

The codes below imports the required libraries.

import pandas as pd

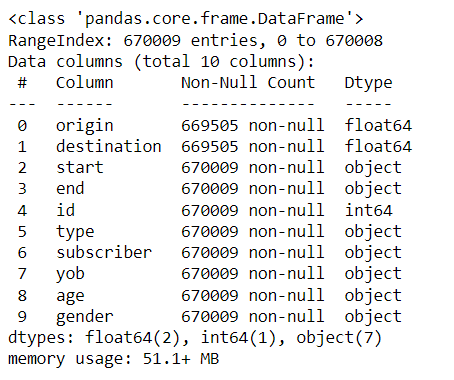
import matplotlib.pyplot as plt

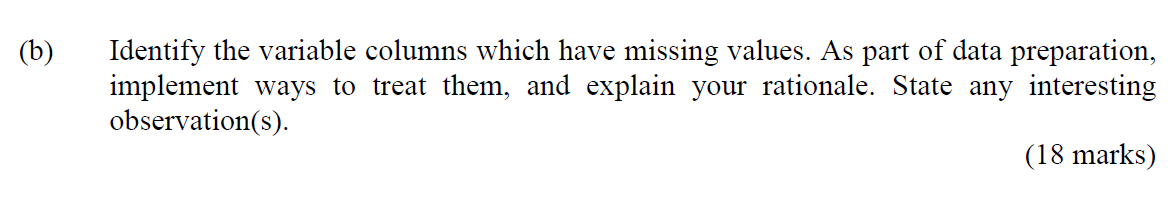
The code belows reads the data.

df = pd.read\_csv('GBA\_data.csv')

Now I will check the dtypes of my data using the code below. I realised that ‘origin’, ‘destination’, ‘id’are not object, which is string. Hence, I can conclude that these metioned variables do not contain ‘-‘, ‘—‘, ‘?’. Therefore, for part b), I will replace each element inside with ‘others’.

df.info()





#Qns 1b)

#Now I put all the variables that requires replacing inside a list

variables = ['start', 'end', 'type', 'subscriber', 'yob', 'age', 'gender']

#Now I place missing things into a variable called 'missing'

missing = ['-', '--', '?']

#I now initate a list called columns\_with\_missing

columns\_with\_missing = []

for variable in variables:

for i in df[variable]:

#To ensure it is converted into string so I can use len() function, and the len() function ensures that it is single character, as

#the variables start and end contains '-'

if (len(str(i)) == 1) or (len(str(i)) == 2):

if i in missing:

#If the variable that contains '-', '--', '?' are already in list, it does not require appending

if (variable in columns\_with\_missing) == False:

columns\_with\_missing.append(variable)Therefore, the variables with missing columns are identified below via the following code and diagram:

print(columns\_with\_missing)



The interesting observation here is that the date also contains ‘-‘, which may be mistaken as missing value. That is why I am required to check for the length first. Also, I can easily identify which variables definitely do not have any missing values because it would then be object instead of integer or float.