Undergraduate Economic Review

Volume 16 | Issue 1 Article 6

2019

A Data Analysis of the World Happiness Index and its Relation to the North-South Divide

Charles Alba
Pennsylvania State University - Main Campus, cja5553@psu.edu

A Data Analysis of the World Happiness Index and its Relation to the North-South Divide

Abstract

In this document, we perform a detailed data analysis on the World Happiness Report with its relation to the socio-economic North-South Divide. In order to do so, we perform some extensive data cleaning and analysis before querying on the World Happiness Report. Our results based on Hypothesis Testing determines the happiness of the Global North is greater than that of the Global South. Furthermore, our queries show that the mean happiness score for the Global North significantly outweighing that of the South. Likewise, the 10 'Happiest' nations all belong to the Global North whereas the 10 'least happy' nations belong to the Global South.

Keywords

World Happiness Index, Data, SQLite, Python

Cover Page Footnote

This document was modified from a final project which originated from the class "DS220: Data Management". I would like to thank and acknowledge the course Professor, Dr. Marc Rigas (Ph.D.) for allowing me to originate this publication from the final project of his course.

Introduction

The aim of this document is to determine if the Socio-economic North-South divide holds true for happiness. In order to do so, we retrieve the data from Kaggle (https://www.kaggle.com/unsdsn/world-happiness) and query the latest 2017 World Happiness Report after doing some data cleaning on this dataset with the help of the 2016 and 2015 datasets. This was completed with SQLite, Python and Jupyter Notebook. Finally the hypothesis testing was completing using R after exporting the cleaned and updated dataset.

Background Information

The World Happiness Report is an annual report on the state of global happiness and is currently participated by 155 countries [1]. The first report was released in 2012 following the adoption of resolution 65/309 titled "Happiness: Towards a Holistic Definition of Development" by the UN General Assembly [2].
Since the publication of its first report, its survey methodology has remained the same: The reports are based on answers to a series of real-life questions asked in the poll. This is known as Cantril ladder where respondents are asked to think of a ladder, with the best possible life for them being a 10, and the worst possible life being a 0[3]. More details would be discussed in the methodology.
Since its inception, multiple nations, institutions and organizations have utilized the World Happiness Report in a variety of ways, from economic research between economic growth and happiness [4][5] to its relation based on geography [6] legislation of public policies based on the relationship between happiness and a variety of social factors[7][8][9].

Happiness and its relation to the North-South Divide

We want to determine if the North-South divide holds true in relation to Happiness. In summary, the North-South divide is a socio-economic divide which proves that countries in the Global North are economically wealthier than countries in the Global South [10]. Since the cold-war era, for "more than a generation, this North-South divide was central to the explanation of world poverty" (723, Therien) [12]. Hence, we want to see if such also holds true for happiness. Therefore, we will perform queries using the latest data set, the 2017 data set, to see if this phenomenon holds true for happiness.

