CSC 535/635 Data Mining

## Assignment-1 Report

### Submitted to:

### Dr. Jamil Saquer

### Author:

### Khaled Hossain

**Assignment-1 Report**

**Introduction**

For this project, I used two different data sets to analyze and answer questions. The first data set used in part-1 was provided by the instructor. It contained information on tips received by waitresses of a restaurant over a period of time in 1990. The attributes of the dataset are: -

1. total\_bill: cost of the meal including tax, in US dollars
2. tip: gratuity in US dollars
3. sex: sex of person paying for the meal
4. smoker: Is a smoker in the party? With possible values no and yes
5. day: day of the meal with possible values Thursday, Friday, Saturday, and Sunday
6. time: time of meal with possible values Dinner and Lunch
7. size: number of people in the party

For the second part of the project, I used a dataset from [https://www.kaggle.com/imdevskp/corona-virus-report](https://www.kaggle.com/imdevskp/corona-virus-report%20called%20%20%20%20%20%20%20COVID-19)  called Covid-19 Dataset, which contained information on number of Confirmed, deaths and recovered cases every day across the globe. I chose this dataset to look at the current situation of Corona around the world in different regions. I wanted to find out which countries and regions of the world were affected the most. The countries with the highest number of confirmed cases/ deaths. I also wanted to see which countries are recovering faster. I also wanted to find out why some countries were affected more than others and why some were able to recover faster. The attributes of the data set were: -

1. Country/ Region
2. Number Confirmed
3. Number of Deaths
4. Number of Recovered
5. Active cases
6. New Cases
7. New Deaths
8. New Recovered
9. Deaths / 100 cases
10. Recovered /100 cases

**Background**

For this assignment no algorithm was used to answer the questions. I came to the conclusion by observing trends in the data from the scatter plots, bar charts, pie charts and tables. Using the notebook and the matplotlib library I was able to visualize the data, which allowed me to come to conclusions.

**Implementation and Results**

After analyzing the first data set I was able to answer the following questions: -

1. How much tip do customers usually give on average?

Ans:- The customers give an average of 2.99 ~ $3 of tips on average

1. Do male customers usually give more tip than female customers?

Ans:- Male customers on average give more than female customers. Males give $3.08, females give $2.83.

1. What day(s) of the week is the restaurant the busiest?

Ans:- The busiest days are the weekends. Specially on Saturday.

1. Do customers tend to give more tips on certain days?

Ans:- Customers give more tips on Sundays.

By analyzing the second dataset I was able to answer these questions: -

1. What is the percentage of mean number of confirmed cases across the world?

Ans:- The percentage of the mean number of confirmed cases around the world is shown in the PI chart. From the PI chart we can see that highest number of confirmed cases can be found in the Americas : -

Chart, pie chart

Description automatically generatedA picture containing diagram

Description automatically generated

1. What is the mean number of recovered cases across the world?

Ans:- We can see that the region of south-east Asia has been able to recover more even though they had the second highest confirmed cases.

Chart, pie chart

Description automatically generatedA picture containing diagram

Description automatically generated

1. Why do some regions have more cases of Covid than others?

Ans:- The main reason that some regions have Covid more than others is because of the age of the population. According to the U.S Census Bureau more than 40.3 million U.S residents are older than 65 years (Bureau, U, 2020). A lot of countries in South east Asia, like Singapore have an aging population as well (Team T, 2020).

**Conclusion**

From the analysis of the results in the second data set, we can come to the conclusion that the older population is more at risk and has a higher chance of getting affected by Covid-19. The two regions of the world with the highest number of cases was the Americas and South-East Asia. Both of the regions have an again population. Some other factors that we need to be aware of how much testing was being done, and the population of the whole region. Lastly, some countries were able to recover faster because they followed the rules and regulations set out by the government. On the other hand, the countries that didn’t follow the restrictions and avoided wearing masks are having a harder time recovering from the pandemic.

