學號：R04945022 系級： 生醫電資碩二 姓名：張君澤

1.請說明你實作的 generative model，其訓練方式和準確率為何？

答： probabilistic generative model 的公式為：

$$z = \underbrace{(\mu^1 - \mu^2)^T \Sigma^{-1}}_{\boldsymbol{w^T}} x - \underbrace{\frac{1}{2}(\mu^1)^T \Sigma^{-1} \mu^1 + \frac{1}{2}(\mu^2)^T \Sigma^{-1} \mu^2 + ln\frac{N_1}{N_2}}_{\mathrm{b}}$$

因此，把 training data 讀進來並且去除第一行之後，就分別計算兩個 class 的 covariance 還有每一個 feature 的 mu 值，而最後 covariance 為兩個 class covariance 的加權平均。整理完畢之後就可以把 w.dot(x)-b 丟入 sigmoid function, 並且設定在 sigmoidf(z) > 0.45 時得到 output 為 1 否則為零。

2.請說明你實作的 discriminative model，其訓練方式和準確率為何？

我的 logistic regression 的實作方法為： 將 X_train 讀進來之後刪掉第一行都是英文字的 row，再將 Y_train 讀進來，將這兩的 matrix 執行 concat 之後再發現了 age, capital_gain, capital_loss 這些 feature 跟結果沒有太大的正相關，因此畫圖可以發現些蹊蹺，之後我調整了這些 feature 的一些參數（在本 report 第五題會提到，沒有太多篇幅可以詳細說明），再將這些調整過的 feature 執行 normalization，並且以 gradient descent 的方式來更動 x_train.dot(weight) 相對於 y_train 的值。實驗最佳的結果是 learning rate 0.038, iteration 8000 次。

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： 我的 feature normalization 方式如下

```
def featurescale(x):
    import numpy as np
    x = np.matrix(x)
    meanx = np.zeros(x.shape[1])
    stdx = np.zeros(x.shape[1])
    meanx = np.mean(x, axis=0)
    stdx = np.std(x, axis=0)
    normedx = (x - meanx)/stdx
    return normedx
```

本次的 training set 在 generative model 中不進行 feature normalization 可以獲得 0.83157 分(public)，進行 londa 為 0.1 的 regularization 後可以得到 0.84201 即可通過 public simple baseline。 在 logistic regression 中是否進行 normalization 並沒有太大的影響，因為只要 iteration 多次一點 weight 都可以依據 feature 的影響而達到收斂。在 logistic model 中，normalization 的好處是 iteration 可以少一點就快速收斂，我自己是在 iterate 8000 次左右 weight 就可以收斂了。

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： 本次作業中 regularization 執行方式如下：

```
def gradientdescent(x, y, weight, l_rate, iters, londa):
    x = np.matrix(x)
    y = np.matrix(y)
    weight = np.matrix(weight)
    m = x.shape[0]
    ada = np.zeros(weight.shape)
    for i in range(iters):
        temp = x.T * (sigmoidf(x.dot(weight))-y)
        temp += londa * w
        temp = (l_rate/m)*temp
        ada += np.square(temp)
        temp = temp/np.sqrt(ada)
        weight -= temp
    return weight
```

在 logistic regression 的 model 中 regularization 並沒有太大的幫助，londa 設 0.1 , 0.5, 1.0, 5.0 結果皆大同小異。 並且 londa 在超過 10 之後判斷準確度明顯下降。 因此最後 hw2_best.py 裡頭 londa 只設在 0.1。

5.請討論你認為哪個 attribute 對結果影響最大？

在我尚未對 feature 做任何調整之前，觀察整體 feature 和 y_train 的相關係數，發現是否有結婚(married-civ-spouse) 對年薪是否超過五萬美金相關度最高。

在我調整過 capital_gain 和 capital_loss 的 feature 分佈之後，我認為最重要的 feature 為 capital_gain，即投資收益。由下圖可以發現投資獲利在一萬以下的人、以及三萬到四萬二之間的大多數人年薪不會超過五萬美金，至於詳細原因可能是因為投資獲利在三到四萬二之間的人可能是專業銀行理專，可能這些就是它僅有的收入？！

```
plt.plot(x_train[:,3], y_train[:,0], 'ro')
[<matplotlib.lines.Line2D at 0x9cd9828>]
```