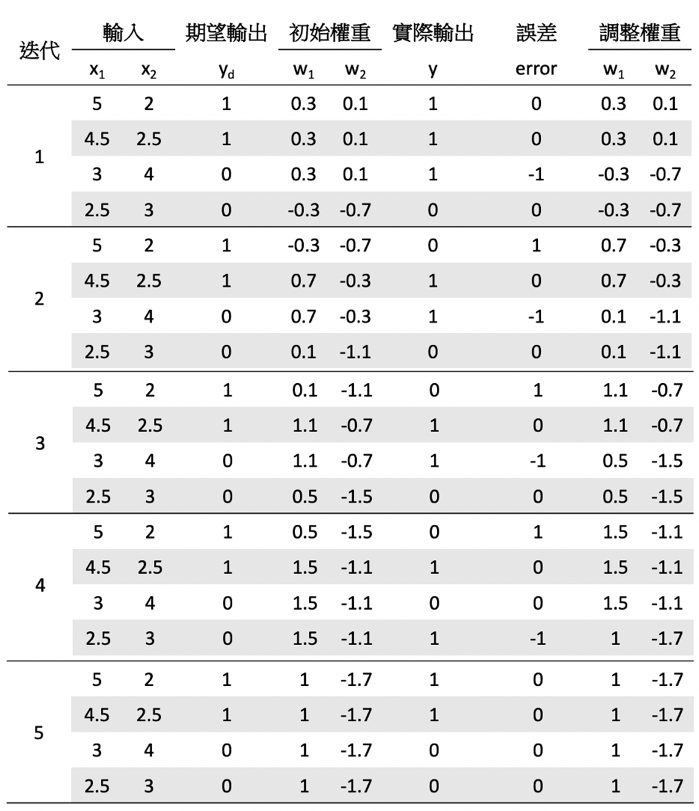
類神經網路

專題實作

班級：＿＿＿＿＿＿＿＿ 姓名：＿＿＿＿＿＿＿＿ 座號：＿＿＿＿＿＿＿＿

◎「學習分類昆蟲資料」

　　我們在課程中，用了昆蟲資料的例子來理解類神經網路的「學習演算法」。本課程最後的專題實作，需要你實作出這個例子中的「訓練過程」。課程的初期，我們也使用程式來實作「類神經網路」，在學習到「激勵函數」時，我們也運用程式來實作各式各樣的「激勵函數」。接下來，請依照專題文件的引導，搭配老師附上的程式範例檔，一步步寫出類神經網路的訓練過程吧。



**步驟一：輸入資料與設定起始權重**



　　由於昆蟲資料有四筆，所以可以看到Data變數的部分，存入了四筆資料，其中有每筆資料的「長度」、「寬度」。而「類別」則是存在Yd變數中，作為「期望輸出」。起始的權重我們設定為[0.3, 0.1]，與教材當中的起始權重相同，事實上類神經網路在訓練前，通常是以亂數產生起始權重，但專題為了方便檢視運算過程有沒有正確，所以統一設定[0.3, 0.1]。

|  |
| --- |
| **程式碼：** |

　　設定完資料與權重後，可以執行下方的程式，熟悉一下如何輸出陣列中的每筆資料，這個概念在後續的專題內容會使用到。

|  |
| --- |
| **程式碼：** |

**步驟二：定義類神經網路函數**

|  |
| --- |
|  |

上圖是我們在課堂上所使用的類神經網路模型，𝜃值與𝛼值都設定為0.2，所以後續我們在程式中，也設定為0.2。

　　在程式範例檔當中，你可以找到一個為完成的「nn函數」，這個步驟是要來完成它中間被挖空的程式。

|  |
| --- |
| **程式碼：** |

|  |
| --- |
| **提示：**  　　試著執行下方的程式，在這邊就是將資料的「長、寬」與對應的權重相乘後加總，再減去𝜃值。  **程式碼：** |

　　若將挖空的程式完成，可以試著執行下方的程式，其輸出結果應該要與提示的程式相同。如果不同，可能需要再想想哪裡寫錯了。

|  |
| --- |
| **程式碼：** |

**問題1-1：請將你完成的「nn函數」截圖貼在下方。**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **評分標準** |  | | 完成程式並截圖 | 7分 | |

**問題1-2：請說明為什麼「第二筆資料」輸入到「nn函數」後，輸出值為「1.4」，簡述運算過程即可。**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **第二筆資料為[4.5, 2.5]，**  **權重為[0.3, 0.1]，theta設定為0.2。**  **輸入進nn函數後，**  **x[0]為4.5、x[1]為2.5、w[0]為0.3、w[1]為0.1，**  **所以X = 4.5\*0.3 + 2.5\*0.1–0.2 = 1.4。**   |  |  | | --- | --- | | **評分標準** |  | | 正確說明哪些參數輸入至函數  (包含x、w、theta) | 每個2分 | | 正確說明x值陣列中的數值如何被提取 | 4分 | | 正確說明w值陣列中的數值如何被提取 | 4分 | | 清楚寫下X值的計算過程 | 6分 | |

**步驟三：定義「步階函數」**

　　因為在本例中，是使用「步階函數」作為「激勵函數」，所以需要用程式定義此函數。

|  |
| --- |
| **程式碼：** |

　　「步階函數」在輸入值大於或等於0時，會輸出+1；反之會輸出0，試著想想空格中需要填入什麼吧。

　　完成後，你可以運用下方的程式碼測試你寫好的「步階函數」，如果輸出值相同，那就代表你寫對了。

|  |
| --- |
| **程式碼：** |

**問題2-1：請將你完成的「步階函數」截圖貼在下方。**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **評分標準** |  | | 完成程式並截圖 | 5分 | |

**問題2-2：請說明為什麼「測試步階函數」的時候，第一筆輸出為「1」，第二筆則為「0」。**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **第一筆輸出為「1」，**  **是因為輸入值為5，而5大於0，**  **程式則回傳1。**  **第二筆輸出為「0」，**  **是因為輸入值為(−1)，而(−1)小於0，**  **程式則回傳0。**   |  |  | | --- | --- | | **評分標準** |  | | 正確說明函數的輸入值 | 每個3分 | | 正確說明步階函數當中的條件判斷 | 每個3分 | |

