人工智慧 AI HW1

系級:資工111 姓名:李展緯 學號:40747044s

1. 我的AI是用來預測天氣的降雨機率，而資料是由The Weather Channel及中央氣象局取得，輸入項目為溫度、濕度、風力及風向共4個項目，其中溫度調整為介於0到1之間((溫度-台灣歷史最低溫)/台灣歷史最高溫-台灣歷史最低溫)，濕度本為0到1之間，風力調整至0到1之間(風力/100)，風向則亦調整至0到1之間(以北為零度順時針為正，風向度數/360)，輸出則為降雨機率一項。而我用9組當作訓練資料，9組當作測試資料。機器的軟體為Microsoft window10，硬體為Intel(R)Core(TM)i5-8250U [CPU@1.6GHz](mailto:CPU@1.6GHz) 系統類型:64位元作業系統，x64型處理器，而我的聯絡電話為0987654321
2. 一個神經元: activation function : tanh()

次數:1000次 時間:0.59(s) 訓練MSE:0.237 測試MSE:0.1532

次數:5000次 時間:1.718(s) 訓練MSE:0.222 測試MSE:0.063

次數:10000次 時間:3.21(s) 訓練MSE:0.236 測試MSE:0.132

次數:50000次 時間:29.53(s) 訓練MSE:0.235 測試MSE:0.113

首先將9組資料設成9\*4的陣列，降雨機率設成9\*1的陣列，將訓練資料丟進f(x) = exp(2x)-1/exp(2x)+1的activation funtion，出來的結果計算MSE，並將結果丟進權重調整公式 :

weight = weight+train\*(結果-理想)\*(1-結果2)

1. 一個神經元: activation function : sigmoid()

次數:1000次 時間:0.359(s) 訓練MSE:0.039 測試MSE:0.072

次數:5000次 時間:1.546(s) 訓練MSE:0.0357 測試MSE:0.099

次數:10000次 時間:2.968(s) 訓練MSE:0.0357 測試MSE:0.101

次數:50000次 時間:17.093(s) 訓練MSE:0.0357 測試MSE:0.101

首先將9組資料設成9\*4的陣列，降雨機率設成9\*1的陣列，將訓練資料丟進f(x) = 1/1+exp(x)的activation funtion，出來的結果計算MSE，並將結果丟進權重調整公式 :

weight = weight+train\*(結果-理想)\*結果(1-結果)

結果發現，sigmoid 比起 tanh 對於資料來說更為精準，

1. 我將兩個神經元前後相連接，並且皆用sigmoid () 函數作為activation function ，但因為兩個神經元的角色不一樣，因此權重的更新也不一樣，Backprogation 推導如下，其中:

X1:第一神經元輸入

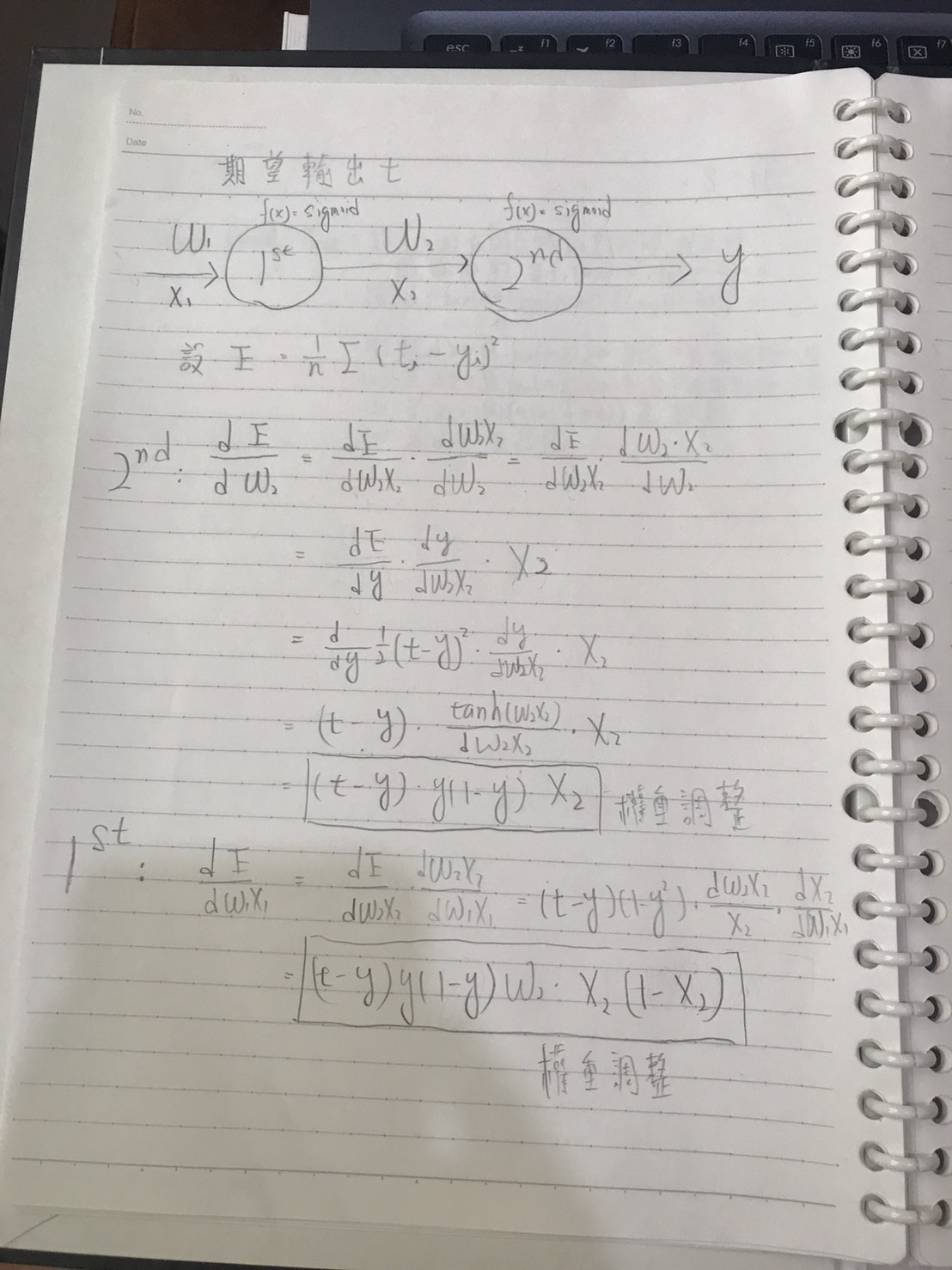
W1:第一神經元輸入的權重

X2:第一神經元輸出，同時也是第二神經元的輸入

X2:第二神經元輸入的權重

Y:第二神經元的輸出

E:目標輸出與實際輸出的誤差



1. 二個神經元:

第一個神經元 : activation function : sigmoid() bias:-0.82

第二個神經元 : activation function : sigmoid() bias:-0.132

次數:1000次 時間:0.343(s) 訓練MSE:0.052 測試MSE:0.017

次數:5000次 時間:1.546(s) 訓練MSE:0.0517 測試MSE:0.0205

次數:10000次 時間:4.000(s) 訓練MSE:0.0517 測試MSE:0.0206

次數:50000次 時間:18.95(s) 訓練MSE:0.0517 測試MSE:0.0206

1. 我嘗試了用 tanh 的activation function 來設計兩個神經元，但因為我的輸入資料皆為 0~1，權重的調整使得權重值越來越小，而tanh的函數為e-2x-1/e-2x+1，因此當資料輸入後，數值會越來越小，最後overflow。我嘗試修改權重的調整幅度，但並無太大的效果。因此我認為我的資料不太適合用tanh來做為activation function
2. 將第一與第二神經元分別兩個輸入，並將輸出的結果併成第三神經元的輸入，而每個神經元的activation function 皆為 sigmoid 函數，其中

