**Machine Learning (Assignment # 2)**

**Name: Sindhu Rajanala**

**Student id: 700741228**

1. Numpy:

Using NumPy create random vector of size 15 having only Integers in the range 1-20.

import numpy as np

x = np.random.randint(1,20,15)

print(x)

output:

Background pattern

Description automatically generated with medium confidence

1. Reshape the array to 3 by 5

x=x.reshape(3,5)

print(x)

output:

A picture containing graphical user interface

Description automatically generated

1. Print array shape.

x.shape

output:



1. Replace the max in each row by 0

a=np.where(x==x.max(axis=1)[:,None],0,x)

print(a)

output:

Graphical user interface

Description automatically generated with medium confidence

2. Pandas

1.Read the provided CSV file ‘data.csv’. <https://drive.google.com/drive/folders/1h8C3mLsso-sIOLsvoYwPLzy2fJ4IOF?usp=sharing>

import pandas as pd

da = pd.read\_csv("data.csv")

da

output:

Graphical user interface, application

Description automatically generated

2.Show the basic statistical description about the data

da.describe()

output:

Graphical user interface, application, Word

Description automatically generated

3.Check if the data has null values.

da.isnull().any()

output:

Graphical user interface

Description automatically generated with medium confidence

a. Replace the null values with the mean

da.fillna(da.mean(), inplace=True)

da.isnull().any()

output:

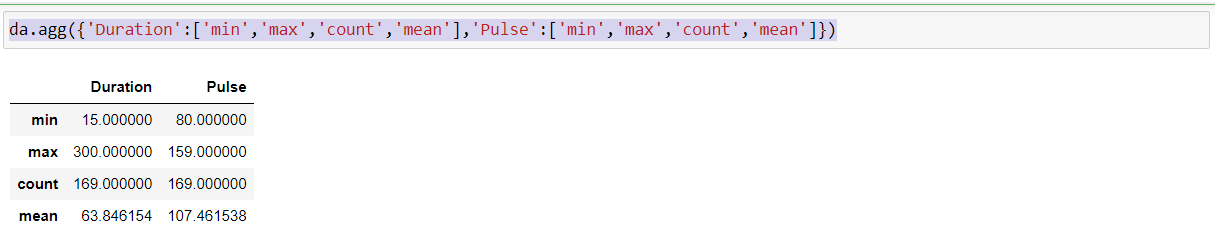
A picture containing graphical user interface

Description automatically generated

4.Select at least two columns and aggregate the data using: min, max, count, mean.

da.agg({'Duration':['min','max','count','mean'],'Pulse':['min','max','count','mean']})

output:



5.Filter the dataframe to select the rows with calories values between 500 and 1000.

da.loc[(da['Calories']>500)&(da['Calories']<1000)]

output:

Graphical user interface, application

Description automatically generated

6. Filter the dataframe to select the rows with calories values > 500 and pulse < 100.

da.loc[(da['Calories']>500)&(da['Pulse']<100)]

output:Graphical user interface, application

Description automatically generated

7. Create a new “df\_modified” dataframe that contains all the columns from df except for “Maxpulse”

df\_modified = da[['Duration','Pulse','Calories']]

df\_modified

output:

Graphical user interface, application, Word

Description automatically generated

8. Delete the “Maxpulse” column from the main df dataframe

df=da.drop('Maxpulse',axis=1)

df

output:

Graphical user interface, application

Description automatically generated

9. Convert the datatype of Calories column to int datatype.

da['Calories'] = da['Calories'].astype(np.int64)

da.dtypes

output:

Graphical user interface, text, application

Description automatically generated

10. Using pandas create a scatter plot for the two columns (Duration and Calories).

da.plot.scatter(x='Duration',y='Calories',c='DarkBlue')

output:

Graphical user interface, chart

Description automatically generated with medium confidence

3. Matplotlib

Write a Python programming to create a below chart of the popularity of programming Languages.

import matplotlib.pyplot as pl

Programming\_languages = 'Java', 'Python', 'PHP', 'JavaScript', 'C#', 'C++'

popularity = [22.2, 17.6, 8.8, 8, 7.7, 6.7]

color = ["#1f77b4", "#ff7f0e", "#2ca02c", "#d62728", "#9467bd", "#8c564b"]

explode = (0.1, 0, 0, 0,0,0)

pl.pie(popularity, explode=explode, labels=Programming\_languages, colors=color,

autopct='%1.1f%%', shadow=True, startangle=140)

pl.axis('equal')

pl.show()

output:

Chart

Description automatically generated with low confidence