

## 1. How do you select features for your model input, what preprocessing did you do ?

- preprocessing:

- Missing data : 在開始進行數據分析前，有發先些許“?”的 missing data , 這個部分在經過篩選後發現占總 data 比例不高，因此選擇直接刪除

```
# Delete missing data
col_names = dataset.columns
num_data = dataset.shape[0]
for c in col_names:
    num_non = dataset[c].isin(["?"]).sum()
    if num_non > 0:
        print (c)
        print (num_non)
        print ("{0:.2f}%".format(float(num_non) / num_data * 100))
        print ("\n")

dataset = dataset[dataset["workclass"] != "?"]
dataset = dataset[dataset["occupation"] != "?"]
dataset = dataset[dataset["native-country"] != "?"]
```

```
workclass
2217
5.67%
```

```
occupation
2225
5.69%
```

```
native-country
687
1.76%
```

- Categorical data encoding : 因為本次作業有單純數據的 data , 同時也有類別行的 data , 因此要一同進行計算時，類別型的 data 就需要重新編碼才能進行後續的處理。

我選擇的編碼方式是 Label Encoding : 將每個分類值轉換為一個整數。

```
In [8]: # Deal with categorical columns

category_col =['workclass', 'race', 'education','marital-status', 'occupation',
'relationship', 'gender', 'native-country', 'income']

for col in category_col:
    b, c = np.unique(dataset[col], return_inverse=True)
    dataset[col] = c

dataset.head()
```

Out[8]:

	age	workclass	fnlwgt	education	educational-num	marital-status	occupation	relationship	race	gender	capital-gain	capital-loss	hours-per-week	native-country	income
0	52	2	210736	15	10	2	3	0	4	1	3103	0	55	38	1
1	62	2	209844	15	10	0	0	4	4	0	0	0	30	38	0
2	25	2	410240	11	9	4	2	3	4	1	0	0	40	38	0
3	28	2	90547	11	9	2	7	5	2	0	0	0	23	38	0
4	28	2	132326	11	9	4	0	3	4	1	0	0	40	38	0

- Feature selection : 透過計算了各個特徵變量和收入之間的相關係數 (correlation) 和絕對相關係數 (abs\_corr) , 包括點二列相關係數 (Point-Biserial Correlation) 和斯皮爾曼等級相關係數 (Spearman Rank Correlation) , 可以發現取前四個 features 分別是['educational-num' 'relationship' 'age' 'gender']

```

#Create dataframe for visualization
param_df = pd.DataFrame({'correlation':correlation,'parameter':param, 'abs_corr':abs_corr})

#Sort by absolute correlation
param_df = param_df.sort_values(by=['abs_corr'], ascending=False)

#Set parameter name as index
param_df = param_df.set_index('parameter')

print(param_df)

best_col = param_df.index[0:4].values

print(best_col)

```

	correlation	abs_corr
parameter		
educational-num	0.330767	0.330767
relationship	-0.250225	0.250225
age	0.236322	0.236322
gender	0.232375	0.232375
hours-per-week	0.224649	0.224649
capital-gain	0.220880	0.220880
marital-status	-0.192672	0.192672
capital-loss	0.149591	0.149591
education	0.083913	0.083913
race	0.076426	0.076426
occupation	0.047696	0.047696
native-country	0.020730	0.020730
workclass	0.015571	0.015571
fnlwgt	-0.007827	0.007827
['educational-num' 'relationship' 'age' 'gender']		