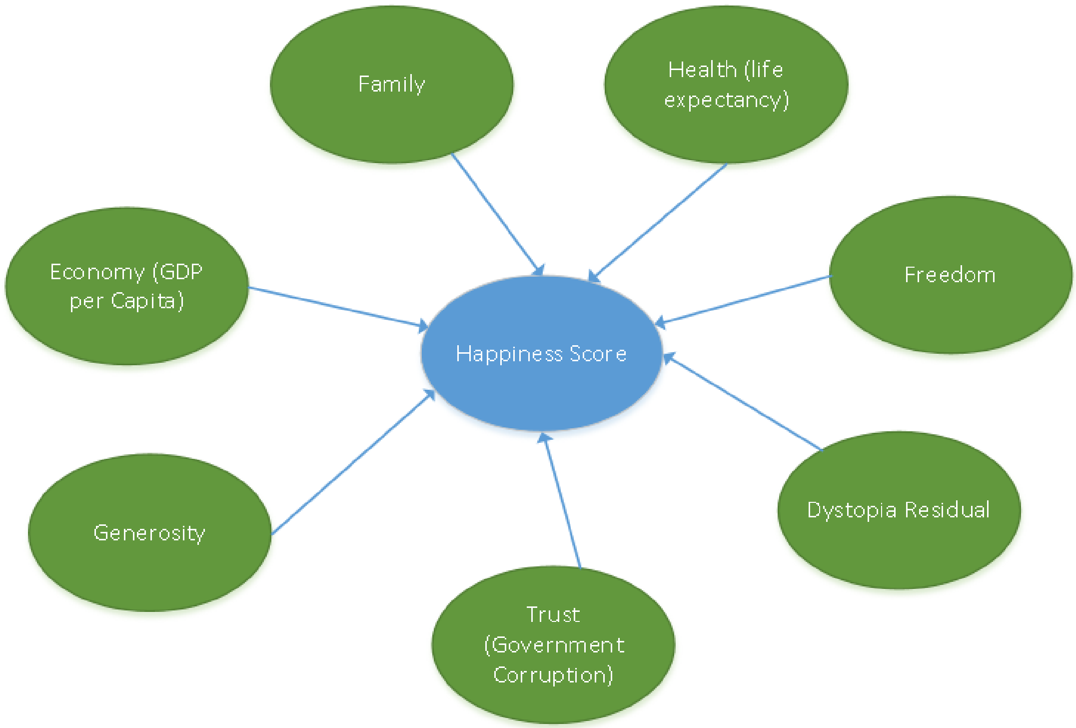
Methods

Description of database

Despite mentioning above that we want to use only the 2017 data set because it's the latest dataset made available to us, we could not help but notice at first glance that most crucial component for determining the Global North from the Global South - the 'Region' attribute - was not available in the 2017 data set. However, it was available in the 2016 and 2015 dataset. As a result, we would now have to explore the datasets across all 3 years because, at some point we have to refer to the previous year's data set in order to determine the regions for the nations (since they are geographically determined and manually inputted) for the 2017 dataset. The Kaggle dataset consists of tables from datasets for the World Happiness Report for years 2015, 2016 and 2017. While different tables across varied years consists of differing attributes, there are 10 attributes that have remained consistent over the tables of 3 different years. These attributes are: Country, Happiness Rank, Happiness Score, Economy (GDP Per Capita), Family, Health (Life expectancy), Freedom, Generosity, Trust (Government Corruption) and Dystopia Residual. The latter 8 attributes remain consistent across all tables because the Happiness Score is derived as follows:

$$\begin{aligned} &\mathit{Happiness\ Score} \\ &\quad = \mathit{Economy}\ (\mathit{GDP\ Per\ Capita}) + \mathit{Family} \\ &\quad + \mathit{Health}\ (\mathit{Life\ expectancy}) + \\ &\quad \mathit{Freedom} + \mathit{Generosity} + \mathit{Trust}\ (\mathit{Government\ Corruption}) \\ &\quad + \mathit{Dystopia\ Residual} \end{aligned}$$

This can be furthered illustrated in the table below:



As mentioned earlier, the reports are based on answers to a series of real life questions asked in the poll, varying across 6 factors - Economy, Family support, life expectancy, freedom, generosity and absence of corruption. This is known as Cantril ladder where respondents are asked to think of a ladder, with the best possible life for them being a 10, and the worst possible life being a 0 [3]. These scores are from nationally representative and utilize the Gallup weights to make the estimates representative. The respective attributes estimate the extent to which each of six factors contribute to making life evaluations higher in each country than they are in Dystopia, a hypothetical country that has values equal to the world's lowest national averages for each of the six factors. Hence, the values reflected in the 6 attributes are also known as residuals, reflecting the extent to which the six variables contribute to the Happiness Score. These 6 attributes are therefore computed and calculated with the help of external data before being reflected in the data tables. Finally, The Dystopia Residual metric actually is the Dystopia Happiness Score (1.85) added with the Residual value [11]. The sum of these 7 residuals give us the Happiness Score as illustrated in the formula and diagram above. Based on the Happiness Score, the Happiness Rank is retrieved and computed based on which country has the higher Happiness Score.

With the explanations of the common attributes mentioned above, we shall proceed to explain attributes which are not common amongst the 2015, 2016 and 2017 tables.

- 2015 Data Table:
  Region: These are entered manually and reflects the geographical classification based on the reflected country
  Standard Error: The standard error of the Happiness Index, statistically computed before being reflected in the data table
- 2016 Data Table:
  Region: Refer to above
  Lower Confidence Interval: Lower limit in the 2 sided confidence interval, statistically computed
  Upper Confidence Interval: Upper limit in the 2 sided confidence interval, statistically computed
- 2017 Data Table:
  Whisker.High: These Whiskers denote the higher confidence region for the estimates
  Whisker.Low: These whiskers denote the lower confidence region for the estimates.

Bringing the Data into the Database and challenges encountered

SQLite allows us to easily import the 2015, 2016 and 2017 .csv files into SQLite. We ran a few quick queries and discovered that when trying to insert a region column in the 2017 dataset by referring to the 2016 dataset through a JOIN statement, we now have 150 rows as compared to 155 original rows (Refer to appendix G). This calls for more data cleaning and exploration across all 3 years. Given this, thanks to SQLite we can choose the Primary Key using SQLite without the need to write lengthy code. For the World Happiness Report, we identified the country as the Primary Key for each of the tables as a country's name is unique and no country could be identically named. Refer to appendix A for detailed screenshots of how the data was loaded into the database.

Data Cleaning

In terms of data cleaning, the first thing that catches one's eye is the fact that the .csv files are named ‘2015’, ‘2016’ and ‘2017’ respectively. Such are not desirable table names for performing SQL queries. Therefore, we have to rename these tables ‘data2015’, ‘data2016’ and ‘data2017’ respectively. This could be easily done as SQLite allows us to rename the table names differently from that of the .csv files when importing the CSV files to SQLite (refer to appendix B)
Another thing that catches our attention is the fact that most of the attribute titles in the 2017 table contains a period (.) as in the attribute titles.

This is not very desirable for our future queries as SQL does not respond well to such titles. Hence, we have to change replace all '.' with '_'. This could be easily accomplished by using the modify function in SQLite which will allow us to rename our attribute titles with ease (refer to appendix B).
Now that we have our database (db file), we could utilise Jupyter Notebook to carry on with some data cleaning with Python before proceeding to run queries. (refer to appendix C to see how the db file was loaded into Jupyter Notebook)
On top of the reduction of rows in data2017 when doing a JOIN mentioned above, we also could not help but notice we had an inconsistent amount of rows for 2015, 2016 and 2017 tables respectively. We also want to ensure that the names in 'Country' attribute was consistent across all years. Hence, we had to perform queries which showed us the countries that were present in a particular year but were missing in another year. (We provide one complete example here, Refer to Appendix D for the entire code)
For example, we perform the following query to find which countries were present in the 2015 table but were absent in 2016 table:

```
query = '''SELECT A.Country FROM data2015 A WHERE Country NOT
IN(SELECT Country FROM Data2016)'''
uncommon2015to2016 = pd.read_sql_query(query, conn)
uncommon2015to2016
```

We receive an output of the following, which shows us the countries **present in the 2015 table but were absent in the 2016 table**.

