4105931機器學習(Machine Learning)

Midterm

總分**100** 分

中文作答

1. a) (5%) 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?

(a) (依據編寫完整性給分) (5分)

機器學習(Machine Learning = ML)是透過演算法將收集到的資料進行分類或預測模型訓練，當未來得到新的資料時，可以透過訓練出的模型進行預測 ，如果這些效能評估可以透過利用過往資料來提升的話，就叫機器學習。

(b) (各 1 分)

人臉辨識、信用卡核卡、語音辨識、機器人對話、文本生成

2. (10%)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 10000 張照片，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 200 張照片，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，來決定我的穿搭夠不夠時尚。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?

(各 2.5 分)

Output space y	regression
Different Data Label y	unsupervised
Different Protocol f	batch
Different Input Space x	Concrete feature: 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

3. (10%) 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。

(依據編寫完整性給分) (10 分)

(定義問題)

定義何謂溫拿(winner)及魯蛇(loser)，例如 60 歲時有房、有車、有妻、有子、存款數目、社會地位等各種條件，每一個條件要確定數值化的方式，例如有房、有車、有妻、有子標為 0 或 1；存款數目為實數；社會地位分十級等。

(資料標記)

定義標記的方法(例如專家學者評斷給分)

(訓練)

1. 收集溫拿(winner)及魯蛇(loser)的資料，~~並且將這些資料分類處理~~
2. 將資料數值化
3. 資料進行前處理(normalization 等)，並將資料切分成訓練集及測試集
4. 決定使用的模型，決定損失函數為 squared loss
5. 整理資料格式並輸入模型訓練
6. 透過交叉驗證來確定模型效果

(測試)

7. 將要被預測的人的資料輸入訓練好的模型做預測
8. 產生結果

4. a) (5%)試解釋 Perceptron Learning Algorithm 中，更新 w 公式的原理為何? b) (5%) 試比較 Perceptron Learning Algorithm 與 Pocket Algorithm 相同與相異之處為何? c) (5%) 試說明 Perceptron Hypothesis 中，若資料維度是 d，為什麼 Perceptron Hypothesis $h(x)$維度是 d+1?

(a) (依據編寫完整性給分) (5分)

利用 W_t 的向量與 X 的向量做內積，且利用 $sign\left(w_t^T x_{n(t)}\right) \neq y_{n(t)}$ 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 W_t 與 X 的向量夾角過大，所以 W_t 要往 X 向量移動，也就是 $W_t + X$ 來修正 W_{t+1} 的向量，反之 W_t 則是遠離 X 向量，$W_t - X$，最後修正到沒有錯誤的點為止。

(b) (依據編寫完整性給分) (各2.5分)

相同：在線性可分中，都可以求到w 的解，使所有資料正確切開，判斷其類別；尋找錯誤樣本，以及更新 w 的方法是相同

相異：在每一次的 iteration 中，Pocket 需要確認 W_{t+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。

(c) (依據編寫完整性給分) (5分)

$h(x) = sign\left(\left(\Sigma_{i=1}^{d} w_i x_i\right) - threshold\right)$

也可以化作為$h(x) = sign\left(\left(\Sigma_{i=1}^{d} w_i x_i\right) + (-threshold) * (+1)\right)$

接著將 $(-threshold)$ 視為 w_0、$(+1)$ 視為 x_0，代回公式後可以得到 $h(x) = sign\left(\Sigma_{i=1}^{d} w_i x_i\right)$，

因為 summation 變成從$\mathrm{i} = 0$ 到$\mathrm{i} = \mathrm{d}$，所以 Perceptron Hypothesis $h(x)$的維度是 $d + 1$

5. a) (5%) 試說明 classification、linear regression、logistic regression 這三種問題的錯誤衡量方式為何? b) (5%) 這三種錯誤衡量的方式，詳細說明讓錯誤最小化的方法是什麼?

