Analysis of the current COVID-19 vaccination status across the World

Jana Karas

Introduction

In this study, I will analyse the current vaccination data collected from across the world. I aim to answer three main questions:

- In which countries do I have reliable data about the vaccination status?
- How is the availability of vaccines distributed across the world?
- Do the wealth of a country and the quality of its healthcare system influence the vaccination rate?

To be able to analyse the data properly, the raw data will first be collected from different datasets, preprocessed and cleaned. In the following, the clean data will be analysed and the results will be displayed in several plots. Finally, a conclusion will summarize the findings of the analysis.

Importing and Joining the Datasets

Importing libraries

First of all, the necessary libraries are imported.

```
library(tidyverse)
library(viridis)
library(sf)
library(rnaturalearth)
library(rnaturalearthhires)
library(DataExplorer)
library(lubridate)
library(showtext)
library(patchwork)
```

Importing Vaccination Data

The dataset is provided by “Our World in Data”. “Our World in Data” is a project of the charity organization “Global Change Data Lab”. The project focuses on large global problems such as poverty, climate change and diseases. Information on the dataset can be found here: https://ourworldindata.org/coronavirus. The dataset does not only contain information on vaccinations, but also on cases, deaths, tests, hospitalizations and more related to the COVID-19 pandemic. Furthermore, the dataset contains additional contextual information, such as indices for the wealth of a country. The dataset can be downloaded from GitHub at the following URL: https://github.com/owid/covid-19-data/tree/master/public/data.

The dataset is provided as a csv-file which can be easily read using readr.

```
cov <- read_csv("owid-covid-data.csv")
```

Selecting features

First of all, I will inspect the dataset and explore the features.

```
head(cov)
```

```
## # A tibble: 6 x 67
##   iso_code continent location  date       total_cases new_cases new_cases_smoot~
##   <chr>    <chr>     <chr>     <date>           <dbl>     <dbl>            <dbl>
## 1 AFG      Asia      Afghanis~ 2020-02-24           5         5           NA
## 2 AFG      Asia      Afghanis~ 2020-02-25           5         0           NA
## 3 AFG      Asia      Afghanis~ 2020-02-26           5         0           NA
## 4 AFG      Asia      Afghanis~ 2020-02-27           5         0           NA
## 5 AFG      Asia      Afghanis~ 2020-02-28           5         0           NA
## 6 AFG      Asia      Afghanis~ 2020-02-29           5         0            0.714
## # ... with 60 more variables: total_deaths <dbl>, new_deaths <dbl>,
## #   new_deaths_smoothed <dbl>, total_cases_per_million <dbl>,
## #   new_cases_per_million <dbl>, new_cases_smoothed_per_million <dbl>,
## #   total_deaths_per_million <dbl>, new_deaths_per_million <dbl>,
## #   new_deaths_smoothed_per_million <dbl>, reproduction_rate <dbl>,
## #   icu_patients <dbl>, icu_patients_per_million <dbl>, hosp_patients <dbl>,
## #   hosp_patients_per_million <dbl>, weekly_icu_admissions <dbl>, ...
```

As you can see above, the data has 67 features and about 130000 records. Many of the features are not relevant for the analysis. To keep a better overview, I will drop unnecessary features before adding GDP data and geometric data to the dataset.

As I am aiming to analyse the current vaccination status (and not the effect the vaccinations have), all the features related to the cases, deaths and hospitalizations can be dropped.

I will keep the features of the following categories:

- Location data
- Vaccination data
- Data related to the wealth of the country
- Basic country data (e.g. population)

```
cov <- cov %>%
  select(iso_code, continent, location, date, total_vaccinations, people_vaccinated, people_vaccinated_p

head(cov)
```

```
## # A tibble: 6 x 23
##   iso_code continent location    date       total_vaccinations people_vaccinated
##   <chr>    <chr>     <chr>       <date>                  <dbl>             <dbl>
## 1 AFG      Asia      Afghanistan 2020-02-24                 NA                NA
## 2 AFG      Asia      Afghanistan 2020-02-25                 NA                NA
## 3 AFG      Asia      Afghanistan 2020-02-26                 NA                NA
## 4 AFG      Asia      Afghanistan 2020-02-27                 NA                NA
## 5 AFG      Asia      Afghanistan 2020-02-28                 NA                NA
## 6 AFG      Asia      Afghanistan 2020-02-29                 NA                NA
## # ... with 17 more variables: people_vaccinated_per_hundred <dbl>,
## #   people_fully_vaccinated <dbl>, total_boosters <dbl>,
## #   new_vaccinations <dbl>, new_vaccinations_smoothed <dbl>,
## #   total_vaccinations_per_hundred <dbl>,
## #   people_fully_vaccinated_per_hundred <dbl>,
## #   total_boosters_per_hundred <dbl>,
## #   new_vaccinations_smoothed_per_million <dbl>, ...
```

Now I am left with 23 features that are relevant for the analysis.

The vaccination features are described by Our World in Data as follows:

- *total_vaccinations*: Total number of COVID-19 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

Filtering the values

Removing continent data The COVID dataset also contains codes for regions and continents (e.g. the code "OWID_AFR" for Africa, see below). Since I can easily summarize this using the remaining records, I want to remove these records. They all start with "OWID", which is why they can be easily removed.

```
cov %>%
  filter((str_detect(iso_code, "OWID"))) %>%
  select(iso_code, continent, location) %>%
  head()
```

```
## # A tibble: 6 x 3
##   iso_code continent location
##   <chr>    <chr>     <chr>
## 1 OWID_AFR <NA>      Africa
## 2 OWID_AFR <NA>      Africa
## 3 OWID_AFR <NA>      Africa
## 4 OWID_AFR <NA>      Africa
## 5 OWID_AFR <NA>      Africa
## 6 OWID_AFR <NA>      Africa
```

The following code block deletes the "OWID" records.

```
cov <- cov %>%
  filter(!(str_detect(iso_code, "OWID")))
```

Removing earlier dates and very recent dates The first person that was ever vaccinated against COVID was vaccinated on the 9th of December 2020 as you might remember from the news (it was a 90-year-old lady in Great Britain). Therefore, I will only keep the data beginning in December 2020. The data in January 2022 is not reliable yet since usually it takes some time for countries to report data and "Our World in Data" to collect it, which is why I will remove this month as well.

```
cov <- cov %>%
  filter(date >= "2020-12-01") %>%
  filter(date <= "2021-12-31")
```

```
cov %>%
  select(date) %>%
  distinct() %>%
  count()
```

```
## # A tibble: 1 x 1
##       n
##   <int>
## 1   396
```

Now I am left with data for 396 days (01.12.2020 until 31.12.2021).

Importing Data on Healthcare System

To evaluate the quality of the Healthcare System, I am using the UHC service coverage index. This is the index that the World Health Organization (WHO) officially uses to measure the progress of its goal to reach universal health coverage. The indicator is computed as the geometric mean of 14 tracer indicators such as child health, infectious diseases and service capacity.

The latest data available is from 2017. This is beneficial for our analysis since the COVID-19-pandemic does not influence the indicator. Therefore, it is an independent variable.

The dataset and its metadata can be accessed on the WHO webpage at this link: https://www.who.int/data/gho/data/indicators/indicator-details/GHO/uhc-index-of-service-coverage.

```
service_coverage <- read_csv("service_coverage.csv")
head(service_coverage)
```

```
## # A tibble: 6 x 34
##   IndicatorCode      Indicator        ValueType ParentLocationC~ ParentLocation
##   <chr>              <chr>            <chr>     <chr>            <chr>
## 1 UHC_INDEX_REPORTED UHC index of es~ numeric   EMR              Eastern Medite~
## 2 UHC_INDEX_REPORTED UHC index of es~ numeric   AFR              Africa
## 3 UHC_INDEX_REPORTED UHC index of es~ numeric   AFR              Africa
## 4 UHC_INDEX_REPORTED UHC index of es~ numeric   AFR              Africa
## 5 UHC_INDEX_REPORTED UHC index of es~ numeric   AFR              Africa
## 6 UHC_INDEX_REPORTED UHC index of es~ numeric   EMR              Eastern Medite~
## # ... with 29 more variables: Location type <chr>, SpatialDimValueCode <chr>,
## #   Location <chr>, Period type <chr>, Period <dbl>, IsLatestYear <lgl>,
## #   Dim1 type <lgl>, Dim1 <lgl>, Dim1ValueCode <lgl>, Dim2 type <lgl>,
## #   Dim2 <lgl>, Dim2ValueCode <lgl>, Dim3 type <lgl>, Dim3 <lgl>,
## #   Dim3ValueCode <lgl>, DataSourceDimValueCode <lgl>, DataSource <lgl>,
## #   FactValueNumericPrefix <lgl>, FactValueNumeric <dbl>, FactValueUoM <lgl>,
## #   FactValueNumericLowPrefix <lgl>, FactValueNumericLow <lgl>, ...
```

Filtering the values

I want to only filter for the latest value of the year. Therefore, I check, if for any country, the latest year is not available.

```
service_coverage %>%
  group_by(Location) %>%
  summarise(max_period = max(Period)) %>%
  distinct(max_period)
```

```
## # A tibble: 1 x 1
##   max_period
##        <dbl>
## 1       2017
```

This is the proof that the most recent value is available for every country. Therefore, I can filter the Period to 2017.

```
service_coverage <- service_coverage %>%
  filter(Period == 2017)
```

Selecting features

Removing features with only one distinct values Some of the features can probably be removed. Let's check how many distinct values are in each column.

```
service_coverage %>% summarise_all(n_distinct)
```

```
## # A tibble: 1 x 34
##   IndicatorCode Indicator ValueType ParentLocationCode ParentLocation
##           <int>     <int>     <int>              <int>          <int>
## 1             1         1         1                  6              6
## # ... with 29 more variables: Location type <int>, SpatialDimValueCode <int>,
## #   Location <int>, Period type <int>, Period <int>, IsLatestYear <int>,
## #   Dim1 type <int>, Dim1 <int>, Dim1ValueCode <int>, Dim2 type <int>,
## #   Dim2 <int>, Dim2ValueCode <int>, Dim3 type <int>, Dim3 <int>,
## #   Dim3ValueCode <int>, DataSourceDimValueCode <int>, DataSource <int>,
## #   FactValueNumericPrefix <int>, FactValueNumeric <int>, FactValueUoM <int>,
## #   FactValueNumericLowPrefix <int>, FactValueNumericLow <int>, ...
```

There is only one distinct value in many of the 34 features. I will remove all these features from the dataframe.

```
service_coverage <- service_coverage %>%
  select(ParentLocationCode, ParentLocation, SpatialDimValueCode, Location, FactValueNumeric, Value)
head(service_coverage)
```

```
## # A tibble: 6 x 6
##   ParentLocationCode ParentLocation  SpatialDimValue~ Location  FactValueNumeric
##   <chr>              <chr>           <chr>            <chr>                <dbl>
## 1 EMR                Eastern Medite~ SOM              Somalia                 25
## 2 AFR                Africa          TCD              Chad                    28
## 3 AFR                Africa          MDG              Madagasc~               28
## 4 AFR                Africa          SSD              South Su~               31
## 5 AFR                Africa          CAF              Central ~               33
## 6 EMR                Eastern Medite~ AFG              Afghanis~               37
## # ... with 1 more variable: Value <dbl>
```

Removing equivalent features Now I am left with the location data and two different values, “FactValueNumeric” and “Value”. At the first glance, it looks like these two values are the same. The following code block checks if this assumption is true:

```
service_coverage %>%
  transmute(same = (FactValueNumeric == Value)) %>%
  distinct()
```

```
## # A tibble: 1 x 1
##   same
##   <lgl>
## 1 TRUE
```

The assumption is true. Therefore, I can delete the feature “FactValueNumeric”.

```
service_coverage <- service_coverage %>%
  select(-FactValueNumeric)
```

Removing unneeded location data Furthermore, I will delete all the location data but the “SpatialDimValueCode”. This is the best feature to join this dataset with the COVID dataset. Since in the covid dataset this type of value is named “iso_code”, I will rename the feature. Furthermore, when joining the datasets later, the name “Value” for the feature will not be meaningful. Therefore, I will rename it to “essential_service_coverage”.

```
service_coverage <- service_coverage %>%
  select(SpatialDimValueCode, Value) %>%
  rename("iso_code" = SpatialDimValueCode, "essential_service_coverage" = Value)
head(service_coverage)
```

```
## # A tibble: 6 x 2
##   iso_code essential_service_coverage
##   <chr>                         <dbl>
## 1 SOM                              25
## 2 TCD                              28
## 3 MDG                              28
## 4 SSD                              31
## 5 CAF                              33
## 6 AFG                              37
```

Joining the COVID dataset and the health service dataset

To join the COVID dataset with the health service dataset, I will use the iso_code of the country. Before doing the join, I should check whether the iso_code columns in both datasets contain the same values.

```
cov_iso_codes <- cov %>%
  select(iso_code, location) %>%
  distinct()
cov_iso_codes %>% count()
```

```
## # A tibble: 1 x 1
##       n
##   <int>
## 1   223
```

There are 222 distinct values for countries in the feature iso_code of the COVID dataset.

```
service_iso_codes <- service_coverage %>%
  select(iso_code) %>%
  distinct()
service_iso_codes %>% count()
```

```
## # A tibble: 1 x 1
##       n
##   <int>
## 1   183
```

In the health service dataset, there are only 183 distinct values. Let's check if any of these 183 values are in the Health Service Dataset but not in the COVID dataset:

```
anti_join(service_iso_codes, cov_iso_codes, by = "iso_code")
```

```
## # A tibble: 1 x 1
##   iso_code
##   <chr>
## 1 PRK
```

"PRK" is the code for North Korea. Let's do the check additionally the other way round: which values are in the COVID dataset but not in the Health Service dataset?

```
anti_join(cov_iso_codes, service_iso_codes, by = "iso_code")
```

```
## # A tibble: 41 x 2
##    iso_code location
##    <chr>    <chr>
##  1 AND      Andorra
##  2 AIA      Anguilla
##  3 ABW      Aruba
##  4 BMU      Bermuda
##  5 BES      Bonaire Sint Eustatius and Saba
##  6 VGB      British Virgin Islands
##  7 CYM      Cayman Islands
##  8 COK      Cook Islands
##  9 CUW      Curacao
## 10 DMA      Dominica
## # ... with 31 more rows
```

North Korea is not in the list, so apparently there is no COVID data present for North Korea. This is not very surprising due to the political situation in the country. Until spring 2021, North Koreas dictator Kim Jong Un claimed that there was not a single case. Afterwards, he admitted that there were cases but still the country does not publish any numbers.

That is why, for this analysis, I will drop North Korea:

```
service_coverage %>%
  filter(!(iso_code == "PRK"))
```

```
## # A tibble: 182 x 2
##    iso_code essential_service_coverage
##    <chr>                         <dbl>
##  1 SOM                              25
##  2 TCD                              28
##  3 MDG                              28
##  4 SSD                              31
##  5 CAF                              33
##  6 AFG                              37
##  7 GIN                              37
##   8 NER                              37
##   9 ERI                              38
##  10 MLI                              38
## # ... with 172 more rows
```

The countries that are not present in the health service dataset are mostly very small countries or regions that are not independent countries (such as British Oversea Territories). These countries and regions are not that interesting for the analysis and obviously, obtaining additional data such as the Health Service Coverage or values such as the GDP is difficult due to their small size. Therefore, I will drop these countries. The only two bigger and sovereign countries in the above list are Taiwan and Palestine. Since I want to include Taiwan and Palestine in the analysis of the vaccination data, I will not exclude them although I do not have any data on the Health Service Coverage.

