姓名:_____
學號:_____

4105931機器學習 (Machine Learning)
Midterm
總分100 分
中文作答

得分

1. a) (2%) 試解釋什麼是'機器學習'? b) (5%) 試舉出五種可以應用機器學習的實際問題?

Ans:

a) In Lecture 2 p.5
利用資料來學習出資料的特性和模式，進而能做出決策或增進效能衡量
b) In Lecture 2 p.13
人臉辨識、信用卡核卡、語音辨識、機器人對話、文本生成

2. (8%)
有鑑於理工科的男生都不太會穿搭，所以我做了下面這樣的事情。我首先收集了2018年整年的紐約時裝周服飾照片，然後依照影像處理的作法，擷取了顏色、花紋、材質三種特徵，每一張影像將三種特徵串接成一種新的特徵表示法，然後分群成100群，接著我拿出我自己穿搭的照片，取出這種新的特徵表示法，跟100群每一群的群中心算距離，假設距離小於預設的門閥值50，我就得一分，看看最終總得分的高低，來決定我的穿搭帥不帥。試以'Type of Learning'的角度，說明這個方法是屬於什麼樣的機器學習方法?

Ans: In Lecture 3 p.2
1. Output space y: Regression
2. Different Data Label y : Unsupervised Learning
3. Different Protocol f : Batch Learning
4. Different Input Space x : Concrete Features

3. (7%) 當使用某一個已經分好訓練、和測試資料的dataset，如何利用分成三包的交叉驗證技巧，訓練一個好的模型? 請從訓練、驗證到最終的測試，詳細列出全部的步驟。

Ans: In Lecture 4 p.17
在3-fold cross validation中，在我們有只有Training set與Testing set而沒有Validation set的情況下，首先將Training set分成三等份且互斥的subset，每次進行訓練時從中依序挑選兩個subset做為training data，剩下一個subset做為validation data，每經過三次訓練後，將這三次validation的結果計算平均，做為cross validation的驗證結果.Cross Validation驗證沒問題後，接下來用所有的訓練資料訓練模型，最後再由Testing set來對訓練完成的模型進行最終的測試。

4. (7%) 如果我想訓練一個 "人生勝利組預測模型"，知道一個人到60歲的時候是溫拿(winner)還是魯蛇(loser)，試說明如何設計這樣的模型? 從資料、特徵、模型、損失函數、訓練、到測試，說明整個步驟。

Ans: In Lecture 4, p24
首先我們須先定義60歲時的winner、loser的特徵(標準)，例：年薪>1千萬、財產>1億、有交往對象，以上條件皆符合則稱為溫拿，若其中有條件不符則為魯蛇，再來我們收集60歲為溫拿或魯蛇年輕時的資料，由於所蒐集資料與時間相關，我們訓練一個時序性模型擷取資料特徵，最後再利用一個二元分類器對溫拿及魯蛇進行分類，訓練完成後，我們將測試者的資料傳入　　模型即可得到一個兩類別的機率(溫拿：1% ｜魯蛇：99%)作為預測結果。

資料：60歲的溫拿與魯蛇的資料，資料內容如下: (年齡、年薪、財產、交往對象有無)
特徵：溫拿 - 年薪>1千萬 and 財產>1億 and 有交往對象
　　魯蛇 - 年薪<1千萬 or 財產<1億 or 無交往對象
模型：Neural network(分類器)
損失函數：binary crossentropy (二元交叉熵)

$$loss = -\sum_{i=1}^{n} \hat{y}_i log y_i + (1-\hat{y}_i) log(1-\hat{y}_i)$$

訓練：將所有蒐集到的資料以6:2:2比例劃分為Training set、Validation set與Testing set，對Training set進行訓練，並在每個訓練週期(epoch)使用validation set進行驗證。
測試：等到模型訓練完成後(loss curve收斂)，利用testing set進行測試。

5. a) (2%)試說明什麼是Perceptron? b) (2%)試說明Perceptron Learning Algorithm中，尋找分類錯誤的x
，為什麼可以用公式 ? c) (4%)試說明w_{t+1}的更新為什麼可以用$w_t + y_{n(t)}x_{n(t)}$取代? d) (2%)
保證能讓Perceptron Learning Algorithm的停止條件是什麼?如何能保證演算法一定停的下來? e) (2%)
試比較Perceptron Learning Algorithm與Pocket Algorithm相同與相異之處為何? f) (4%)試說明若資料維度是d
，為什麼Perceptron Hypothesis h(x)的維度是d+1?

