Homework 2 Report - Income Prediction

學號：R06921002 系級：電機碩一 姓名：張哲誠

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

Ans:

Logistic regression 的準確率較高。以我手刻的 hw2_logistic.py 來說，最後在 training data 的 categorical accuracy(分類準確率)可以到 0.85368，而在 kaggle 上的最高分數為 0.85700；相對的，hw2_generative.py 在 training data 上的 categorical accuracy 是 0.842480 左右，而在 kaggle 上的分數有僅只有 0.84545，可見得不論是在 training data 或是 testing data，都是 Logistic regression 表現較優異。

雖然 hw2_generative.py 的結果較差，但我也發現 logistic model 對於 learning rate 與 regularization 值的選取相當敏感。若是我的 learning rate 太小，在 logistic model 上容易卡在鞍點(實測的鞍點準確率約為 0.72 左右)。然而，若是我的 learning rate 可以在 0.5~0.01 左右，除了上升速度快，也比較不容易卡在鞍點。Generative model 雖然準確率較低，但也因為不用調整 learning rate 與 regularization 值，通常表現比較穩定。

至於 generative model 會比較差的原因，我個人認為在於有無更新 w,b 參數是最重要的影響。通常在做 logistic model 時，若是某個 feature 不重要或對 predict 所產生的結果很小，我會發現 w 是會趨近於 0 的。但以 generative model 沒有做訓練、且視各 feature 同等重要下去算 mean 跟 covariance 的情況下，不重要的 feature 可能會影響最終分類的結果。因此，**在 training data 數量夠的情況底下，我認為會是 logistic model 的結果較佳**。

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

Ans:

本次的 hw2_best.py 是根據 hw2_logistic.py 下去做改寫。和 hw2_logistic.py 不同的是，我多加了 **age, sex_Female, sex_Male, capital_gain, capital_loss 與 hours_per_week 的平方項**當作新的 training data，並且設置 learning rate=0.00001, $\lambda = 0$。W 與 b 的部分，首先我先用 training data 與 label data 與 np.linalg.pinv 求**解析解**，並以解析解的答案當作初始值，開始做 logistic regression。如此作法，在 training data 中的準確率達到 0.8578974847209853，在 testing data，也就是 kaggle 上的成績，可以達到 0.85749~0.85786 之間的分數。

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)

Ans:

本次我實作的是參考頁面中的 Standardization，即是算出每個 feature 的平均值與標準差之後，對每一筆 data 減平均值再除標準差，即得到標準化的結果，這邊實作要

注意的是，我在除以標準差的部分多加一個值10^{-21}，這是為了避免標準差是 0，除的結果會導致無限大。

在本次作業中，有無做標準化對於 training 的結果影響很劇烈。我個人認為從兩點可以看出，分別是(1) **收斂準確率偏低**、(2)**收斂過程的不穩**，以下舉例皆以 hw2_logistic.py 執行結果為討論。

(1) **收斂準確率偏低→** 在這次的作業，我嘗試使用同樣的 learning rate(lr = 0.05)與 regularization($\lambda = 10$)，觀察有無標準化的結果。結果顯示，沒有標準化的結果約是收斂在 0.7936 的準確率左右，且上升幅度不明顯，我認為有可能是卡在某個鞍點。然而有做標準化的結果可以收斂到準確率 0.8424。

(2) **收斂過程的不穩→** 一樣是同樣的 learning rate 與 regularization 值，沒有做標準化的收斂過程會在 0.77~0.79、有時甚至會跳到 0.24 等，在整個訓練過程中，跳動幅度相當劇烈。但在有加 normalization 的情況下，相同的場景可以從 0.4498 穩定收斂到 0.8424。

綜合以上兩點，我認為 normalization 對模型的準確率有很大的影響，而有作 normalization 對於 training, testing 的結果皆較佳。以下 Fig. 1 為兩個不同場景的準確率成長圖。

