

# BABY BOOM OR BABY BUST



Submitted to

Dr. Bryan Hammer,

Oklahoma State University,

Stillwater, Oklahoma.

October 24, 2021

## PROJECT TEAM MEMBERS

Rupom Bhattacharjee	-	A20221013
Adwoa Boadi-Asamoah	-	A20198067
Chitra Boorla	-	A20349295
Srikanth Daruru	-	A20349204
Kodjo Botchway	-	A20338464

## TABLE OF CONTENTS

TABLE OF CONTENTS.....	2
EXECUTIVE SUMMARY .....	3
STATEMENT OF SCOPE.....	3
Project Goal:.....	4
Unit of Analysis:.....	4
PROJECT SCHEDULE.....	5
DATA PREPARATION .....	5
Data Access.....	5
Data Cleaning .....	7
Data Reduction.....	7
Data Dictionary.....	8
Variable Visualization .....	10
Data Transformation .....	19
ANALYSIS .....	21
MULTIPLE REGRESSION.....	21
TEXT MINING AND SENTIMENTAL ANALYSIS .....	23
RESULTS FROM NAMED ENTITY RECOGNITION .....	29
CONCLUSION .....	29
APPENDIX.....	30
LIST OF FIGURES .....	30
CODES .....	31

## **EXECUTIVE SUMMARY**

The changes in the general trends of population growth have been consequential regarding various aspects of life. There exist changes across the world in terms of booms or busts in population figures. Be it a surge or a plunge; these changes tend to leave a significant footprint in their wake, which would require targeted mitigations for effective re-stabilization. The Covid-19 pandemic did no less to contribute to this issue. This project seeks to discuss the changes in the general population figures based on a set of accumulated data. The population changes, specifically, the increase in the number of newborns over a period, which in turn, depending on existing capabilities, will affect the stability of any country as long-term economic growth depends on three factors, i.e., population, participation, and productivity. The potential problem identified is the absence of adequate preparations based on whether there is a baby boom (surge), or baby bust (plunge) in each country. In this project, we will try to highlight the above-related trends of population shifts or trends. This study will further benefit the families of newborns and the policymakers to understand the current scenario and prepare for the upcoming challenges.

## **STATEMENT OF SCOPE**

In this project, we aim to determine how covid has impacted population growth in countries worldwide. COVID-19 has had its fair share of impact on various sectors of the world, including but not limited to economic and financial stability, health, and other related development. We are going to understand the population effect on some of these sectors. For example, suppose there is a baby boom in an under-populated country. In that case, it will help in the proper application of underutilized resources and increased economic status since new ventures will be created because of the increase in the availability of labor. On the other hand, in a highly populated country, the baby boom will result in strains in various systems, so they need to have a proper plan to effectively use resources and labor.

## **Project Goal:**

- To identify the changes in the different countries from the year before and after covid.
- What affected the birthrate during the pandemic? Can we attribute it to solely covid-19, or is there any other contributing factor?

## **Project Objectives:**

- To determine if there was a rise or fall in the birth rate for countries around the world.
- To identify factors that affected the surge or plunge in birthrate across the globe
- To perform a multiple regression, sentiment analysis, and named entity recognition to conclude.

## **Unit of Analysis:**

The unit of analysis for our project would be the fertility rate. The fertility rate will help us understand how covid has contributed to the varying rate of pregnancies and its consequences.

**Variables:** The following are the variables we are taking into consideration for our project:

1. Fertility Rate (Live Births / Woman)
2. Gross Domestic Product
3. Reproductive choices for women
4. Cost of living
5. Quality of life
6. COVID-19 data (e.g., total cases, deaths, etc.)
7. Tweets on the public reaction on the effect of covid 19 in birthrate

The project was set to be worked on and completed over 16 weeks. The overview of the project schedule is provided in the GANTT chart shown below, with all the dates and timelines concerned with the completion of the project along with roles, assignments, and duration. For a better view, please refer to the [online](#) version of the source file.



## Data Access

5

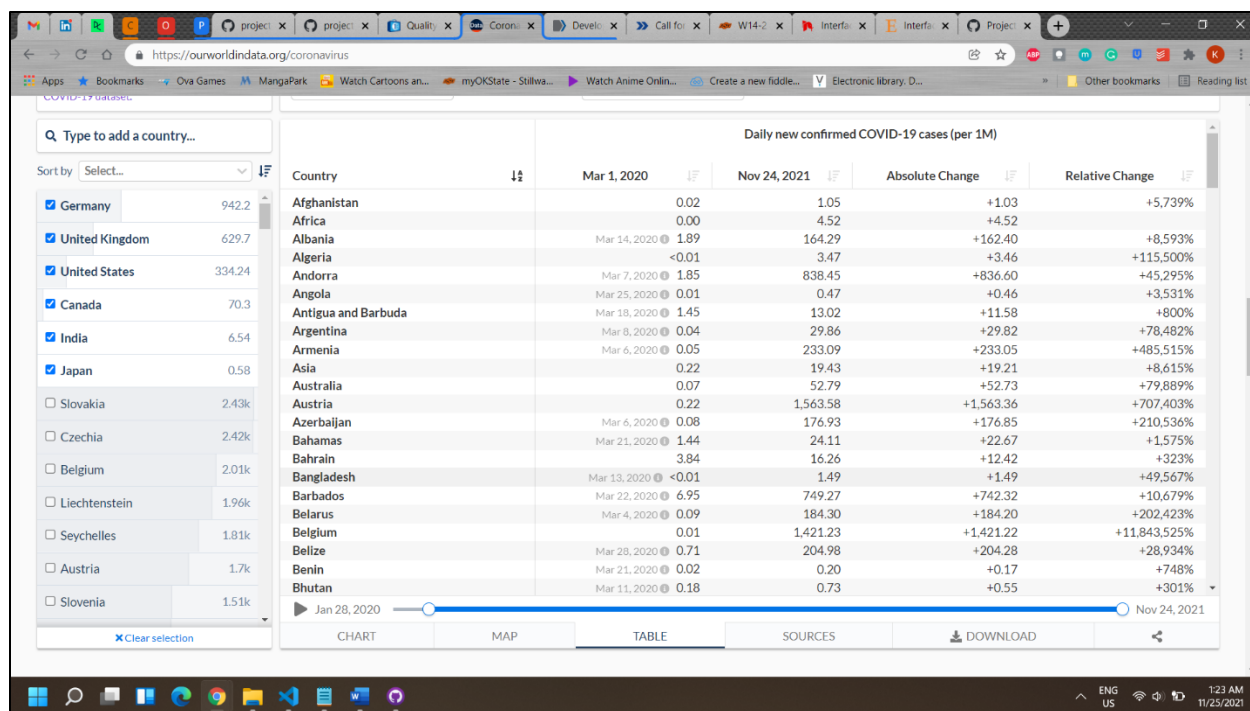


Figure 2: Web Source

The next step involved was to obtain the data. A significant volume of the data that we used we primarily did not have to extract from the stated data source through web-scraping. On the other hand, the supporting websites did some scrapping with the 'rvest' function to extract the tables and convert them to useable data frames. Consequently, the data still had to be worked on since there were missing and irrelevant data that needed to be cleaned out, which would be further explained in the report. The CSV files for different countries under various variables were downloaded and loaded into a data frame for further analysis. The scraped data was also merged with existing data by the country and then the year to retain the consistency among the different CSV files. The data consisted of the factors contributing to the increase or the decrease in the number of babies born within a particular period. The consequences were described by another data set that investigated the general ramifications for the selected period. The full codes for the scraping and cleaning of the data can be seen in the **appendix**.

## Data Cleaning

The number of countries present in the dataset that we obtained encompassed all the available countries around the world for which the information from the websites could account for. This included aggregated data for regions and sub-regions classified on a continental basis. We had to remove this data and filter out other non-country information. For example, in our *cleaned family planning data for married women.csv* dataset, we had information on the world, region, and subregion aggregations. We had to clean that dataset by removing all the observations for the index subregion. We also had to remove some of the columns we are not using in our work, such as *country type*, *timeframe*, and *FIPS*. Also, in merging some of our data sources, we realized that some of the tabular information contained countries that did not have data regarding other columns. To clarify, we had almost every detail for the countries and their Covid-19 breakdowns but not all these countries had data for the reproductive choices for women. The plan was to represent these values by NA values, but we removed these countries from the data frames on further discussion. Since the analysis will be conducted as a guideline to the effects of the population changes, the conclusions would still be valid and applicable to every country depending on the variables taken into consideration. We also had the Covid-19 data for each day, so we created a column for each country's cumulative numbers recorded by the month. We wrote a regular expression for obtaining the final cumulative count for the entire year.

## Data Reduction

Primarily, for data reduction, the plan is to consider some selected countries from different sections of the world and obtain some correlations on the data we have obtained and use those correlations for analysis with all the other data. We have smaller data sets provided from countries where a population difference would be glaring, including the USA, China, India, and Nigeria.

## Data Consolidation

The primary datasets we obtained were finally put into one combined file named `baby.csv`. In that light, the other databases used were given similar names describing the information they contained, `Baby_***.csv`. The codes were written in Python named `Baby_Script.py` and R and named `Data-Combined.R`, which comprised the file handling, the cleaning, extraction, and reduction write-ups. The section of code used for all of this would be entered in the appendix after significant further work, minimal edits, and parsing have been done.

## Data Dictionary

We have two significant datasets that we used for the report, and here are the data dictionaries of the datasets.

### *Baby\_Covid.csv*

Attribute Name	Description	Data Type	Source
Location	Name of the country	char(30)	<a href="https://ourworldindata.org/coronavirus">https://ourworldindata.org/coronavirus</a>
Year	Year for the following data	integer	<a href="https://ourworldindata.org/coronavirus">https://ourworldindata.org/coronavirus</a>
Iso_Code	Country short code	char(30)	<a href="https://ourworldindata.org/coronavirus">https://ourworldindata.org/coronavirus</a>
Population	Population of the country	integer	<a href="https://ourworldindata.org/coronavirus">https://ourworldindata.org/coronavirus</a>
GDP	GDP of the country	float	<a href="https://ourworldindata.org/coronavirus">https://ourworldindata.org/coronavirus</a>
Cases	The number of Covid cases recorded	integer	<a href="https://ourworldindata.org/coronavirus">https://ourworldindata.org/coronavirus</a>
Deaths	The number of Covid deaths recorded	integer	<a href="https://ourworldindata.org/coronavirus">https://ourworldindata.org/coronavirus</a>
Cases_per_Million	The number of Covid cases recorded per 1 million population	float	<a href="https://ourworldindata.org/coronavirus">https://ourworldindata.org/coronavirus</a>
Deaths_per_Million	The number of Covid deaths recorded per 1 million population	float	<a href="https://ourworldindata.org/coronavirus">https://ourworldindata.org/coronavirus</a>



Hosp_Patients_per_Million	Hospital patients recorded per million population	float	<a href="https://ourworldindata.org/coronavirus">https://ourworldindata.org/coronavirus</a>
Hospital_beds_per_thousand	Hospital beds per million population	float	<a href="https://ourworldindata.org/coronavirus">https://ourworldindata.org/coronavirus</a>
Icu_Patients_per_Million	ICU Patients per million population	float	<a href="https://ourworldindata.org/coronavirus">https://ourworldindata.org/coronavirus</a>

