Computational Neuroscience

2022

It's time to test your **Python** skills!

You will:

```
Perform an exploratory analysis of a given dataset.
Gain data-driven insight into potential issues or findings regarding the dataset you chose.
Visualise data distributions.
```

INSTRUCTIONS

Dataset:

- **Covid-19**. Time series data tracking the number of people affected by COVID-19 worldwide.

  Data source: https://github.com/datasets/covid-19/tree/master/data
  Data source: https://ourworldindata.org/coronavirus-source-data

With the dataset, accomplish:

a) - Quickly describe a dataset, number of rows/columns, features, feature description, missing data, data types, preview the last 7 rows. **(25 pts)**

b) - Compute mean, median and standard deviation for three features of the dataset. *Use a for loop.* Don't forget about the units. **(25 pts)**

c) - Convert your dataset into a Pandas Object (Pandas DataFrame) and execute: print(dataset.describe()). What information is displayed? Provide a **clear** explanation of it. What attributes are used and why? **(25 pts)**

d) - Calculate and visualise (At least three graphs) relations between variables, correlations. All graphs should include title, axis names and units of measurement. Provide a **clear** explanation and conclusions for each graph. **(25 pts)**

Note: This assignment must be completed individually – any submitted work that is suspected of being the product of collusion will be thoroughly investigated and those involved will be penalised. Credit will not be given to material that is copied (either unchanged or minimally modified) from published sources, including web sites.

1 - Libraries

Let's first import all the packages that you will need during this assignment.

In [1]:
```
# import libraries

import numpy as np
import pandas as pd
import seaborn as sb
import matplotlib.pyplot as plt
from contextlib import nullcontext
import random
from tkinter.font import names
import math as mp
import random as r
from matplotlib import pyplot as plt
import pandas as pd
columns=['Date', 'Country/Region','Province/State', 'Confirmed', 'Recovered','Deaths']
```

2 - Dataset

Now, let's get the dataset you will work on.

In [30]:
```
# Import dataset
DataSet = pd.read_csv('time-series-19-covid-combined.csv',names = columns,low_memory=False)
```

In [3]:
```
# Preview the first 8 rows of the dataset and the last 11 rows of the dataset
firstEight= pd.read_csv('time-series-19-covid-combined.csv', skiprows=[i for i in range(8,226916)],names = columns)
firstEight
```

Out[3]:

	Date	Country/Region	Province/State	Confirmed	Recovered	Deaths
0	Date	Country/Region	Province/State	Confirmed	Recovered	Deaths
1	2020-01-22	Afghanistan	NaN	0	0	0
2	2020-01-23	Afghanistan	NaN	0	0	0
3	2020-01-24	Afghanistan	NaN	0	0	0
4	2020-01-25	Afghanistan	NaN	0	0	0
5	2020-01-26	Afghanistan	NaN	0	0	0
6	2020-01-27	Afghanistan	NaN	0	0	0
7	2020-01-28	Afghanistan	NaN	0	0	0

In [4]:
```
lastEleven= pd.read_csv('time-series-19-covid-combined.csv', skiprows=226906,names = columns)
lastEleven
```

Out[4]:

	Date	Country/Region	Province/State	Confirmed	Recovered	Deaths
0	2022-03-20	Zimbabwe	NaN	244452	0	5426
1	2022-03-21	Zimbabwe	NaN	244885	0	5429
2	2022-03-22	Zimbabwe	NaN	244885	0	5429
3	2022-03-23	Zimbabwe	NaN	244958	0	5432
4	2022-03-24	Zimbabwe	NaN	245194	0	5432
5	2022-03-25	Zimbabwe	NaN	245645	0	5436
6	2022-03-26	Zimbabwe	NaN	245645	0	5436
7	2022-03-27	Zimbabwe	NaN	245820	0	5436
8	2022-03-28	Zimbabwe	NaN	245927	0	5438
9	2022-03-29	Zimbabwe	NaN	246042	0	5439
10	2022-03-30	Zimbabwe	NaN	246182	0	5440

3 - Quickly describe the dataset:

- Description of dataset
- Number of rows/columns
- Preview the last 7 rows of the dataset
- Description of features
- Missing data (if any)
- Data types

This is the very first step towards the exploratory analysis.

Description of the Dataset

In this document, the many linked charts, our COVID-19 Data Explorer, and the Complete COVID-19 dataset we report and visualize the data on confirmed cases and deaths from Johns Hopkins University (JHU).

