DS5500_HW1_Mounica

October 7, 2019

[1]:
```
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np

import plotly.express as px
```

Problem 1

What score did you receive? Did any of the answers surprise you? I scored 69% and yes almost half of the answers were surprising. Irrespective the environmental damage we do, the natural disaster damages to mankind life has decreased to less than half. ##### Choose a question from the test, re-state it, and answer it using visualization and summarization. Provide a figure and any relevant output with your answer. - Worked on data of extreme poverty percent of people whose wage is below USD $ 1.90/day.$ - The proportion of people living in extreme poverty as almost reduced by half.

[2]:
```
# read in data
poverty_data = pd.read_csv('extreme_poverty_percent_people_below_190_a_day.csv')
```

[3]:
```
poverty_data.head(5)
```

[3]:
```
     country  1977  1978  1979  1980  1981  1982  1983  1984  1985  ...  2008  \
0    Albania   NaN   NaN   NaN   NaN   NaN   NaN   NaN   NaN   NaN  ...   0.4
1    Algeria   NaN   NaN   NaN   NaN   NaN   NaN   NaN   NaN   NaN  ...   NaN
2     Angola   NaN   NaN   NaN   NaN   NaN   NaN   NaN   NaN   NaN  ...  30.1
3  Argentina   NaN   NaN   NaN   0.4   NaN   NaN   NaN   NaN   NaN  ...   2.6
4    Armenia   NaN   NaN   NaN   NaN   NaN   NaN   NaN   NaN   NaN  ...   1.4

   2009  2010  2011  2012  2013  2014  2015  2016  2017
0   NaN   NaN   NaN   1.1   NaN   NaN   NaN   NaN   NaN
1   NaN   NaN   0.5   NaN   NaN   NaN   NaN   NaN   NaN
2   NaN   NaN   NaN   NaN   NaN   NaN   NaN   NaN   NaN
3   2.6   1.1   0.9   0.8   0.8   0.7   NaN   0.6   NaN
4   1.9   1.9   2.2   1.6   2.2   2.3   1.9   1.8   NaN

[5 rows x 42 columns]
```

[4]:
```
p_d = poverty_data.set_index('country').stack()
```

[5]:
```
p_d = p_d.reset_index()
```

[6]:
```
proc_data = p_d.rename(columns={'level_1':'year'})
```

[7]:
```
# proc_data = proc_data.groupby('year').filter(lambda x: x['year'].count()>25).
↪reset_index().drop(columns=['index'])
```

[8]:
```
proc_data = proc_data.groupby('year').mean().reset_index().rename(columns={0:
↪'mean'})
```

