2023 AIoT InnoWorks 物聯網專題競賽

以 WISE-PaaS 建構隱私保護分散式資料蒐集與分析

隊伍名稱：*Intelligence Net*

指導老師：游家牧 黃俊穎

組員：陳畹潾 周信廷

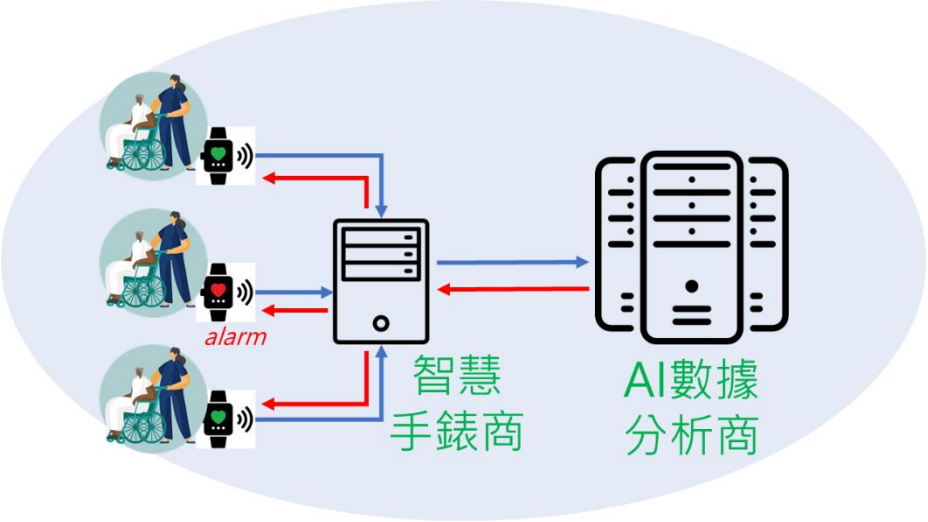
許嘉宜 林志訓

專案介紹

本地化差分隱私 (local differential privacy, LDP) 可以應用於多種智慧城市場景中，以保證在不侵犯個人隱私的情況下幫助政府更好地了解市民的需求和行為。譬如社會安全可以利用 LDP 技術收集市民的安全相關數據，例如犯罪發生地點、頻率、類型等。譬如環境監測則可利用 LDP 蒐集市民的環境監測數據，例如空氣品質、水質、噪音等。本專案著重考慮極端高隱私的用戶資訊；具體來說，假設用戶端具有低算力的微裝置且裝置能即時監測個人健康數據。本專案將使用心臟病患數據偵測的情境，模擬 LDP 蒐集病患的健康相關數據，例如血壓、血糖、心率等，並幫助智慧醫療於即時預警的應用。更甚至，可以結合聯邦學習 (federated learning, FL) 建立 LDP-FL，在具有隱私保護的情況下以不破壞個人隱私為前提建立深度學習模型來發展更好的智慧應用。

專案情境

本專案考慮情境為心臟病患數據偵測的情境，如下圖(一)。此處考慮角色分別有心臟病潛在患者、智慧手錶開發商、AI 數據分析商。假設智慧手錶商分別和 AI 數據分析商具有合作協議，由智慧手錶商提供即時健康醫療偵測技術以隨時監看病患生理狀況並收集相關健康數據，而後才由 AI 數據分析商透過智慧手錶商收集到的健康數據進行分析以利於其他疾病偵測的技術開發。此處，本專案關注的點在於智慧手錶收集相關健康數據所涉及的隱私侵犯議題。眾所皆知，在 AI 世代的趨勢下，科技越發蓬勃發展而有許多數據利用引起的個資侵犯案例，以至於無論是歐盟、美國等皆有嚴格規範個資蒐集與利用的法規。於是，本專案將就歐盟與美國相關法規中的建議，使用本地化差分隱私技術作為病患健康數據收集的隱私保護機制。同時，為促進未來 AI 醫療的發展，本專案還考慮聯邦學習的情境，透過所有配戴智慧手錶的病患所提供的數據，進行健康醫療的分類模型訓練。



