

**ANL252**

**Python for Data Analytics**

**Group-Based Assignment (GBA01)**

**July 2022 Presentation**

**Submitted By:**

|  |  |
| --- | --- |
| **Name** | **PI No.** |
| Tan Boon May (Chen Wenmei) | W2210814 |
| Gan Ke Yi | B1972108 |
| Chan Zhi Hao | N2070913 |
| Lee Zhu | Q2110768 |

**Tutorial Group: T09 (Group 3)**

**Instructor’s Name: Dr. Kumar Munish**

**Submission Date: 28/08/2022**

Table of Contents

**Question 1 (a) …………………………..….……………………………………………….** 3-4

**Question 1 (b) ………………………...……………….………………….………………...** 5-7

**Question 1 (c) …………………………......………….…………....…………………….......** 8

**Question 1 (d) ………………………...……………….………………….……………….....** 9

**Question 1 (e) …………………………......………….…………....…………………….....**10-16

**Question 1a)**

|  |  |
| --- | --- |
| In [1]: | ### Import file and read the dataset in as a Pandas dataframe.  import pandas as pd  data = pd.read\_csv ("GBA\_data.csv")  data.head() |
| Out [1]: |  |
| In [2]: | ### Extracted the missing values ‘--’ from columns ‘yob’ & ‘age’ and have discovered that the missing values from both categories are from the same 10 sets of data  ### Filter the missing values under "yob"  group = data.groupby('yob')  group.get\_group('--') |
| Out [2]: |  |
| In [3]: | ### Filter the missing values under "age"  group = data.groupby('age')  group.get\_group('--') |
| Out[3]: |  |
| In [4]: | ### Extracted a total of 10 missing values ‘?’ from column ‘type’ and 3 missing values ‘-’ from column ‘gender’  ### Filter the missing values under "type"  group = data.groupby('type')  group.get\_group('?') |
| Out[4]: |  |
| In [5]: | ### Filter the missing values under "gender"  group = data.groupby('gender')  group.get\_group('-') |
| Out[5]: |  |

**Question 1b)**

|  |  |
| --- | --- |
| In [1]: | ### Imported file and read the dataset in as a Pandas dataframe.  import pandas as pd  data = pd.read\_csv("GBA\_data.csv")  data.head() |
| Out[1]: |  |
| In [2]: | ### To obtain summarized information. The summary clearly indicated that there are a total of 670009 sets of data but columns ‘origin’ & ‘destination ' only accounted for 669505.  ### Obtain summary  data.info() |
| Out[2]: |  |
| In [3]: | ### From the summary it was deduced that columns ‘origin’ & ‘destination ' both have missing values amounting to the same 504 sets of data.  ### Identify missing data  data.isna().sum() |
| Out[3]: |  |
| In [4]: | ### Sorted the data in ascending order by column ‘origin’ for analysis study but no irregularity found.  #To sort the data in ascending order by 'origin'  data.sort\_values(by=['origin'],ascending=True) |
| Out[4]: |  |
| In [5]: | ### Extracted the 504 sets of missing data from columns ‘origin’ & ‘destination ' for further analysis. We have discovered that the missing values’ id’ were ranging from 3767 to 4297 after we sorted them in ascending order. Also, from the data shown, the same ‘id’ is repeated several times in the records and is shared among different profiles.  #Extract 504 sets of data with missing data  data[data.isna().any(axis=1)]  data\_Nan = data[data.isna().any(axis=1)]  data\_Nan.sort\_values(by=['id'],ascending=True) |
| Out[5]: |  |

**Conclusion**

* As informed, the provided dataset is that of a shared mobility service, with the likes of devices for, example - bikes, electric bikes, and electric scooters.

* 504 sets of missing values for ‘origin’ & ‘destination’ are ranging from id no 3767 to 4297 after sorting in ascending order. Possibly could be due to some technical glitch during the extraction or exportation of data
* The 504 sets of missing values as compared to the total no of 670009, is only 0.075%. Hence, we strongly believe the missing values will not form a significant impact on the analyzed results.

Words count: 96

**Question 1c)**

In order to solve the problem with the data's quality, we will first explain what the overall goal of this project is. Our goal is to determine whether or not there is a connection between the customer profile type and the age of the client, as well as whether or not there is a link between the customer profile type and the subscription type of the customer.