**References**

Bureau, U. (2020, August 19). National Senior Citizens Day: August 21, 2020. Retrieved September 25, 2020, from https://www.census.gov/newsroom/stories/senior-citizens-day.html

Kp, D. (2020, August 07). COVID-19 Dataset. Retrieved September 25, 2020, from https://www.kaggle.com/imdevskp/corona-virus-report

Team, T. (2020, February 16). Southeast Asia's rapidly ageing population  . Retrieved September 25, 2020, from https://theaseanpost.com/article/southeast-asias-rapidly-ageing-population

**Code**

The code should be run using anaconda Jupyter notebook

# In[1]:

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

# In[2]:

# 1 - load data

tips = pd.read\_csv('tips.csv')

tips

# In[26]:

# 2 names of the columns

column\_names = tips.columns

print(column\_names)

# In[27]:

# 3 Rename columns

tips.rename(columns={'time':'meal', 'size':'party size'}, inplace = True)

print(tips.columns)

# In[11]:

# 4 View parts of data

tips.head()

# In[23]:

# 5 retrive info from 2nd and third row using loc

tips.loc[[1,2]]

# In[25]:

# 5 retrive info using slicing notation

tips.loc[1:2]

# In[29]:

# 6 retrive information from second and third columns and use head

bills\_tips = tips[['total\_bill', 'tip']]

bills\_tips.head(3)

# In[42]:

# 7 Get the information stored in the cell at the intersection of the second row and third column (PROBLEM)

second\_row = tips.loc[1]

#print(second\_row)

third\_col = tips[['sex']]

#print(third\_col)

second\_row['sex']

# In[48]:

# 8 Find the number of ovservations in the dataset (PROBLEM)

tips.count()

# In[58]:

# 9 Use describe to get basic statistics

total\_bill = tips['total\_bill'].describe()

tip = tips['tip'].describe()

party\_size = tips['party\_size'].describe()

stats = [total\_bill, tip, party\_size]

pd.DataFrame(stats)

# In[64]:

# 10 Use describe to get statistics on all values

tips.describe(include='all')

# In[65]:

# 11 How much tip do customers usually give on average?

tips.tip.mean()

# In[66]:

# 12 Basics of tip

tips.tip.describe()

# In[68]:

# 13 Boxplot of the tip column

plt.boxplot(tips.tip)

# In[6]:

# 14 Retrive rows where tip ammount is >=6

tips[tips.tip >= 6]

# From the table we can see male customers give more tips than females

# In[15]:

# 15 - Groupby function to find the average tip of male and female

tips.groupby(['sex']).mean()

# In[26]:

# 16 - df name males and a random sample of 10

males = tips[tips["sex"] == 'Male']

males.sample(10)

# In[27]:

# 17 DF called females

females = tips[tips["sex"] == 'Female']

females.sample(frac = 0.1, replace = True)

# In[35]:

# 18 number of male and number of female

print("Females:-", len(females.index))

print("Males:-", len(males.index))

# In[41]:

# 19 Bar chart

f = len(females.index)

m = len(males.index)

fig = plt.figure()

plt.bar(f, m)

plt.xlabel("Number of females")

plt.ylabel("Number of males")

plt.title("Comparing the number of customers")

# In[64]:

# 20 scater plot of the tips given by male customers and a scatter plot of the tips

plt.plot(males.tip, 'o', marker='P', label = "male", color="red")

plt.plot(females.tip, 'o', marker='\*', label = "female", color = "blue")

plt.legend()

# In[69]:

# 21 two box plots to show tips per sex

data = [ females.tip, males.tip]

plt.boxplot(data, labels=['Female', 'Male'])

plt.xlabel("Gender")

plt.ylabel("Tip")

plt.title("Boxplot of Tips by Gender")

# In[70]:

# 22 Describe function on day column

tips.day.describe()

# most busy on Saturday

# In[151]:

# 23 groupby function to form groups based on the days of the week

grouped =tips['size'].groupby(tips['day'])

table = pd.DataFrame(grouped.count())

table

# In[158]:

# 24 Line plot that shows number of customers served everyday

xValues = [62, 19, 87, 76]

plt.xticks([0.0,1.0,2.0,3.0],["Thursday",'Friday',"Saturday","Sunday"])

plt.grid(True)

plt.plot(xValues)

# customers give more tips on Saturdays

# In[15]:

# 25 groupby function to find the average tip per day

grouped = tips['tip'].groupby(tips['day'])

meanValues = pd.DataFrame(grouped.mean())

fri = meanValues.loc['Fri']

sat = meanValues.loc['Sat']

thu = meanValues.loc['Thur']

sun = meanValues.loc['Sun']

pd.DataFrame([fri, sat, sun, thu])

