Python 部分

第一題

發現是 Gender, Age, EstimatedSalary 有遺漏值

第一題

```
import numpy as np
import pandas as pd

raw_data = pd.read_csv("Churn_Modelling.csv")
df = pd.DataFrame(raw_data)
df.head()
```

	CustomerId	CredRate	Geography	Gender	Age	Tenure	Balance	Prod Number	HasCrCard	ActMem	EstimatedSalary	Exited
0	15634602	619	France	Female	42.0	2	0.00	1	1	1	101348.88	1
1	15647311	608	Spain	Female	41.0	1	83807.86	1	0	1	112542.58	0
2	15619304	502	France	Female	42.0	8	159660.80	3	1	0	113931.57	1
3	15701354	699	France	Female	39.0	1	0.00	2	0	0	93826.63	0
4	15737888	850	Spain	Female	43.0	2	125510.82	1	1	1	79084.10	0

```
df.isnull().sum()
```

```
CustomerId         0
CredRate           0
Geography          0
Gender             4
Age                6
Tenure             0
Balance            0
Prod Number        0
HasCrCard          0
ActMem             0
EstimatedSalary    4
Exited             0
dtype: int64
```

第二題

以平均值填入 EstimatedSalary 的遺漏值，以眾數填入 Age 與 Gender 的遺漏值

```
mean_values = df["EstimatedSalary"].mean()
df["EstimatedSalary"].fillna(value=mean_values, inplace=True)

mode_values_gender = df["Gender"].mode()[0]
mode_values_age = df["Age"].mode()[0]
df["Gender"].fillna(value=mode_values_gender,inplace = True)
df["Age"].fillna(value=mode_values_age,inplace = True)
```

```
df.isnull().sum()
```

```
CustomerId         0
CredRate           0
Geography          0
Gender             0
Age                0
Tenure             0
Balance            0
Prod Number        0
HasCrCard          0
ActMem             0
EstimatedSalary    0
Exited             0
dtype: int64
```

第三題

修改欄位

第三題

```
df = df.rename(columns={'CredRate': 'CreditScore','ActMem':'IsActiveMember','Prod Number':'NumOfProducts','Exited':'Churn'})
df
```

	CustomerId	CreditScore	Geography	Gender	Age	Tenure	Balance	NumOfProducts	HasCrCard	IsActiveMember	EstimatedSalary	Churn
0	15634602	619	France	Female	42.0	2	0.00	1	1	1	101348.88	1
1	15647311	608	Spain	Female	41.0	1	83807.86	1	0	1	112542.58	0
2	15619304	502	France	Female	42.0	8	159660.80	3	1	0	113931.57	1
3	15701354	699	France	Female	39.0	1	0.00	2	0	0	93826.63	0
4	15737888	850	Spain	Female	43.0	2	125510.82	1	1	1	79084.10	0
...	...	...	...	...	...	...	...	...	...	...	...	...
9995	15606229	771	France	Male	39.0	5	0.00	2	1	0	96270.64	0
9996	15569892	516	France	Male	35.0	10	57369.61	1	1	1	101699.77	0
9997	15584532	709	France	Female	36.0	7	0.00	1	0	1	42085.58	1
9998	15682355	772	Germany	Male	42.0	3	75075.31	2	1	0	92888.52	1
9999	15628319	792	France	Female	28.0	4	130142.79	1	1	0	38190.78	0

10000 rows × 12 columns

第四題

第四題

```
df.drop('CustomerId',axis=1,inplace= True)
df
```

	CreditScore	Geography	Gender	Age	Tenure	Balance	NumOfProducts	HasCrCard	IsActiveMember	EstimatedSalary	Churn
0	619	France	Female	42.0	2	0.00	1	1	1	101348.88	1
1	608	Spain	Female	41.0	1	83807.86	1	0	1	112542.58	0
2	502	France	Female	42.0	8	159660.80	3	1	0	113931.57	1
3	699	France	Female	39.0	1	0.00	2	0	0	93826.63	0
4	850	Spain	Female	43.0	2	125510.82	1	1	1	79084.10	0
...	...	...	...	...	...	...	...	...	...	...	...
9995	771	France	Male	39.0	5	0.00	2	1	0	96270.64	0
9996	516	France	Male	35.0	10	57369.61	1	1	1	101699.77	0
9997	709	France	Female	36.0	7	0.00	1	0	1	42085.58	1
9998	772	Germany	Male	42.0	3	75075.31	2	1	0	92888.52	1
9999	792	France	Female	28.0	4	130142.79	1	1	0	38190.78	0