Ans:

a)一個線性二元分類器

b) (Lecture 5, p.8)

在直線方程 $w_0+w_1x_1+w_2x_2=0$中，平面上的點落在直線右邊，$w \cdot x > 0$；

否則 $w \cdot x < 0$，若sign($w \cdot x$)與x的label不同，則代表被直線分到不同邊(分類錯誤)。

c) (Lecture 5, p.9)

利用Wt的向量與X的向量做內積，且利用sign((t) ≠ (t))

來找出錯誤的點。假如有一個被分類錯誤的點預期輸出為正時，代表Wt與X的向量夾角過大，所以Wt要往X向量移動，也就是Wt + X來修正Wt+1的向量，反之Wt則是遠離X向量，Wt – X，最後修正到沒有錯誤的點為止。

d) (Lecture 5, p.10)

(1)資料為線性可分

(2) 因為w每一輪的更新，都可以讓w越來越接近理想中正確的w_f。

e) (Lecture 5, p.9、p.28)

相同: 在線性可分中，更新w的原理一樣

相異: 在每一次的iteration中，Pocket 需要確認Wt+1做完所有的資料後，整體結果有沒有比W好才更新，PLA每次只看一筆資料，不用算完所有的資料。因此Pocket比PLA慢，且Pocket演算法可以用在線性不可分。

PLA

Pocket

f) (Lecture 5, p.18)

多出來的一維為X0對應到threshold (W0)上來做內積，X0 = 1

6. 假設dataset X有五筆資料$x_1, x_2,\ldots,x_5$，其資料維度為2，每一筆資料的答案為$y_1,y_2,\ldots,y_5$，a) (2%)試寫出要求出$y_1,y_2,\ldots,y_5$的regression model公式解。b) (3%)假設X^TX可逆，試寫出公式解中每一個矩陣或向量的維度。c) (2%)
假設datatset每一筆資料維度為10，利用這個dataset求出的線性回歸模型，共會有多少的參數?

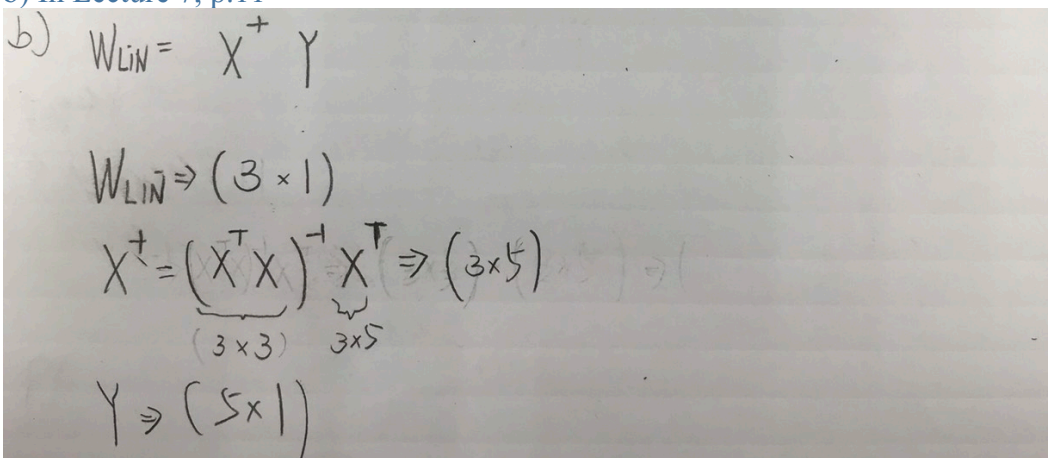
Ans:

a) In Lecture 7, p.11、p.12

，，…，

，

b) In Lecture 7, p.11



c) In Lecture 7, p.12

Dimension of W : 11*1 -> 共有11個參數

7. a) (3%)試說明PLA演算法跟Gradient Decent 演算法，在表示w_{t+1}的解時，是如何寫成相同的形式。b)(2%)Logistic Regression演算法中，若資料筆數是10，資料維度是2 ，label維度是1，則 維度是多少? c) (2%)承上題，

的維度是多少?

