Hands on Activity 10.1 Data Analysis using Python

Catulay, Weslie Jee L.

Submitted to: Engr. Roman Richard

Chosen Data Problem: CO2 Greenhouse Emissions

Source: https://ourworldindata.org

```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
```

```
CarbonEEF = pd.read_csv('Carbon_Emission_Effects_Yearly.csv')
CarbonEEF
```

	Country Name	country_code	Region	Indicator Name	1990	1991	1992	1993	1994	1995	...	2011	2012	
0	Aruba	ABW	Latin America & Caribbean	CO2 emissions (metric tons per capita)	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	
1	Afghanistan	AFG	South Asia	CO2 emissions (metric tons per capita)	0.191745	0.167682	0.095958	0.084721	0.075546	0.068468	...	0.296506	0.259295	0
2	Angola	AGO	Sub-Saharan Africa	CO2 emissions (metric tons per capita)	0.553662	0.544539	0.543557	0.708984	0.836804	0.912141	...	0.985522	0.950696	1
3	Albania	ALB	Europe & Central Asia	CO2 emissions (metric tons per capita)	1.819542	1.242810	0.683700	0.638307	0.645355	0.605436	...	1.669423	1.503240	1
4	Andorra	AND	Europe & Central Asia	CO2 emissions (metric tons per capita)	7.521832	7.235379	6.963079	6.724178	6.541579	6.733479	...	5.850886	5.944654	5
...	...	...	...	...	...	...	...	...	...	...	...	...	...	
210	Samoa	WSM	East Asia & Pacific	CO2 emissions (metric tons per capita)	0.552836	0.609756	0.604266	0.658221	0.592807	0.705675	...	1.066843	1.057703	1
211	Yemen, Rep.	YEM	Middle East & North Africa	CO2 emissions (metric tons per capita)	0.567037	0.690937	0.704793	0.627105	0.653256	0.706081	...	0.963978	0.858491	1
212	South Africa	ZAF	Sub-Saharan Africa	CO2 emissions (metric tons per capita)	6.729799	6.424622	6.175430	6.219194	6.215847	6.378790	...	7.869816	8.077958	8
213	Zambia	ZMB	Sub-Saharan Africa	CO2 emissions (metric tons per capita)	0.340930	0.349232	0.337224	0.289956	0.241270	0.234153	...	0.217497	0.278601	0
214	Zimbabwe	ZWE	Sub-Saharan Africa	CO2 emissions (metric tons per capita)	1.585444	1.713321	1.694416	1.539741	1.417186	1.356619	...	0.884886	0.915735	0

215 rows × 35 columns

```
# Need to Clean and Remove the uncessary things in the dataset
# Create new variable for asean countries
# Source = https://asean.org/member-states/

Countries =  ['Vietnam', 'Indonesia', 'Philippines', 'Thailand', 'Myanmar', 'Cambodia', 'Malaysia', 'Lao PDR', 'Singapore', 'Brunei Darussal



newSet = CarbonEEF[CarbonEEF['Country Name'].isin(Countries)]
newSet.drop(['Region', 'country_code', 'Indicator Name', '2019.1'], axis = 1, inplace = True)
newSet.set_index('Country Name', inplace = True)
newSet
```

```
<ipython-input-33-ec61ed6326d2>:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-cc
  newSet.drop(['Region', 'country_code', 'Indicator Name', '2019.1'], axis = 1, inplace = True)
```

