你需要說明以下幾點：

前向過程（前向計算）是幫助模型計算每層的權重的過程，所得到的計算會得到一個結果yp。這時候會計算損失函數的值；損失函數的這個值能體現模型的優劣程度。如果這個損失函數不夠好，我們就需要找到一種能夠降低這個損失函數的值的方法。神經網絡的訓練目標實際上就是最小化某個損失函數。損失函數L(yp,yt) 表示yp 模型的輸出值與yt 數據標籤的實際值之間的差異程度。

為了降低損失函數的值，我們需要使用導數。反向傳播能幫助我們計算網絡每一層的導數。基於每一層上導數的值，優化器（Adam、SGD、AdaDelta 等）可通過梯度下降來更新網絡的權重。

反向傳播會使用鍊式法則機製或導數函數，從最後一層到第一層計算每一層的梯度值。

問題6：激活函數有什麼含義？激活函數的飽和點是什麼？

1. 激活函數的含義

激活函數的目的是突破神經網絡的線性性質。我們可以將這些函數簡單理解成是一種過濾器，作用是決定信息是否可以通過神經元。在神經網絡訓練期間，激活函數在調整導數斜率方面具有非常重要的

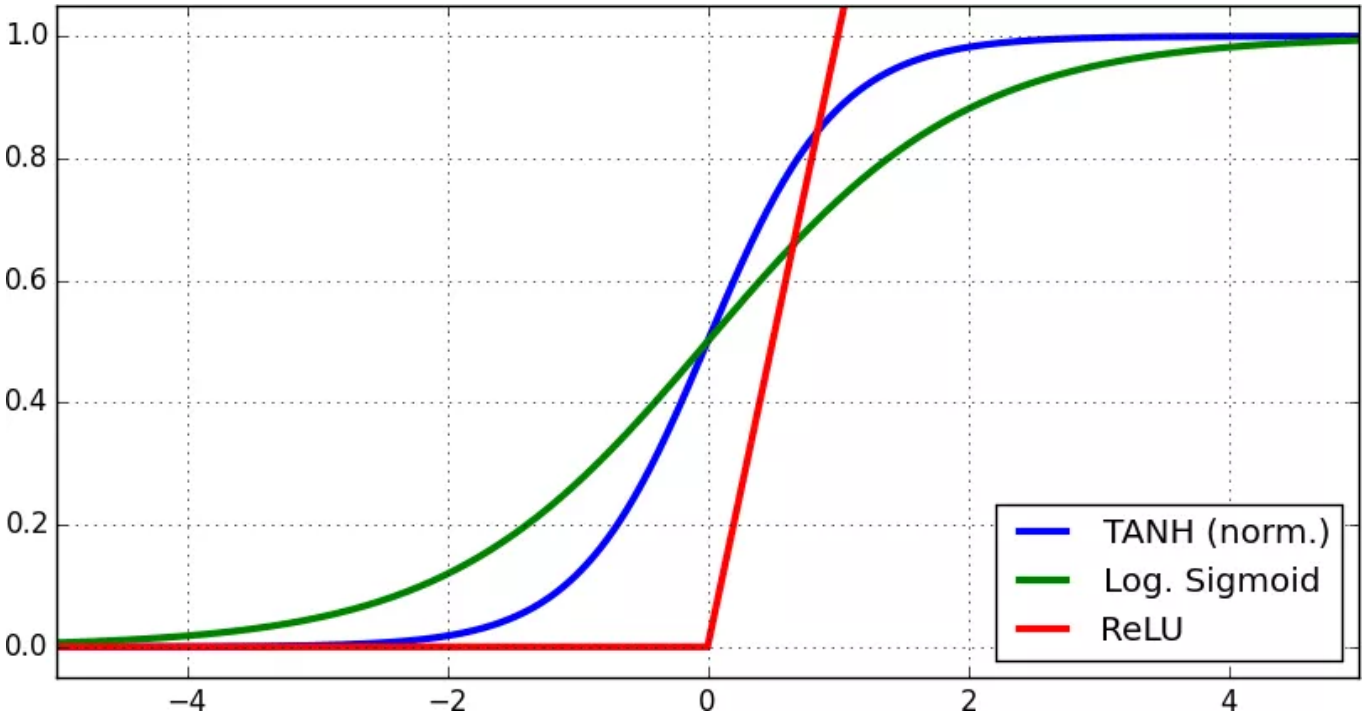
作用。

相比於使用線性函數，使用非線性激活函數能讓神經網絡學習更複雜的函數表徵；但為了有效地使用它們，我們需要理解這些非線性函數的性質。大多數激活函數都是連續可微的函數。

