UN Data Exploration

Tomo Umer, MS

Nashville Software School - Data Science 6

10/1/2022

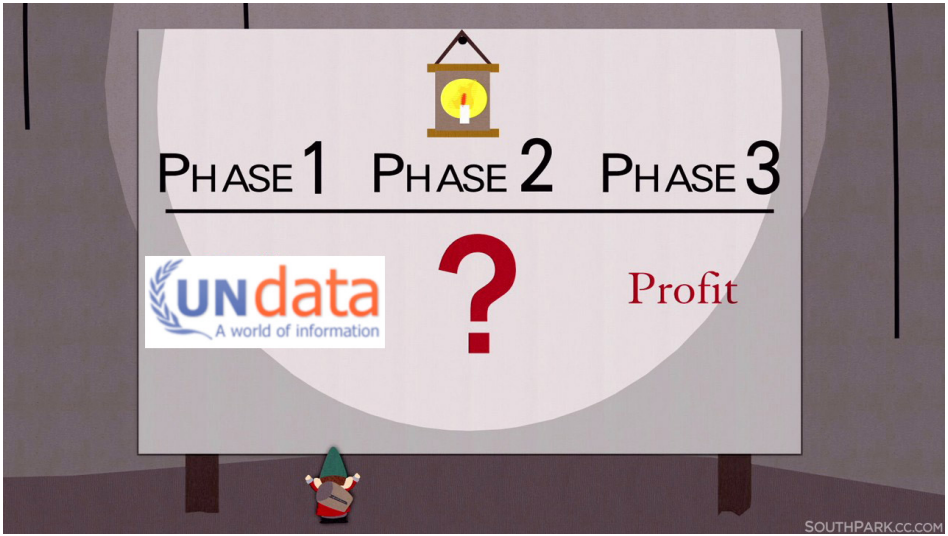
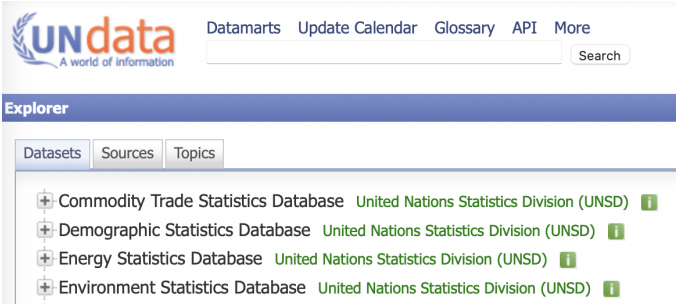


Figure: South Park

In all seriousness - where to start?



- Got a look at various tables - some in MS Excel only
- Starting idea: check tables that could be compared to GDP
- Downloaded a few, started with Commodity Trade (ALL)

First problem: Country names!!!

```
trade['Country'] = (
    trade['Country']
        .str.replace('USA', 'United States')
        .str.replace('Brunei Darussalam', 'Brunei')
        .str.replace(r'Bolivia.+', 'Bolivia', regex=True)
)
```

USA was also United States and United States of America!
Bolivia (Plurinational State of) - couldn't get it to work until regex!

```
g1 = sns.FacetGrid(
        gdp_trade.loc[gdp_trade['Country'].isin(['Slovenia', 'Italy', '
                                                 United States', 'China'])],
        col='Country',
        hue='Flow',
        height=6)
g1.map(sns.scatterplot, 'Year', 'Trade (USD)').add_legend()
```

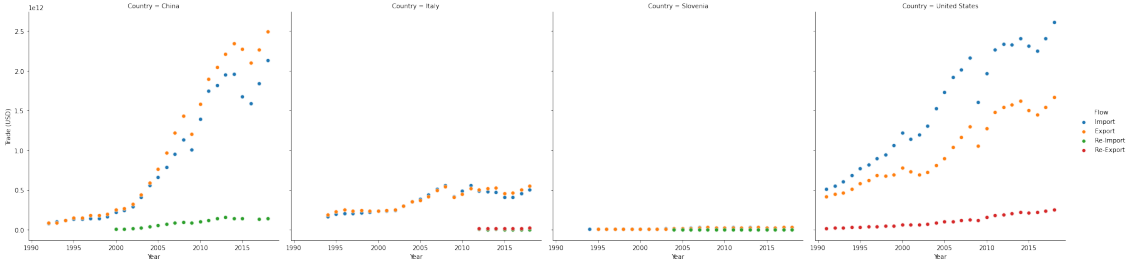


Figure: Trade values in time - China, Ita, Slo, U.S.