Country Name	1990	1991	1992	1993	1994	1995	1996	1997	1998	1999	...	2010	
Brunei Darussalam	12.600787	12.696839	13.107548	13.951804	14.752014	15.482377	15.855820	16.764862	14.163285	13.580043	...	18.449235	18.6
Indonesia	0.818738	0.879779	0.913552	0.969813	1.027138	1.135811	1.184185	1.287603	1.276950	1.339692	...	1.718202	1.9
Cambodia	0.140381	0.139946	0.140276	0.138405	0.143440	0.137949	0.139307	0.145151	0.162062	0.160687	...	0.359134	0.3
Lao PDR	0.119761	0.125593	0.133323	0.132036	0.135140	0.138245	0.157538	0.164346	0.165222	0.168088	...	0.460861	0.4
Myanmar	0.098705	0.095965	0.100939	0.112147	0.130234	0.158081	0.164446	0.161014	0.173422	0.188605	...	0.160669	0.1
Malaysia	3.029425	3.515130	3.534768	3.748544	3.991489	4.212791	4.693205	4.916803	4.779185	5.050462	...	7.097979	7.0
Philippines	0.663703	0.640614	0.677021	0.717783	0.755491	0.884872	0.941434	1.024087	0.993947	0.937820	...	0.871904	0.8
Singapore	9.507301	9.719041	9.722357	10.913027	11.108355	10.662487	10.493900	10.052586	9.434171	9.548534	...	8.353799	8.6
Thailand	1.577490	1.713538	1.867291	2.103611	2.327972	2.619592	2.905529	2.924404	2.545908	2.629113	...	3.488056	3.4
Vietnam	0.284311	0.285151	0.293580	0.335268	0.362248	0.419167	0.460505	0.529474	0.582085	0.587202	...	1.721201	1.7