這些函數是連續函數，也就是說如果輸入有較小的可微分的變化（在其定義域中的每個點上都有導數），那麼輸出也會有較小的變化。當然，如前面提到的那樣，導數的計算是非常重要的，而且決定了我們的神經元是否可以訓練。值得提及的幾種激活函數有Sigmoid、Softmax 和ReLU。

2. 激活函數的飽和範圍

Tanh、Sigmoid 和ReLU 函數等非線性激活全都有飽和區間。



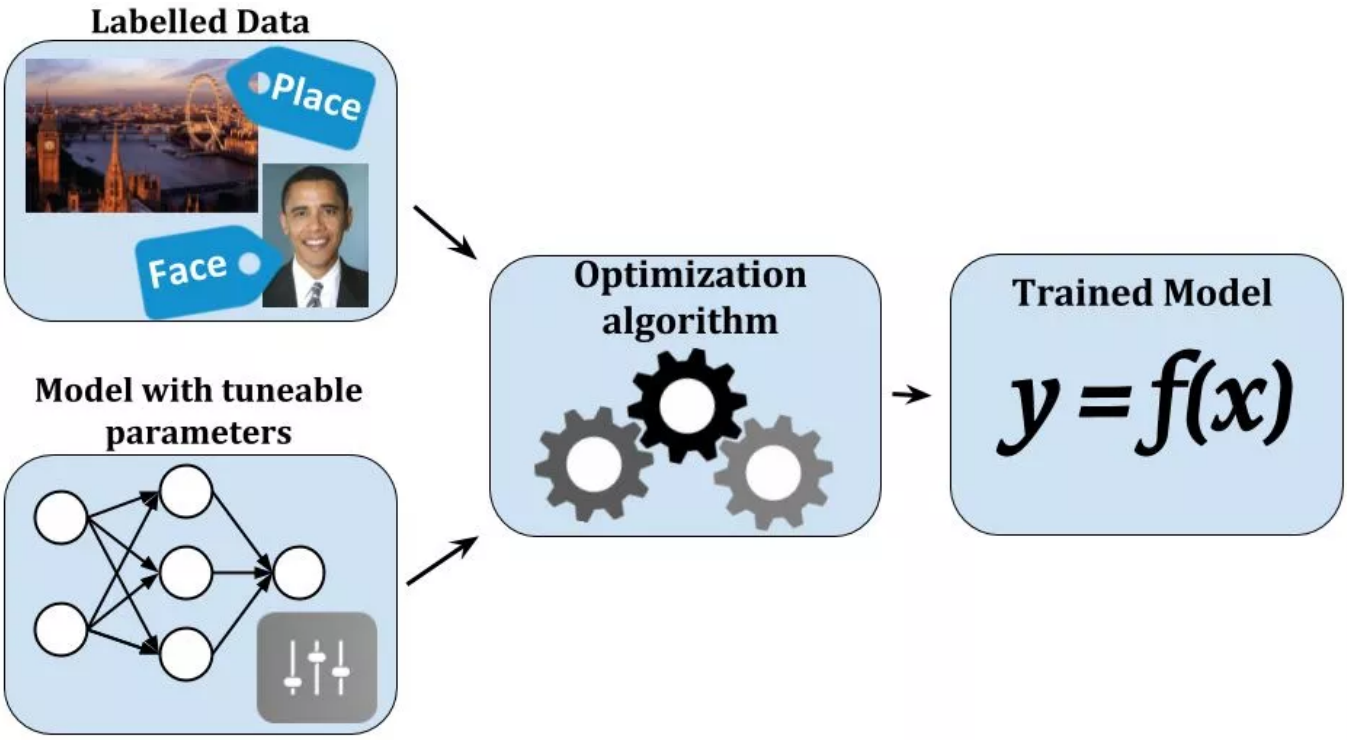
很容易理解，激活函數的飽和範圍就是當輸入值變化時輸出值不再變化的區間。這個變化區間存在兩個問題。