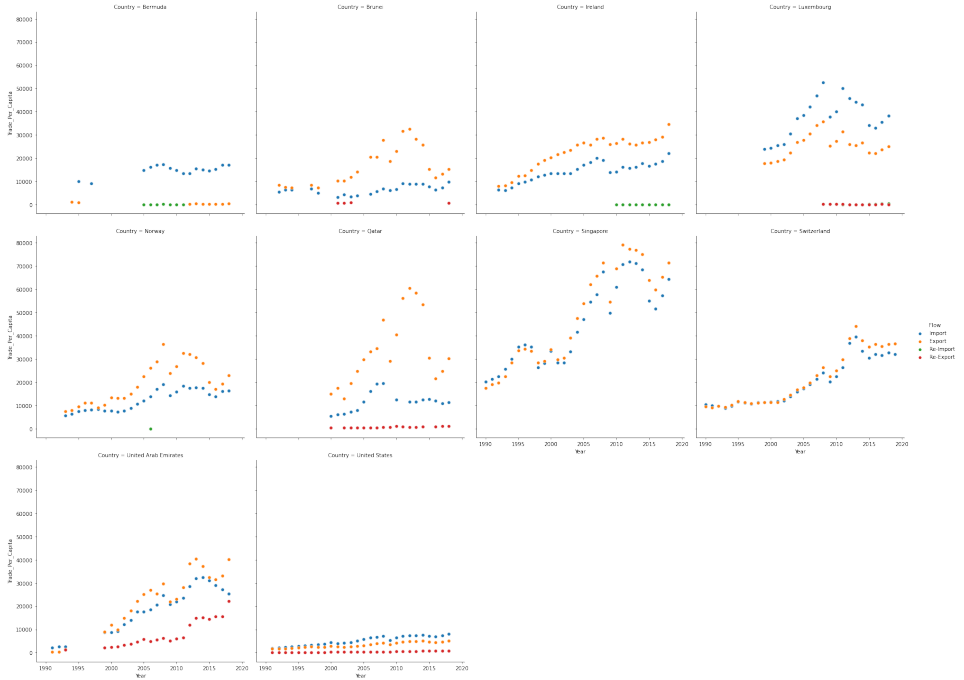
Second problem: Huge disparity in overall Trade values

- Capture top10 (or bot 10) countries based on GDP in most recent year 2019
- Looking at them both and decided to not capture bot 10
- In top 10, rejected Macau SAR, since it's a special administrative region of China


Country = Bermuda
Country = Brunei
Country = Ireland
Country = Luxembourg
Country = Norway
Country = Qatar
Country = Singapore
Country = Switzerland
Country = United Arab Emirates
Country = United States
1e12
Trade (USD)
Year
Flow
Import
Export
Re-Import
Re-Export

- The problem didn't go away - need to normalize Trade per Capita
- World Population Prospects: The 2019 Revision
  - total population, both sexes combined (thousands)
- File was too big, had to filter on the website
- After a inner merge, simply divide Trade column by Population
* Error on my part - didn't catch that population was expressed in thousands

All of this finalized my idea to look at how the GDP changes depending on trade


Country = Bermuda
Country = Brunei
Country = Ireland
Country = Luxembourg
Country = Norway
Country = Qatar
Country = Singapore
Country = Switzerland
Country = United Arab Emirates
Country = United States
trade_per_capita
Year
Flow
Import
Export
Re-Import
Re-Export


Country = Bermuda
Country = Brunei
Country = Ireland
Country = Luxembourg
Country = Norway
Country = Qatar
Country = Singapore
Country = Switzerland
Country = United Arab Emirates
Country = United States
GDP_Per_Capita
Trade_Per_Capita
Flow
Import
Export
Re-Import
Re-Export

Third problem: What if I want to have import, export, re-import, re-export in the same line? (in order to be able to get a surplus/deficit)

GDP_Per_Capita	Flow	Trade (USD)	Population	Trade_Per_Capita
2033.779002	Import	7.406590e+09	37171.921	199252.288434
2033.779002	Export	8.845045e+08	37171.921	23794.963310
2033.779002	Re-Export	9.263097e+06	37171.921	249.196086

I did not want this (sub-columns):

GDP_Per_Capita	Population	Trade (USD)				Trade_Per_Capita			
		Export	Import	Re-Export	Re-Import	Export	Import	Re-Export	Re-Import
1484.114729	27722.276	5.400656e+08	3.019860e+09	NaN	NaN	19481.286241	108932.618988	NaN	NaN
1758.904043	28394.813	4.034410e+08	3.336435e+09	NaN	NaN	14208.264235	117501.558506	NaN	NaN
1957.029338	29185.507	3.884836e+08	5.154250e+09	NaN	NaN	13310.840720	176603.060793	NaN	NaN

Solution:

```
gdp_trade_pop = (
    gdp_trade_pop
        .pivot(
            index=['Country', 'Year', 'GDP_Per_Capita', 'Population'],
            columns='Flow',
            values='Trade (USD)'
         )
        .reset_index()
        .rename_axis(None, axis=1)
        .fillna(0)
)
```

This only works with one sets of values (Trade) and not the normalized one. Oh well!


Country = Bermuda
Country = Brunei
Country = Ireland
Country = Luxembourg
Country = Norway
Country = Qatar
Country = Singapore
Country = Switzerland
Country = United Arab Emirates
Country = United States
GDP_Per_Capita
Surplus_Deficit_Per_Capita
Year
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019

Fourth problem: because of inner merges, not only did I lose some countries, but also years (did not realize the trade database has so much missing data)!