2. Discuss which attribute has the greatest impact on income prediction accuracy, visualize the result and explain it.

透過 excel 的分析工具箱計算詳細的線性迴歸係數，如下表所示

	係數	標準誤	t 統計	P- 值	下限 95%	上限 95%	下限 95.0%	上限 95.0%
截距	-0.57128	0.021121	-27.0477	1.6E-159	-0.61268	-0.52989	-0.61268	-0.52989
age	0.005035	0.000159	31.59174	4.2E-216	0.004722	0.005347	0.004722	0.005347
workclass	-0.01555	0.002062	-7.54286	4.71E-14	-0.0196	-0.01151	-0.0196	-0.01151
fnlwgt	6.87E-08	1.86E-08	3.702328	0.000214	3.23E-08	1.05E-07	3.23E-08	1.05E-07
education	-0.00293	0.000548	-5.33709	9.5E-08	-0.004	-0.00185	-0.004	-0.00185
educational-num	0.047799	0.000844	56.62511	0	0.046144	0.049453	0.046144	0.049453
marital-status	-0.0233	0.001382	-16.8625	1.49E-63	-0.026	-0.02059	-0.026	-0.02059
occupation	0.000786	0.00049	1.603896	0.108746	-0.00017	0.001747	-0.00017	0.001747
relationship	-0.01696	0.001581	-10.73	8.07E-27	-0.02006	-0.01387	-0.02006	-0.01387
race	0.018153	0.002393	7.586692	3.36E-14	0.013463	0.022843	0.013463	0.022843
gender	0.11099	0.005213	21.29024	5.7E-100	0.100772	0.121208	0.100772	0.121208
capital-gain	9.05E-06	2.64E-07	34.24324	6.1E-253	8.53E-06	9.57E-06	8.53E-06	9.57E-06
capital-loss	0.000113	4.87E-06	23.24821	1.1E-118	0.000104	0.000123	0.000104	0.000123
hours-per-week	0.003331	0.000173	19.23092	5.24E-82	0.002992	0.003671	0.002992	0.003671
native-country	-0.00072	0.000328	-2.20115	0.027732	-0.00136	-7.9E-05	-0.00136	-7.9E-05

各個 features 的係數：代表 features 變動一個單位會造成 income 有多少單位的變動，  
 假設 age 的係數為 0.005035，則 age 變動 2 單位時，income 僅會變動  $2 * 0.005035 = 1$  單位。

P-value : P-value 代表此 features 變數是不是能有效的影響 income 變數，通常使用 0.05 判斷，若 P-value 小於 0.05，則代表此 features 變數能有效的影響 income 變數，若 P-value 大於 0.05，則代表此 features 變數的變動 機率上來說不太能影響到 income。就表格的結果來看，只有 occupation 這個變量和結果無關，其他的變量多少都會影響 income 的結果。

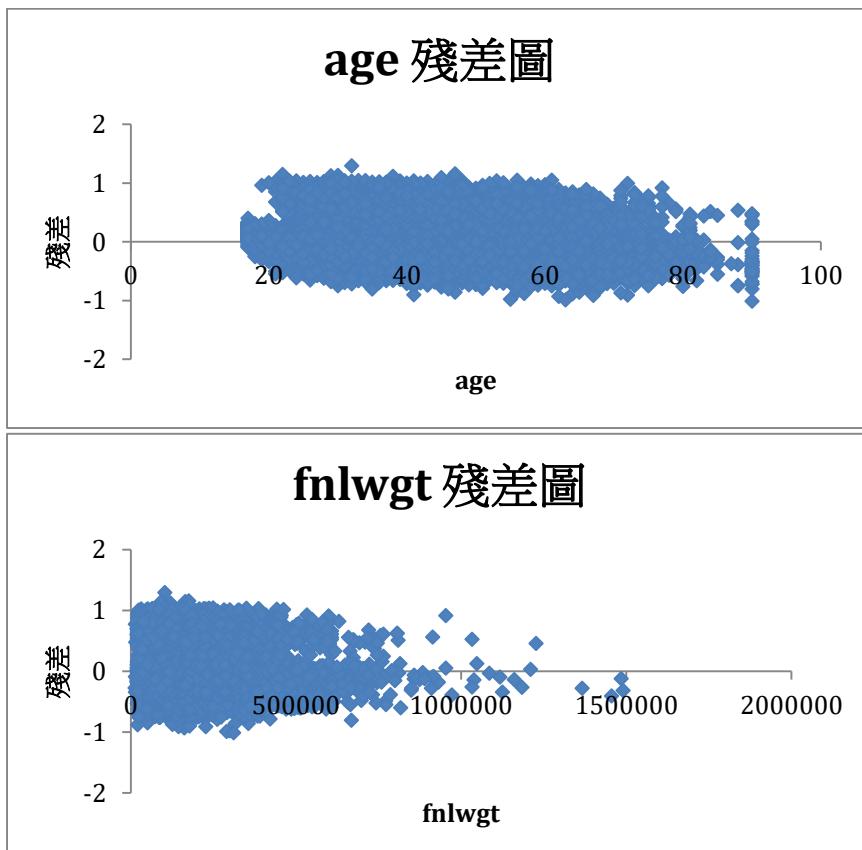
在殘差圖中，應該看到隨機的分佈圖案。如果殘差值呈現一定的趨勢或模式，那麼這可能表明模型未能捕捉到資料中的某些重要特徵，或者存在某些線性回歸模型的假設未被滿足。