Country
0 Central African Republic
1 Djibouti
2 Lesotho
3 Mozambique
4 Oman
5 Somaliland region
6 Swaziland

Likewise, to find countries **present in the 2016 table but were absent in the 2015 table**, we have: (refer to appendix D for the entire code)

Country
0 Belize
1 Namibia
2 Puerto Rico
3 Somalia

4 Somaliland Region
5 South Sudan

From this we can identify that one cause of inconsistency across the tables was the fact that country "Somaliland Region" was named "Somaliland region" in 2015 but "Somaliland Region" in 2016. (The capitalization of the word "region" made a difference)

We also find such similar discrepancies when running queries to find the list of countries **present in the 2016 table but absent in the 2017 table:** (Refer to appendix D for the entire code)

Country
0 Comoros
1 Hong Kong
2 Laos
3 Puerto Rico
4 Somaliland Region
5 Suriname
6 Taiwan

Likewise to find countries **present in the 2017 table but absent in the 2016 table**:
(Refer to appendix D for the entire code)

Country
0 Central African Republic
1 Hong Kong S.A.R., China
2 Lesotho
3 Mozambique
4 Taiwan Province of China

We find that 2 countries were named differently in 2017 as compared to 2016. Countries "Hong Kong" and "Taiwan" in the 2016 table were named "Hong Kong S.A.R., China" and "Taiwan Province of China" respectively in 2017.

As for the remaining countries, we can confirm that they were present in a particular year but absent in another due to the decision to not participate in the report in the selected years.

Therefore, as part of our data cleaning process, we have to change the names of these countries which were named differently in the respective tables. Firstly, we have to identify if these countries were present in the other tables. If so, we would standardize by using the names present in the other tables. In order to do so, we ran queries on Jupyter Notebook. We ran the first example, to find if country “Somaliland Region” was present in the 2017 data: (refer to Appendix E for the full code)

```
query = '''SELECT Country FROM data2017 WHERE country LIKE
'%Somaliland%' '''
Somaliland = pd.read_sql_query(query, conn)
Somaliland

Country
```

Similarly we ran queries to determine the naming of ‘Taiwan’ and ‘Hong Kong’ under attribute ‘Country’ in the 2017 data table. (refer to Appendix E for the full code)

```
Country
0 Hong Kong
1 Taiwan
```

Hence, we can conclude that ‘Somaliland Region’ was not present in the 2017 data but ‘Hong Kong’ and ‘Taiwan’ were present in the 2015 data. Therefore to ensure consistency, we shall rename “Somaliland region” in the 2015 table to “Somaliland Region”, following the standard capitalization of starting letters in the words words for attribute “country”. Likewise to ensure consistency in 3 tables across the 3 different years, we shall rename “Hong Kong S.A.R., China” and “Taiwan Province of China” in the 2017 table to “Hong Kong” and “Taiwan” respectively. We do so by running the following code:

```
c.execute('''UPDATE data2017
SET Country = "Hong Kong"
WHERE Country = "Hong Kong S.A.R., China"''')

<sqlite3.Cursor at 0x1d15d5268f0>
```

We provided “Hong Kong” as an example, refer to Appendix F the complete code for the renaming of all 3 countries in their respective tables.

Hence, we have to use collective data of the countries and their respective regions from both the 2015 and 2016 data tables. We would utilize this collective data to fill the newly inserted ‘Region’ column in the 2017 data table. We do so as follow:

```
# create region table
c.execute(''' ALTER TABLE data2017
ADD Region varchar(255)''')
```

```
<sqlite3.Cursor at 0x1d15d5268f0>
```

```
# insert the columns for ‘region’
c.execute('''UPDATE data2017
        SET region=(SELECT Region
        FROM (SELECT Country, Region FROM data2015 UNION SELECT Country,
Region FROM data2016) AS A
        WHERE A.country=data2017.country)''')
```

```
<sqlite3.Cursor at 0x1d15d5268f0>
```

The output gives us 155 rows, which is the same number of rows as the original data2017 data table. With all the nations and their respective regions accounted for, we can now proceed to perform queries to find the Happiness Index’s relation to the North-South Divide.

Results

Inserting and determining the Global North and South

Before we could use queries to determine if the North-South divide holds true to happiness as it does from a socio-economic perspective, we first have to determine which nations and regions are classified as North or South respectively. We base our Global North and the Global South from various reliable sources including the World Atlas [14] and the Longman Dictionary [10]. Hence, the following regions and countries have been identified as the Global North: North America, all European Union Nations, Australia and New Zealand as well as developed Asian Nations South Korea and Japan. The rest are classified as the Global South [10][14].

Because we want to use only the latest data, we shall stick to table "data2017" containing only the 2017 data.

With this, we create a new attribute "North_or_south" on the table "data2017" and insert its relevant data - 'North' or 'South' (refer to appendix G for the full code)

Performing Queries to address our question

Now that we have our final table which includes all the necessary information including the Global North and the Global South, we can now perform queries to determine if this socio-economic phenomenon holds true to happiness.