**步驟四：將「nn函數」與「步階函數」串接在一起**

|  |
| --- |
|  |

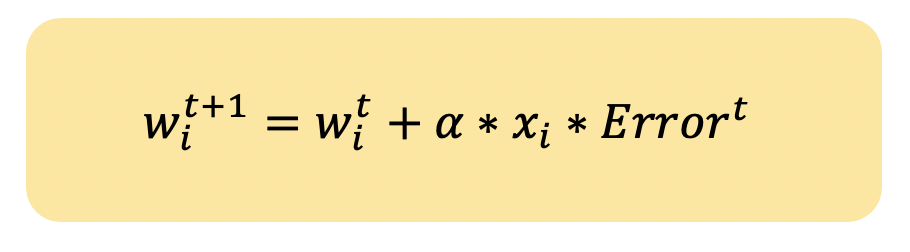
在先前的步驟，是分別完成這個模型的不同部分(橘、紅框處)，在這步驟，我們將這兩個部分串接在一起。

|  |
| --- |
| **程式碼：** |

　　這部分你只要執行就可以了，依照上方程式碼定義，就可以把兩個部分串接在一起，執行後可以接續執行下方的程式，如果目前的輸出值都是1，那你目前所寫的程式應該沒有錯誤。

|  |
| --- |
| **程式碼：** |

**步驟五：定義「學習演算法」**



　　在課程中，我們了解到「學習演算法」，但在程式當中該如何實作呢？請先在程式範例檔中，找到下方的程式碼，再閱讀接續的提示，思考看看如何完成吧。

|  |
| --- |
| **程式碼：** |

　　首先，由於「學習演算法」需要「誤差」來做權重調整，所以你可以在「new\_w函數」的輸入值看到「y」、「yd」，「y」是用來輸入類神經網路的輸出值，「yd」則是輸入「期望輸出」，所以在此可以先思考「error」應該如何定義。

　　接著是「權重」的部分，雖然上方的演算法有t紀錄「迭代」到第幾次，但我們在本專題中，先將更新的權重直接存在原本的變數中。只要思考w[0]、w[1]後方要怎麼定義。

　　如果定義好「學習演算法」，可以試著執行下方的程式碼，輸出的結果與截圖的相同，那也代表你寫的程式應該沒錯。

|  |
| --- |
| **程式碼：** |

**問題3-1：請將你完成的「學習演算法」截圖貼在下方。**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **評分標準** |  | | 完成程式並截圖 | 8分 | |

**問題3-2：請說明為何「第三筆資料」在被讀取後，類神經網路會將權重調整為[-0.3, -0.7]，簡述運算過程即可。**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **第三筆資料為[3, 4]，權重為[0.3, 0.1]，**  **經過nn\_step函數計算出的y值為1。**  **Yd[2]為0，也就是期望輸出為0，**  **alpha值被設定為0.2，**  **所以在new\_w中計算error為(−1)，**  **w[0]則調整為0.3+0.2\*3\*(−1) = −0.3，**  **w[1]則調整為0.1+0.2\*4\*(−1) = −0.7。**   |  |  | | --- | --- | | **評分標準** |  | | 正確說明哪些參數輸入至函數  (包含x、w、y、yd、alpha) | 每個2分 | | 正確說明y值是透過nn\_step函數計算出來 | 5分 | | 正確說明期望輸出是如何從Yd中提取出來 | 5分 | | 清楚寫下調整權重的計算過程 | 7分 | |

**步驟六：讓訓練過程執行「五次迭代」**

　　最後，我們要讓「學習演算法」執行五次迭代，在上個步驟我們已經定義好「學習演算法」，而且在執行前一步驟最後的程式碼，也知道我們應該如何逐步地執行學習演算法讓權重被調整。

　　現在我們需要利用「for迴圈」的方式，讓整個過程更自動化，以下是我們會運用到的雙層迴圈結構，請參考它的執行結果，並思考該如何修改。

|  |
| --- |
| **程式碼：** |

　　依照上方的程式提示，可以發現j表示讀取第幾筆資料，而i代表著第幾次迭代，再多加參考前一步驟的程式，思考看看該如何將下方的程式碼整合到「for迴圈」當中。

|  |
| --- |
| **程式碼：** |

　　如果你在此步驟是成功的，試著執行看看，你的程式輸出結果應該會與下方相同。

|  |
| --- |
| **程式碼：** |

**問題4-1：請將你修改好的「for迴圈」截圖貼在下方。**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | **評分標準** |  | | 完成程式並截圖 | 9分 | |

**問題4-2：請簡述這個「for迴圈」的運作過程。**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **在「for j in range(4)」這個迴圈中，**  **會讀取每一筆資料並且調整權重，**  **最後透過「print(w)」輸出每次調整完的權重，**  **而在「for i in range(5)」這個迴圈中，**  **是讓這整個過程重複5次迭代。**   |  |  | | --- | --- | | **評分標準** |  | | 正確說明內層迴圈的目的是讀取資料並調整權重 | 6分 | | 正確說明外層迴圈的目的是進行5次迭代 | 6分 | |