Documentation - exercise 2
dataset : our world in data - covid dataset

Professor : Dr. Kheradpisheh
Teacher Assistant : MohammadReza Khanmohammadi

By: Katayoun Kobraei

1. Introduction to Dataset

Our world in data covid 19 dataset is a complete COVID-19 dataset is a collection of the COVID-19 data and provided by Our World in Data cite. Our World in Data team updates it daily throughout the duration of the COVID-19 pandemic.

These are the following information that includes in the dataset:

- **iso_code:** ISO 3166-1 alpha-3 – three-letter country codes
- **continent:** Continent of the geographical location
- **location:** Geographical location
- **date:** Date of observation
- **total_cases:** Total confirmed cases of COVID-19
- **new_cases:** New confirmed cases of COVID-19
- **new_cases_smoothed:** New confirmed cases of COVID-19 (7-day smoothed)
- **total_cases_per_million:** Total confirmed cases of COVID-19 per 1,000,000 people
- **new_cases_per_million:** New confirmed cases of COVID-19 per 1,000,000 people
- **new_cases_smoothed_per_million:** New confirmed cases of COVID-19 (7-day smoothed) per 1,000,000 people
- **total_deaths:** Total deaths attributed to COVID-19
- **new_deaths:** New deaths attributed to COVID-19
- **new_deaths_smoothed:** New deaths attributed to COVID-19 (7-day smoothed)
- **total_deaths_per_million:** Total deaths attributed to COVID-19 per 1,000,000 people
- **new_deaths_per_million:** New deaths attributed to COVID-19 per 1,000,000 people
- **new_deaths_smoothed_per_million:** New deaths attributed to COVID-19 (7-day smoothed) per 1,000,000 people
- **excess_mortality:** Percentage difference between the reported number of weekly or monthly deaths in 2020–2021 and the projected number of deaths for the same period based on previous years.
- **excess_mortality_cumulative:** Percentage difference between the cumulative number of deaths since 1 January 2020 and the cumulative projected deaths for the same period based on previous years.

- **excess_mortality_cumulative_absolute:** Cumulative difference between the reported number of deaths since 1 January 2020 and the projected number of deaths for the same period based on previous years.
- **excess_mortality_cumulative_per_million:** Cumulative difference between the reported number of deaths since 1 January 2020 and the projected number of deaths for the same period based on previous years, per million people.
- **icu_patients:** Number of COVID-19 patients in intensive care units (ICUs) on a given day
- **icu_patients_per_million:** Number of COVID-19 patients in intensive care units (ICUs) on a given day per 1,000,000 people
- **hosp_patients:** Number of COVID-19 patients in the hospital on a given day
- hosp_patients_per_million: Number of COVID-19 patients in hospital on a given day per 1,000,000 people
- **weekly_icu_admissions:** Number of COVID-19 patients newly admitted to intensive care units (ICUs) in a given week
- **weekly_icu_admissions_per_million:** Number of COVID-19 patients newly admitted to intensive care units (ICUs) in a given week per 1,000,000 people
- **weekly_hosp_admissions:** Number of COVID-19 patients newly admitted to hospitals in a given week
- **weekly_hosp_admissions_per_million:** Number of COVID-19 patients newly admitted to hospitals in a given week per 1,000,000 people
- **stringency_index:** Government Response Stringency Index: composite measure based on 9 response indicators including school closures, workplace closures, and travel bans, rescaled to a value from 0 to 100 (100 = strictest response)
- **reproduction_rate:** Real-time estimate of the effective reproduction rate (R) of COVID-19.
- **total_tests:** Total tests for COVID-19
- **new_tests:** New tests for COVID-19 (only calculated for consecutive days)
- **total_tests_per_thousand:** Total tests for COVID-19 per 1,000 people
- **new_tests_per_thousand:** New tests for COVID-19 per 1,000 people
- **new_tests_smoothed:** New tests for COVID-19 (7-day smoothed). For countries that don't report testing data on a daily basis, we assume that testing changed equally on a daily basis over any periods in which no data was reported. This produces a complete series of daily figures, which is then averaged over a rolling 7-day window

- **new_tests_smoothed_per_thousand:** New tests for COVID-19 (7-day smoothed) per 1,000 people
- **positive_rate:** The share of COVID-19 tests that are positive, given as a rolling 7-day average (this is the inverse of tests_per_case)
- **tests_per_case:** Tests conducted per new confirmed case of COVID-19, given as a rolling 7-day average (this is the inverse of positive_rate)
- **tests_units:** Units used by the location to report its testing data
- **total_vaccinations:** Total number of COVID-19 vaccination doses administered
- **people_vaccinated:** Total number of people who received at least one vaccine dose
- **people_fully_vaccinated:** Total number of people who received all doses prescribed by the vaccination protocol
- **total_boosters:** Total number of COVID-19 vaccination booster doses administered (doses administered beyond the number prescribed by the vaccination protocol)
- **new_vaccinations:** New COVID-19 vaccination doses administered (only calculated for consecutive days)
- **new_vaccinations_smoothed:** New COVID-19 vaccination doses administered (7-day smoothed). For countries that don't report vaccination data on a daily basis, we assume that vaccination changed equally on a daily basis over any periods in which no data was reported. This produces a complete series of daily figures, which is then averaged over a rolling 7-day window
- **total_vaccinations_per_hundred:** Total number of COVID-19 vaccination doses administered per 100 people in the total population
- **people_vaccinated_per_hundred:** Total number of people who received at least one vaccine dose per 100 people in the total population
- **people_fully_vaccinated_per_hundred:** Total number of people who received all doses prescribed by the vaccination protocol per 100 people in the total population
- **total_boosters_per_hundred:** Total number of COVID-19 vaccination booster doses administered per 100 people in the total population
- **new_vaccinations_smoothed_per_million:** New COVID-19 vaccination doses administered (7-day smoothed) per 1,000,000 people in the total population
- new_people_vaccinated_smoothed: Daily number of people receiving their first vaccine dose (7-day smoothed)
- **new_people_vaccinated_smoothed_per_hundred:** Daily number of people receiving their first vaccine dose (7-day smoothed) per 100 people in the total population
- **population:** latest available values of population

- **population_density:** Number of people divided by land area, measured in square kilometers, most recent year available
- **median_age:** Median age of the population, UN projection for 2020
- **aged_65_older:** Share of the population that is 65 years and older, most recent year available
- **aged_70_older:** Share of the population that is 70 years and older in 2015
- **gdp_per_capita:** Gross domestic product at purchasing power parity (constant 2011 international dollars), most recent year available
- **extreme_poverty:** Share of the population living in extreme poverty, most recent year available since 2010
- **cardiovasc_death_rate:** Death rate from cardiovascular disease in 2017 (annual number of deaths per 100,000 people)
- **diabetes_prevalence:** Diabetes prevalence (% of the population aged 20 to 79) in 2017
- **female_smokers:** Share of women who smoke, most recent year available
- **male_smokers:** Share of men who smoke, most recent year available
- **handwashing_facilities:** Share of the population with basic hand washing facilities on-premises, most recent year available
- **hospital_beds_per_thousand:** Hospital beds per 1,000 people, most recent year available since 2010
- **life_expectancy:** Life expectancy at birth in 2019
- **human_development_index:** A composite index measuring average achievement in three basic dimensions of human development—a long and healthy life, knowledge, and a decent standard of living.

