

**ANL252**

**Python for Data Analytics**

**End-of-Course Assessment**

**July 2022 Presentation**

**Name: Lee Pei Shi**

**SUSS PI No: Y2210991**

**ECA Submission Deadline: Monday, 05 September 2022 12:00 pm**

**Code in text:**

# Import Libraries

import pandas as pd

import numpy as np

from sklearn.linear\_model import LinearRegression

from scipy import stats

import seaborn as sns

import matplotlib.pyplot as plt

df = pd.read\_csv('ECA\_Data.csv') #read data

#check df

df

#Question 2

df.duplicated().sum()

#delete duplicate row

df = df.drop\_duplicates().reset\_index()

del df['index']

#check df

df

#check the sum of missing value

df.isnull().sum()

#replace missing value to 0

df[["EDUCATION","MARITAL"]] = df[["EDUCATION","MARITAL"]].fillna(0)

#check data type

df.dtypes

#change of data type

df['R3'] = df['R3'].str.replace(r'\D', '')

df = df.astype({'R3':int})

#check data type

df.dtypes

#drop invalid age

df = df.drop(df[df.AGE.isin([-1, 199])].index)

#check number rows lesser

df

#Question 3

#Bar chart

sns.set(font\_scale=1.4)

df['RATING'].value\_counts().plot(kind='bar', figsize=(7, 6), rot=0)

plt.xticks ([0, 1], ['Good', 'Bad'])

plt.ylabel("Count of People", labelpad=14)

plt.title("Ratings", y=1.02);

#Question 3

#Boxplot

bins = [20, 30, 40, 50, 60, 70, 80]

labels = ['21-30', '31-40', '41-50', '51-60', '61-70', '71-80']

df['agerange'] = pd.cut(df.AGE, bins, labels = labels,include\_lowest = True)

d1 = df[df['agerange'].str.contains('21-30')]

d1 = d1.filter(['INCOME'])

d2 = df[df['agerange'].str.contains('31-40')]

d2 = d2.filter(['INCOME'])

d3 = df[df['agerange'].str.contains('41-50')]

d3 = d3.filter(['INCOME'])

d4 = df[df['agerange'].str.contains('51-60')]

d4 = d4.filter(['INCOME'])

d5 = df[df['agerange'].str.contains('61-70')]

d5 = d5.filter(['INCOME'])

d6 = df[df['agerange'].str.contains('71-80')]

d6 = d6.filter(['INCOME'])

data = np.array([d1, d2, d3, d4, d5, d6], dtype=object)

fig = plt.figure(figsize =(6, 6))

ax = fig.add\_axes([0, 0, 1, 1])

bp = ax.boxplot(data)

box = plt.boxplot(data, #array to be plotted

patch\_artist=True, #fill with color

flierprops={'markeredgecolor': 'None'}, #no marker edger for outliers

showmeans=True, #show the mean

meanprops={"marker":"x","markerfacecolor":"black", "markeredgecolor":"black"})

plt.xlabel('Age Range')

plt.ylabel('Income')

plt.xticks([1, 2, 3, 4, 5, 6], ['21-30', '31-40', '41-50', '51-60', '61-70','71-80'])

plt.show()

#Question 3

#Scatter plot

plt.figure(figsize=(12,6))

plt.title('Limit against Income')

sns.scatterplot(x='LIMIT',

y= 'INCOME',

data=df)

plt.xlabel('Limit')

plt.ylabel('Income')

plt.show()

#Question 3

#Scatter plot

plt.figure(figsize=(12,6))

plt.title('Limit against Balance')

sns.scatterplot(x='LIMIT',

y= 'BALANCE',

data=df)

plt.xlabel('Limit')

plt.ylabel('Balance')

plt.show()

#Question 3

#Bar graph

df['EDUCATION'].value\_counts().plot(kind='barh', figsize=(8, 6))

plt.yticks([0, 1, 2, 3], ['Tertiary','Postgraduate','High School', 'Others'])

plt.xlabel("Count of people", labelpad=14)

plt.ylabel("Education", labelpad=14)