(a) (各 2 分，總和最高 5 分)

classification error：分到錯的數量/總數

linear regression error：通常用 L1 norm: $\frac{1}{N}\sum_{n=0}^{N}|h(x_n) - y_n|$或 L2 norm: $\frac{1}{N}\sqrt[2]{\sum_{n=0}^{N}(h(x_n) - y_n)^2}$

logistic regression error：cross entropy:$\ln(1 + \exp(-\mathrm{ywx}))$

(b) (各 2 分，總和最高 5 分)

classification : 因為要計算「分到錯的數量」，屬於不可微分的計算(其中包含 indicator function)，較難解出，只能用近似的方式，如 pocket 的作法。

linear regression :-L2 的損失函數可以直接微分求極值，進而得到解。

logistic regression: cross entropy 無法直接微分求解，以 gradient descent 最小化 cross entropy。

6. a) (5%) 試解釋 Gradient Descent 方法的做法為何? b) (5%) 試解釋 Gradient Descent 方法的公式是怎麼推導而來? c) (5%)試說明 Gradient Descent 與 Stochastic gradient descent (SGD) 方法上的差異，與其各自的優缺點為何。

(a) (依據編寫完整性給分) (5 分)

從初始化的解，尋找其損失函數對於模型參數梯度方向的反方向，希望一步步找到降低損失函數的參數

(b) (依據編寫完整性給分) (5 分)

目的是找 $\min_{\|v\|=1} E_{in}(W_t + \eta v)$

$W_{t+1} = W_t + \eta v$，v 為修正錯誤的方向，η 為一次修正多少，

利用泰勒展開式讓非線性優化能使用線性公式做線性逼近，

故 $E_{in}(W_t + \eta v)$ 經泰勒展開式會近似為 $E_{in}(W_t) + \eta v^T \nabla E_{in}(W_t)$，

且當 η 很小時，$E_{in}(W_t + \eta v) \approx E_{in}(W_t) + \eta v^T \nabla E_{in}(W_t)$

則目標為求 $\min_{\|v\|=1} \left(\underbrace{E_{in}(W_t)}_{known} + \underbrace{\eta}_{given\ positive} v^T \underbrace{\nabla E_{in}(W_t)}_{known} \right)$，

$E_{in}(W_t)$ 與 η 為已知，v 為單位向量，$\nabla E_{in}(W_t)$ 為向量，

故目標是求 $v^T \nabla E_{in}(W_t)$ 的最小值，而已知兩個反方向的向量相乘會得最小值，

故 v 與 $\nabla E_{in}(W_t)$ 應為反方向才能得到最小值，並將 v 設為 $-\frac{\nabla E_{in}(W_t)}{|\nabla E_{in}(W_t)|}$，

更新公式為 $\mathrm{W}_{t+1} \leftarrow W_t - \eta \frac{\nabla E_{in}(W_t)}{|\nabla E_{in}(W_t)|}$

(c) (依據編寫完整性給分) (5 分)

Gradient Descent 為使用全部的資料來取得更新的方向，SGD 則為把資料視為均勻分佈，並從中隨機取一筆資料，來取得更新方向。SGD 相當於 Gradient Descent 再加上以 0 為平均值的 noise，只要經過足夠的步驟，平均真實梯度會近似於平均隨機梯度。

SGD 的優點為計算簡單、效率高，且對於大數據分析或線上學習方面會很有用。缺點則為取得的平均隨機梯度，可能仍與真實平均梯度不一樣，可能會取得不佳的資料而讓結果有偏差。

Gradient Descent 的優點為每次更新都會朝著正確的方向進行，最後能夠保證收斂於極值點。缺點為在於學習時間與 SGD 相比較長。

7. 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) (5%) 試寫出要求出 $y_1,y_2,\ldots,y_5$ 的 regression model 公式解。b) (5%) 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) (5%) 假設 datatset 每一筆資料維度為 10，利用這個 dataset 求出的線性迴歸模型，共會有多少的參數? d) (5%) For the feature X∈R^5, when using the polynomial transform $\Phi(x)=(x_0^4,\ x_1^3,\ x_2^2, x_3^1, x_4^0)$, give the close-form solution of perceptron w using $\{\Phi(x_n), y_n\}$ by non-linear transform of linear regression. Note that you need to provide the details of matrices X and y.