It can be seen that some of these are useless. Therefore, we decided to drop following features:

- 'total_cases_per_million', 'new_cases_per_million', 'new_cases_smoothed_per_million', 'total_deaths_per_million', 'new_deaths_per_million', 'new_deaths_smoothed_per_million', 'icu_patients_per_million', 'hosp_patients_per_million', 'new_vaccinations_smoothed', 'total_vaccinations_per_hundred', 'people_vaccinated_per_hundred', 'people_fully_vaccinated_per_hundred', 'total_boosters_per_hundred', 'new_vaccinations_smoothed_per_million', 'new_people_vaccinated_smoothed', 'new_people_vaccinated_smoothed_per_hundred', 'hosp_patients_per_million', 'excess_mortality_cumulative_per_million', 'total_tests_per_thousand', 'new_tests_per_thousand', 'new_tests_smoothed_per_thousand' : I found that these features pretty much repeat information that's already covered by other features in our dataset. I've made the call to drop these specific features to avoid redundancy. This not only makes our dataset simpler but also ensures we hold onto the key info we need for our analysis.

2. Abstract

In this assignment, we're diving into the vast world of data to tackle the impact of the global COVID-19 pandemic. Over the last three years, we've all been dealing with the ripple effects of the virus, highlighting the need for sharp analysis. The main goal here is to dig deep into the data, finding patterns and trends to reach key insights that help us navigate these tough times.

This project has two main parts: First, cleaning the data, where we remove outliers and null values to make sure our dataset is accurate. After, we shift into exploratory data analysis (EDA), bringing out the artistic side in creating intricate visualizations that go beyond the usual charts. The grades will reflect how well we clean the data, the creativity in our visuals, and the depth of knowledge we draw from the data.

3. Data Cleaning

3.1.Handling null values

3.1.1. Handling in categorical features:

The only feature where null values exist in it is continent. We could find the continent feature of each data from its location which is its country. We decide to deal with null values using ‘‘pycountry’’ library, which country is given and continent is returned.
In this part we realized some wrong values of countries that are not actually a country like World, Upper middle income, High income, Low income and Upper middle income. So we dropped all these data from the dataset which was almost 7000 rows. Some other wrong values were Africa, Asia, Europe, North America, South America and Oceania that we decided to use the exact name for their continent because they are actually a continent name.

3.1.2. Handling in numerical features:

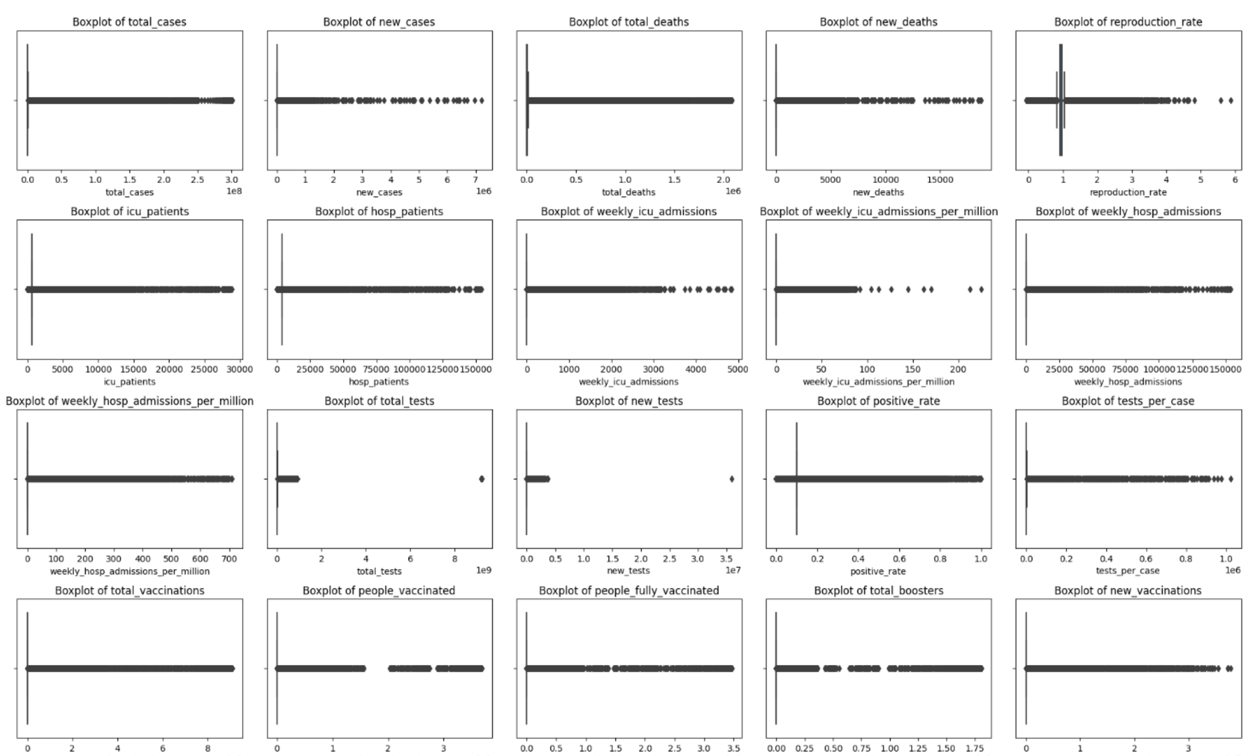
Because of the nature of numerical features we could not pick only one strategy to handle null values. For total_cases, new_cases, new_cases_smoothed, total_deaths, new_deaths and new_deaths_smoothed it was better to fill null values with forward fill or backward fill technique because the total and new cases or deaths are cumulative values, and they are likely to follow a temporal trend. Forward filling or backward filling missing values based on the order of the dates can capture the temporal progression of the disease.
The stringency index is a measure of the strictness of government policies. These policies are typically consistent for a certain period. Therefore, forward filling or backward filling can be appropriate.
For the weekly_icu_admissions and Related Features we assume that missing values in ICU and hospital admissions mean no admissions occurred during those weeks so we use 0 instead of each null value.
We use Forward fill for total_tests filling with 0 for new_tests. because total_tests is likely cumulative, and new_tests can be 0 if no new tests are reported.
For positive_rate and tests_per_case we use the mean to maintain the central tendency of the data.
For all vaccination features we assume 0 vaccinations for the initial missing values (worst case).
Reproduction Rate, ICU Patients, Hosp Patients are continuous and may not follow a strict temporal trend. We decide to Impute missing values with the mean or median helps to maintain the central tendency of the data without introducing drastic changes.