In the vector "exclude_countries", I am going to save the ISO codes of the 38 small or not independent countries that I want to exclude from the dataset by performing an anti join.

```
exclude_countries <- anti_join(cov_iso_codes, service_iso_codes, by = "iso_code") %>%
  filter(!(iso_code == "PSE" | iso_code == "TWN")) %>%
  select(iso_code)
```

In the following, I am going to exclude the countries.

```
cov <- anti_join(cov, exclude_countries, by = "iso_code")
```

Now, everything is prepared for the join. I will add the Health Service Index to the COVID dataset by performing a left join.

```
cov <- left_join(cov, service_coverage, by = "iso_code")
```

Importing GeoData

For importing the GeoData that is necessary to create maps, I am using the Natural Earth Package. I am going to import the country polygons for the bigger countries in the world. Tiny countries cannot be imported as polygons but only as points, which will not be visible on a world map anyways. Since the goal of this project is not to analyze tiny countries in particular, I will not display them on the map. For the maps, I am going to use the Eckert IV projection since this projection keeps the area of the countries undistorted. It is already a problem that the size of a country does not represent the number of people that live in that country - but displaying data for different countries in a world map is still very common. Therefore, I do not want to distort that picture even more by using a projection that does not keep the area.

```
geo <- ne_countries(returnclass = "sf") %>%
  st_transform("+proj=eck4")
```

Renaming features

To join the GeoData with the COVID dataset, I need to rename the features "sov_a3" (this is the iso_code) and the "sovereignt" (which equals the location in the COVID dataset).

```
geo <- geo %>%
  rename(iso_code = sov_a3, location = sovereignt)
```

Preparing the GeoData for a later join with the COVID dataset

To create map plots later, I need to add GeoData for the countries that are present in the COVID dataset. Again, I will use the iso_codes for joining the two datasets.

```
cov_iso_codes <- cov %>%
  select(iso_code, location) %>%
  distinct()
cov_iso_codes %>% count()
```

```
## # A tibble: 1 x 1
##       n
##   <int>
## 1   184
```

There are now 184 distinct values for countries in the feature iso_code of the COVID dataset.

```
geo_iso_codes <- geo %>%
  tibble() %>%
  select(iso_code, location) %>%
  distinct()
geo_iso_codes %>% count()
```

```
## # A tibble: 1 x 1
##       n
##   <int>
## 1   171
```

In the GeoData, there are only 171 distinct values. Let's check if any of these 171 values are in the GeoData but not in the COVID dataset:

```
temp1 <- anti_join(geo_iso_codes, cov_iso_codes, by = "iso_code")
count(temp1)
```

```
## # A tibble: 1 x 1
##       n
##   <int>
## 1    16
```

```
head(temp1)
```

```
## # A tibble: 6 x 2
##   iso_code location
##   <chr>    <chr>
## 1 ATA      Antarctica
## 2 FR1      France
## 3 AU1      Australia
## 4 CH1      China
## 5 CYN      Northern Cyprus
## 6 DN1      Denmark
```

This is the case for 16 countries. Since countries such as France or Denmark are for sure present in the COVID dataset, I will do the check the other way round to find out about iso_codes that are simply not the same.

```
temp2 <- anti_join(cov_iso_codes, geo_iso_codes, by = "iso_code")
count(temp2)
```

```
## # A tibble: 1 x 1
##       n
##   <int>
## 1    29
```

Countries that simply have another code are:

```
inner_join(temp1, temp2, by = "location")
```

```
## # A tibble: 9 x 3
##   iso_code.x location       iso_code.y
##   <chr>      <chr>          <chr>
## 1 FR1        France         FRA
## 2 AU1        Australia      AUS
## 3 CH1        China          CHN
## 4 DN1        Denmark        DNK
## 5 FI1        Finland        FIN
## 6 GB1        United Kingdom GBR
## 7 NL1        Netherlands    NLD
## 8 NZ1        New Zealand    NZL
## 9 SDS        South Sudan    SSD
```

…plus the United States.

In the following code block, I will rename these codes.

```
geo <- geo %>%
  mutate(iso_code = fct_recode(iso_code, AUS = "AU1",CHN = "CH1",DNK ="DN1", FIN="FI1",
                               NLD="NL1", FRA="FR1", NZL="NZ1", SSD = "SDS",
                               GBR = "GB1", USA = "US1"))
```

Let's go back and check which countries are missing in each dataset again. The following code block checks which countries are present in the GeoData but not in the COVID dataset:

```
geo_iso_codes <- geo %>%
  tibble() %>%
  select(iso_code, location) %>%
  distinct()

anti_join(geo_iso_codes, cov_iso_codes, by = "iso_code")
```

```
## # A tibble: 6 x 2
##   iso_code location
##   <fct>    <chr>
## 1 ATA      Antarctica
## 2 CYN      Northern Cyprus
## 3 KOS      Kosovo
## 4 PRK      North Korea
## 5 SAH      Western Sahara
## 6 SOL      Somaliland
```

These six countries are not present in the COVID dataset.

I will still keep the GeoData for these countries to display a complete map and show countries for which I do not have any COVID data (which is also an interesting finding).

Let's also see which countries are present in the COVID dataset but not in the GeoData.

```
anti_join(cov_iso_codes, geo_iso_codes, by = "iso_code")
```

```
## # A tibble: 19 x 2
##    iso_code location
##    <chr>    <chr>
##  1 ATG      Antigua and Barbuda
##  2 BHR      Bahrain
##   3 BRB      Barbados
##   4 CPV      Cape Verde
##   5 COM      Comoros
##   6 GRD      Grenada
##   7 KIR      Kiribati
##   8 MDV      Maldives
##   9 MLT      Malta
##  10 MUS      Mauritius
##  11 FSM      Micronesia (country)
##  12 PSE      Palestine
##  13 LCA      Saint Lucia
##  14 VCT      Saint Vincent and the Grenadines
##  15 WSM      Samoa
##  16 STP      Sao Tome and Principe
##  17 SYC      Seychelles
##  18 SGP      Singapore
##  19 TON      Tonga
```

As already stated above, smaller countries cannot be displayed as polygons and won't be visible on the map anyways. Therefore, to display the numbers in these countries, I will need to find another type of plotting.

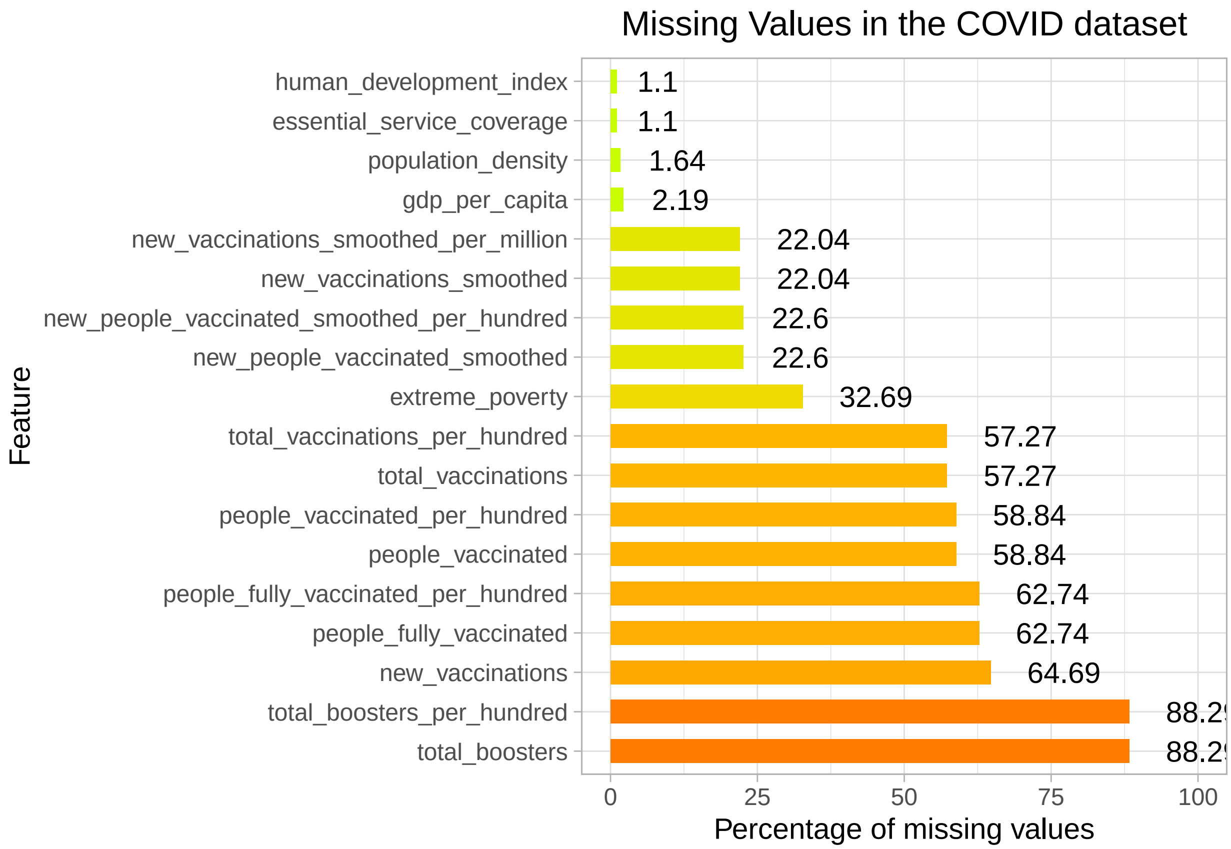
I will join the dataset with the GeoData later in the preprocessing of the maps, since it is faster to join when the COVID dataset is already summarized.

Exploration and Preprocessing

Analysis of Missing Values

A visual check of the dataset showed that missing values are encoded as "NA" and not with another symbol such as "?" or "-". Therefore, I can use the functions for identifying and handling NA values.

```
rows <- nrow(cov)

missing_values <- cov %>%
  gather(key = "key", value = "val") %>%
  mutate(is_missing = is.na(val)) %>%
  group_by(key, is_missing) %>%
  summarise(num_missing = n()) %>%
  filter(is_missing==T) %>%
  select(-is_missing) %>%
  arrange(desc(num_missing)) %>%
  mutate(percent_missing = round(num_missing / rows * 100, digits = 2))

missing_values$key <- factor(missing_values$key, levels = missing_values$key)

missing_values_cov <- missing_values %>%
  ggplot(aes(x=key, y=percent_missing, fill = percent_missing)) +
  geom_col(width = 0.6) +
  geom_text(aes(label = percent_missing), hjust = -0.5) +
  coord_flip() +
  scale_fill_gradient2(low="#c8ff00", mid = "#ffc300", high = "#ff6500", midpoint = 50) +
  scale_y_continuous(limits = c(0, 100)) +
  theme_light() +
  theme(plot.title = element_text(hjust = 0.5)) +
  labs(title = "Missing Values in the COVID dataset",
      y = "Percentage of missing values",
      x = "Feature"
  ) +
  guides(fill = "none")

missing_values_cov
```



As you can see from the plot, there are a lot of missing values in the COVID dataset.

- For the general country data such as the life expectancy or the population, there is hardly any missing data (the only exemption is the “extreme_poverty” feature). This data is constant for each country and is not affected by the time series.
- For the vaccination data, there are a lot of missing values. Analyzing the missing data will also be interesting since this shows how the reporting system in different countries work (or do not work).
- Since booster vaccinations just recently started in richer countries, there is only few data available. This is not surprising, but the available data should be used with care.

It will be interesting to see how the missing values are distributed across time. For instance, I would expect that the available data on booster vaccinations is very recent and for previous dates there are only NA values existent in the dataset.

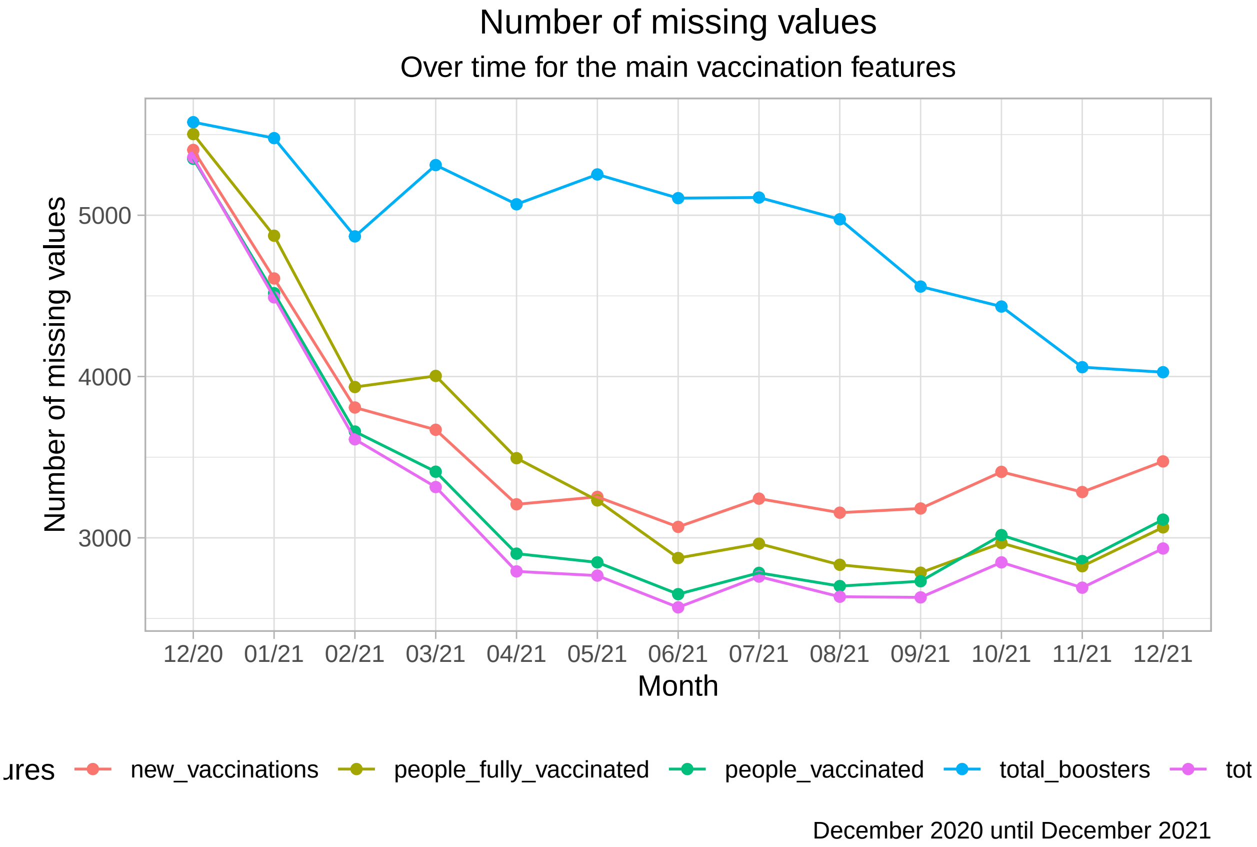
For this analysis, I will use the following features on vaccinations:

- new_vaccinations
- total_vaccinations
- people_vaccinated
- people_fully_vaccinated
- total_boosters.