Ans:

a) In Lecture 8 p.20

$$\mathbf{w}_{t+1} \leftarrow \mathbf{w}_t + \underbrace{1}_{\eta = \nabla E_{in}(w_t)} \cdot \underbrace{\left(\left[\!\left[\operatorname{sign}\left(\mathbf{w}_t^T \mathbf{x}_n\right) \neq y_n\right]\!\right] \cdot y_n \mathbf{x}_n\right)}_{v = \|\nabla E_{in}(w_t)\|}$$

在PLA更新公式當中，可以把最後一項的 η 看成是gradient decent中的step size，$\left[\!\left[\operatorname{sign}\left(\mathbf{w}_t^T \mathbf{x}_n\right) \neq y_n\right]\!\right] \cdot y_n \mathbf{x}_n$ 可以看成gradient decent中的梯度更新的方向 $-\frac{\nabla E_{in}(\mathbf{w}_t)}{\|\nabla E_{in}(\mathbf{w}_t)\|}$。

η 是包含梯度 $-\frac{\nabla E_{in}(\mathbf{w}_t)}{\|\nabla E_{in}(\mathbf{w}_t)\|}$ 的分母(投影片中紫色的) η ; v是只有 $-\frac{\nabla E_{in}(\mathbf{w}_t)}{\|\nabla E_{in}(\mathbf{w}_t)\|}$ 的分子

b) 2 (In Lecture 8, p.17)

c) 1 (In Lecture 8, p.17)

7.

b)

$$\nabla E_{in}(w_t) = \frac{1}{N} \sum_{n=1}^{N} \theta\left(-y_n w_t^T x_n\right)\left(-y_n x_n\right)$$

dimension : $x : 2$, $y : 1$

size : $x : 10$

$$\nabla E_{in}(w_t) = \frac{1}{N} \sum_{n=1}^{N} \theta\left(-y_n \begin{bmatrix} w_{t,1} & w_{t,2} \end{bmatrix} \begin{bmatrix} x_{n,1} \\ x_{n,2} \end{bmatrix}\right)\left(-y_n \begin{bmatrix} x_{n,1} \\ x_{n,2} \end{bmatrix}\right)$$

Dimension : $(1) \times (1 \times 2) \times (2 \times 1) \times (1) \times (2 \times 1)$

$\Rightarrow (2 \times 1) \Rightarrow$ 2-dimension ※

c)

$$\theta\left(-y_n w_t^T x_n\right)$$

$$= \theta\left(-y_n \begin{bmatrix} w_{t,1} & w_{t,2} \end{bmatrix} \begin{bmatrix} x_{n,1} \\ x_{n,2} \end{bmatrix}\right)$$

Dimension : $(1) \times (1 \times 2) \times (2 \times 1)$

$=$ 1-dimension ※

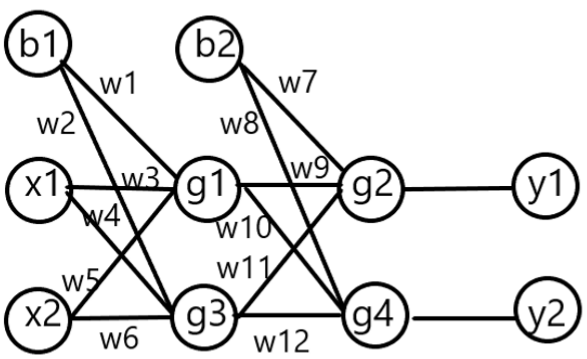
8. a) (2%)說明深度學習架構中，一個節點、一層與多層，其功能差異為何? b) (4%)
試以數學證明多次線性組合的結果還是線性。c) (2%)如何引入非線性的機制到深度學習模型中? d) (3%)2-4-6-2 NNet的模型參數總共有多少個? e) (4%)詳細說明Backpropagation Algorithm每一個步驟的意義。

