學號：R05921033　系級：電機所碩一　姓名：余政穎

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

答：我實作的 generative model，主要是參照老師上課所說，先將資料分成兩類，使用的 features 為助教所提供的 X_train 以及 X_test，且有經過 normalize，將每一個 attribute 的資料都作正規化，使其權重相等；接著，分別算出兩類資料中每一個 attribute 的平均值以及其 covariance matrix，以計算(類別 1 代表大於等於 50k，label 為 1；類別 2 則否，label 為 0)：

$$\mathbf{w^T} = (\mu^1 - \mu^2)^T\Sigma^{-1}, b = -\frac{1}{2}(\mu^1)^T\Sigma^{-1}\mu^1 + \frac{1}{2}(\mu^2)^T\Sigma^{-1}\mu^2 + \ln(\frac{N_1}{N_2})$$

計算完畢之後，便將其拿去預測測試資料，以$\mathrm{P}(C_1|\mathbf{x}) = \sigma(\mathbf{w^T}\cdot\mathbf{x} + \mathrm{b})$是否大於 0.5 作為基準，若是，則預估此資料 label 為 1，反之則為 0；所預估出來的測試資料結果，放上 kaggle 所得到的 public score 為 0.82236。

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

答：我實作的 discriminative model，即是用 logistic regression 完成，使用的 features 也是助教所提供的 X_train 以及 X_test，且有將連續的資料經過 normalize，將這些 attributes 的資料都作標準化(此處沒特別針對已經是 discrete data 做標準化，因這樣比較接近原本特性，有則 1 沒有則 0)，並將 bias 項直接併到矩陣中(作為 w 向量的第一個元素，或可稱 w_0)；接著，初始化 w，套入 logistic regression 的計算公式，$\mathbf{w} \leftarrow \mathbf{w} - \eta\sum_n -\big(\hat{y}^n - f_w(\mathbf{x}^n)\big)\mathbf{x}^n$，其中$f_w(\mathbf{x}^n) = \sigma(\mathbf{w^T}\cdot\mathbf{x})$，此處 b 已經被包含在$\mathbf{w}$中，n 為第 n 個 example。

接著，將十分之一的訓練資料取出作為 validation set，validation set 的資料的 label 分布與原訓練資料類似，在每一個 epoch，我先將 training set 進行 shuffle，並使用兩種更新$\mathbf{w}$的方法：第一種是將全部 training set 丟進去更新$\mathbf{w}$，第二種是用 mini-batch，將訓練資料切成許多份資料，一次用其中一份資料更新$\mathbf{w}$；epoch 上限設為 2000，但同時會使用 validation set 做檢查，減低 overfitting 發生的可能性，此處的檢查方法是看目前的模型在 validation set 上的正確率，若連續 10 個 epoch 都沒有再下降的話，便停止訓練，並以此模型預測測試資料。

其餘參數設定，經跑過不同參數測試，上述兩種方法跑出來的效果差不多，而我所得最好結果是將η設為 1，使用 adagrad 更新$\mathbf{w}$，上傳 Kaggle 的 public score 之結果為 0.85356。

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：此部分我以三種不同特徵標準化的設定，並分別使用 discriminative model (logistic regression)與 generative model 來比較其效果(其他設定同 1, 2 題)，觀察標準化對於模型準確率的影響，準確率是用第二題中得到的 validation set 衡量。第一種完全不使用標準化，第二種是針對 continuous 資料使用標準化，第三種是針對全部資料使用標準化，得到結果如下：

	第一種	第二種	第三種
Discriminative model	0.75921 (Overflow)	0.85197	0.85197
Generative Model	0.75921	0.53165	0.82236

此部分我們可以觀察到，基本上做標準化對於兩種模型都比較好，否則預測出來的結果基本上就等於全部猜 0；對 discriminative model，後兩種的標準化方式看起來沒有很大差異，如在 2.敘述的原因，最後這邊採用第二種標準化方式；較意外的是，generative model 需要對於