當殘差圖中出現以下情況時，應該進一步檢查模型：

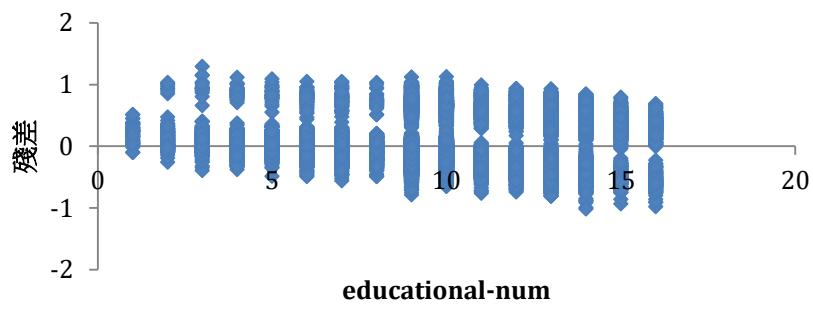
1. 非隨機分佈 - 如果殘差呈現某些明顯的模式或趨勢，那麼模型可能存在一些重要的未建模變數或模型規範未被滿足。
2. 異常值 - 如果在殘差圖中出現一些極端值，那麼這些可能是極端的資料點，可以在進行模型擬合時將其排除。
3. 非常數方差 - 如果殘差的變異量隨著預測值的增加而變大或變小，那麼模型可能存在異常的變異量，這可能是由於某些非線性效應或交互作用導致的。

