JoePayyappilly_227_Lab10,11

September 23, 2023

Lab Exercise 10

Q1. Apply regular expression for form validation. Create your domain-form using Tkinter Module. 1. Form should contain Text box [For Name, Email Id, Phone number], Dropdown [for Gender], Spinbox [for Year/DoB] and other necessary widgets required for your domain. 2. Validate Your Name, Email Id, Phone number in the form.

[ ]:
```
import tkinter as tk
from tkinter import ttk
import re

def validate_name(name):
    # letters and spaces allowed
    return re.match(r'^[A-Za-z\s]+$', name)

def validate_email(email):
    return re.match(r'^[a-zA-Z0-9._%+-]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,}$', email)

def validate_phone(phone):
    #10 digits allowed
    return re.match(r'^\d{10}$', phone)

def submit_form():
    name = name_entry.get()
    email = email_entry.get()
    phone = phone_entry.get()
    gender = gender_combobox.get()
    dob = dob_spinbox.get()

    if validate_name(name) and validate_email(email) and validate_phone(phone):
        print("Name:", name)
        print("Email:", email)
        print("Phone:", phone)
        print("Gender:", gender)
        print("Year of Birth:", dob)

        result_label.config(text="Form submitted successfully!")
    else:
```

```
        result_label.config(text="Invalid input.")


root = tk.Tk()
root.title("Domain Form")


root.geometry("400x300")


name_label = tk.Label(root, text="Name:")
name_label.pack()
name_entry = tk.Entry(root)
name_entry.pack()

email_label = tk.Label(root, text="Email:")
email_label.pack()
email_entry = tk.Entry(root)
email_entry.pack()

phone_label = tk.Label(root, text="Phone:")
phone_label.pack()
phone_entry = tk.Entry(root)
phone_entry.pack()


gender_label = tk.Label(root, text="Gender:")
gender_label.pack()
gender_options = ["Male", "Female", "Other"]
gender_combobox = ttk.Combobox(root, values=gender_options)
gender_combobox.set(gender_options[0])
gender_combobox.pack()


dob_label = tk.Label(root, text="Year of Birth:")
dob_label.pack()
dob_spinbox = tk.Spinbox(root, from_=1900, to=2023)
dob_spinbox.pack()


submit_button = tk.Button(root, text="Submit", command=submit_form)
submit_button.pack()


result_label = tk.Label(root, text="")
result_label.pack()
```

```
root.mainloop()
```

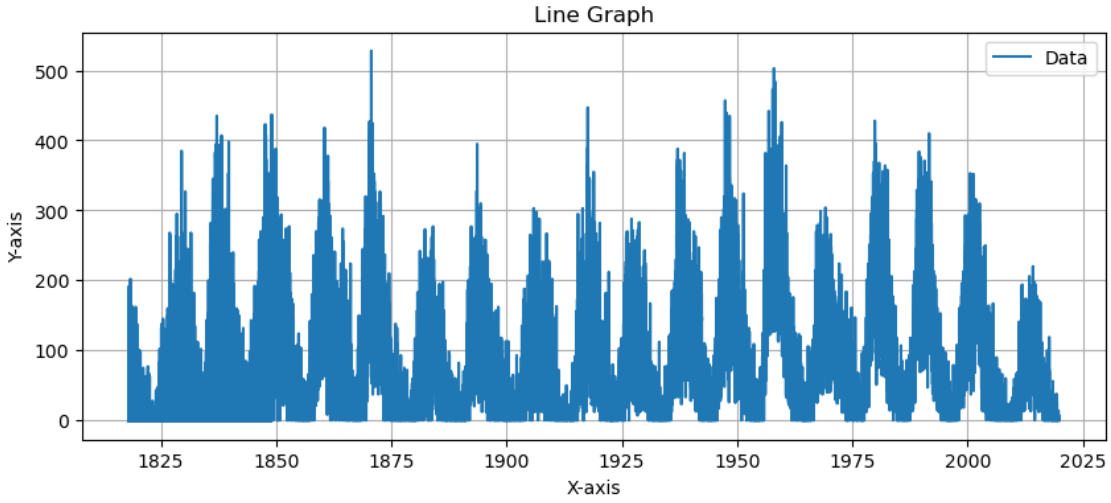
```
Name: Joe
Email: jjj@mm.com
Phone: 8971836281
Gender: Male
Year of Birth: 1900
```