#Question 4

x = df.loc[:,'B1'].values.reshape(-1,1)

y = df.loc[:,'B2'].values.reshape(-1,1)

linear\_regressor = LinearRegression()

linear\_regressor.fit(x, y)

y\_pred = linear\_regressor.predict(x)

plt.scatter(x, y)

plt.plot(x, y\_pred, color='red')

plt.show()

**Question 1**

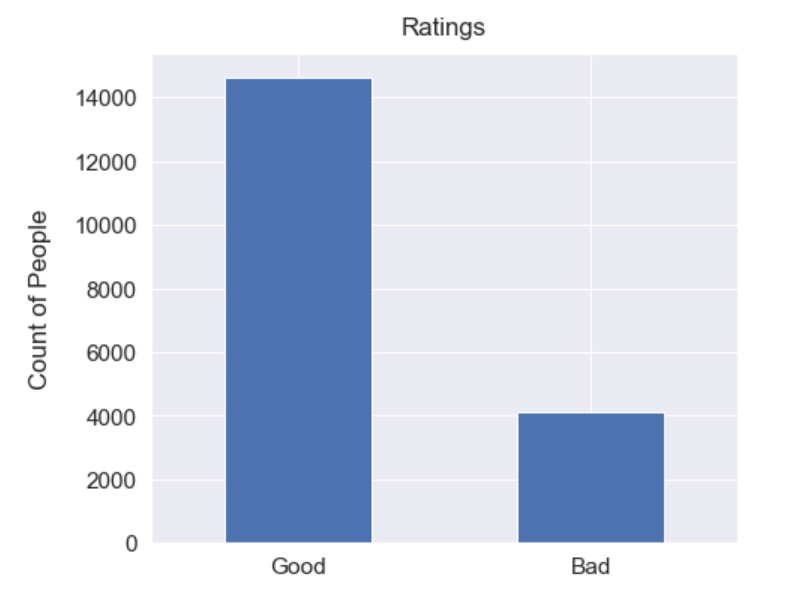
Rating, Gender, Education, Marital, Age and S1-S5 are categorical variables.

Limit, Balance, Income, B1-B5 and R1-R5 are numeric variables.