第一個問題是在神經網絡的前向方向上，落在激活函數的飽和範圍內的層的值將會逐漸得到許多同樣的輸出值。這會導致整個模型出現同樣的數據流。這個現像被稱為協方差偏移（covariance shifting）。

第二個問題是在反向方向上，飽和範圍內的導數為零，由此導致網絡幾乎無法再學習到任何東西。這就是我們在批歸一化問題中提到的要將值的範圍設定為零均值的原因。

問題7：模型的超參數是什麼？超參數與參數有何不同？

1. 模型參數是什麼？



先稍微回顧一下機器學習的本質，要做機器學習，我們需要有一個數據集。沒有數據我們怎麼學習呢？一旦有了數據，機器需要找到數據之間的關聯。

假設我們的數據是溫度和濕度等天氣信息，我們希望機器執行的任務是找到這些因素與我們的愛人是否生氣之間的關聯。這聽起來似乎並無關聯，但機器學習的待辦事項有時候確實很可笑。現在，我們用變量y 表示我們的愛人是否生氣，變量x_1、x_2、x_3……表示天氣元素。我們用下面的函數f(x)表示這些變量之間的關係：

$$y = f(x) = w_1.x_1 + w_2.x_2 + w_3.x_3$$

看到係數w_1、w_2、w_3 了嗎？這就代表了數據和結果之間的關係，這就是所謂的模型參數。因此，我們可以這樣定義「模型參數」：

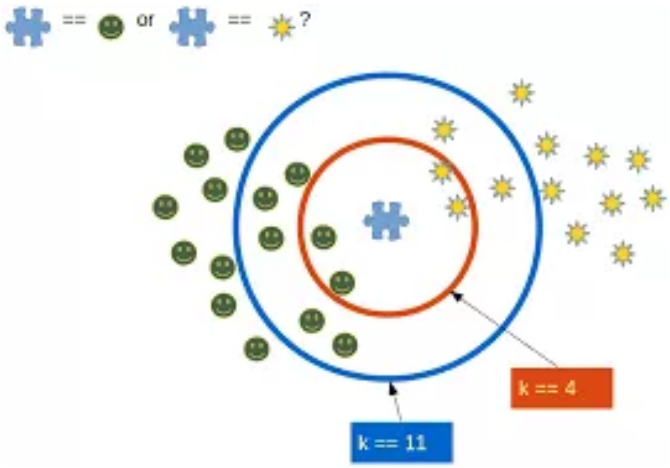
模型參數是模型基於訓練數據生成的值，有助於展示數據中數據量之間的關係。

所以當我們說要為某問題找到最佳的模型時，我們的意思是要基於已有的數據集為該問題找到最合適的模型參數。模型參數有如下特性：

- 可用於預測新數據；
- 能展現我們使用的模型的能力，通常通過準確度等指標表示；
- 是直接從訓練數據集學習到的；
- 不是由人類人工設置的。

模型參數也有不同的形式，比如在神經網絡中是權重、在支持向量機中是支持向量、在線性回歸和 logistic 回歸算法中是係數。

2. 什麼是模型超參數?



可能有人認為模型超參數就是或者像是模型參數，但事實並非如此。實際上這兩個概念是完全不同的。模型參數是從訓練數據集建模的，而模型超參數卻完全不是這樣，其完全位於模型之外而且不依賴於訓練數據。所以模型超參數的作用是什麼？實際上它們有以下任務：