全部資料做標準化，才能得到較準確的結果，這部分與助教所呈現的 code 不同，其中的差異可能是因為我在計算 covariance matrix 之時，是使用 numpy.cov，造成不同之效果。

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：在 logistic regression 中，我實做了正規化，將 w 的長度也考慮進去，使其不要太大，並使用 2.中的所切出的 validation set，跑 logistic regression，以預測正確率為標準，紀錄每一個 epoch 的所得準確率，並以最小能獲得的錯誤率為基準，進而觀察正規化影響模型準確率。

	$\lambda = 0$	$\lambda = 0.0001$	$\lambda = 0.0002$	$\lambda = 0.0003$	$\lambda = 0.0004$	$\lambda = 0.0005$
準確率	0.85197	0.84982	0.84705	0.84582	0.84521	0.84398
	$\lambda = 0.00005$	$\lambda = 0.00015$	$\lambda = 0.00025$	$\lambda = 0.00035$	$\lambda = 0.00045$	$\lambda = 0.00055$
準確率	0.85166	0.84797	0.84613	0.84552	0.84429	0.84429

可以觀察到，加上正規化的模型，其預測準確率有些影響，正規化越多，準確率越差，這可能是因為我們在 logistic regression 是針對 cross entropy 做最佳化，而非直接對正確率做最佳化，且資料量其實不算太多，所以過多正規化反而造成模型預測不夠準確，亦有可能是λ還是取太大，如果選用再更小一點的λ，也許會得到較好的結果。

5.請討論你認為哪個 attribute 對結果影響最大？

答：這部分我使用兩種方法來觀察 attribute 對結果之影響，第一個是輸出兩類資料中每一個 attribute 的平均值，以及針對 discrete 資料，統計兩類資料該 attribute 出現 1 的比例，看看其中的差異，進而推出較有影響力的 attribute；經觀察，continuous 資料幾乎滿重要的，兩類資料所得的平均值相差甚大，此外，部分的 discrete 資料，如 Married-civ-spouse、Never-Married，在整體資料中出現 1 的比例頗高，且於兩類資料中的分布也有明顯差距；為了確認，我亦嘗試只使用部分 features 拿去訓練，看看出來結果之正確率，不過，可能是因為資料量不算多，基本上無論如何挑選，使用部分 features 所訓練出的模型準確率都會變差。

不過單就第一種方式，只能大致分類，尚難以看出哪一個 attribute 對結果影響最大，因此我便實作了 Adaboost 演算法，利用 decision stump 這個較弱的分類器以符合 Adaboost 之要求。Decision stump 是從所有的 attributes 挑一個，並從此 attribute 找出一個適合的 threshold 切一刀，假如所選的是資料中第 i 個 attribute 的第 j 個 threshold，則資料中第 i 個 attribute 的值若大於此 threshold，則 label 為 1，小於則為 0(或者相反，小於此 threshold 為 1，大於為 0)；對於 discrete 資料，只有四種情況(不論 attribute 為何，皆預測 1 或 0；attribute 為 1 預測 1，否則預測 0；attribute 為 0 預測 1，否則預測 0)，而對於 continuous 資料，我利用資料的平均值切十個 thresholds，並從中找尋最佳 threshold；Adaboost 則是一個著名的分類演算法，簡而言之，是透過每一輪將預測錯誤的資料放大，使得每一次所得的弱分類器在下一輪看起來像是隨機猜，並將一堆弱分類器結合，進而成一個強分類器。由於透過此法可以得到一堆弱分類器，經由統計這些弱分類器分別來自哪些 attribute 的數量以及準確率，可以大概推測出哪個 attribute 對結果影響最大。經觀察，capital_gain 這個 attribute 最常出現在演算法所得的分類器中，且單使用”一個”capital_gain 之 decision stump，便可得到超過八成的正確率(若全預測 0，得 0.76 左右)，由此可見，capital_gain 這個 attribute 應對結果影響最大！