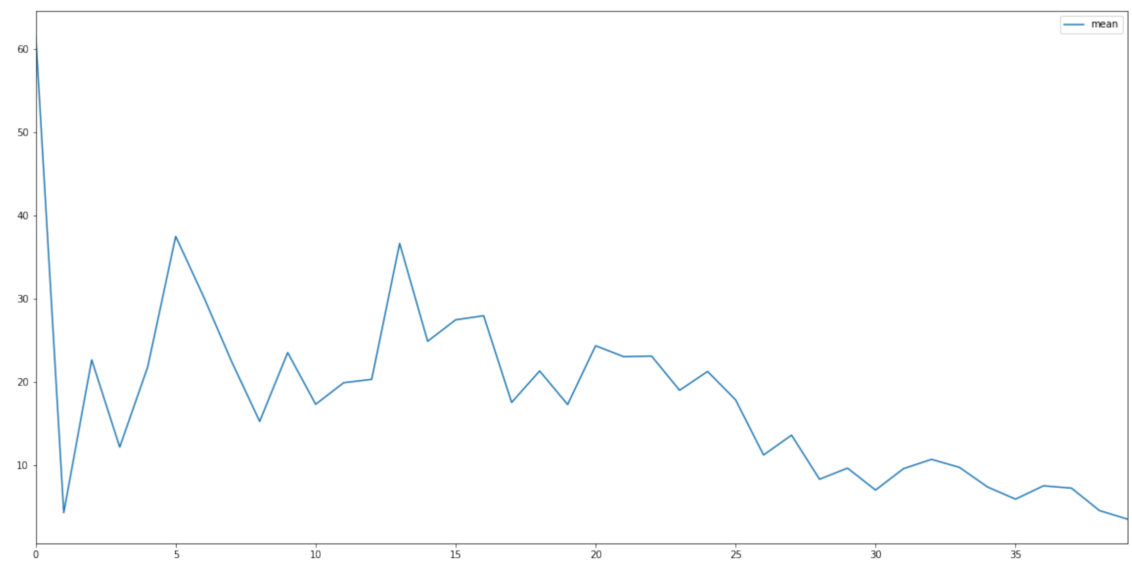
[9]:
```
proc_data.head()
```

[9]:

	year	mean
0	1977	61.600000
1	1979	4.300000
2	1980	22.650000
3	1981	12.166667
4	1982	21.800000

[10]:
```
proc_data.plot(figsize = (20,10))
```

[10]: <matplotlib.axes._subplots.AxesSubplot at 0x1f0dd95cf98>



Problem 2

Interpretation

- The world income is growing each year since 1800.
- But there was a lag inbetween where during the period of 1800 to mid 1900, the growth of income is slow and there might plenty of reasons like technology, low population, low demands and supplies, wars etc.

- There were couple of recision in between mid 1900 and 2010.
- African countries had lower income(GDP per capita) constantly.

[11]:
```
# read in data
income_data = pd.
↪read_csv('ddf--datapoints--gdppercapita_us_inflation_adjusted--by--geo--time.
↪csv')
```

[12]:
```
income_data.head()
```

[12]:
```
   geo  time  gdppercapita_us_inflation_adjusted
0  abw  2010                         24271.94042
1  afg  2002                           364.57057
2  afg  2003                           376.75871
3  afg  2004                           364.09544
4  afg  2005                           389.41636
```

[13]:
```
# income_data = income_data.set_index('country').stack().reset_index().
↪rename(columns={'level_1':'year',0:'income'})
```

[14]:
```
# read in data
geo_data = pd.read_csv('ddf--entities--geo--country.csv')
```

[15]:
```
geo_data = geo_data.drop(columns=['gwid','landlocked','g77_and_oecd_countries',
                                                                                ␣
↪'world_6region','main_religion_2008','gapminder_list','income_groups',
                                                                                ␣
↪'alternative_1','arb1','arb2','arb3','arb4','arb5','arb6','un_state','latitude',
                                                                                ␣
↪'longitude','alternative_2','alternative_3','alternative_4_cdiac','pandg',
                                                                                ␣
↪'god_id','alt_5','upper_case_name','iso3166_1_alpha2','iso3166_1_alpha3',
                                                                                ␣
↪'iso3166_1_numeric','iso3166_2','unicode_region_subtag','is--country'])
```

[16]:
```
geo_data.head()
```

[16]:
```
     country                    name world_4region
0       abkh                Abkhazia        europe
1        afg             Afghanistan          asia
2  akr_a_dhe  Akrotiri and Dhekelia        europe
3        ala                   Åland        europe
4        alb                 Albania        europe
```

[17]:
```
merged_data = pd.merge(income_data, geo_data, left_on='geo', right_on='country')
```

[18]:
```
merged_data = merged_data.drop(columns=['country'])
```

[19]:
```
merged_data = merged_data.rename(columns={'gdppercapita_us_inflation_adjusted':
↪'gdpPercap','name':'country','time':'year','world_4region':'continent'})
```

[20]:
```
merged_data.head()
```

[20]:
```
    geo  year    gdpPercap      country continent
0   abw  2010  24271.94042        Aruba  americas
1   afg  2002    364.57057  Afghanistan      asia
2   afg  2003    376.75871  Afghanistan      asia
3   afg  2004    364.09544  Afghanistan      asia
4   afg  2005    389.41636  Afghanistan      asia
```