10 rows × 30 columns

```
CarbonEEF.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 215 entries, 0 to 214
Data columns (total 35 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   Country Name    215 non-null    object
 1   country_code    215 non-null    object
 2   Region          215 non-null    object
 3   Indicator Name  215 non-null    object
 4   1990            185 non-null    float64
 5   1991            186 non-null    float64
 6   1992            189 non-null    float64
 7   1993            189 non-null    float64
 8   1994            189 non-null    float64
 9   1995            190 non-null    float64
 10  1996            190 non-null    float64
 11  1997            190 non-null    float64
 12  1998            189 non-null    float64
 13  1999            189 non-null    float64
 14  2000            190 non-null    float64
 15  2001            190 non-null    float64
 16  2002            191 non-null    float64
 17  2003            191 non-null    float64
 18  2004            191 non-null    float64
 19  2005            191 non-null    float64
 20  2006            191 non-null    float64
 21  2007            191 non-null    float64
 22  2008            191 non-null    float64
 23  2009            191 non-null    float64
 24  2010            191 non-null    float64
 25  2011            191 non-null    float64
 26  2012            191 non-null    float64
 27  2013            191 non-null    float64
 28  2014            191 non-null    float64
 29  2015            191 non-null    float64
 30  2016            191 non-null    float64
 31  2017            191 non-null    float64
 32  2018            191 non-null    float64
 33  2019            191 non-null    float64
 34  2019.1          191 non-null    float64
dtypes: float64(31), object(4)
memory usage: 58.9+ KB
```

```
u = CarbonEEF['Country Name'].unique()
u
```

```
array(['Aruba', 'Afghanistan', 'Angola', 'Albania', 'Andorra',
       'United Arab Emirates', 'Argentina', 'Armenia', 'American Samoa',
       'Antigua and Barbuda', 'Australia', 'Austria', 'Azerbaijan',
       'Burundi', 'Belgium', 'Benin', 'Burkina Faso', 'Bangladesh',
       'Bulgaria', 'Bahrain', 'Bahamas, The', 'Bosnia and Herzegovina',
       'Belarus', 'Belize', 'Bermuda', 'Bolivia', 'Brazil', 'Barbados',
       'Brunei Darussalam', 'Bhutan', 'Botswana',
       'Central African Republic', 'Canada', 'Switzerland', 'Chile',
       'China', "Cote d'Ivoire", 'Cameroon', 'Congo, Dem. Rep.',
       'Congo, Rep.', 'Colombia', 'Comoros', 'Cabo Verde', 'Costa Rica',
       'Cuba', 'Curacao', 'Cayman Islands', 'Cyprus', 'Czech Republic',
       'Germany', 'Djibouti', 'Dominica', 'Denmark', 'Dominican Republic',
       'Algeria', 'Ecuador', 'Egypt, Arab Rep.', 'Eritrea', 'Spain',
       'Estonia', 'Ethiopia', 'Finland', 'Fiji', 'France',
       'Faroe Islands', 'Micronesia, Fed. Sts.', 'Gabon',
       'United Kingdom', 'Georgia', 'Ghana', 'Gibraltar', 'Guinea',
       'Gambia, The', 'Guinea-Bissau', 'Equatorial Guinea', 'Greece',
       'Grenada', 'Greenland', 'Guatemala', 'Guam', 'Guyana',
       'Hong Kong SAR, China', 'Honduras', 'Croatia', 'Haiti', 'Hungary',
       'Indonesia', 'Isle of Man', 'India', 'Ireland',
       'Iran, Islamic Rep.', 'Iraq', 'Iceland', 'Israel', 'Italy',
       'Jamaica', 'Jordan', 'Japan', 'Kazakhstan', 'Kenya',
       'Kyrgyz Republic', 'Cambodia', 'Kiribati', 'St. Kitts and Nevis',
       'Korea, Rep.', 'Kuwait', 'Lao PDR', 'Lebanon', 'Liberia', 'Libya',
       'St. Lucia', 'Liechtenstein', 'Sri Lanka', 'Lesotho', 'Lithuania',
       'Luxembourg', 'Latvia', 'Macao SAR, China',
       'St. Martin (French part)', 'Morocco', 'Monaco', 'Moldova',
       'Madagascar', 'Maldives', 'Mexico', 'Marshall Islands',
       'North Macedonia', 'Mali', 'Malta', 'Myanmar', 'Montenegro',
       'Mongolia', 'Northern Mariana Islands', 'Mozambique', 'Mauritania',
       'Mauritius', 'Malawi', 'Malaysia', 'Namibia', 'New Caledonia',
       'Niger', 'Nigeria', 'Nicaragua', 'Netherlands', 'Norway', 'Nepal',
       'Nauru', 'New Zealand', 'Oman', 'Pakistan', 'Panama', 'Peru',
       'Philippines', 'Palau', 'Papua New Guinea', 'Poland',
       'Puerto Rico', "Korea, Dem. People's Rep.", 'Portugal', 'Paraguay',
       'West Bank and Gaza', 'French Polynesia', 'Qatar', 'Romania',
       'Russian Federation', 'Rwanda', 'Saudi Arabia', 'Sudan', 'Senegal',
       'Singapore', 'Solomon Islands', 'Sierra Leone', 'El Salvador',
       'San Marino', 'Somalia', 'Serbia', 'South Sudan',
       'Sao Tome and Principe', 'Suriname', 'Slovak Republic', 'Slovenia',
       'Sweden', 'Eswatini', 'Sint Maarten (Dutch part)', 'Seychelles',
       'Syrian Arab Republic', 'Turks and Caicos Islands', 'Chad', 'Togo',
       'Thailand', 'Tajikistan', 'Turkmenistan', 'Timor-Leste', 'Tonga',
       'Trinidad and Tobago', 'Tunisia', 'Turkiye', 'Tuvalu', 'Tanzania',
       'Uganda', 'Ukraine', 'Uruguay', 'United States', 'Uzbekistan',
       'St. Vincent and the Grenadines', 'Venezuela, RB',
       'British Virgin Islands', 'Virgin Islands (U.S.)', 'Vietnam',
       'Vanuatu', 'Samoa', 'Yemen, Rep.', 'South Africa', 'Zambia',
       'Zimbabwe'], dtype=object)
```

```
co2_plot = pd.melt(newSet.reset_index(), id_vars=['Country Name'], value_vars=newSet.columns)
co2_plot.columns = ['Country', 'Year', 'CO2 Emission']
```

```
plt.figure(figsize=(22, 16))
sns.lineplot(x='Year', y='CO2 Emission', hue='Country', data=co2_plot)
plt.xticks(rotation=45)
plt.title('CO2 Emissions per Country Over Time')
plt.xlabel('Yearly Effects of Carbon Emission')
plt.ylabel('CO2 Emission')
plt.show()
```