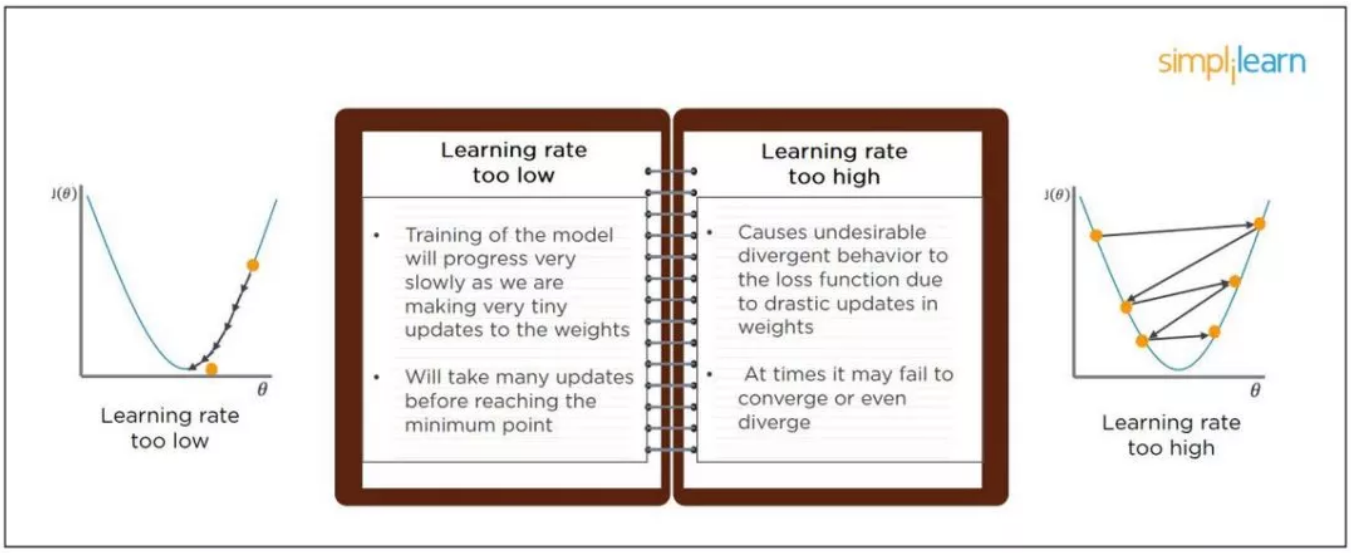
- 在訓練過程中使用，幫助模型尋找最合適的參數；
- 通常是在模型設計時由人工選擇的；
- 可基於幾種啟發式策略來定義。

對於某個具體問題，我們完全不知道最佳的超參數模型是怎樣的。因此，實際上我們需要使用某些技術（比如網格搜索）來估計這些值的最佳範圍（比如, k 最近鄰模型中的k 係數）。下面是模型超參

數的一些示例：

- 訓練人工神經網絡時的學習率指數；
- 訓練支持向量機時的C 和σ 參數；
- k 最近鄰模型中的k 係數。

問題8：當學習率過高或過低時會怎樣？

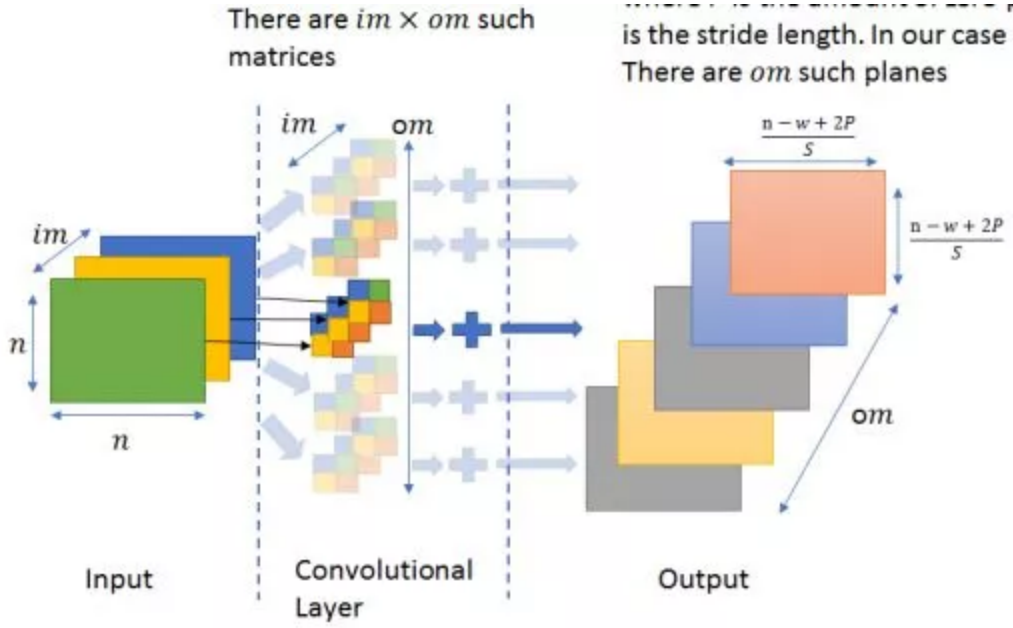


當模型的學習率過低時，模型的訓練速度會變得非常慢，因為其每次對權重的更新會變得非常小。模型將需要大量更新才能到達局部最優點。

如果學習率過高，模型很可能無法收斂，因為權重的更新過大。在加權的步驟中，模型有可能無法實現局部優化，然後使模型難以更新到最優點（因為每步更新都跳得過遠，導致模型在局部最優點附近搖擺）。