10000 rows × 11 columns

```
df['Geography']=df['Geography'].astype('category')
df['Gender']=df['Gender'].astype('category')
df['HasCrCard']=df['HasCrCard'].astype('category')
df['Churn']=df['Churn'].astype('category')
df['IsActiveMember']=df['IsActiveMember'].astype('category')
```

```
df.dtypes#欄位屬性輸出
```

```
CreditScore           int64
Geography          category
Gender             category
Age                 float64
Tenure                int64
Balance             float64
NumOfProducts         int64
HasCrCard          category
IsActiveMember     category
EstimatedSalary     float64
Churn              category
dtype: object
```

```
df.to_csv('Churn_Modelling_new.csv', encoding = 'utf-8-sig',index=False) #匯出新csv
```

去除 CustomerId,欄位，並將 Geography、Gender、HasCrCard、 Churn、IsActiveMember 修改資料型態為 category，印出所有欄位的資 料型態，並存成新的 CSV 檔 (設定 index=False)。

第五題

(一)

```
group_sizes_HasCrCard = df.groupby('HasCrCard').size()
print('有信用卡的人比例:', group_sizes_HasCrCard.iloc[1]/df.shape[0])
print('無信用卡的人比例:', group_sizes_HasCrCard.iloc[0]/df.shape[0])
#5-1
```

```
有信用卡的人比例: 0.7055
無信用卡的人比例: 0.2945
```

(二)

```
group_sizes_Churn = df.groupby('Churn').size()
print('流失的客戶比例:', group_sizes_Churn.iloc[1]/df.shape[0])
#5-2
```

```
流失的客戶比例: 0.2037
```

(三)

```
group_sizes_IsActiveMember = df.groupby('IsActiveMember').size()
print('仍是活躍狀態的客戶比例:', group_sizes_IsActiveMember.iloc[1]/df.shape[0])
#5-3
```

```
仍是活躍狀態的客戶比例: 0.5151
```

(四)

```
no_churn_df = (df["Churn"]==0)#5-4
has_churn_df=(df["Churn"]==1)
```

```
df.loc[no_churn_df].mean()
```

```
CreditScore          651.853196
Age                   37.411277
Tenure                 5.033279
Balance            72745.296779
NumOfProducts          1.544267
HasCrCard              0.707146
IsActiveMember         0.554565
EstimatedSalary    99718.932023
Churn                  0.000000
dtype: float64
```

```
df.loc[has_churn_df].mean()
```

```
CreditScore           645.351497
Age                    44.837997
Tenure                  4.932744
Balance             91108.539337
NumOfProducts           1.475209
HasCrCard               0.699067
IsActiveMember          0.360825
EstimatedSalary    101465.677531
Churn                   1.000000
dtype: float64
```

	流失客戶	未流失客戶
CreditScore	較低	較高
Age	較高	較低
Tenure	較低	較高
Balance	較高	較低
NumOfProducts	較低	較高
HasCrCard	較低	較高
IsActiveMember	較低	較高
EstimatedSalary	較高	較低