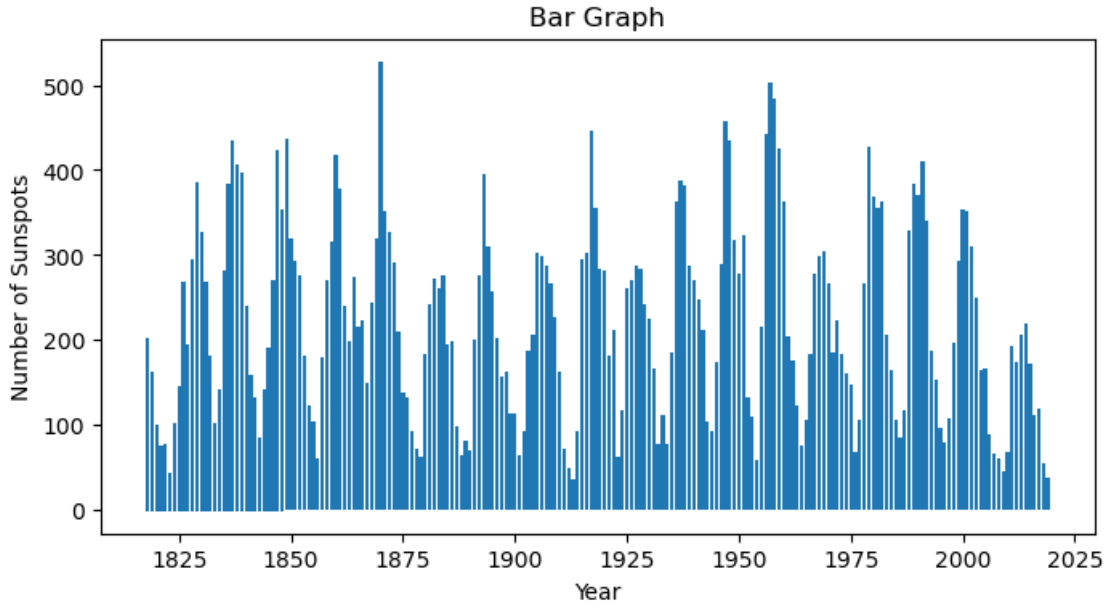
Lab Exercise 11

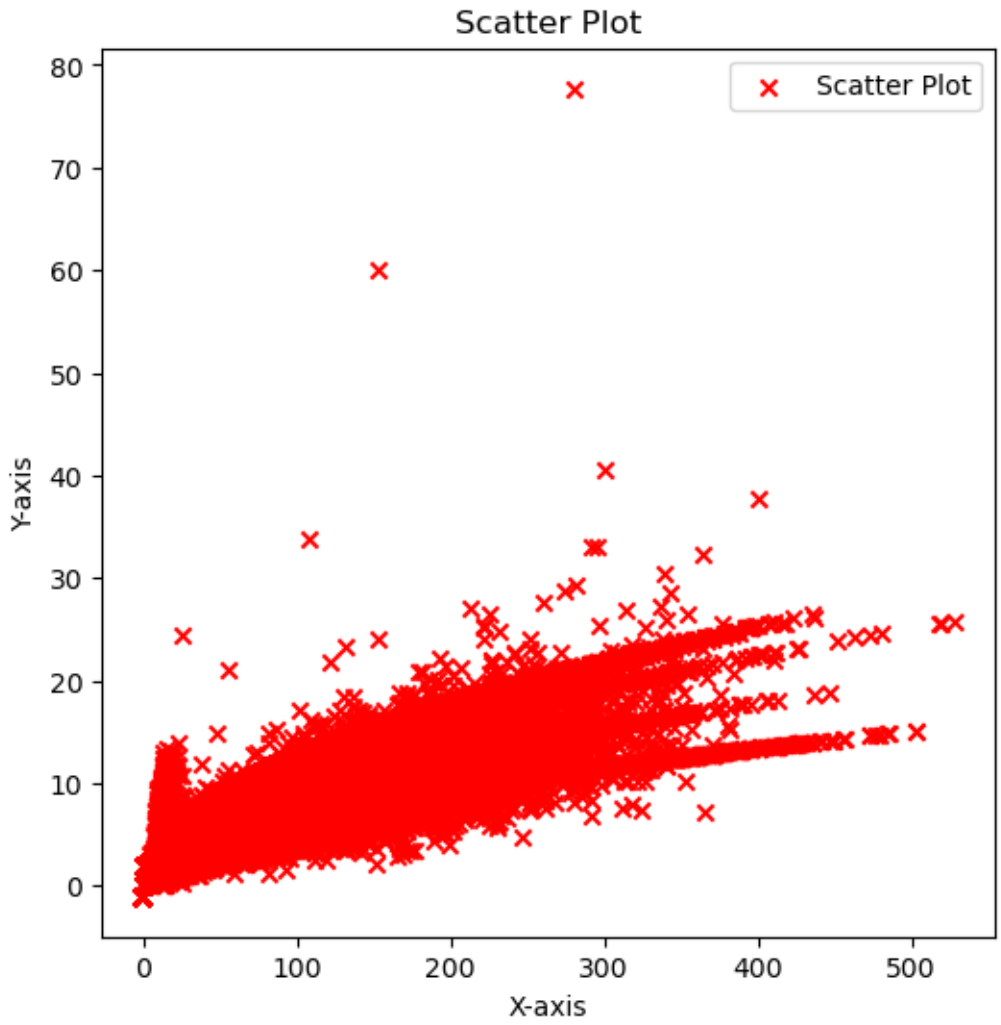
Q2. Perform the Exploratory Data Analysis on your domain-based dataset and demonstrate the retrieved insights using "Matplotlib" modules. Visualize hidden insights using appropriate plots (graphs) [Usage of line plot and scatter plot are mandatory]

