

Machine Learning HW2 Report

學號：B06901045 系級：電機三 姓名：曹林熹

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？

ANS:

- a. 在 generative model 時，training accuracy 為下圖所示，此時 covariance matrix 在 class 0 與 class 1 是不同的：

Training accuracy: 0.8693232084930699

若 covariance matrix 在 class 0 與 class 1 改成相同的，這時會變成 linear boundary，training accuracy 為下圖所示：

Training accuracy: 0.8674985255086995

可以發現 covariance matrix 相同時，accuracy 反而下降，此時我們的模型會變成 linear model，分類的情況會是線性的，因此模型受限較多。

- b. 在使用 logistic model 時，所有的 loss 與 accuracy 為下圖所示：

Training time is 184.4669210910797
Training loss: 0.2638425902794183
Development loss: 0.2849908689326334
Training accuracy: 0.8861970100348147
Development accuracy: 0.8778105418356064

可以明顯看出，運用 logistic model 的準確率較高，不管是 Training accuracy 或是 Development accuracy，logistic model 的機率模型不像 generative model 一樣受到固定的機率分布(ex. Gaussian, Bernoulli)，因此預測的彈性又相較高了一點。

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。(有關 regularization 請參考 <https://goo.gl/SSWGhf> p. 35)

ANS:

a. L1 Regularization (Lasso Regression)

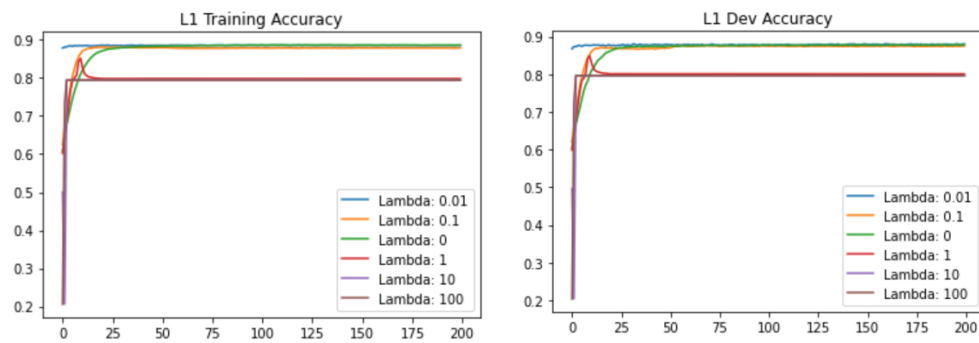
使用 L1 Regularization，我們可以改寫 Loss Function 如下所示，其中 $\lambda > 0$ ：

$$L_1 = (wx + b - y)^2 + \lambda|w|$$

經過 gradient ascent 後，得到的 W 為：

$$\begin{aligned} w_{\text{new}} &= w - \eta \frac{\partial L_1}{\partial w} \\ &= w - \eta \cdot [2x(wx + b - y) + \lambda \frac{d|w|}{dw}] \\ &= \begin{cases} w - \eta \cdot [2x(wx + b - y) + \lambda] & w > 0 \\ w - \eta \cdot [2x(wx + b - y) - \lambda] & w < 0 \end{cases} \end{aligned}$$

要注意的是，我們之前在初始化 W 時，都是將 W 設為 0，然而在 L1 Regularization 的情況下，初始化為 0 的 W 無法微分，因此這部分我初始化為 1。而得到的趨勢圖如下，總共採用 500 epoch，圖中顯示 200 epoch 有利於觀察。



b. L2 Regularization (Ridge Regression)

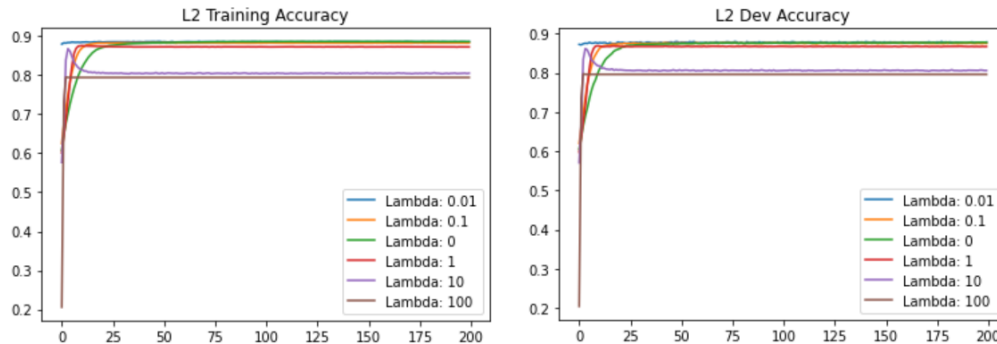
使用 L2 Regularization，我們可以改寫 Loss Function 如下所示，其中 $\lambda > 0$ ：

$$L_2 = (wx + b - y)^2 + \lambda w^2$$

經過 gradient ascent 後，得到的 W 為：

$$\begin{aligned} w_{\text{new}} &= w - \eta \frac{\partial L_2}{\partial w} \\ &= w - \eta \cdot [2x(wx + b - y) + 2\lambda w] \end{aligned}$$

為了方便與 L1 比較，初始化一樣使用 $W = 1$ 。得到的趨勢圖如下，總共採用 500 epoch，圖中顯示 200 epoch 有利於觀察。



總結 L1 與 L2 的兩張圖，可以觀察到兩者共通點是當我們的 λ 愈大通常效果愈不好，而 $\lambda = 0.01$ 時都比沒有用 regularization 還要好。此外，兩者的趨勢圖非常像，說明 L1 與 L2 在這個例子下並沒有太大差別，而且因為 training acc 與 dev acc 實在很接近，說明模型沒有 overfitting，因此不加 regularization 好像沒有什麼大影響。

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

ANS:

這邊我使用的 best model 是 logistic model，採用不同的 λ 後得到當 $\lambda = 0.01$ 時的 L2 logistic model 有最好的表現，dev acc 約為 0.878 - 0.879 左右。讀者可以使用 neural networks 的架構訓練，或許可以得到更好的正確率。

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。

ANS:

可以看到沒有經過 feature normalization 的模型會使訓練效果下降，主因是沒有 feature normalization 的會讓 model 收斂變困難，難以找到 loss 的極值。經過 feature normalization 的資料會被壓縮在同一個 scale 內，使得模型好收斂。(上面幾題都是有經過 feature normalization，下圖顯示 logistic model 沒有使用的訓練趨勢圖，然而我們發現 generative model 的 training acc = 0.876 反而上升，可能是因為在 generative model 不需要運用梯度，因此沒有無法收斂的問題，不標準化資料則使特徵差距更顯著)