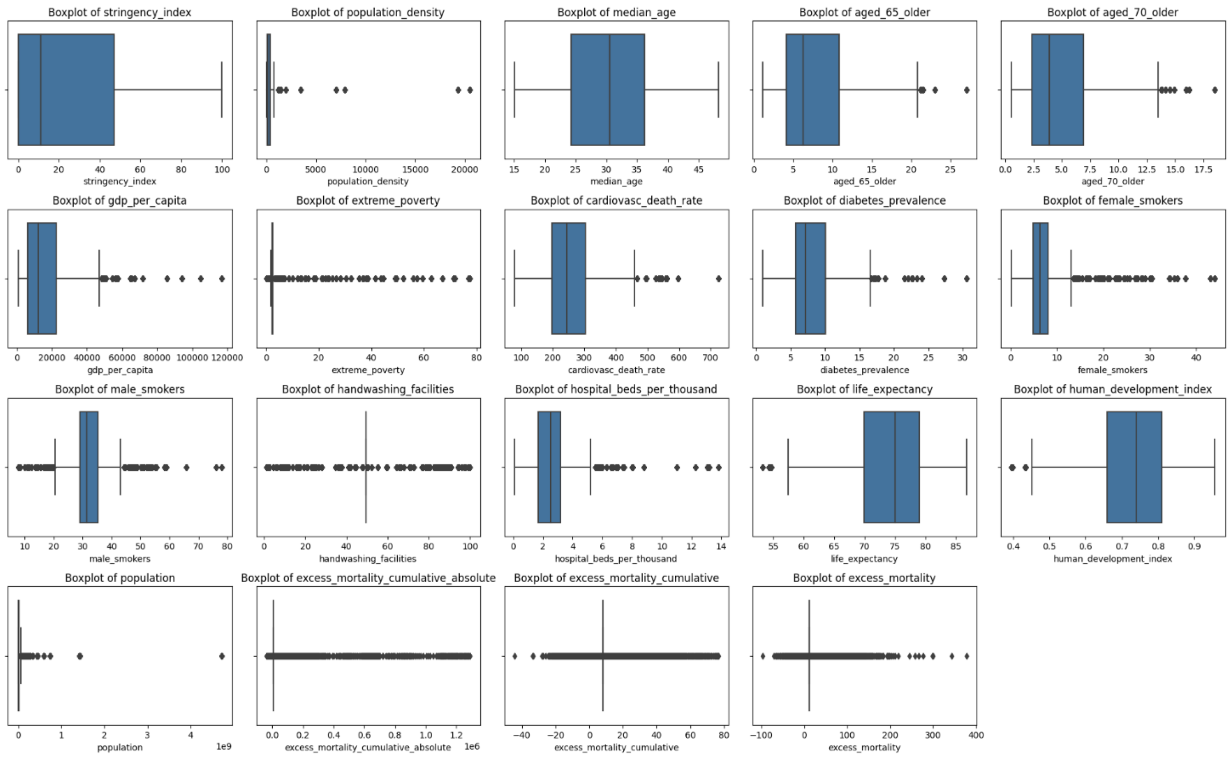
All Demographic and health metrics like Population Density, Median Age are often stable over time and might not have strong temporal patterns. We Imputed missing values with the mean or median to maintain the central tendency.
Excess mortality metrics may not follow a strict temporal trend and could be influenced by various factors. Imputing missing values with the mean or median can be a reasonable approach.
Testing and Vaccination Metrics may have a smooth trend over time. Interpolation or regression imputation methods can be used to estimate missing values based on the overall trend in the data.
And for all other features I preferred to use mean value technique for imputting.

3.2. Handling Outliers

We could detect outliers by identifying data points that deviate significantly from the majority of the data. There are some common methods for detecting outliers and we used so we explain below:

Visual Inspection: First and foremost we used visualization tools like box plots to visually represent the distribution of data. We can identify outliers using box plots as points beyond the "whiskers" of the box. We did these for all numerical features.





We see that there are outliers existing in features like human_dvelopment_index or gdp_per_capita. So we should clean all these features that seems to have outliers.

Statistical Methods: We use some statistical methods like z-score to find outliers in columns. The Z-score is a measure of how many standard deviations a data point is from the mean of a group of data. The basic idea behind Z-score-based outlier detection is that values with a Z-score greater than a certain threshold (we set it to 3) are considered outliers.

```
Outliers in total_cases: 81765
Outliers in new_cases: 81765
Outliers in total_deaths: 81765
Outliers in new_deaths: 81765
Outliers in reproduction_rate: 81765
Outliers in icu_patients: 81765
Outliers in hosp_patients: 81765
Outliers in weekly_icu_admissions: 81765
Outliers in weekly_icu_admissions_per_million: 81765
Outliers in weekly_hosp_admissions: 81765
Outliers in weekly_hosp_admissions_per_million: 81765
Outliers in total_tests: 81765
Outliers in new_tests: 81765
Outliers in positive_rate: 81765
Outliers in tests_per_case: 81765
Outliers in total_vaccinations: 81765
Outliers in people_vaccinated: 81765
Outliers in people_fully_vaccinated: 81765
Outliers in total_boosters: 81765
Outliers in new_vaccinations: 81765
Outliers in stringency_index: 81765
Outliers in population_density: 81765
Outliers in median_age: 81765
Outliers in aged_65_older: 81765
Outliers in aged_70_older: 81765
Outliers in gdp_per_capita: 81765
Outliers in extreme_poverty: 81765
Outliers in cardiovasc_death_rate: 81765
Outliers in diabetes_prevalence: 81765
Outliers in female_smokers: 81765
Outliers in male_smokers: 81765
Outliers in handwashing_facilities: 81765
Outliers in hospital_beds_per_thousand: 81765
Outliers in life_expectancy: 81765
Outliers in human_development_index: 81765
Outliers in population: 81765
Outliers in excess_mortality_cumulative_absolute: 81765
Outliers in excess_mortality_cumulative: 81765
Outliers in excess_mortality: 81765
rows with outliers:  3188835
```

Then we drop all outliers we found with this score and drop those rows from our dataset. Our dataset became 261375 * 45 after we dropped outliers.

3.3. Handling duplicate Entries

Duplicates can introduce inaccuracies in analysis by inflating the importance of certain observations and they create inconsistencies and confusion when analyzing or interpreting data. So removing duplicates sets the foundation for meaningful analysis and decision-making based on high-quality data. Finding duplicate entries in a DataFrame involves identifying rows that have identical values across all columns or a subset of columns. First we check duplicates and then we drop all of it from the dataset.

4. EDA (Exploratory data analysis)

4.1. Confirmed Cases Over Time

The following chart is used to display the total cases and total deaths for each country, with a gradient background indicating the magnitude of these values. Darker shades indicate higher values, providing a quick and intuitive way to understand the distribution of cases and deaths across countries.

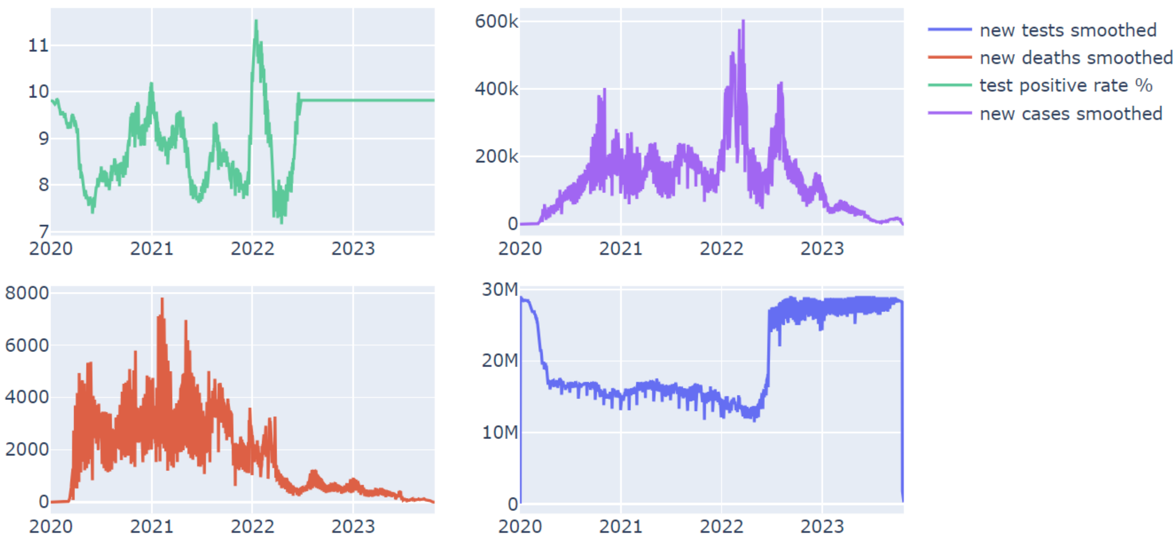
	location	total_cases	total_deaths
153	Russia	14147660837.000000	274449471.000000
192	Turkey	11718730533.000000	78227120.000000
93	Italy	11506991527.000000	105648192.000000
198	United Kingdom	9956970216.000000	98471546.000000
68	France	9383995155.000000	45119074.000000
137	Oceania	7171969943.000000	14349462.000000
205	Vietnam	6500944596.000000	27833577.000000
7	Argentina	6441724621.000000	91361486.000000
177	Spain	6361358530.000000	63263626.000000
73	Germany	6116114985.000000	36669819.000000
88	Iran	5406742384.000000	112892151.000000
10	Australia	5203333814.000000	9699844.000000
38	Colombia	4602497158.000000	107504173.000000
125	Netherlands	4405339067.000000	15546205.000000
116	Mexico	4108559330.000000	214862019.000000
148	Poland	3821185835.000000	75511378.000000
196	Ukraine	3734606743.000000	77436744.000000
149	Portugal	3473545635.000000	21991162.000000
11	Austria	3274708770.000000	15853711.000000
32	Canada	3264888779.000000	41286840.000000
145	Peru	3195294019.000000	196910973.000000
146	Philippines	3012209855.000000	49086040.000000
174	South Africa	2877126602.000000	76581710.000000

17	Belgium	2839771905.000000	26598452.000000
109	Malaysia	2823575262.000000	21552828.000000
92	Israel	2796258134.000000	8360460.000000
187	Thailand	2637120371.000000	19573261.000000
182	Switzerland	2509158691.000000	11186186.000000
37	Chile	2507836656.000000	33278299.000000
48	Czechia	2311144290.000000	23485540.000000
89	Iraq	2098629164.000000	24132188.000000
152	Romania	2087362465.000000	46788956.000000
181	Sweden	1950356703.000000	19602347.000000
49	Denmark	1681205711.000000	4664811.000000
15	Bangladesh	1631056854.000000	24520140.000000
86	Hungary	1558477282.000000	38997429.000000
26	Brazil	1486477556.000000	42000442.000000
139	Pakistan	1429064922.000000	29063970.000000
96	Jordan	1327525854.000000	12102156.000000
72	Georgia	1213140058.000000	12374583.000000
97	Kazakhstan	1113406903.000000	15218038.000000
126	New Zealand	1095165327.000000	1334705.000000
170	Slovakia	1081801659.000000	14306905.000000
121	Morocco	1061715778.000000	14742162.000000
90	Ireland	1032362671.000000	6781237.000000
104	Lebanon	963623743.000000	9510211.000000
136	Norway	862432538.000000	2939878.000000
45	Cuba	840037262.000000	6559786.000000
28	Bulgaria	829347672.000000	26871033.000000
124	Nepal	823834490.000000	10154800.000000
197	United Arab Emirates	819176720.000000	2053996.000000
22	Bolivia	803923847.000000	19200380.000000
171	Slovenia	799258707.000000	6502989.000000
42	Costa Rica	797908651.000000	7060549.000000
44	Croatia	775966276.000000	11648105.000000
80	Guatemala	774200703.000000	14984841.000000
142	Panama	770237940.000000	8061222.000000
164	Saudi Arabia	749950471.000000	9630205.000000
67	Finland	738610482.000000	4642042.000000
108	Lithuania	722203115.000000	6191040.000000
201	Uruguay	696564582.000000	6016268.000000

150	Puerto Rico	685434832.000000	4316467.000000
119	Mongolia	649665866.000000	1478630.000000
12	Azerbaijan	637619538.000000	8024053.000000
53	Ecuador	623297332.000000	25433031.000000
103	Latvia	553971380.000000	4723506.000000
178	Sri Lanka	542507898.000000	12981350.000000
100	Kuwait	542098194.000000	2396988.000000
141	Palestine	540311542.000000	4737829.000000
52	Dominican Republic	529034869.000000	4389330.000000
144	Paraguay	520191252.000000	14858941.000000
14	Bahrain	504504988.000000	1373833.000000
122	Myanmar	501432452.000000	15809615.000000
204	Venezuela	463415454.000000	5025364.000000
118	Moldova	456180748.000000	9856012.000000
62	Europe	453358905.000000	30308535.000000
63	European Union	434360950.000000	26456636.000000
54	Egypt	432345519.000000	21921092.000000
106	Libya	421465099.000000	5599623.000000
61	Ethiopia	414058851.000000	6415445.000000
85	Honduras	407646441.000000	10198390.000000
47	Cyprus	391995712.000000	902509.000000
59	Estonia	365384056.000000	1909884.000000
8	Armenia	354738417.000000	7057842.000000
138	Oman	344301719.000000	4107505.000000
175	South America	330366772.000000	11546257.000000
24	Bosnia and Herzegovina	319620260.000000	13300358.000000
151	Reunion	295398327.000000	626630.000000
98	Kenya	290222345.000000	5045619.000000
132	North Macedonia	262015594.000000	8136243.000000
129	Nigeria	247553785.000000	3102805.000000
1	Algeria	242715206.000000	6418578.000000
25	Botswana	235131473.000000	2153388.000000
0	Albania	219200973.000000	2739436.000000
99	Kosovo	195581185.000000	2736768.000000
210	Zimbabwe	192663482.000000	4529621.000000
55	El Salvador	158225533.000000	3835195.000000
74	Ghana	155778793.000000	1376450.000000
195	Uganda	138461476.000000	2931788.000000

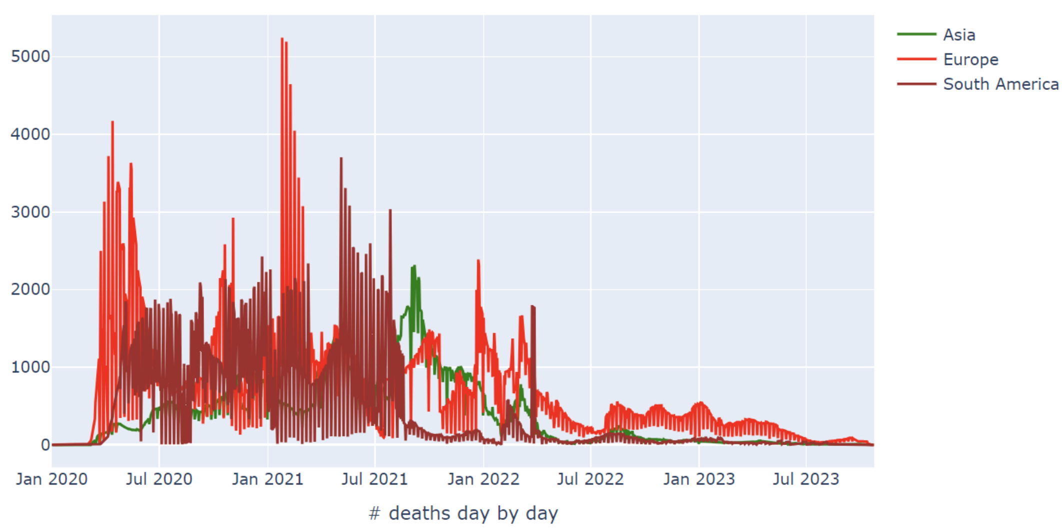
110	Maldives	137231055.000000	258356.000000
113	Martinique	136489796.000000	760265.000000
123	Namibia	135427080.000000	3239022.000000
102	Laos	128895975.000000	401549.000000
78	Guadeloupe	123734580.000000	763678.000000
87	Iceland	122957369.000000	106011.000000
191	Trinidad and Tobago	113146063.000000	2693882.000000
31	Cameroon	111925233.000000	1809597.000000
94	Jamaica	110406267.000000	2477342.000000
30	Cambodia	109532184.000000	2405539.000000
4	Angola	82208216.000000	1627290.000000
112	Malta	79103789.000000	650068.000000
166	Senegal	77633315.000000	1804844.000000
43	Cote d'Ivoire	76399327.000000	690470.000000
69	French Guiana	69931747.000000	329315.000000
101	Kyrgyzstan	68651497.000000	778867.000000
131	North America	63228260.000000	2336100.000000
16	Barbados	61228486.000000	353374.000000
60	Eswatini	58497128.000000	1228640.000000
180	Suriname	57843987.000000	1023830.000000
179	Sudan	57306327.000000	4314150.000000
83	Guyana	53114382.000000	1024337.000000
66	Fiji	51473128.000000	654834.000000
33	Cape Verde	50786016.000000	366013.000000
18	Belize	49447516.000000	586455.000000
114	Mauritania	47781861.000000	883236.000000
183	Syria	47519135.000000	2740376.000000
70	Gabon	42262131.000000	269318.000000
95	Jersey	40777067.000000	127440.000000
21	Bhutan	36318121.000000	13267.000000
82	Guinea	35698893.000000	394804.000000
167	Seychelles	35624383.000000	133066.000000
143	Papua New Guinea	35536236.000000	511133.000000
3	Andorra	33417730.000000	153095.000000

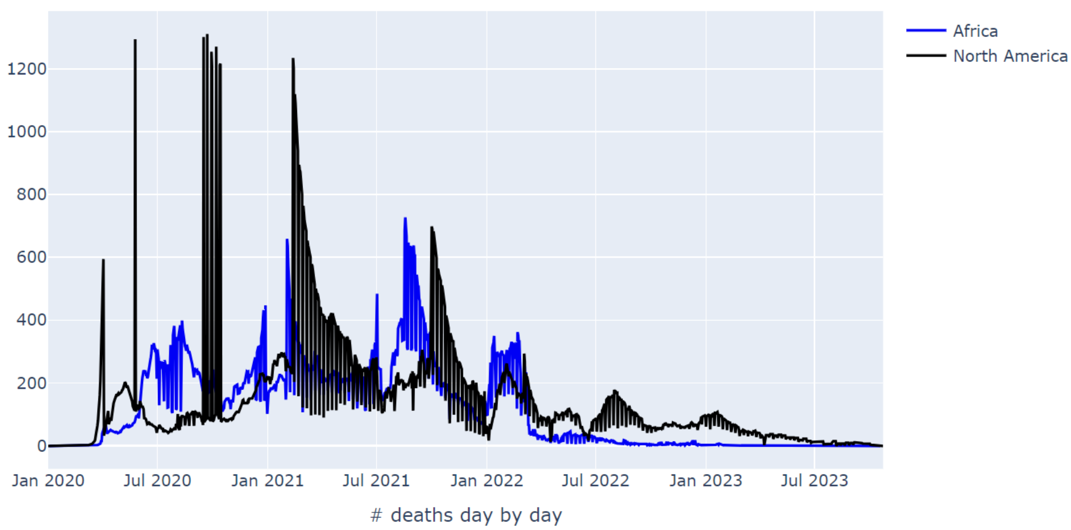
Then we use the Plotly library to create a subplot of four line charts representing different COVID-19-related metrics over time:



Each trace represents a line chart for a specific metric (new_tests_smoothed, new_deaths_smoothed, positive_rate, new_cases_smoothed) over time.
The x-axis is the date, and the y-axis is the corresponding metric.
We can see trends and rise and fall of these features over time. We see at first of the pandemic there were not many test positive rates but the number of deaths over time was increasing. Then vaccines came and although there was more test positive rate but new death was decreasing hopefully.

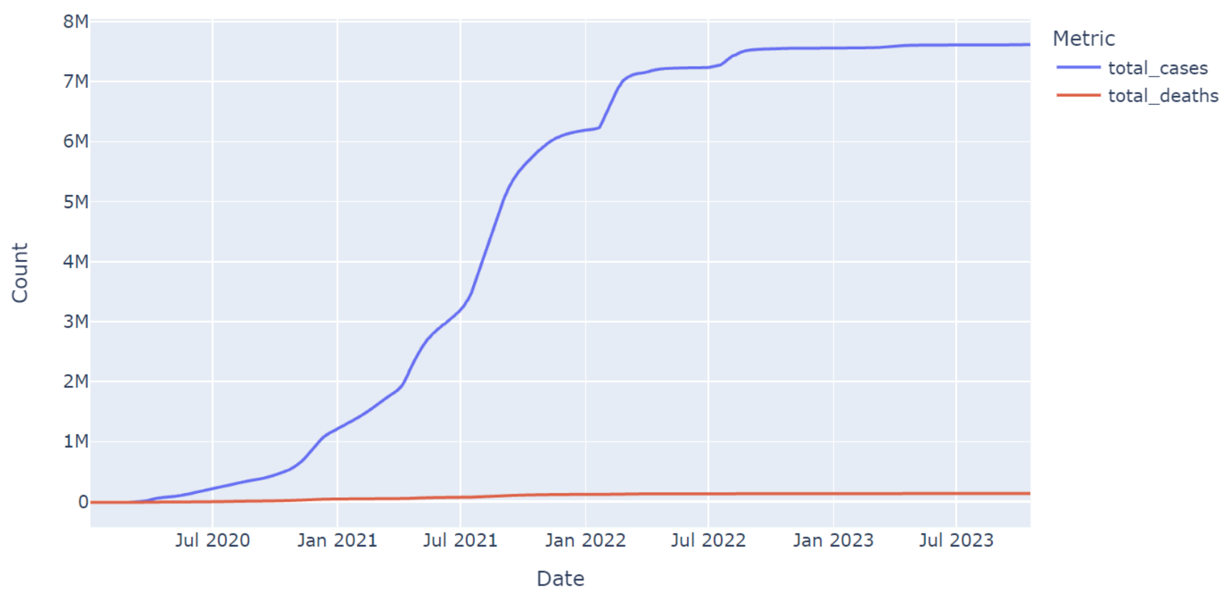
We also can explore new death everyday for each continent separately or in a chart:





We see that pandemic has more effect on Europe even though it started from Asia and Asia has much more population over Europe. It has also the worst influence on North America at first pick.

We could actually see this chart in any country that we want like Iran:



Iran had the same flow as the rest of the world.

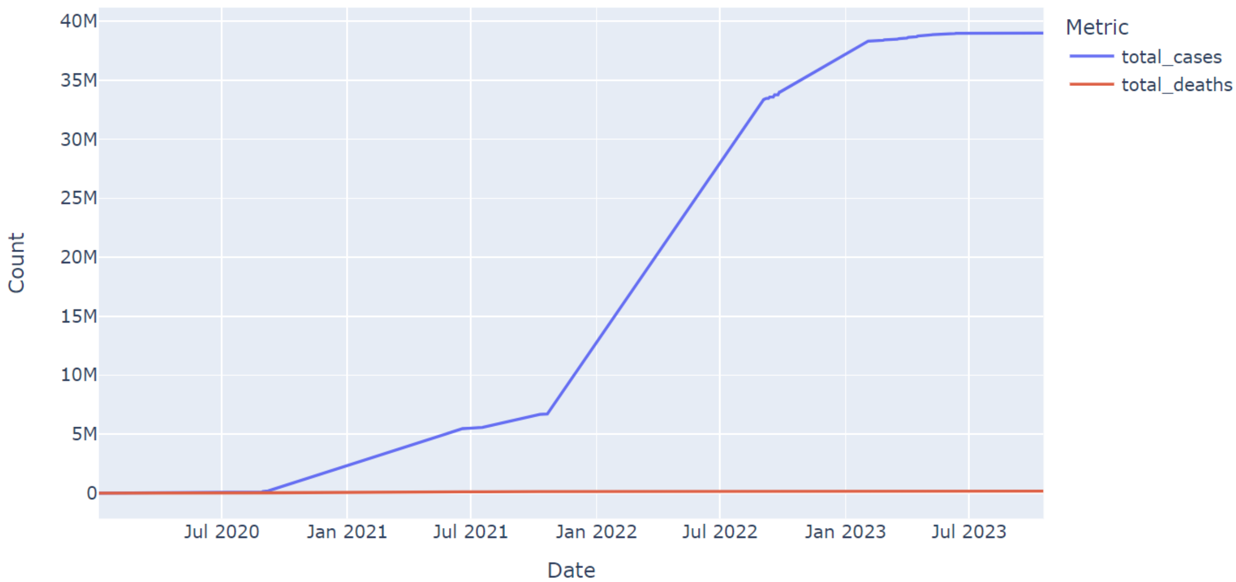
Then we check the total number of deaths for any country that we want. For example:

```
Total deaths in France: 167985.0
```

```
Total deaths in Brazil: 569492.0
```

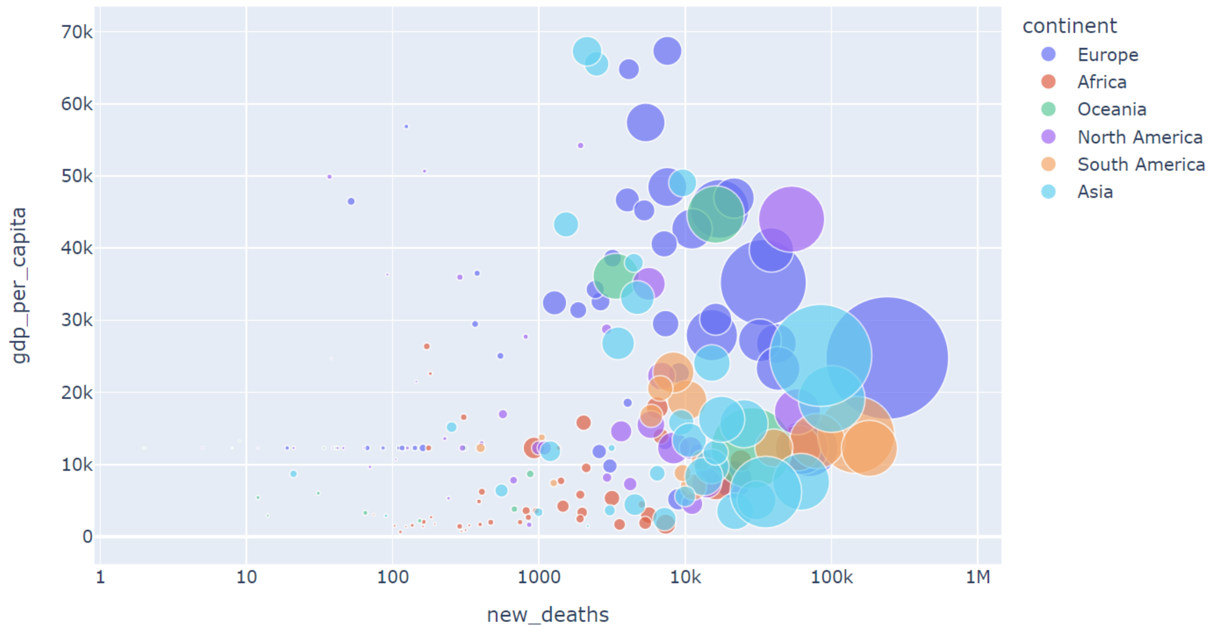
```
Total deaths in Russia: 400102.0
```

And the plot is:



4.2. gdp_per_capita and new_cases clusters over countries

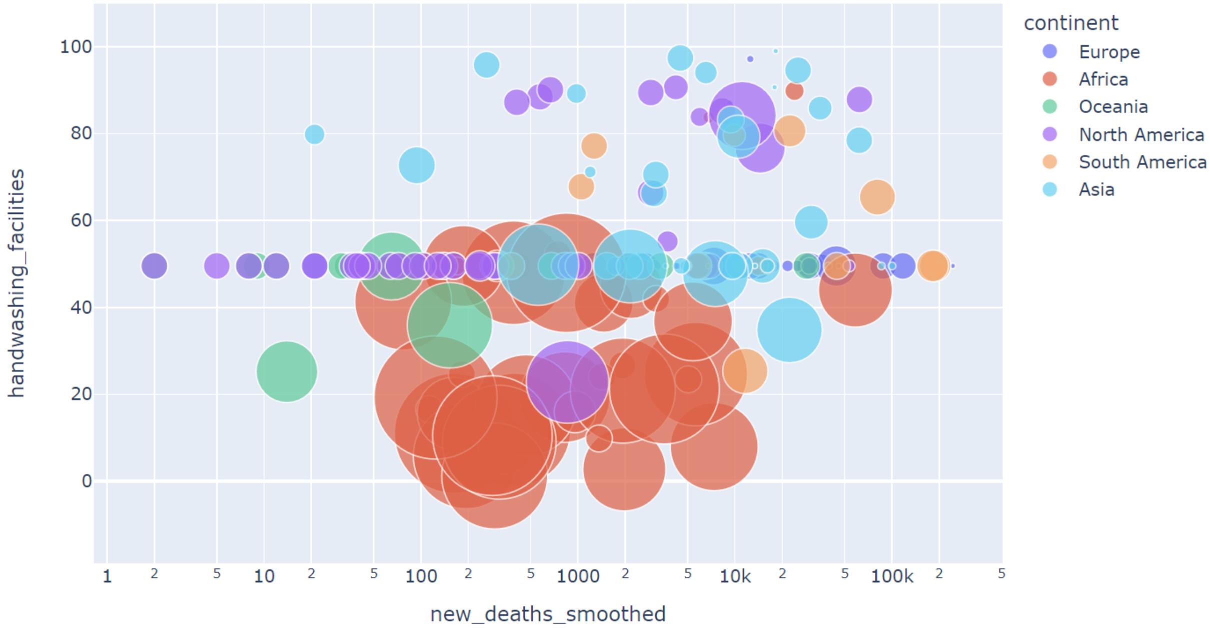
We use the Plotly Express library to create a scatter plot that visualizes the relationship between new deaths, GDP per capita, and new cases for different locations and continents.



The data was grouped by 'location' and 'continent'.Aggregations were performed, summing 'new_deaths', taking the mean of 'gdp_per_capita', and summing 'new_cases' for each group. In the plot, The x-axis represents the total new deaths and the y-axis represents the mean GDP per capita and the size of the markers represents the total new cases. The color of the markers is based on the continent. The hover labels display the location.