#### Baby.csv

Attribute Name	Description	Data Type	Source
Country	Name of the country	char(30)	<a href="https://ourworldindata.org/coronavirus">https://ourworldindata.org/coronavirus</a>
Year	Year for the following data	integer	<a href="https://ourworldindata.org/coronavirus">https://ourworldindata.org/coronavirus</a>
Code	Country short code	char(30)	<a href="https://www.numbeo.com/quality-of-life/rankings_by_country.jsp?title=2020">https://www.numbeo.com/quality-of-life/rankings_by_country.jsp?title=2020</a>
Cost.of.Living.Index	Indexes for the cost of living for the countries	float	<a href="https://www.numbeo.com/cost-of-living/rankings_by_country.jsp?title=2020&amp;displayColumn=-1">https://www.numbeo.com/cost-of-living/rankings_by_country.jsp?title=2020&amp;displayColumn=-1</a>
Rent.Index	Indexes for the rent for the countries	float	<a href="https://www.numbeo.com/cost-of-living/rankings_by_country.jsp?title=2020&amp;displayColumn=-1">https://www.numbeo.com/cost-of-living/rankings_by_country.jsp?title=2020&amp;displayColumn=-1</a>
Health.Care.Index.x	Indexes for Health Care for the countries	float	<a href="https://www.numbeo.com/health-care/rankings_by_country.jsp?title=2020">https://www.numbeo.com/health-care/rankings_by_country.jsp?title=2020</a>
Quality.of.Life.Index	Indexes for Quality of Life for the countries	float	<a href="https://www.numbeo.com/quality-of-life/rankings_by_country.jsp?title=2020">https://www.numbeo.com/quality-of-life/rankings_by_country.jsp?title=2020</a>
Annual Births per Country	The annual number of births by the country for that given year	integer	<a href="https://ourworldindata.org">https://ourworldindata.org</a>
Family Planning	Family Planning indexes for countries	float	<a href="https://ourworldindata.org">https://ourworldindata.org</a>
Live Births per Woman per Country	Live births per woman per country	float	<a href="https://ourworldindata.org">https://ourworldindata.org</a>

## Variable Visualization

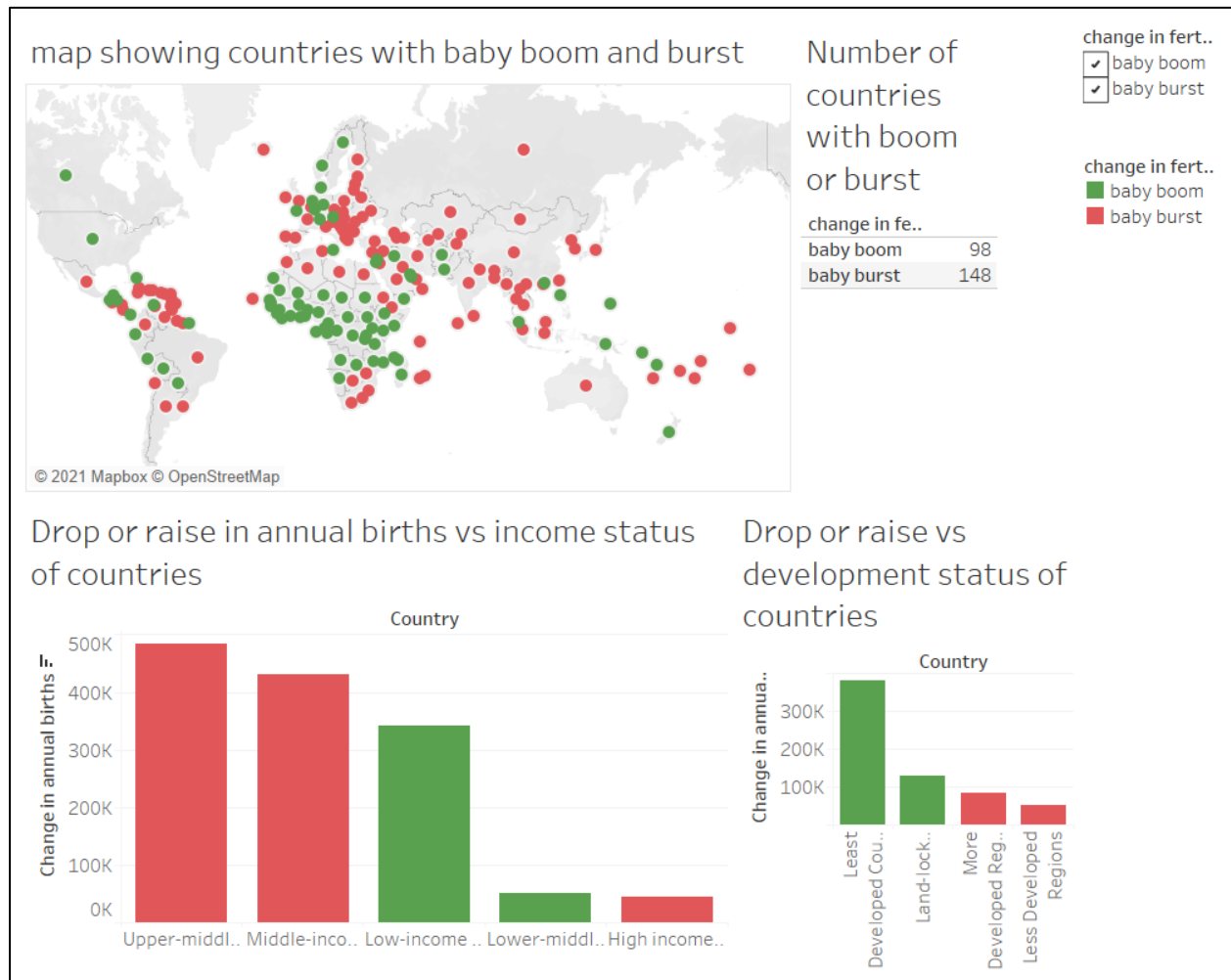


Figure 3: Dashboard showing how birthrate from 2019 and 2020 are different for different countries and continents.

The columns or circles in green represent an increase in birthrate from 2019 to 2020, representing a baby boom, whereas the red colors show a drop in birth rates. There were more countries with baby busts than boom, as can be seen on the geographical plot. The bar chart on the left shows how the income status of a country affects the birthrate. The upper-middle-, middle- and high-income countries faced baby burst, and lower-income nations had a baby boom. The right bar chart shows how these numbers change with the development status of the countries. The least developed countries had a baby boom, but the more developed nations faced a baby burst.

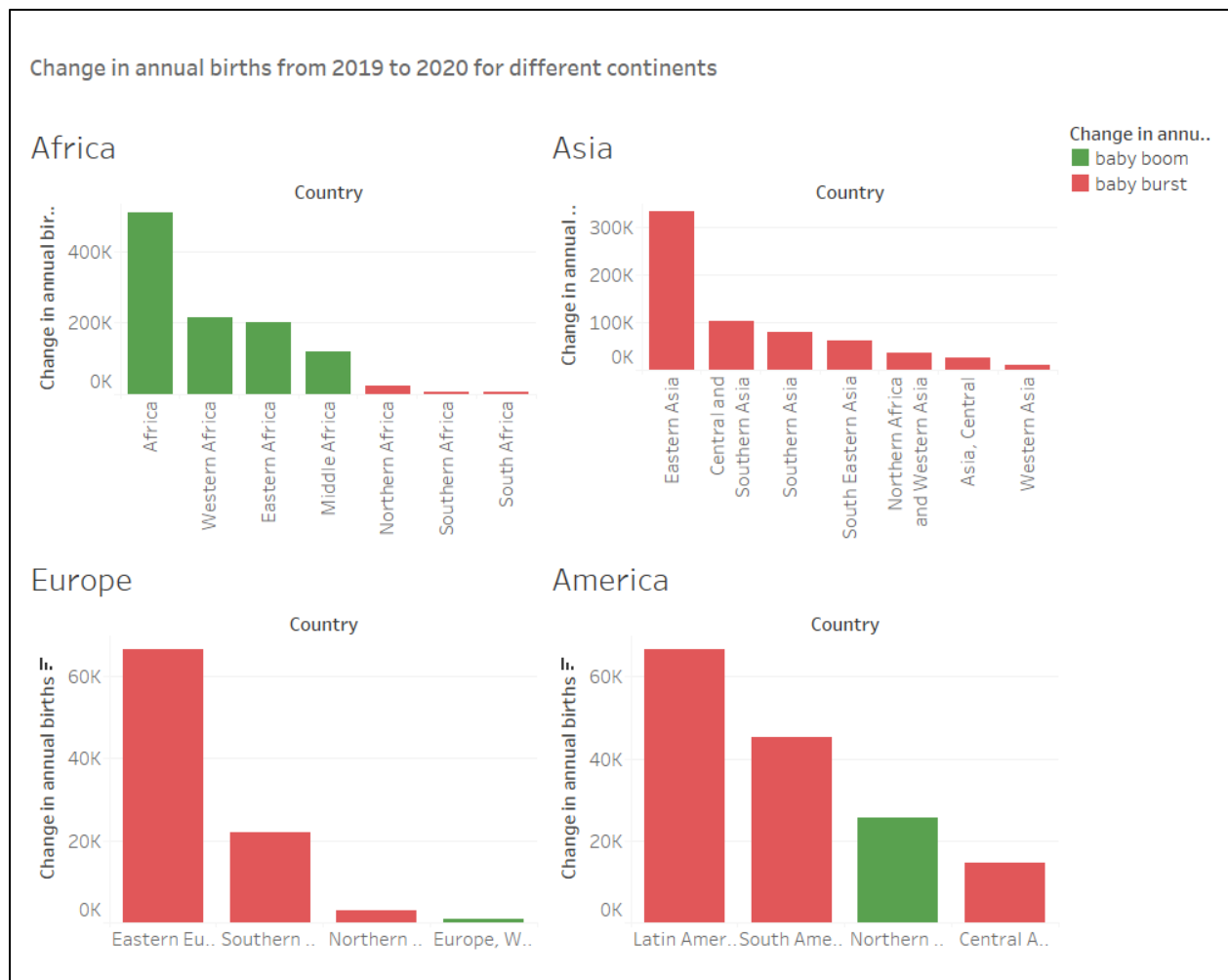


Figure 4: Dashboard showing how birthrate from 2019 and 2020 are different for different countries from different continents. Countries in Asia and Europe all faced baby busts; Africa was one of the continents with more boom than busts.