# In[98]:

# 26 Tips given on different days

friday = tips[tips["day"] == 'Fri']

saturday = tips[tips["day"] == 'Sat']

sunday = tips[tips["day"] == 'Sun']

thursday = tips[tips["day"] == 'Thur']

# plt.figure(figsize=(8, 8))

fig, axs = plt.subplots(2, 2)

fig.subplots\_adjust(left=0.125, right=1, bottom=-0.5, top=0.78, wspace=0.2, hspace = 0.2)

fig.suptitle("Tips Given On Different Days", fontdict={'color': 'red'}, fontsize = 25)

ax1 = fig.add\_subplot(221)

plt.plot(thursday.tip)

plt.plot(thu, marker="\*")

ax1.title.set\_text('Thursday')

plt.axis('off')

ax2 = fig.add\_subplot(222)

plt.plot(friday.tip)

plt.plot(fri, marker="\*")

ax2.title.set\_text('Friday')

plt.axis('off')

ax3 = fig.add\_subplot(223)

plt.plot(saturday.tip)

plt.plot(sat, marker="\*")

ax3.title.set\_text('Saturday')

plt.axis('off')

ax4 = fig.add\_subplot(224)

plt.plot(sunday.tip)

plt.plot(sun, marker="\*")

ax4.title.set\_text('Sunday')

# plt.tight\_layout()

# plt.axis('off')

# In[105]:

# 27 pie chart

data = [friday.tip.sum(), saturday.tip.sum(), sunday.tip.sum(), thursday.tip.sum()]

labels = ['Fri', 'Sat', 'Sun', 'Thur']

plt.pie(data, labels=labels, autopct = '%0.2f')

plt.axis("equal")

# In[4]:

# Part-2 (non-trivial question)

cases = pd.read\_csv('country\_wise\_latest.csv')

cases.head(5)

# In[39]:

df3 = cases[['Country/Region','Confirmed', 'New cases', 'Deaths / 100 Cases', 'New recovered', 'WHO Region']]

# In[40]:

easternMedi = df3[df3["WHO Region"] == 'Eastern Mediterranean']

europe = df3[df3["WHO Region"] == 'Europe']

africa = df3[df3["WHO Region"] == 'Africa']

americas = df3[df3["WHO Region"] == 'Americas']

southEAsia = df3[df3["WHO Region"] == 'South-East Asia']

westernPacific = df3[df3["WHO Region"] == 'Western Pacific']

europe.describe()

# In[41]:

e = europe['Confirmed'].mean()

em = easternMedi['Confirmed'].mean()

a = africa['Confirmed'].mean()

am = americas['Confirmed'].mean()

sea = southEAsia['Confirmed'].mean()

wp = westernPacific['Confirmed'].mean()

data = [e, em, a, am, sea, wp]

labels = ['Europe', 'Eastern Medi.','Africa','Americas', 'South-East Asia', 'West Pacific']

plt.pie(data, autopct = '%0.2f')

plt.axis("equal")

plt.legend(labels)

# In[42]:

e = europe['New recovered'].mean()

em = easternMedi['New recovered'].mean()

a = africa['New recovered'].mean()

am = americas['New recovered'].mean()

sea = southEAsia['New recovered'].mean()

wp = westernPacific['New recovered'].mean()

data = [e, em, a, am, sea, wp]

labels = ['Europe', 'Eastern Medi.','Africa','Americas', 'South-East Asia', 'West Pacific']

plt.pie(data, autopct = '%0.2f')

plt.axis("equal")

plt.legend(labels)

# In[45]:

e = europe['Deaths / 100 Cases'].mean()

em = easternMedi['Deaths / 100 Cases'].mean()

a = africa['Deaths / 100 Cases'].mean()

am = americas['Deaths / 100 Cases'].mean()

sea = southEAsia['Deaths / 100 Cases'].mean()

wp = westernPacific['Deaths / 100 Cases'].mean()

data = [e, em, a, am, sea, wp]

labels = ['Europe', 'Eastern Medi.','Africa','Americas', 'South-East Asia', 'West Pacific']

plt.pie(data, autopct = '%0.2f')

plt.axis("equal")

plt.legend(labels)