圖一、心臟病患數據偵測的情境

方法與架構

此處，會先介紹隱私保護分散式資料蒐集與分析的兩個重要技術，本地化差分隱私與聯邦學習。隨後，會全面性的講解本專案的系統架構與流程，期間還會說明 WISE-PaaS 如何居中扮演協助者的功能。

差分隱私（differential privacy，DP）是一種保護個人資料隱私的方法，通過在資料中加入一定的雜訊，使得資料釋放後不會揭露個人資訊。差分隱私的原理可理解成，若對僅有一筆記錄不同的兩個資料庫和作運算分析，無論兩者差異是添加、刪除或修改該筆紀錄，其透過隨機演算法的分析結果將不會有重大差異，即分析結果的差異是可控的。這也意味著一個具有差分隱私保護的系統或演算法在分析過程中能有效地隱藏個人的參與資訊。換言之，差分隱私提供了一種透過設定隱私損失（ ε ，或稱隱私預算）在數學上可量化的隱私保護框架，可以應用於各種資料分析與資料共享的服務，並確保個人隱私得到適當的保障。

差分隱私的數學定義如下：
令隱私預算 ε 為一正實數，而 A 為一隨機演算法，以一資料庫為該演算法的輸入。令 S 為演算法 A 所映射的空間。若對所有僅有一筆記錄（例如某個人的資料）不同的兩個資料庫 D_1 和 D_2，以及 S 的所有子集 s，符合下列不等式，則稱該演算法 A 可以提供 ε-差分隱私。其中，取機率的隨機性來自於演算法 A。

$$Pr[A(D_1) \in s] \leq exp(\varepsilon) \cdot Pr[A(D_2) \in s]$$

然而，差分隱私技術所考慮的情況為保護資料集的統計資訊，而本專案的情境為即時病患的健康醫療數據，每次的數據收集為單筆資料故須改採本地化差分隱私(local differential privacy, LDP)技術。為符合本地化差分隱私，同一位病患的任何時刻之任意兩筆數據將不考慮之間的關聯性，意即視每筆紀錄本身就是一份資料集，並不需要去計算隱私預算總消耗的問題。舉例來說，在差分隱私機制情況中，隱私預算若為 10，則在 10 步更新的設置下，每一步更新均獲得隱私預算 1 的配置。然而，在本地化差分隱私機制情況中，隱私預算若為 10，則在 10 步更新的設置下，每一步的更新均會以隱私預算 1 去配置。隱私預算越多則添加雜訊越少，所提取的資訊也越接近真實資訊。

聯邦學習（federated learning, FL）是一種具備隱私保護機制的分散式機器學習方法，其主要結構由多個分散式端點和一個中央服務器組成，透過這種方式進行模型訓練。在聯邦學習中，各分散式端點可以使用自身的資料集訓練本地模型，並僅傳送該次模型更新之梯度權重至中央服務器，進行權重的聚合以及全局模型的更新。這樣的設計使得參與者無需傳輸本地端的真實資料，同時仍能參與協作式