The other features on vaccinations were derived from these features, which can also be seen from the similar amount of missing values (e.g. new_vaccinations_smoothed_per_million has the same percentage of missing values as the new_vaccinations_smoothed feature).

```
na_values_per_month <- cov %>%
  select(date, new_vaccinations,  total_vaccinations, people_vaccinated, people_fully_vaccinated,
         total_boosters) %>%
  group_by(month = lubridate::floor_date(date, "month")) %>%
  replace_na(list(new_vaccinations = -1, new_vaccinations_smoothed = -1, total_vaccinations = -1,
                  people_vaccinated = -1, people_fully_vaccinated = -1, total_boosters = -1)) %>%
  pivot_longer(cols = new_vaccinations:total_boosters, names_to = "feature", values_to = "test") %>%
  filter(test == -1) %>%
  group_by(month, feature) %>%
  summarize(number_of_na = n()) %>%
  mutate(month = as.character(month))
```

```
month_labs = c("12/20", "01/21", "02/21", "03/21", "04/21", "05/21", "06/21", "07/21", "08/21", "09/21"

na_values_per_month %>%
  ggplot(aes(x=month, y=number_of_na, color = feature)) +
  geom_point() +
  geom_line(aes(group = feature)) +
  theme_light() +
  theme(aspect.ratio = 0.5,
        legend.position = "bottom",
        plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5)) +
  scale_x_discrete(labels = month_labs) +
  labs(title = "Number of missing values",
       subtitle = "Over time for the main vaccination features",
       caption = "December 2020 until December 2021",
       color = "Features",
       x = "Month",
       y = "Number of missing values")
```



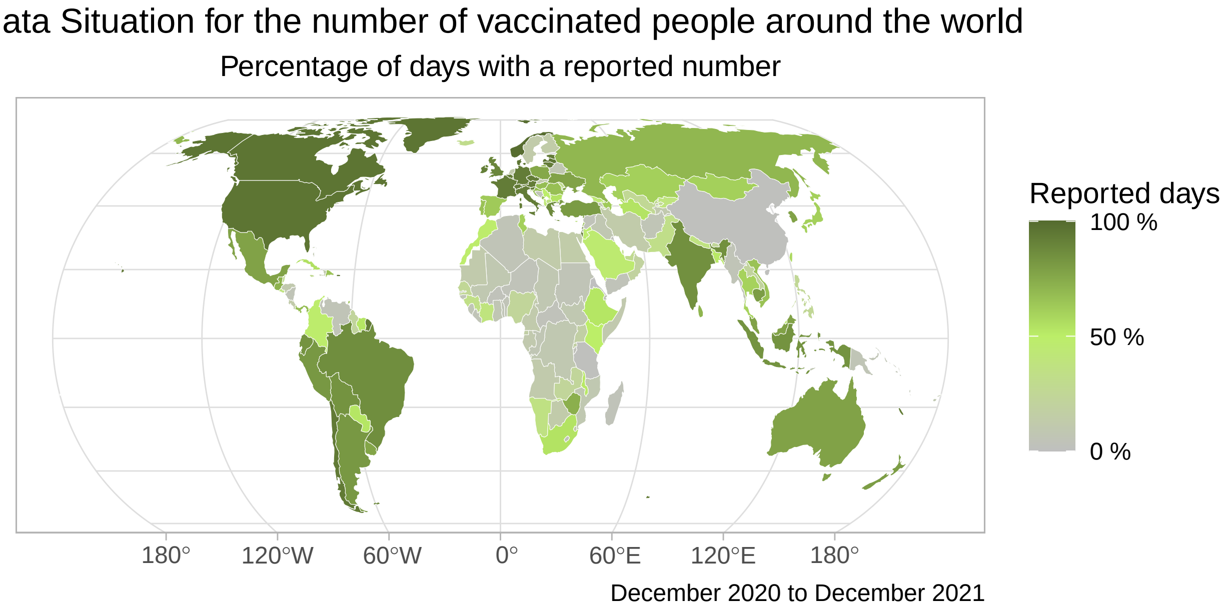
As expected, there are a lot of NA values at the beginning of the time period and less NA values for later dates. Furthermore, the least data is available for booster vaccinations, which was also expected.

Another interesting analysis could be which countries have the most NA values. Since the different features have a roughly similar trend for the different features, I will choose one of the features for this analysis. I will use the feature “people_vaccinated”.

```
na_values_by_countries <- cov %>%
  select(iso_code, location, people_vaccinated, essential_service_coverage, continent) %>%
  replace_na(list(people_vaccinated = -1)) %>%
  filter(people_vaccinated == -1) %>%
  group_by(iso_code, location, essential_service_coverage, continent) %>%
  summarise(number_of_na = n()) %>%
  mutate(reported_days = as.integer((396-number_of_na)*100/396)) %>%
  select(-number_of_na) %>%
  as.data.frame()
```

For creating a map, I have to now add the GeoData to the dataframe. Since I already checked the join criterion, the iso code, in the previous section, I can do the join now without worrying.

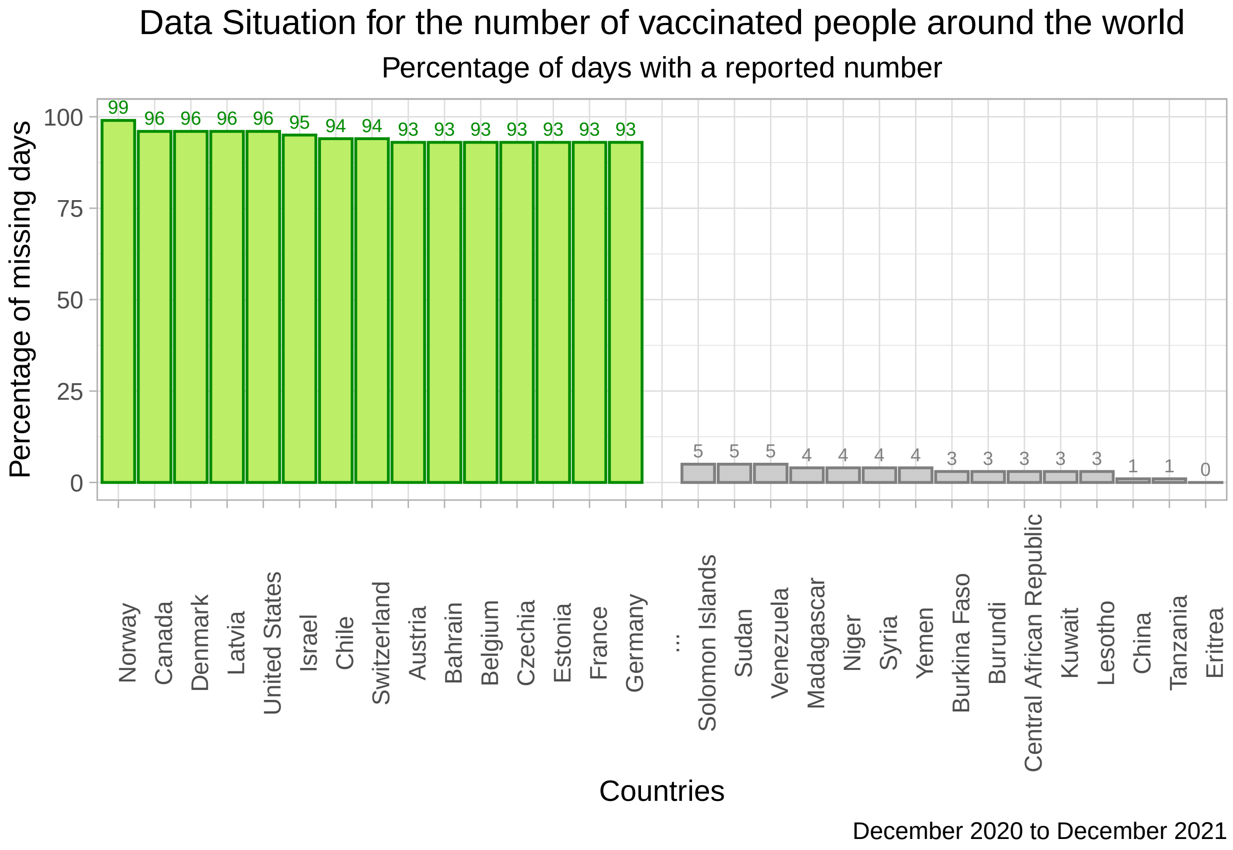
```
geo %>%
  inner_join(na_values_by_countries, by = "iso_code") %>%
  ggplot(aes(geometry = geometry)) +
  geom_sf(aes(fill = reported_days), color = "white", size = 0.1) +
  labs(title = "Data Situation for the number of vaccinated people around the world",
       subtitle = "Percentage of days with a reported number",
       caption = "December 2020 to December 2021",
       fill = "Reported days") +
  theme_light() +
  theme(plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5)) +
  scale_fill_gradient2(low = "gray75", mid = "darkolivegreen2", high = "darkolivegreen", midpoint = 50,
                       limits = c(0, 100),
                       breaks = c(0, 50, 100),
                       labels = c("0 %", "50 %", "100 %"))
```


ata Situation for the number of vaccinated people around the world
Percentage of days with a reported number
Reported days
100 %
50 %
0 %
180°
120°W
60°W
0°
60°E
120°E
180°
December 2020 to December 2021

As one can see from the map, there are a lot of missing values in Africa, Middle East and China. Additionally, some countries in Central America and Europe have quite some missing values.

```
na_values_by_countries_2 <- na_values_by_countries %>%
  arrange(desc(reported_days), location) %>%
  add_row(location = "...", reported_days = NA, .before = 16) %>%
  filter(row_number() < 17 | row_number() > 170) %>%
  mutate(col=ifelse(reported_days>50, "greater", "less"))
```

```
na_values_by_countries_2$location <- factor(na_values_by_countries_2$location, levels = na_values_by_cou

na_values_by_countries_2 %>%
  ggplot(aes(x=location, y=reported_days, fill=col, color=col)) +
  geom_col() +
  theme_light() +
  scale_y_continuous(limits=c(0, 100)) +
  theme(axis.text.x = element_text(angle = 90), legend.position = "none") +
  geom_text(aes(label = reported_days), vjust = -0.5, size = 2.5) +
  scale_fill_manual(values = c("darkolivegreen2", "gray80")) +
  scale_color_manual(values = c("green4", "gray50")) +
  labs(title = "Data Situation for the number of vaccinated people around the world",
       subtitle = "Percentage of days with a reported number",
       caption = "December 2020 to December 2021",
       x = "Countries",
       y = "Percentage of missing days"
       ) +
  theme(plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5))
```



The bar chart shows again that many European countries are amongst the countries which reported most reliably. On the other hand, countries in Africa and Middle East reported the least reliably.

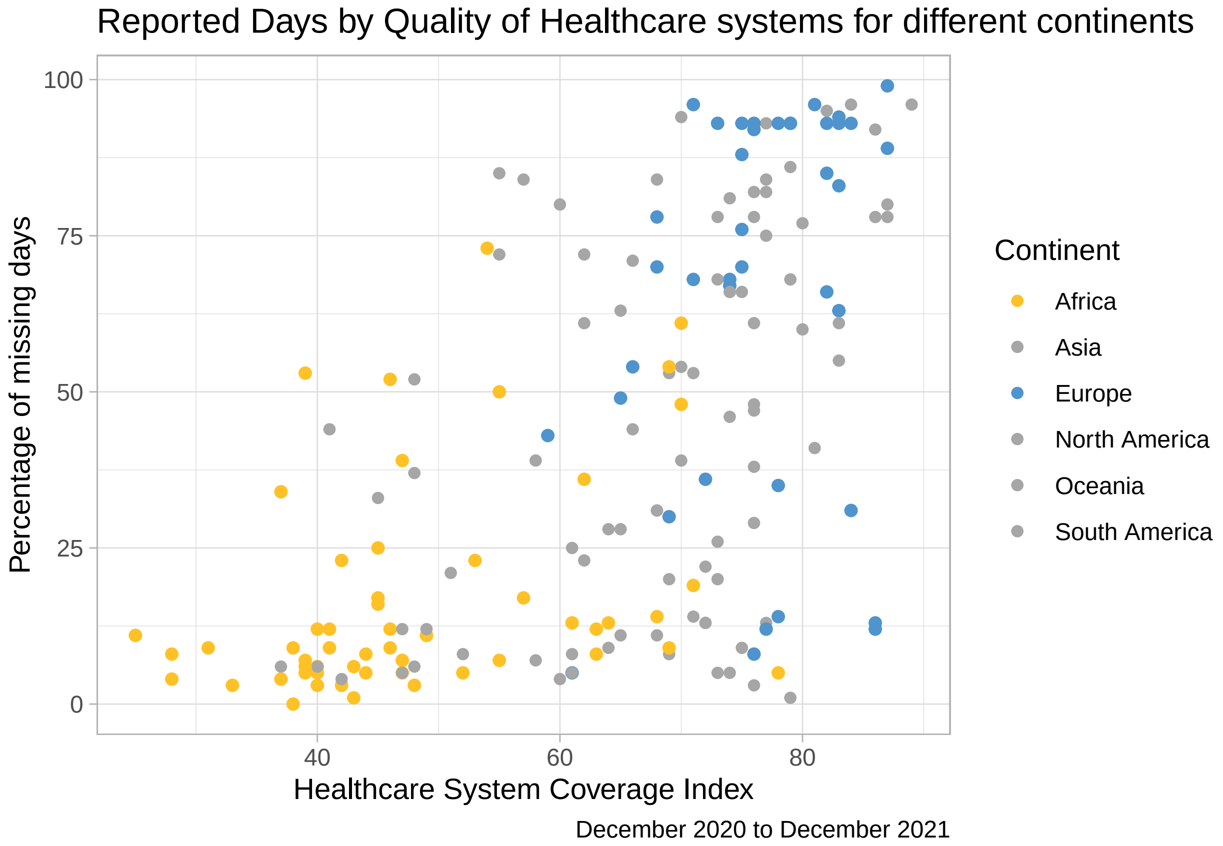
There could be multiple reasons for less reported days:

- Our World in Data did not collect the data
- The data is not reported every day (Sweden, for instance, reports every 7 days as can be seen from the data)
- The data was reported irregularily, there are missing days
- There were no vaccinations on some days (although theoretically, then the country should report a 0)

One possible explanation for not reporting numbers are political reasons. An example here is North Korea (which is not even mentioned in the dataset). In other countries, humanitarian crises and armed conflicts might lead to problems in reporting. This could be (amongst others) the case for Sudan, Venezuela or Yemen.

Another reason could be the healthcare system of a country: It is easy to believe that a bad healthcare system would cause struggles in the reporting of values. Let's see whether I can find a correlation by using the health service coverage index.

```r
na_values_by_countries %>%
  ggplot(aes(x=essential_service_coverage, y=reported_days, color = continent, size = continent)) +
  geom_point() +
  theme_light() +
  labs(title = "Reported Days by Quality of Healthcare systems for different continents",
       color = "Continent",
       caption = "December 2020 to December 2021",
       x = "Healthcare System Coverage Index",
       y = "Percentage of missing days") +
  scale_color_manual(values = c("goldenrod1", "grey65", "steelblue3", "grey65", "grey65", "grey65")) +
  scale_size_manual(values = c(1.7, 1.5, 1.7, 1.5, 1.5, 1.5)) +
  guides(size = "none")
```



As you can see, there are hardly any datapoints in the upper left corner. This shows, that countries with a low value for the Health Service Coverage are indeed reporting less numbers. However, the other way round the assumption is not true: There is no guarantee that countries with a good healthcare system necessarily report more numbers. This is most likely a proof for the above mentioned assumption, that there can be multiple reasons for not reporting numbers.