[21]:
```
merged_data = merged_data.sort_values(by=['year'])
```

[22]:
```
fig = px.scatter(merged_data, x="gdpPercap", y="year", animation_frame="year",␣
 ↪animation_group="country",
           size="gdpPercap", color="continent", hover_name="country",
           log_x=True, size_max=45, range_x=[300,200000], range_y=[1950,2030])
```

[23]:
```
fig.show()
```

[24]:
```
fig1 = px.area(merged_data, x="year", y="gdpPercap", color="continent",␣
 ↪line_group="country")
fig1.show()
```

Problem 3

Interpretations

- There is strong correlation observed between income (GDP / capita), life expectancy, and child mortality over time.
- There is a positive correlation between GDP and Life expectancy.
- Negative correlation between GDP and child mortality rates and also child mortality rates and life expectancy.

[25]:
```
# read in data
ch_mortality_data = pd.
 ↪read_csv('child_mortality_0_5_year_olds_dying_per_1000_born.csv')
life_exp_data = pd.
 ↪read_csv('ddf--datapoints--life_expectancy_years--by--geo--time.csv')
```

[26]:
```
life_exp_data = life_exp_data.rename(columns={'time':'year'})
```

[27]:
```
life_exp_data.head()
```

[27]:
```
   geo  year  life_expectancy_years
0  abw  1800                  34.42
1  abw  1801                  34.42
2  abw  1802                  34.42
3  abw  1803                  34.42
4  abw  1804                  34.42
```

[28]:
```
ch_mortality_data = ch_mortality_data.set_index('country').stack().reset_index()
```

[29]:
```
ch_mortality_data = ch_mortality_data.rename(columns={'level_1':'year',0:
 ↪'mortality rate'})
```

[30]:
```python
ch_mortality_data['year'] = ch_mortality_data['year'].astype(str).astype(int)
```

[31]:
```python
ch_mortality_data.head()
```

[31]:
```
       country  year  mortality rate
0  Afghanistan  1800           469.0
1  Afghanistan  1801           469.0
2  Afghanistan  1802           469.0
3  Afghanistan  1803           469.0
4  Afghanistan  1804           469.0
```

[32]:
```python
merged_data.head()
```

[32]:
```
      geo  year     gdpPercap           country continent
6631  png  1960    1135.86589  Papua New Guinea      asia
6095  nld  1960   16354.54003       Netherlands    europe
2375  dza  1960    2466.03824           Algeria    africa
2259  dnk  1960   21075.59952           Denmark    europe
1004  bmu  1960   27838.39304           Bermuda  americas
```

[33]:
```python
merged_data2 = pd.merge(life_exp_data,
                 merged_data[['country','geo','gdpPercap','continent']],
                 on='geo')
```

[34]:
```python
merged_data2_1 = merged_data2.groupby(['continent', 'year','country']).mean().
 ↪reset_index()
```

[35]:
```python
merged_data2_1.head()
```

[35]:
```
  continent  year       country  life_expectancy_years    gdpPercap
0    africa  1800       Algeria                  28.82  3474.401881
1    africa  1800        Angola                  26.98  2428.382828
2    africa  1800         Benin                  31.00   645.156246
3    africa  1800      Botswana                  33.60  3427.721718
4    africa  1800  Burkina Faso                  29.20   391.407213
```

[36]:
```python
prob3_data = pd.merge(merged_data2,
                 ch_mortality_data[['country','mortality rate']],
                 on='country')
```

[37]:
```python
dt = prob3_data
dt.head()
```

[37]:
```
   geo  year  life_expectancy_years      country  gdpPercap continent  \
0  afg  1800                  28.21  Afghanistan  364.57057      asia
1  afg  1800                  28.21  Afghanistan  364.57057      asia
2  afg  1800                  28.21  Afghanistan  364.57057      asia
3  afg  1800                  28.21  Afghanistan  364.57057      asia
4  afg  1800                  28.21  Afghanistan  364.57057      asia

   mortality rate
0           469.0
```

```
1          469.0
2          469.0
3          469.0
4          469.0
```

[38]:
```python
cont_group = dt.groupby(['continent', 'year','country']).mean().reset_index()
```