The next step we took was to explore the continuous variables of the datasets. To effectively utilize the variables, we need to understand the relationships and interdependencies with the other values, including the target variable. We identified the distribution of the values amongst all the given variables. We obtained the mean and median values to generate comparable statistics across all the data we collected for the respective countries. We completed this procedure using JMP software. The breakdown of the data was given as:

Cost.of.Living.Index.x

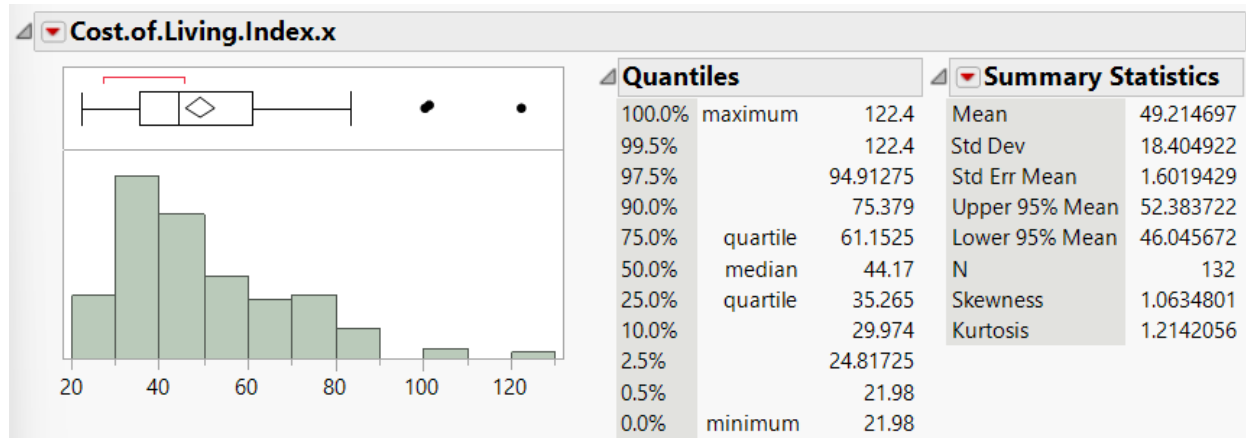


Figure 55: Distribution of the variable Cost of Living Index

The figure above shows the frequency, skewness, and summary statistics of the variable—the distribution of the Rent. The index variable is right-skewed with a mean of 49.215, ignoring the impact of the missing values on the summary.

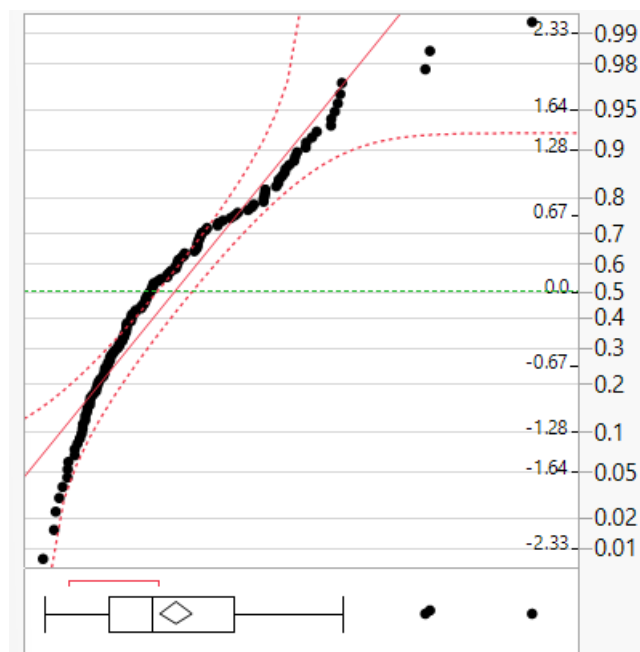


Figure 66: Normality plot for the Cost-of-Living Index

This figure above illustrates the normality distribution. This reveals the non-normality of the variable.

Rent.Index

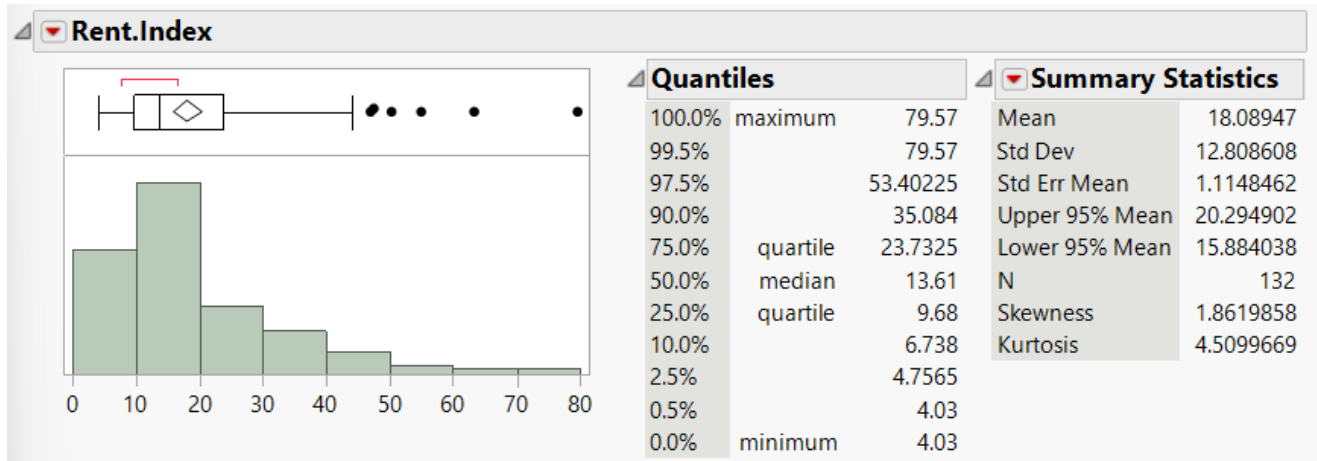


Figure 77: Distribution of the variable Rent Index

The figure above shows the frequency, skewness, and summary statistics of the variable—the distribution of the Rent. The index variable is right skewed with a mean of 18.089, ignoring the impact of the missing values on the summary.

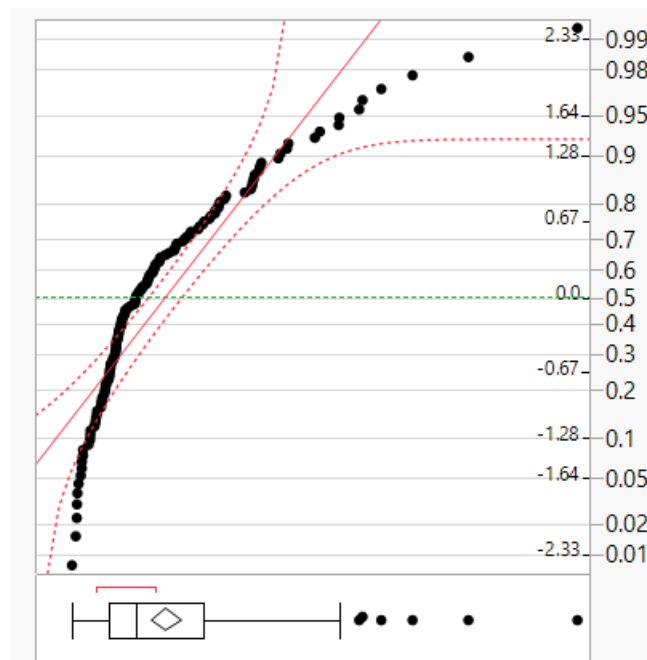


Figure 88: Normality plot for Rent Index

This figure above illustrates the normality distribution. This reveals the non-normality of the variable.

Health.Care.Index.x

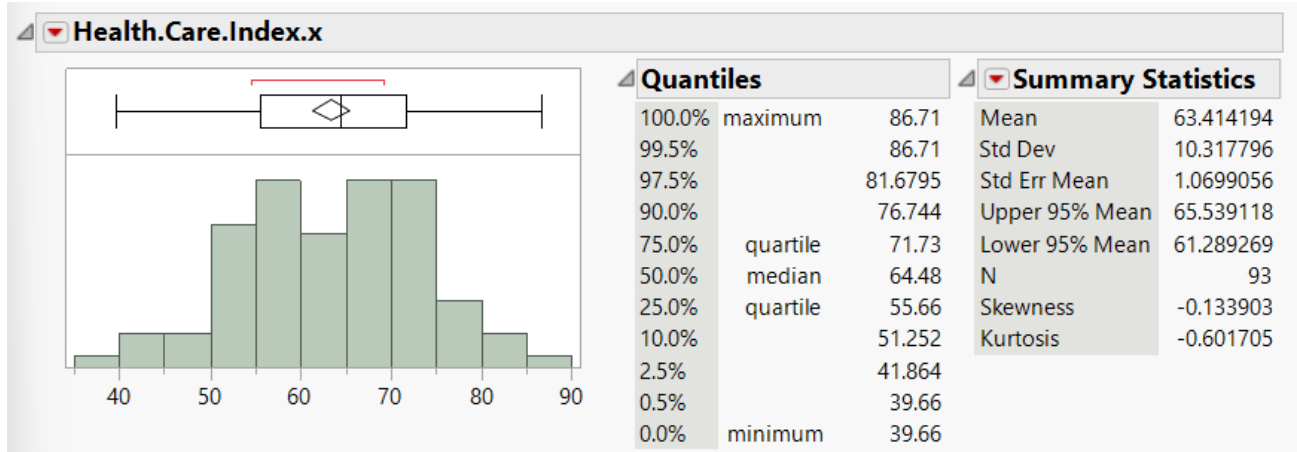


Figure 99: Distribution of the variable Health Care Index

The figure above shows the frequency, skewness, and summary statistics of the variable—the distribution of the *Health.Care.Index* variable is left-skewed with a mean of 63.414, ignoring the impact of the missing values on the summary.

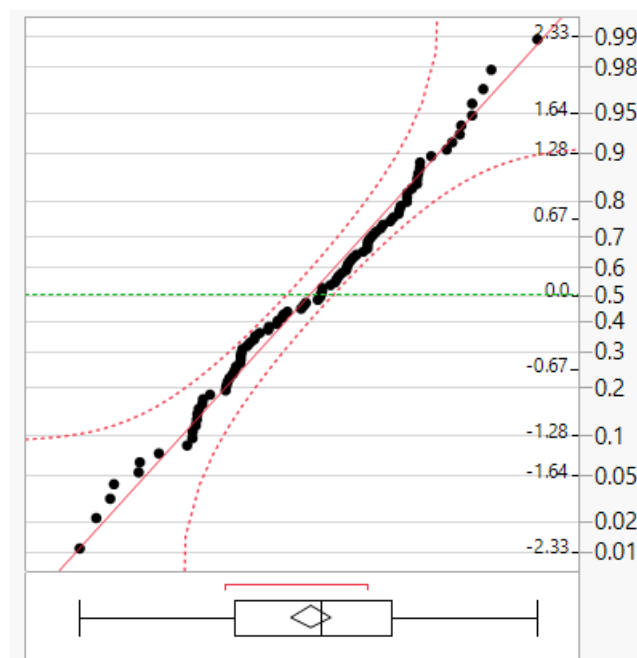


Figure 1010: Normality plot for Health Care Index

This figure above illustrates the normality distribution. This reveals that the variable is standard.