Since the previous plots showed that the biggest difference between continents could be the one between Europe and Africa, these two continents are highlighted in this plot. As you can see, the healthcare systems are a lot better in Europe than in Africa. Also, there tend to be more reported days for Europe than for Africa. There are more blue (European) points in the upper half of the plot whereas there are more yellow (African) points in the lower half of the plot.

This plot, therefore, leads to the assumption that the Healthcare System indeed has an influence on the reliability of the reporting in a country. However, it also shows that there are most likely other influences.

Dealing with Missing Values

Now I know that there are missing values and how they are distributed across the dataset. But how can I deal with them?

For some features, it does not make sense to replace NA values. This is the case for:

- new_vaccinations
- new_vaccinations_smoothed
- new_vaccinations_smoothed_per_million

These features are daily values and if there is no number for that in the dataset, we cannot know what that number would be. Also, I am not going to replace values in the "smoothed" features since these features already contain some calculations by Our World in Data.

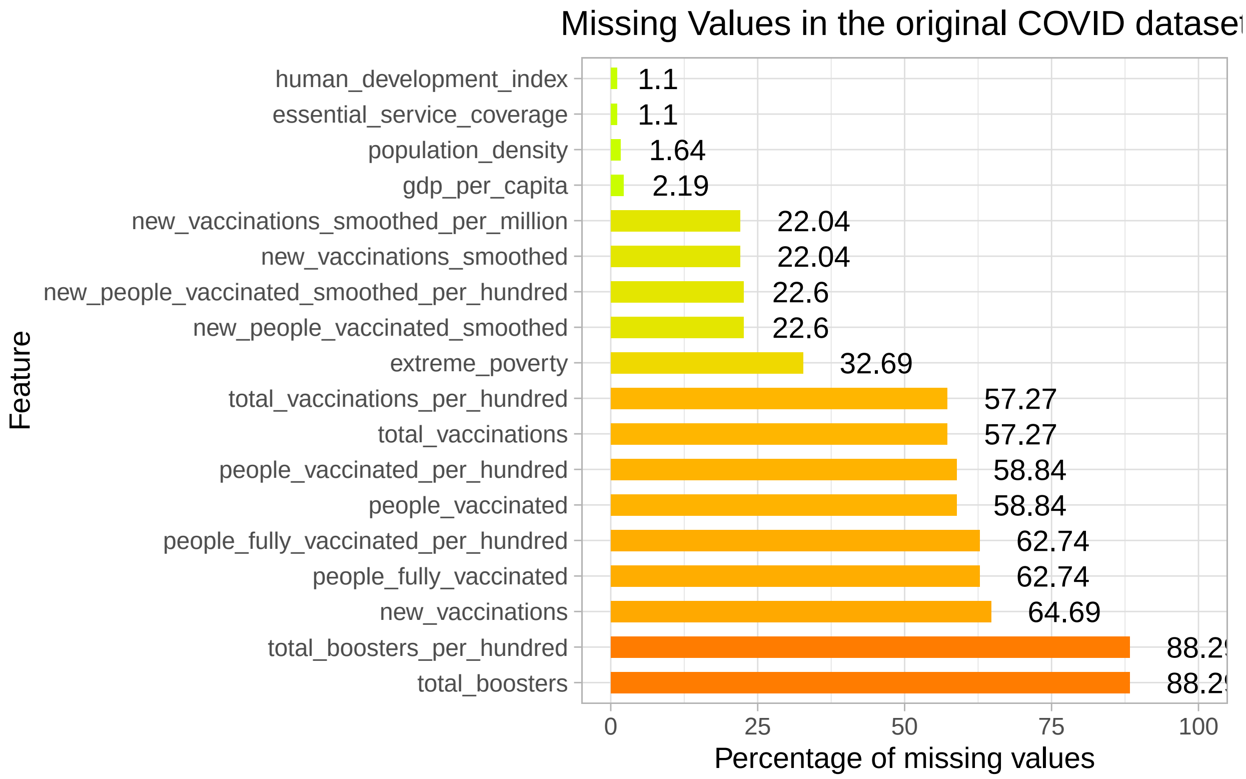
However, for the other vaccination features (e.g. "people_vaccinated" or "total_vaccinations") we have some information from the previous dates. If, for instance, there is no reported data for today, we still know that there are at least as many people vaccinated as yesterday.

Therefore, if a previous value exists, I can simply replace the NA value with that previous value. Implementing this in R is rather complicated, which is why I exported the dataset and edited it with a Python script which I will attach to this project.

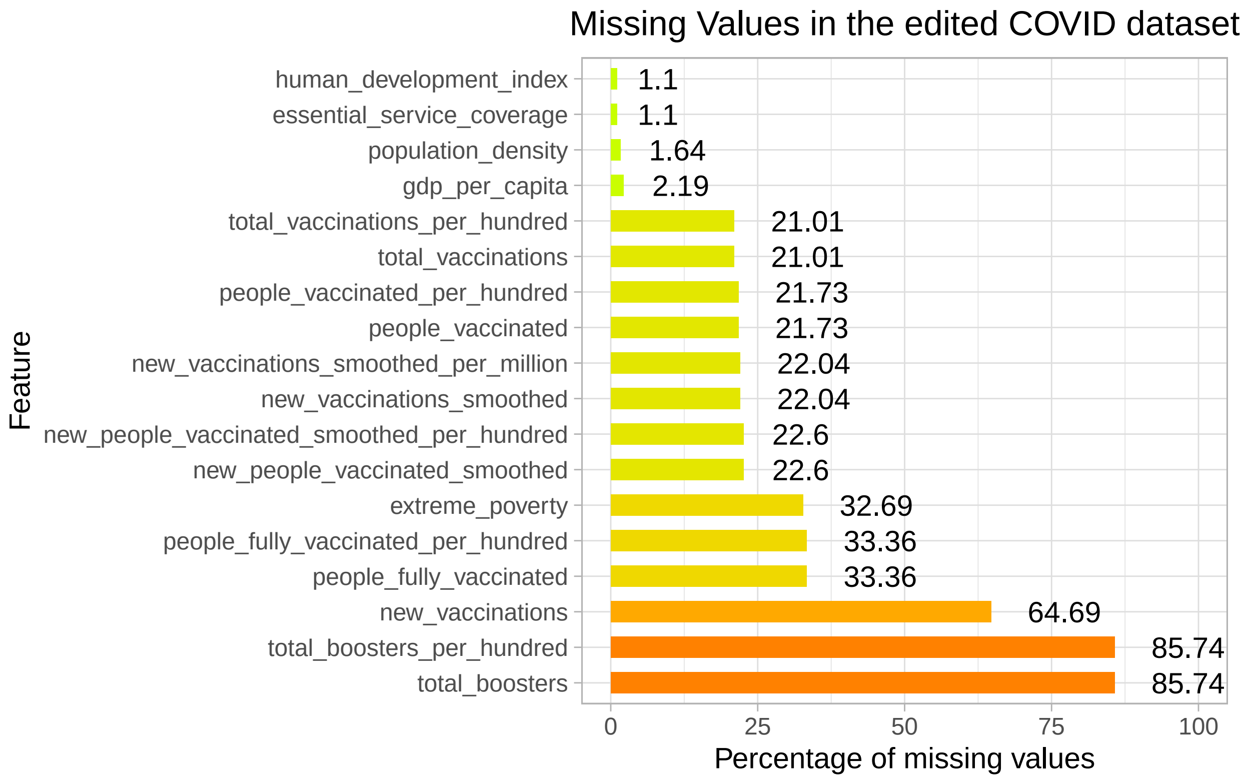
```
write.csv(cov, "cov.csv", row.names = FALSE)
```

```
cov_less_na <- read_csv("cov-less-na.csv")
```

```
rows <- nrow(cov)

missing_values <- cov %>%
  gather(key = "key", value = "val") %>%
  mutate(is_missing = is.na(val)) %>%
  group_by(key, is_missing) %>%
  summarise(num_missing = n()) %>%
  filter(is_missing==T) %>%
  select(-is_missing) %>%
  arrange(desc(num_missing)) %>%
  mutate(percent_missing = round(num_missing / rows * 100, digits = 2))

missing_values$key <- factor(missing_values$key, levels = missing_values$key)

missing_values %>%
  ggplot(aes(x=key, y=percent_missing, fill = percent_missing)) +
  geom_col(width = 0.6) +
  geom_text(aes(label = percent_missing), hjust = -0.5) +
  coord_flip() +
  scale_fill_gradient2(low="#c8ff00", mid = "#ffc300", high = "#ff6500", midpoint = 50) +
  scale_y_continuous(limits = c(0, 100)) +
  theme_light() +
  theme(plot.title = element_text(hjust = 0.5),
        aspect.ratio = 1) +
  labs(title = "Missing Values in the original COVID dataset",
       y = "Percentage of missing values",
       x = "Feature"
  ) +
  guides(fill = "none")
```



```
rows <- nrow(cov)

missing_values_edited <- cov_less_na %>%
  gather(key = "key", value = "val") %>%
  mutate(is_missing = is.na(val)) %>%
  group_by(key, is_missing) %>%
  summarise(num_missing = n()) %>%
  filter(is_missing==T) %>%
  select(-is_missing) %>%
  arrange(desc(num_missing)) %>%
  mutate(percent_missing = round(num_missing / rows * 100, digits = 2))

missing_values_edited$key <- factor(missing_values_edited$key, levels = missing_values_edited$key)

missing_values_edited %>%
  ggplot(aes(x=key, y=percent_missing, fill = percent_missing)) +
  geom_col(width = 0.6) +
  geom_text(aes(label = percent_missing), hjust = -0.5) +
  coord_flip() +
  scale_fill_gradient2(low="#c8ff00", mid = "#ffc300", high = "#ff6500", midpoint = 50) +
  scale_y_continuous(limits = c(0, 100)) +
  theme_light() +
  theme(plot.title = element_text(hjust = 0.5),
        aspect.ratio = 1) +
  labs(title = "Missing Values in the edited COVID dataset",
       y = "Percentage of missing values",
       x = "Feature"
  ) +
  guides(fill = "none")
```

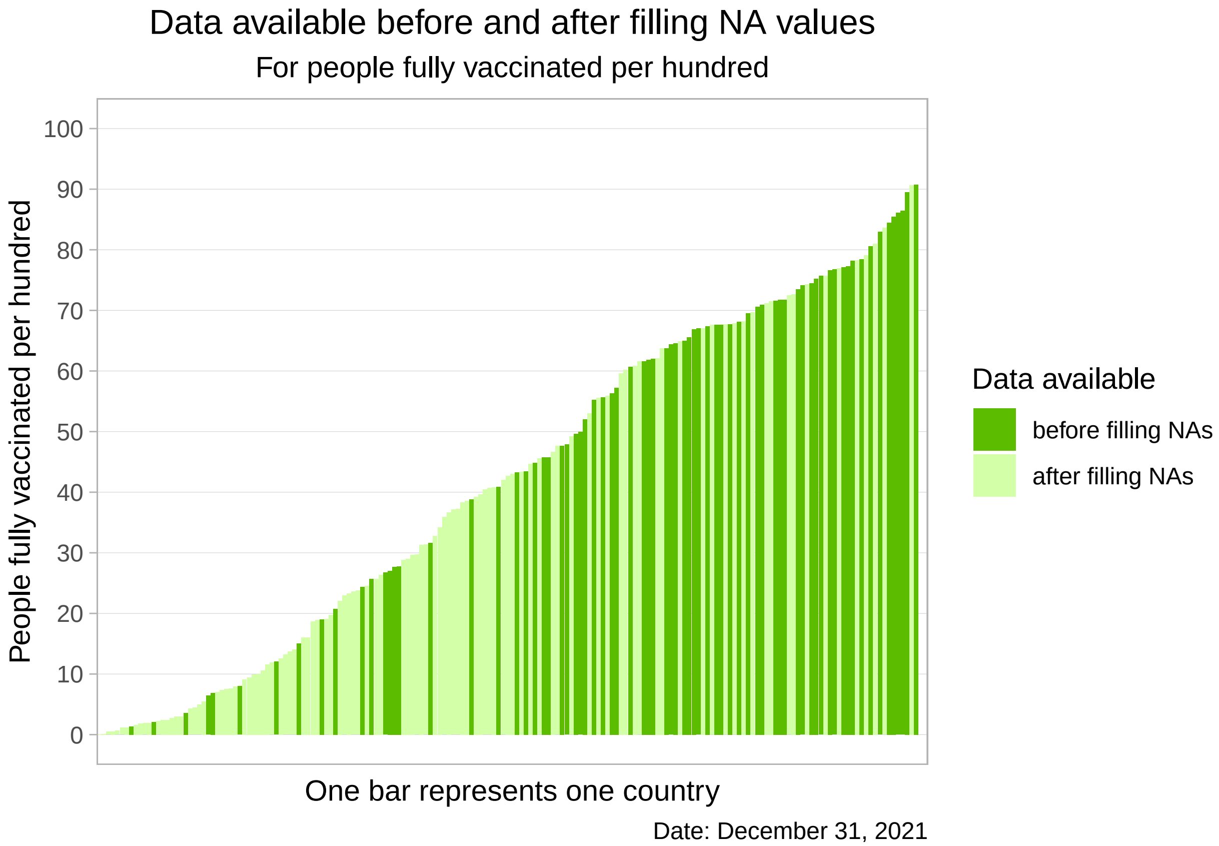


As you can see, the amount of NA values was reduced drastically. For instance, the "total_vaccinations" feature contained 57 % missing values, whereas now it only contains about 21 % missing values.

