Homework 2 Report - Income Prediction

學號：b050902043 系級：資工二 姓名：劉鴻慶

1. (1%) 請比較你實作的 generative model、logistic regression 的準確率，何者較佳？

	Generative model	Logistic regression model
Kaggle score	0.84557 / 0.84203	0.85036 / 0.84227

上表直接拿 Kaggle 的分數當作判斷準確率的依據，兩個 model 都是使用 all feature 且有作 normalization 的 training set，可以發現 Logistic regression model 的準確率較高，由於使用的 generative model 是在 Gaussian Naïve Bayes 的假設下，其表現並不如完全沒有假設分佈直接找 weight 和 bias 的 logistic regression model，可以推測造成差異的原因可能是不是所有 feature 都是 Gaussian 分佈所造成的影響。也有論文顯示通常 logistic 的表現在分類上較 generative 的表現要好。

2. (1%) 請說明你實作的 best model，其訓練方式和準確率為何？

我這次 best model 所使用的方法是 Boosting tree Method，是一種以 decision tree 當做基函數，後面的樹修正前一棵樹的誤差，一直遞迴並 fit 殘差的加法模型

Data Preprocessing : 使用助教提供的 provide train feature 並作 feature normalization

Training Method : 利用 scikit-learn 的 decision tree regressor 實作 Boosting Tree ，decision tree regressor 的 depth = 3，iteration = 100，每次 iteration 去 fit 目前 model 和 label 的差，然後將 model 加上這次 iteration fit 的決策樹

Ref: https://en.wikipedia.org/wiki/Gradient_boosting
Ref: https://blog.csdn.net/shine19930820/article/details/65633436

準確率 Kaggle score：0.87235 / 0.87016 (public / private)

3. (1%) 請實作輸入特徵標準化(feature normalization)，並討論其對於你的模型準確率的影響。(有關 normalization 請參考：https://goo.gl/XBM3aE)

	沒有做 normalization	有做 normalization
Logistic regression	0.74791/0.74216	0.85036/0.84227
Generative model	0.84557/0.84203	0.84557/0.84203

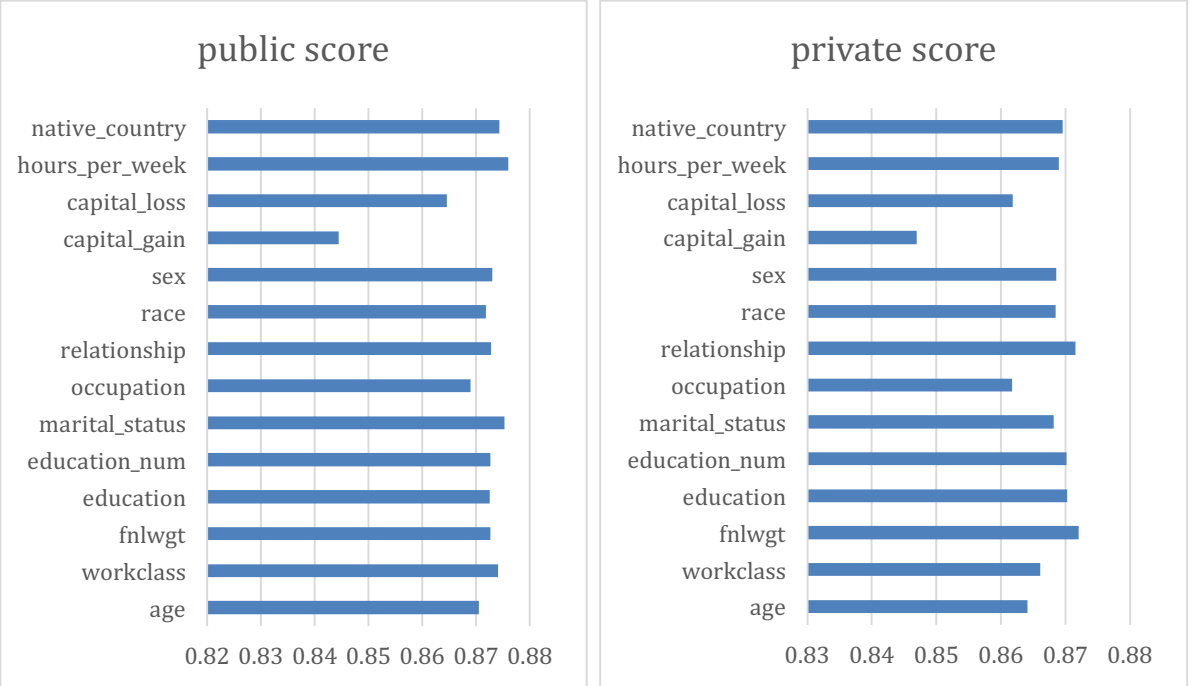
上表為對兩個 model 做 normalization 後的結果，可以發現，logistic regression model 在做 normalization 之後的表現較之前的好很多，而且比較不會產生發散導致 overflow 的狀況。而 generative model 做 normalization 前後並沒有差異。

4. (1%) 請實作 logistic regression 的正規化(regularization)，並討論其對於你的模型準確率的影響。(有關 regularization 請參考：https://goo.gl/SSWGhf P.35)

λ	1	0.1	0.01	0
Kaggle score	0.80674/0.80481	0.82285/0.81832	0.82260/0.84227	0.85036/0.84227

由上表可知正規化對這次的 logistic model 並沒有比較好的結果，越大的 λ 對 kaggle 分數並沒有幫助，反而 λ = 0 的表現最好。由於這次的 model 選用的是線性模型，所以比較不會產生 overfitting 的問題，把沒有做正規化的 weight 拿出來看也沒有出現極大的值。若是選用並增加 λ，penalty 反而限制了某些 weight 值的選用，造成準確率下降。

5. (1%) 請討論你認為哪個 attribute 對結果影響最大？



我用的方法是將每個 attribute 分別刪除後 train 出來的 kaggle score 為依據，如果相對越低分就代表刪除的那個 attribute 影響很大，由上表可知 capital_gain 在 private 和 public 都是很明顯的最低分，我認為 capital_gain 影響最大。