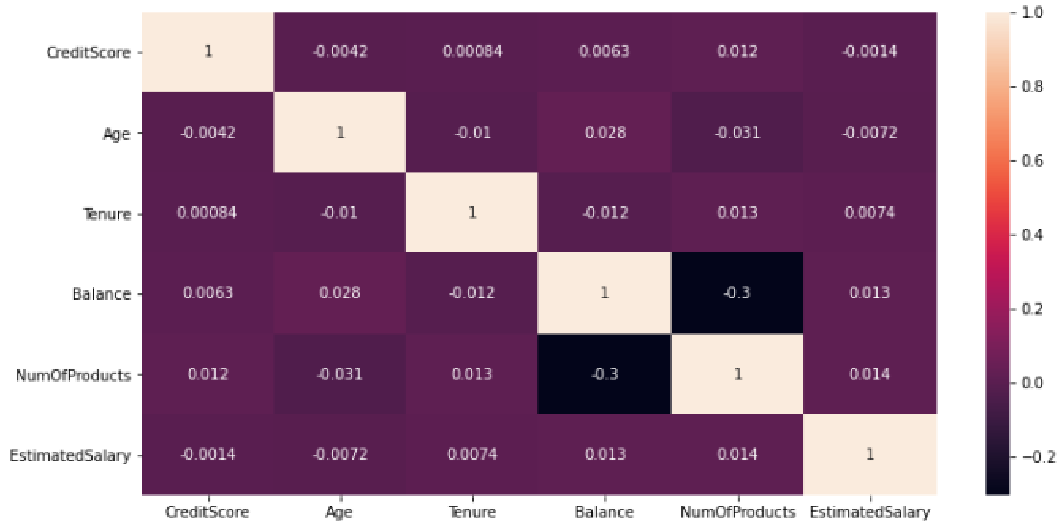
(五)

```
#5-5
df.corr()
```

	CreditScore	Age	Tenure	Balance	NumOfProducts	EstimatedSalary
CreditScore	1.000000	-0.004179	0.000842	0.006268	0.012238	-0.001352
Age	-0.004179	1.000000	-0.009996	0.028141	-0.030590	-0.007215
Tenure	0.000842	-0.009996	1.000000	-0.012254	0.013444	0.007407
Balance	0.006268	0.028141	-0.012254	1.000000	-0.304180	0.013129
NumOfProducts	0.012238	-0.030590	0.013444	-0.304180	1.000000	0.014132
EstimatedSalary	-0.001352	-0.007215	0.007407	0.013129	0.014132	1.000000

```
import seaborn as sns#5-5
import matplotlib.pyplot as plt
corr = df.corr()
plt.figure(figsize=(12,6))
sns.heatmap(corr,annot=True)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0xb879a18>
```



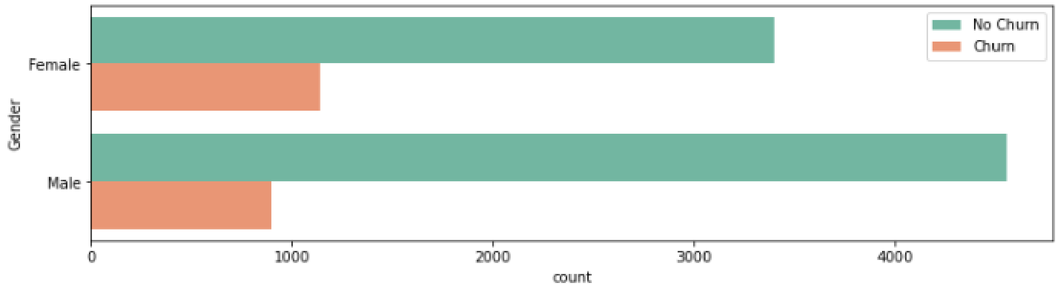
計算屬性間的相關係數，seaborn 繪製出熱力圖

第六題

(一) 改顏色跟改 label ，沒有流失的人在男女都比較多，其中女性客戶流失率明顯較高。

第六題

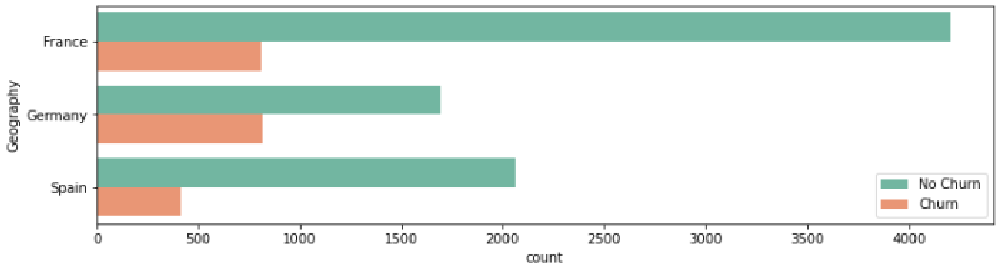
```
fig,ax = plt.subplots(figsize = (12,3))
sns.countplot(data = df,hue = df['Churn'],y = df['Gender'],palette='Set2')
plt.legend( labels=['No Churn', 'Churn'])
plt.show()
```



(二) Germany 客戶流失率最高，France 、Spain 差不多

:

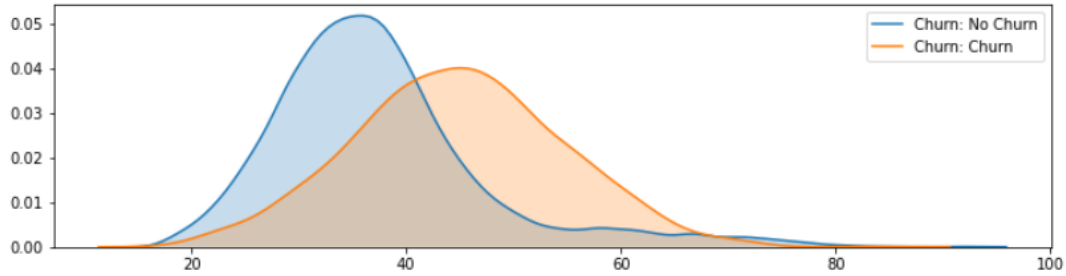
```
fig,ax = plt.subplots(figsize = (12,3))
sns.countplot(data = df,hue = df['Churn'],y = df['Geography'],palette='Set2')
plt.legend( labels=['No Churn', 'Churn'])
plt.show()
```