Hence, the first thing that we would be improving the data is by grouping the age, analysis can be done better, and it can aid in visualization in the later part of our project. We would create a new column named “age group range”. Currently, the age data we have is from 20 to 71, hence we propose to create 6 bins each with an interval of 11, starting from 20 to 31, 32 to 42, 43 to 53, 54 to 64 and lastly 65 to 75. This is also to determine whether or not there was a pattern of behavior among the various age groups in terms of the kinds of preferences they have and the degree to which they are interested in subscribing to a newsletter. This may assist the organization get a deeper understanding of their clientele, allowing them to better use this knowledge for, for instance, marketing purposes.

Secondly, we would be removing the variables that would not be needed to reach our objective. We would be removing variables like origin, destination, start, end, id ,yob and gender. By removing variables that we do not need could help to increase the processing speed of the program making it more efficient.

Lastly, we would need to create dummy variables for types we could create 2 dummy variables, instead of having 3 categories which are “Ad-Hoc”,” Regular” and “Concession”, we can create 2 dummy variables that are just “Ad-Hoc” and “Concession” with 0 being assign to “Ad-Hoc” and 1 being assign to “Concession”. Simplifying the categories could aid in the algorithms we might be using for our visualization later.

Words count: 338

**Question 1d)**

|  |  |
| --- | --- |
| In [1] | ### Install and import all the related file  import pandas as pd  from datetime import datetime |
| In [2] | ### Retrieve the excel file  data=pd.read\_csv("GBA\_data.csv") |
| In [3] | ### Read the data from the excel file to check is the correct file  data.info |
| Out [3] |  |
| In [4] | ### define function\_name (argument1, argument2,....):  def commuterstimehighest():  ### first instruction - convert the datetime to 12 hours format data["start\_hour"]=pd.to\_datetime(data["start"]).dt.strftime("%#I %p")  ### second instruction - find the more frequency hours occur from data["start"]  commuters\_freq= data["start\_hour"].groupby(data["start\_hour"]).count().reset\_index(name="commuter\_no").sort\_values(by=["commuter\_no"],ascending=False)  ### third instruction - display the head in highest\_time and print  Highest\_time = commuters\_freq["start\_hour"].head(1)  print(f"The highest number of commuters starts their journey at {Highest\_time.iloc[0]}.")  commuterstimehighest() |
| Out [4] | The highest number of commuters starts their journey at 5 PM. |

