**INT 7623**

**Data Science for Business**

01/29/2024

Due 02/11/2024 11:59 PM

Assignment 1

(30 points)

**Note: Please refer to the syllabus for the submission format.**

Data visualization is an important component of the entire data science workflow. Learning how to leverage a software tool to visualize data will enable you as a data scientist to understand the structure of the data, identify potential issues, extract information, better understand the data, and make more effective decisions. In this assignment you will practice how to use **Python** to create fundamental charts using small datasets.

In each question, you have to include the following:

* **Python code**
  + copy & paste to this document but be sure about indentation.
  + Include Python file in your submission (filename.py). Comment your code. (4 points)
* The **chart** you created. Make sure to format the chart by adding labels, change colors, etc. mention what format you did. (4 points)
* Discuss what **findings** you discover after you visualize the data. For example, identify any potential pattern/ issue in the dataset or suggest solutions, etc. (2 points)

**Exercise #1 (10 points):**

1. Assume that you have the following raw dataset about some patients’ heights in centimeters:

[143,144,141,141,141,145,149,147,152,154,152,151,154,153,

154,150,153,154,156,158,159,158,159,155,158,157,158,159,

159,158,158,159,156,156,157,156,159,155,157,162,160,160,

164,160,164,160,163,163,163,164,163,164,162,162,163,161,

160,163,162,162,164,164,160,161,163,166,166,168,166,168,

167,167,167,168,168,168,165,169,167,166,166,169,166,168,

168,168,169,168,166,168,165,169,167,169,167,168,167,165,

169,168,165,167,167,165,166,165,166,165,167,165,168,165,

166,169,168,169,167,167,170,173,173,172,171,174,174,174,

173,172,172,174,171,174,173,173,170,172,174,171,172,174,

170,170,174,174,170,170,174,173,171,174,170,170,170,174,

170,173,170,174,173,171,173,171,170,174,171,174,172,173,

170,172,173,174,170,171,170,170,172,173,172,171,174,172,

171,179,178,179,179,176,175,175,176,177,177,177,177,179,

178,178,178,175,175, 183,184,182,182,184,183,180,182,180,

182,180,183,182,180,180,184,181,181,180,182,180,187,189,188]

Create a **histogram** graph to represent the data and make the bin size 5. Make sure to label the x and y axis properly.

**Exercise #1 Code**

#####################

## Dalton Murray #

## 01/31/2024 #

## A1 - Exercise 1 #

#####################

import sys # Required for system functions

import matplotlib as mplib # Imports matplotlib library and sets an alias to mplib

from matplotlib import pyplot as plt # Imports pyplot from matplotlib and sets an alias to plt

# Defines the \_\_main\_\_ function

def \_\_main\_\_():

heightData= [143,144,141,141,141,145,149,147,152,154,152,151,154,153,

154,150,153,154,156,158,159,158,159,155,158,157,158,159,

159,158,158,159,156,156,157,156,159,155,157,162,160,160,

164,160,164,160,163,163,163,164,163,164,162,162,163,161,

160,163,162,162,164,164,160,161,163,166,166,168,166,168,

167,167,167,168,168,168,165,169,167,166,166,169,166,168,

168,168,169,168,166,168,165,169,167,169,167,168,167,165,

169,168,165,167,167,165,166,165,166,165,167,165,168,165,

166,169,168,169,167,167,170,173,173,172,171,174,174,174,

173,172,172,174,171,174,173,173,170,172,174,171,172,174,

170,170,174,174,170,170,174,173,171,174,170,170,170,174,

170,173,170,174,173,171,173,171,170,174,171,174,172,173,

170,172,173,174,170,171,170,170,172,173,172,171,174,172,

171,179,178,179,179,176,175,175,176,177,177,177,177,179,

178,178,178,175,175, 183,184,182,182,184,183,180,182,180,

182,180,183,182,180,180,184,181,181,180,182,180,187,189,188] # Defines the variable heightData and creates a list with the already given data

plt.hist(heightData, bins = 5) # Create a histogram out of the variable heightData's data and sets bin sizes to 5

plt.xlabel("Height in CM") # Sets the x axis label

plt.ylabel("Number of persons") # Sets the y axis label

plt.title("Histogram for Height Distribution") # Sets the title of the plot

plt.show() # Shows the plot

# Checks if the "\_\_name\_\_" variable equals "\_\_main\_\_"

if \_\_name\_\_ == "\_\_main\_\_":

sys.exit(\_\_main\_\_()) # Calls the "\_\_main\_\_" function and then after running exits smoothly

**Exercise #1 Picture of code**

**A computer screen shot of a computer screen

Description automatically generated**

**Exercise #1 Formatting**

I didn’t really deem too many formatting changes to this plot, at least in regard to color as it looked perfectly readable to me. I have the bin size to 5 as it was in the requirements. I also set the x label to “Height in CM” as that to me is a proper axis title and I set the y label to “Number of persons” as that seems like the proper axis title to me too. I also set the title of the plot to “Histogram for Height Distribution” as that is what this plot is. I wouldn’t really consider this formatting in regards to the plot but I also always use functions so I have the main code in the main function and I have the main function being called and then exiting smoothly with the sys library exit function. I also have matplotlib and pyplot as aliases in the imports.