[39]:
```python
cont_group1 = dt.groupby(['continent', 'year']).mean().reset_index()
```

[40]:
```python
cont_group.head()
```

[40]:
```
  continent  year       country  life_expectancy_years    gdpPercap  \
0    africa  1800       Algeria                  28.82  3474.401881
1    africa  1800        Angola                  26.98  2428.382828
2    africa  1800         Benin                  31.00   645.156246
3    africa  1800      Botswana                  33.60  3427.721718
4    africa  1800  Burkina Faso                  29.20   391.407213

   mortality rate
0      343.880365
1      394.377626
2      355.227397
3      295.155708
4      380.451598
```

[41]:
```python
cont_group1.head()
```

[41]:
```
  continent  year  life_expectancy_years    gdpPercap  mortality rate
0    africa  1800              30.645566  1822.412373      342.614743
1    africa  1801              30.491868  1822.412373      342.614743
2    africa  1802              30.491993  1822.412373      342.614743
3    africa  1803              30.645940  1822.412373      342.614743
4    africa  1804              30.646065  1822.412373      342.614743
```

[42]:
```python
dt_corr = dt[['mortality rate', 'life_expectancy_years', 'gdpPercap']].corr()
dt_corr.style.background_gradient(cmap='coolwarm').set_precision(3)
```

[42]: <pandas.io.formats.style.Styler at 0x1f0e08134e0>

[43]:
```python
fig2 = px.line(cont_group, x="year", y="life_expectancy_years", title='Life␣
↪expectancy across the continents',
               color='continent',range_x=[1790,2020], range_y=[0,90])
fig2.show()
```

[44]:
```python
fig3 = px.line(cont_group1, x="year", y="mortality rate", title='Life␣
↪expectancy across the continents',color='continent')
fig3.show()
```

[45]:
```python
fig4 = px.line(cont_group1, x="year", y="gdpPercap", title='Life expectancy␣
↪across the continents',color='continent')
fig4.show()
```

Problem 4

Choose two variables you have not investigated yet, and visualize their distributions, their relationship with each other, and how these change over time. The female school data represents mean years in school spent by women of age 25 to 34 years. Also taking the population of female from age 20 to 39. Now calculating the female count who have attended school during 25 to 34 years and plotting it.It is pretty much almost 60% of the female population have attended schools.

[46]:
```
# read in data
female_sch_data = pd.read_csv('mean_years_in_school_women_25_to_34_years.csv')
female_pop_data = pd.read_csv('population_aged_20_39_years_female_percent.csv')
```

[47]:
```
female_sch_data1 = female_sch_data.set_index('country').stack().reset_index()
female_sch_data1 = female_sch_data1.rename(columns={'level_1':'year',0:'mean'})
```

[48]:
```
female_sch_data1['year'] = female_sch_data1['year'].astype(str).astype(int)
```

[49]:
```
female_sch_data1.head()
```

[49]:

	country	year	mean
0	Afghanistan	1970	0.21
1	Afghanistan	1971	0.22
2	Afghanistan	1972	0.22
3	Afghanistan	1973	0.23
4	Afghanistan	1974	0.24

[50]:
```
female_pop_data1 = female_pop_data.set_index('country').stack().reset_index()
female_pop_data1 = female_pop_data1.rename(columns={'level_1':'year',0:'ratio'})
```

[51]:
```
female_pop_data1['year'] = female_pop_data1['year'].astype(str).astype(int)
```

[52]:
```
female_pop_data1.head()
```

[52]:

	country	year	ratio
0	Afghanistan	1950	27.9
1	Afghanistan	1955	28.0
2	Afghanistan	1960	28.0
3	Afghanistan	1965	27.8
4	Afghanistan	1970	28.0

[53]:
```
prob4_data = pd.merge(female_sch_data1,
                      female_pop_data1[['country','ratio']],
                      on='country')
```

[54]:
```
dff = prob4_data
grouped = dff.groupby(['country', 'year']).mean().reset_index()
```