In [5]:
```
print('Total Registers (ROWS):', DataSet.shape[0])
print('Total Attributes (COLUMNS):', DataSet.shape[1])

lastSeven= pd.read_csv('time-series-19-covid-combined.csv', skiprows=226910,names = columns)
lastSeven
```

```
Total Registers (ROWS): 226917
Total Attributes (COLUMNS): 6
```

Out[5]:

	Date	Country/Region	Province/State	Confirmed	Recovered	Deaths
0	2022-03-24	Zimbabwe	NaN	245194	0	5432
1	2022-03-25	Zimbabwe	NaN	245645	0	5436
2	2022-03-26	Zimbabwe	NaN	245645	0	5436
3	2022-03-27	Zimbabwe	NaN	245820	0	5436
4	2022-03-28	Zimbabwe	NaN	245927	0	5438
5	2022-03-29	Zimbabwe	NaN	246042	0	5439
6	2022-03-30	Zimbabwe	NaN	246182	0	5440

In [24]:
```
print('DESCRIPTION OF FEATURES')
print(columns[0]+' is the date where COVID cases, recoveries and decease have been reported.')
print(columns[1]+' are the countries of interest in our data set')
print(columns[2]+' are the provinces of the countries of interest in our data set')
print(columns[3]+' refers to the confirmed cases of COVID-19.')
print(columns[4]+' refers to the confirmed recoveries of COVID-19.')
print(columns[5]+' refers to the confirmed deaths due to COVID-19.')
```

```
DESCRIPTION OF FEATURES
Date is the date where COVID cases, recoveries and decease have been reported.
Country/Region are the countries of interest in our data set.
Province/State are the provinces of the countries of interest in our data set
Confirmed refers to the confirmed cases of COVID-19.
Recovered refers to the confirmed recoveries of COVID-19.
Deaths refers to the confirmed deaths due to COVID-19.
```

In [25]:
```
DataSet.isnull()
```

Out[25]:

	Date	Country/Region	Province/State	Confirmed	Recovered	Deaths
0	False	False	False	False	False	False
1	False	False	True	False	False	False
2	False	False	True	False	False	False
3	False	False	True	False	False	False
4	False	False	True	False	False	False
...	...	...	...	...	...	...
226912	False	False	True	False	False	False
226913	False	False	True	False	False	False
226914	False	False	True	False	False	False
226915	False	False	True	False	False	False
226916	False	False	True	False	False	False

226917 rows × 6 columns

In [26]:
```
DataSet.dtypes
```

Out[26]:
```
Date              object
Country/Region    object
Province/State    object
Confirmed         object
Recovered         object
Deaths            object
dtype: object
```

4 - Convert your dataset into a Pandas Object (Pandas DataFrame) and execute: print(dataset.describe()). What attributes are mentioned and why?

What information is displayed? Provide a clear explanation of it.

In [27]:
```
# Dataset.describe()
DataSet = pd.DataFrame(DataSet)
DataSet.describe()
```

Out[27]:

	Date	Country/Region	Province/State	Confirmed	Recovered	Deaths
count	226917	226917	71112	226917	214133	226917
unique	800	199	90	87314	44178	29388
top	2021-02-24	China	Macau	0	0	0
freq	284	27166	799	18011	87981	45675

5 - Compute mean, median and standard deviation for three features of the dataset. *Use a for loop* to generate the information. What can you conclude with the information?

Don't forget the units of measurement.

In [41]:
```
# Compute some statistics
DataSet= pd.read_csv('time-series-19-covid-combined.csv', skiprows=1,names = columns)
DataSet
for i in range(3,6):
    print('Indexed by '+columns[i])
    print('Mean of total cases: '+str(DataSet[columns[i]].mean()))
    print('Median of total cases: '+str(DataSet[columns[i]].median()))
    print('Standard Deviation of total cases: '+str(DataSet[columns[i]].std()))
```