First we find the average of the North and South respectively. We obtain the following queries and results:

```
for row in c.execute(''' SELECT North_or_south||'(average score):
'||AVG(Happiness_score) FROM data2017 GROUP BY North_or_south '''):
   print (row)
```

```
('North(average score): 6.04162500585829',)
('South(average score): 4.96507070522116',)
```

We find that the Global North's average Happiness Index Score outnumbers the Global South's average score by 1.08 points.

Likewise, we perform queries to find the "happiest" countries based on their Happiness Score's ranking:

```
for row in c.execute(''' SELECT 'Country: '||Country||'|Global North/South: '||
North_or_south||'|Score: '|| Happiness_Score FROM data2017 ORDER BY
Happiness_score DESC LIMIT 10'''):
    print (row)
```

```
('Country: Norway|Global North/South: North|Score: 7.53700017929077',)
('Country: Denmark|Global North/South: North|Score: 7.52199983596802',)
('Country: Iceland|Global North/South: North|Score: 7.50400018692017',)
('Country: Switzerland|Global North/South: North|Score: 7.49399995803833',)
('Country: Finland|Global North/South: North|Score: 7.4689998626709',)
('Country: Netherlands|Global North/South: North|Score: 7.3769998550415',)
('Country: Canada|Global North/South: North|Score: 7.31599998474121',)
('Country: New Zealand|Global North/South: North|Score: 7.31400012969971',)
('Country: Sweden|Global North/South: North|Score: 7.28399991989136',)
('Country: Australia|Global North/South: North|Score: 7.28399991989136',)
```

We find that the 10 ‘happiest’ countries belong to the Global North!

Similarly, we perform queries to find the “least happy” countries based on their Happiness Score’s ranking:

```
for row in c.execute(''' SELECT 'Country: '||Country||'|Global North/South: '||
North_or_south||'|Score: '|| Happiness_Score FROM data2017 ORDER BY
Happiness_score ASC LIMIT 10'''):
    print (row)
```

```
('Country: Central African Republic|Global North/South: South|Score:
2.69300007820129',)
('Country: Burundi|Global North/South: South|Score: 2.90499997138977',)
('Country: Tanzania|Global North/South: South|Score: 3.34899997711182',)
('Country: Syria|Global North/South: South|Score: 3.46199989318848',)
('Country: Rwanda|Global North/South: South|Score: 3.47099995613098',)
('Country: Togo|Global North/South: South|Score: 3.49499988555908',)
('Country: Guinea|Global North/South: South|Score: 3.50699996948242',)
('Country: Liberia|Global North/South: South|Score: 3.53299999237061',)
('Country: South Sudan|Global North/South: South|Score: 3.59100008010864',)
('Country: Yemen|Global North/South: South|Score: 3.59299993515015',)
```

We find that the 10 “least happy” countries all belong to the Global South!

In order to establish our claim, we perform a statistical hypothesis test with the following Hypothesis:

$$H0: \mu N\text{-}\mu S=0$$
$$Ha: \mu N=\mu S>0$$

After exporting our dataset, reading it in R and performing a t.test in R (refer to Appendix I for the full code), we obtain the following results:

```
Welch Two Sample t-test

data:  Happiness_Score by North_or_south
t = 6.5972, df = 125.8, p-value = 5.243e-10
alternative hypothesis: true difference in means is greater than 0
95 percent confidence interval:
 0.8061497       Inf
sample estimates:
mean in group North mean in group South
          6.041625            4.965071
```

Hence, we reject the null hypothesis at significance level $\alpha = 0.01$ (since our p-value= 5.243e-10 < α) and can conclude that we have strong evidence to suggest that the happiness of the Global North is greater than that of the Global South.

Conclusion and future work

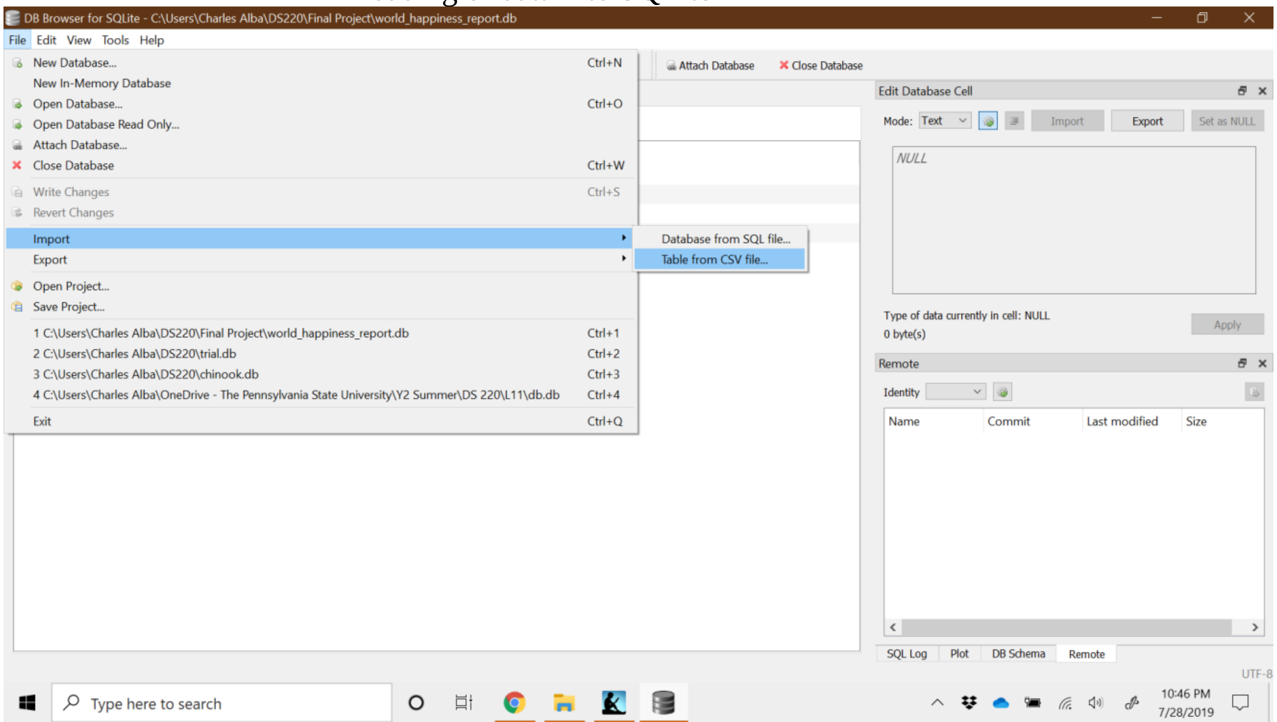
With the results above, the Global North-South divide is visible in relation to Happiness.
This is concluded based on our Statistical Hypothesis test performed. This is further backed by our queries which show that the average Happiness Score of the Global North significantly outweighs the Global South as well as the fact the 10 Happiest countries belong to the Global North while the 10 least happy countries all belong to the Global South.