(a) (5 分)

目標: 找到 W_{LIN} 使得 $\frac{2}{N}(X^TXw - X^Ty) = \nabla E_{in}(w) = 0$

- 可逆的(invertible) X^TX
  - $W_{LIN} = (X^TX)^{-1}X^Ty$
  - 通常情況因為 $N \gg d+1$
- 奇異的(singular) X^TX:
  - 多個最佳解
  - 其中一解: $W_{LIN} = X^+y$

(b) (各 1.25 分，總和最高 5 分)

$W_{LIN} = (X^TX)^{-1}X^Ty$

資料維度是 2，再加上常數項維度為 3

X^T: $\mathbf{3 \times 5}$

X: $\mathbf{5 \times 3}$

$(X^TX)^{-1}$: $\mathbf{3 \times 3}$

y: $\mathbf{5 \times 1}$

$w = (X^TX)^{-1}X^Ty$: $\mathbf{3 \times 1}$

(c) (5 分)

x_n 維度:10

$x^Tw + b = y$，共會有 10+1=**11 個參數**

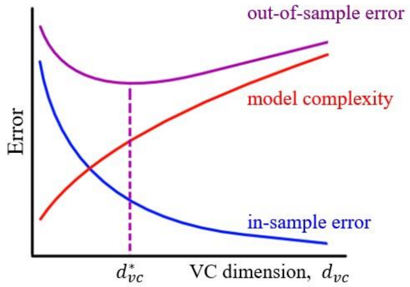
(d) (5 分)

$$\tilde{w}^T\Phi(x) = y = \tilde{w}_0x_0^4 + \tilde{w}_1x_1^3 + \tilde{w}_2x_2^2 + \tilde{w}_3x_3^1 + \tilde{w}_4x_4^0$$

$$x_{\{第幾筆,第幾維\}} \qquad X = \begin{bmatrix} x_{0,0}^4 & x_{0,1}^3 & x_{0,2}^2 & x_{0,3}^1 & x_{0,4}^0 \\ x_{1,0}^4 & x_{1,1}^3 & x_{1,2}^2 & x_{1,3}^1 & x_{1,4}^0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{n,0}^4 & x_{n,1}^3 & x_{n,2}^2 & x_{n,3}^1 & x_{n,4}^0 \end{bmatrix} \qquad Y = \begin{bmatrix} y_0 \\ y_1 \\ \vdots \\ y_n \end{bmatrix}$$

$$W_{LIN} = \left(\begin{bmatrix} x_{0,0}^4 & x_{1,0}^4 & \cdots & x_{n,0}^4 \\ x_{0,1}^3 & x_{1,1}^3 & \cdots & x_{n,1}^3 \\ x_{0,2}^2 & x_{1,2}^2 & \cdots & x_{n,2}^2 \\ x_{0,3}^1 & x_{1,3}^1 & \cdots & x_{n,3}^1 \\ x_{0,4}^0 & x_{1,4}^0 & \cdots & x_{n,4}^0 \end{bmatrix} \begin{bmatrix} x_{0,0}^4 & x_{0,1}^3 & x_{0,2}^2 & x_{0,3}^1 & x_{0,4}^0 \\ x_{1,0}^4 & x_{1,1}^3 & x_{1,2}^2 & x_{1,3}^1 & x_{1,4}^0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{n,0}^4 & x_{n,1}^3 & x_{n,2}^2 & x_{n,3}^1 & x_{n,4}^0 \end{bmatrix} \right)^{-1} \begin{bmatrix} x_{0,0}^4 & x_{1,0}^4 & \cdots & x_{n,0}^4 \\ x_{0,1}^3 & x_{1,1}^3 & \cdots & x_{n,1}^3 \\ x_{0,2}^2 & x_{1,2}^2 & \cdots & x_{n,2}^2 \\ x_{0,3}^1 & x_{1,3}^1 & \cdots & x_{n,3}^1 \\ x_{0,4}^0 & x_{1,4}^0 & \cdots & x_{n,4}^0 \end{bmatrix} \begin{bmatrix} y_0 \\ y_1 \\ \vdots \\ y_n \end{bmatrix}$$

8. (10%)試解釋下圖對於 VC dimension 的增加或減少，如何影響 in-sample error, out-sample error, model complexity 的結果。



(依據編寫完整性給分) (錯一部分扣 3 分)

d_{VC} 愈大，代表模型愈複雜

d_{VC} 愈大，在訓練資料表現愈好，代表 E_{in} 愈小。

d_{VC} 愈大，代表模型愈複雜，模型在提升複雜度時，E_{out} 會隨之降低，並在模型最佳複雜度時，E_{out} 會降到最低點，E_{out} 與 E_{in} 差距最接近。模型在更複雜時，愈容易發生 overfitting，也就代表 E_{out} 會隨之提升， E_{out} 與 E_{in} 差距也愈大。