學號：B06902006 系級：資工三 姓名：王俊翔

1. (2%) 請比較實作的 generative model 及 logistic regression 的準確率，何者較佳？請解釋為何有這種情況？

Logistic regression	最高到 0.8875391705069123 (在 cross validation 平均)
Generative model	0.8710373046299027

在 Generative model 中，由於我們侷限在一個假設的機率模型中 (例如 Gaussian)並計算其最優的結果，所以在給定的模型中，可以找到一個最佳的結果，且無法再被優化；反之在 Logistic regression 中，分類的方式並沒有一個基礎的 ”臆測”，可以說是考慮較 general 的清況，在 data 夠多的情況往往單憑數據就足以做出一個不錯的邊界預測，在作業中總共有 50000 多筆的 data，所以使用 logistic 能做到較好的準確率。

2. (2%) 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。接著嘗試對正規項使用不同的權重 (lambda)，並討論其影響。

實作:

```
def _cross_entropy_loss(y_pred, Y_label, w, lamb = 1):
    # This function computes the cross entropy.
    #
    # Arguements:
    #     y_pred: probabilistic predictions, float vector
    #     Y_label: ground truth labels, bool vector
    # Output:
    #     cross entropy, scalar
    cross_entropy = -np.dot(Y_label, np.log(y_pred)) - np.dot((1 - Y_label), np.log(1 - y_pred)) + lamb * (np.sum(w))**2
    return cross_entropy
```

```
w = w - learning_rate/np.sqrt(step) * (w_grad + lamb * np.sum(w))
b = b - learning_rate/np.sqrt(step) * b_grad
```

(np.sum(w))**2 應改成 (np.sum(np.power(w,2)))

Lambda	Accuracy	Entropy Loss
0	0.8785477331367489	0.2849275716471048
0.1	0.8785477331367489	0.28492381277220064

0.5	0.8778105418356064	0.2979190579340101
1	0.8783634353114633	0.3005435101428558

由上表可知，使用 regularization 並沒有使 Loss 下降，準確率也沒有特別上升，我認為原因是我的 model 設計的較為簡單，沒有加入二次或三次項去做 training，因此幾乎不會有甚麼 overfitting 的問題，如此的話，w 的大小其實應該不會影響預測結果，過大的 lambda 只會造成整個值域變小，得出的結果也將受到侷限。

(這裡 iter = 200, rate = 0.1, batch size = 128)

3. (1%) 請說明你實作的 best model，其訓練方式和準確率為何？

   我的 best model 作了以下的處理:

   1. 做 10-fold cross validation，再取其平均，避免資料分布過於極端。

   2. 依照 w 取最重要的 250 個 feature。

   3. 對於每個非 binary 的 feature 去做區間的 encoding (按照比例，人數等等手動做區間)，且 encoding 的方式為先將其區間照 50000+比例排序，比例相近的的 encoding 會較為相似，例如將年齡分成 a, b, c 三類 (由少到多)，則 encoding 方式為[0, 0, 0], [1, 0, 0], [1, 1, 0]。

   4. Iter = 500 batch size = 128 learning rate = 0.05

4. (1%) 請實作輸入特徵標準化 (feature normalization)，並比較是否應用此技巧，會對於你的模型有何影響。

   在我的 model 中已經不需要去做 normalization，因為全部都已經去做 binary encode 的處理了，我也有實驗去做 encoding 前 normalize 的情形，發現只對非 binary 的去做 normalization 有最好的效果，其他的由於已經固定一個不大的區間，去做 normalize 就不太會有差了 (反而會更差)。