Let's see how the filling of values changes the vaccination rates for fully vaccinated people on 31st of December:

```
cov_lna_newest <- cov_less_na %>%
  filter(date == "2021-12-31") %>%
  arrange(people_fully_vaccinated_per_hundred)

cov_lna_newest$location <- factor(cov_lna_newest$location, levels = cov_lna_newest$location)

cov_newest <- cov %>%
  filter(date == "2021-12-31") %>%
  arrange(people_fully_vaccinated_per_hundred)

cov_newest$location <- factor(cov_newest$location, levels = cov_newest$location)
```

```
 cov_lna_newest %>%
  ggplot(aes(x=location, y=people_fully_vaccinated_per_hundred)) +
  geom_col(aes(fill = "#d2ffa7")) +
  geom_col(data=cov_newest, aes(fill = "#5cbd00")) +
  theme_light() +
  theme(panel.grid.major.x = element_blank(),
        panel.grid.major.y = element_line(size =.1, color = "lightgrey"),
        panel.grid.minor = element_blank(),
        axis.text.x = element_blank(),
        axis.ticks.x = element_blank(),
        plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5)) +
  scale_y_continuous(limits = c(0, 100), breaks = c(0, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100)) +
  scale_fill_identity(name = 'Data available',
                      guide = 'legend',
                      labels = c("before filling NAs", "after filling NAs")) +
  labs(title = "Data available before and after filling NA values",
       subtitle = "For people fully vaccinated per hundred",
       x = "One bar represents one country",
       y = "People fully vaccinated per hundred",
       caption = "Date: December 31, 2021")
```



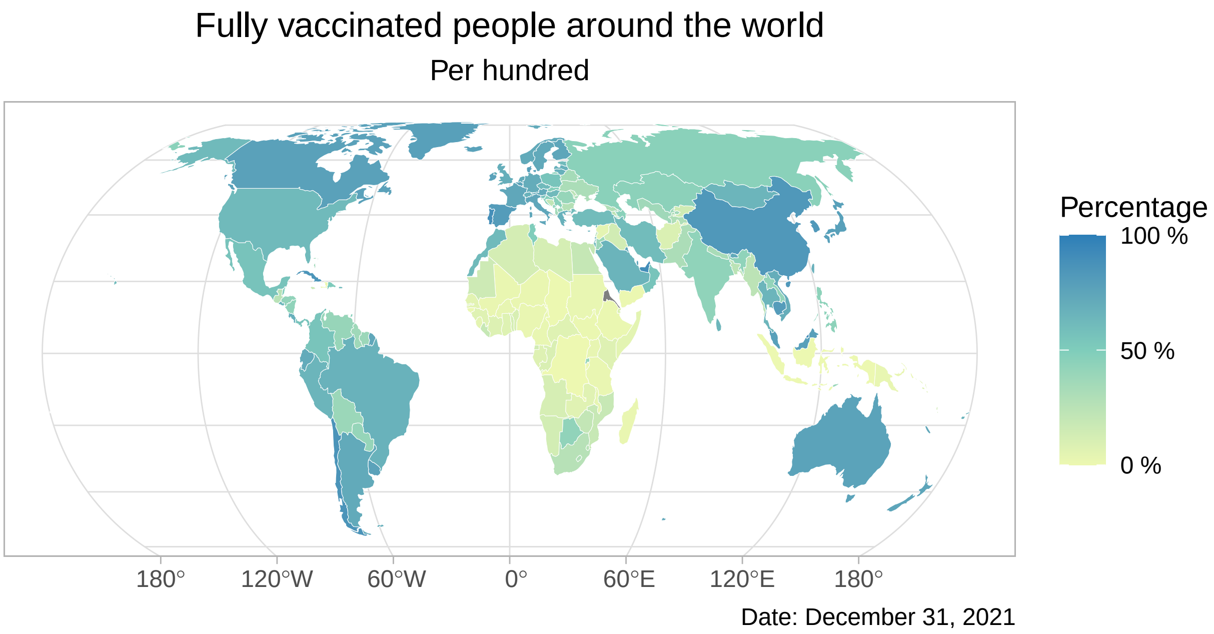
This plot shows several interesting things. First of all, by replacing the NA values we reached a much better availability of data for specific days. We do not anymore have to be lucky and catch the day where a new number of vaccinated people was published by a country, but we will always refer to the latest available number. Additionally, this plot shows that currently, there are still countries with a vaccination rate close to zero and countries with a vaccination rate of over 90 % - a huge difference! Finally, it shows that countries with lower vaccination rates also have more missing days in the dataset (less dark stripes for lower bars!).

The next chapter is going into more detail which countries have a higher or a lower vaccination rate.

Analysis of the vaccination progress in the world

Let's see which countries currently (December 31, 2021) have the biggest share of fully vaccinated people in their population:

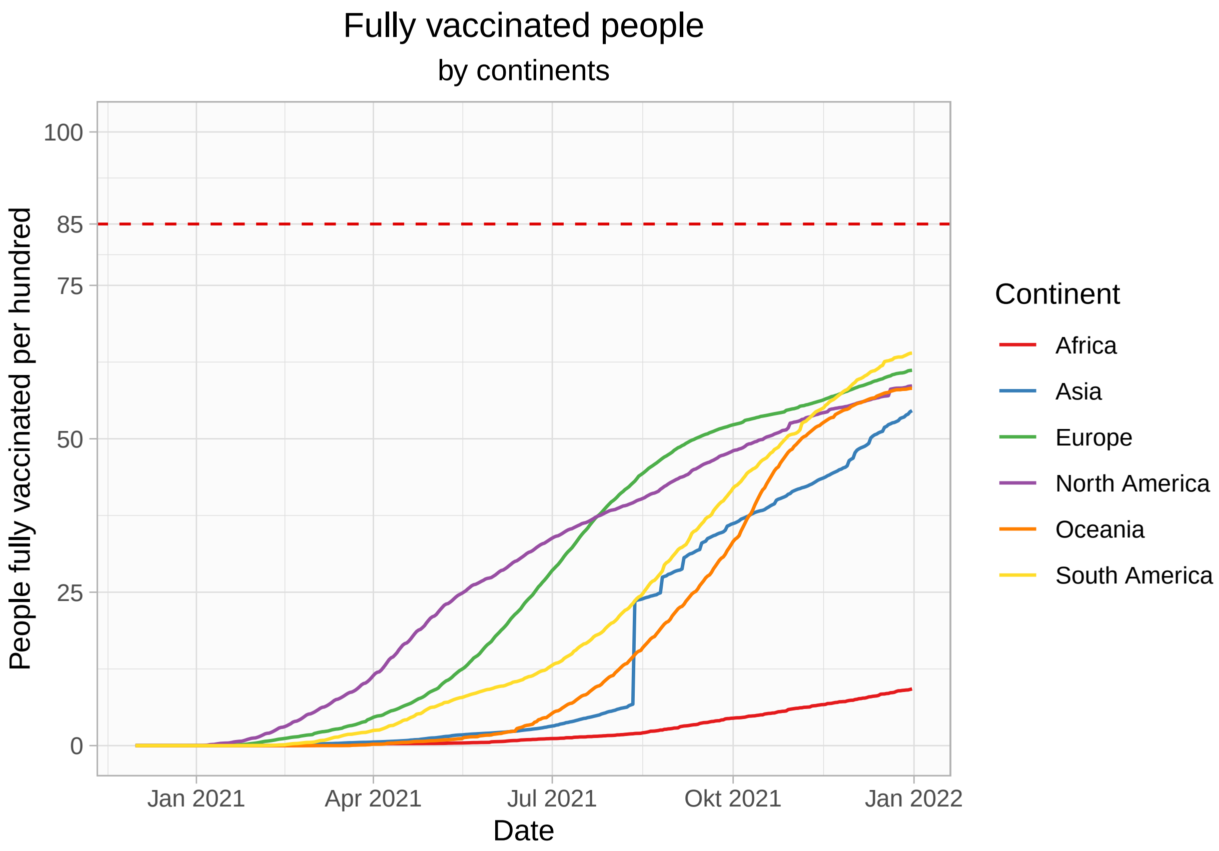
```
cov_less_na_newest <- cov_less_na %>%
  filter(date == "2021-12-31")

geo %>%
  inner_join(cov_less_na_newest, by = "iso_code") %>%
  ggplot(aes(geometry = geometry)) +
  geom_sf(aes(fill = people_fully_vaccinated_per_hundred), color = "white", size = 0.1) +
  labs(title = "Fully vaccinated people around the world",
       subtitle = "Per hundred",
       caption = "Date: December 31, 2021",
       fill = "Percentage") +
  theme_light() +
  theme(plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5)) +
  scale_fill_gradient2(low = "#edf8b1", mid = "#7fcdbb", high = "#2c7fb8", midpoint = 50,
                       limits = c(0, 100),
                       breaks = c(0, 50, 100),
                       labels = c("0 %", "50 %", "100 %"))
```



It is clearly visible that the Americas, Europe, Australia and China have the highest vaccination rate (for fully vaccinated people), whereas in Africa, the vaccination rate is very low.

The following plot shows this in more detail:

```
cov_less_na %>%
  group_by(continent, date) %>%
  summarize(people_fully_vaccinated = sum(people_fully_vaccinated, na.rm = T),
            population = sum(population)) %>%
  mutate(people_fully_vaccinated_percent = (people_fully_vaccinated / population)*100) %>%
  ggplot(aes(x=date, y=people_fully_vaccinated_percent, color = continent)) +
  scale_y_continuous(limits = c(0, 100), breaks = c(0, 25, 50, 75, 85, 100)) +
  geom_hline(yintercept = 85, linetype="dashed", color = "#dd0000") +
  geom_line(size = 0.6) +
  theme_light() +
  theme(plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5),
        panel.background = element_rect(fill = "#fbfbfb")) +
  labs(title = "Fully vaccinated people",
       subtitle = "by continents",
       color = "Continent") +
  xlab("Date") +
  ylab("People fully vaccinated per hundred") +
  scale_colour_manual(values = c("#e41a1c","#377eb8","#4daf4a","#984ea3","#ff7f00","#ffdc29"))
```

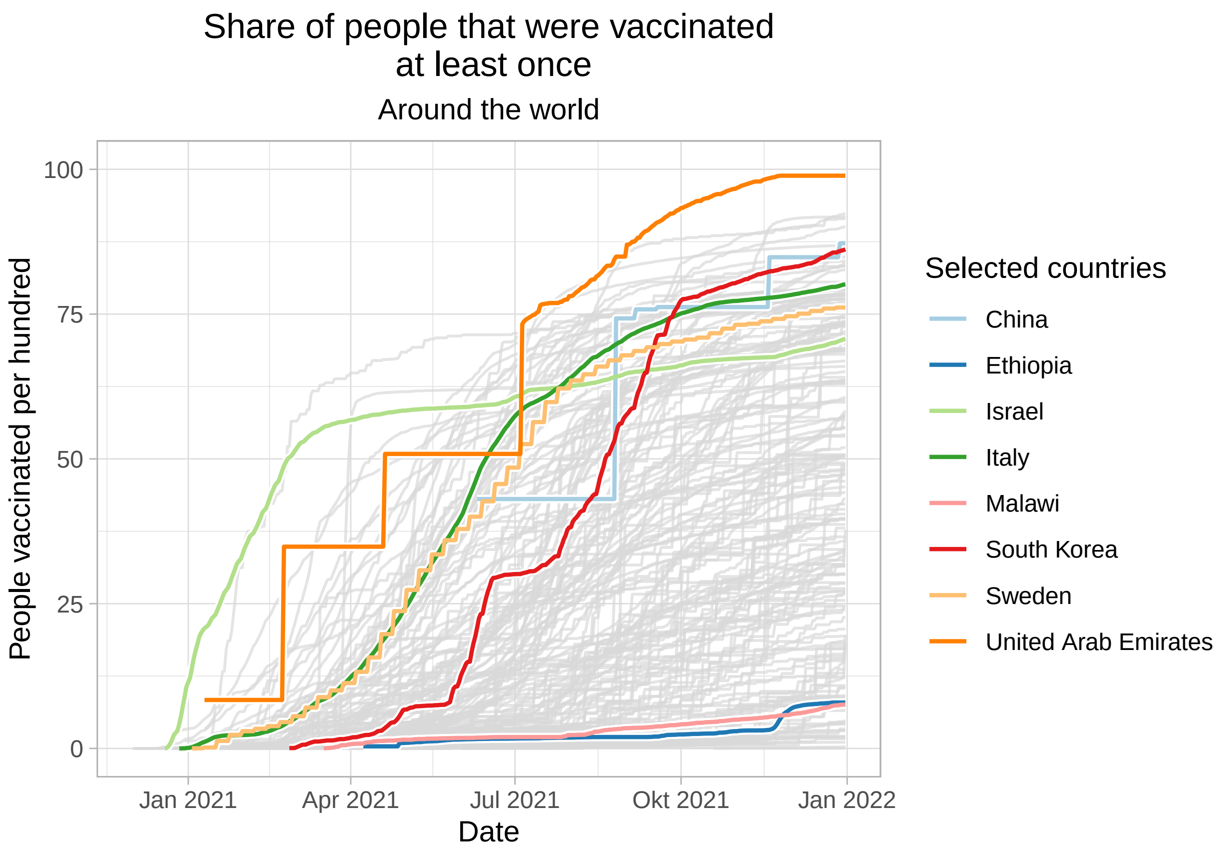


```
# Color scale by "ColorBrewer" - link: https://bl.ocks.org/emeeks/8cdec64ed6daf955830fa723252a4ab3
# Validated if suitable for Color Blindness with "Viz Palette" - link: https://projects.susielu.com/viz
```

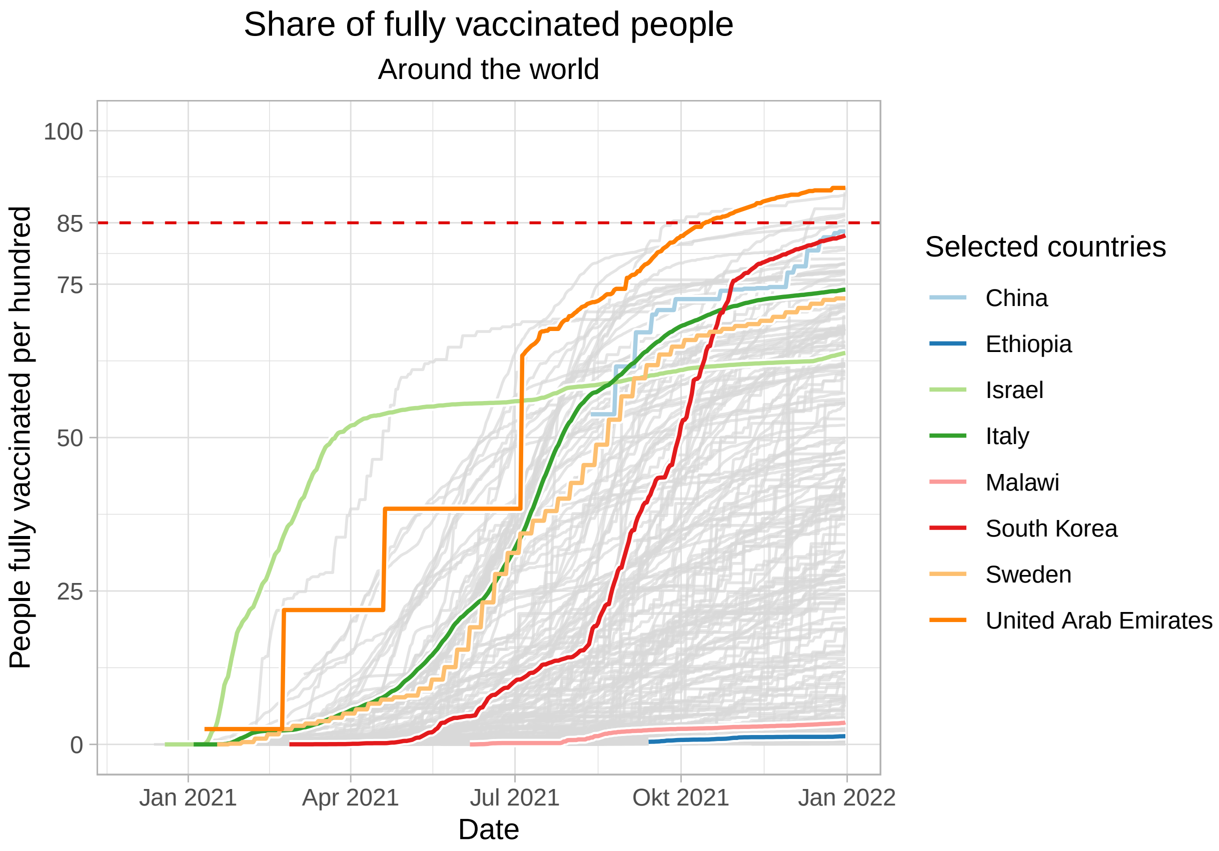
From all continents, South America has the highest average vaccination rate. However, Europe, North America, Oceania and Asia follow soon after. Only in Africa the vaccination rate is much lower at about 10 % (as already indicated in the map). But also the other continents are far from reaching the much discussed herd immunity (shown by the dashed red line). Currently, experts estimate that for the variants Delta and Omikron, a herd immunity of 80 % to 90 % is needed.