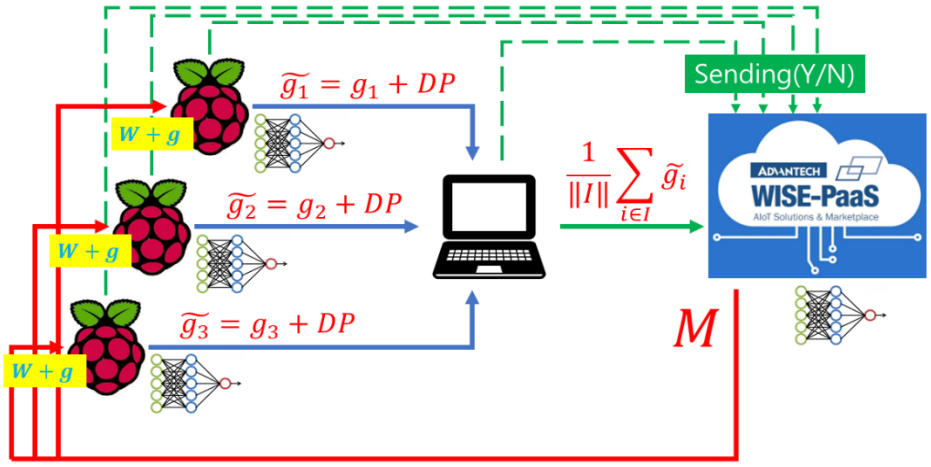
機器學習訓練的過程。

聯邦式機器學習的運作流程如下：

1. 下載全局模型：中央服務器初始化模型和參數，然後讓所有參與者下載並用來進行訓練。
2. 訓練模型：各分散式端點使用本地資料進行訓練，調整模型參數，並產生本地梯度權重。
3. 上傳本地梯度權重：參與者無需上傳原始資料，只需上傳本地梯度權重至中央服務器。
4. 模型聚合：中央服務器將所有參與者上傳的權重參數進行聚合，形成新一輪的全局模型。

透過聯邦學習，系統可以在保護數據隱私的同時，實現協作訓練，並節省傳輸成本。這種方法在多個領域都有應用，例如移動設備上的應用程式、醫療保健領域和物聯網等。然而，儘管聯邦學習具備一定程度的隱私保護，但在近年來的研究報告[1,2,3]指出隱私洩漏的問題依然存在。因此，結合本地化差分隱私技術之聯邦學習就成了隱私保護的關鍵技術。

系統架構為本地化差分隱私與聯邦學習的結合，並在機制的運行中將引入 WISE-PaaS 之廣泛功能以獲得更好的系統運行體驗。本專案將就兩階段進行說明，分別為模型訓練階段以及預警階段。圖(二)展示了模型訓練階段的架構圖，其中樹莓派圖標代表了心臟病潛在患者所配戴的智慧手錶；筆電圖標代表智慧手錶開發商（中端聚合器）；WISE-PaaS 圖標代表的是 AI 數據分析商。首先，在模型正式

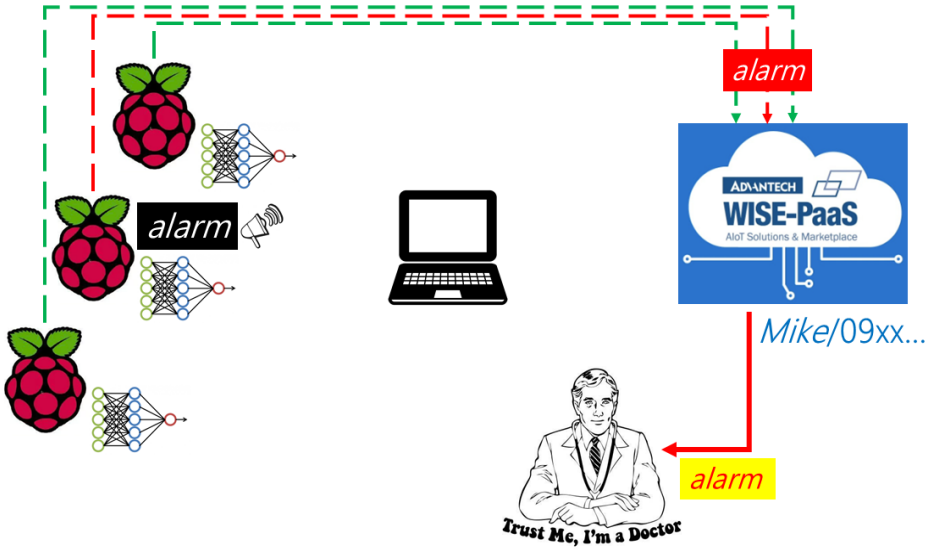


圖二、系統架構/訓練階段

[1] Boenisch, Franziska, et al. "When the curious abandon honesty: Federated learning is not private." arXiv preprint arXiv:2112.02918 (2021).
[2] Geiping, Jonas, et al. "Inverting gradients-how easy is it to break privacy in federated learning?." Advances in Neural Information Processing Systems 33 (2020): 16937-16947.
[3] Zhu, Ligeng, Zhijian Liu, and Song Han. "Deep leakage from gradients." Advances in neural information processing systems 32 (2019).

