**人工智慧導論期末報告**

用機器學習實現中風預測

國立金門大學

資訊工程學系

組長：資工三 李易

組員：資工三 陳先正

組員：資工三 李元樺

指導教授：李錫捷 教授

目錄

1. 摘要
2. 研究動機與研究問題
   1. 研究動機
   2. 研究問題
   3. 研究目的
3. 文獻探討與回顧
4. 研究方法與步驟
   1. 研究流程
   2. 模型
      1. SVM
      2. KNN
      3. Logistic Regression
      4. Deep Learning
5. 預期結果
6. 參考文獻

**(一)摘要**

本次研究使用機器學習演算法預測中風，追求85%以上的預測準確率。透過深入研究動機、問題、目的及文獻探討，我們將致力於構建一個卓越的中風預測模型。該模型的應用不僅用於早期的中風診斷，更可降低風險，對提升患者生活品質具有重要意義。透過預測，我們期望實現更迅速、精準的干預，從而改善患者的照護方式，為醫療領域帶來實質進展。

**(二)研究動機與研究問題**

**研究動機：**

中風是全球健康問題，及早的中風預測可以為患者提供更有效的醫療干預，減少中風的發生。然而，傳統的中風風險評估方法存在著一定的限制，無法全面而準確地預測中風的風險。因此，我們決定透過機器學習算法，來更有效地分析大量的臨床數據和生理參數，從而發現潛在的中風風險因素，建立更準確的預測模型。

**研究問題：**

1. 如何透過機器學習演算法提高中風的預測準確率至少達到85%？

2. 哪些因素對中風的預測有最大的影響力？

3. 選擇哪一種機器學習模型以達到高效能的中風預測？

4. 能否透過非傳統的深度學習技術達到更好的效果?

**研究目的：**

透過解決上述問題，我們的目標是建立一個高準確率的中風預測模型，為患者提供更好的醫療服務和生活品質。透過深入研究中風的相關因素和風險指標，我們希望能夠發現新的預測因子，同時優化現有的預測模型，為臨床醫生提供一個有力的工具，協助他們及早辨識高風險群體，實現個體化的預防和治療策略，進而降低中風的發生率。

**(三)文獻探討與回顧**

根據世界衛生組織 (WHO) 的數據，中風是全球第二大死因，約佔總死亡人數的 11%。中風的危險因子包括高齡、肥胖、高血壓、曾有中風病史或短暫性腦缺血發作、糖尿病、肥胖、膽固醇過高、抽菸、心房顫動等。

一般而言中風的診斷需要配合醫學影像檢查，例如斷層掃描或是核磁共振攝影，並配合理學檢查其他檢驗－例如心電圖和血液檢查－則有助於發現導致中風的危險因子並排除其他可能的原因。然而由於資料集的限制，我們現階段只能進行數值上的分析。

先前的研究也主要集中在使用機器學習演算法來進行中風預測，尤其是決策樹、隨機森林和邏輯迴歸等模型在醫療領域的應用。我們將借鑒這些研究經驗，以提高我們的中風預測模型的性能。以下便是我們預計使用的演算法：

**SVM：**

在機器學習中，支援向量機是在分類與迴歸分析中分析資料的監督式學習模型與相關的學習演算法。給定一組訓練實例，每個訓練實例被標記為屬於兩個類別中的一個或另一個，SVM訓練演算法建立一個將新的實例分配給兩個類別之一的模型，使其成為非概率二元線性分類器。SVM模型是將實例表示為空間中的點，這樣對映就使得單獨類別的實例被儘可能寬的明顯的間隔分開。然後，將新的實例對映到同一空間，並基於它們落在間隔的哪一側來預測所屬類別。

**KNN：**

在圖型識別領域中，最近鄰居法（KNN演算法，又譯K-近鄰演算法）是一種用於分類和迴歸的無母數統計方法，由美國統計學家伊芙琳·費克斯和小約瑟夫·霍奇斯於1951年首次提出，後來由托馬斯·寇弗擴充。在這兩種情況下，輸入包含特徵空間中的k個最接近的訓練樣本。

* 在k-NN分類中，輸出是一個分類族群。一個物件的分類是由其鄰居的「多數表決」確定的，k個最近鄰居（k為正整數，通常較小）中最常見的分類決定了賦予該物件的類別。若k = 1，則該物件的類別直接由最近的一個節點賦予。
* 在k-NN迴歸中，輸出是該物件的屬性值。該值是其k個最近鄰居的值的平均值。