問題9：當輸入圖像的尺寸加倍時，CNN參數的數量會增加多少倍？為什麼？

對於參加面試的人來說，這個問題很有誤導性，因為大部分人思考這個問題的方向都是CNN 的參數數量會增加多少倍。但是，我們看看CNN 的架構：



Each output plane is generated by convolving the weight matrices with the input planes and summing over all the input planes, i.e.,

$$y_j = \sum_i x_i * w_{ij}$$

可以看到，CNN 模型的參數數量取決於過濾器的數量和大小，而非輸入圖像。因此，將輸入圖像的尺寸加倍不會改變模型的參數數量。

問題10：處理數據不平衡問題的方法有哪些?

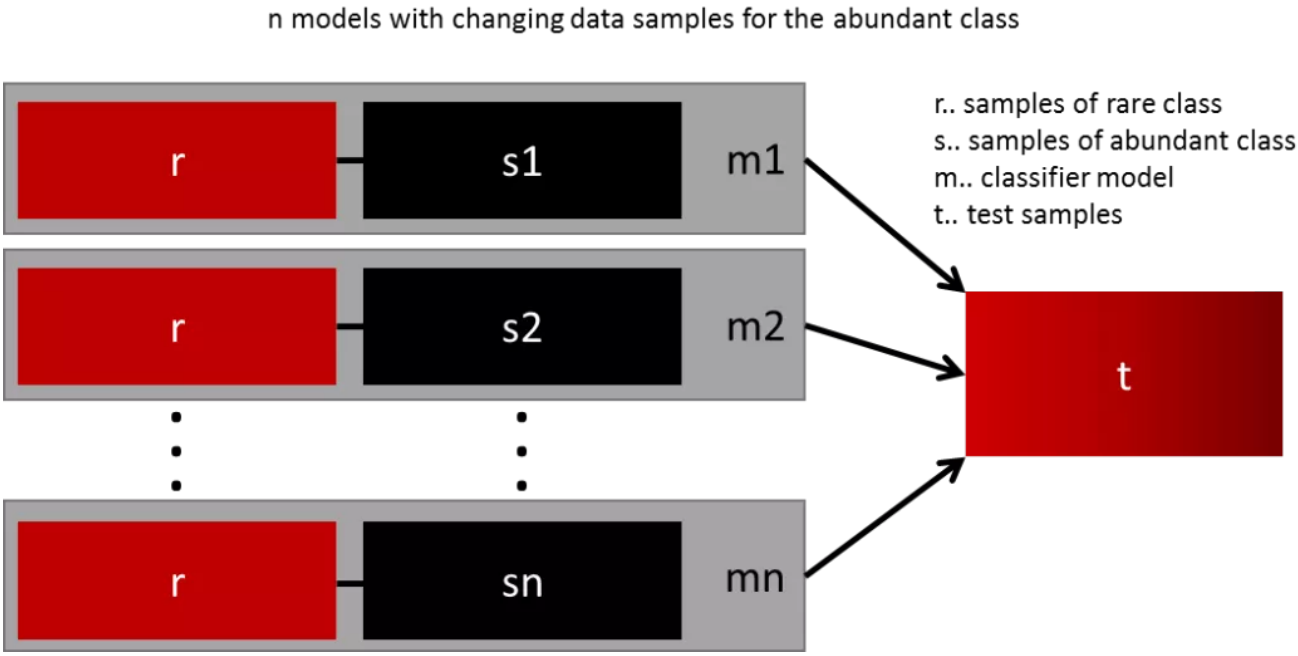
這個問題檢驗的是面試者是否知道處理有真實數據的問題的方法。通常來說，實際數據和样本數據（無需調整的標準數據集）在性質和數據量上都有很大的不同。使用真實數據集時，數據有可能是不平衡的，也就是說不同類別的數據不平衡。針對這個問題，我們可以考慮使用以下技術：

