import csv

import matplotlib.pyplot as plt

import statistics as st

import pygal

import datetime as dt

filename = 'activity.csv'

with open(filename) as f:

reader = csv.reader(f)

header\_row = next(reader)

dict = {}

dictInterval = {}

dictIntervalWeekEnd = {}

dictIntervalWeekDays = {}

for row in reader:

steps = row[0]

if(steps != "NA"):

date = row[1]

date2 = int(dt.datetime.strptime(date, '%Y-%m-%d').day)

interval = int(row[2])

dict.setdefault(str(date), [])

dict[str(date)].append(int(steps))

dictInterval.setdefault(interval, [])

dictInterval[interval].append(int(steps))

if(date2 % 7 == 0):

dictIntervalWeekEnd.setdefault(interval, [])

dictIntervalWeekEnd[interval].append(int(steps))

else:

dictIntervalWeekDays.setdefault(interval, [])

dictIntervalWeekDays[interval].append(int(steps))

# print(len(dict.keys()))

listDate = []

listTotal = []

listAvg = []

# listMean = []

# listMedian = []

for i in dict.keys():

listDate.append(i)

listTotal.append(sum(dict.get(i)))

listAvg.append(st.mean(dict.get(i)))

plt.hist(listTotal)

plt.title("Total Steps per day")

plt.xlabel("Steps per day")

plt.ylabel("Frequency")

plt.yticks(range(0, 25, 5))

plt.savefig('figure1-version1.svg')

plt.close()

# plt.show()

hist = pygal.Bar()

hist.title = "Total steps per day"

hist.x\_title = "Steps per day"

hist.y\_title = "Frequency"

hist.x\_labels = listDate

hist.add('Total Number of steps', listTotal)

hist.render\_to\_file('figure1-version2.svg')

print("Mean : " + str(st.mean(listTotal)))

q = sorted(listTotal)

print("Median : " + str(st.median(q)))

# ------------------------ second case ------------------------

# What is the average daily activity pattern?

listAveragePerInterval = []

for i in dictInterval.keys():

listAveragePerInterval.append(st.mean(dictInterval.get(i)))

fig = plt.figure(dpi=80, figsize=(20, 6))

plt.plot(list(dictInterval.keys()),listAveragePerInterval, c = 'blue')

plt.title("Average daily activity")

plt.xlabel("Time Interval")

plt.ylabel("Average number of steps taken")

fig.autofmt\_xdate()

plt.savefig("figure2.svg")

plt.close()

# print(listAveragePerInterval)

maxValue = max(listAveragePerInterval)

n = 0

max = ""

indexMax = listAveragePerInterval.index(maxValue)

for i in dictInterval.keys():

if(n == indexMax):

max = i

break

n+=1

print("maximum number of steps in interval : " + str(max))

# print("File saved to figure1-version1.svg and figure1-version2.svg")

listWeekDays = []

listWeekEnd = []

for i in dictIntervalWeekDays.keys():

listWeekDays.append(st.mean(dictIntervalWeekDays.get(i)))

for i in dictIntervalWeekEnd.keys():

listWeekEnd.append(st.mean(dictIntervalWeekEnd.get(i)))

fig = plt.figure(dpi=80, figsize=(20, 6))

plt.plot(list(dictInterval.keys()),listWeekDays, c = 'blue' , label = 'WeekDays')

plt.plot(list(dictInterval.keys()),listWeekEnd, c = 'red' , label = 'WeekEnd')

plt.legend(loc = 'upper left')

plt.title("All week days and weekend days")

plt.xlabel("Time Interval")

plt.ylabel("Average number of steps taken")

fig.autofmt\_xdate()

plt.savefig("figure4.svg")

plt.close()

import csv

import matplotlib.pyplot as plt

import statistics as st

filename = 'activity.csv'

with open(filename) as f:

reader = csv.reader(f)

header = next(reader)

wr = open("newDataSet.csv", "w")

wr.write(str(header[0]) + "," + str(header[1]) + "," + str(header[2]))

wr.write("\n")

n = 0

for row in reader:

if(row[0] == "NA"):

row[0] = 0

n += 1

wr.write(str(row[0]) + "," + str(row[1]) + "," + str(row[2]))

wr.write("\n")

wr.close()

print("The total numbers of missing values in dataset is : " + str(n))

print()

print("New dataset created successfully")

newfilename = 'newDataSet.csv'

with open(newfilename) as f:

reader = csv.reader(f)

header\_row = next(reader)

dict = {}

dictInterval = {}

for row in reader:

steps = row[0]

if(steps != "NA"):

date = row[1]

interval = int(row[2])

dict.setdefault(str(date), [])

dict[str(date)].append(int(steps))

dictInterval.setdefault(interval, [])

dictInterval[interval].append(int(steps))

# print(len(dict.keys()))

listDate = []

listTotal = []

listAvg = []

# listMean = []

# listMedian = []

for i in dict.keys():

listDate.append(i)

listTotal.append(sum(dict.get(i)))

listAvg.append(st.mean(dict.get(i)))

plt.hist(listTotal)

plt.title("Total Steps per day")

plt.xlabel("Steps per day")

plt.ylabel("Frequency")

plt.yticks(range(0, 25, 5))

plt.savefig('figure3(inputting missing values).svg')

plt.close()

import csv

import matplotlib.pyplot as plt

from datetime import datetime

# firstdate = datetime.strptime('2014-7-1', '%Y-%m-%d')

# print(firstdate)

filename = 'sitka\_weather\_2014.csv'

with open(filename) as f:

reader = csv.reader(f)

header\_row = next(reader)

# for index, column\_header in enumerate(header\_row):

# print(index, column\_header)

dates, highs , lows = [] , [] , []

for row in reader:

try:

current\_date = datetime.strptime(row[0], '%Y-%m-%d').strftime("%B %y")

#

high = int(row[1])

low = int(row[3])

except ValueError:

print(current\_date, 'Missing data')

else:

dates.append(current\_date)

highs.append(high)

lows.append(low)

# 2014-1-1

print(dates)

# Plot data.

fig = plt.figure(dpi=100, figsize=(10, 6))

plt.plot(dates, highs, c='red', alpha = 0.5)

plt.plot(dates, lows, c = 'blue', alpha = 0.5)

plt.fill\_between(dates, highs, lows, facecolor='blue', alpha=0.1)

# Format plot.

title = "Daily high and low temperatures - 2014\nDeath Valley, CA"

plt.title(title, fontsize=20)

plt.xlabel('', fontsize=16)

fig.autofmt\_xdate()

plt.ylabel("Temperature (F)", fontsize=16)

plt.tick\_params(axis='both', which='major', labelsize=16)

plt.show()