**Logistic Regression：**

在統計學中，邏輯斯迴歸是一種對數機率模型（Logit model），是離散選擇法模型之一，屬於多變量分析範疇，是社會學、生物統計學、臨床、數量心理學、計量經濟學、市場行銷等統計實證分析的常用方法。

在二元邏輯迴歸中，有一個二元應變數，由指示變量編碼，其中兩個值標記為「0」和「1」，標記為「1」的值的相應機率可以在0和1之間變化；將在這變化中的對數發生率轉換為機率的函數就是邏輯斯函數。

**Deep Learning:**

深度學習（deep learning）是機器學習的分支，是一種以人工神經網路為架構，對資料進行表徵學習的演算法。深度學習中的形容詞「深度」是指在網路中使用多層。早期的工作表明，線性感知器不能成為通用分類器，但具有非多項式啟用功能和一個無限寬度隱藏層的網路可以成為通用分類器。

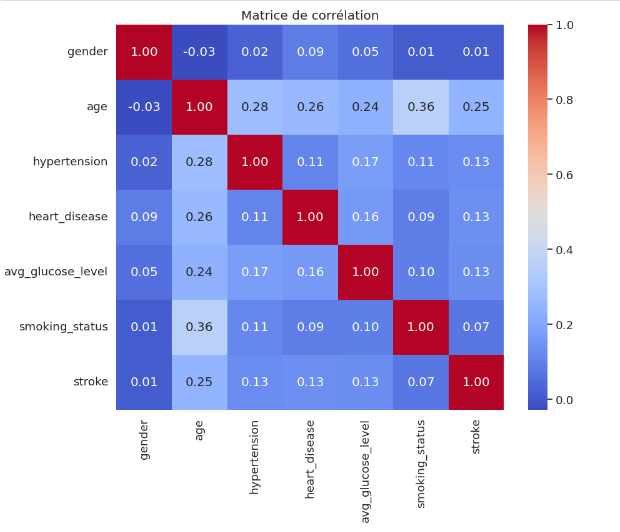
深度學習是機器學習中一種基於對資料進行表徵學習的演算法。觀測值（例如一幅圖像）可以使用多種方式來表示，如每個像素強度值的向量，或者更抽象地表示成一系列邊、特定形狀的區域等。而使用某些特定的表示方法更容易從實例中學習任務（例如，臉部辨識或面部表情辨識）。深度學習的好處是用非監督式或半監督式的特徵學習和分層特徵提取高效演算法來替代手工取得特徵。

**(四)研究方法與步驟**

為了能更有效率的完成任務，我們決定使用Python，並搭配現有的機器學習工具來開發模型。我們使用pip指令安裝了各種相關的套件，包括但不限於pandas、numpy、matplotlib、scikit-learn和TensorFlow。以下便是我們實際的研究流程：

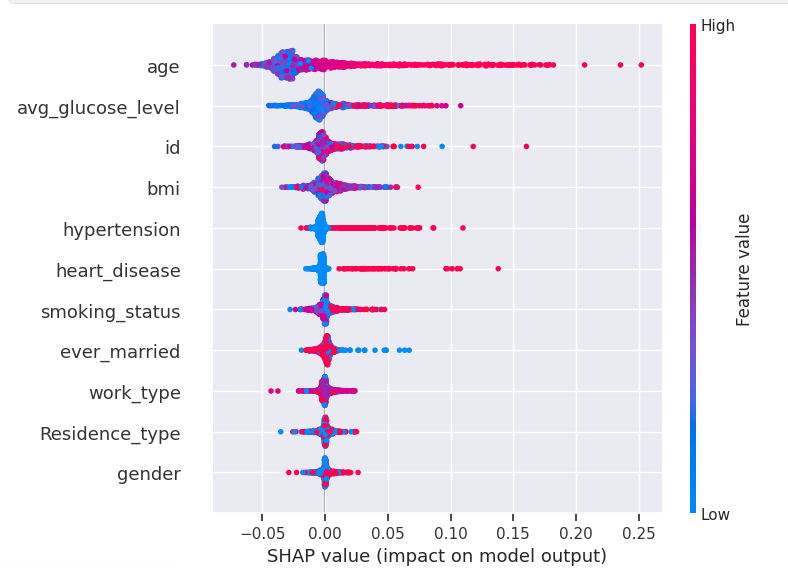
1. 資料收集：使用 Kaggle 的 Stroke Prediction Dataset 作為資料來源。

