**Appendix**

Code – See Data Description for how data was manipulated to achieve the same results

import pandas as pd

import numpy as np

bart = pd.read\_csv("Bart\_Daily\_Station\_Exits\_filter.csv")

## https://data.sfgov.org/Transportation/BART-Daily-Station-Exits/m2xz-p7ja

biz = pd.read\_csv("Registered\_Business\_Locations\_-\_San\_Francisco.csv"

## https://data.sfgov.org/Economy-and-Community/Registered-Business-Locations-San-Francisco/g8m3-pdis

wfh = pd.read\_csv("ACS WFH Data.csv")

##2021 data: <https://data.census.gov/table?q=DP03&g=160XX00US0667000>

##2019 data: https://data.census.gov/table?q=DP03&g=160XX00US0667000&tid=ACSDP1Y2019.DP03

bartgroup = bart.groupby(by='Year')

bartgroup.mean()

x = bartgroup['Year'].mean()

y = bartgroup['Total'].mean()

bizgroup = biz.groupby(by='Year')

display(bizgroup)

x2 = bizgroup['Year'].mean()

y2 = bizgroup['Active'].count()

bars = biz['Active']

barss = bars.loc[biz.Active=="Yes"]

bars2 = barss.count()

#plot

import matplotlib.pyplot as plt

fig, ax = plt.subplots()

ax.plot(x,y,color='royalblue')

ax.set\_xlabel('Year')

ax.set\_ylabel('Number of Station Exits')

ax.set\_title("Average Number of Daily BART Station Exits per Year")

#barchart

bars1 = biz.loc[biz.Active=="No"]

bars2 = barss.count()

plt.bar(x2, bars2, color='grey', label='Active')

plt.bar(x2, y2, bottom=bars2, color='tomato', label='Closed')

plt.xlabel('Year')

plt.ylabel('Number of Businesses')

plt.title("Total Number of Active and Closed Businesses in San Francisco")

plt.legend(bbox\_to\_anchor=(1.25, 1),loc="upper right")

plt.show()

#pie

labels = 'WFH', 'Commuters'

wfh2 = wfh[["WFH","Commute"]]

wfh2 = wfh2.loc[0]

plt.pie(wfh2,labels=labels,autopct='%1.1f%%')

plt.title("Percentage of Workers Working From Home vs. Commuting in 2019")

wfh3 = wfh[["WFH","Commute"]]

wfh3 = wfh3.loc[1]

plt.pie(wfh3,labels=labels,autopct='%1.1f%%')

plt.title("Percentage of Workers Working From Home vs. Commuting in 2021")

fig = plt.figure()

ax1 = fig.add\_subplot(1, 2, 1)

ax2 = fig.add\_subplot(1, 2, 2)

ax1.pie(wfh2,labels=labels,autopct='%1.1f%%')

ax2.pie(wfh3,labels=labels,autopct='%1.1f%%')

ax1.set\_title("2019")

ax2.set\_title("2021")

fig.suptitle("Percentage of Workers Working From Home vs. Commuting in 2019 and 2021")

#y refers to ridership, y2 refers to businesses

import scipy.stats as stats

correlation, p\_value = stats.pearsonr(y, y2)

print(correlation, p\_value)