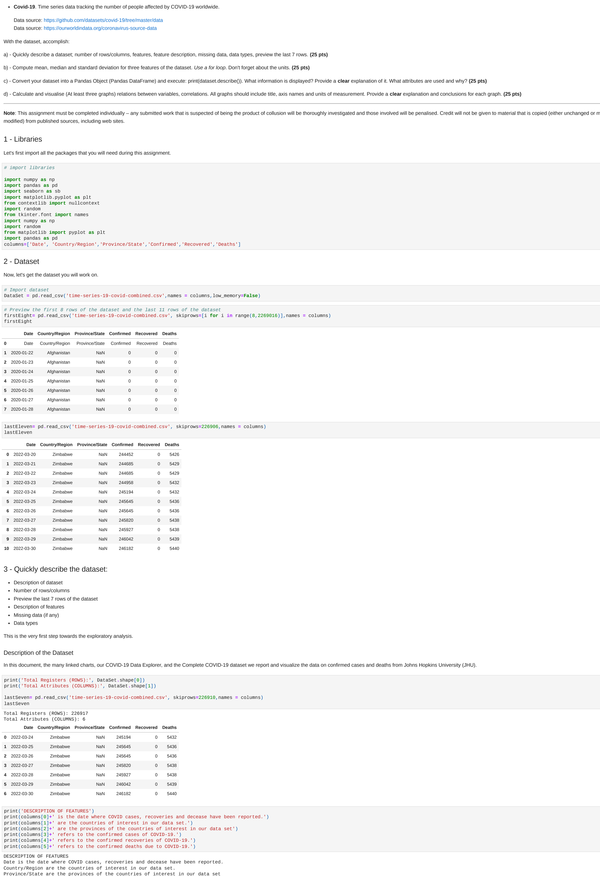
```
Indexed by Confirmed
Mean of total cases: 486833.42673170069
Median of total cases: 5333.0
Standard Deviation of total cases: 2877439.9538756423
Indexed by Recovered
Mean of total cases: 180471.44555227617
Median of total cases: 79.0
Standard Deviation of total cases: 849483.3101250608
Indexed by Deaths
Mean of total cases: 9512.573586701687
Median of total cases: 69.0
Standard Deviation of total cases: 40752.511773361526
```

6 -Calculate and visualise (At least three graphs) relations between variables, correlations. Provide a **clear** explanation of each graph.

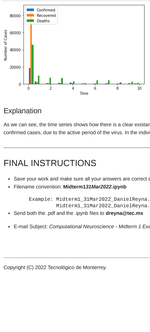
All graphs should include title, axis names and units of measurement.

In [70]:
```
fig = plt.figure(figsize = (15,15), )
ax = fig.gca()
plt.subplot(2,2,1)

plt.hist(DataSet[columns[3]], bins = 15)

plt.title('A Histogram of confirmed cases from the COVID-19 Data Set')
plt.xlabel('Time [Units Undefined]')
plt.ylabel('Number of Cases')

plt.subplot(2,2,2)

plt.hist(DataSet[columns[4]], bins = 15)

plt.title('A Histogram of recovered cases from the COVID-19 Data Set')
plt.xlabel('Time [Units Undefined]')
plt.ylabel('Number of Cases')

plt.subplot(2,2,3)

plt.hist(DataSet[columns[5]], bins = 15)

plt.title('A Histogram of decease cases from the COVID-19 Data Set')
plt.xlabel('Time [Units Undefined]')
plt.ylabel('Number of Cases')
```

Out[70]:
```
Text(0, 0.5, 'Number of Cases')
```



In [71]:
```
bins = np.linspace(0, 10, 20)

plt.hist([DataSet[columns[3]], DataSet[columns[4]],DataSet[columns[5]]], bins, label=['Confirmed', 'Recovered','Deaths'])
plt.legend(loc='upper left')
plt.xlabel('Time [Units Undefined]')
plt.ylabel('Number of Cases')
plt.show()
```



Explanation

As we can see, the time series shows how there is a clear existance of delay of Deaths after a certain time where the cases where confirmed. Nonetheless, it would be difficult to assure that the high number of recovered cases in the first entry is related to the current confirmed cases, due to the active period of the virus. In the individual histograms shown above, it is easy to observe how the scale behaves exponentially in time, reducing the sensibility of the data distribution.

FINAL INSTRUCTIONS

- Save your work and make sure all your answers are correct and in good format.
- Filename convention: **Midterm13JAnr2022.ipynb**

  Example: Midterm1_31Mar2022_DanielHeyna.ipynb
  Midterm1_31Mar2022_DanielHeyna.pdf
- Send both the .pdf and the .ipynb files to **dheyna@tec.mx**
- E-mail Subject: *Computational Neuroscience - Midterm 1 Exam*