訓練之前還需前置作業，每台樹莓派以及 WISE-PaaS 上將配置相同權重的初始化模型。隨後，樹莓派在接收到病患健康資訊後會將數據利用事先設置的模型進行訓練，並且模型的正確標籤（label）由病患身旁護理人員協助輸入。模型在計算出更新梯度後便會以本地化差分隱私技術注入相應雜訊並將其傳送至中端聚合器，見圖(二)藍線處上方算式。中端聚合器在確認接受到所有樹莓派傳送數據後便會直接聚合並將其傳送至 WISE-PaaS 上，見圖(二)筆電圖示右方算式。最終，WISE-PaaS 會以此聚合梯度對全局模型進行更新並且回傳全局模型給每台樹莓派，樹莓派再以該全局模型覆蓋自身模型。為了更好地監看每個步驟的數據傳遞，系統中的樹莓派與中端聚合器在每次傳送數據且正確傳送後，都會發出一道訊號至 WISE-PaaS 上，透過 WISE-IoTSuite/SaaS Composer 就可以實現數據流的查看。這對於系統開發者而言，可以更瞭解網路連線與程序運行快慢的狀況。

圖(三)展示了模型訓練階段的架構圖，與圖(二)不同處在於多了醫生的角色在其中。回歸到本專案預期服務對象為配戴智慧手錶之病患，透過訓練階段取得的良好預測效果之模型，一旦病患健康數據被模型判斷為正確(True)則會觸發預警機制。預警機制由 WISE-IoTSuite/Notification 實現，樹莓派會持續監看病患健康數據並將模型預測結果傳送至 WISE-PaaS 上，若為警告訊號則會依照事先設定的對象發送信件提醒。