實際的殘差結果如下(只參考連續變量的資料)：

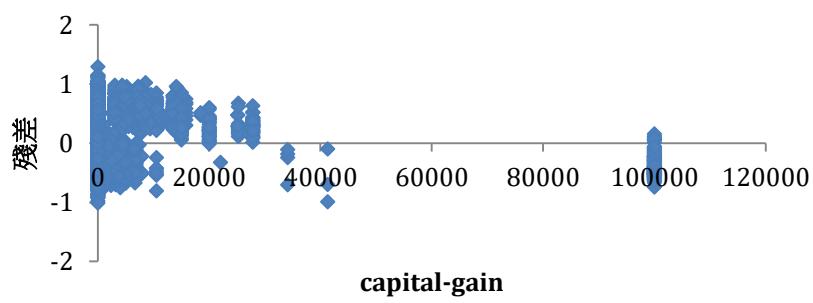
可以推估 fnlwgt 沒辦法符合迴歸模型的假設



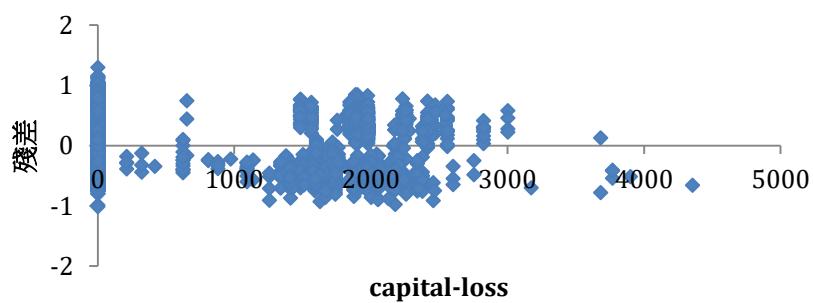
### **educational-num 殘差圖**

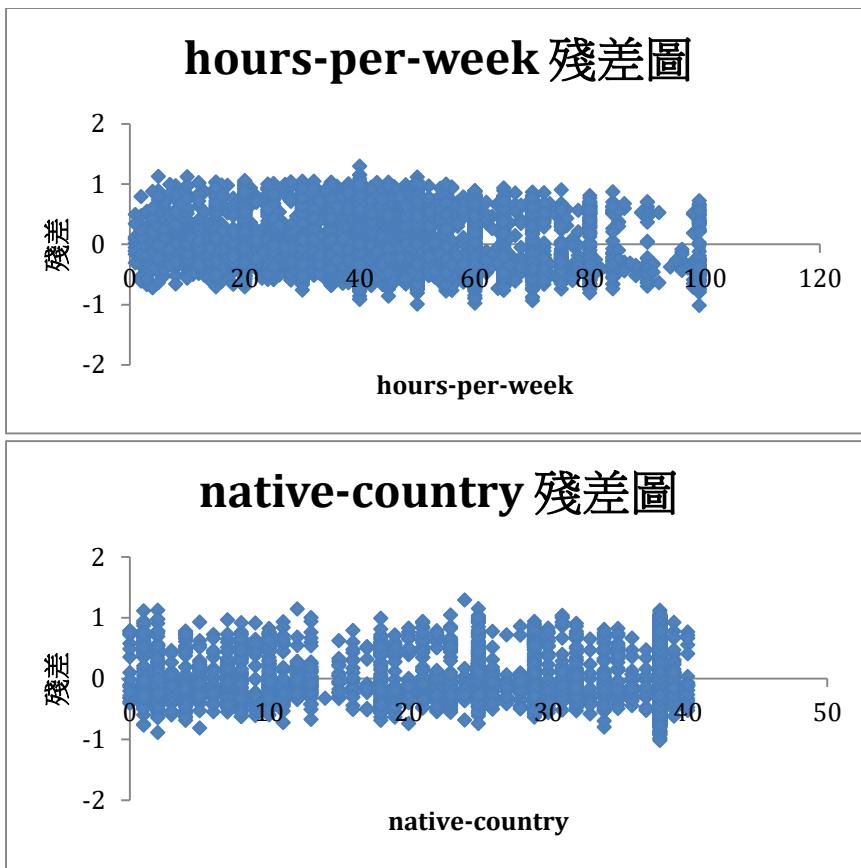


### **capital-gain 殘差圖**



### **capital-loss 殘差圖**





3. Discuss the impact of regularization on income prediction accuracy.

hw2 和 hw1 最大的區別，就在於存在大量的類別資料。預測模型的訓練數據既包含類別資料又包含數值資料，正則化仍然可以有效地幫助提高模型的泛化能力和準確度。通常情況下，正則化主要針對模型的權重進行調整，因此對於類別型特徵而言，正則化並不會產生太大的影響。對於數值型特徵，正則化可以幫助減小特徵的權重，從而避免過度擬合數據，提高模型的泛化能力。

不同的正則化方法可能會對不同類型的特徵產生不同的影響。例如，**L1** 正則化可以有效地減少模型中不重要的特徵，因此對於包含大量冗余特徵的數據集而言，**L1** 正則化可能會產生更好的效果。而 **L2** 正則化則可以平滑權重向量，從而降低模型的方差，對於數據中存在較多噪聲的情況下，**L2** 正則化可能會更適合。