為模型的評估選擇適當的指標：當使用的數據集不平衡時，使用準確度來進行評估是很不合適的（前面已經提到過），而應該選擇精確度、召回率、F1 分數、AUC等評估指標。

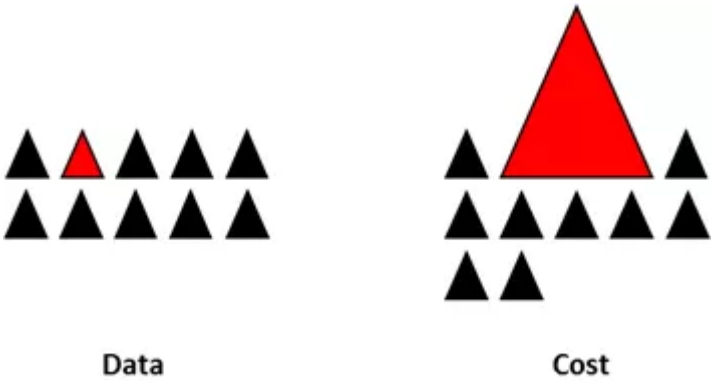
對訓練數據集進行重新採樣：除了使用不同的評估指標外，人們還可以通過某些技術來獲得不同的數據集。基於不平衡的數據集創建平衡的數據集的方法有兩種：欠採樣和過採樣，具體技術包括重複、自舉或SMOTE（合成少數過採樣技術）。

集成多個不同模型：通過創建更多數據來實現模型的通用性在實踐中是不可取的。舉個例子，假設你有兩個類別：一個有1000 個數據樣本的罕見類別以及一個有10000 個數據樣本的常見類別。我們可以不必努力為罕見類別尋找9000 個數據樣本來進行模型訓練，而是可以採用一種10 個模型的訓

練方案。其中每個模型都使用1000 個罕見數據樣本和1000 個常見數據樣本進行訓練。然後使用集成技術得到最佳結果。



重新設計模型——成本函數：在成本函數中使用懲罰技術來嚴厲懲罰數據豐富的類別，以幫助模型自身更好地學習罕見類別的數據。這能使損失函數的值更全面地覆蓋所有類別。



問題11：在訓練深度學習模型時，epoch、batch（批）和iteration（迭代）這些概念都是什麼意思?

這些是訓練神經網絡時非常基本的概念，但實際上很多面試者在區分這些概念時常常搞混淆。具體來說，你應該這樣回答：

- epoch：代表在整個數據集上的一次迭代（所有一切都包含在訓練模型中）；
- batch：是指當我們無法一次性將整個數據集輸入神經網絡時，將數據集分割成的一些更小的數據集批次；

- iteration：是指運行一個epoch 所需的batch 數。舉個例子，如果我們的數據集包含10000張圖像，批大小（batch_size）是200，則一個epoch 就包含50 次迭代（10000 除以200）。

問題12：數據生成器的概念是什麼？使用數據生成器需要什麼？

生成函數在編程中也非常重要。數據生成函數可幫助我們在每個訓練batch 中生成能直接擬合模型的數據。

使用生成函數在訓練大數據時大有助益。因此數據集並不是需要全部都載入RAM，這是浪費內存；此外，如果數據集過大，還可能導致內存溢出，對輸入數據的處理時間也會變得更長。

總結

上面就是我常在面試過程中向參加面試的人提出的12 個有關深度學習的面試問題。但是，根據每個面試者的情況不同，提問的方式可以也會各不相同，另外也會有其它一些根據面試者的經歷而提出的問題。

儘管這篇文章只涉及技術問題，但也是與面試相關的。在我個人看來，態度是面試成功的一半。所以除了讓你自己積累知識和技能之外，一定要用真正、進取又謙虛的態度展現你自己，這樣能讓你在對話中取得很大的成功。

參考鏈接：

https://medium.com/@itchishikicomm/12-deep-learning-interview-questions-you-should-not-be-missed-part-1-8a61f44cadac

AI學習路線和優質資源，在後台回复"AI"獲取