Health.CareExp.Index

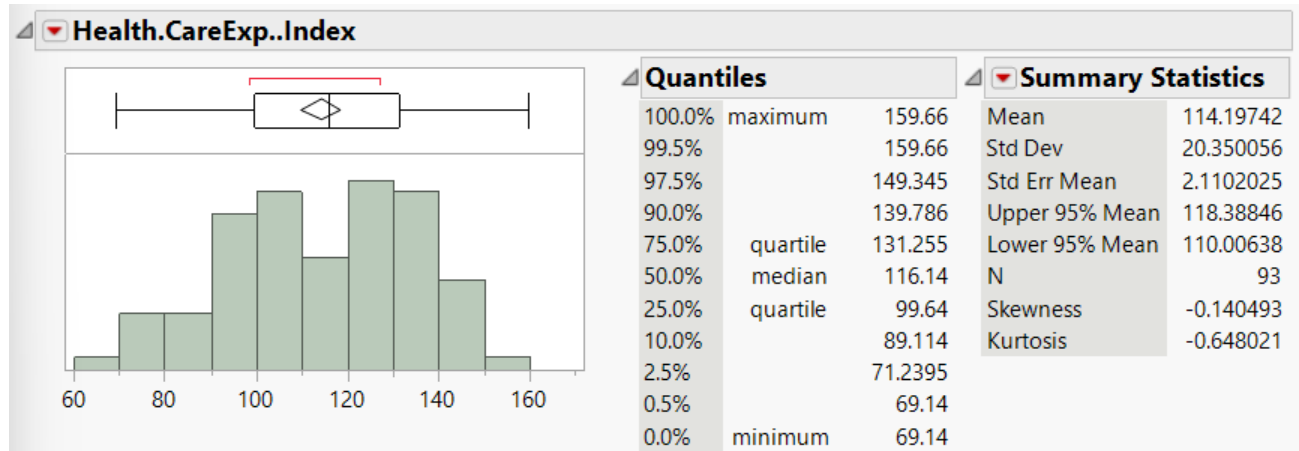


Figure 1111: Distribution of the variable Health Care Expertise

The figure above shows the frequency, skewness, and summary statistics of the variable—the distribution of *health.CareExp.index* variable is left-skewed with a mean of 114.197, ignoring the impact of the missing values on the summary.

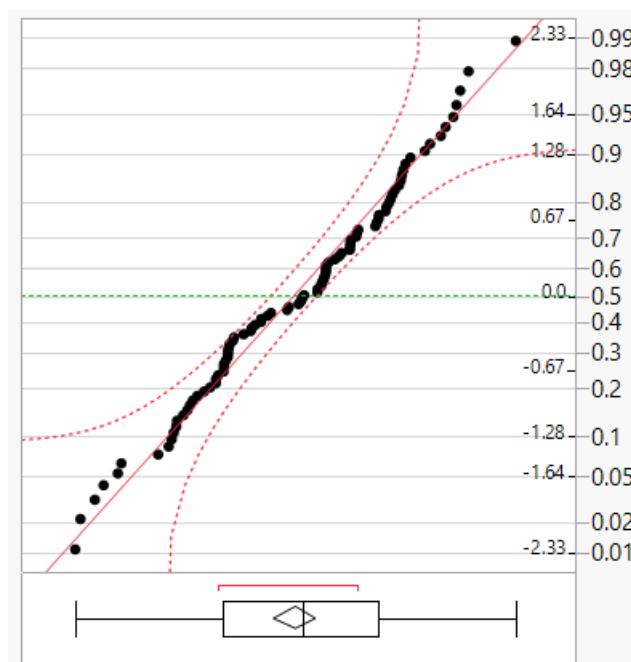


Figure 1212: Normality plot for Health Care Expertise

This figure above illustrates the normality distribution. This reveals that the variable is normal.

### Quality.of.Life.Index

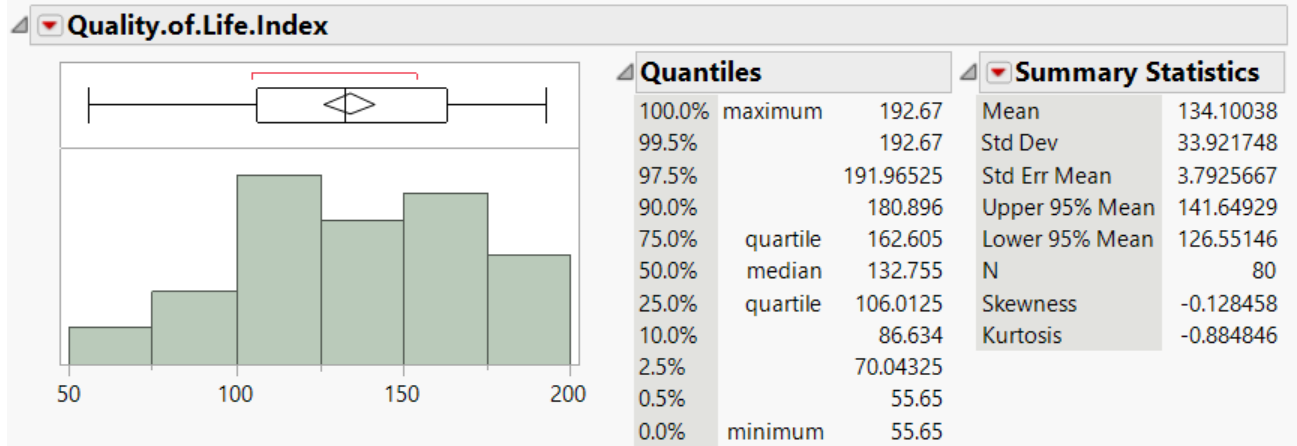


Figure 1313: Distribution of the variable Quality of Life Index

The figure above shows the frequency, skewness, and summary statistics of the variable. The distribution of the *Quality.of.Life.index* variable is left-skewed with a mean of 134.100, ignoring the impact of the missing values on the summary.

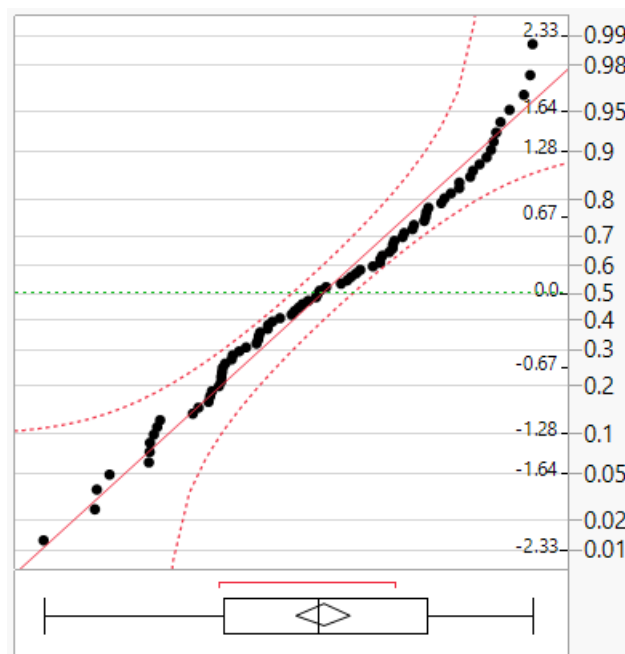


Figure 1414: Normality plot for Quality-of-Life Index

This figure above illustrates the normality distribution. This reveals that the variable is normal.

## Family Planning



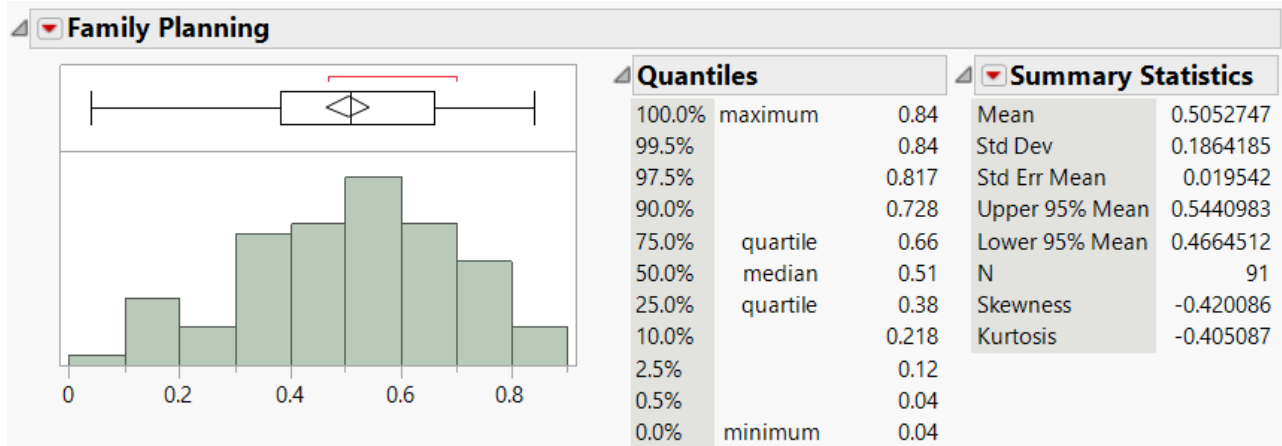


Figure 1515: Distribution of the variable Family Planning

The figure above shows the frequency, skewness, and summary statistics of the variable. The distribution of the Family Planning variable is left-skewed with a mean of 0.505, ignoring the impact of the missing values on the summary.

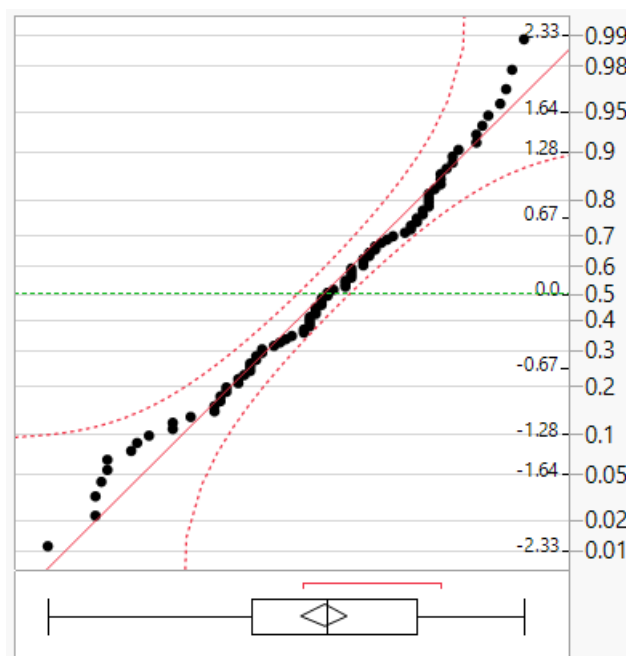


Figure 1616: Normality plot for Family Planning

This figure above illustrates the normality distribution. This reveals that the variable is normal.