[55]:
```
grouped.head()
```

[55]:

	country	year	mean	ratio
0	Afghanistan	1970	0.21	28.403226
1	Afghanistan	1971	0.22	28.403226
2	Afghanistan	1972	0.22	28.403226

```
3  Afghanistan  1973  0.23  28.403226
4  Afghanistan  1974  0.24  28.403226
```

[56]:
```
# geo_data
geo_data1 = geo_data.rename(columns={'country':'geo','name':'country'})
geo_data1.head()
```

[56]:
```
        geo                 country world_4region
0      abkh                Abkhazia        europe
1       afg             Afghanistan          asia
2  akr_a_dhe  Akrotiri and Dhekelia        europe
3       ala                   Åland        europe
4       alb                 Albania        europe
```

[57]:
```
cons = pd.merge(grouped,
                 geo_data1[['country','world_4region']],
                 on='country')
cons.head()
```

[57]:
```
       country  year  mean      ratio world_4region
0  Afghanistan  1970  0.21  28.403226          asia
1  Afghanistan  1971  0.22  28.403226          asia
2  Afghanistan  1972  0.22  28.403226          asia
3  Afghanistan  1973  0.23  28.403226          asia
4  Afghanistan  1974  0.24  28.403226          asia
```

[58]:
```
tot_pop_data = pd.read_csv('population_total.csv')
```

[59]:
```
tot_pop_data = tot_pop_data.set_index('country').stack().reset_index()
tot_pop_data = tot_pop_data.rename(columns={'level_1':'year',0:'count'})
```

[60]:
```
tot_pop_data['year'] = tot_pop_data['year'].astype(str).astype(int)
```

[61]:
```
tot_pop_data.head()
```

[61]:
```
       country  year    count
0  Afghanistan  1800  3280000
1  Afghanistan  1801  3280000
2  Afghanistan  1802  3280000
3  Afghanistan  1803  3280000
4  Afghanistan  1804  3280000
```

[62]:
```
cons_group = pd.merge(cons,
                 tot_pop_data[['country','count']],
                 on='country')


cons_group = cons_group.groupby(['country', 'year', 'world_4region']).mean().
 ↪reset_index()
```

[63]:
```
cons_group['female_total'] = (cons_group['count'] * cons_group['ratio']).
 ↪astype(int)
```

[64]:
```python
cons_group.head()
```

[64]:
```
       country  year world_4region  mean      ratio         count  \
0  Afghanistan  1970          asia  0.21  28.403226  2.300429e+07
1  Afghanistan  1971          asia  0.22  28.403226  2.300429e+07
2  Afghanistan  1972          asia  0.22  28.403226  2.300429e+07
3  Afghanistan  1973          asia  0.23  28.403226  2.300429e+07
4  Afghanistan  1974          asia  0.24  28.403226  2.300429e+07

  female_total
0    653395921
1    653395921
2    653395921
3    653395921
4    653395921
```

[65]:
```python
cons_group = cons_group[(cons_group.year >= 2005) & (cons_group.year <= 2015)]
```

[66]:
```python
fig5 = px.line(cons_group, x="year", y="ratio", title='Mean years of WOMEN in␣
 ↪school',
               color='world_4region',range_x=[2004,2016], range_y=[20,40])
fig5.show()
```

[67]:
```python
fig6 = px.line(cons_group, x="year", y="count",
               color='world_4region',range_x=[2004,2016])
fig6.show()
```

Problem 5

Did you use static or interactive plots to answer the previous problems? A mix of both static and interative plots. ##### Explore the data using the interactive visualization tools at https://www.gapminder.org/tools, and watch the TED talk "The best stats you've ever seen" at https://www.youtube.com/watch?v=hVimVzgtD6w. Tried visualization tools in gapminder site and watched the video as well. ##### Discuss the advantages, disadvantages, and relative usefulness of using interactive/dynamic visualizations versus static visualizations. ###### Static plots - They are still, they can be downloaded and saved, good for simple data with small range

Dynamic plots

- Animated plots with interaction involved, can customize the data we want to see in the plot, best for large range of voluminous data

Refernces

- Reference: https://plot.ly/python/plotly-express/