Should any further future work be conducted, it hopes to focus mainly on discovering leading factors Happiness and its relation to the Global North-South divide. For instance, one would wish to determine if the difference in the North-South divide is purely due to the relatively higher GDP per capita of the Global North, especially since the Happiness index is the function of GDP per capita.

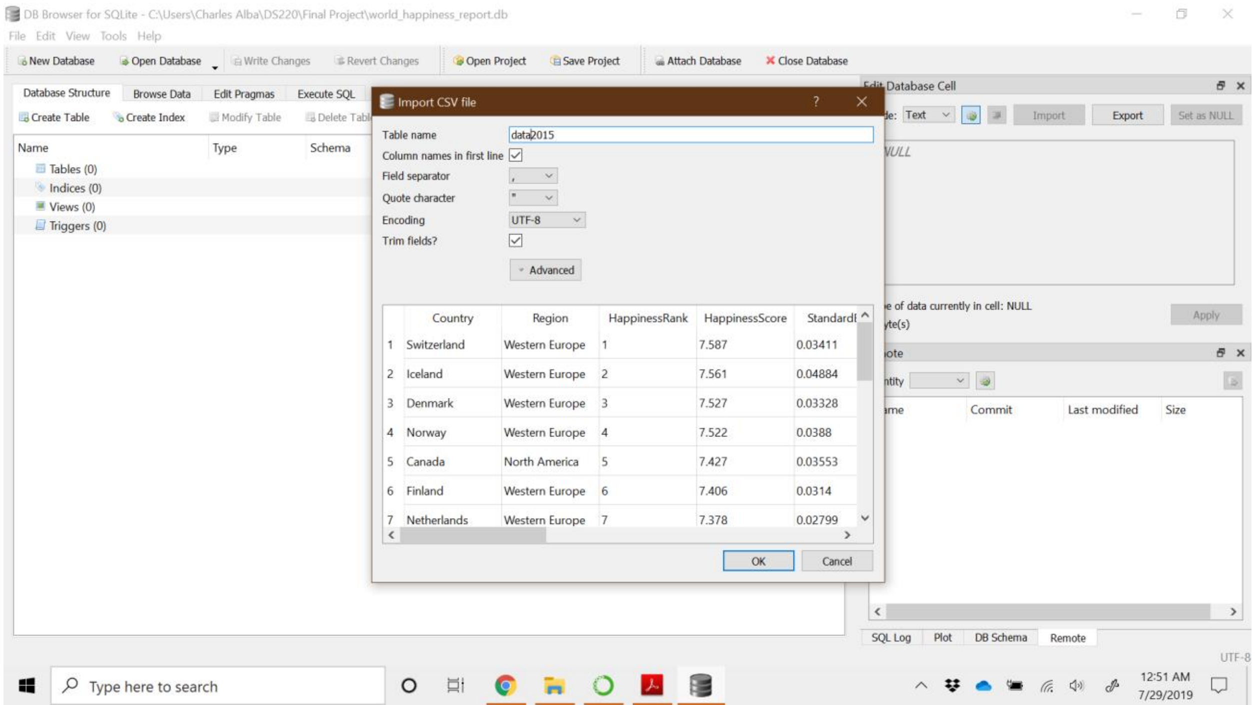
One could also wish to find if there is any bias within any forms of statistical computation conducted with the Happiness Index and how such biases could be controlled and accounted for.

Appendix

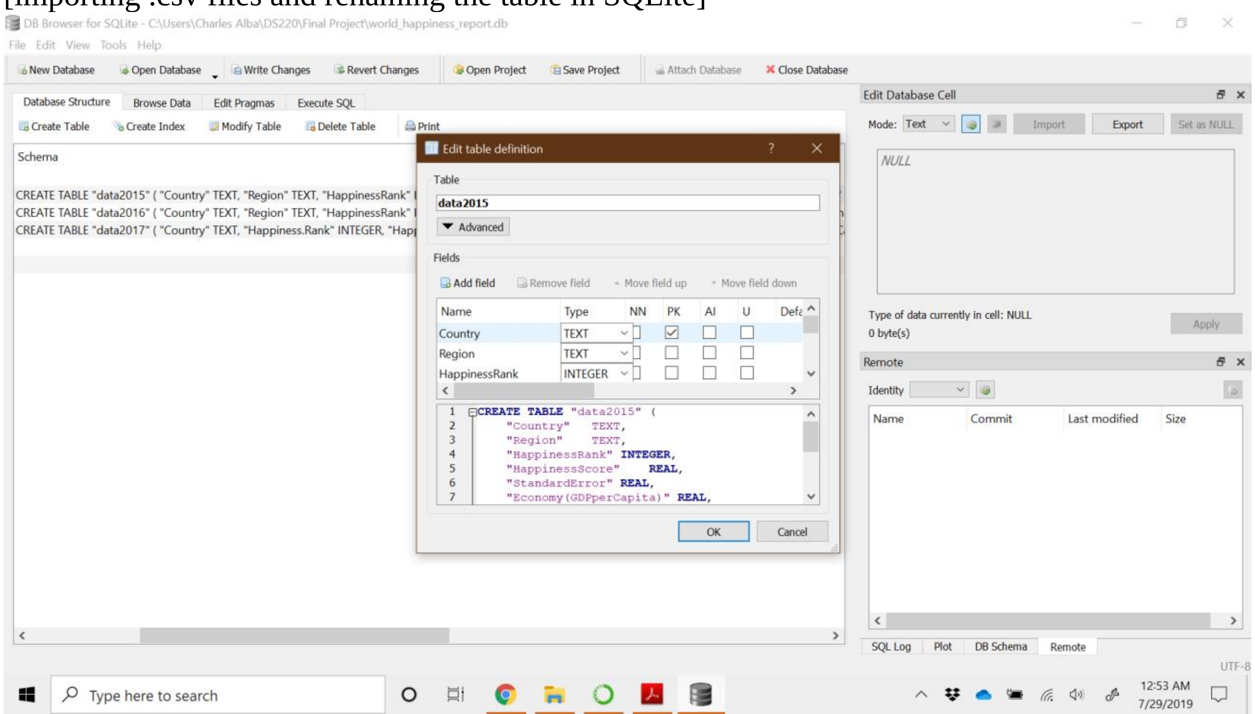
APPENDIX A.
Loading of data into SQLite



[Importing the .csv files into SQLite]



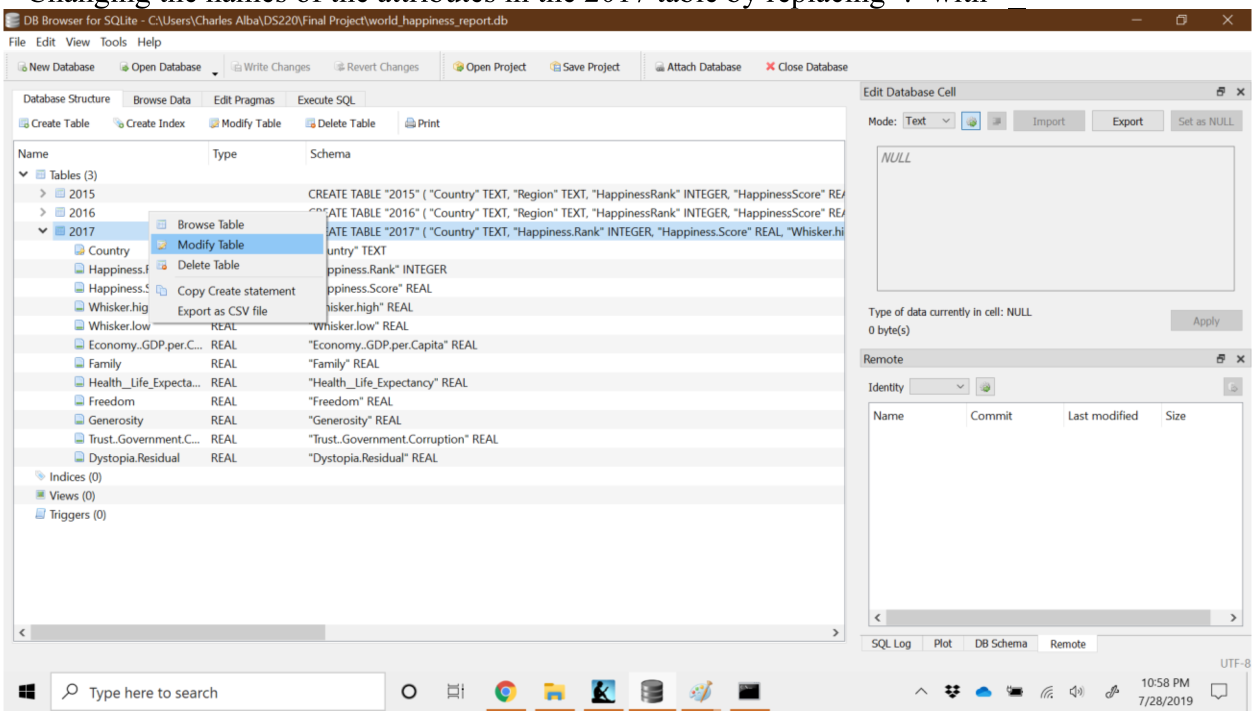
[importing .csv files and renaming the table in SQLite]