### Live Births per Woman per Country

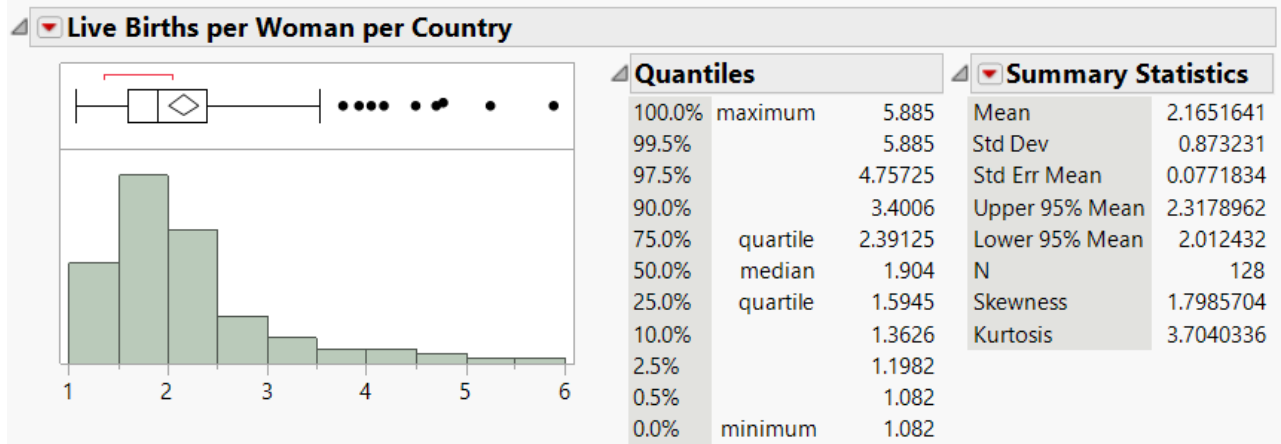


Figure 1717: Distribution of the variable Live Births per Woman per Country

The figure above shows the frequency, skewness, and summary statistics of the variable. The distribution of the Live Births per Woman per Country variable is right-skewed with a mean of 2.165, ignoring the impact of the missing values on the summary.

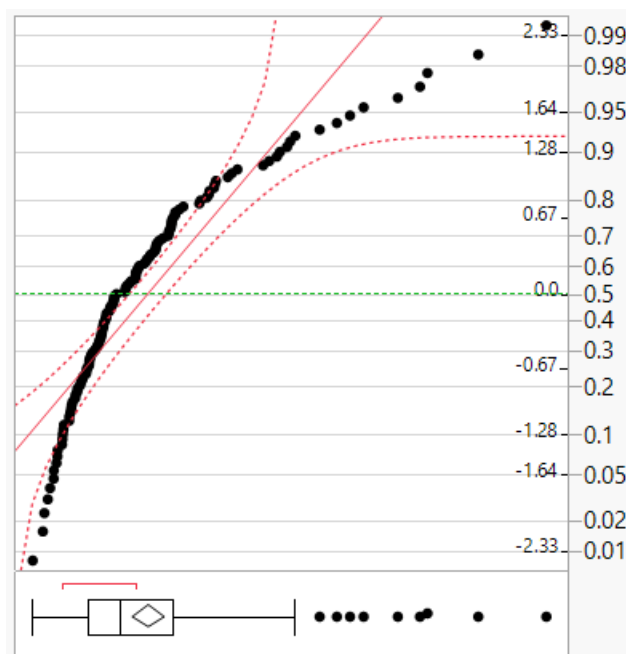


Figure 1818: Normality plot for Live Births per Woman per Country

This figure above illustrates the normality distribution. This reveals the non-normality of the variable.

### Annual Births per Country (Target Variable)

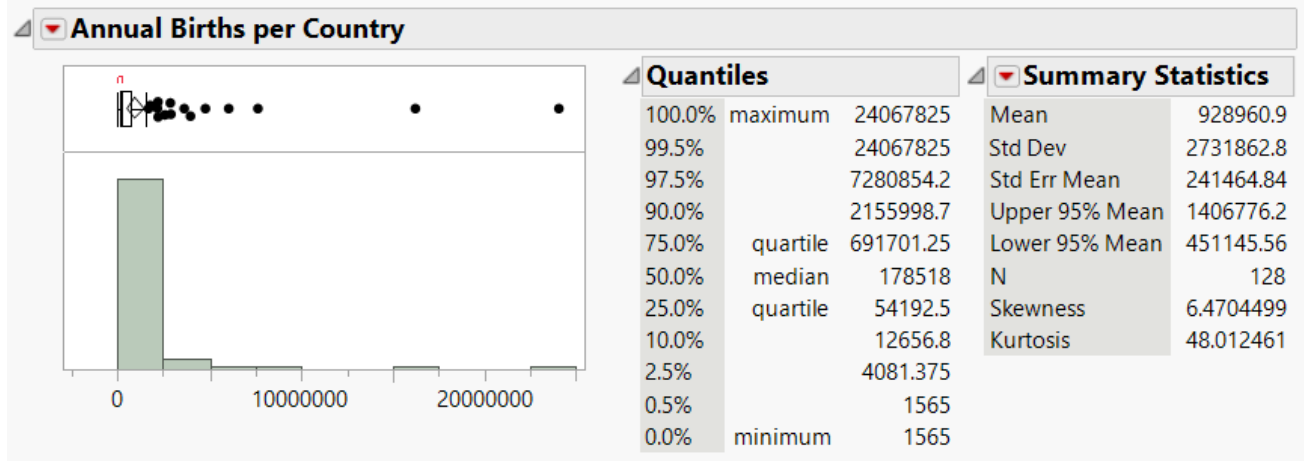


Figure 1919: Distribution of the variable Annual Births per Country

The figure above shows the frequency, skewness, and summary statistics of the variable. The distribution of the Annual Births per Country variable is right-skewed with a mean of 928960.9, ignoring the impact of the missing values on the summary.

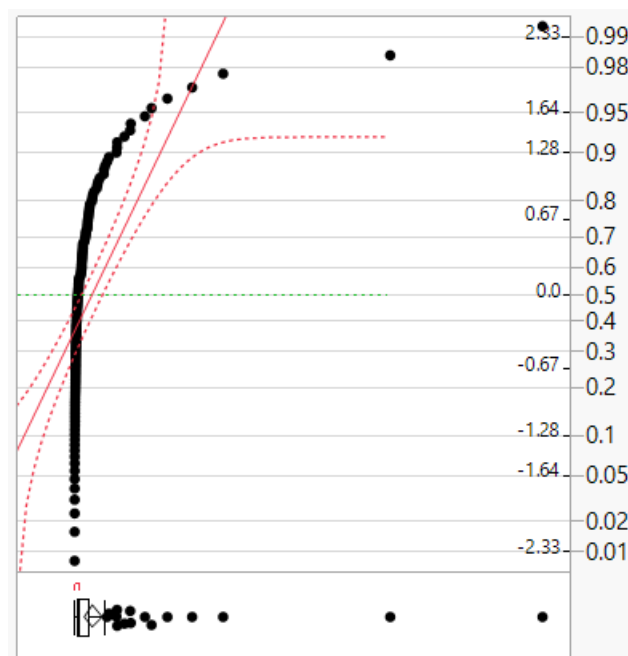


Figure 2020: Normality plot for Annual Births per Country

This figure above illustrates the normality distribution. This reveals the non-normality of the variable.

## Data Transformation

We plan to transform the variables that strongly deviated from the normality assumption. The reason behind the need to transform the variables and make sure the variables conform to these assumptions is to perform a regression analysis to determine the effects of the variables on the annual number of births in that given year. The assumptions of linear regression in this study will be based on the normality of error, constancy of variance, linearity between predictors and variables. Since 2020 was massively impacted by Covid as a known phenomenon, it was not exactly intuitive to compare the year 2020 to the other year in terms of the selected variables in that time and hence the removal of the year variable in the analysis in this section. These were identified as the Cost of Living, Rent Index, Live Births per Woman per Country, and Annual Births per Country. The tendency of the continuous variables to be linearly related to the target variable. A log transformation would be applied to make these conform to the made assumptions. We then run a correlation matrix to identify which of the variables has strong correlations with each other.

Correlations								
	Cost of Living Index	Rent Index	Health Care Index	Health Care Exp. Index	Quality of Life Index	Family Planning	Live Births per Woman per Country	Annual Births per Country
Cost of Living Index	1.0000	0.8069	0.5865	0.6115	0.6946	0.3961	-0.3883	-0.2247
Rent Index	0.8069	1.0000	0.4392	0.4632	0.4968	0.3224	-0.3179	-0.1318
Health Care Index	0.5865	0.4392	1.0000	0.9977	0.6109	0.4244	-0.4185	-0.0274
Health Care Exp. Index	0.6115	0.4632	0.9977	1.0000	0.6345	0.4274	-0.4374	-0.0376
Quality of Life Index	0.6946	0.4968	0.6109	0.6345	1.0000	0.2648	-0.5913	-0.2174
Family Planning	0.3961	0.3224	0.4244	0.4274	0.2648	1.0000	-0.3492	-0.0013
Live Births per Woman per Country	-0.3883	-0.3179	-0.4185	-0.4374	-0.5913	-0.3492	1.0000	0.1606
Annual Births per Country	-0.2247	-0.1318	-0.0274	-0.0376	-0.2174	-0.0013	0.1606	1.0000

Figure 2121: Correlation Matrix of Variables

From **Figure 21**, we identified the strong correlations between each and then went further to use VIF values to take out the variables with high possible affiliations that will distort the model's predictions. Rent Index and Cost of Living showed a very high correlation factor, and between the Health Care Exp Index and the Health Care Index, we were only going to select just one of them.

## ANALYSIS

### MULTIPLE REGRESSION

The building of multiple regression prediction equations was to help identify the included variables and their order of importance in the said equation. To further elaborate on the ineffectiveness of incorporating both variables with high correlation, the first model was created with all of them present.

Effect Summary			
Source	LogWorth		PValue
Family Planning	1.842		0.01440
Log(Live Births per Woman per Country)	1.523		0.03001
Quality.of.Life.Index	1.369		0.04272
Health.Care.Index.x	0.228		0.59174
Log(Rent.Index)	0.166		0.68194
Health.CareExp..Index	0.162		0.68816
Log(Cost.of.Living.Index.x)	0.043		0.90616

Figure 2222: Multiple Regression with all factors

Summary of Fit	
RSquare	0.424382
RSquare Adj	0.336788
Root Mean Square Error	1.399208
Mean of Response	13.01286
Observations (or Sum Wgts)	54

Figure 2323: Summary of fit for Model 1

Parameter Estimates				
Term	Estimate	Std Error	t Ratio	Prob> t
Intercept	10.314994	4.73865	2.18	0.0347*
Log(Cost.of.Living.Index.x)	-0.177083	1.494004	-0.12	0.9062
Log(Rent.Index)	-0.308069	0.746967	-0.41	0.6819
Health.Care.Index.x	0.1935363	0.358344	0.54	0.5917
Health.CareExp..Index	-0.075588	0.187144	-0.40	0.6882
Quality.of.Life.Index	-0.020035	0.009613	-2.08	0.0427*
Family Planning	3.0387443	1.194711	2.54	0.0144*
Log(Live Births per Woman per Country)	2.1093421	0.941951	2.24	0.0300*

Figure 2424: Parameter Estimates of Model 1

The figures above (**Figure 20, 21, and 22**) illustrate the significance and order of importance of the respective variables used in the equation. From **Figure 21**, the difference between the RSquare and adjusted RSquare value indicates the presence of variables in the model that are not necessary to predict the effect on the annual births. The significance values (Prob>|t|) show the relevance of the variables to the primary target variable. In this model, the only significant variables would be Family Planning, Live Births per Woman, and Quality of Life in this case. However, from the previous correlation analysis, we identified some of the parameters affiliated with other parameters. Taking those out and running another regression model, we came out with Figures **23, 24, and 25**.