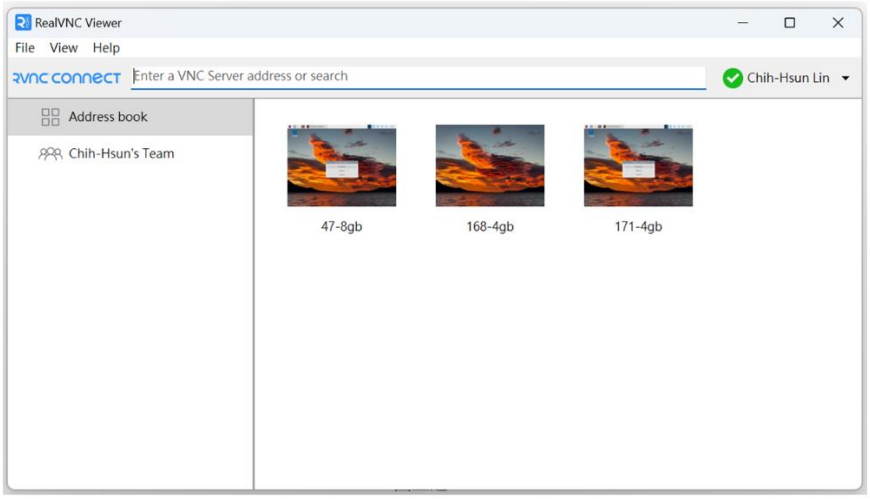


圖三、系統架構/預警階段

校正實驗與結果

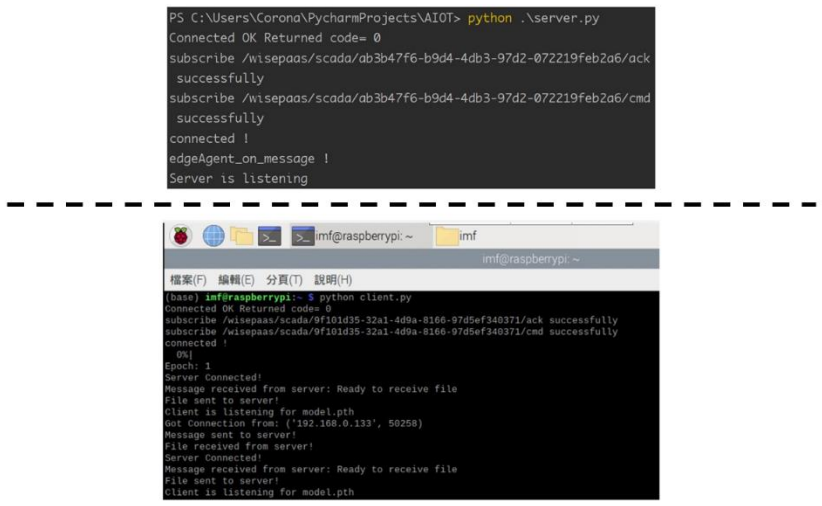
在展示實驗結果之前，本報告將就實際開發環境與實驗參數設定進行介紹，隨後透過圖文導覽流程，並在結尾展示不同參數下的實驗表現。

實驗的樹莓派為 Raspberry Pi 4 Model B，而進行樹莓派與筆電的連接以 RealVNC[4] 運行。透過 RealVNC 可以很輕易完成樹莓派的遠端操控，以及同時監看多台樹莓派的運作情況，如圖四。聯邦學習的訓練會以一般神經網路（Neural Network）為模型架構。本地化差分隱私的機制會使用ℓ_2-norm = 1 作為梯度輸出的權重限制(gradient clipping，梯度剪裁)，並且實驗會分別以隱私預算為 1、2、4、8、*inf* 進行差異比較。訓練資料集的來源為 Kaggle 競賽的 Heart Failure Prediction Dataset[5]，該資料集為二分類目標且資料屬性有 12 項，資料總數為 918 筆。訓練資料集的分配為每台樹莓派配置 270 筆且均不重複，未獲分配的 108 筆則作為測試資料集。



圖四、RealVNC 遠端操控樹莓派

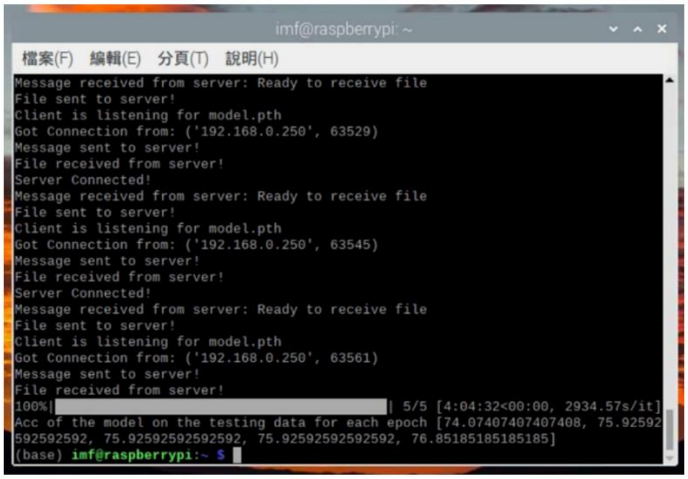
實驗運作初期會先運行筆電的運行檔並且等待樹莓派傳送梯度，隨後樹莓派依序啟動並且回傳等待的訊息，如下圖(五)。此時，就算是完成樹莓派與筆電的對接且開始聯邦學習的運作流程，不斷地計算梯度、聚合梯度、模型更新。並且在訓