**Exercise #1 Output**

**A graph on a computer screen

Description automatically generated**

**Exercise #1 Findings**

After analyzing the data and the plot I am able to see that it is more common for someone to have a height between 170 CM and 180 CM. Once it gets to 180 CM there is a huge drop in the number of people who are that height. It is also less common for this group of people to be less than 160 CM in height. This seems like a fairly normal distribution to me. It seems like there might also be a little bit of a skew of a distribution however this could be because of the sample size or because of the amount of bins.

**Exercise #2 (10 points):**

A Frozen yogurt shop records how much Frozen yogurt it sells versus the noon temperature on that day. Here is the data for the last 10 days:



Create a **scatter plot** to represent the data. What kind of things can you conclude from the data. You need to label the x and y axis properly.

**Exercise #2 Code**

#####################

## Dalton Murray #

## 01/31/2024 #

## A1 - Exercise 2 #

#####################

import sys # Required for system functions

import matplotlib as mplib # Imports matplotlib library and sets an alias to mplib

from matplotlib import pyplot as plt # Imports pyplot from matplotlib and sets an alias to plt

# Defines the \_\_main\_\_ function

def \_\_main\_\_():

temperatureData = [57.56, 61.52, 53.42, 59.36, 65.3, 71.78, 66.92, 77.18, 74.12, 64.58] # Temperature data given to us

salesData = [215, 325, 185, 332, 406, 522, 412, 614, 544, 421] # Sales data given t ous

plt.scatter(temperatureData, salesData) # Sets up a scatter plot with x and y as temperatureData and salesData

plt.xlabel("Degress Fahrenheit") # Sets the x label to Degrees Fahrenheit as that is the data we've been given

plt.ylabel("Sales Per Day USD") # Sets the y label to Sales USD as this is the amount of sales they had on that day

plt.title("Frozen Yogurt Sales vs Temperature") # Sets the title of the plot to Frozen Yogurt Sales vs Temperature as that is what this plot is showing

plt.show() # Shows the plot

# Checks if the "\_\_name\_\_" variable equals "\_\_main\_\_"

if \_\_name\_\_ == "\_\_main\_\_":

sys.exit(\_\_main\_\_()) # Calls the "\_\_main\_\_" function and then after running exits smoothly

**Exercise #2 Picture of code**

**A screenshot of a computer screen

Description automatically generated**

**Exercise #2 Formatting**

In regard to formatting for this exercise I didn’t really see a need to overcomplicate it or change the colors. I have the x label set to Degrees Fahrenheit as this is that the x axis is and I have the y label set to Sales per Day USD as this is showing how much money they made per day in USD. I also set the title to Frozen Yogurt Sales vs Temperature as that is what I am trying to find a trend or pattern for.

**Exercise #2 Output**

**A screen shot of a graph

Description automatically generated**

**Exercise #2 Findings**

While looking at the scatter plot there is a very clear trend in that the higher the degrees Fahrenheit are in that day the higher sales in USD per day there will be.

**Exercise #3 (10 points):**

**Part A)** Assume that we have a company sales data for last year (12 months) as in the table below:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| month\_number | facecream | facewash | toothpaste | bathingsoap | shampoo | moisturizer | total\_units | total\_profit |
| 1 | 2500 | 1500 | 5200 | 9200 | 1200 | 1500 | 21100 | 211000 |
| 2 | 2630 | 1200 | 5100 | 6100 | 2100 | 1200 | 18330 | 183300 |
| 3 | 2140 | 1340 | 4550 | 9550 | 3550 | 1340 | 22470 | 224700 |
| 4 | 3400 | 1130 | 5870 | 8870 | 1870 | 1130 | 22270 | 222700 |
| 5 | 3600 | 1740 | 4560 | 7760 | 1560 | 1740 | 20960 | 209600 |
| 6 | 2760 | 1555 | 4890 | 7490 | 1890 | 1555 | 20140 | 201400 |
| 7 | 2980 | 1120 | 4780 | 8980 | 1780 | 1120 | 29550 | 295500 |
| 8 | 3700 | 1400 | 5860 | 9960 | 2860 | 1400 | 36140 | 361400 |
| 9 | 3540 | 1780 | 6100 | 8100 | 2100 | 1780 | 23400 | 234000 |
| 10 | 1990 | 1890 | 8300 | 10300 | 2300 | 1890 | 26670 | 266700 |
| 11 | 2340 | 2100 | 7300 | 13300 | 2400 | 2100 | 41280 | 412800 |
| 12 | 2900 | 1760 | 7400 | 14400 | 1800 | 1760 | 30020 | 300200 |

Create a **line graph** to display the total profit of all months. Make sure to label the x and y axis properly.