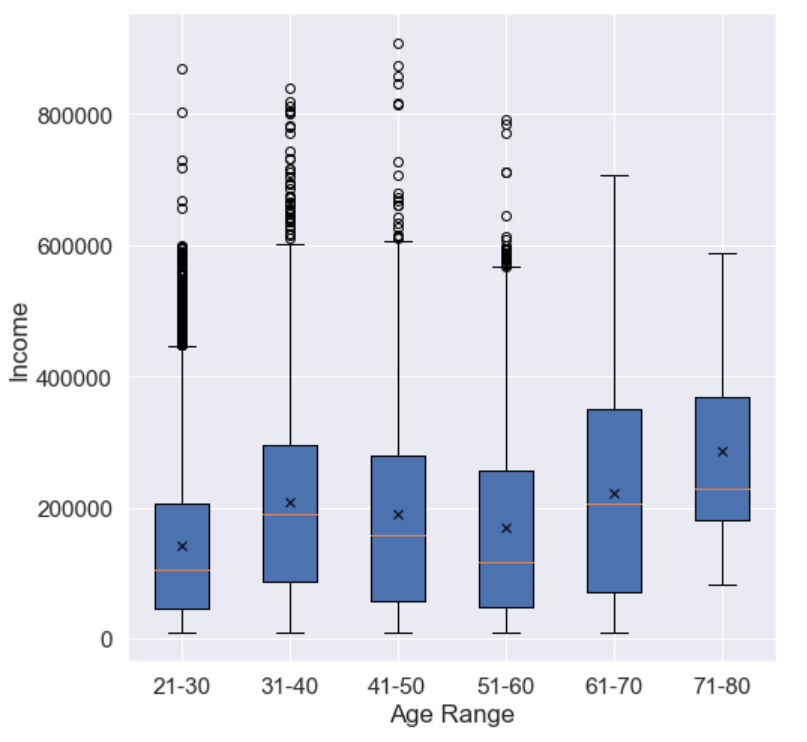
**Question 2**

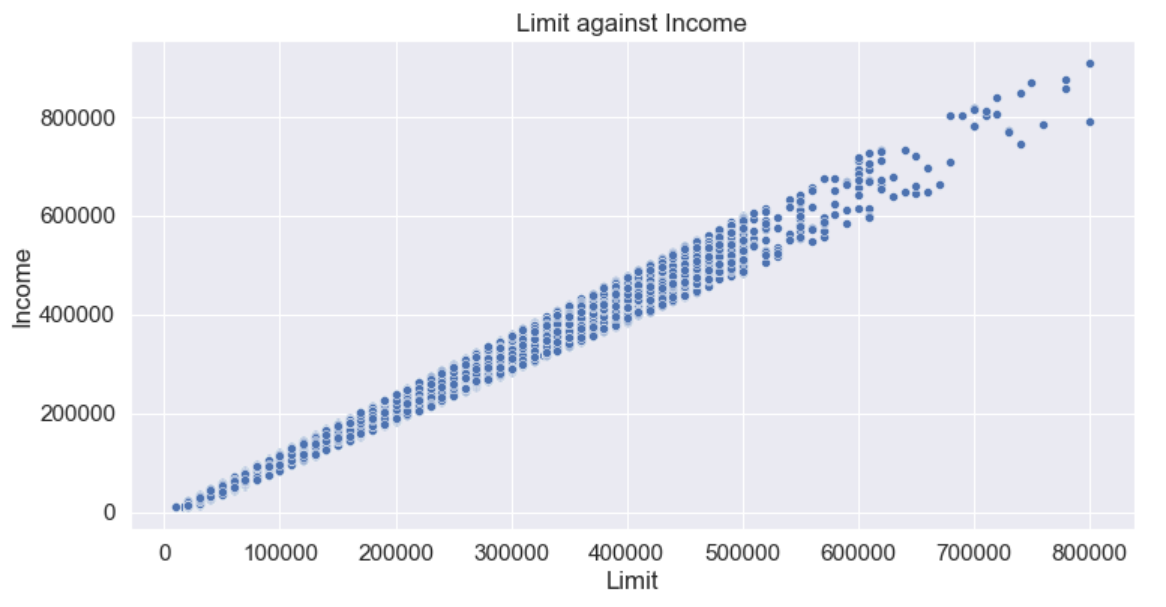
1. There are 3 duplicate rows found. Drop() method was use to remove the duplicate rows. Before there were 18769 rows, after removing duplicates left with 18766 rows which tally the numbers of duplicated rows found.
2. Columns for Education and Marital missing value found. Replace value instead of dropping to keep the data accurate since both columns with the option of others
3. Columns R3 with the different data types (object), change to data to int64
4. Age columns with invalid age (-1,199), remove rows with invalid age, after removing left with 18756 rows of data.