(三)

可以看出未流失客戶的年齡分布相較流失客戶來的年輕

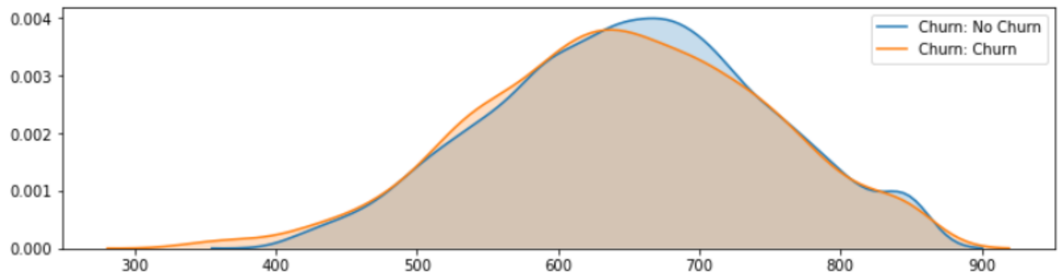
```
fig,ax = plt.subplots(figsize = (12,3))
sns.kdeplot(df.loc[no_churn_df]['Age'], label= 'Churn: No Churn',shade=True)
sns.kdeplot(df.loc[has_churn_df]['Age'], label= 'Churn: Churn',shade=True)

plt.show()
```



(四)

兩者(流失客戶與未流失客戶)CreditScore 分布都差不多

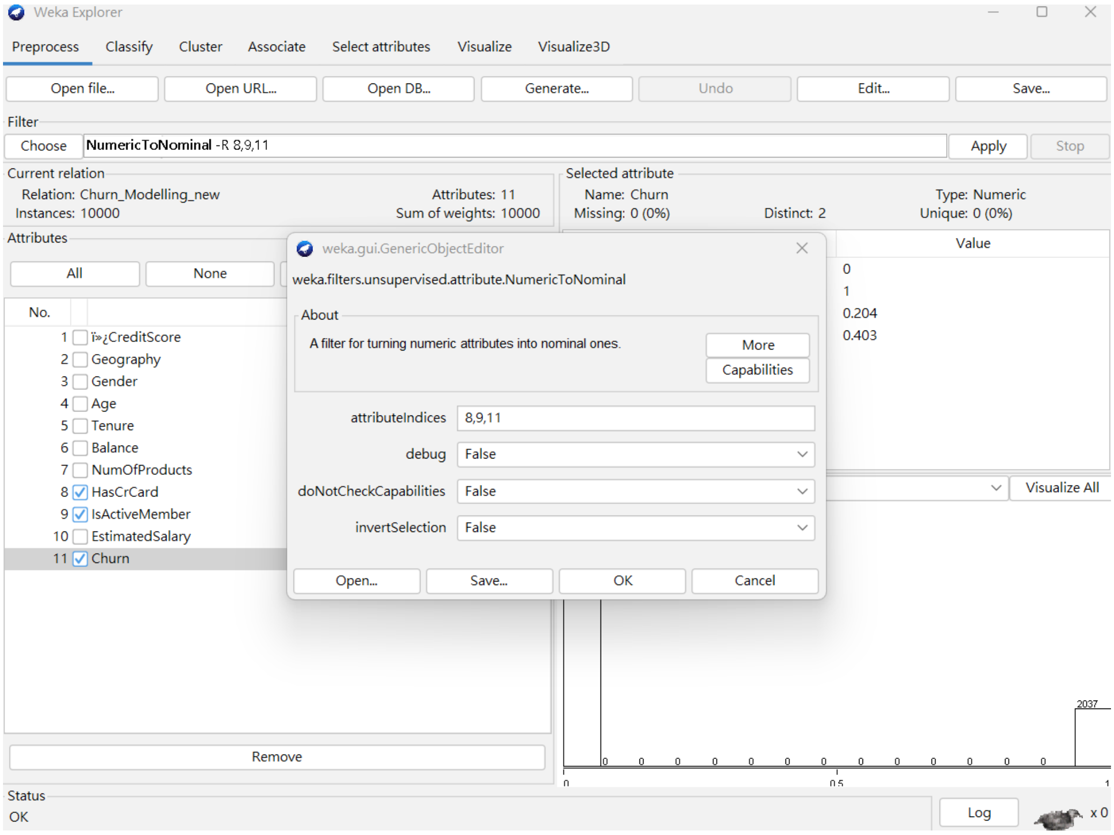
```
fig,ax = plt.subplots(figsize = (12,3))
sns.kdeplot(df.loc[no_churn_df]['CreditScore'], label= 'Churn: No Churn',shade=True)
sns.kdeplot(df.loc[has_churn_df]['CreditScore'], label= 'Churn: Churn',shade=True)

plt.show()
```