Another interesting detail is the big jump in the line of Asia. The following plots will show the reason for this peculiarity, as they give more detail about particularily interesting countries for people that were vaccinated at least once (plot 1) and people that are fully vaccinated (plot 2).

```
of_interest <- c("Italy", "Malawi", "Ethiopia", "Sweden", "Israel", "South Korea", "United Arab Emirate


once_plot <- cov_less_na %>%
  ggplot(aes(x=date, y=people_vaccinated_per_hundred, group = location)) +
  geom_line(color = "gray85", alpha = 0.7, size = 0.5) +
  labs(title = "People fully vaccinated") +
  theme_light()


once_plot +
  geom_line(
    data = function(d) {filter(d, location %in% of_interest)},
    size = 1.5,
    colour = "white"
  ) +
  geom_line(
    data = function(d) {filter(d, location %in% of_interest)},
    aes(color = location),
    size = 0.75
  ) +
  scale_y_continuous(limits = c(0, 100)) +
  theme(plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5)) +
  labs(title = "Share of people that were vaccinated\n at least once",
       subtitle = "Around the world",
       color = "Selected countries",
       y = "People vaccinated per hundred",
       x = "Date") +
  scale_color_manual(values =
                       c("#a6cee3","#1f78b4","#b2df8a","#33a02c","#fb9a99","#e31a1c","#fdbf6f","#ff7f00
```



```
fully_plot <- cov_less_na %>%
  ggplot(aes(x=date, y=people_fully_vaccinated_per_hundred, group = location)) +
  geom_line(color = "gray85", alpha = 0.7, size = 0.5) +
  theme_light()


fully_plot +
  geom_line(
    data = function(d) {filter(d, location %in% of_interest)},
    size = 1.5,
    colour = "white"
  ) +
  geom_line(
    data = function(d) {filter(d, location %in% of_interest)},
    aes(color = location),
    size = 0.75
  ) +
  geom_hline(yintercept = 85, linetype="dashed", color = "#dd0000") +
  scale_y_continuous(limits = c(0, 100), breaks = c(0, 25, 50, 75, 85, 100)) +
  theme(plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5)) +
  labs(title = "Share of fully vaccinated people",
       subtitle = "Around the world",
       color = "Selected countries",
       y = "People fully vaccinated per hundred",
       x = "Date") +
  scale_color_manual(values =
                       c("#a6cee3","#1f78b4","#b2df8a","#33a02c","#fb9a99","#e31a1c","#fdbf6f","#ff7f00
```



```
# Color scale by "ColorBrewer" - link: https://bl.ocks.org/emeeks/8cdec64ed6daf955830fa723252a4ab3
# Validated if suitable for Color Blindness with "Viz Palette" - link: https://projects.susielu.com/viz
```

The two plots above show the vaccination progress for every country in the world. The first plot shows the share of people that were vaccinated at least once, the second plot shows only fully vaccinated people.

As you can see, the trends in the two plots are pretty similar, which shows that the countries followed the principle to vaccinate people twice as fast as possible.

Only very few countries reached the threshold for herd immunity so far. In both plots, I highlighted some particularily interesting countries:

- *China*: Here we see the reason for the jump in vaccinations in Asia in Summer 2021. China reported a jump in vaccination rates from about 40 % to 75 %. As we already saw in the analysis of missing values, China did not report numbers very regularily or often. Therefore, of course, the numbers are also not really transparent.

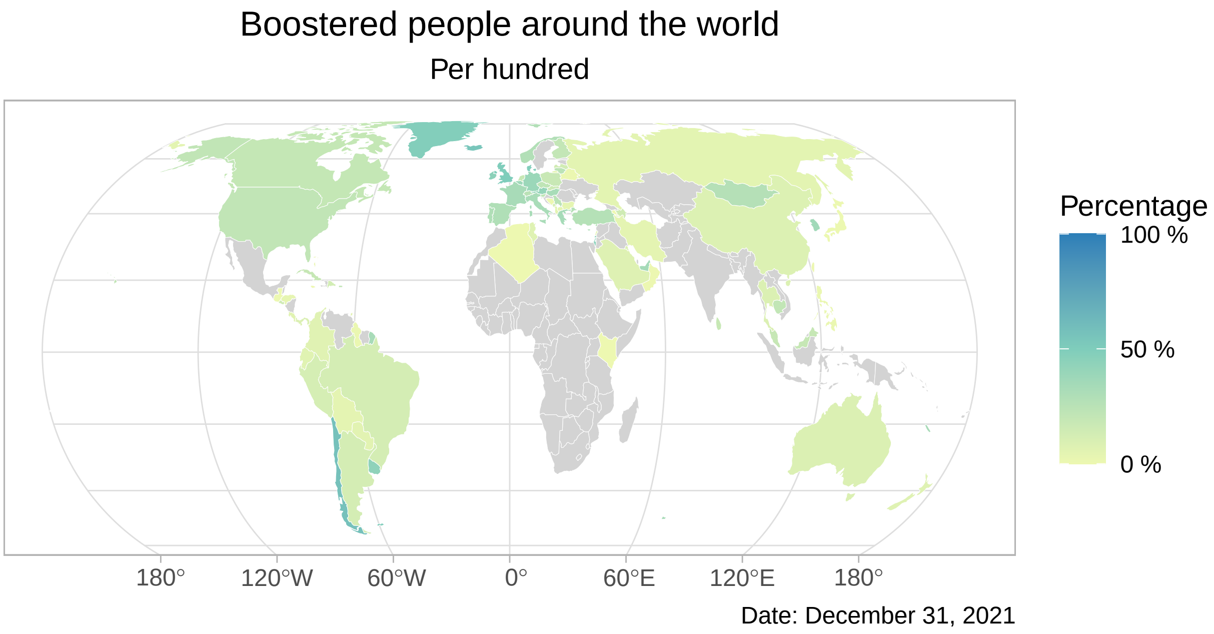
- The *United Arab Emirates* claimed in November to be the first country in which the whole population has been vaccinated at least once. In this dataset, for the UAE, the share of people vaccinated at least once on December 31, 2021 is 98 %. Like this, they are the country with the largest share of population that has been vaccinated at least once. Also regarding fully vaccinated people, the United Arab Emirates are top three in the world.
- *South Korea* also has a very high vaccination rate. Reports state that this is due to a very high vaccination readiness caused by the previous SARS and MERS epidemics.

- For *Sweden*, one can see from the curve that the numbers are reported only every 7 days, which is why the line is not smooth but looks like stairs.
- *Israel* was the first country to start a vaccination campaign, as the plot shows. However, since April, the country apparently did not make a lot of process.
- The countries *Ethiopia* and *Malawi* show again how far behind Africa is with its vaccination campaign.
- I added *Italy* as a reference.

Booster vaccination progress

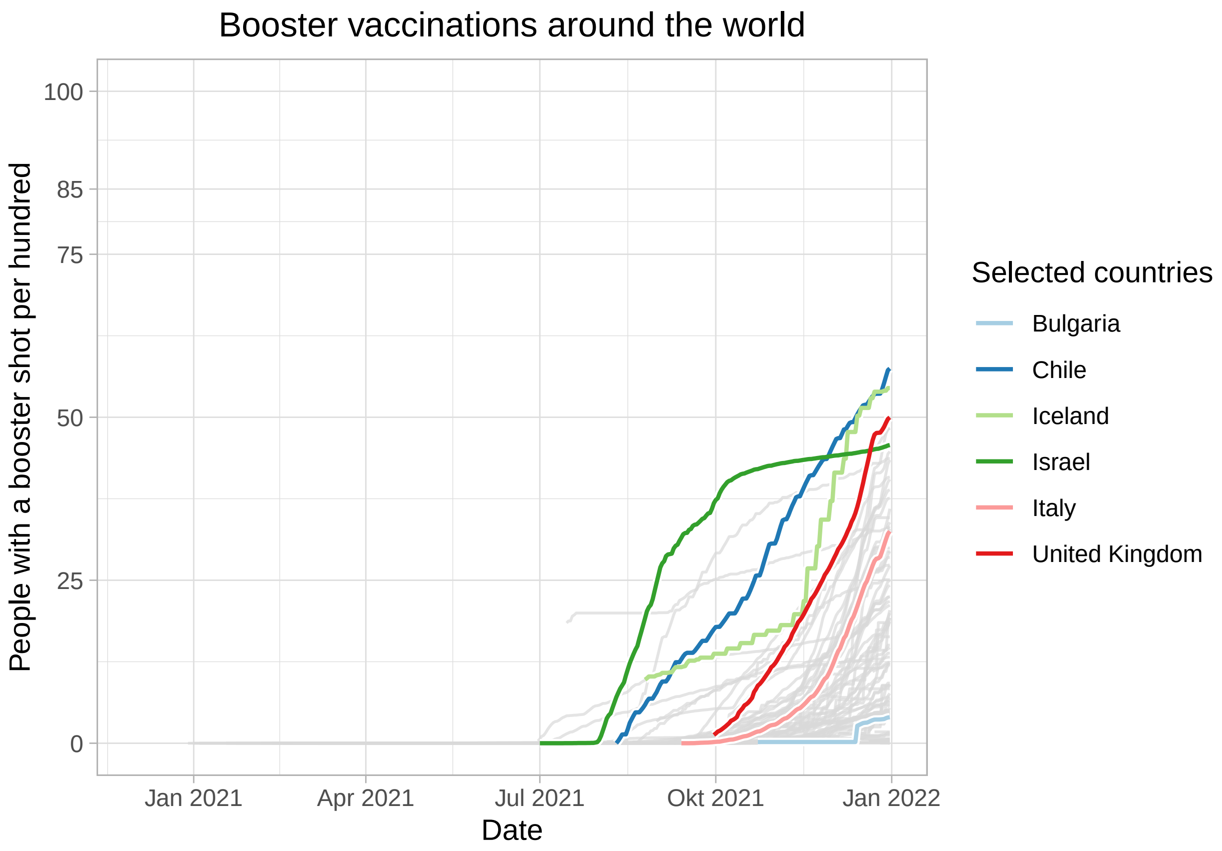
In this chapter, I will look at the available data for booster vaccinations. As we saw above, there is not yet a lot of data available, since many countries just started their campaign for booster vaccinations.

```
geo %>%
  inner_join(cov_less_na_newest, by = "iso_code") %>%
  ggplot(aes(geometry = geometry)) +
  geom_sf(aes(fill = total_boosters_per_hundred), color = "white", size = 0.1) +
  labs(title = "Boostered people around the world",
       subtitle = "Per hundred",
       fill = "Percentage",
       caption = "Date: December 31, 2021") +
  theme_light() +
  theme(plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5)) +
  scale_fill_gradient2(low = "#edf8b1", mid = "#7fcdbb", high = "#2c7fb8",
                       na.value = "lightgrey",
                       midpoint = 50,
                       limits = c(0, 100),
                       breaks = c(0, 50, 100),
                       labels = c("0 %", "50 %", "100 %"))
```



The map for Booster vaccinations looks (as expected) very different from the one for twice vaccinated people. There is no data available (at all) for most African countries and also many countries in Asia.

```
of_interest <- c("Chile", "Iceland", "United Kingdom", "Italy", "Bulgaria", "Israel")
fully_plot <- cov_less_na %>%
  ggplot(aes(x=date, y=total_boosters_per_hundred, group = location)) +
  geom_line(color = "gray85", alpha = 0.7, size = 0.5) +
  theme_light()


fully_plot +
  geom_line(
    data = function(d) {filter(d, location %in% of_interest)},
    size = 1.5,
    colour = "white"
  ) +
  geom_line(
    data = function(d) {filter(d, location %in% of_interest)},
    aes(color = location),
    size = 0.75
  ) +
  scale_y_continuous(limits = c(0, 100), breaks = c(0, 25, 50, 75, 85, 100)) +
  theme(plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5)) +
  labs(title = "Booster vaccinations around the world",
       color = "Selected countries",
       y = "People with a booster shot per hundred",
       x = "Date") +
  scale_color_manual(values = c("#a6cee3","#1f78b4","#b2df8a","#33a02c","#fb9a99","#e31a1c"))
```



```
# Color scale by "ColorBrewer" - link: https://bl.ocks.org/emeeks/8cdec64ed6daf955830fa723252a4ab3
# Validated if suitable for Color Blindness with "Viz Palette" - link: https://projects.susielu.com/viz

top_5_boosters <- cov_less_na %>%
  filter(date == "2021-12-31") %>%
  select(location, total_boosters_per_hundred) %>%
  arrange(desc(total_boosters_per_hundred)) %>%
  top_n(5) %>%
  rename("Country" = location, "Total boosters per hundred" = total_boosters_per_hundred)

knitr::kable(top_5_boosters, caption = "Top 5 countries by booster vaccinations per hundred")
```

Table 1: Top 5 countries by booster vaccinations per hundred

Country	Total boosters per hundred
Chile	57.48
Iceland	54.48
United Kingdom	49.98
Bahrain	48.34
Denmark	48.30

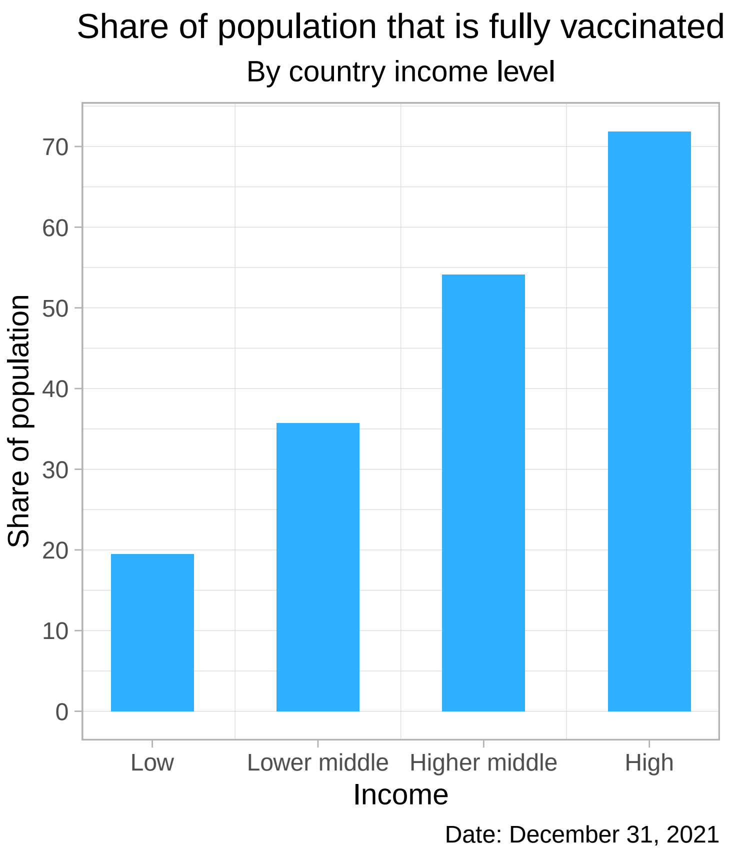
The plot and the table show that most countries just started with booster vaccinations in fall 2021. *Chile* is the country with the highest share of boostered people of more than 57 % in their population. According to Reuters, they are even starting to offer a fourth dose. *Iceland* is number two in the race for booster vaccinations - this might be due to the low number of inhabitants of only about 350.000 people. Another European country is on the third spot - the *United Kingdom*. However, the highlighted trend for *Bulgaria* shows that also in Europe, there are huge differences in booster rates: Bulgaria has just started to administer the third dose. Israel, again, started very early with booster vaccinations but did not manage to reach such a high rate (so far) as other countries. The reason why I did not highlight any *African country* is because there is hardly any data available yet.

Vaccination and GDP

The previous chapter analysed the vaccination progress in general. Now, let's see whether we can find some correlations which might explain the huge differences in vaccination rates across the world. One assumption is that the Gross Domestic Product might influence the availability of vaccines and therefore the vaccination rate in a country. The Gross Domestic Product is commonly accepted as an indicator for the prosperity of a country.

The following plot explores whether the GDP influences the share of the population that is fully vaccinated by December 31, 2021.

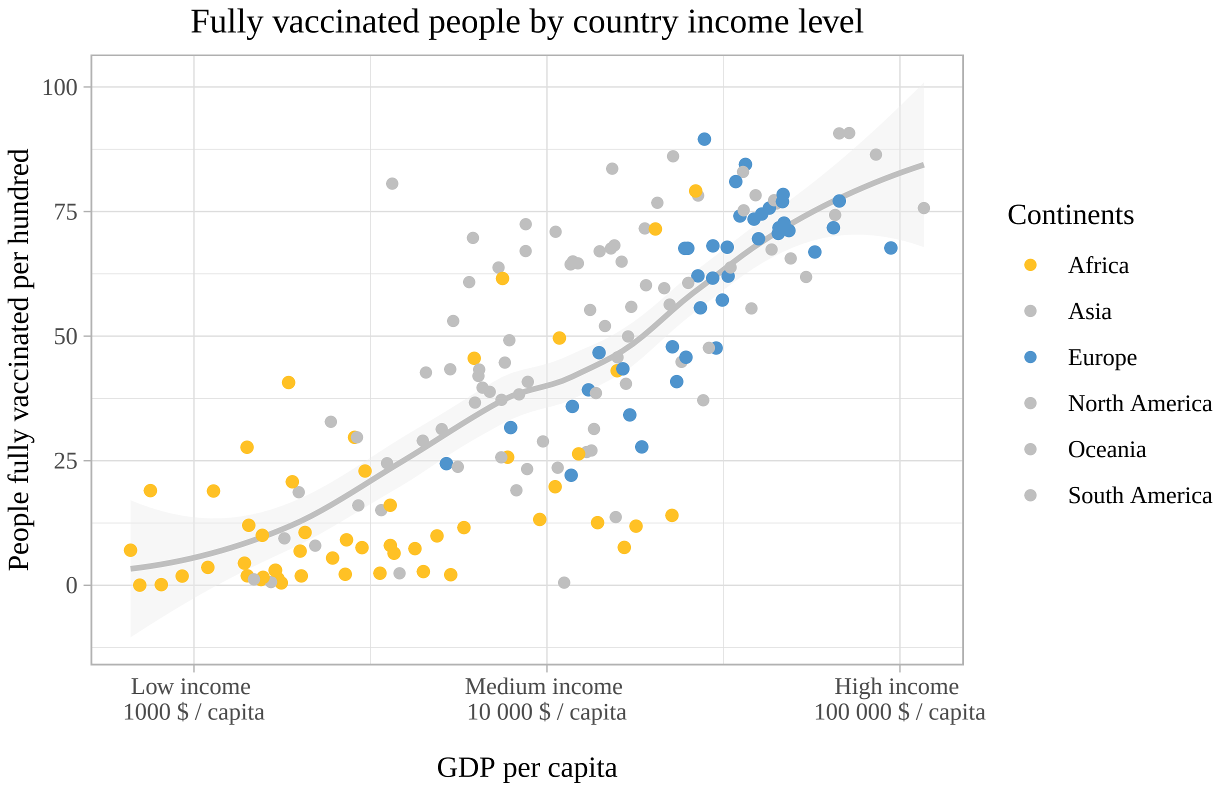
```r
cov %>%
  filter(date == "2021-12-31") %>%
  group_by(location) %>%
  summarize(people_fully_vaccinated_per_hundred =
              mean(people_fully_vaccinated_per_hundred, na.rm = T),
            gdp_per_capita = mean(gdp_per_capita)) %>%
  arrange(gdp_per_capita) %>%
  filter(!(is.na(gdp_per_capita))) %>%
  mutate(quantile = ntile(gdp_per_capita, 4)) %>%
  group_by(quantile) %>%
  summarise(avg_vac = mean(people_fully_vaccinated_per_hundred, na.rm = T)) %>%
  ggplot(aes(x=quantile, y = avg_vac)) +
  geom_col(stat="identity", fill = "#2eafff", width = 0.5) +
  theme_light() +
  theme(aspect.ratio = 1,
        plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5),
        panel.grid.major.x = element_blank(),
        panel.grid.major.y = element_line(size =.1, color = "lightgrey")) +
  labs(title = "Share of population that is fully vaccinated",
       subtitle = "By country income level",
       x = "Income",
       y = "Share of population",
       caption = "Date: December 31, 2021") +
  scale_x_continuous(breaks = c(1,2,3,4),
                     labels = c("Low", "Lower middle", "Higher middle", "High")) +
  scale_y_continuous(breaks = c(0, 10, 20, 30, 40, 50, 60, 70))
```



The plot shows a clear picture. The countries which are in the first quantile (the poorest 25 %) have an average share of only close to 20 people per 100 that are fully vaccinated, whereas the richest 25 % of countries have an average vaccination rate of over 70 %.

The following plot shows this in more detail.

```
cov_less_na %>%
  filter(date == "2021-12-31") %>%
  ggplot() +
  geom_smooth(aes(x= gdp_per_capita,
                  y= people_fully_vaccinated_per_hundred),
              color = "grey75",
              fill = "grey93") +
  geom_point(aes(x = gdp_per_capita,
                 y = people_fully_vaccinated_per_hundred,
                 color = continent,
                 size = continent)) +
  scale_x_log10(breaks = c(1000, 10000, 100000),
                labels = c("Low income \n 1000 $ / capita \n",
                           "Medium income \n 10 000 $ / capita \n",
                           "High income \n 100 000 $ / capita \n")) +
  theme_light() +
  theme(plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5),
        text = element_text(family = "serif")) +
  labs(title = "Fully vaccinated people by country income level",
       x = "GDP per capita",
      y = "People fully vaccinated per hundred",
      caption = "GDP per capita = Gross Domestic Product per capita in constant 2011 international dol
      color = "Continents") +
  scale_color_manual(values = c("goldenrod1", "grey75", "steelblue3", "grey75", "grey75", "grey75")) +
  scale_size_manual(values = c(1.7, 1.5, 1.7, 1.5, 1.5, 1.5)) +
  guides(size = "none")
```

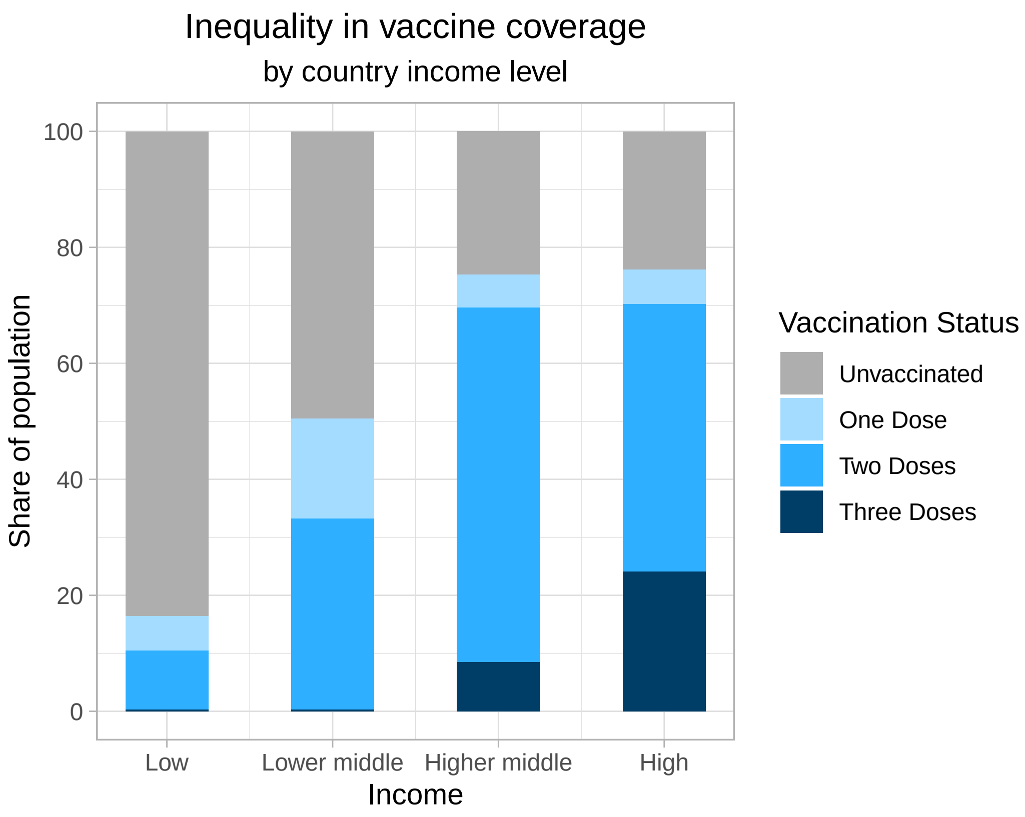


GDP per capita = Gross Domestic Product per capita in constant 2011 international dollars
Date: December 31, 2021

As you can see from the grey line and the distribution of the dots, there is a clear correlation between the GDP of a country and the share of people that is fully vaccinated. African countries are cluttered in the section with low / medium income and a low / medium vaccination rate, whereas European countries show mostly in the medium / high income and medium / high vaccination rate corner.

Let's refine this picture even more by including the administered first doses, second doses and booster doses.

```
cov_less_na %>%
  filter(date == "2021-12-31") %>%
  arrange(gdp_per_capita) %>%
  filter(!(is.na(gdp_per_capita))) %>%
  mutate(quantile = ntile(gdp_per_capita, 4)) %>%
  group_by(quantile) %>%
  summarise(
    people_vaccinated = sum(people_vaccinated, na.rm = T),
    people_fully_vaccinated = sum(people_fully_vaccinated, na.rm = T),
    total_boosters = sum(total_boosters, na.rm = T),
    population = sum(population, na.rm = T)) %>%
  transmute(
    boostered_people = total_boosters,
    doubly_vaccinated_people = people_fully_vaccinated - total_boosters,
    once_vaccinated_people = people_vaccinated - people_fully_vaccinated,
    unvaccinated_people = population - boostered_people - doubly_vaccinated_people
    - once_vaccinated_people,
    population = population,
    quantile = quantile) %>%
  pivot_longer(cols = boostered_people:unvaccinated_people,
               names_to = "vaccination_status",
               values_to = "People") %>%
  ggplot(aes(x=quantile, y=People, fill = forcats::fct_rev(vaccination_status))) +
  geom_col(stat="identity", width = 0.5, position = "fill") +
  theme_light() +
  theme(aspect.ratio = 1,
        plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5)) +
  scale_x_continuous(breaks = c(1,2,3,4),
                     labels = c("Low", "Lower middle", "Higher middle", "High")) +
  scale_y_continuous(breaks = c(0, 0.2, 0.4, 0.6, 0.8, 1.0),
                     labels = c(0, 20, 40, 60, 80, 100)) +
  labs(title = "Inequality in vaccine coverage",
       subtitle = "by country income level",
       x = "Income",
       y = "Share of population",
       fill = "Vaccination Status",
       caption = "Date: December 31, 2021") +
  scale_fill_manual(values=c("#aeaeae", "#a4dcff", "#2eafff", "#003d67"),
                    labels = c("Unvaccinated", "One Dose", "Two Doses", "Three Doses"))
```

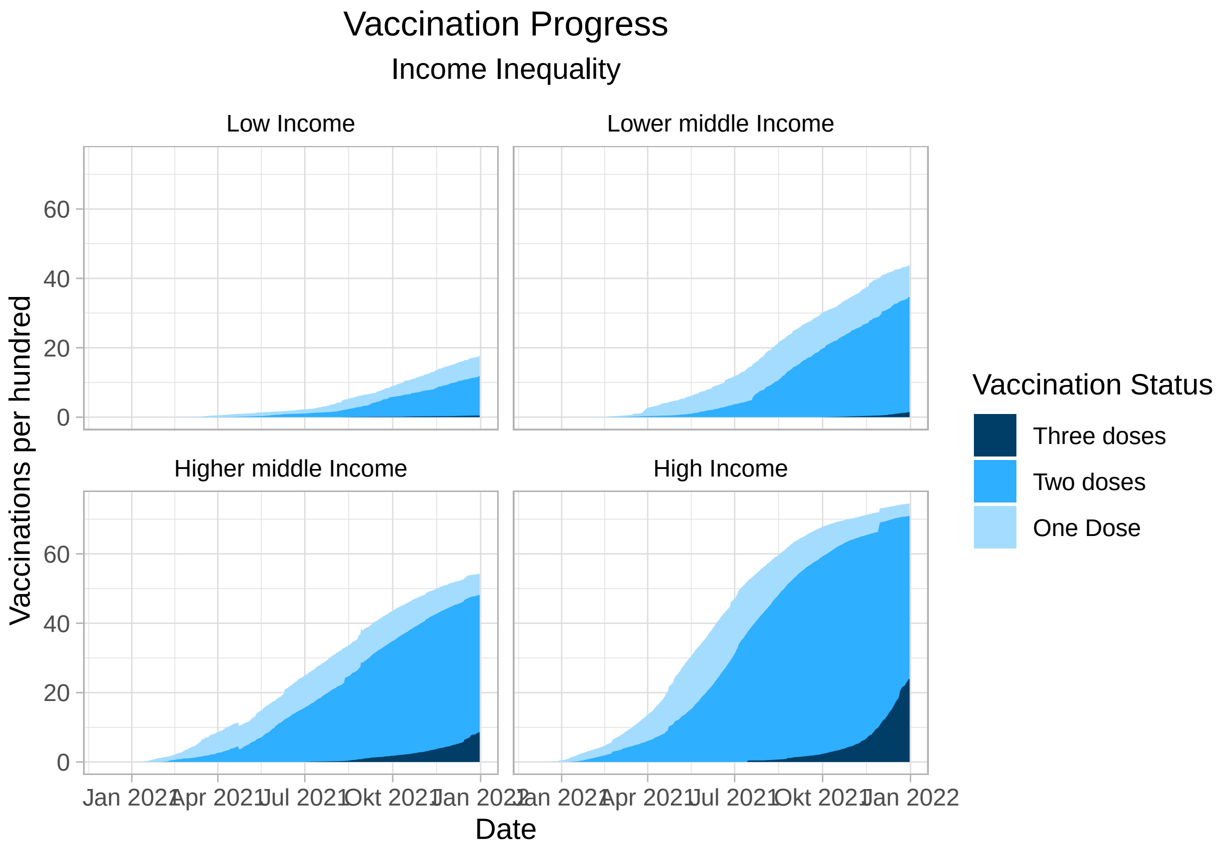


Date: December 31, 2021

This plot shows that, by December 31, 2021, the richest countries have more boostered people in their population than the poorest countries once-vaccinated people!
The poorer half of the countries administered hardly any booster dose so far.