**Question 1e)**

|  |  |
| --- | --- |
| **Stacked Bar Chart (Customer profile type Vs Agegroup)** | |
| In [1] | ### Install and import all the related file  import pandas as pd  from datetime import datetime  import matplotlib.pyplot as plt  import numpy as np |
| In [2] | ### Retrieve the excel file  df=pd.read\_csv("GBA\_data.csv") |
| In [3] | ### Read the data from the excel file to check is the correct file  df.head() |
| Out [3] |  |
| In [4] | ### Check the how many rows and column  df.shape |
| Out [4] | (670009, 10) |
| In [5] | ### Remove the "--" missing value  df[df["age"]=="--"].index  df1 = df.drop(index=df[df["age"]=="--"].index) |
| In [6] | ### Remove the "?" missing value in type header  df1[df1["type"]=="?"].index  df2 = df1.drop(index=df1[df1["type"]=="?"].index) |
| In [7] | ### Check the missing value row have been removed  df2.shape |
| Out [7] | (669989, 10) |
| In [8] | ### Check the info to determine the Dtype is int for age  df2.info() |
| Out [8] |  |
| In [9] | ### Convert the age dtypes to int  df2["age"] = df2["age"].astype(str).astype(int)  print(df2.dtypes) |
| Out [9] |  |
| In [10] | ### Use discretisation to categorise the age group  bins = [18, 30, 40, 50, 60, 70, 120]  labels = ['18-29', '30-39', '40-49', '50-59', '60-69', '70+']  df2['agegroup'] = pd.cut(df2["age"], bins, labels = labels)  df2.head() |
| Out [10] |  |
| In [11] | ### Use groupby method can be any method for the calculation of the aggregated statistics  df3=df2.groupby(["agegroup","type"])["agegroup"].count()  print(df3) |
| Out [11] |  |
| In [12] | ### Use the stacked bar chart for this data analysis  data = pd.DataFrame([['18-29', 22668, 0, 142605], ['30-39', 30034, 0, 263433], ['40-49', 9964, 0, 114423], ['50-59', 4402, 33378, 28699],['60-69', 1117, 13205, 5410],['70+', 13, 0, 638]], columns=['X-Axis', 'Ad-Hoc', 'Concession', 'Regular']) |
| In [13] | ### Result  data.plot(x='X-Axis', kind='bar', stacked=True, title='Agegroup and customer profile type stacked bar', xlabel="Agegroup", ylabel="Customer profile type") |
| Out [13] |  |
| **Bar Chart (Number of commuter VS Type)** | |
| In [14]: | ### Install and import all the related file  import matplotlib.pyplot as plt  import pandas as pd |
| In [15]: | ### Retrieve the excel file  df = pd.read\_csv('GBA\_data.csv', dtype='unicode')  df.head() |
| Out [15]: |  |
| In [16]: | ### Remove the "?" missing value in type header  df[df["type"]=="?"].index  df1 = df.drop(index=df[df["type"]=="?"].index)  df1.shape |
| Out [16]: | (669999, 10) |
| In [17]: | ### Barchart that is based on type and Result  df1.groupby(["type"]).size().plot.bar()  plt.xlabel("Type")  plt.ylabel("Number of commuters")  plt.title("Number of commuter for each type Bar chart") |
| Out [17]: |  |
| **Pie Chart (Type and Subscribe)** | |
| In [18]: | ### Install and import all the related file  import matplotlib.pyplot as plt  import numpy as np |
| In [19]: | ### Retrieve the excel file  df=pd.read\_csv("GBA\_data.csv") |
| In [20]: | ### Read the data from the excel file to check is the correct file  df.head() |
| Out [20]: |  |
| In [21]: | ### Remove the "?" missing value in type header  df[df["type"]=="?"].index  df1 = df.drop(index=df[df["type"]=="?"].index)  df1.shape |
| Out [21]: | (669999, 10) |
| In [22]: | ### Use groupby method can be any method for the calculation of the aggregated statistics  df2=df1.groupby(["type","subscriber"])["type"].count()  print(df2) |
| Out [22]: |  |
| In [23]: | ### Use python to calculate the percentage  print(((68198/(68198+40478+6106+500109+55108))\*100), ((40478/(68198+40478+6106+500109+55108))\*100), ((6106/(68198+40478+6106+500109+55108))\*100), ((500109/(68198+40478+6106+500109+55108))\*100), ((55108/(68198+40478+6106+500109+55108))\*100)) |
| Out [23]: | 10.178821162419645 6.0415015544799315 0.9113446437979758 74.64324573618767 8.22508690311478 |
| In [24]: | ### Create pie chart with Pyplot and Result  y = np.array([10.2, 6, 1, 75.6, 8.2])  mylabels = ["Ad-Hoc (No)", "Concession (No)", "Concession (Yes)", "Regular (No)", "Regular (Yes)"]  plt.pie(y, startangle = 90)  plt.title ("Type and Subscribe")  plt.legend(labels = mylabels, loc ="best")  plt.show() |
| Out [24]: |  |

**Summary:**

1. **Stacked Bar Chart (Customer profile type Vs Agegroup)**

In this stacked bar chart, we are comparing age group vs customer profile type to analyse the data. The result showed 30 to 39 age group have the highest number of people who are the regular type. In this age group, more of them are working adults so they will take public transport to work. The data showed 60 to 69 age group have the highest number of people who are the concession type. In this age group, they will have senior citizen concession cards to enjoy the discount rate. In this dataset, we have eliminated 10 users that do not provide their age and year of birth. The highest number of commuters are age group 30 to 39 years old and the lowest number of commuters are age group 70+. This is one of the interesting insights found in the data. This dataset can’t be accurate because they are different numbers of commuters in each age group category.

## **Bar Chart (Number of commuter VS Type)**

This is a simple bar chart, we want to know what type of customer profile is the more popular and the more popular than the regular type. The reason is the regular price is somehow cheaper than the ad-hoc price but not the concessions type. This is one of the interesting insights found in the data. Ad-hoc price is like customers whose EZ-link card does not have money then they pay by cash or tourist. Concessions are for students and the elderly who can enjoy the discount. The dataset the younger is 20 years and based on the stacked bar chart it showed that 50 to 69 years old have a higher concession than the regular type. For our age group 20-29 years old, we are the regular type. In the economic factor, the regular type is more affordable than the ad-hoc or concession types.

## **Pie Chart (Type and Subscribe)**

For the pie chart, we are looking at the type and number of subscriptions. There is a higher number of customers who have not subscribed to the regular type. This is one of the interesting insights found in the data. The higher rate of customers is regular, hence the subscriber is low. Our group think that the service subscriber rate is low due to the customer flexibility, we found out that this dataset has multiple repeat ID but a different person which mean this dataset has a data quality issue.

To improve the data accuracy, there are some more we can improve such as using 1 ID for each person and each age group need to have the same number of commuters.

Words count: 446

Total words count: 880

**The End**