**梯度下降法**  
其目標是找到損失函數的最小值。在多維空間中，損失函數可被視為一個曲面，而梯度下降法需要找到這個曲面的最低點。梯度是一個向量，表示損失函數在某一點的斜率，如果沿著梯度的反方向即損失函數減少最快的方向前進就可以找到最小值或是局部最小值。

**反向傳播**  
是一種計算梯度的方法。在深度身神經網路中，反向傳播按照輸出層到輸入層的順序逐層計算梯度；而梯度可定義為損失函數對模型參數(權重和偏差)的偏導數。反向傳播會先計算的誤差，然後將這個誤差反向帶回以計算每一個前層的誤差和對應的梯度。以此方法來更新神經網路的權重以得到損失函數的最小值。

簡而言之，梯度下降法依賴反向傳播來提供更新權重所需的梯度訊息。

**超參數**

1. Learning rate  
   learning rate決定了梯度下降過程中參數更新的步長。learning rate太大可能無法收斂而找不到佳解(來回震盪)；learning rate太小則需要更多的時間來收斂也可能卡在局度最小值。而適時的調整learning rate可以加快學習速度並提高收斂性。

在對於不同大小的batch size所選用的learning rate也有所不同；當batch size小時，這時的梯度估計變異性較高，這增加了梯度下降過程的不穩定性，使用較小的learning rate可以防止模型做出過大的跳躍，從而減少損失函數的震盪。當batch size大時，這時的梯度估計的變異性較低，可以使用較大的leaning rate以提升收斂的速度，但也不能無限制增加，因為這樣可能錯過最損失函數的最小值。

關於learning rate的動態調整有以下幾種策略: **time-base decay, step-decay, exponential decay, adaptive learning rate**等方式。以達到快速收斂、避免過擬合(over fitting)、穩定訓練的目的。

1. Epoch  
   指整個訓練集被遍歷過一次的次數，過少的epoch會使模型未被充分訓練而表現不如預期，而過多的epoch則可能導致over fitting徒增計算成本。因此適時地調整epoch次數很重要。
2. Batch size

是從訓練集中抓出來進行權重更新的樣本數量。Batch size小，可以增加模型訓練過程中的noise避免模型over fitting，從而提高對新數據的泛化能力(Generalization，泛化能力表示模型有學習到規律而不是記憶訓練集的特定樣本)，不過batch size太小會造成梯度估計有著較大的變異，故訓練模型會較不穩定。Batch size大，可以提高模型的計算效率，不過也會增加over fitting的風險；另外，較大的batch size也能導致記憶體步不足。

1. Activate function  
   決定深度學習中神經元的輸出方式，它賦予身經網路非線性的特性，而對於學習複雜的數據分布有所幫助。Activate function在hidden layer中的將原本權重及偏差的線性組合轉換為非線性輸出至下一層，非線性增加了模型的學習能力。在hidden layer常見的activate function有: ReLU、Sigmoid以及Tanh。而output layer的activate function取決於任務的型態，二元分類可以使用Sigmoid函數將輸出壓縮在0~1之間故可以作為機率；在多分類則可以使用Softmax函數，Softmax能將輸出層的各個數值轉換為機率分布且所以機率總和為1。而回歸問題，因為是預測連續性的數值則可以視情況使用或不使用activate function，如要使用則可選擇linear activate function。
2. Loss function  
   作為衡量真實結果和模型預測結果之間的差異，數值越低表示模型預測結果越接近實際數據的表現。依照不同的問題可以選擇相對應的loss function才具備合理的參考性。回歸問題：可以使用MSE和MAE，兩者的差異在於MSE對於離群值比較敏感，可能導致髓師過度放大。MAE則是對離群較不敏感。分類問題：則可以使用cross entropy，它對於二元或是多元分類的機率量化的處理比較好，當預測值和實際值差異很大的時候，損失會增加得很快而能凸顯分類的差異性。不過使用上需注意輸出值的機率在合理的範圍。Loss function的梯度和learning rate共同決定了參數更新的幅度，如果學習率設定不當，即使損失函數設計得很好，模型也可能無法收斂。另外，為了防止模型over fitting會在loss function中加入正則化(Regularization)，能夠避免模型過度複雜。L1正則化是對權重取絕對值之和，L2正則化是對權重取平方和。正則化能有效抑制權重，避免權重過度放大而梯度消失。