Ans:

a) Lecture 6, p.7、p.8

每一個節點代表每一個Perceptron，用來進行二元分類

每一層裡有多個Perceptron，主要是能呈現出smooth boundary
多層結合則是表示更複雜的分類界線，如XOR
b) Lecture 6, p.13



g1(x) = x1*w3 + x2*w5 + b1*w1
g3(x) = x1*w4 + x2*w6 + b1*w2
g2(x) = g1*w9 + g3*w11 + b2*w7 = y1
g4(x) = g1*w10 + g3*w12 + b2*w8 = y2

g1,g3 代入 g2,g4

g2 = (x1*w3 + x2*w5 + b1*w1)*w9 + (x1*w4 + x2*w6 + b1*w2)*w11 + b2*w7 = (w3*w9+w4*w11)*x1+(w5*w9+w6*w11)*x2 +(w1*w9+w2*w11)*b1+ r1*b1 = α1*x1 + β1*x2+ γ1*b1

G4 = (x1*w3 + x2*w5 + b1*w1)*w10 +(x1*w4 + x2*w6 + b1*w2)*w12 +b2*w8 = (w3*w10+w4*w12)*x1+(w5*w10+w6*w12)*x2+(w1*w10+w2*w12)*b1 + r2*b1 = α2*x1 + β2*x2 + γ2*b1

由此可以看出結論：雖然 g2在hidden layer的第二層，但可以直接寫成x1, x2, b1的線性組合，表示第一層白做了。

c) Lecture 6, p.13、p.14
加入non linear activation function
d) Lecture 6, p.17
(2+1)*4 + (4+1)*6 + (6+1)*2 = 56
e) 初始化所有權重wij(l)
For t =0,….,T
(1) stochastic:隨機從{1,2,….,N}中選取n個 (因採用SGD方式做梯度下降，故須先隨機選取)
(2) forward:用$x^{(0)} = x_n$計算所有的 $x_i^{(l)}$ (backward的起始步驟需要參考到forwarding output計算出的誤差，因此須先做forwarding pass，從第1層往後算至第l層)
(3) backward:由$x^{(0)} = x_n$計算出所有的$\delta_j^{(l)}$ (gradient descent步驟須參考到$\delta_j^{(l)}$，從第l層往前推至第1層)
(4) gradient descent: $w_{ij}^{(l)} \leftarrow w_{ij}^{(l)} - \eta\, x_i^{(l-1)}\, \delta_j^{(l)}$ (更新權重值)
RETURN gNNet(x) =(…tanh($_{ij}^{(1)}$x$_j$)))

9. a) (3%)試說明Convolutional Neural Network與一般類神經網路架構的相同處為何? b) (3%) 與一般類神經網路相比，CNN如何降低模型的參數量? c) (4%)訓練一般類神經網路模型時，要學習的參數是什麼? 訓練CNN時，要學習的參數是什麼? d) (2%)何謂mini-batch? e)(2%)說明epoch, iteration與batch size的關係。

Ans:
a) Lecture 9, p.14、p.15、p.16
CNN在做完convolution後，需要將convolution parameter展開(flatten)做全連接fully connected layer，在這一部份和一般類神經網路架構相同
。此外，前面作convolution層，概念上也是跟作全連接層一樣，只是沒有對到filter的地方，權重是0
b) Lecture 9, p.15
使用convolution時inputs資訊量只會有n*n (n = # of kernel / filter size)
，而非全連接所有神經元；並且filter再進行convolution時參數共享的，所以會有較少參數量；max pooling 也可以降低參數量。

c) Lecture 9, p.14、p.15、p.16
NN : 不同fully connected layer 中 neuron 彼此連接的權重。
CNN : convolutional layer裡面的每一層所有的filters+ fully connected layer 中 neuron間連接的 權重。
d) Lecture 9, p.26
在作Backpropagation演算法時，去計算gradient decent公式所用的最小資料量。
e) Lecture 9, p.26

Epoch = Iteration * Batch Size

10. a) (5%)請提出兩個方法，能增進你對課程的理解。b) (5%)寫出對本課程的建議事項。
Ans: