Introduction to Data Science and AI – Report for Assignment 5

Big Data Technologies, Sofia University "St. Kliment Ohridski"

Stefan Dimitrov Velev, 0MI3400521

***Question 1**: In this question I had to work with data sets from **Our World in Data©** and **Python** so as to make thoughtful analyses and interesting visualisations.*

In the beginning let's analyse and process the data before making the visualisations.

The packages I've used in this assignment are: ***Pandas*** (for reading and storing data), ***NumPy*** (for making statistics), ***Matplotlib*** (for visualisation), and ***Scikit-learn*** (machine learning library that supports supervised and unsupervised learning).

I've downloaded three data sets from ***Our World in Data©*** which I will analyse later.

1. Self-reported life satisfaction

Average of responses to the 'Cantril Ladder' question in the Gallup World Poll. The survey asks respondents to think of their current place on a ladder, with the best possible life for them being a 10, and the worst possible life being a 0.

Data URL: *https://ourworldindata.org/grapher/happiness-cantril-ladder?time=latest*

Data sources: *World Happiness Report (2012-2024); Wellbeing Research Centre (2024); Population based on various sources (2023)*

The data is stored in a *csv file*. The name of the separate columns can be found in the *Jupyter notebook* accompanying this report. The *csv file* contains 1 787 entries. However, most of them will not be used for our purposes. Here comes the **data cleaning** part. First, I decided to remove the columns which are not required for the task. In that case it is only one column – 'Code'. After that, I renamed one of the left columns with more concise names:

'Entity' → 'Country'

The next step was to choose the year which I wanted to examine. Since I'd like up-to-date statistics, my focus was on 2021. The reason not to choose more contemporary year is that not all entries are updated for later period which would disrupt the analyses later. Then I dropped the rows with missing values (if there are any). After examining the data, I decided to remove the following aggregated entries with country field: 'High-income countries', 'Low-income countries', 'Lower-middle-income countries', 'Upper-middle-income countries', 'World'. The reason is that they are not necessary for the statistics about individual countries.

Since I wanted to analyse data for the separate continents (such data is also provided in the data set) as well, I decided to do **data segregation** and extract the entries for the continents into another data set. After that, I removed these rows from the original data set (without the entry for 'Australia' since it is a country and a continent). Finally, I had a *DataFrame* with 147 entries which is a satisfactory result since the total number of countries is 195 and I assume entries for some of them are probably missing or incomplete.

2. Share in extreme poverty vs. life expectancy

The period life expectancy at birth, in a given year. Extreme poverty is defined as living below the International Poverty Line of $2.15 per day.

Data URL: *https://ourworldindata.org/grapher/extreme-poverty-headcount-ratio-vs-life-expectancy-at-birth*

Data sources: *UN, World Population Prospects (2024); World Bank Poverty and Inequality Platform (2024); HYDE (2023); Gapminder - Population v7 (2022); Gapminder - Systema Globalis (2022)*

The data is stored in a *csv file*. The name of the separate columns can be found in the *Jupyter notebook* accompanying this report. The *csv file* contains 60 100 entries. However, most of them will not be used for our purposes. Here comes the **data cleaning** part. First, I decided to remove the columns which are not required for the task – 'Code', '990305-annotations', 'World regions according to OWID'. After that, I renamed some of the left columns with more concise names:

'Entity' → 'Country'

'Life expectancy – Sex: all – Age: 0 – Variant: estimates' → 'Life expectancy'

'$2.15 a day – Share of population in poverty' → 'Share in extreme poverty'

'Population (historical)' → 'Population'

The next step was to choose the year which I wanted to examine. Since I'd like up-to-date statistics, my focus was on 2021. The reason not to choose more contemporary year is that not all entries are updated for later period which would disrupt the analyses later. Then I dropped the rows with missing values (if there are any). After examining the data, I decided to remove the following aggregated entry with country field: 'World'. The reason is that it is not necessary for the statistics about individual countries.

While inspecting the data, I noticed that there was an outlier. That is for the life expectancy of the Central African Republic for 2021 – according to the *World Health Organization Data* for 2021[1] it is 52.31 years, not 40.279 years. For the purposes of the task and for better visualisation, I decided to correct it.

3. Political corruption index

Based on the expert estimates and index by V-Dem. It captures the extent to which the executive, legislative, judiciary, and bureaucracy engage in bribery and theft, and the making and implementing of laws are susceptible to corruption.

Data URL: https://ourworldindata.org/grapher/extreme-poverty-headcount-ratio-vs-life-expectancy-at-birth

Data sources: V-Dem (2024)

The data is stored in a *csv file*. The name of the separate columns can be found in the *Jupyter notebook* accompanying this report. The *csv file* contains 33 090 entries. However, most of them will not be used for our purposes. Here comes the **data cleaning** part. First, I decided to remove the columns which are not required for the task. In that case it is only one column – 'Code'. After that, I renamed some of the left columns with more concise names:

'Entity' → 'Country'

'Political corruption index (best estimate, aggregate: average)' → 'Political corruption index'

The next step was to choose the year which I wanted to examine. Since I'd like up-to-date statistics, my focus was on 2021. The reason not to choose more contemporary year is that not all entries are updated for later period which would disrupt the analyses later. Then I dropped the rows with missing values (if

[1] https://data.who.int/countries/140

there are any). After examining the data, I decided to remove the following aggregated entry with country field: 'World'. The reason is that it is not necessary for the statistics about individual countries.

Since I wanted to analyse data for the separate continents (such data is also provided in the data set) as well, I decided to do **data segregation** and extract the entries for the continents into another data set. After that, I removed these rows from the original data set (without the entry for 'Australia' since it is a country and a continent). Finally, I had a *DataFrame* with 180 entries which is a satisfactory result since the total number of countries is 195 and I assume entries for some of them are probably missing or incomplete.

For the visualisation tasks I've made several plots using different approaches. They would help me answer some meaningful questions. The rationale behind choosing these data sets is connected to the type of questions to which I wanted to receive an answer. We will start looking at the separate plots and to the questions that they answer. We will state any assumptions and motivate any decision when selecting data to be plotted, and in combining data. We will discuss any observations or insights obtained from the data visualisations.

I. A scatter plot of Share of Population in Extreme Poverty vs Life Expectancy for 2021

The first scatter plot I've presented is one made with the help of ***Pandas***. This diagram can only be used for an overall picture of how data is distributed. That is why the name of the countries is not shown on the scatter plot.

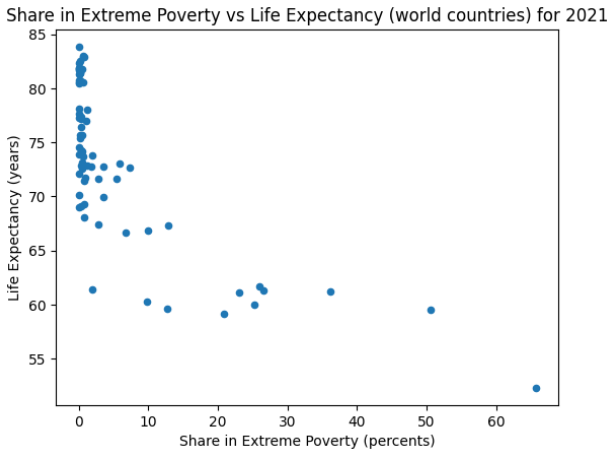

Figure 1: *Share in Extreme Poverty vs Life Expectancy (world countries) for 2021*

The most detailed scatter plot is the second one for which I've used ***Matplotlib***. For that, some considerations were taken into account:

- the scatter plot is much larger as the number of entries is 70 (most of them settled in one area)
- the circles of the scatter plot are proportional to the population of each country for 2021
- the name of each country can be found next to its circle – for better clarity it would be good if it is visualised only on hoovering over it; however this cannot be achieved with ***Matplotlib***

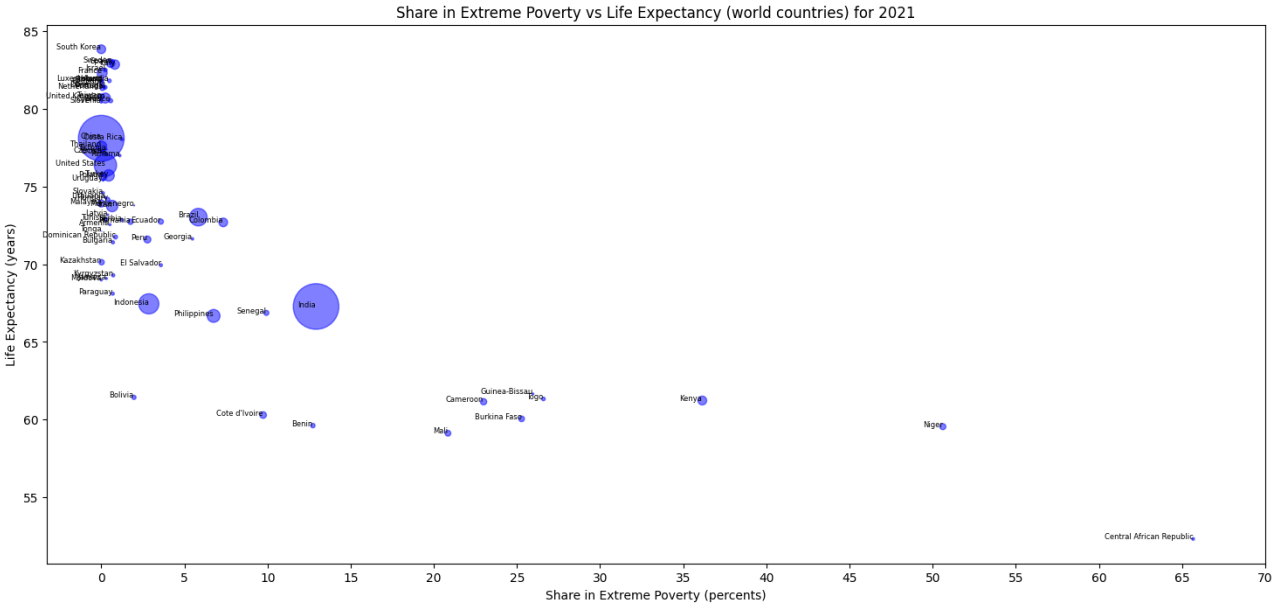

Figure 2: *Share in Extreme Poverty vs Life Expectancy (world countries) for 2021*

However large the scatter plot is, so many names of countries cannot be presented on just one diagram. That's why I've taken the decision to divide this scatter plot into three separate ones. The criteria used is 'Share in extreme poverty' – less than 0.5% in the first scatter plot, between 0.5% and 5% in the second scatter plot, above 5% in the third scatter plot. In that way, the names of the countries become clearer.

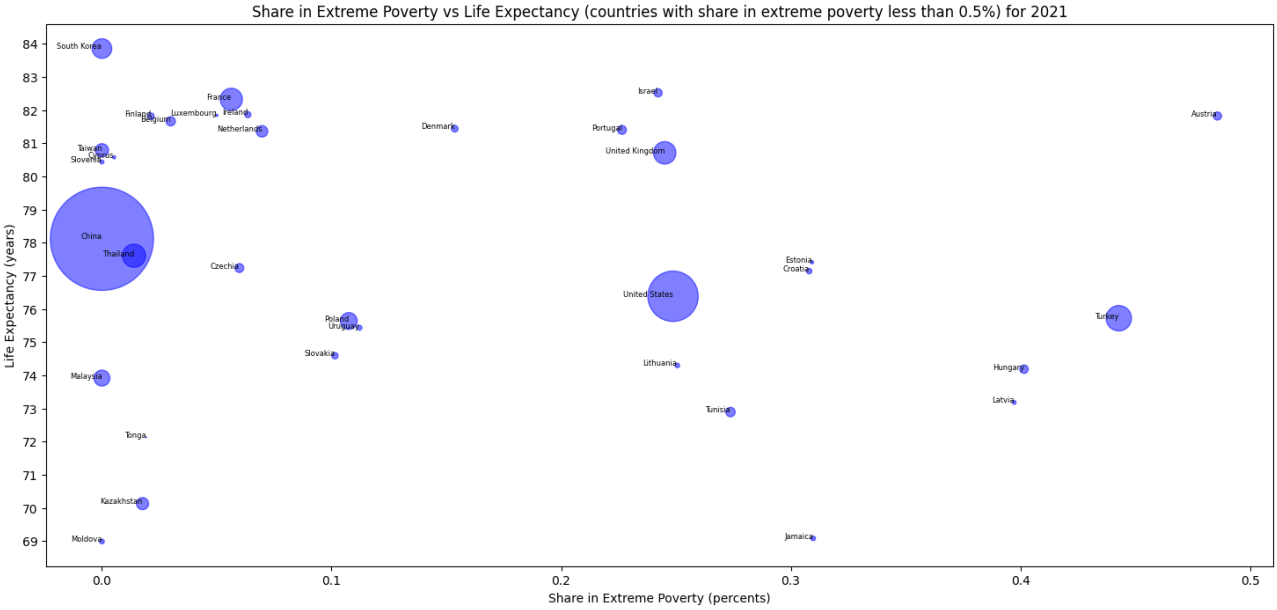

Figure 3: *Share in Extreme Poverty vs Life Expectancy (countries with share in extreme poverty less than 0.5%) for 2021*

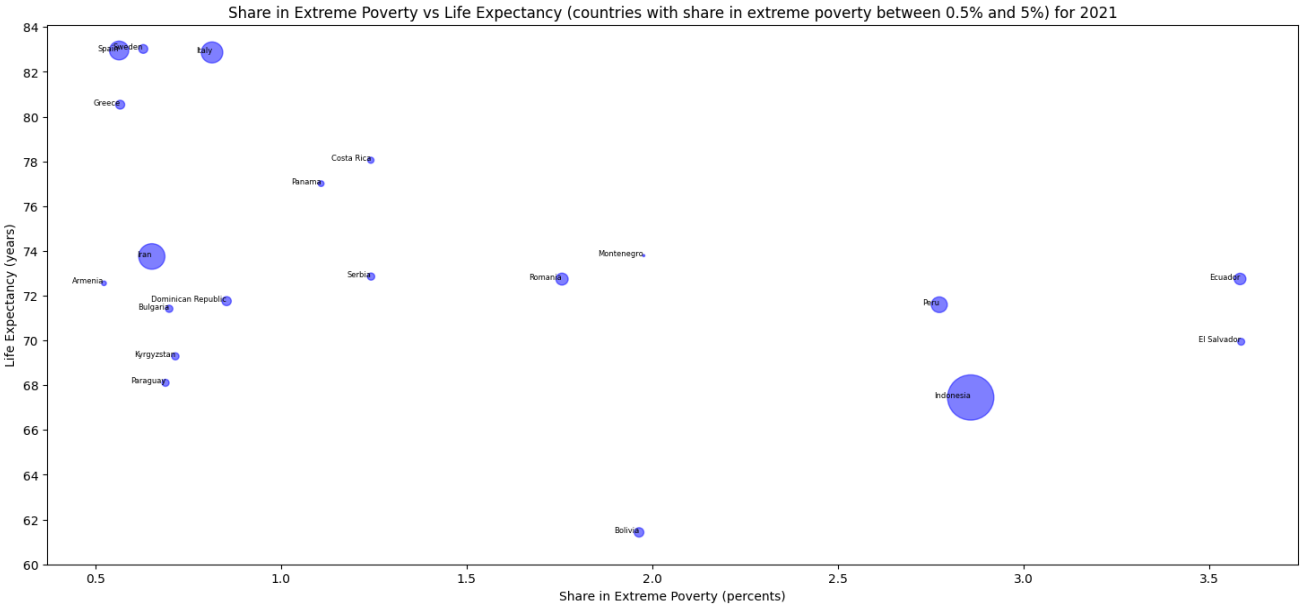

Figure 4: *Share in Extreme Poverty vs Life Expectancy (countries with share in extreme poverty between 0.5% and 5%) for 2021*

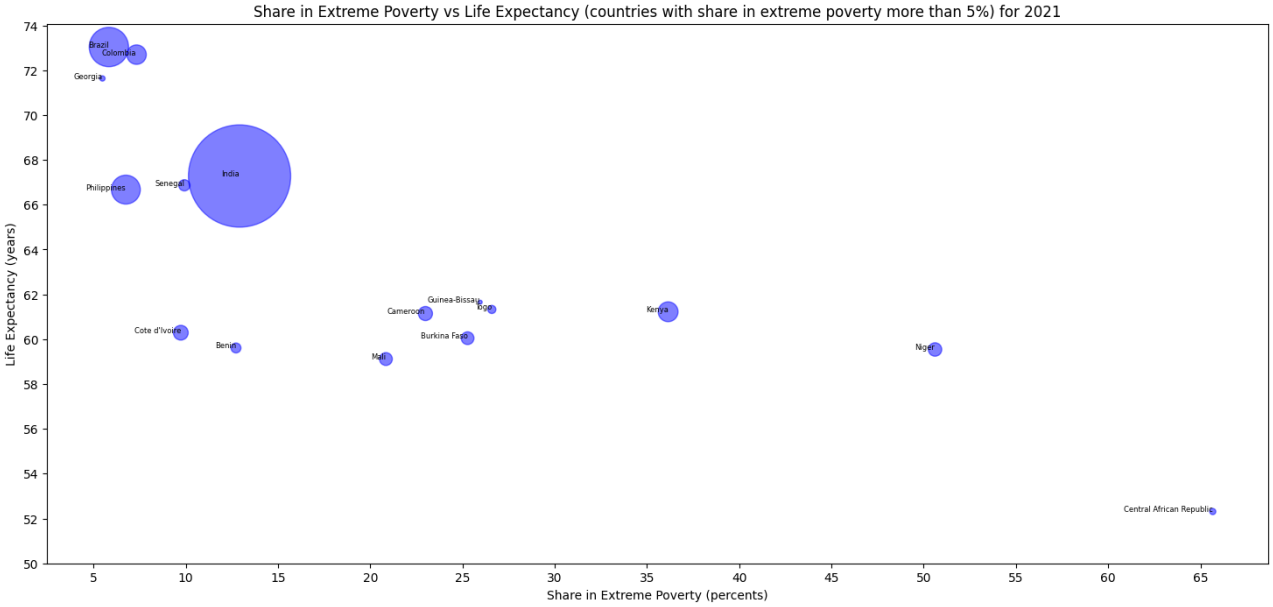

Figure 5: *Share in Extreme Poverty vs Life Expectancy (countries with share in extreme poverty more than 5%) for 2021*

Questions to be answered with the help of the scatter plot:

1. Check which countries have high life expectancy but have higher share in extreme poverty for 2021?

With ***NumPy*** I've determined that the median life expectancy of the sample is 73.77 and the median share in extreme poverty is 0.54. So, using ***Pandas*** I've concluded that the number of the countries for 2021 which have life expectancy above the found median and above the median share in extreme poverty is 7 and they are as follows:

Costa Rica, Greece, Italy, Montenegro, Panama, Spain, Sweden

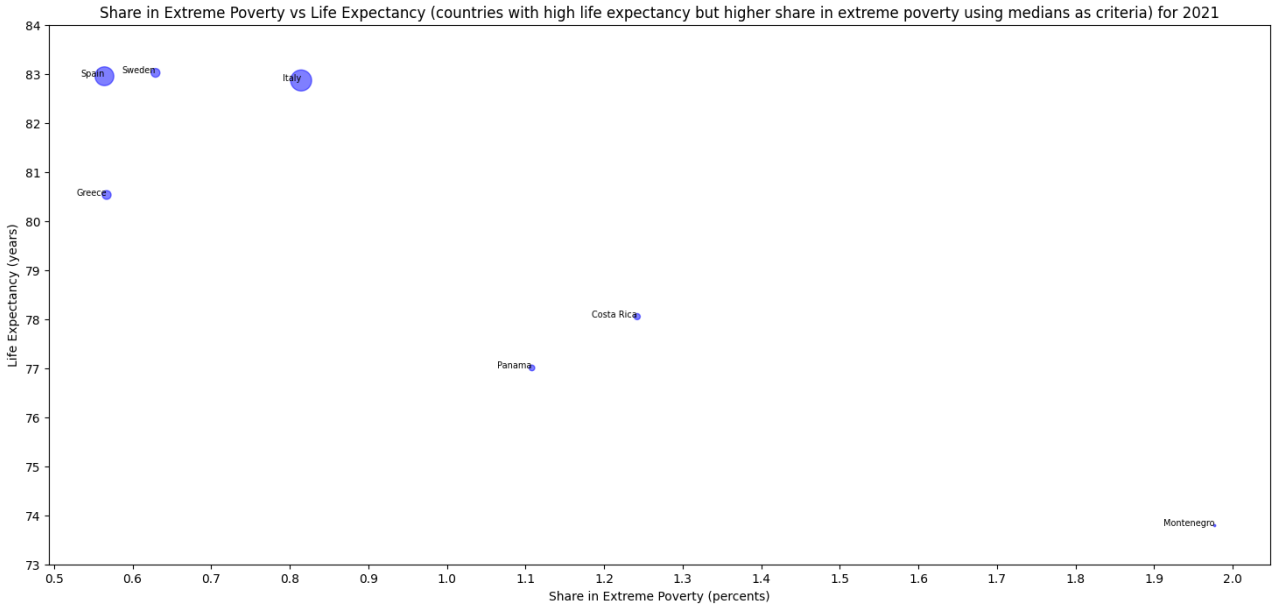

Figure 6: *Share in Extreme Poverty vs Life Expectancy (countries with high life expectancy but higher share in extreme poverty using medians as criteria) for 2021*

If we use the mean as criteria for high life expectancy and high share in extreme poverty, we will find that there are no such countries. This means we can draw a conclusion that the countries with high life expectancy tend to have a smaller share in extreme poverty. That is natural since high life expectancy means that local people maintain a healthier lifestyle which contrasts with extreme poverty. We can notice the great difference in the mean and the median for the column 'Share in extreme poverty'. The mean is 5.41 and the median is 0.54. By definition, the mean is the number we get by dividing the sum of a set of values by the number of values in the set. In contrast, the median is the middle number in a set of values when those values are arranged from smallest to largest. The big difference is due to the skewness of the data.

2. Find whether each country with lower share in extreme poverty have high life expectancy?

For defining lower share in extreme poverty, I use the 25th percentile of the column 'Share in extreme poverty'. This will be my upper boundary for the searched countries. The exact share is 0.1029%. With that consideration the searched number of countries is 18 and they are:

Belgium, China, Cyprus, Czechia, Finland, France, Ireland, Kazakhstan, Luxembourg, Malaysia, Moldova, Netherlands, Slovakia, Slovenia, South Korea, Taiwan, Thailand, Tonga

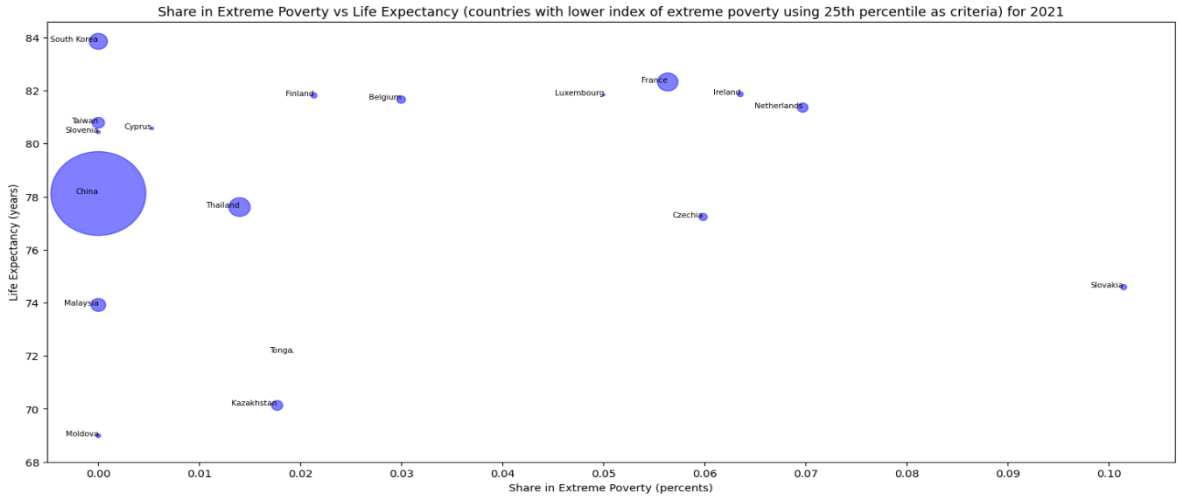

Figure 7: *Share in Extreme Poverty vs Life Expectancy (countries with lower index of extreme poverty using 25th percentile as criteria) for 2021*

For defining higher life expectancy, I use the 75th percentile of the column 'Life expectancy'. This will be my lower boundary for high life expectancy. In addition to that, I find the mean life expectancy and the 25th percentile representing the upper boundary for low life expectancy. I will provide the answer to the question by plotting and observing the following scatter plot:

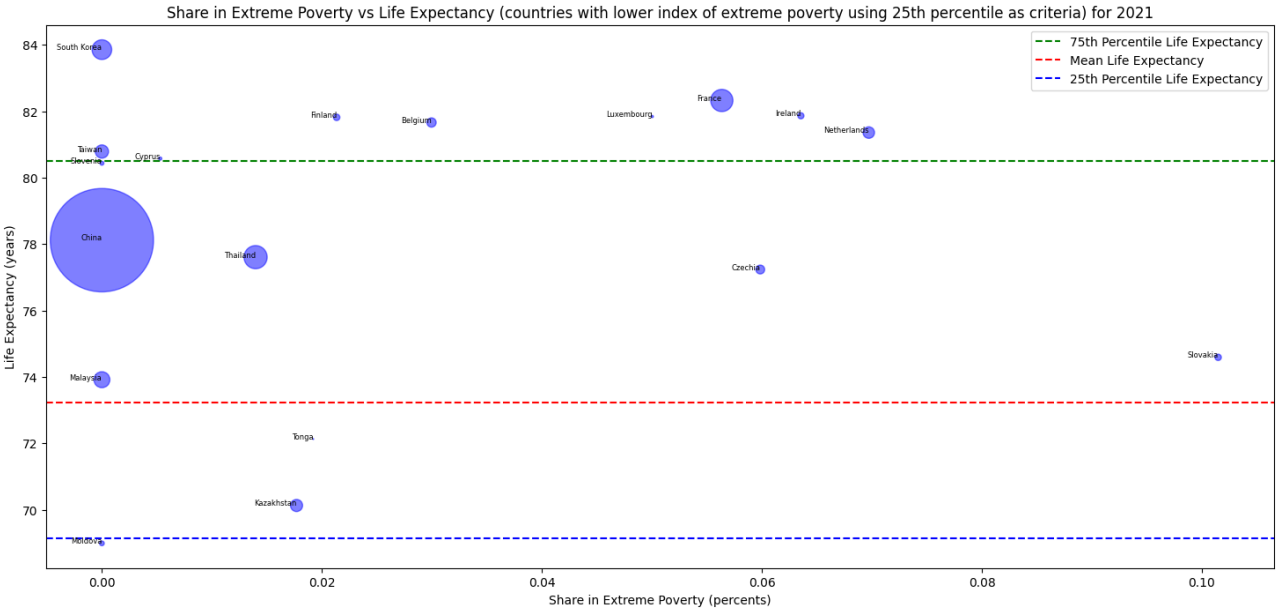

Figure 8: *Share in Extreme Poverty vs Life Expectancy*
(countries with lower index of extreme poverty using 25th percentile as criteria) for 2021

The circles represent each country with lower share in extreme poverty. Now we have to see if all of the above countries have high life expectancy. Apparently, 9 countries are above the 75th percentile of life expectancy which is enough to call them countries with higher level of life expectancy. 6 countries are within the mean and the 75th percentile of life expectancy. As they are above the average value, we can call them countries with high level of life expectancy. 2 of the countries are under the mean life expectancy but still above the 25th percentile which classifies them as countries with low life expectancy. There is one country (Moldova) that is below the 25th percentile of life expectancy which means that it is a country with one of the lowest life expectancies. This can answer the question that there are countries with lower share of extreme poverty that are still further away from high life expectancy standard.

From the obtained results for the 18th countries we can notice something odd. Countries like China, Malaysia, Moldova, Slovenia, South Korea, Taiwan have 0% share in extreme poverty. Most of these entries (e.g. Moldova) have missing values for that column. That is why it is important to verify the results for outliers.

II. A scatter plot of Self-Reported Life Satisfaction (Cantril Ladder Score) vs Political Corruption Index for 2021

I want to make a scatter plot merging two separate data sets. For that purpose, I used ***Pandas*** and its *merge* function to perform inner join on the 'Country' column. As a result, I get a data frame with 145 rows. On the following scatter plot I've done with the help of ***Pandas***, we can see the distribution of the data. This diagram can only be used for an overall picture. That is why the name of the countries is not shown on the scatter plot.

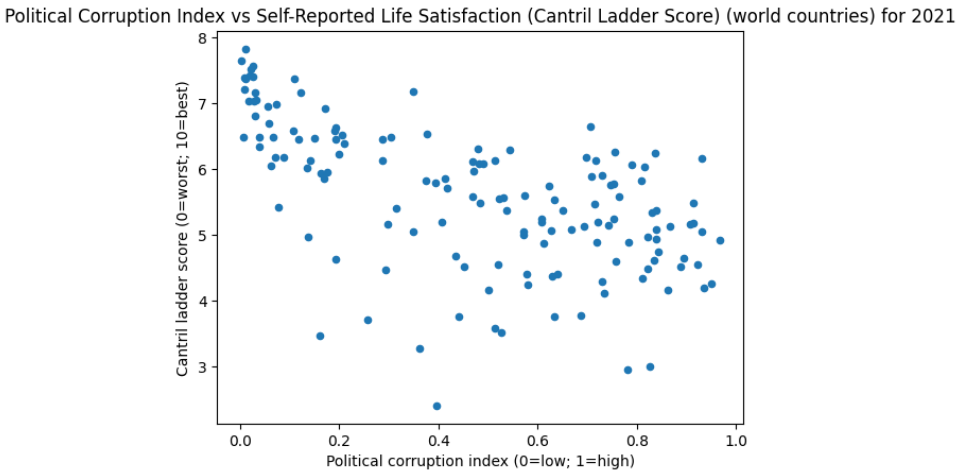

Figure 9: *Political Corruption Index vs Self-Reported Life Satisfaction (Cantril Ladder Score) (world countries) for 2021*

The most detailed scatter plot is the second one for which I've used ***Matplotlib***. For that, some considerations were taken into account:

- the scatter plot is much larger as the number of entries is 145 (most of them settled in one area)
- the name of each country can be found next to its circle – for better clarity it would be good if it is visualised only on hoovering over it; however this cannot be achieved with ***Matplotlib***

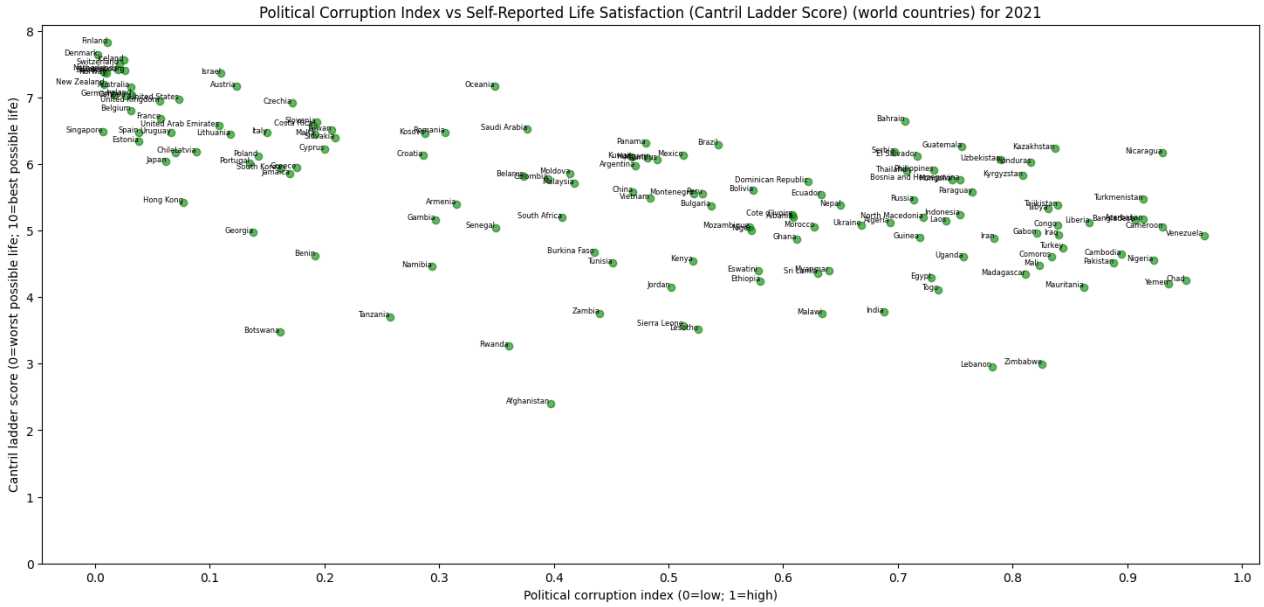

Figure 10: *Political Corruption Index vs Self-Reported Life Satisfaction (Cantril Ladder Score) (world countries) for 2021*

However large the scatter plot is, so many names of countries cannot be presented on just one diagram. That's why I've taken the decision to divide this scatter plot into two separate ones. The criteria used is 'Political corruption index' – less than 0.3% in the first scatter plot, above 0.3% in the second scatter plot. In that way, the names of the countries become clearer.

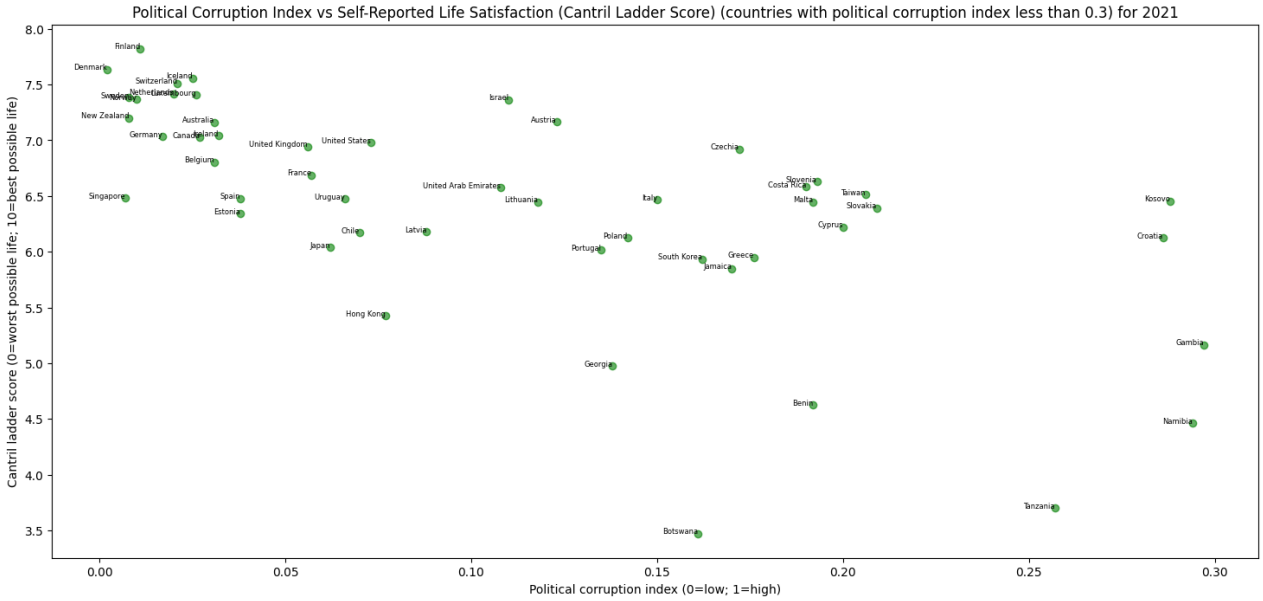

Figure 11: *Political Corruption Index vs Self-Reported Life Satisfaction (Cantril Ladder Score) (countries with political corruption index less than 0.3) for 2021*

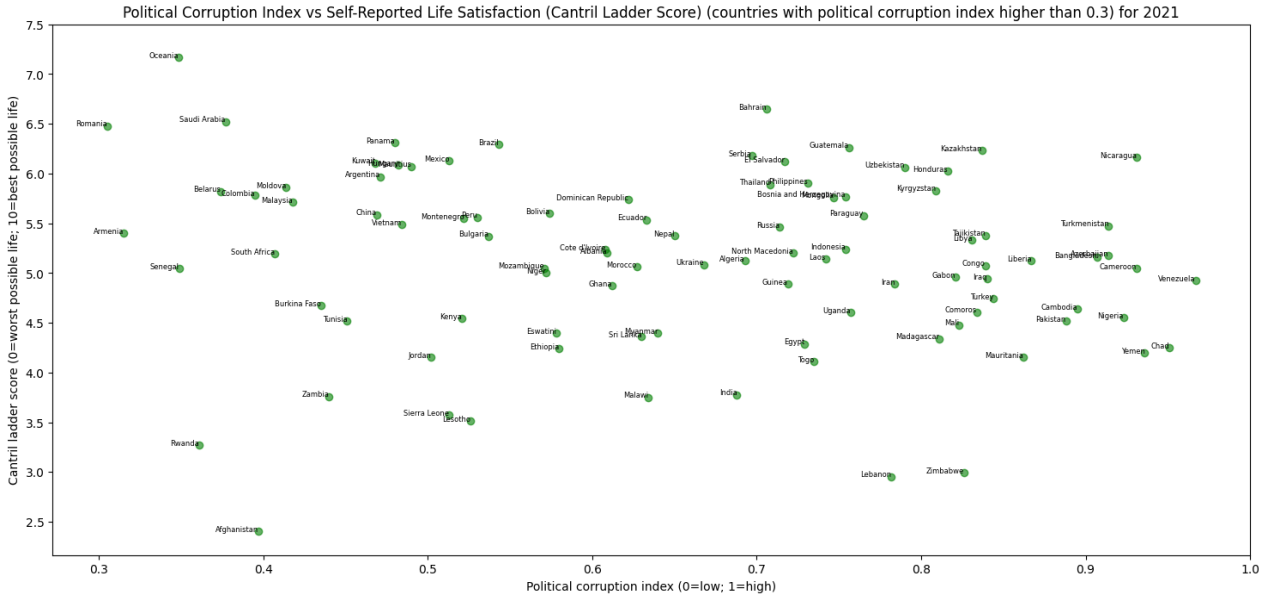

Figure 12: *Political Corruption Index vs Self-Reported Life Satisfaction (Cantril Ladder Score) (countries with political corruption index higher than 0.3) for 2021*

Questions to be answered with the help of the scatter plot:

1. Which are the top 10 happiest world countries (using the Cantril ladder score)?

With ***Pandas*** and its method *sort_values* performed on the merged data frame, I sorted the data frame on the 'Cantril ladder score' column in descending order and took the first 10 rows with the *head* method. The results I visualised using horizontal bar chart with the help of ***Matplotlib***:

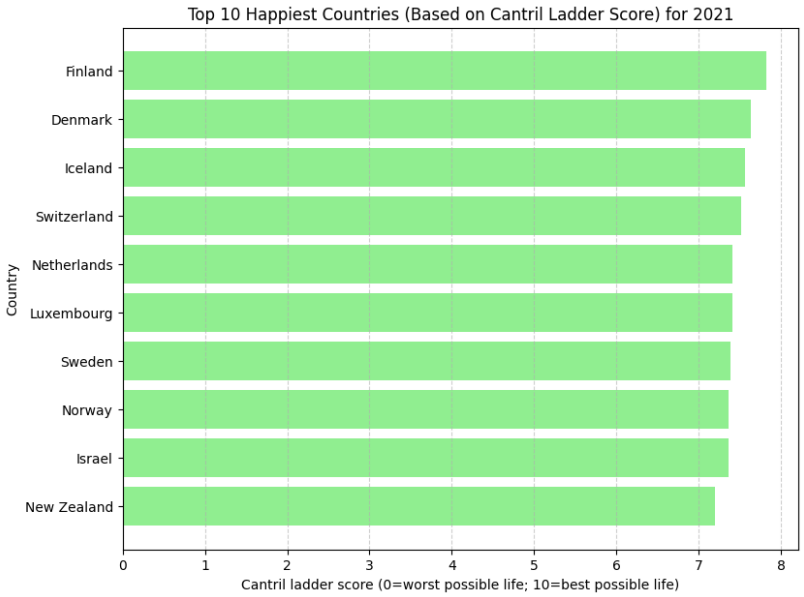

Figure 13: *Top 10 Happiest Countries (Based on Cantril Ladder Score) for 2021*

2. Which are the top 20 most politically corrupted world countries?

With ***Pandas*** and its method *sort_values* performed on the merged data frame, I sorted the data frame on the 'Political corruption index' column in descending order and took the first 20 rows with the *head* method. The results I visualised using horizontal bar chart with the help of ***Matplotlib***:

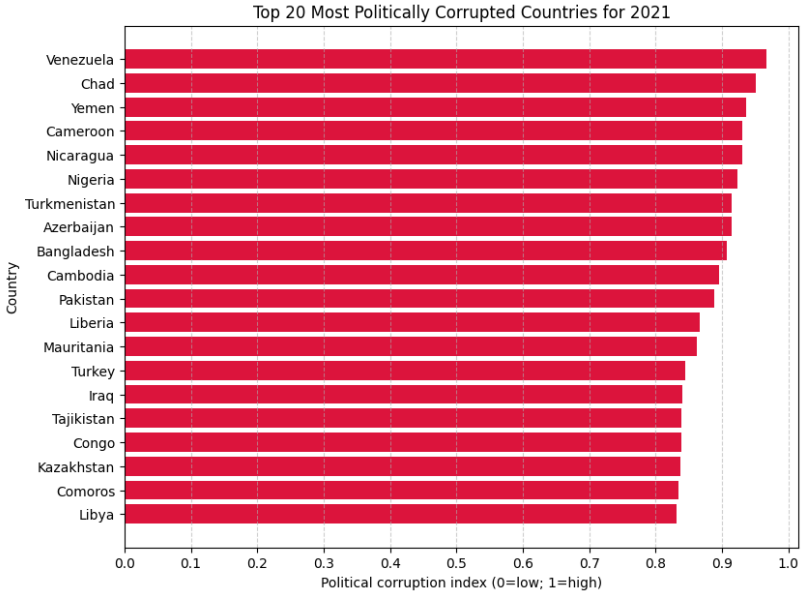

Figure 14: *Top 20 Most Politically Corrupted Countries for 2021*

3. Find whether each country with low political corruption index have high Cantril ladder score?

For defining low political corruption index, I use the 25th percentile of the column 'Political corruption index'. This will be my upper boundary for the searched countries. The exact value is 0.172. With that consideration the searched number of countries is 37 and they are:

Australia, Austria, Belgium, Botswana, Canada, Chile, Czechia, Denmark, Estonia, Finland, France, Georgia, Germany, Hong Kong, Iceland, Ireland, Israel, Italy, Jamaica, Japan, Latvia, Lithuania, Luxembourg, Netherlands, New Zealand, Norway, Poland, Portugal, Singapore, South Korea, Spain, Sweden, Switzerland, United Arab Emirates, United Kingdom, United States, Uruguay

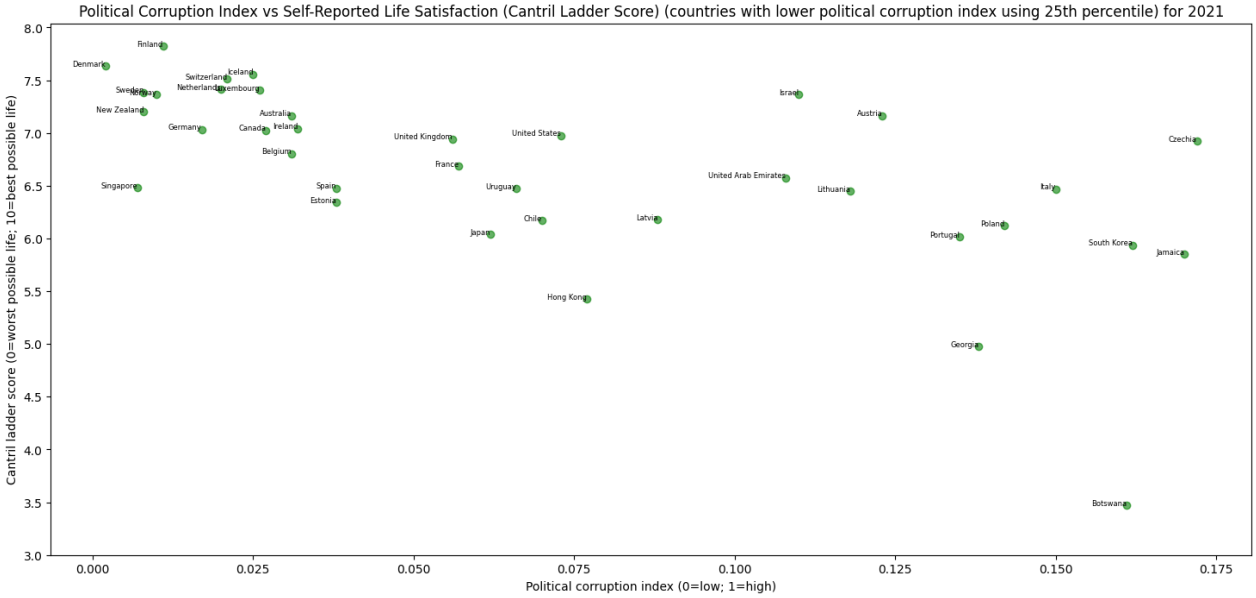

Figure 15: *Political Corruption Index vs Self-Reported Life Satisfaction (Cantril Ladder Score) (countries with lower political corruption index using 25th percentile) for 2021*

For defining higher Cantril ladder score (better possible life), I use the 75th percentile of the column 'Cantril ladder score'. This will be my lower boundary for high life satisfaction. In addition to that, I find the mean Cantril ladder score and the 25th percentile representing the upper boundary for low life satisfaction. I will provide the answer to the question by plotting and observing the following scatter plot:

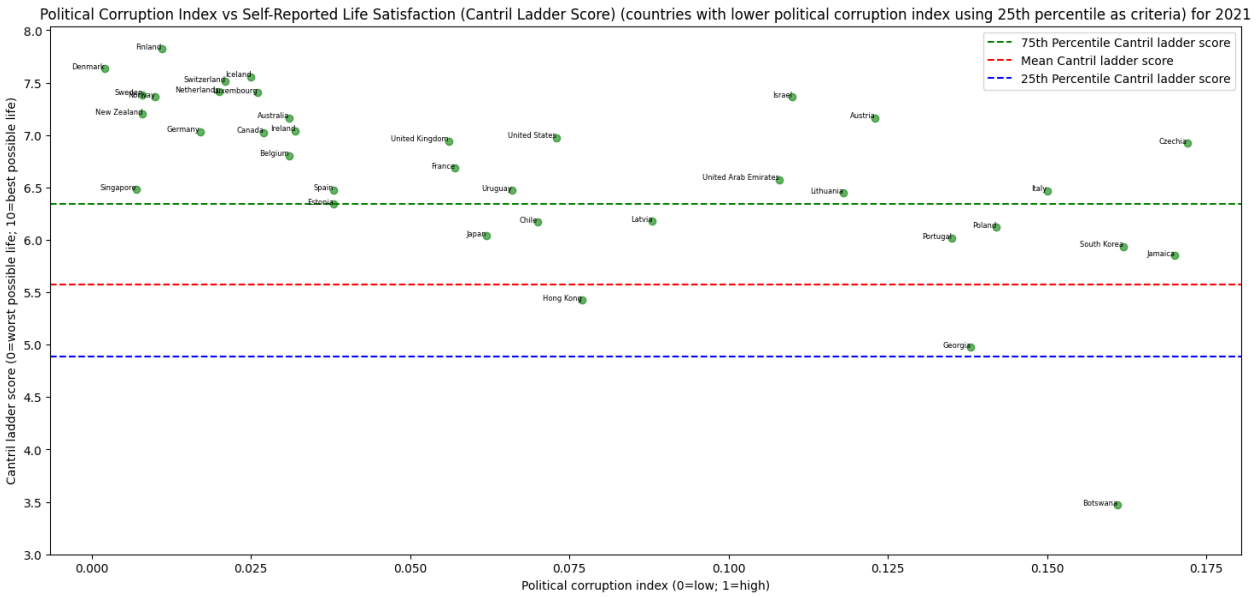

Figure 16: *Political Corruption Index vs Self-Reported Life Satisfaction (Cantril Ladder Score) (countries with lower political corruption index using 25th percentile) for 2021*

We can see that there are 3 countries below the mean Cantril ladder score that are with low index of political corruption. Most of the corrupted-free countries are happy which strengthens the trend.

Botswana is the only African country that is with low level of political corruption but is still with low level of life satisfaction. That must be related to the poor quality of life of the local population. Generally, we can conclude that the lower the political corruption is, the higher level of life satisfaction a country will have (most countries on the above scatter plot prove that).

4. Check if there are highly politically corrupted countries that have high Cantril ladder score?

For defining high political corruption index, I use the 75th percentile of the column 'Political corruption index'. This will be my lower boundary for the searched countries. The exact value is 0.735. With that consideration the searched number of countries is 37 and they are:

Azerbaijan, Bangladesh, Bosnia and Herzegovina, Cambodia, Cameroon, Chad, Comoros, Congo, Gabon, Guatemala, Honduras, Indonesia, Iran, Iraq, Kazakhstan, Kyrgyzstan, Laos, Lebanon, Liberia, Libya, Madagascar, Mali, Mauritania, Mongolia, Nicaragua, Nigeria, Pakistan, Paraguay, Tajikistan, Togo, Turkey, Turkmenistan, Uganda, Uzbekistan, Venezuela, Yemen, Zimbabwe

For defining higher Cantril ladder score (better possible life), I use the 75th percentile of the column 'Cantril ladder score'. This will be my lower boundary for high life satisfaction. In addition to that, I find the mean Cantril ladder score and the 25th percentile representing the upper boundary for low life satisfaction. I will provide the answer to the question by plotting and observing the following scatter plot:

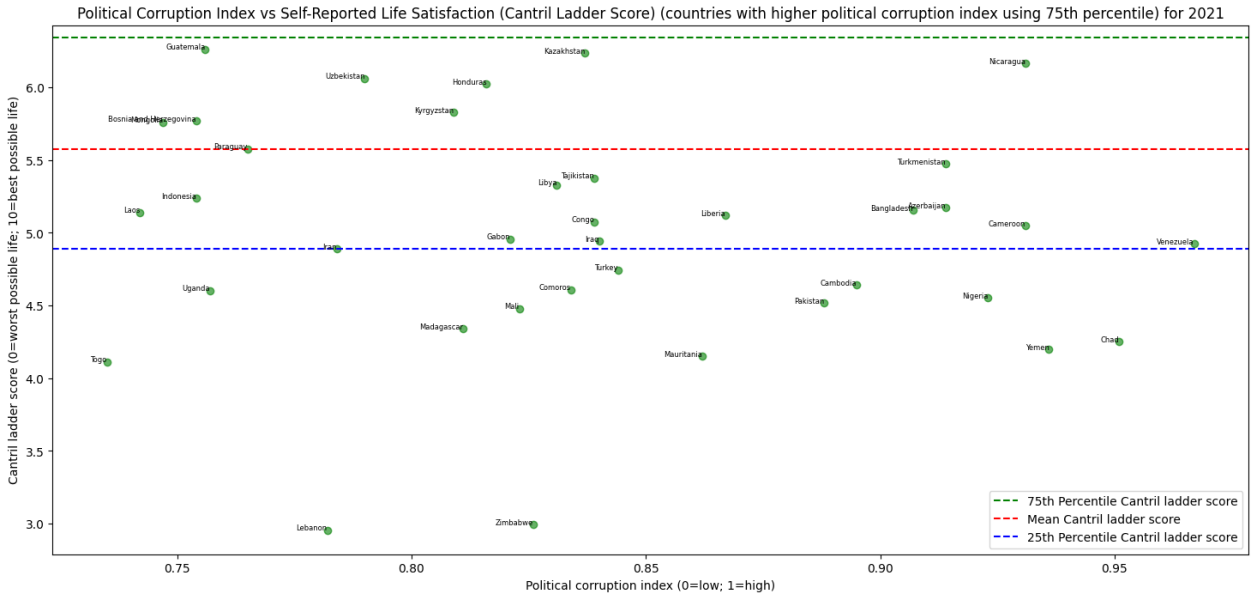

Figure 17: *Political Corruption Index vs Self-Reported Life Satisfaction (Cantril Ladder Score) (countries with higher political corruption index using 75th percentile) for 2021*

We can see that the plenty of the most politically corrupted countries tend to be unhappy (with low level of life satisfaction). There are still 9 countries which are above the mean Cantril ladder score although being highly corrupted. Most of these countries are poor countries and as such it is possible that they can be easily manipulated to think they are happy although living in a corrupted country. As a conclusion, we have to fight for eradicating political corruption if we want to live in a better, fairer world.

III. A grouped bar plot of Life Satisfaction (Cantril Ladder Score) and Political Corruption Index by Continents for 2021

We have already extracted the necessary data for the continents. As we have to compare two indices for the same entities (the six populated continents) I decided to use a grouped bar plot. For those purposes, it is important that the scale is the same for the two features – Cantril ladder score and political corruption index. For that reason, I transformed the political corruption index from boundaries

0-1 to 0-10 by multiplying the values for all the continents by 10. In this way, I can ensure equal index scale.

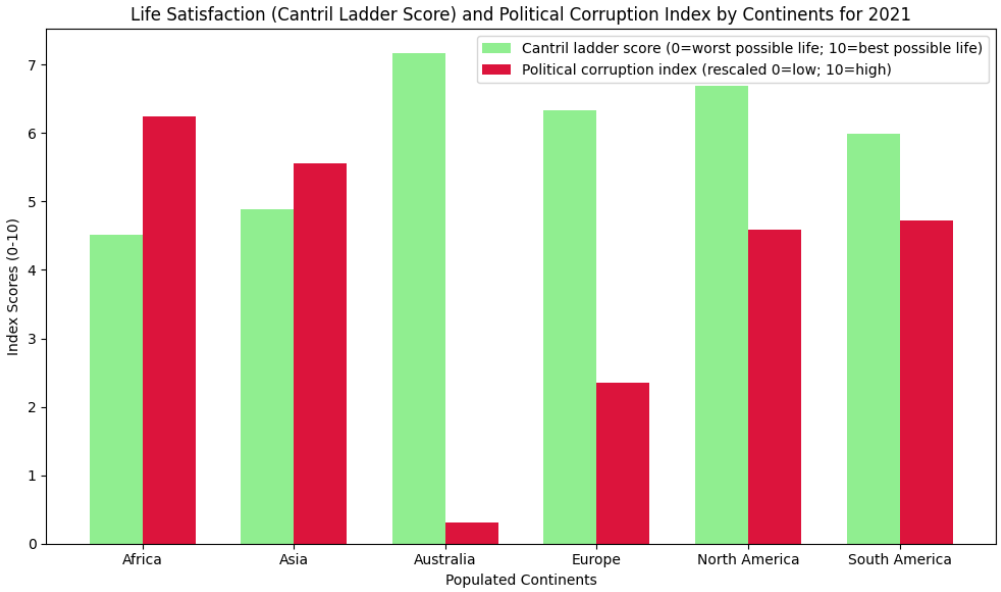

Figure 18: *Life Satisfaction (Cantril Ladder Score) and Political Corruption Index by Continents for 2021*

From the grouped bar plot we can see that Australia is the continent with highest Cantril ladder score, Africa – with the lowest one. Africa is the continent with highest index of political corruption whereas Australia is the continent with the lowest. Therefore, Australia seems to be the winner among the other continents. Europe is the second-best continents for living – with high Cantril ladder score and comparatively low level of political corruption.

Question 2*: In this question I had to investigate whether clustering can be used to identify volcanoes that lie on the same tectonic plate boundary.*

In this task I have to investigate whether clustering can be used to identify volcanoes that lie on the same tectonic plate boundary. For that reason, the first thing I have to do is to make a ***SPARQL*** query in ***Wikidata Query Service*** so as to find the latitude and the longitude of all volcanoes around the world.

```
SELECT ?volcano ?volcanoLabel (SAMPLE(?coordinate) AS ?singleCoordinate) (SAMPLE(?latitude) AS
?singleLatitude) (SAMPLE(?longitude) AS ?singleLongitude) (SAMPLE(?countryLabel) AS
?singleCountryLabel) (SAMPLE(?elevation) AS ?singleElevation)
WHERE {
  ?volcano wdt:P31/wdt:P279* wd:Q8072 .  # instance of volcano
  ?volcano wdt:P625 ?coordinate . # coordinate location
  ?volcano wdt:P2044 ?elevation . # elevation above sea level
  ?volcano wdt:P17 ?country . # country
  ?country rdfs:label ?countryLabel .  # country label
  FILTER(LANG(?countryLabel) = "en")
  SERVICE wikibase:label { bd:serviceParam wikibase:language "[AUTO_LANGUAGE],en". }
  BIND(geof:latitude(?coordinate) AS ?latitude).
  BIND(geof:longitude(?coordinate) AS ?longitude).
}
GROUP BY ?volcano ?volcanoLabel
```

The query finds all instances of ‘volcano’ and its subclasses (since some volcanoes are ‘stratovolcano’ or ‘active volcano’). Then with suitable predicates I get the coordinate, the elevation and the country which the volcano lies in. With ‘volcanoLabel’ and ‘countryLabel’ I get the human-readable name of the item. There are situations when for a specific volcano there are two or more pairs of coordinates or two or more countries. In that case, I get an arbitrary value from the results as these fields are not of much

importance to our task. I decided to investigate only those volcanoes that contain all the searched data. As a result, the output of the ***SPARQL*** query contains 1655 results. If 'elevation' and 'country' were optional, I would get 2764 results which is a big number of records for processing.

After having written the ***SPARQL*** query, the next step is to execute it in ***Python*** with the help of the ***request Python*** package and the ***Wikidata API***. I processed the response and transformed the data into ***Pandas*** data frame which is easier to manipulate. After that, I was ready to continue with the investigation task related to clustering algorithms. On the following scatter plot I've done with the help of ***Pandas***, we can see the distribution of the data. The aim of this diagram is to be used only for an overall picture.

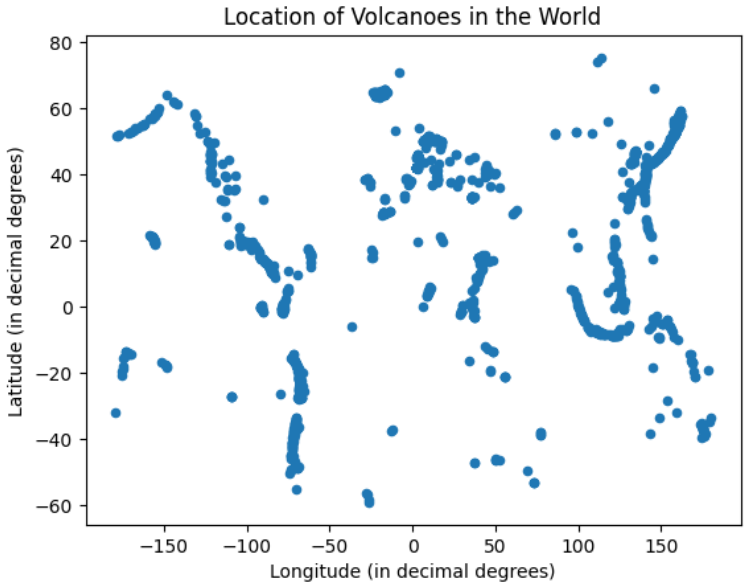

Figure 19: *Location of Volcanoes in the World*

In order to validate the future results, I decided to find a data set that contains the geographic coordinates of all tectonic plate boundaries. I downloaded it from the ***Kaggle*** website:

Tectonic Plate Boundaries

This data contains latitude and longitude data that completely encloses 56 tectonic plates. Further information on these plates can be found in the corresponding Wikipedia article.

Data URL: https://www.kaggle.com/datasets/cwthompson/tectonic-plate-boundaries/data

Data sources: *Bird (2003); Argus et al. (2011) for the*

The data is stored in a *csv file*. The name of the separate columns can be found in the *Jupyter notebook* accompanying this report. The *csv file* contains 12 321 entries. I use only the 'lat' and 'lon' columns for the visualisations.

Another additional improvement of the diagrams is the world map which I decided to put as a background. That would help me imagine the location of the volcanoes better. I realised this with the help of the ***Python GeoPandas*** package for working with geospatial data. I downloaded the map as a ***shp*** file from the ***Natural Earth*** website:

1:110m Cultural Vectors: Admin 0 - Countries

Data source: *Natural Earth*

https://www.naturalearthdata.com/downloads/110m-cultural-vectors/110m-admin-0-countries/

With all that data (world map, tectonic plate boundaries, volcano locations) I constructed the following scatter plot:

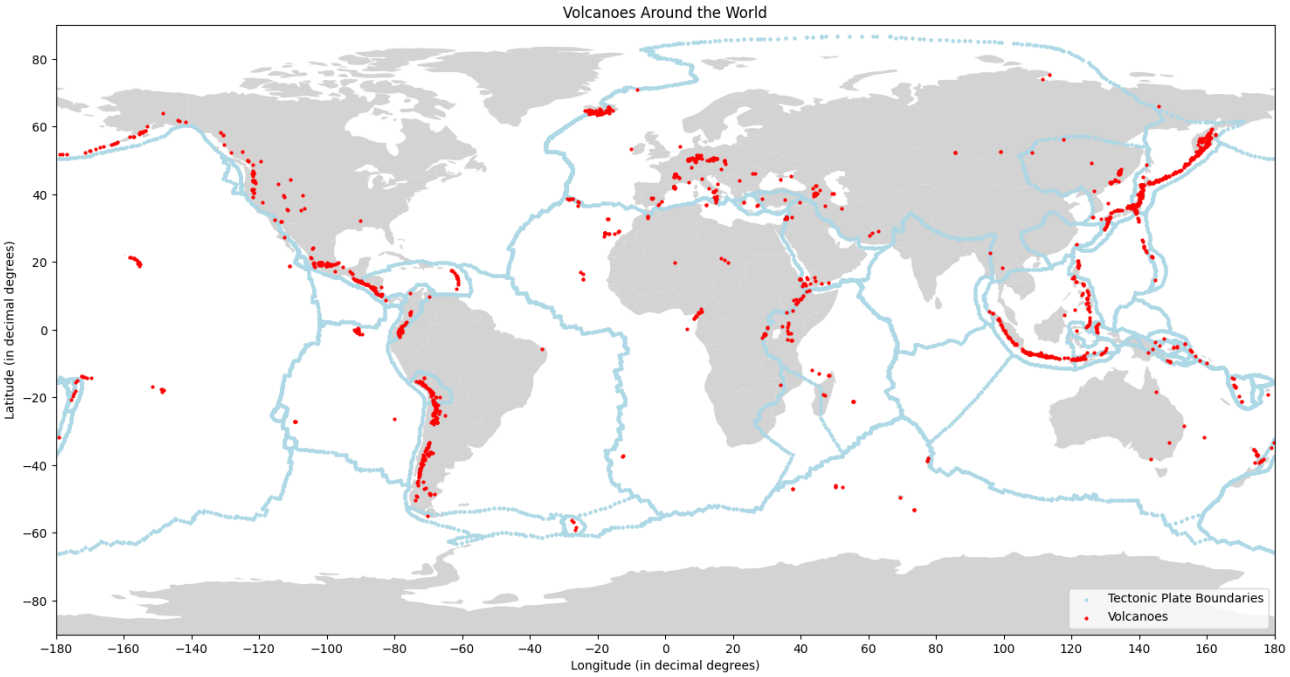

Figure 20: *Volcanoes Around the World*

As we can see from the above diagram, volcanoes are most often found where tectonic plates are diverging or converging, and because most of Earth's plate boundaries are underwater, most volcanoes are found underwater. Now, we have to check whether clustering can be used to identify volcanoes that lie on the same tectonic plate boundary. I will do that with the help of two clustering algorithms (using the ***Scikit-learn*** library) – **DBSCAN** and **K-means**. First, I will apply the **DBSCAN** algorithm. To do so, some parameters have to be assigned. For the minimum number of samples in the neighbourhood for a point to be considered as a core point I chose min_samples = 3 and for the maximum distance between two samples belonging to the same neighbourhood – eps = 6. The taken decisions are in conformity with the specifics of the problem. The number of formed clusters in that case is 37 with 33 outliers.