**Part B)** Use the same dataset in the previous exercise to create a **bar chart** that display the total number of units sold per month. Make sure to label the x and y axis properly.

**Part A and B**

**Exercise #3 Code**

#####################

## Dalton Murray #

## 01/31/2024 #

## A1 - Exercise 3 #

#####################

import sys # Required for system functions

import matplotlib as mplib # Imports matplotlib library and sets an alias to mplib

from matplotlib import pyplot as plt # Imports pyplot from matplotlib and sets an alias to plt

# Defines the \_\_main\_\_ function

def \_\_main\_\_():

months = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12] # Create a list of the number for each month

# Unused data due to not being included in required parts of the exercise, however, I converted it from the given data to lists for future analysis as needed

# faceCreamData = [2500, 2630, 2140, 3400, 3600, 2760, 2980, 3700, 3540, 1990, 2340, 2900]

# faceWashData = [1500, 1200, 1340, 1130, 1740, 1555, 1120, 1400, 1780, 1890, 2100, 1760]

# toothPasteData = [5200, 5100, 4550, 5870, 4560, 4890, 4780, 5860, 6100, 8300, 7300, 7400]

# bathingSoapData = [9200, 6100, 9550, 8870, 7760, 7490, 8980, 9960, 8100, 10300, 13300, 14400]

# shampooData = [1200, 2100, 3550, 1870, 1560, 1890, 1780, 2860, 2100, 2300, 2400, 1800]

# moisturizerData = [1500, 1200, 1340, 1130, 1740, 1555, 1120, 1400, 1780, 1890, 2100, 1760]

# It is possible for me to make a dictionary for the following two variables however this is a simple way to do is also

totalUnitsData = [21100, 18330, 22470, 22270, 20960, 20140, 29550, 36140, 23400, 26670, 41280, 30020] # Given data for the total units sold per month

totalProfitData = [211000, 183300, 224700, 222700, 209600, 201400, 295500, 361400, 234000, 266700, 412800, 300200] # Given data for the total profit per month

# Part A

plt.plot(months, totalProfitData) # Takes in for a regular line graph the months as x and totalProfitData as y

plt.title("Total Profit vs. Month") # Sets the tiel to total profit vs month

plt.xlabel("Month of The Year") # Sets the x label to month of the year

plt.ylabel("Total Profit per Month USD") # Sets the y label to the total profit per month usd

plt.show() # Shows the plot

# Part B

plt.bar(months, totalUnitsData) # Takes in months and totalUnitsData as the x and y for a bar graph

plt.title("Total Units Sold vs. Months of year") # Sets the title to total units sold vs months of year

plt.xlabel("Month of The Year") # Sets the x axis label to month of the year

plt.ylabel("Total Units Sold per Month") # Sets the y axis label to total units sold per month

plt.show() # Shows the bar graph

# Checks if the "\_\_name\_\_" variable equals "\_\_main\_\_"

if \_\_name\_\_ == "\_\_main\_\_":

sys.exit(\_\_main\_\_()) # Calls the "\_\_main\_\_" function and then after running exits smoothly

**Exercise #3 Picture of code**

**A screenshot of a computer program

Description automatically generated**

**Exercise #3 Formatting**

I didn’t feel much formatting as needed for part A or part B other than setting the title, and x and y axis labels. I decided to set the total unit and total profit data as lists rather than dictionaries as it is simpler to just display that in the plot than having setup a dictionary to associate the months with the units sold and profit data. I also had made lists for all of the other data in the given data to us just in case for ease of use for further data analysis that wasn’t needed in the given exercise. I was considering in the graphs making it not skip over every other month on the x axis however I feel like that wasn’t needed as the information shows well already.

**Exercise #3 Output  
Part A**

A graph on a screen

Description automatically generated

**Part B**

**A screenshot of a graph

Description automatically generated**

**Exercise #3 Findings**

**Part A**

In part A’s line graph I can see that months 1-6 aren’t very profitable for the company, months 6-9 has an increase in sales before dipping on month 10 and then sharply increasing to month 11 before dropping again at month 12. The month of the year, 6, is July which is the second month of summer so they may have had an event or their product is more for the summer. Month 11 of the year is also November which is the last month before Winter officially starts in December.

**Part B**

In part B’s bar chart I have the months of the year and total units sold which I can see is directly related to part A’S profit as the more units sold means a higher amount of profit. I can also see that they have a relatively decent total amount of units sold per month and that the total profit seems to just add a 0 to the amount of profit compared to the total units. For example, month 11 has 40,000 units sold and the profit is 400,000.

I have neither given nor received unauthorized aid in completing this work, nor have I presented someone else's work as my own.

*Dalton Murray*