2. 資料分析：用混淆矩陣以視覺化的方式呈現數據框中各列之間的相關性，這對於理解數據中變數之間的關係和趨勢很有幫助。



圖(1)

除此之外，我們也使用SHAP來顯示每個特徵如何影響模型對個別樣本的預測。SHAP 值用於將預測的貢獻分配給每個特徵，SHAP 值的絕對值越大，表示該特徵對於預測的影響越大。



圖(2)

3. 資料預處理：包括處理缺失值、標準化資料、異常值處理等步驟。在此我們還使用SMOTE來解決數據不平衡的情況：圖(3)是我們處理後的結果，一開始先將資料集切割為特徵和目標兩個部分，切割資料集為訓練集和測試集，使用SMOTE進行過抽樣，將平衡後的資料重新組合成資料集，再檢查新的資料集，最後結果成功將有中風(1)與沒有中風(0)的比例調整到相同數值。



圖(3)

4. 特徵選擇：透過相關性分析和特徵重要性評估來確定關鍵特徵。

5. 演算法選擇：試試決策樹、隨機森林、邏輯迴歸等機器學習演算法。

6. 模型訓練：使用訓練集訓練模型，並用cross-validation加強驗證。

7. 模型評估：在測試集上評估模型性能，追求至少85%的準確率。

8. 參數調整：對效能不佳的模型進行參數調整，或是更改演算法。

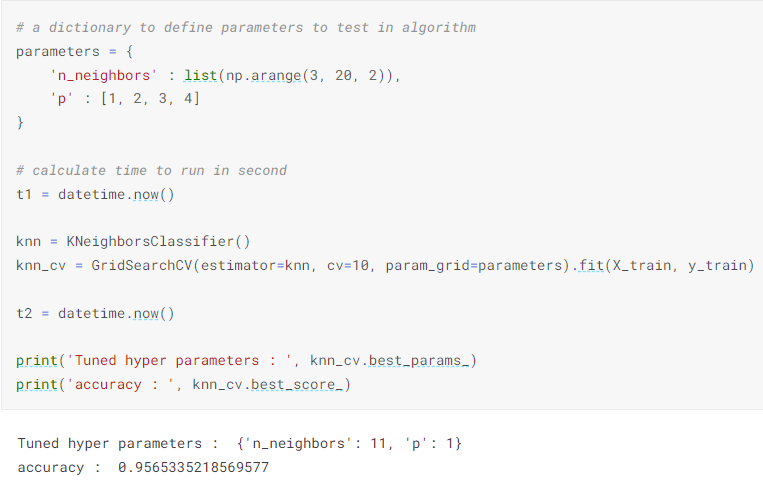
9. 結果分析：分析模型的預測結果，確定關鍵特徵對於預測的貢獻。

10.重複確認：不斷重複上述步驟，以求能達到最好的結果。

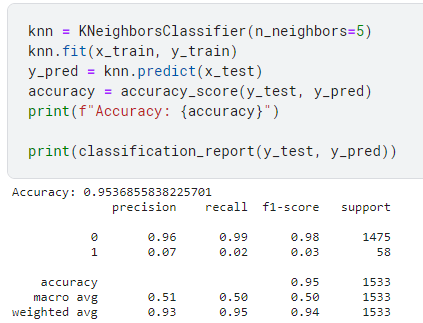
以下便是我們在本次研究中使用到的模型：

KNN模型:

圖(4)是別人使用GridSearchCV工具來自動搜尋最佳參數組合預測KNN模型的結果，準確率為95%。而圖(5)是我們測試原生KNN模型的結果，圖(6)我們也是與圖(4)一樣使用GridSearchCV工具來自動搜尋最佳參數組合，採用了n\_neighbors、weights、metric等3個參數來進行測試，最終我們測試完的結果沒有使用預設的KNN模型還要好，準確率與別人對比的話都是一樣高，有95%。



圖(4)



