學號：B07901069 系級：電機一 姓名：劉奇聖

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

   答:

Kaggle 分數	generative	logistic
Public	0.84643	0.85171
Private	0.84117	0.85345

   由上表可看出 logistic model 的表現較 generative model 佳，推測原因是 generative model 對 function set 作了限制，而 logistic model 則是在比較 general 的 function set 中尋找最好的 function，故 logistic model 的表現較佳。(logistic model 使用了 adagrad，iteration 為 10000 次，learning rate = 1。此外兩種 model 都有做特徵標準化)

2. 請說明你實作的 best model，其訓練方式和準確率為何？

   答:

   我的 best model 使用 sklearn 的 GradientBoostingClassifier，參數方面 n_estimators 設為 1000，max_depth 設為 2，此外我有對 training data 作特徵標準化的動作。

   準確率的部分，在 kaggle 上的 public score 為 0.87579，private score 為 0.87630。

3. 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

   答:

   Generative:

Kaggle 分數	無特徵標準化	有特徵標準化
Public	0.84631	0.84643
Private	0.84105	0.84117

   Logistic:

Kaggle 分數	無特徵標準化	有特徵標準化
Public	0.80626	0.85171
Private	0.80604	0.85345

   由上面兩個表格可看出特徵標準化對 generative model 幾乎沒有影響，但對 logistic model 卻有非常顯著的影響，推測原因是 generative model 並不是採用 gradient

descent 的方式實作的，故特徵標準化對其無太大的影響，但 logistic model 則是 gradient descent，受到特徵標準化的影響便會非常顯著。(上面的 logistic model 在無特徵標準化時 learning rate = 0.0006，有特徵標準化時 learning rate = 1，同樣都使用了 adagrad，iteration 皆為 10000 次)

4. 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。

答:

Kaggle 分數	無正規化	$\lambda = 1$	$\lambda = 10$	$\lambda = 100$
Public	0.85171	0.85171	0.85196	0.85257
Private	0.85345	0.85333	0.85370	0.85210

上表是實做 logistic regression 正規化(L2-norm)的結果，有使用 adagrad，iteration 10000 次，有特徵標準化。大致上實作正規化對準確率並無太大的影響，雖然如此，在$\lambda = 10$ 時在 public 和 private 的準確率都有提升。推測影響不太顯著的原因是因為我原本的 model 就沒有太嚴重的 overfit，因此正規化便對我的模型無太大的幫助。

5. 請討論你認為哪個 attribute 對結果影響最大？

答:

	原本	age	fnlwgt	sex	capital-gain	capital-loss	hours_per_week
public	0.87579	0.82665	0.86695	0.86683	0.85171	0.86805	0.84324
private	0.87630	0.82213	0.86193	0.86893	0.85284	0.86733	0.84375

	workclass	education	marital-status	occupation	relationship	race	native-country
public	0.87432	0.86105	0.83329	0.86572	0.87186	0.87395	0.87432
private	0.87298	0.86242	0.82950	0.86672	0.87262	0.87605	0.87323

上表是將 testing data 的各個 attribute 分次一一設為 0，丟進我的 best model 後預測並上傳 kaggle 後的分數。因為有做特徵標準化使得平均為 0 標準差為 1，因此將某個 attribute 全設為 0 等同於將那個 attribute 全設為平均值。

觀察此表格可發現將 age 全設為 0 後上傳 kaggle 的分數下降最多，因此我認為 age 這個 attribute 對結果的影響最大。