

Requirements

This is the readme file for the data analysis task on North and Southern Wales crime report. The libraries used for this project are listed below; - Pandas - Matplotlib.pyplot - Seaborn - Numpy - Os - Glob - Folium - tkinter - Pillow

Important information

These are information the user of this program needs to be aware of to get optimum usage of this code. 1. The initial code was run using the file path as seen on the source machine, when you want to use this program ensure you change all the paths to the right one as seen on your machine else this code would not work 2. Ensure all modules are imported appropriately 3. Ensure the Data Exploration app is closed when not in use.

Dataset Used

These are information on data used. The North and South Wales dataset from January 2019 till October 2020 was used in this analysis

Pandas

pandas is a software library written for the Python programming language for data manipulation and analysis.

Installation

Find pandas installation details in link below.

```
https://pandas.pydata.org/pandas-docs/stable/install.html
```

Usage

```
import pandas as pd

pd.read_csv('')
pd.concat()
pd.read_json()
```

glob

The glob module is used to retrieve files/pathnames matching a specified pattern.

Installation

Find Glob installation details in link below.

```
https://pypi.org/project/glob2/0.4.1/
```

Usage

```
import glob2

for filename, (version,) in glob2.iglob('./binaries/project-*.zip', with_matches=True):
    print version
```

Seaborn

Seaborn is a Python data visualization library based on matplotlib. It provides a high-level interface for drawing attractive and informative statistical graphics.

Installation

Find Seaborn installation details in link below.

```
https://seaborn.pydata.org/installing.html
```

Usage

```
import seaborn as sns
df = sns.load_dataset("penguins")
sns.pairplot(df, hue="species")
```

Matplotlib.pyplot

matplotlib.pyplot is a collection of functions that make matplotlib work like MATLAB. Each pyplot function makes some change to a figure: e.g., creates a figure, creates a plotting area in a figure, plots some lines in a plotting area, decorates the plot with labels, etc.

Installation

Find Matplotlib.pyplot installation details in link below.

```
https://matplotlib.org/3.3.3/users/installing.html
```

Usage

```
import numpy as np
import matplotlib.pyplot as plt

x = np.arange(0, 5, 0.1);
y = np.sin(x)
plt.plot(x, y)
```

folium

folium makes it easy to visualize data that's been manipulated in Python on an interactive leaflet map.

Installation

Find Folium installation details in link below.

```
https://matplotlib.org/3.3.3/users/installing.html
```

Usage

```
import folium
m = folium.Map(location=[45.5236, -122.6750])
```

Numpy

NumPy is a library for the Python programming language, adding support for large, multi-dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays.

Installation

Find numpy installation details in link below.

```
https://numpy.org/install/
```

Usage

```
import numpy as np

x = np.arange(0, 5, 0.1);
y = np.sin(x)
```

Tkinter

The tkinter package ("Tk interface") is the standard Python interface to the Tk GUI toolkit. Both Tk and tkinter are available on most Unix platforms, as well as on Windows systems.

Installation

Find Tkinter installation details in link below.

```
https://tkdocs.com/tutorial/install.html
```

Usage

```
>>> import tkinter
>>> tkinter._test()
>>> tkinter.Tcl().eval('info patchlevel')
```

pillow

The Python Imaging Library adds image processing capabilities to your Python interpreter.

This library provides extensive file format support, an efficient internal representation, and fairly powerful image processing capabilities.

Installation

Find pillow installation details in link below.

```
https://pillow.readthedocs.io/en/stable/installation.html
```

Usage

```
from PIL import Image # Import Image class from the library.

image = Image.open("file.jpg") # Load the image.
rotated_image = image.rotate(180) # Rotate the image by 180 degrees.
rotated_image.save("file_rotated.jpg") # Save the rotated image.
```

scipy.stats

All of the statistics functions are located in the sub-package `scipy.stats` and a fairly complete listing of these functions can be obtained using `info(stats)` function.

Installation

Find `scipy.stats` installation details in link below.

```
https://www.scipy.org/install.html
```

Usage

```
from scipy import stats
from scipy.stats import norm

print('bounds of distribution lower: %s, upper: %s' % (norm.a, norm.b))
bounds of distribution lower: -inf, upper: inf
```