Robert Gorman

HW-2 Submission

#1:

'''

1. Numpy

a. Create an array with 2 rows and 2 columns using the following matrix

3 1

2 4

b. Select the second row of the array.

c. Select the second column of the array.

d. Select the number in the second row and second column

e. Replace the numbers >2 with 0

'''

import numpy as np

myarray0 = np.array([[3, 1], [2, 4]])

#Select the second row of the array

second\_row = myarray0[1]

#Select the second column of the array.

second\_column = myarray0[:,1]

#Select the number in the second row and second column

second\_row\_second\_column = myarray0[1,1]

#Replace the numbers >2 with 0

myarray0[myarray0 > 2] = 0

print (myarray0)

print (second\_row)

print (second\_column)

print (second\_row\_second\_column)

print (myarray0)

#2:

'''

2. Plot

a. Generate 10 random numbers between 0 and 1 from uniform distribution and plot the numbers

using line chart. Use these 10 random numbers as the Y variable, and define your own X variable.

Use dashed line and set line color as grey. Add chart title “line chart,” legend “random numbers,”

and axis labels “x variable, random numbers.”

b. Generate 10 random numbers from the normal distribution with 0 mean and 1 standard deviation.

Plot the numbers using bar chart. Use these 10 random numbers as the Y variable, and define your

own X variable. Set bar color as red. Add chart title “bar chart,” legend “random numbers\_normal

distribution,” and axis labels “x variable, random numbers.”

c. Generate 1000 random numbers from the normal distribution with 0 mean and 1 standard

deviation. Plot the numbers using a histogram with 50 bins. Set color as blue. Add chart title

“Histogram,” legend “random numbers\_normal distribution,” and axis labels “Bin, Frequency.”

'''

import matplotlib.pyplot as plt

import numpy as np

#Generate 10 random numbers between 0 and 1 from uniform distribution

x = np.linspace(1, 10, 10)

y = np.random.uniform(0, 1, 10) #Use these 10 random numbers as the Y variable

plt.figure(figsize=(10,8))

plt.plot(x, y, linestyle='dashed', color='grey', label='random numbers')

#Add chart title “line chart,” legend “random numbers,” and axis labels “x variable, random numbers.”

plt.xlabel('x variable')

plt.ylabel('random numbers')

plt.title('line chart')

plt.legend()

plt.show()

#b

x = np.linspace(1, 10, 10)

y = np.random.normal(0, 1, 10) #Generate 10 random numbers from the normal distribution with 0 mean and 1 standard deviation

plt.figure(figsize=(8, 6)) #Plot the numbers using bar chart

plt.bar(x, y, color='red', label='random numbers\_normal distribution')

#Set bar color as red. Add chart title “bar chart,” legend “random numbers\_normal distribution,” and axis labels “x variable, random numbers.”

plt.xlabel('x variable')

plt.ylabel('random numbers')

plt.title('bar chart')

plt.legend()

plt.show()

#c

y = np.random.normal(0, 1, 1000) #Generate 1000 random numbers from the normal distribution with 0 mean and 1 standard deviation.

plt.figure(figsize=(8, 6))

plt.hist(y, bins=50, color='blue', edgecolor='grey', alpha=0.7, label='random numbers\_normal distribution') # Plot histogram

#Plot the numbers using a histogram with 50 bins. Set color as blue.

#Add chart title “Histogram,” legend “random numbers\_normal distribution,” and axis labels “Bin, Frequency.”

plt.xlabel('bin')

plt.ylabel('frequency')

plt.title('histogram')

plt.legend()

plt.show()

#3:

'''

3. Pandas. Use the NBA dataset

a. Sort the data by position and then by salary in ascending order.

b. Select the rows for players in Philadelphia 76ers and age>30

c. Calculate the standard deviation of players’ salary in each team and report the team with the smallest standard deviation.

d. Calculate the mean and 80th percentile salary for each position-age combination

'''

import pandas as pd

df = pd.read\_csv(r'nba\_csv-1.csv')

#Sort the data by position and then by salary in ascending order.

df\_sorted = df.sort\_values(by=['Position', 'Salary'], ascending=True)

#Select the rows for players in Philadelphia 76ers and age>30

df\_sixers = df.query("Team == 'Philadelphia 76ers' and Age > 30")

#Calculate the standard deviation of players’ salary in each team and report the team with the smallest standard deviation.

team\_std\_dev = df.groupby('Team')['Salary'].std()

team\_min\_std\_dev = team\_std\_dev.idxmin()

print(f"\n C: The team with smallest salary standard deviation is the {team\_min\_std\_dev}\n")

#Calculate the mean and 80th percentile salary for each position-age combination

salary\_stats = df.groupby(['Position', 'Age'])['Salary'].describe(percentiles=[0.8])

print (f"\nD: The mean and 80th percentile salary for each position-age combination is below\n")

print (salary\_stats)

--------------------------------------------------------------------------------------------------------------------------------------