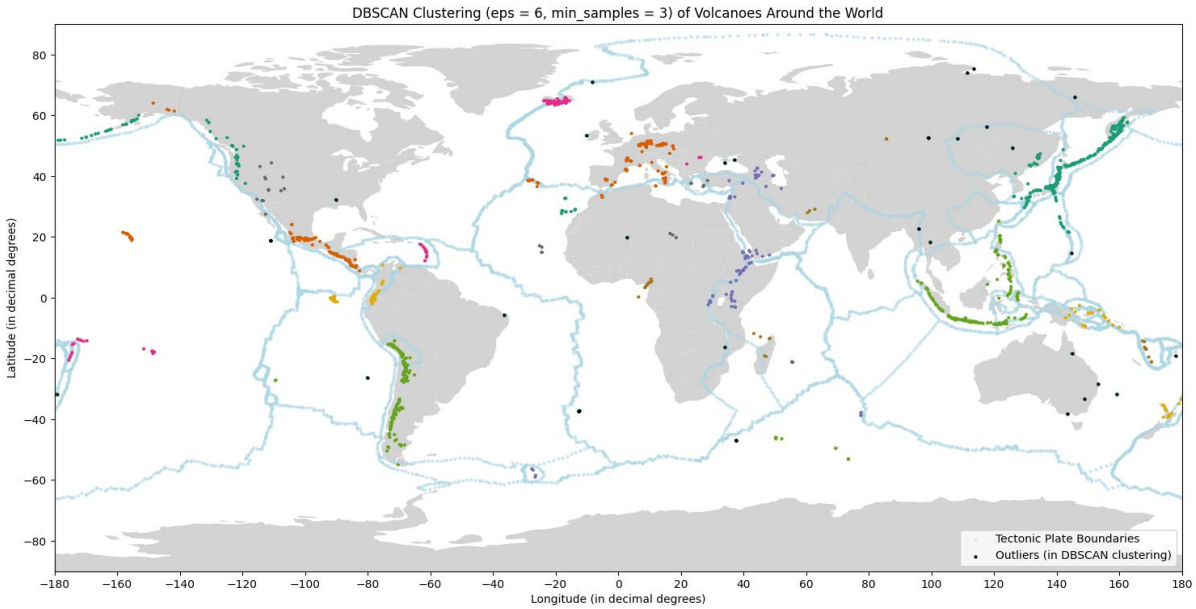

Figure 21: *DBSCAN Clustering (eps = 6, min_samples = 3) of Volcanoes Around the World*

As we know, **DBSCAN** can classify some points as outliers. This is the case for volcanoes that are alone or only two nearby. Usually, they are not near the tectonic plate boundaries. They can be underwater as hotspot volcanoes as well. From the map we see that in most of the cases the clustering algorithm **manages to identify volcanoes** that lie on the same tectonic plate boundary. This can be seen from the fact that the coordinate points are in one colour (forming a cluster). These clusters are denser near the tectonic plate boundaries which is what theory says when it comes to the conditions for forming a volcano.

If we perform **DBSCAN** for the volcano points without loading the world map and tectonic plate boundaries we get the same results. However, we cannot validate them.

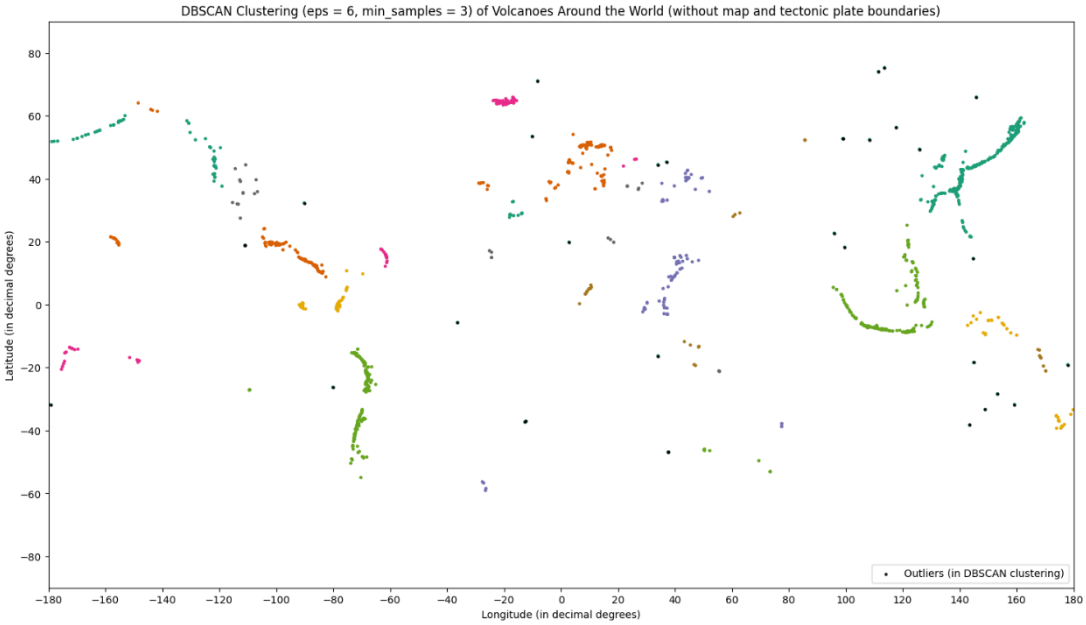

Figure 22: *DBSCAN Clustering (eps = 6, min_samples = 3) of Volcanoes Around the World (without map and tectonic plate boundaries)*

Another clustering algorithm that we can use is **K-means**. We apply it for k = 9. The results are satisfactory as well.

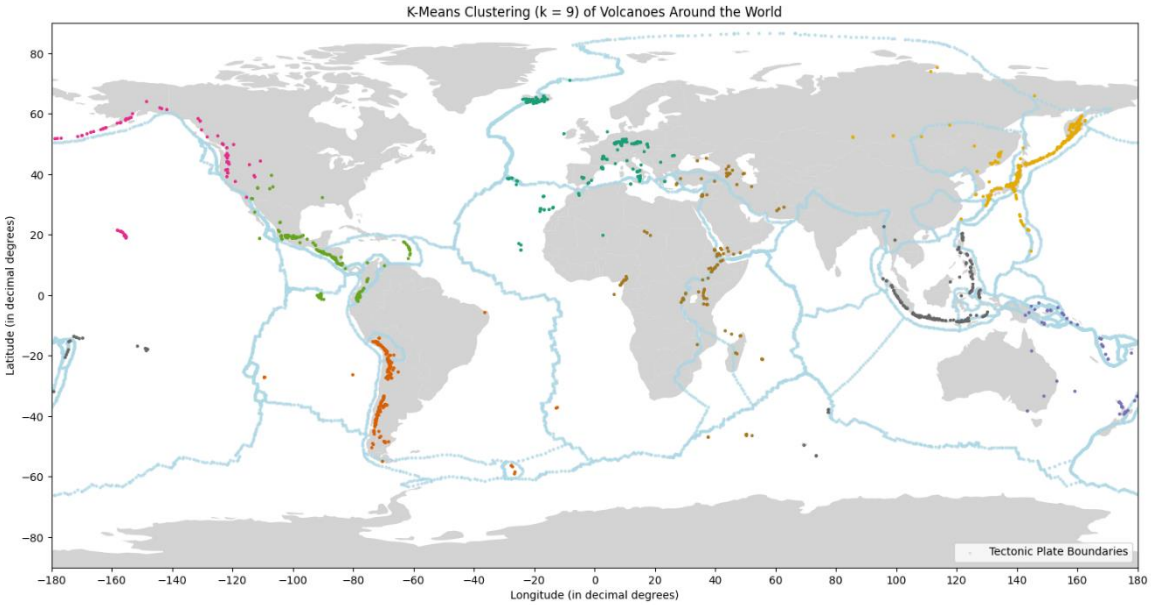

Figure 23: *K-Means Clustering (k = 9) of Volcanoes Around the World*

With that algorithm, we do not have outliers. All volcanoes are classified to a cluster. Nevertheless, the results are still enough to answer positively to the task question. Again, without loading the world map and tectonic plate boundaries we get the following scatter plot:

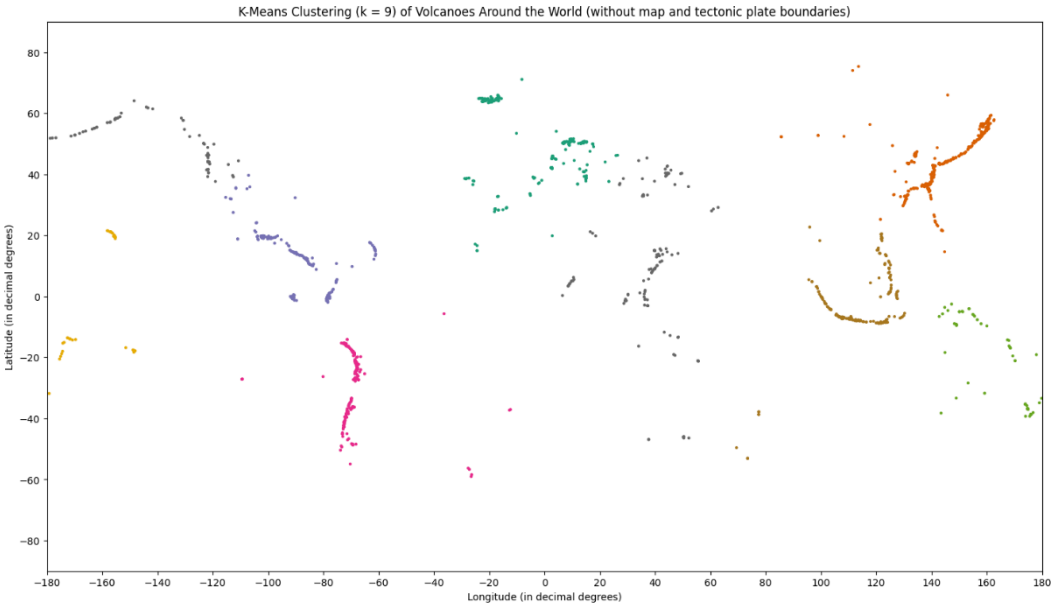

Figure 24: *K-Means Clustering (k = 9) of Volcanoes Around the World (without map and tectonic plate boundaries)*

We can notice a case that we had the same in **Assignment 3** as well. Since, the Earth is a sphere the points next to the left border and the points to the right border are close to each other. This means that it would be logically correct if they are part of the same cluster. I achieved that in **Assignment 3** with the help of polar coordinates using the trigonometric functions sine and cosine. The same technique can be applied here.