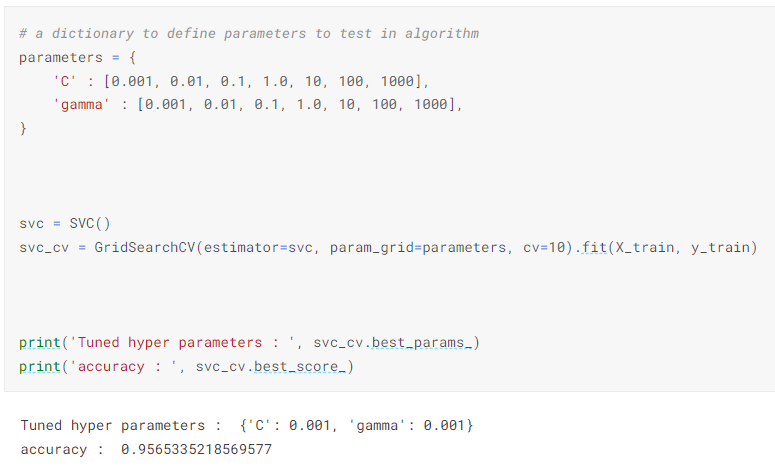
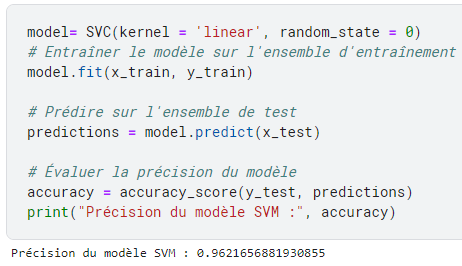
圖(5)



圖(6)

SVM模型:

圖(7)是別人使用GridSearchCV工具來自動搜尋最佳參數組合預測SVM模型的結果，準確率為95%，圖(8)是我們使用Support Vector Machine（SVM）演算法來建立一個二元分類模型，操作流程：初始化 SVM 模型-->使用fit方法將模型擬合到訓練數據集上-->進行預測-->評估模型準確度，我們得出的結果為96%，最後結果與別人對比的話準確率是我們比較高。

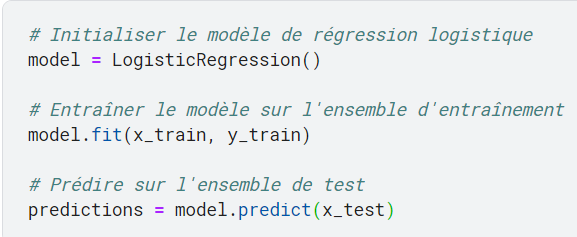
圖(7)圖(8)

## Regression Logistique模型:

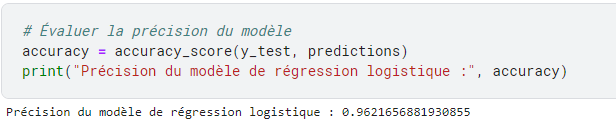
圖(9)是別人使用GridSearchCV工具來自動搜尋最佳參數組合預測Regression Logistique模型的結果，準確率為76%，圖(10)、圖(11)是我們使用了Logistic Regression（邏輯回歸）模型去做二元分類，操作流程：先初始化模型-->使用fit方法將模型擬合到訓練數據集上-->進行預測-->評估模型準確度，操作流程的部分與SVM模型評估是一樣的，我們得出的結果為96%，最後結果與別人對比的話準確率是我們比較高，且高了20%。



圖(9)



圖(10)

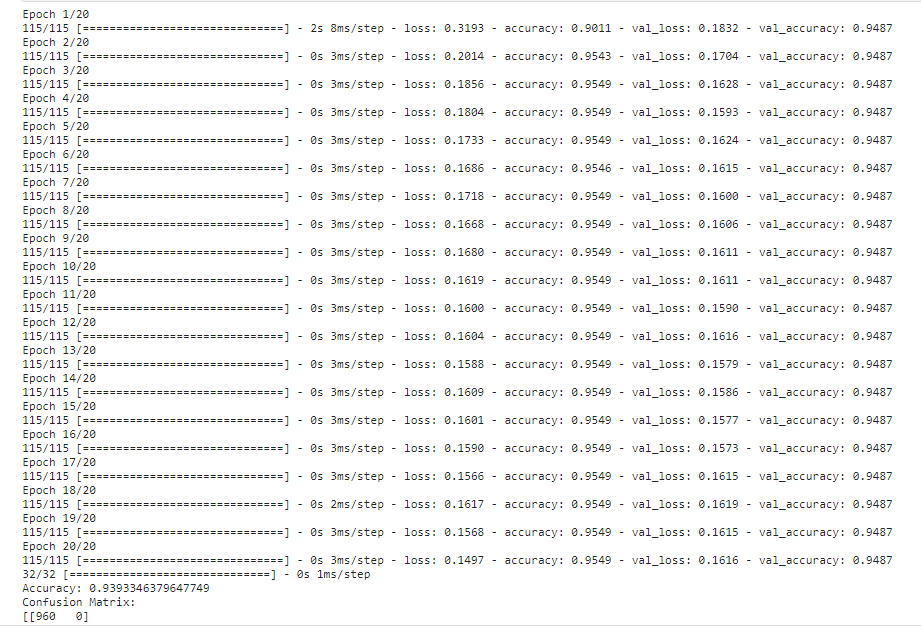


圖(11)