- Back to the beginning!
- Merge GDP with continents and find highest and lowest GDP for each continent in 2019
  Had to remove Kosovo (not present in Trade), Haiti (only until 2017)
  Also removed Macau SAR, China as previously
- Did some additional verifications every step, to not lose any of the countries!
- Merge with Population
- Left merge with trade, Pivot, normalize trade values
- Ended up with **filtered_gdp_cont_pop_trade** what a beautiful dataframe name!
- fun with plots!! (additional and interactive ones available on GitHub)

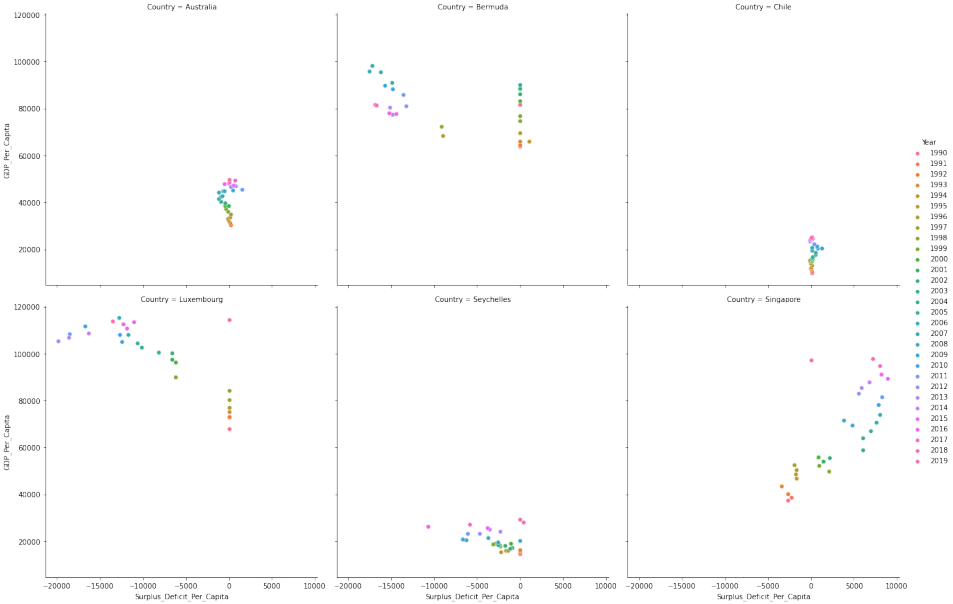


Figure: Countries with highest GDP in each continent

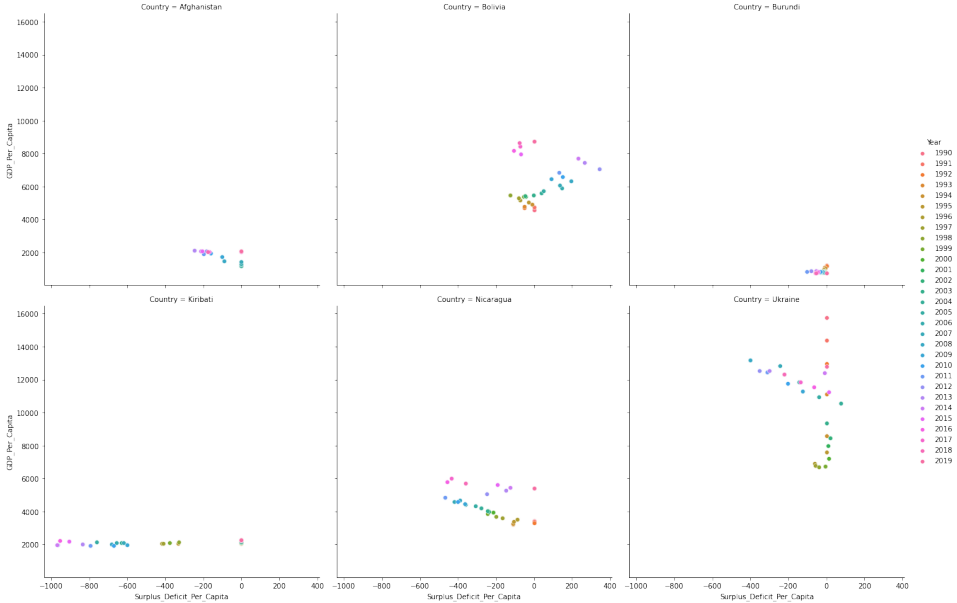


Figure: Countries with lowest GDP in each continent

```
def countries_plot(
  dataset, country_list, nplots_row, plot_x, plot_y, hue_var=None) :
    """ Function to Plot 2 variables on separate plots for each Country """
    g = (
        sns.FacetGrid(
            dataset
                .loc[dataset['Country']
                .isin(country_list)],
            col='Country',
            hue=hue_var,
            height=6,
            col_wrap=nplots_row
        )
        .map(
            sns.scatterplot,
            plot_x,
            plot_y
        )
        .add_legend()
    );
```