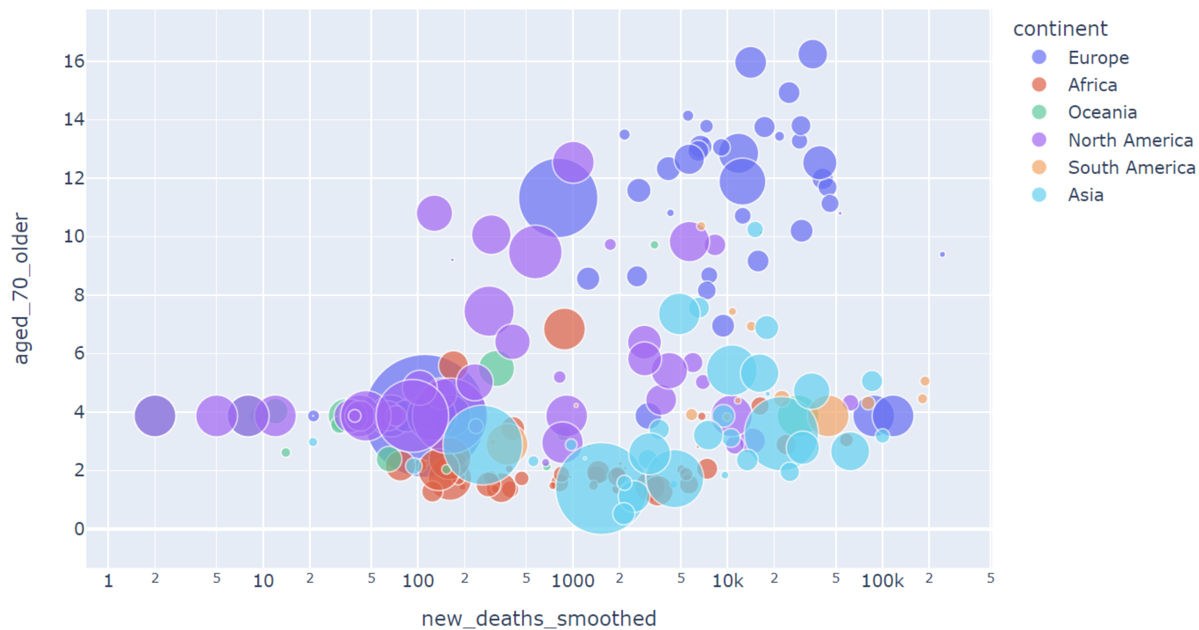
4.3. new_deaths_smoothed_per_million, handwashing_facilities and extreme_poverty clusters over countries

We can also see new_deaths_smoothed_per_million, handwashing_facilities and extreme_poverty clusters over countries related to their continent.



logically , we see more handwashing facilities cause less death and test positive rates. Because we know that viruses are transmitted through hands and mouth so this chart is evidence of that.

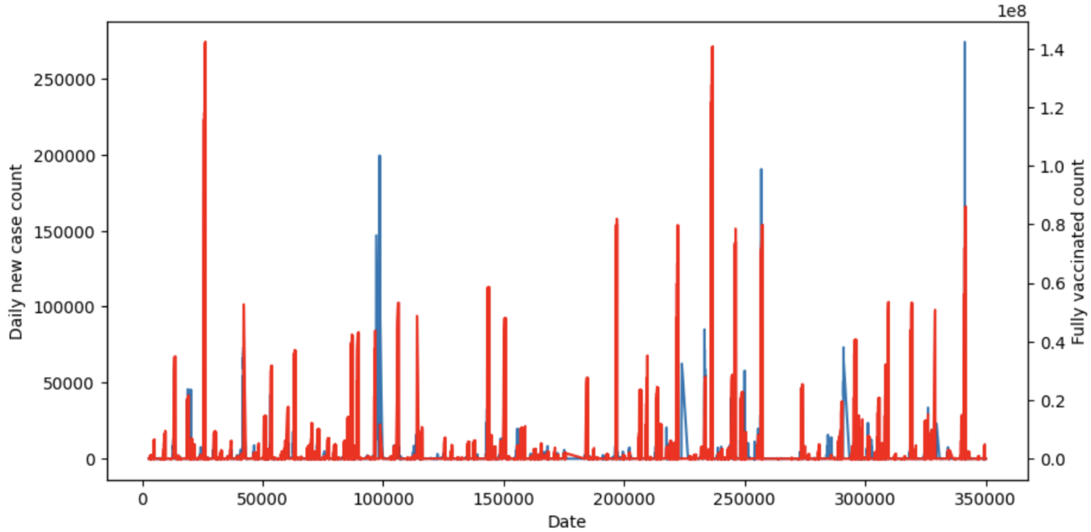
4.4. new_deaths_smoothed, aged_70_older and population_density clusters over countries



We see that Corona-19 virus has more effect on old people and and there are many of them in Europe. So more of these old people were dead in Europe in compared to other continents.

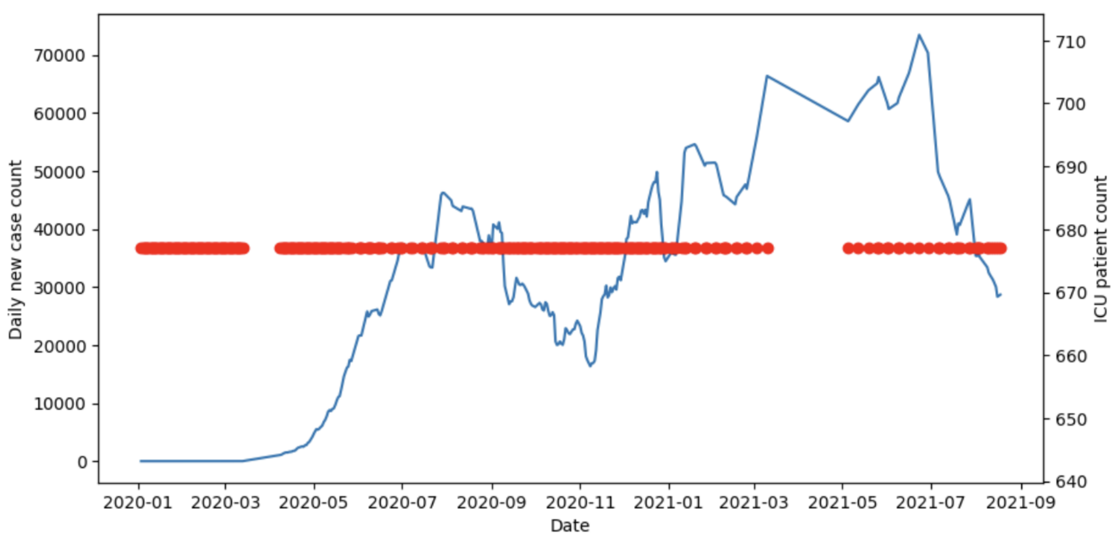
4.5. people_fully_vaccinated and new_cases

This plot visualizes the daily new case count on the left y-axis and the fully vaccinated count on the right y-axis over the specified date range.



We see when vaccination was not started there was not any change in the flow of virus and people still became infected and died because of the virus.

