學號：B03901161　系級：　電機四　姓名：楊耀程

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

答：

Generative 實作: 使用助教抽取好的 feature，使用相同的 covariance matrix，值得注意的是做共用 covariance matrix 的 inverse 時，發現他 determinant 非常接近 0，因此用 pseudo inverse 來取代一般的 inverse 後，才得到較理想的結果。

Logistic regression 實作: 使用助教抽取好的 feature，並做 normalization。使用 cross entropy 作為 loss，用 gradient descent (adagrad)。我實做了兩種: 只取一次方的 feature，和一次二次 feature 都取的 logistic regression。

準確度: logistic 二次>logistic 一次>generative，不論在 validation set 還是 kaggle public 都成立，整體而言 logistic regression 準確度較 generative 佳。

	generative	logistic 一次	logistic 二次
validation set	0.8401	0.8488	0.8541
kaggle public	0.84533	0.8538	0.8597

2.請說明你實作的 best model，其訓練方式和準確率為何？

答：

先後嘗試了 logistic 二次，deep neural network (using tensorflow)，還有 xgboost，發現 xgboost 的表現在 validation 以及 kaggle public 都明顯優於前兩者，因此我選擇 xgboost 實作的版本為 best model。

訓練方式: 使用助教抽取好的 feature，並做 normalization，利用 xgboost 的 classifier，搭配 cross validation 調參數，參數與整體架構如下圖。

```
model = XGBClassifier(
 learning_rate =0.2,
 max_depth=5,
 min_child_weight=1,
 gamma=0,
 subsample=0.8,
 colsample_bytree=0.8,
 objective= 'binary:logistic',
 nthread=4,
 scale_pos_weight=1,
 seed=27)

#model.fit(x_train2, y_train_raw2.ravel())
model.fit(x_train, raw_label.ravel())

# make predictions for test data
#y_pred = model.predict(x_train1)
y_pred = model.predict(x_test)

predictions = [round(value) for value in y_pred]
```

(max depth 不選太深，可避免 overfit)

準確度: 可以看到在 validation 與 kaggle public，預測的準確都相當好，明顯優於 generative、 logistic regression 的結果。

	validation set	kaggle public
xgboost	0.8717	0.87727

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：

	generative	logistic 一次
有標準化/無標準化	0.84533/ 0.84496	0.8538/0.7947

結論: 特徵標準化對於 Generative 影響很小，不過對於 logistic regression 影響很大，沒有標準化的準確度大幅下降，推測是因為某幾樣 continuous 的 feature 數字很大，像是 capital loss 有些是 0，有些卻是 99999，如果沒做標準化，做 gradient descent 時就很可能失準，反之 generative 不需要用到 gradient descent，則無此問題(微小差距可能來自於算 covariance matrix pseudo inverse 時的誤差影響)。

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：

reg lambda	0	0.1	0.01	0.001	0.0001
acc on public	0.8538	0.85368	0.8538	0.85442	0.85442

討論: 可以觀察到在 lambda=0.001 以及 0.0001 時，準確度有微幅提升，但是 lambda =0.01 時，準確度就沒有提升了，lamda=0.1 則準確度有些微下降，整體準確度差距都不大。可見此處 regularization 的影響不是太大，而當 lambda = 0.001, 0.0001 時會對於準越度有些微幫助。

5.請討論你認為哪個 attribute 對結果影響最大？

答：

fnlwgt	sex	capital gain	capital loss	age	work class
using 1	using 2	using 3	using 4	using 0	using 6 to 14
0.761487833	0.761487833	0.799106334	0.770726405	0.750075479	0.763540849
education	marital	occupation	relationship	race	nation
using 15 to 30	using 31 to 37	using 38 to 52	using 53 to 59	using 59 to 63	using 64 to 105
0.780448041	0.761487833	0.761487833	0.759374434	0.761487833	0.76142745

討論: 分別只用單一個 attribute 做，看在 validation set 準確度結果如何，可以發現只用 capital gain 做，準確度就達到了 0.799，代表這項有相當大的影響力，因此我認為 capital gain 對結果影響最大。其次是 education 與 capital loss。另外我也嘗試用全部的 feature，扣掉部分 attribute，看其影響。也是不用 capital gain 影響最大，準確度下降最多。受限於篇幅，就只放一個抽掉 sex 的影響，可以發現影響幅度差很多，不用其他 attribute 時(一次只扣除一個 attribute)，都沒有扣掉 capital gain 來的影響大。

	feature全用	不用capital gain	不用sex
準確度	0.848801401	0.834913351	0.847654127