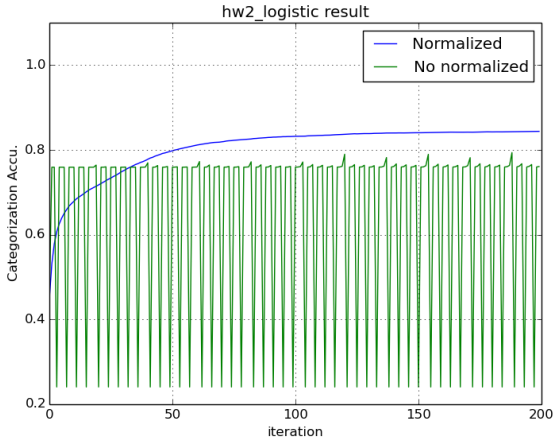
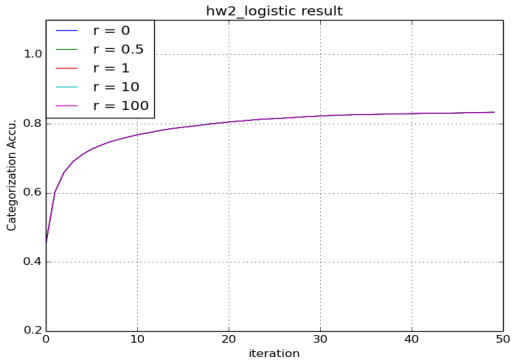


Fig. 1 有無標準化之準確率收斂圖

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)

Ans:

本題我在相同的 learning rate 的情況下，實作五種不同的 regularization 值，並探



討其影響。Table. 1 與 Fig. 2 顯示在不同的λ情況下，Accuracy 的分佈圖與最終的準確值。可以觀察到，不同的λ對於準確值的影響不大，準確值的收斂過程也不會因為不同的λ而有不同。會有這樣的現象，我個人認為是 raw data 本身的錯誤率就已經很低了，而 regularization 的原意在於讓 Loss function 更平滑、更能抵抗雜訊。但由於 raw data 本身雜訊就低的情況下，自然 regularization 的作用就不大了。

Regularization λ	Final Accu.
0	0.857191118209
0.5	0.856914713922
1	0.856976137097
10	0.856914713922
100	0. 857313964559

Table. 1 不同λ下的 Accuracy

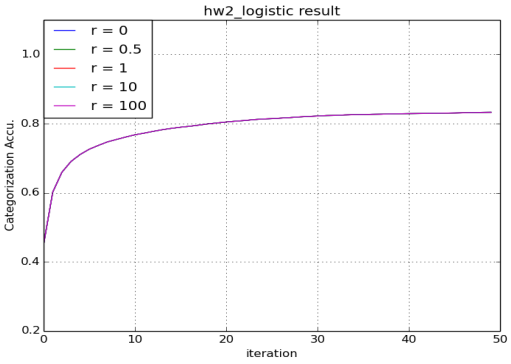


Fig. 2 不同λ下的 Accuracy 分佈圖

5. (1%) 請討論你認為哪個 attribute 對結果影響最大？

Ans:

我認為這次 attribute 影響最大的是 **age**，其次為 capital_gain, capital_loss 與 hours_per_week。

(1) age → 在 hw2_logistic.py 檔中，我除了 normalization 與 regularization 外，沒有其他資料處理，如此在 training data 中的準確率為 0.853~0.854 之間，上傳到 kaggle 上也可以到 0.87400(最高成績)。但在 hw2_best.py 中，我加入 age 的平方項後，不僅在 training data 的 accuracy 可以來到 0.857，在 kaggle 上的成績更可以突破 strong baseline。因此我認為，age 是這一次影響最大的 attribute。

(2) capital_gain, capital_loss 與 hours_per_week → 這幾項因素在我加上 age 的平方向後，也隨之加上這幾項 attribute 的平方項，雖然在 kaggle 上的成績沒有很顯著的幫忙，但在 training data 中，accuracy 可以穩定維持於 0.858 以上。

其餘如 sex, race 等 attribute 也有類似的影響，但我認為沒有像上述所提及的 4 種特徵影響的劇烈。