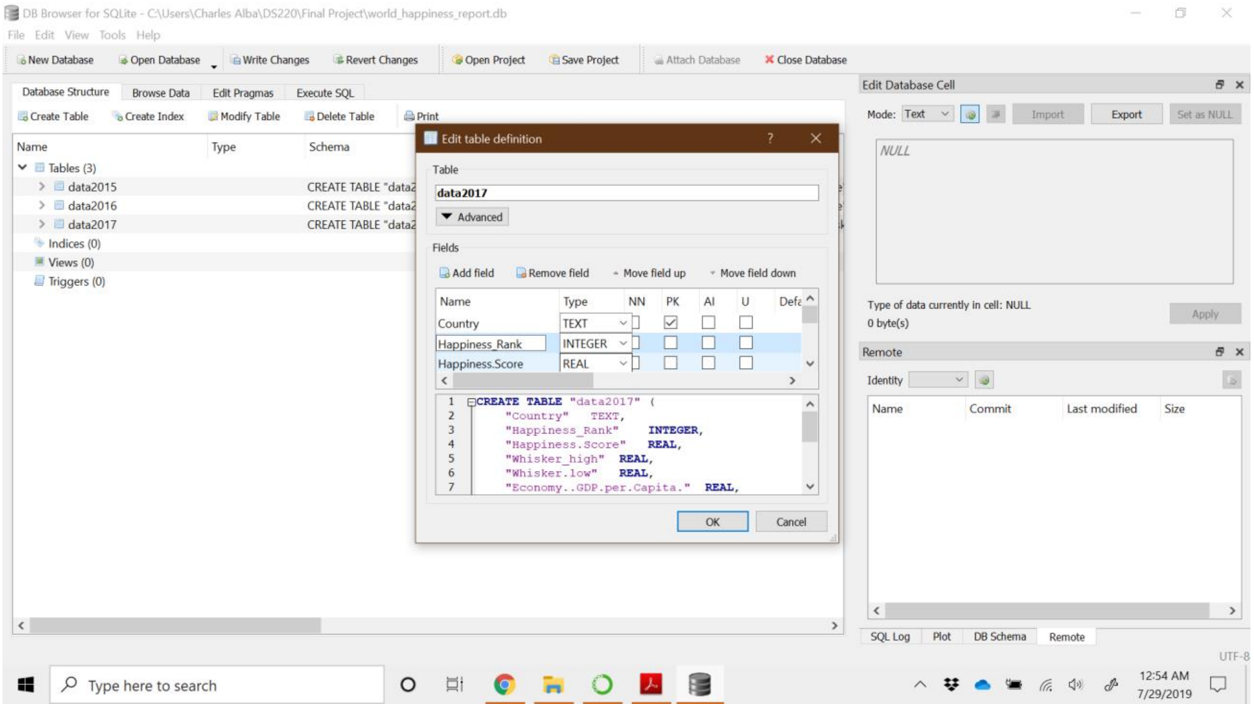


[Assigning country as the Primary Key]

APPENDIX B.

Changing the names of the attributes in the 2017 table by replacing ‘.’ with ‘_’





[Modifying the table to replace '.' with '_']

APPENDIX C.
Loading the db file into Jupyter Notebook

Code:

```python
#import libraries
import pandas as pd
import re
import sqlite3
```

```python
conn = sqlite3.connect('world_happiness_report.db')
c = conn.cursor()
```

APPENDIX D.

Running queries to determine countries that were present in one year but not present in another.

```
# To find countries present in 2015 table but absent in 2016.
query = '''SELECT A.Country FROM data2015 A WHERE Country NOT IN(SELECT Country FROM Data2016)'''
uncommon2015to2016 = pd.read_sql_query(query, conn)
uncommon2015to2016
```

	Country
0	Central African Republic
1	Djibouti
2	Lesotho
3	Mozambique
4	Oman
5	Somaliland region
6	Swaziland

```
# To find countries present in 2016 table but absent in 2015
query = '''SELECT A.Country FROM data2016 A WHERE Country NOT IN(SELECT Country FROM Data2015)'''
uncommon2016to2015 = pd.read_sql_query(query, conn)
uncommon2016to2015
```

	Country

0	Belize
1	Namibia
2	Puerto Rico
3	Somalia
4	Somaliland Region
5	South Sudan

```
# To find countries present in 2016 table but absent in 2017
query = '''SELECT A.Country FROM data2016 A WHERE Country NOT IN(SELECT Country FROM Data2017)'''
uncommon2016to2017 = pd.read_sql_query(query, conn)
uncommon2016to2017
```

	Country
0	Comoros
1	Hong Kong
2	Laos
3	Puerto Rico
4	Somaliland Region
5	Suriname
6	Taiwan

```
# To find countries present in 2017 table but absent in 2016
query = '''SELECT A.Country FROM data2017 A WHERE Country NOT IN(SELECT Country FROM Data2016)'''
uncommon2017to2016 = pd.read_sql_query(query, conn)
uncommon2017to2016
```

	Country
0	Central African Republic
1	Hong Kong S.A.R., China
2	Lesotho
3	Mozambique
4	Taiwan Province of China

APPENDIX E.
Running queries to find the naming of countries that were absent in subsequent years but present in the other years

```
# search for Somaliland in 2017 data
query = '''SELECT Country FROM data2017 WHERE country LIKE '%Somaliland%'
'''
Somaliland = pd.read_sql_query(query, conn)
Somaliland
```

	Country

```
# search for Hong Kong and Taiwan in 2015 data
query = '''SELECT Country FROM data2015 WHERE country LIKE '%Hong Kong%'
OR country LIKE '%Taiwan%' '''
Taiwan_and_HK = pd.read_sql_query(query, conn)
Taiwan_and_HK
```

	Country
0	Hong Kong
1	Taiwan

APPENDIX F.