Effect Summary			
Source	LogWorth		PValue
Family Planning	1.923		0.01195
Log(Live Births per Woman per Country)	1.671		0.02135
Quality.of.Life.Index	1.473		0.03365
Health.Care.Index.x	1.246		0.05675
Log(Cost.of.Living.Index.x)	0.476		0.33447

Figure 2525: Regression with the selected factors

Summary of Fit	
RSquare	0.420102
RSquare Adj	0.359696
Root Mean Square Error	1.37483
Mean of Response	13.01286
Observations (or Sum Wgts)	54

Figure 2626: Summary of fit for Model 2

Parameter Estimates				
Term	Estimate	Std Error	t Ratio	Prob> t
Log(Cost.of.Living.Index.x)	-0.830858	0.852198	-0.97	0.3345
Health.Care.Index.x	0.0496502	0.025432	1.95	0.0568
Quality.of.Life.Index	-0.020564	0.009403	-2.19	0.0337*
Log(Live Births per Woman per Country)	2.0878898	0.877355	2.38	0.0213*
Family Planning	3.0454082	1.165474	2.61	0.0119*
Intercept	12.568633	2.689607	4.67	<.0001*

Figure 2727: Parameter estimates for Model 2

The analytical breakdown of the model with the new parameters' states that the vital model variable that contributed to the response for the annual births per country in the year 2020 amidst the Covid pandemic was the availability of Family Planning measures to the women and the couples or people on the relationship in general. The country's fertility rate followed this in that country represented by the Live Births per Woman. The presence of the Health Care Index and the Cost-of-Living index, even though, were not proven to be significant, were still included in the model because these were deemed to be essential and could not be overlooked in performing the analysis.

Generally, the build of a regression model is very sensitive to outliers, and hence we considered the outliers present in all the predictor variables. These outliers could not be removed as well since India was the predominant outlier in all variables.

## TEXT MINING AND SENTIMENTAL ANALYSIS

We also decided to go ahead and perform some sentimental analysis on Tweets regarding the pandemic situation and its effect on the birthrate. We scraped all the tweets having pandemics and birth rate as hashtags and then used the well-known regular text mining process to generate some information on what people think about the impact of this pandemic on the birth rate. We have not explicitly used baby boom or baby bust as hashtags because we wanted to get that information by analyzing the public opinion on birth rate and pandemics. The text data we were going to be used for the analysis were first of all cleaned. The order of the cleaning of the files included:

1. Removing the stop words in the text.

2. Removing the standalone numerical values since some of the numerical values may be affiliated with some text. E.g., Covid-19.
3. Removing all punctuation and extra spaces and formatting options.
4. Changing the words to lower cases so that two words with different capitalizations are not identified as separate words.
5. Removal of the words in the breakdown list that are not necessary to the analysis of the tweets.
6. Stemming/ Lemmatizing the words, so their root forms are the ones that are kept.
7. Post-stemming visual analysis to see if any more words need to be removed.
8. Tokenizing the words and converting texts into a document term matrix

After performing the steps, we did the topic analysis, generated a word cloud, assessed different emotions/feelings, generated a classification model, and performed a named entity recognition analysis. The codes are available in the script: sentiment\_analysis.ipynb and sentiment.R.

### **Topic Analysis:**

Top 10 words for topic #0:

```
['immigr', 'year', 'coronaviru', 'push', 'popul', 'rise', 'say', 'think',  
'econom', 'declin']
```

Top 10 words for topic #1:

```
['year', 'ha', 'peopl', '2020', 'japan', 'acceler', 'covid19', 'birth', 'drop',  
'declin']
```

Top 10 words for topic #2:

```
['becaus', 'boom', 'guarante', 'half', 'born', 'bounc', 'million', 'fewer',  
'2021', 'babi']
```

Top 10 words for topic #3:



```
['educ', 'spiral', 'climat', 'includ', 'global', 'cost', 'health', 'child',  
'chang', 'care']
```

As can be seen, words like coronavirus, population, decline, drop, and birth as the top words in the topics indicate that people are talking about covid 19 as a pandemic on the drop of birthrate. The words like the *economy*, *education*, *cost*, *health*, and *care* indicate the drop's anticipated reasons. **Figure 28** shows the list of the top 15 words from the tweets with their counts. **Figure 29** shows the word cloud of the tweets. Similar to the findings from the topic analysis, the word cloud also shows that the decline (or drop) in the birthrate is the one people are most talking about.

	word	n
1	declin	28
2	babi	22
3	u.	22
4	birth	21
5	covid	19
6	drop	17
7	care	16
8	coronavirus	15
9	chang	14
10	popul	13
11	rate	13
12	child	12
13	peopl	12
14	acceler	11
15	japan	11

Figure 2828: Overall top 15 stemmed terms from the tweets with their frequency

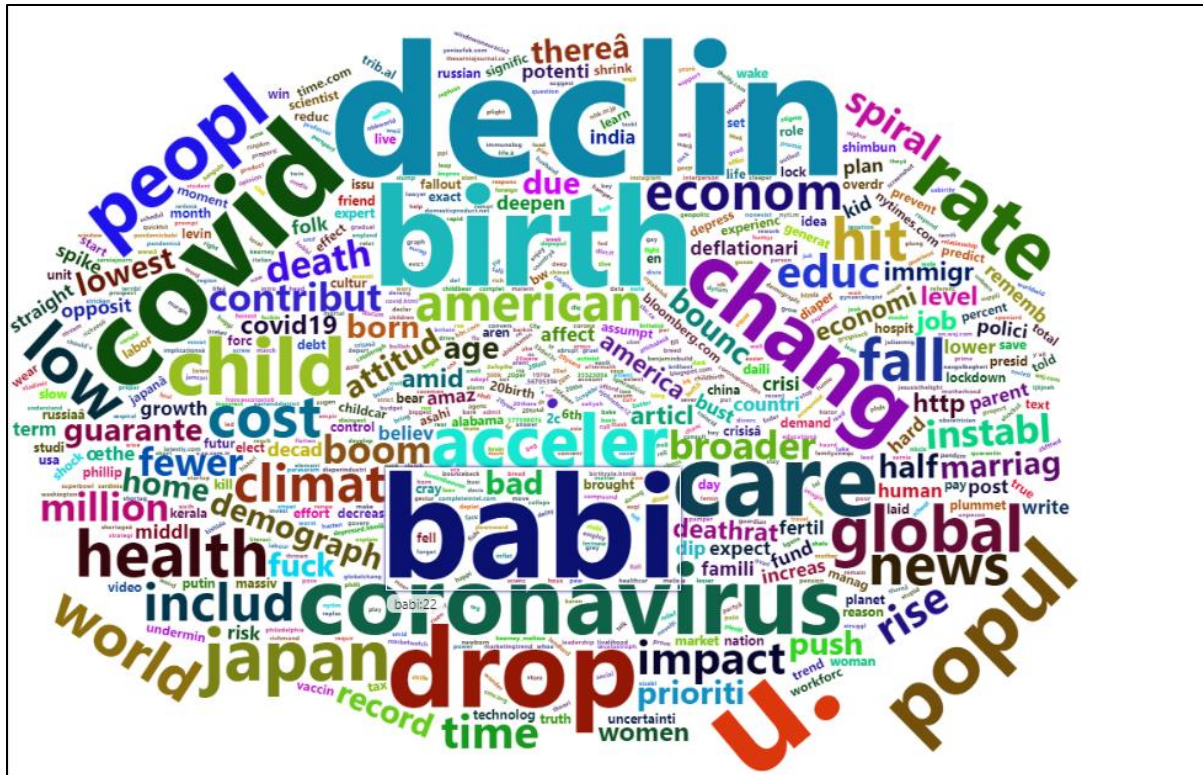


Figure 2929: Word cloud of most frequent word used in the tweets regarding pandemic and birthrate

## Emotions and Feelings

## Surprise and Anticipation

	word	anticipation	surprise	supriseness	linenumber
1	shock	0	2	2	24
2	abrupt	0	1	1	1
3	alarm	0	1	1	2
4	rapid	0	1	1	19
5	stagger	0	1	1	25
6	death	6	6	0	6
7	expect	3	3	0	10
8	labor	2	2	0	13
9	brilliant	1	0	-1	4
10	depart	1	0	-1	7
11	develop	1	0	-1	8
12	enjoy	1	0	-1	9
13	gradual	1	0	-1	11
14	grow	1	0	-1	12
15	mother	1	0	-1	14
16	plight	1	0	-1	17
17	result	1	0	-1	20
18	seek	1	0	-1	23
19	store	1	0	-1	27
20	vow	1	0	-1	29

Figure 3030: Surprise vs. Anticipation on the tweets

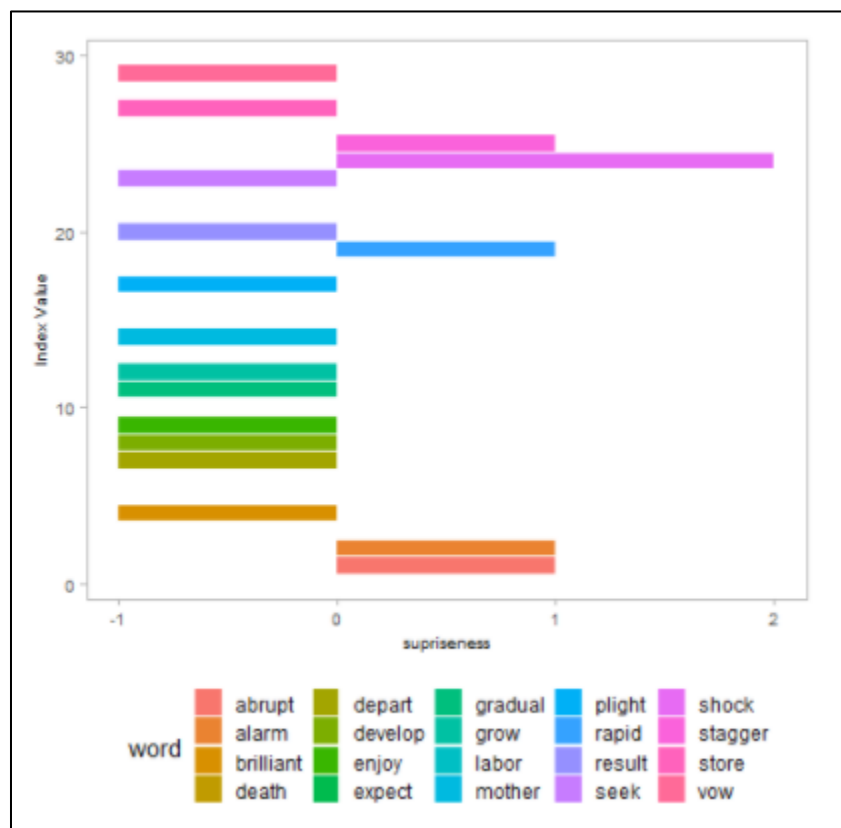


Figure 3131: Degree of surprise (anticipation-surprise) associated with words from the tweets