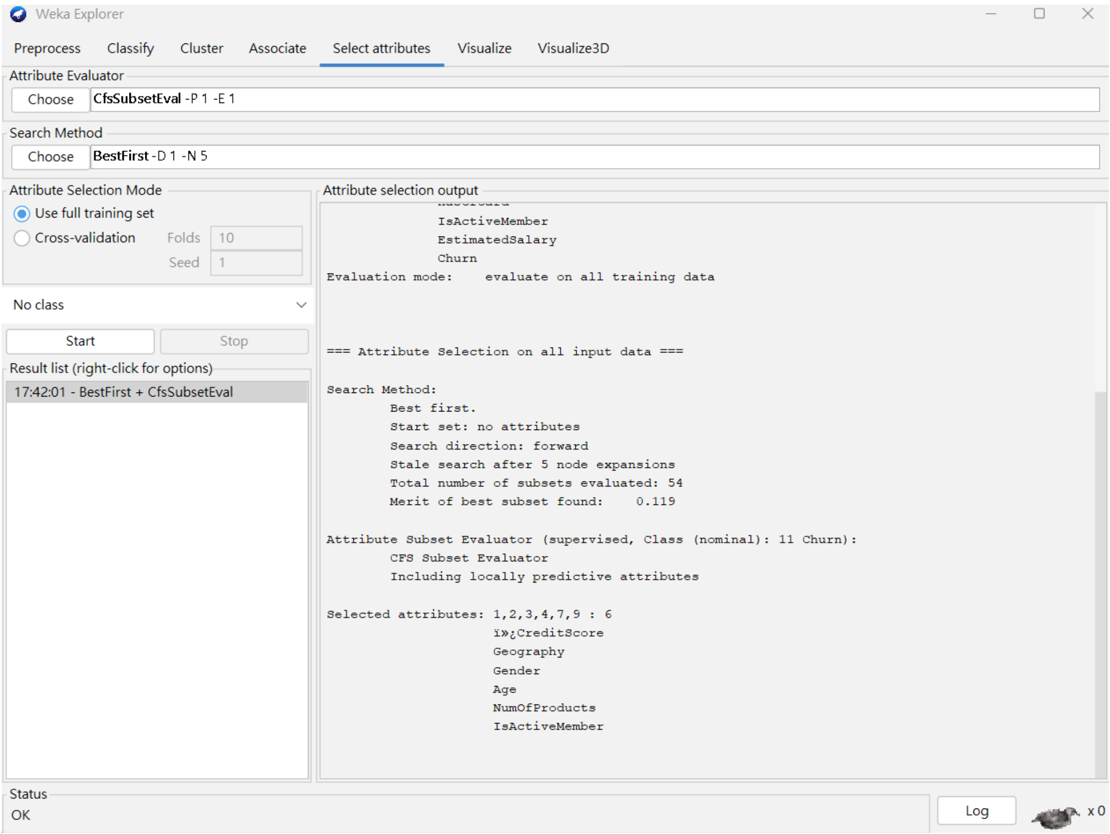
Weka 部分

第七題

(一) 將 HasCrCard, IsActiveMember, Churn 轉成 Nominal 屬性，選 index 8、9、11



(二) 使用 Attribute Selection，以 CfsSubsetEval 及 BestFirst 來篩選屬性



根據結果，使用 CfsSubsetEval 和 BestFirst 算法進行屬性選擇後，選擇的屬性子集包括以下六個屬性：

1.CreditScore

2.Geography

3.Gender

4.Age

5.NumOfProducts

6.IsActiveMember

這些屬性在預測 Churn 有較高的相對重要性。

結論，這些屬性在客戶流失率的預測中扮演著重要的角色，因此在建立模型或進行分析時，應該重點關注這些屬性。