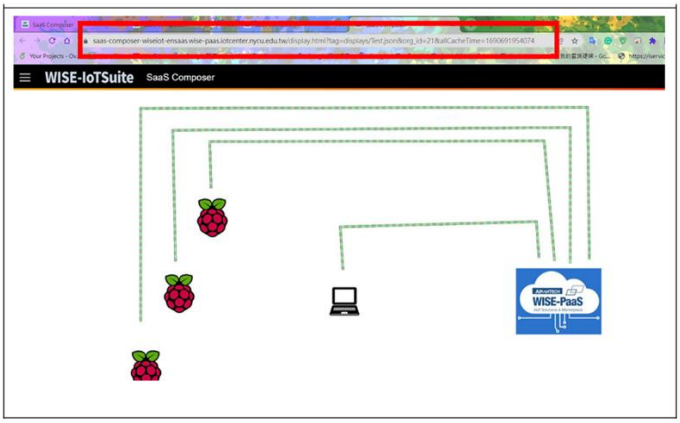
圖五、筆電與樹莓派進行對接

[4] https://www.realvnc.com/en/connect/download/viewer/
[5] https://www.kaggle.com/datasets/fedesoriano/heart-failure-prediction?sort=votes

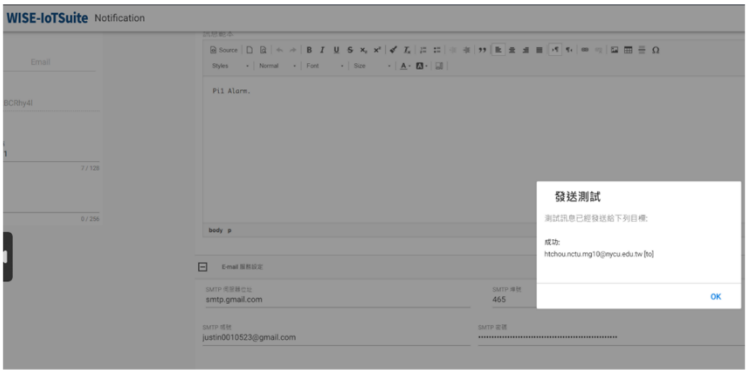
練步數達到所設置的閥值時，就會停止訓練並且顯示模型對於測試資料集的預測準確度，如圖(六)。除了 RealVNC 的介面顯示，在 WISE-IoTSuite/SaaS Composer 也會即時顯示數據傳遞的情況，如圖(七)。同時，對於預警階段的執行，WISE-IoTSuite/Notification 經手動測試也有正確發送信件到信箱，見圖(八)。



圖六、筆電與樹莓派進行模型訓練及預測



圖七、WISE-IoTSuite/SaaS Composer 顯示數據流



圖八、WISE-IoTSuite/Notification 發送預警信件

最終，模型訓練分別以隱私預算為 1、2、4、8、*inf* 進行，並且每隔 270 步更新就會對測試資料進行預測以獲得準確度，圖(九)為實驗結果的表格紀錄，表格橫軸為不同步數而縱軸為不同隱私預算所需注入的雜訊規模。順帶一提地，圖(九)的雜訊規模為透過差分隱私特有計算所獲得，由上而下依序為 $\varepsilon = 1、2、4、8、inf$，且該計算可以透過 Tensorflow 的開源專案[6]在 GitHub 上輕鬆調用。儘管如此，此套件實際套用在檢驗的需求上還是稍有不便之處，例如無法根據給定的隱私預算和查詢次數計算出雜訊位置的雜訊參數（noise scale，σ）。為因應這個情況，本專案修改內部程式，利用二分法迭代找出最佳解，圖(十)顯示部分程式碼。

Acc	Epoch1	Epoch2	Epoch3	Epoch4	Epoch5
STD 4.045385369	47.222	47.222	47.222	48.148	47.222
STD 2.149678469	47.222	47.222	55.555	68.518	62.962
STD 1.157769257	47.222	48.148	67.592	69.444	70.370
STD 0.638086714	47.222	63.888	71.296	77.777	75.0
STD 0	74.074	75.925	75.925	75.925	76.851