## Joy and Sadness

	word	joy	sadness	contentment	linenumber
1	birth	21	0	21	4
2	child	12	0	12	6
3	friend	2	0	2	18
4	labor	2	0	2	22
5	pay	2	0	2	29
6	save	2	0	2	31
7	true	2	0	2	38
8	brilliant	1	0	1	5
9	enjoy	1	0	1	14
10	grow	1	0	1	19
11	motherhood	1	0	1	28
12	vow	1	0	1	39
13	honest	1	1	0	20
14	mother	1	1	0	27
15	aftermath	0	1	-1	1
16	beg	0	1	-1	3
17	dark	0	1	-1	7
18	delay	0	1	-1	10
19	depart	0	1	-1	11
20	die	0	1	-1	13

Figure 3232: Joy and sadness associated with the tweets

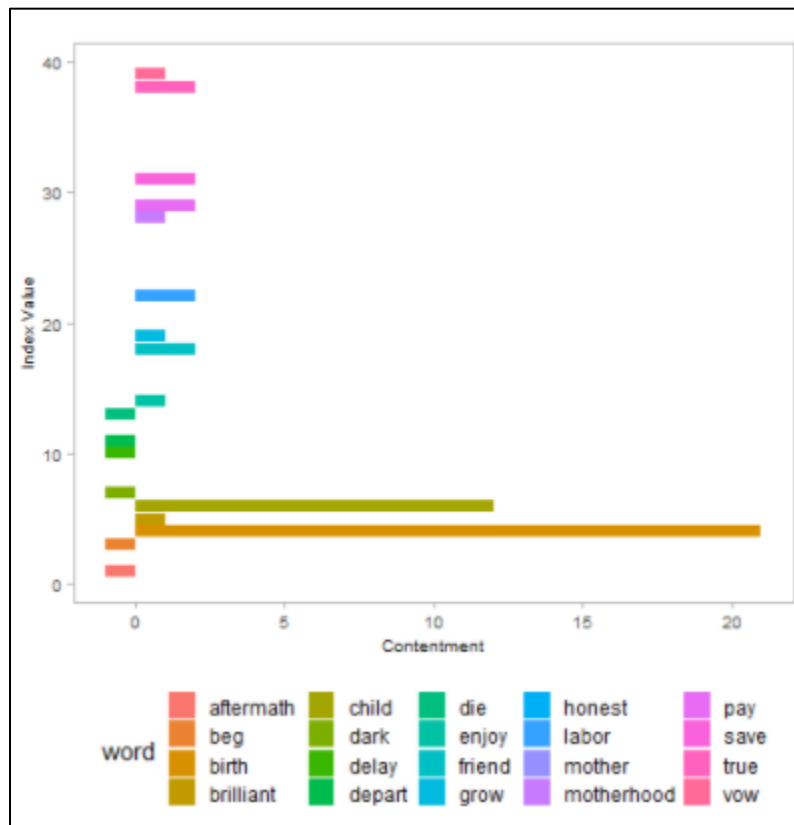


Figure 3333: Degree of contentment (joy-sadness) associated with words from the tweets

The sentiment analysis of the tweets revealed that there was a proper balance of anticipation and surprise (fig. 32) among the people while sharing their thoughts; for a few, the drop came out of no surprise given the economic crisis people will face due to the pandemic, but few people really thought that there would be a rise in birthrate just because the couples will have more time to spend together due to the lockdown. This was certainly not the case not only for the USA but for many other countries, as our study revealed.

## **RESULTS FROM NAMED ENTITY RECOGNITION**

The person named entity recognition shows a decline in the birth rate; it mentions depression, COVID-19, and the names of some world leaders. The location named entity says Europe, and a conclusion cannot be made from just one entity. Date named entity focuses on 2020 (the year under scrutiny), the year after that (2021), and the year before it (2019). This entity makes mention of 1918, which happens to be a pandemic year as well. Finally, the organization named entity indicates a drop in the birthrate of the U.S.

## **CONCLUSION**

Following the regression analysis performed on the variables from the baby.csv file, we identified the family planning predictor as the most critical predictor in identifying the country's annual birth data trends. This is followed by the log of the fertility rate of women in the country and then Quality of Life. These factors go to determine how high or how low the numbers recorded are. The four topics produced by the sentiment analysis leaned toward the decline in the birth rate generally across the world and their effects on education, cost of services, and healthcare services. Results from named entity recognition indicate a fall in the U.S. population, emphasizing the effect of the decline in population in subsequent years.

Results from named entity recognition indicate a fall in the U.S. population, emphasizing the effect of the decline in population in subsequent years. Therefore, it can be concluded that there was a baby bust during the pandemic year.

## APPENDIX

### LIST OF FIGURES

Figure 1: GANTT Chart .....	5
Figure 2: Web Source .....	6
Figure 3: Dashboard showing how birthrate from 2019 and 2020 are different for different countries and continents.....	10
Figure 4: Dashboard showing how birthrate from 2019 and 2020 are different for different countries from different continents. Countries in Asia and Europe they all faced baby bust, Africa was one of the continents with more boom than bust. ....	11
Figure 5: Distribution of the variable Cost of Living Index .....	12
Figure 6: Normality plot for the Cost-of-Living Index .....	12
Figure 7: Distribution of the variable Rent Index .....	13
Figure 8: Normality plot for Rent Index.....	13
Figure 9: Distribution of the variable Health Care Index .....	14
Figure 10: Normality plot for Health Care Index .....	14
Figure 11: Distribution of the variable Health Care Expertise .....	15
Figure 12: Normality plot for Health Care Expertise .....	15
Figure 13: Distribution of the variable Quality of Life Index .....	16
Figure 14: Normality plot for Quality-of-Life Index .....	16
Figure 15: Distribution of the variable Family Planning .....	17
Figure 16: Normality plot for Family Planning.....	17
Figure 17: Distribution of the variable Live Births per Woman per Country .....	18
Figure 18: Normality plot for Live Births per Woman per Country .....	18
Figure 19: Distribution of the variable Annual Births per Country .....	19
Figure 20: Normality plot for Annual Births per Country .....	19
Figure 21: Correlation Matrix of Variables .....	20
Figure 22: Multiple Regression with all factors .....	21
Figure 23: Summary of fit for Model 1 .....	21
Figure 24: Parameter Estimates of Model 1 .....	21
Figure 25: Regression with the selected factors.....	22

Figure 26: Summary of fit for Model 2 .....	22
Figure 27: Parameter estimates for Model 2 .....	23
Figure 28: Overall top 15 stemmed terms from the tweets with their frequency .....	25
Figure 29: Word cloud of most frequent word used in the tweets regarding pandemic and birthrate .....	26
Figure 30: Surprise vs Anticipation on the tweets .....	27
Figure 31: Degree of surprise (anticipation-surprise) associated with words from the tweets.....	27
Figure 32: Joy and sadness associated with the tweets.....	28
Figure 33: Degree of contentment (joy-sadness) associated with words from the tweets .....	28

## CODES

### Setiment analysis.ipynb

```

import selenium
import pandas as pd
from selenium import webdriver
from selenium.webdriver.support.ui import WebDriverWait
from selenium.webdriver.support import expected_conditions as EC
from selenium.webdriver.common.by import By
from selenium.common.exceptions import TimeoutException
from selenium.webdriver.common.keys import Keys
import time

import nltk
from nltk.stem import PorterStemmer
import matplotlib.pyplot as plt
from nltk import word_tokenize, sent_tokenize
from nltk.corpus import stopwords

```

```

from nltk.stem import LancasterStemmer, WordNetLemmatizer, PorterStemmer
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.decomposition import LatentDirichletAllocation
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import classification_report, confusion_matrix,
accuracy_score, plot_confusion_matrix

from nltk import word_tokenize, pos_tag, ne_chunk
from nltk.chunk import conlltags2tree, tree2conlltags

url="https://twitter.com/login"
#driver =
webdriver.Chrome(executable_path=r"D:\chromedriver_win32\chromedriver.exe")
driver = webdriver.Firefox(executable_path=r"D:\geckodriver.exe")
driver.get(url)
username=driver.find_element_by_xpath('//*[@id="layers"]/div[2]/div/div/div/div/div/div[2]/div[2]/div/div/div[2]/div[2]/div[1]/div/div[5]/label/div/div[2]/div/input')
username.send_keys('rupam_27'+Keys.ENTER)

userpass=driver.find_element_by_name("password")
userpass.send_keys('*****'+Keys.ENTER)

search=driver.find_element_by_xpath('//*[@id="react-root"]/div/div/div[2]/main/div/div/div/div[2]/div/div[2]/div/div/div[1]/div/div/div/form/div[1]/div/div/div/div[2]/div/input')
search.send_keys("pandemic birthrate"+Keys.ENTER)

time.sleep(2) # Allow 2 seconds for the web page to open
scroll_pause_time = 1 # You can set your own pause time. My laptop is a bit slow so I use 1 sec
screen_height = driver.execute_script("return window.screen.height;") # get the screen height of the web

```



```

i = 1
n=1
test_list=[]
#while True:
while n<=100:
    # scroll one screen height each time
    driver.execute_script("window.scrollTo(0,
{screen_height}*{i});".format(screen_height=screen_height, i=i))
    i += 1
    n+=1
    time.sleep(scroll_pause_time)
    # update scroll height each time after scrolled, as the scroll height can
change after we scrolled the page
    scroll_height = driver.execute_script("return document.body.scrollHeight;")
    # Break the loop when the height we need to scroll to is larger than the
total scroll height
    test=driver.find_elements_by_xpath('//div[@class="css-
1dbjc4n"]//div[@class="css-1dbjc4n"]//div[@lang="en"]')
    for j in range(len(test)):
        test_list.append(test[j].text)
    if (screen_height) * i > scroll_height:
        break
text=pd.DataFrame(test_list,columns=['tweets'])
text.drop_duplicates(inplace=True,ignore_index=True)

### Create a Term-Document Matrix
# Remove stop words
stop = stopwords.words('english')
text = pd.DataFrame(text)
textcol = text['tweets']
textcol = textcol.apply(lambda x: " ".join(x for x in x.split() if x not in
stop))

# Remove numerical values
patternnum = '\b[0-9]+\b'

```

```

textcol = textcol.str.replace(patternnum, '')
# Remove punctuation
patternpunc = '[^\w\s]'
textcol = textcol.str.replace(patternpunc, '')
# Convert to lowercase
textcol = textcol.apply(lambda x: " ".join(x.lower() for x in x.split()))
# Stem the words
porstem = PorterStemmer()
textcol = textcol.apply(lambda x: " ".join([porstem.stem(word) for word in
x.split()])))

#remove https
pat_http='^http?://'
textcol = textcol.str.replace(pat_http, '')

# Convert data into a document matrix
vectorizer = CountVectorizer()
tokens = pd.DataFrame(vectorizer.fit_transform(textcol).toarray(),
columns=vectorizer.get_feature_names())
tokens.columns
print(tokens.columns.tolist())

# LDA
vectorizer = CountVectorizer(max_df=0.8, min_df=4, stop_words='english')
tweet_values = textcol.values.astype('U') #convert Panda values to unicode
doc_term_matrix = vectorizer.fit_transform(tweet_values)
doc_term_matrix.shape
LDA = LatentDirichletAllocation(n_components=4, random_state=35)
LDA.fit(doc_term_matrix)

for i,topic in enumerate(LDA.components_):
    print(f'Top 10 words for topic #{i}:')
    print([vectorizer.get_feature_names()[i] for i in topic.argsort()[-10:]])
    print('\n')