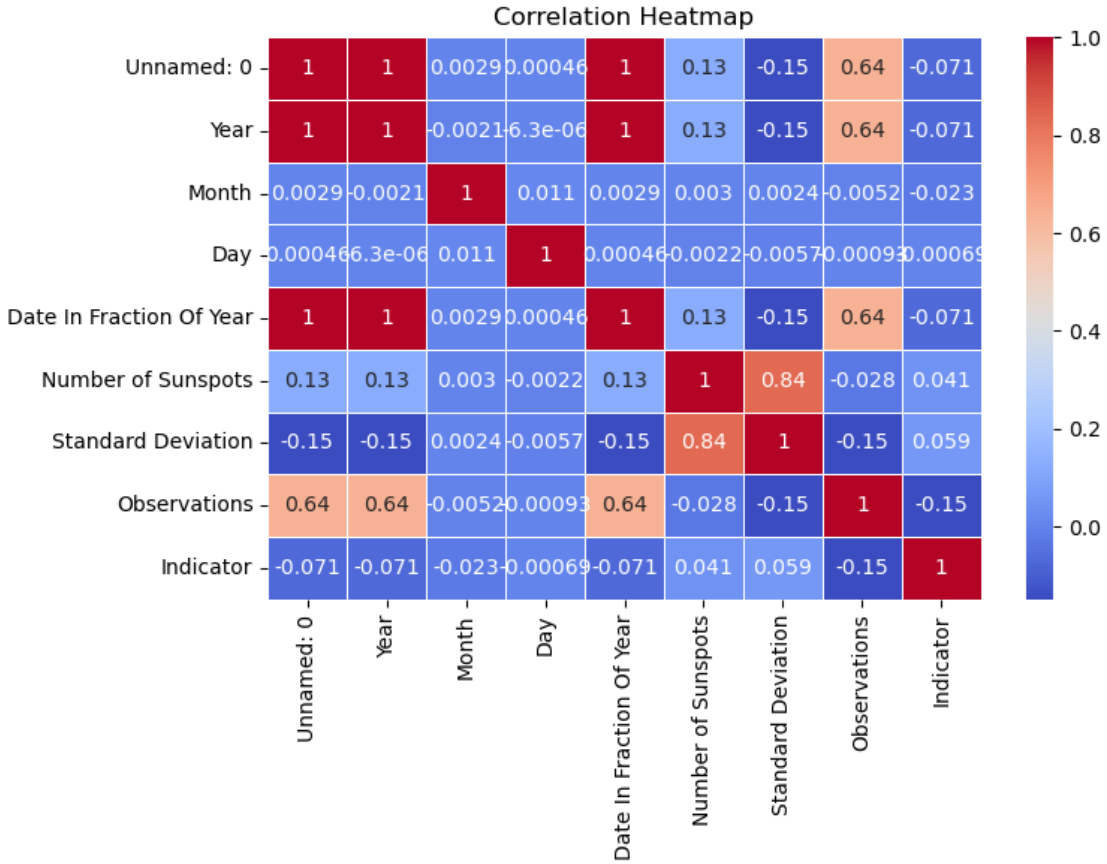
[ ]:
```
import matplotlib.pyplot as plt
import seaborn as sns
import pandas as pd

df = pd.read_csv('sunspot_data.csv')

# Line Graph
plt.figure(figsize=(10, 4))
plt.plot(df['Date In Fraction Of Year'], df['Number of Sunspots'], label='Data')
plt.xlabel('X-axis')
plt.ylabel('Y-axis')
plt.title('Line Graph')
plt.legend()
plt.grid(True)

# Bar Graph
plt.figure(figsize=(8, 4))
plt.bar(df['Year'], df['Number of Sunspots'])
plt.xlabel('Year')
plt.ylabel('Number of Sunspots')
plt.title('Bar Graph')

# Scatter Plot
plt.figure(figsize=(6, 6))
plt.scatter(df['Number of Sunspots'], df['Standard Deviation'], label='Scatter␣
 ↪Plot', c='red', marker='x')
plt.xlabel('X-axis')
plt.ylabel('Y-axis')
plt.title('Scatter Plot')
plt.legend()

# Correlation Heatmap
correlation_data = df.corr()
plt.figure(figsize=(8, 6))
```

```
sns.heatmap(correlation_data, annot=True, cmap='coolwarm', linewidths=0.5)
plt.title('Correlation Heatmap')

plt.tight_layout()
plt.show()
```






Scatter Plot
Scatter Plot
Y-axis
X-axis
0
10
20
30
40
50
60
70
80
0
100
200
300
400
500


Correlation Heatmap
Unnamed: 0
Year
Month
Day
Date In Fraction Of Year
Number of Sunspots
Standard Deviation
Observations
Indicator
1 1 0.0029 0.00046 1 0.13 -0.15 0.64 -0.071
1 1 -0.0021 6.3e-06 1 0.13 -0.15 0.64 -0.071
0.0029 -0.0021 1 0.011 0.0029 0.003 0.0024 -0.0052 -0.023
0.00046 6.3e-06 0.011 1 0.00046 -0.0022 -0.0057 0.00093 0.00069
1 1 0.0029 0.00046 1 0.13 -0.15 0.64 -0.071
0.13 0.13 0.003 -0.0022 0.13 1 0.84 -0.028 0.041
-0.15 -0.15 0.0024 -0.0057 -0.15 0.84 1 -0.15 0.059
0.64 0.64 -0.0052 0.00093 0.64 -0.028 -0.15 1 -0.15
-0.071 -0.071 -0.023 -0.00069 -0.071 0.041 0.059 -0.15 1
1.0
0.8
0.6
0.4
0.2
0.0