圖九、不同隱私預算的實驗準確度表格

```
from _future_ import absolute_import
from _future_ import division
from _future_ import print_function

from absl import app
from absl import flags

from tensorflow_privacy.privacy.analysis.compute_dp_sgd_privacy_lib import compute_dp_

FLAGS = flags.FLAGS

flags.DEFINE_integer('N', None, 'Total number of examples')
flags.DEFINE_integer('batch_size', None, 'Batch size')
flags.DEFINE_integer('epochs', None, 'Number of iterations')
flags.DEFINE_float('eps', 1.0, 'Target epsilon')

# python compute_dp_sgd_privacy.py --N=60000 --batch_size=64 --epochs=10 ---

def main(argv):
    noise_multiplier1 = 0.00001
    thread = 1.e-9
    delta = 1.e-5

    assert FLAGS.N is not None, 'Flag N is missing.'
    assert FLAGS.batch_size is not None, 'Flag batch_size is missing.'
    assert FLAGS.epochs is not None, 'Flag epochs is missing.'

        current_eps, _ = compute_dp_sgd_privacy(FLAGS.N, FLAGS.batch_size, n
        print("_:", current_eps)
        while current_eps > FLAGS.eps:
                noise_multiplier1 += 1
                current_eps, _ = compute_dp_sgd_privacy(FLAGS.N, FLAGS.batch
```

```
        c1 = False
        c2 = True
        noise_multiplier2 = noise_multiplier1 - 1
        while c1 or c2:
                last_eps = current_eps
                noise_multiplier = (noise_multiplier1 + noise_multiplier2)/2
                current_eps, _ = compute_dp_sgd_privacy(FLAGS.N, FLAGS.batch_size, noi
                c1 = current_eps > FLAGS.eps
                if c1:
                        noise_multiplier2 = noise_multiplier
                else:
                        noise_multiplier1 = noise_multiplier
                c2 = abs(current_eps-FLAGS.eps)>thread

        print("eps:", current_eps)
        print("noise_multiplier:", noise_multiplier)

if __name__ == '__main__':
        app.run(main)
```

圖十、差分隱私雜訊規模的快速計算

實驗的討論對於本專案的改進有很大的幫助。對於 Kaggle 競賽的 Heart Failure Prediction Dataset 資料集的前處理，我們會使用 MinMax 方法將數值型屬性數值正規化至 0-1 範圍，而並非使用常見的標準化(standardization)方法，此一目的在於實際情況下，每位病患都會有其數據的分布且會與其他病患有獨立關係，故不

[6] https://github.com/tensorflow/privacy

應利用給定標準差與均值統一對所有病患的數據進行標準化轉換。另一方面，現實情境中病患僅會即時回傳數據，但數據本身並不可能涉及未來，故病患本身數據若想用自身分布去做標準化也不合邏輯。

從圖(九)的結果表現，同一時刻下可以看出，當隱私預算越多時雜訊越少且準確度越高；同一隱私預算可以看出，準確度幾乎都會隨時間穩健上升。事實上，梯度權重在梯度剪裁後，權重值相比於注入雜訊顯得非常小，但由於雜訊是根據常態分布去添加，故大多時候雜訊的均值會貼近 0。一旦連接的樹莓派越多則模型訓練的準確度會越高，這可以從 N 台樹莓派的梯度聚合算式$\frac{1}{N}\sum_{i=1}^{N} g_i + \frac{1}{N}\sum_{i=1}^{N} \rho_i$中的後項注入雜訊來推論。

總結

本專案開發之 LDP-FL 系統提供以 WISE-PaaS 建構隱私保護分散式資料蒐集與分析之功能。儘管預期利用 WISE-PaaS 實現梯度上傳功能的問題以及 WISE-IoTSuite/Notification 無法自動預警仍待處理，並且聯邦學習的實現也存在著模型訓練過慢的情況，但作為以差分隱私達到隱私保護為核心技術，系統可以確實幫助用戶共享資料於協作式訓練。本專案未來將就上述問題進行處理與改善，同時找尋新的資料源以貼近實際用戶情況。