```

```
text.to_csv("tweets on pandemic birth rate.csv")
```

## Sentiment.R

```
library(tidytext)
```

```
library(SnowballC)
```

```
library(tidyverse)
```

```
library(wordcloud2)
```

```
library(RColorBrewer)
```

```
library(tm)
```

```
tweets_df = read.csv('tweets on pandemic birth rate.csv', header = TRUE)
```

```
summary(tweets_df)
```

```
tweets_data=select(tweets_df,tweets)
```

```
tidy_dataset=unnest_tokens(tweets_data,word,tweets)
```

```
head(tidy_dataset)
```

```
counts = count(tidy_dataset, word)
```

```
result1 = arrange(counts, desc(n))
```

```
typeof(result1)
```

```
slice(result1,1:15)
```

```
#removing stop words
```

```
data("stop_words")
```

```
tidy_dataset2 = anti_join(tidy_dataset, stop_words)
```

```
counts2 = count(tidy_dataset2, word)
```

```
result2 = arrange(counts2, desc(n))
```

```
typeof(result2)
```

```
slice(result2,1:15)
```

```
#removing numerical values (and blank spaces)
```

```
patterndigits = '\\b[0-9]+\\b'
```

```
tidy_dataset2$word=str_remove_all(tidy_dataset2$word, patterndigits )
```

```
head(tidy_dataset2)
```

```

counts3 = count(tidy_dataset2, word)
result3 = arrange(counts3, desc(n))
slice(result3,1:15)

#removing certain words
list_remove=c("pandemic","birthrate", "https", "tmobilesprint", "â")
tidy_dataset3 = filter(tidy_dataset2, !(word %in% list_remove))
counts4= count(tidy_dataset3,word)
result4=arrange(counts4,desc(n))
slice(result4,1:15)

#removing new lines
tidy_dataset3$word=str_remove_all(tidy_dataset3$word, '\r?\n')
counts5= count(tidy_dataset3,word)
result5=arrange(counts5,desc(n))
slice(result5,1:15)

#removing spacing and tabs
tidy_dataset3$word = str_replace_all(tidy_dataset4$word, '[ \t]', '')
tidy_dataset4=filter(tidy_dataset3, !(word=='') )
counts6= count(tidy_dataset4,word)
result6=arrange(counts6,desc(n))
slice(result6,1:15)

tidy_dataset5 = mutate_at(tidy_dataset4, "word", funs(wordStem(.),
language="en"))

counts5 = count(tidy_dataset5, word)

arrange(counts5, desc(n)) %>%
  ungroup %>%
  slice(1:15)

install.packages('textdata')
library(textdata)

```

```

get_sentiments('nrc') %>%
  distinct(sentiment)

#JOY and SADNESS
nrc_joysad = get_sentiments('nrc') %>%
  filter(sentiment == 'joy' |
          sentiment == 'sadness')
nrow(nrc_joysad)
newjoin2 = inner_join(tidy_dataset5, nrc_joysad)
counts8 = count(newjoin2, word, sentiment)
spread2 = spread(counts8, sentiment, n, fill = 0)
content_data = mutate(spread2, contentment = joy - sadness, linenumber =
row_number())
tweet_joysad = arrange(content_data, desc(contentment))

#generating plot of top 20
(tweet_joysad2 = tweet_joysad %>%
  slice(1:20,107:127))
ggplot(tweet_joysad2, aes(x=linenumber, y=contentment, fill=word)) +
  coord_flip() +
  theme_light(base_size = 15) +
  labs(
    x='Index Value',
    y='Contentment'
  ) +
  theme(
    legend.position = 'bottom',
    panel.grid = element_blank(),
    axis.title = element_text(size = 10),
    axis.text.x = element_text(size = 10),
    axis.text.y = element_text(size = 10)
  ) +
  geom_col()

```

```

#surprise and anticipation
nrc_surprise_anticipation = get_sentiments('nrc') %>%
  filter(sentiment == 'surprise' |
    sentiment == 'anticipation')
nrow(nrc_surprise_anticipation)
(tweet_surprise_anticipation = tidy_dataset5 %>%
  inner_join(nrc_surprise_anticipation) %>%
  count(word, sentiment) %>%
  spread(sentiment, n, fill = 0) %>%
  mutate(supriseness = surprise - anticipation, linenumber = row_number()) %>%
  arrange(desc(supriseness)) %>%
  slice(1:20, 318:338))
ggplot(tweet_surprise_anticipation, aes(x=linenumber, y=supriseness, fill=word))
+
  coord_flip() +
  theme_light(base_size = 15) +
  labs(
    x='Index Value',
    y='supriseness'
  ) +
  theme(
    legend.position = 'bottom',
    panel.grid = element_blank(),
    axis.title = element_text(size = 10),
    axis.text.x = element_text(size = 10),
    axis.text.y = element_text(size = 10)
  ) +
  geom_col()

#Wordcloud
df=data.frame(counts5)
set.seed(1234)
wordcloud(words = df$word, freq = df$n, min.freq = 1,
max.words=200, random.order=FALSE, rot.per=0.35,
colors=brewer.pal(8, "Dark2"),scale=c(8,0.5))

```

```
set.seed(1234)
wordcloud2(data=df, size=1, color='random-dark')
```

### Data-Combined.R

```
library(rvest)
library(xml2)
library(stringr)
library(reshape)

setwd("C:/Users/quoej/OneDrive/Desktop/MS BAnDS OSU/MSIS 5193 Programming/Data
Files")

new_data = read.csv("owid-covid-data.csv")
new_data2 = read.csv("annual-number-of-births-by-world-region.csv")
new_data3 = read.csv("cleaned family planning data for married women.csv")
new_data4 = read.csv("children-per-woman-UN.csv")
names(new_data)
names(new_data2)
names(new_data3)
names(new_data4)

#new_data[is.na(new_data)] = 0
new_data$year = format(as.Date(new_data$date, format = "%Y-%m-%d"), "%Y")

all = cbind(new_data$new_cases, new_data$new_deaths,
new_data$new_cases_per_million, new_data$new_deaths_per_million,
            new_data$hosp_patients_per_million,
new_data$hospital_beds_per_thousand, new_data$icu_patients_per_million)
all2 = cbind(new_data$population, new_data$gdp_per_capita)
new_datai = aggregate(all ~ year + iso_code + location, data = new_data, FUN =
sum, na.rm = TRUE)
names(new_datai) = c("Year", "Iso_Code", "Location", "Cases", "Deaths",
"Cases_per_Million", "Deaths_per_Million",
                    "Hosp_Patients_per_Million", "Hospital_Beds_per_Thousand",
"Icu_Patients_per_Million")
```

```

new_dataii = aggregate(all2 ~ year + location, data = new_data, FUN = mean, na.rm
= TRUE)
names(new_dataii) = c("Year", "Location", "Population", "GDP")
baby_covid = merge(new_datai, new_dataii, by = c("Location", "Year"))
baby_fertility = new_data2[(new_data2$Year == 2019) | (new_data2$Year == 2020), ]
names(baby_fertility)[1] = "Country"
names(baby_fertility)[4] = "Annual Births per Country"
names(new_data3)[1] = "Country"
names(new_data3)[2] = "Family Planning"
names(new_data4)[1] = "Country"
names(new_data4)[4] = "Live Births per Woman per Country"

link = "https://www.numbeo.com/cost-of-living/rankings_by_country.jsp?title=2020"
scrap = read_html(link)
country_selector = '#t2 > tbody > tr > td.cityOrCountryInIndicesTable.sorting_1'
find_code = html_nodes(scrap, country_selector)
CostOfLiving = html_table(scrap, fill = TRUE)
COL = data.frame(CostOfLiving[[2]])

link1 = "https://www.numbeo.com/health-care/rankings_by_country.jsp?title=2020"
scrap1 = read_html(link1)
country_selector1 = '#t2 > tbody > tr > td.cityOrCountryInIndicesTable.sorting_1'
find_code1 = html_nodes(scrap1, country_selector)
HealthCareIndex = html_table(scrap1, fill = TRUE)
HCI = data.frame(HealthCareIndex[[2]])

link2 = "https://www.numbeo.com/quality-of-
life/rankings_by_country.jsp?title=2020"
scrap2 = read_html(link2)
country_selector2 = '#t2 > tbody > tr > td.cityOrCountryInIndicesTable.sorting_1'
find_code2 = html_nodes(scrap2, country_selector)
QualityOfLife = html_table(scrap2, fill = TRUE)
QOF = data.frame(QualityOfLife[[2]])
names(COL)
names(HCI)

```



```

names(QOF)

cf = merge(x=COL, y=HCI, by="Country", all.x=TRUE)
cf = data.frame(cf)
df = merge(x=cf, y=QOF, by="Country", all.x=TRUE)
names(df)
new = merge(df, baby_fertility[(baby_fertility$Year == 2020), ], by="Country",
all.x=TRUE)
new = merge(new, new_data3, by="Country", all.x=TRUE)
new = merge(new, new_data4[(new_data4$Year == 2020), ], by="Country", all.x=TRUE)

baby_covid[, c(1,2,3,11,12,4,5,6,7,8,9,10)]
new[, c(1,23,22,4,10,11,13,25,28,24)]

write.csv(baby_covid[, c(1,2,3,11,12,4,5,6,7,8,9,10)], "Baby_Covid.csv")
write.csv(baby_fertility, "Baby_Fertility.csv")
write.csv(new[, c(1,23,22,3,4,10,11,13,25,28,24)], "Baby.csv")

```

### named\_entity\_project.py

```

import pandas as pd
import spacy

#reading in data
data=pd.read_csv('/Users/adwoaboadi-asamoah/Desktop/programming/project.csv')
dat= data['0'].to_list()
listi=' '.join(map(str,dat))

# Load SpaCy model
nlp = spacy.load("en_core_web_sm")
doc = nlp(listi)

entities = []
labels = []

```

```

for ent in doc.ents:
    entities.append(ent)
    labels.append(ent.label_)

# creating a dataframe for named entities and labels
df = pd.DataFrame({'Entities':entities,'Labels':labels})

# named entities of interest put into dataframes
loc=df[df['Labels']=='LOC']
person=df[df['Labels']=='PERSON']
org=df[df['Labels']=='ORG']
date=df[df['Labels']=='DATE']

# creating csv files from dataframes

loc.to_csv('/Users/adwoaboadi-asamoah/Desktop/programming/loc.csv')
person.to_csv('/Users/adwoaboadi-asamoah/Desktop/programming/person.csv')
org.to_csv('/Users/adwoaboadi-asamoah/Desktop/programming/org.csv')
date.to_csv('/Users/adwoaboadi-asamoah/Desktop/programming/date.csv')

```