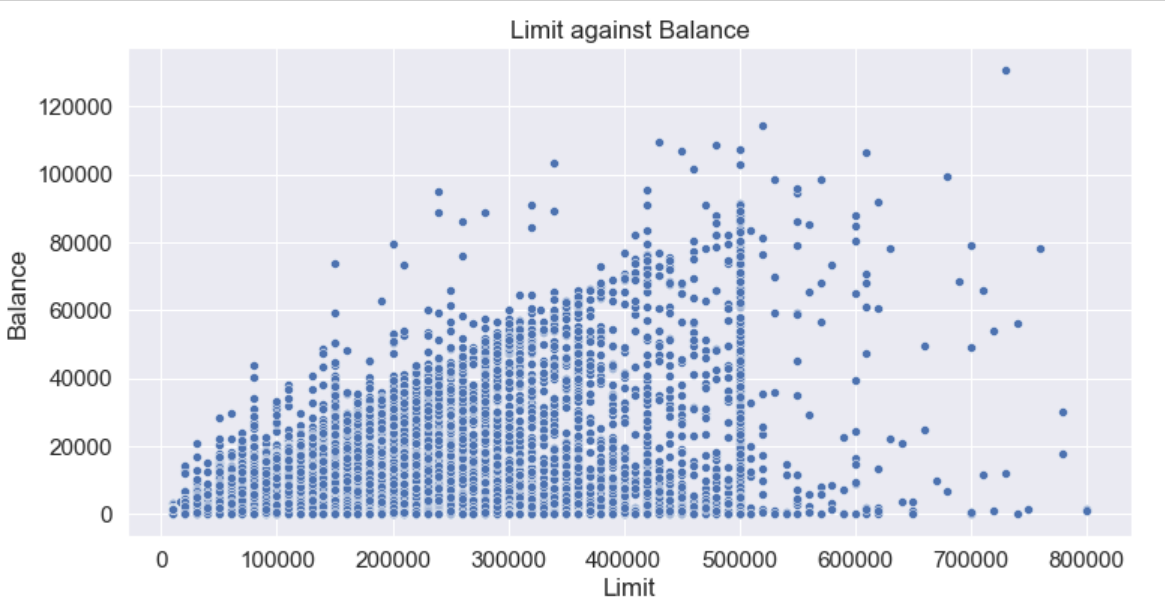
**Question 3**

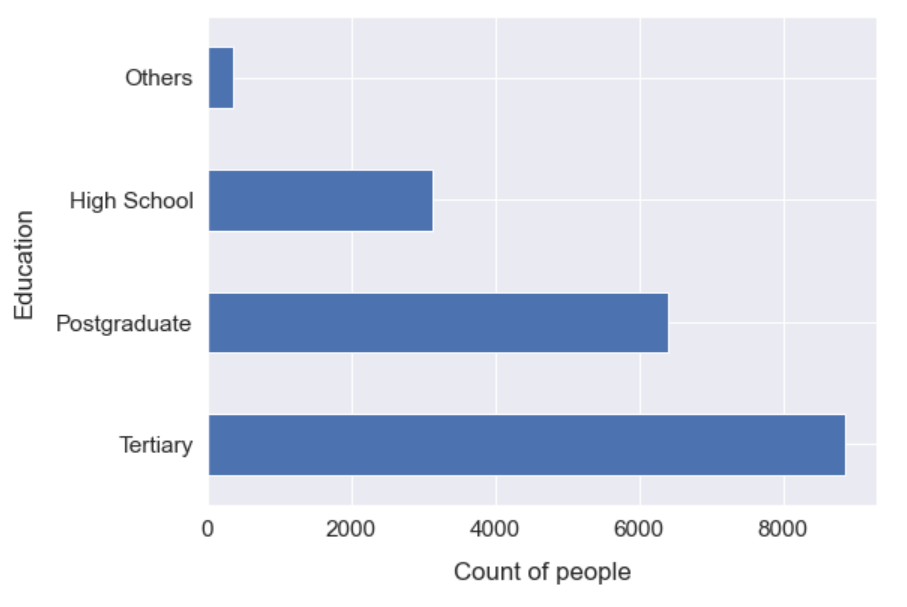


There are more good ratings as compared to bad ratings.

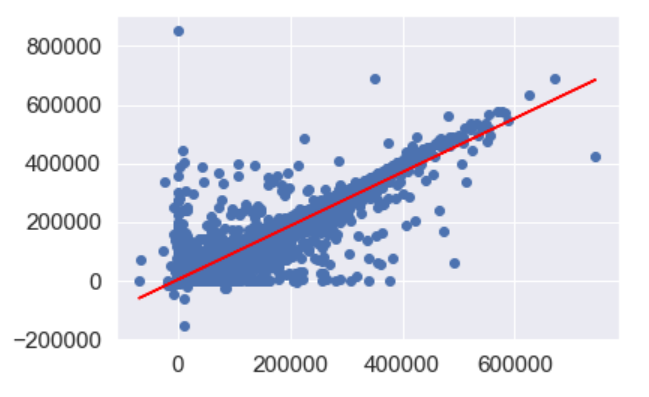








**Question 4**

****