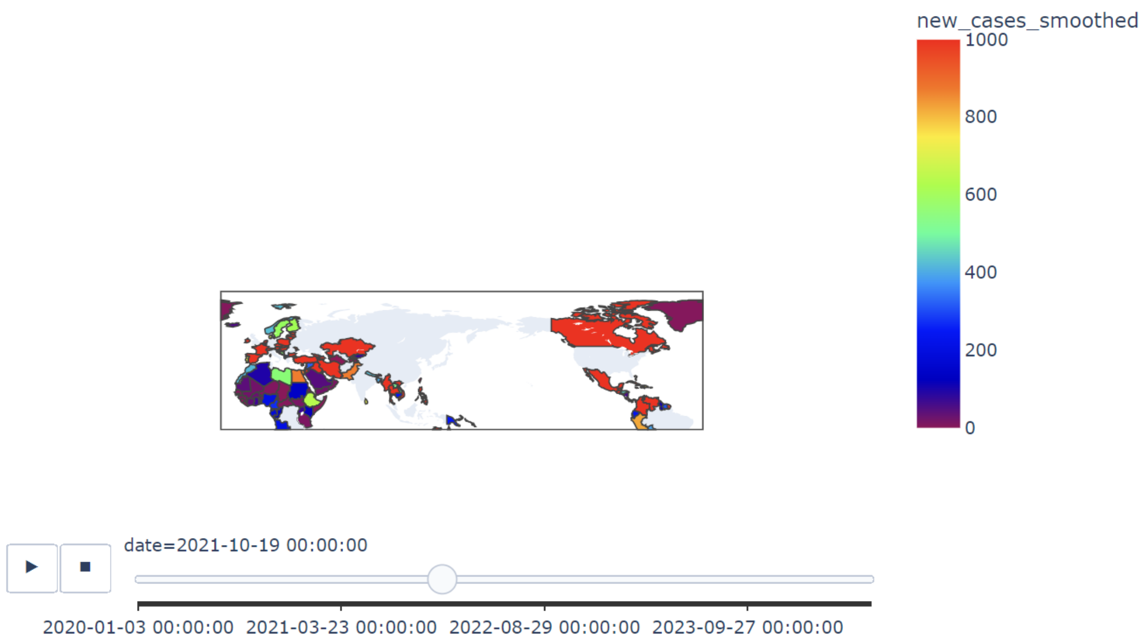
4.6. icu_patients and daily_new_cases

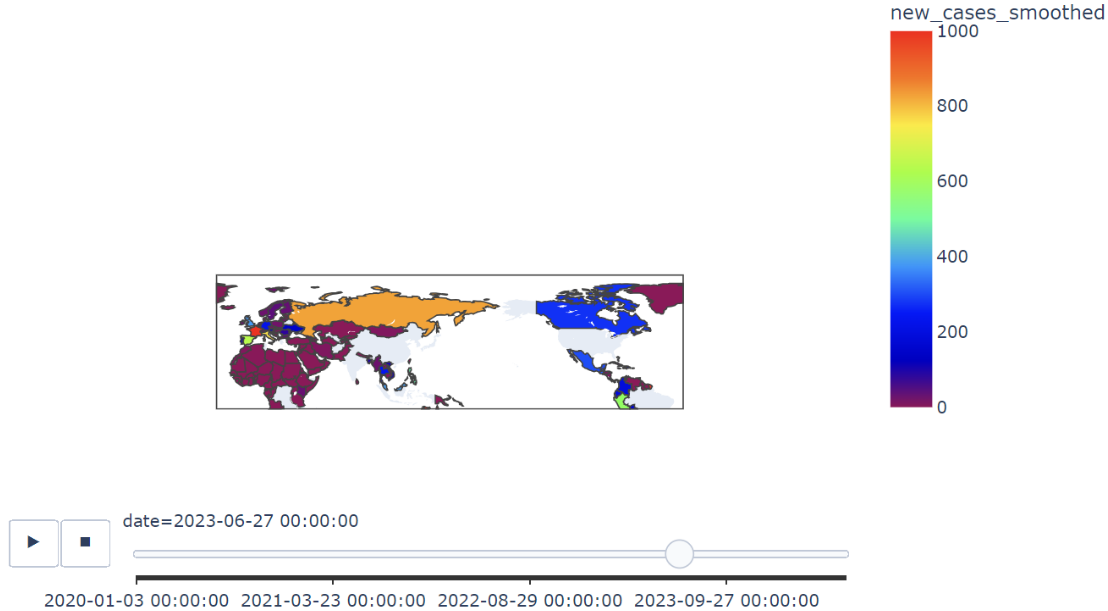


We see in the chart that at first of pandemic most infected people went to icu and dangerous situations but after months less and less people went to icu when they got coronavirus. I guess it is caused by the immunity which was made by our body.

4.7. new_cases in world map over time

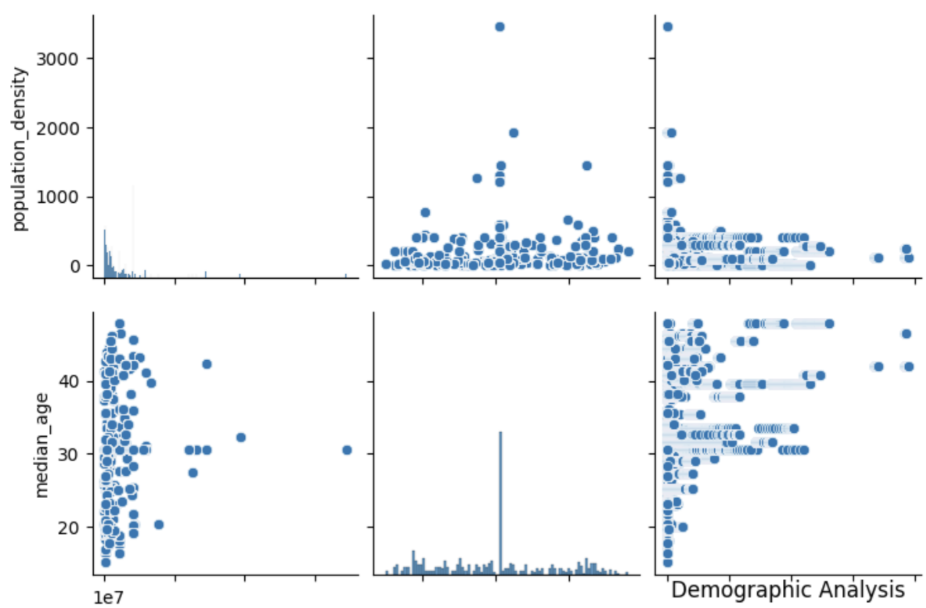
We can see daily new cases over time in each country of the world in the world map using plotly.express library:

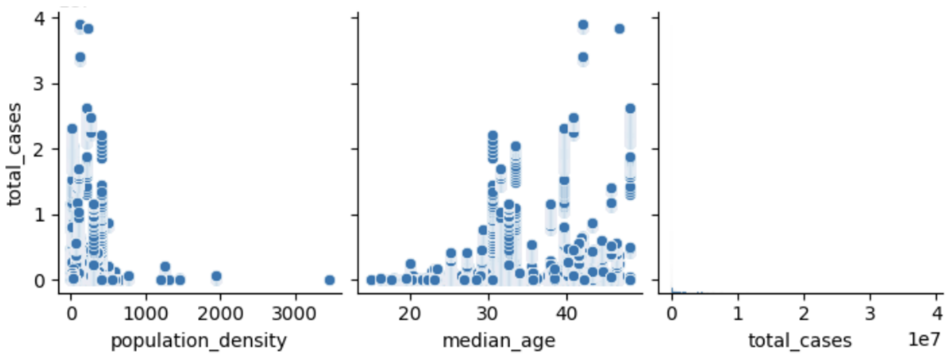




4.8. Demographic Analysis

We check ages and population of the countries that we had in our dataset to see any influence of these in the flow of the virus.





total_cases
4
3
2
1
0
0
1000
2000
3000
population_density
20
30
40
median_age
0
1
2
3
4
total_cases
1e7

5. Conclusion

This assignment serves as a powerful demonstration of how data mining can help us navigate through unprecedented challenges. The synergy between refining data, creating intricate visualizations, and extracting knowledge has not just deepened our comprehension of pandemic dynamics but has also provided us with valuable tools for making informed decisions. As we collaboratively strive to overcome the ongoing global health crisis, the insights gained from this assignment will play a pivotal role in shaping broader discussions. The ultimate aim is to share knowledge with communities, guiding them towards a future that transcends the challenges posed by the pandemic.