X1:第一神經元輸入 W11:第一神經元輸入的權重

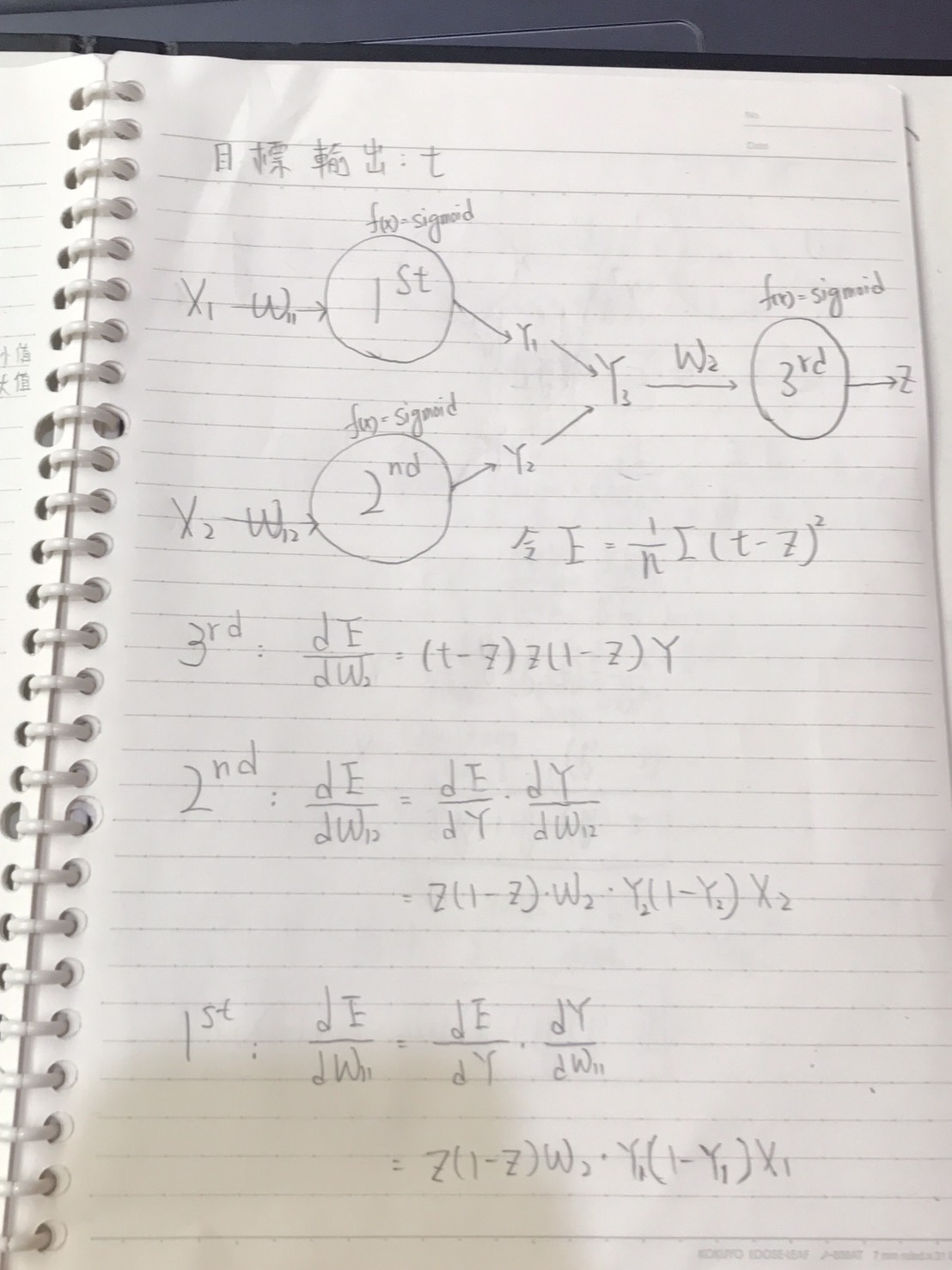
X2:第二神經元輸入 W12:第二神經元輸入的權重

Y1:第一神經元的輸出 Y2:第二神經元的輸出

Y3:將Y1、Y2合併 W2:Y輸入的權重

Z:第三神經元輸出

E:目標輸出與實際輸出的誤差



1. 三個神經元:

第一個神經元 : activation function : sigmoid() bias:-2

輸入資料: 四組 特性: 降雨機率偏高

第二個神經元 : activation function : sigmoid() bias:-1

輸入資料: 四組 特性: 降雨機率偏低

第二個神經元 : activation function : sigmoid() bias:-0.1

將前兩個神經元合併資料合併並乘上權重在放入funtion中

次數:1000次 時間:0.468(s) 訓練MSE:0.02815 測試MSE:0.00709

次數:5000次 時間:2.421(s) 訓練MSE:0.02809 測試MSE:0.01032

次數:10000次 時間:4.921(s) 訓練MSE:0.028107 測試MSE:0.01228

次數:50000次 時間:24.078(s) 訓練MSE:0.028126 測試MSE:0.01414

在測試過程中，我發現當訓練次數在5000到10000時，訓練的MSE會接近最小，而在訓練次數較少時，測試的MSE會接近最小，

1. Alphago的勝利，為機器人學習的領域帶來嶄新的變化，透過演算法，AI可以透過過去的棋盤，推算出在未來最好的一步棋，這讓全世界的人更加關注AI的這項科技。雖然人類被打敗了，但我們可以學習AI的下法，從中學習，這也是人類自古以來的生存法則，而我們也不用太擔心，因為AI目前還無法做到全面取代人類，而我們要做的便是學習一些AI目前還無法學習的東西。未來AI可以應用在多種領域，而其中，自動駕駛是最令我感興趣的，因為在這個交通便利的時代，各種交通工具都能四通八達，然而還是需要人駕駛，而人會有疲憊的時候，這時候自動駕駛便是最好的替代方案，當然自動駕駛不是簡單的開車，對於人體物體的辨識，方向的掌握，速度的拿捏，都是需要經過精密的設計與訓練的。
2. 進入到AI的領域後，又是一個新的世界，許多東西要學，像是如何將資料做處理，例如一開始不知道該怎麼將資料處理好，因此導致輸入的資料超過funtion的界線，因而跑出的數值都是一點多，超過理想值，還有設計三個的神經元時，資料該如何分配到各個神經元處理，以及輸出資料對於權重的調整。
3. 參考文獻

The weather channel:

<https://weather.com/zh-TW/weather/today/l/TWXX0021:1:TW>

中央氣象局:

<https://www.cwb.gov.tw/V8/C/>

零基礎深度學習-神經網路及反向傳播:

<https://www.zybuluo.com/hanbingtao/note/476663>