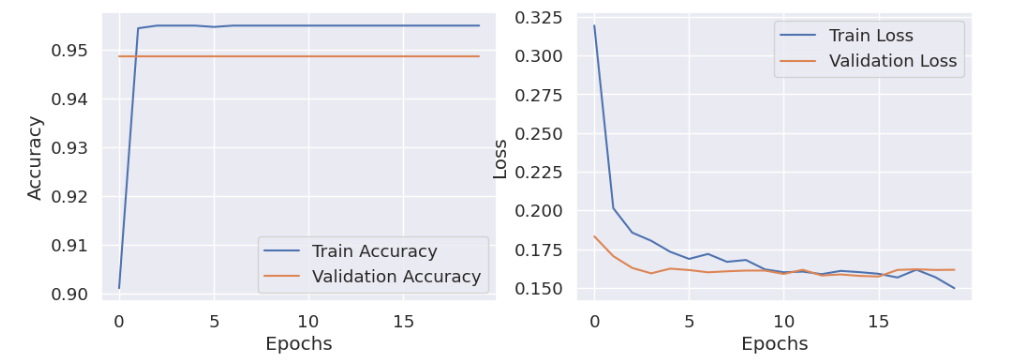
Deep Learning with TensorFlow:

使用來建立一個基於神經網路的二元分類模型，我們採用的TensorFlow，測試流程:數據處理去對類別變數進行編碼-->將缺失值填充為平均值-->將資料拆分為特徵和標籤-->標準化特徵-->切割訓練集和測試集-->建立模型-->編譯模型--> 訓練模型-->評估模型-->最後 繪製訓練過程的準確度和損失曲線，在準確度曲線部分，左邊的子圖顯示了隨著訓練進行的 epoch 次數，模型在訓練集和驗證集上的準確度變化。訓練集準確度由藍色線表示，而驗證集準確度由橙色線表示。隨著 epoch 次數增加，這些曲線可能呈現上升或下降的趨勢，有助於觀察模型是否遭遇過度擬合或過度擬合等問題。在損失曲線的部分：圖(13)顯示了訓練集和驗證集的損失值變化。訓練集損失由藍色線表示，而驗證集損失由橙色線表示。

損失曲線反映了模型在訓練過程中的效能，目標是使損失值盡量趨近於最小值。如果訓練集和驗證集的損失曲線都在逐漸下降，表示模型在學習過程中是有效的，最後呈現出的結果也是不錯，準確率高達93%，損失率的部分也是控制在還不錯的數值。



圖(12)



圖(13)

**(五)預期結果**

通過建構和精進模型的過程，我們預期在中風預測方面取得優異的成果，達到至少85%的預測準確率。我們的研究著重在關鍵特徵的挖掘以及模型性能的提升，為中風預測領域注入新的研究見解。這包括深入探討各項特徵對預測準確性的影響，並透過模型的優化，使其更有效地捕捉和利用這些特徵的資訊。預期結果不僅將提供高水平的預測精度，還將對中風預測的方法論和技術提供有價值的貢獻，推動這一領域的進步與創新。

**(六)參考文獻**

1. <https://franky07724-57962.medium.com/c47247d95dc8>
2. [Stroke Prediction Dataset (kaggle.com)](https://www.kaggle.com/datasets/fedesoriano/stroke-prediction-dataset)
3. <https://scikit-learn.org/stable/user_guide.html>
4. <https://www.tensorflow.org/tutorials>
5. <https://en.wikipedia.org/wiki/Stroke>
6. <https://en.wikipedia.org/wiki/Support_vector_machine>
7. <https://en.wikipedia.org/wiki/K-nearest_neighbors_algorithm>
8. <https://en.wikipedia.org/wiki/Logistic_regression>
9. <https://en.wikipedia.org/wiki/Deep_learning>
10. https://www.kaggle.com/code/nimapourmoradi/healthcare-stroke