Renaming of all the countries

```
# renaming Hong Kong in the 2017 table
c.execute('''UPDATE data2017
SET Country = "Hong Kong"
WHERE Country = "Hong Kong S.A.R., China"''')
```

```
<sqlite3.Cursor at 0x1d15d5268f0>
```

```
# renaming Taiwan in the 2017 table
c.execute('''UPDATE data2017
SET Country = "Taiwan"
WHERE Country = "Taiwan Province of China"''')
```

```
<sqlite3.Cursor at 0x1d15d5268f0>
```

```
# renaming Somaliland Region in the 2015 table
c.execute('''UPDATE data2015
SET Country = "Somaliland Region"
WHERE Country = "Somaliland region"''')
```

<sqlite3.Cursor at 0x1d15d5268f0>

```
# create North_or_south table
c.execute(''' ALTER TABLE data2017
ADD North_or_south varchar(255)''')
```

<sqlite3.Cursor at 0x140dffc7ab0>

APPENDIX G.
Insertion of North_or_south based off whether the country belongs to the Global North or Global South

```
# insert the columns for North based on Region
c.execute("""UPDATE data2017
        SET North_or_south = 'North'
        WHERE Region like '%Europe%' or Region = 'North America' or Region =
'Australia and New Zealand'
        """)
```

<sqlite3.Cursor at 0x140dffc7ab0>

```
# insert the columns for North based on Country
c.execute("""UPDATE data2017
        SET North_or_south = 'North'
        WHERE Country = 'Japan' or Country = 'South Korea'
        """)
```

<sqlite3.Cursor at 0x140dffc7ab0>

```
# insert the columns for Global South
c.execute("""UPDATE data2017
        SET North_or_south = 'South'
        WHERE North_or_south IS NULL
        """)
```

```
<sqlite3.Cursor at 0x140dffc7ab0>
```

APPENDIX H.
Code for performing Hypothesis Testing in R

```
> library(data.table)
> data<-fread("updated2017.csv")
> data<-data[,.(Country, Happiness_Score, North_or_south)]
> t.test(Happiness_Score ~ North_or_south, data = data, alternative = "greater")

	Welch Two Sample t-test

data:  Happiness_Score by North_or_south
t = 6.5972, df = 125.8, p-value = 5.243e-10
alternative hypothesis: true difference in means is greater than 0
95 percent confidence interval:
 0.8061497       Inf
sample estimates:
mean in group North mean in group South
          6.041625            4.965071
```

References

1. "World Happiness Index - Home." *Home | The*, 2018, worldhappiness.report/.
2. United Nations General Assembly. *Resolution Adopted by the General Assembly on 19 July 2011.* United Nations General Assembly, 2011, pp. 1–2, *Resolution Adopted by the General Assembly on 19 July 2011* .
3. "World Happiness Index - FAQ." *FAQ | The*, worldhappiness.report/faq/.
4. Easterlin, Richard A. "Happiness and Economic Growth – The Evidence." *Global Handbook of Quality of Life*, 2015, pp. 283–299., doi:10.1007/978-94-017-9178-6_12.
5. Garces, Eden J., et al. "Analysis on the Relationships on the Global Distribution of the World Happiness Index and Selected Economic Development Indicators." *OALib*, vol. 06, no. 06, 2019, pp. 1–13., doi:10.4236/oalib.1105455.
6. Helliwell, John, et al. "THE GEOGRAPHY OF WORLD HAPPINESS." 29 Apr. 2015.
7. Kabene, Stefane, et al. "Women in Political Positions and Countries' Level of Happiness." *Journal of International Women's Studies*, Aug. 2017.
8. Minkov, Michael, et al. "A New Worldwide Measure of Happiness Explains National Differences in Suicide Rates and Cigarette Consumption." *Cross-Cultural Research*, 2018, p. 106939711879968., doi:10.1177/1069397118799688.
9. Choi, Young-Chool, and Ji-Hyun Jang. "Relationships Among Social Policy Factors, National Competitiveness, and Happiness." *Applied Research in Quality of Life*, vol. 11, no. 4, 2015, pp. 1189–1205., doi:10.1007/s11482-015-9429-4.
10. Sawe, Benjamin Elisha. "What Is the North-South Divide?" *WorldAtlas*, 4 July 2017, www.worldatlas.com/articles/what-is-the-north-south-divide.html.
11. Sustainable Development Solutions Network. "World Happiness Report." *Kaggle*, 14 June 2017, www.kaggle.com/unsdsn/world-happiness#2015.csv.
12. Therien, Jean-Philippe. "Beyond the North-South Divide: The Two Tales of World Poverty." *Third World Quarterly*, vol. 20, no. 4, 1999, pp. 723–742., doi:10.1080/01436599913523.
13. Mimiko, Oluwafemi (2012). Globalization: The Politics of Global Economic Relations and International Business. Durham, N.C.: Carolina Academic. p. 47.
14. "North/South Divide." *LDOCE*, www.ldoceonline.com/dictionary/north-south-divide.