The following plot shows the progress the countries made since the beginning of the vaccination campaign in the world, split by income level.

```
sum_quantile <- cov_less_na %>%
  arrange(gdp_per_capita) %>%
  filter(!(is.na(gdp_per_capita))) %>%
  mutate(quantile = ntile(gdp_per_capita, 4)) %>%
  group_by(quantile, date) %>%
  replace_na(list(people_vaccinated_per_hundred=0, people_fully_vaccinated_per_hundred = 0, total_booste
  summarise(
    people_vaccinated_per_hundred = mean(people_vaccinated_per_hundred, na.rm = T),
    people_fully_vaccinated_per_hundred = mean(people_fully_vaccinated_per_hundred, na.rm = T),
    total_boosters_per_hundred = mean(total_boosters_per_hundred, na.rm = T)) %>%
  arrange(quantile)

labels <- c("1" = "Low Income", "2" ="Lower middle Income", "3" = "Higher middle Income", "4" = "High In

sum_quantile %>%
  ggplot() +
  facet_wrap(vars(quantile), labeller = labeller(quantile = labels)) +
  geom_area(aes(x=date, y=people_vaccinated_per_hundred, fill = "#a4dcff")) +
  geom_area(aes(x=date, y=people_fully_vaccinated_per_hundred, fill = "#2eafff")) +
  geom_area(aes(x=date, y=total_boosters_per_hundred, fill = "#003d67")) +
  theme_light() +
  theme(plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5),
        strip.background =element_rect(fill="white"),
        strip.text = element_text(colour = 'black')) +
  labs(title = "Vaccination Progress",
       subtitle = "Income Inequality",
       x = "Date",
       y = "Vaccinations per hundred") +
  scale_fill_identity(name = 'Vaccination Status',
                    guide = 'legend',
                    labels = c("Three doses", "Two doses", "One Dose"))
```



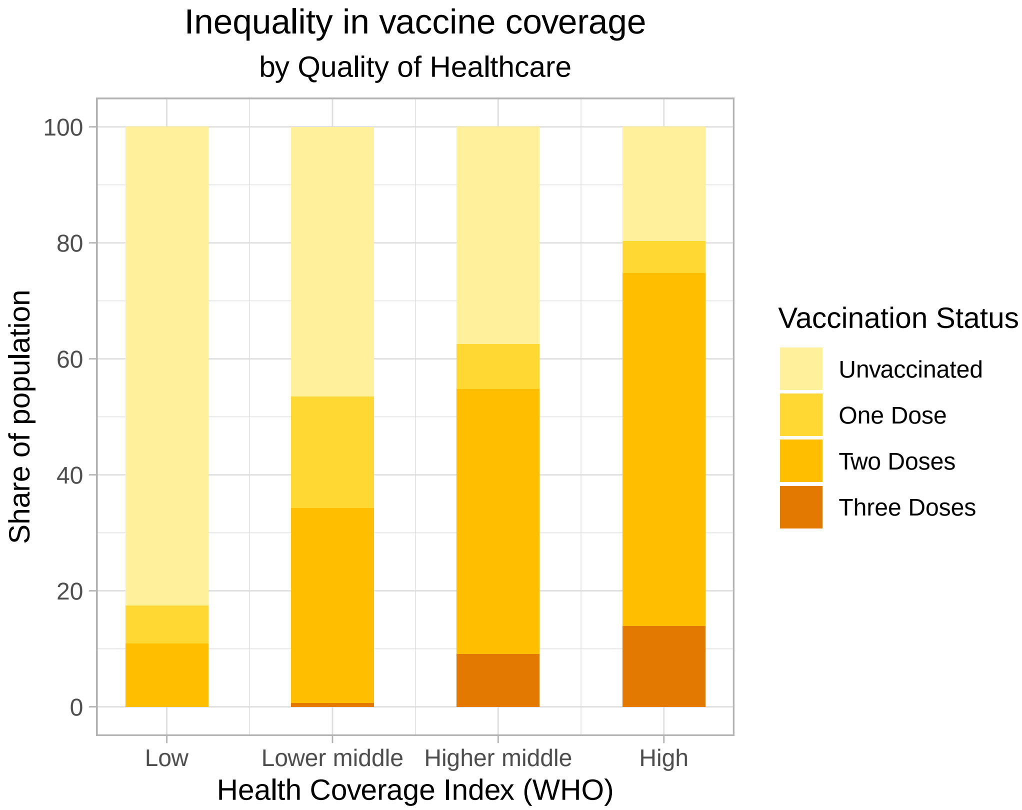
Here, you can see several interesting things:

- The poorer countries started their vaccination campaigns several months after the richer countries.
- The pace of the vaccinations was a lot higher in the richer countries, which is visible from the steepness of the curve.
- However, also for the richer countries, the pace slowed down in the last months. This is probably caused by an unwillingness to get vaccinated in the population.

Vaccination and Quality of Healthcare

Another factor that might influence the success of a vaccination campaign is the Quality of Healthcare in a country. States with a bad healthcare system might, for instance, struggle to organize a fast and efficient vaccination campaign.

```r
quantiles_health <- cov_less_na %>%
  filter(date == "2021-12-31") %>%
  arrange(essential_service_coverage) %>%
  filter(!(is.na(essential_service_coverage))) %>%
  mutate(quantile = ntile(essential_service_coverage, 4)) %>%
  group_by(quantile) %>%
  summarise(
    people_vaccinated = sum(people_vaccinated, na.rm = T),
    people_fully_vaccinated = sum(people_fully_vaccinated, na.rm = T),
    total_boosters = sum(total_boosters, na.rm = T),
    population = sum(population, na.rm = T)) %>%
  transmute(
    boostered_people = total_boosters,
    doubly_vaccinated_people = people_fully_vaccinated - total_boosters,
    once_vaccinated_people = people_vaccinated - people_fully_vaccinated,
    unvaccinated_people = population - boostered_people - doubly_vaccinated_people
    - once_vaccinated_people,
    population = population,
    quantile = quantile) %>%
  pivot_longer(cols = boostered_people:unvaccinated_people,
               names_to = "vaccination_status",
               values_to = "People")

quantiles_health %>%
  ggplot(aes(x=quantile, y=People, fill = forcats::fct_rev(vaccination_status))) +
  geom_col(stat="identity", width = 0.5, position = "fill") +
  theme_light() +
  theme(aspect.ratio = 1,
        plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5)) +
  scale_x_continuous(breaks = c(1,2,3,4),
                     labels = c("Low", "Lower middle", "Higher middle", "High")) +
  scale_y_continuous(breaks = c(0, 0.2, 0.4, 0.6, 0.8, 1.0),
                     labels = c(0, 20, 40, 60, 80, 100)) +
  labs(title = "Inequality in vaccine coverage",
       subtitle = "by Quality of Healthcare",
       x = "Health Coverage Index (WHO)",
       y = "Share of population",
       fill = "Vaccination Status",
       caption = "Date: December 31, 2021") +
  scale_fill_manual(values=c("#fff09c", "#ffd833", "#ffbf00", "#e47900"),
                    labels = c("Unvaccinated", "One Dose", "Two Doses", "Three Doses"))
```



Date: December 31, 2021

Unsurprisingly, this plot shows a similar picture as with the GDP as an influencing variable. Countries with a low Health Coverage Index have a much lower vaccination rate than countries with a high Health Coverage Index. Whereas only close to 20 % of the population in countries with a low index are fully vaccinated, in countries with a high index it is 80 %.

The following code block displays the shares for countries with the lowest Health Coverage Index and the highest Health Coverage Index.

```
table_health <- quantiles_health %>%
  mutate(share = round(People / population * 100, digits = 2)) %>%
  select(quantile, vaccination_status, share) %>%
  rename("Quantile" = quantile,
         "Vaccination Status" = vaccination_status,
         "Share" = share)

knitr::kable(table_health, caption = "Share of Population that is vaccined by Quantiles of the Health Se
```

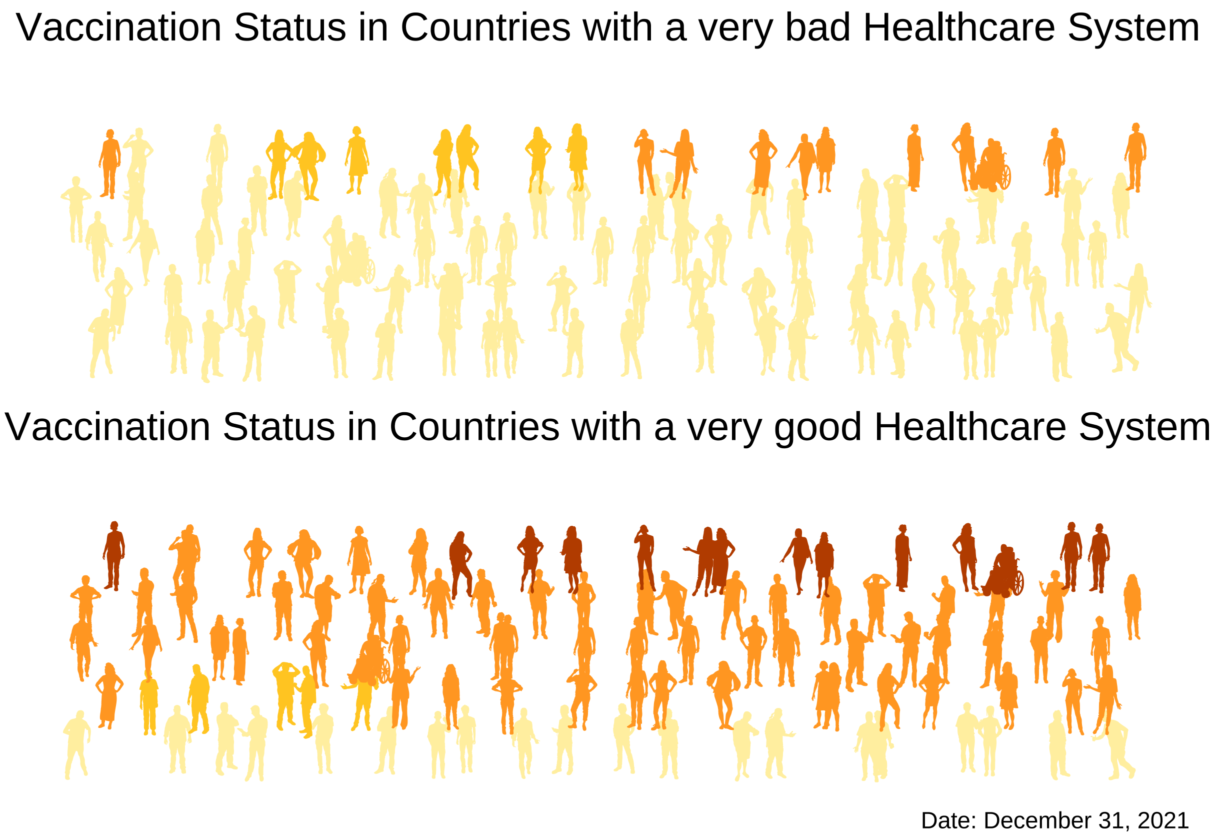
Table 2: Share of Population that is vaccined by Quantiles of the Health Service Index

Quantile	Vaccination Status	Share
1	boostered_people	0.00
1	doubly_vaccinated_people	10.91
1	once_vaccinated_people	6.55
1	unvaccinated_people	82.54
2	boostered_people	0.65
2	doubly_vaccinated_people	33.60

Quantile	Vaccination Status	Share
2	once_vaccinated_people	19.31
2	unvaccinated_people	46.44
3	boostered_people	9.07
3	doubly_vaccinated_people	45.78
3	once_vaccinated_people	7.73
3	unvaccinated_people	37.42
4	boostered_people	13.94
4	doubly_vaccinated_people	60.90
4	once_vaccinated_people	5.45
4	unvaccinated_people	19.70

The following plot displays these numbers. It reuses the color scale of the previous plot.

```
# Download the "WeePeople" font and install it by double clicking on the ttf file
# Link: https://github.com/propublica/weepeople/blob/master/weepeople.ttf
# Change the user name to your user name on your PC

user_name <- "Janas Laptop"
font_add("weepeople", regular = paste("C:\\Users\\", user_name,  "\\AppData\\Local\\Microsoft\\Windows\

showtext_auto()
letters <- "ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz"

poorest <- tibble(
  id = 1:100,
  color = c(rep("#ffee9f", 82), rep("#ffc421", 7), rep("#ff9621", 11))) %>%
rowwise() %>%
mutate(l_idx = (id %% str_length(letters)) + 1,
       l = substr(letters, l_idx, l_idx),
       y = ((id - 1)%/% 20) - runif(1, 0, 0.2),
       x = (id %% 20) -  runif(1, 0, 1.0)) %>%
ggplot(aes(x = x, y = y, label = l, color = color)) +
  geom_text(family="weepeople", size=13) +
  scale_color_identity() +
  theme_void() +
  labs(title = "Vaccination Status in Countries with a very bad Healthcare System") +
  theme(aspect.ratio = 0.3,
        plot.title = element_text(hjust = 0.5, size = 15)) +
  scale_y_continuous(limits=c(-1, 6))

richest <- tibble(
  id = 1:100,
  color = c(rep("#ffee9f", 20), rep("#ffc421", 5), rep("#ff9621", 61), rep("#b03b00", 14))) %>%
rowwise() %>%
mutate(l_idx = (id %% str_length(letters)) + 1,
       l = substr(letters, l_idx, l_idx),
       y = ((id - 1)%/% 20) - runif(1, 0, 0.2),
       x = (id %% 20) -  runif(1, 0, 1.0)
       ) %>%
ggplot(aes(x = x, y = y, label = l, color = color)) +
  geom_text(family="weepeople", size=13) +
  scale_color_identity() +
  theme_void() +
  labs(title = "Vaccination Status in Countries with a very good Healthcare System",
       caption = "Date: December 31, 2021") +
  theme(aspect.ratio = 0.3,
        plot.title = element_text(hjust = 0.5, size = 15)) +
  scale_y_continuous(limits=c(-1, 6))

wrap_plots(poorest, richest, nrow = 2)
```



The plot shows 100 representative people in the 25 % countries with the worst and the 25 % countries with the best healthcare system in the world. This shows vividly the huge difference that the quality of healthcare makes. Most people in the countries with a bad health service coverage are unvaccinated, only some have been vaccinated once or twice. Not a single person received a booster dose. In the countries with a good health service coverage, the picture is very different. A large number of people is fully vaccinated with two doses and 14 people have already received a booster dose.

Conclusion

This project analyzes the vaccination status and progress across the world. It shows clearly the large inequality in vaccine distribution. Countries that are richer and have a better healthcare system have a much higher vaccination rate. It is safe to assume that this is due to the fact that richer countries can buy more vaccines. Additionally, the better the health service coverage is, the easier it is to vaccinate the population. For instance, in some African countries, there are rural regions in which the next doctor is often very far away. This is of course a large obstacle for a vaccination campaign.

While there were only few doses available in poor countries, in Europe, at times, there were news about vaccine doses expiring because there were more doses than needed or too many doses of less popular vaccines. Exporting the doses, according to some articles, is problematic due to bureaucratic obstacles such as rights of the vaccine producers.

The huge problem is, as we know all by now, that if in some countries in the world, there is a low vaccination rate and a high number of cases, it is very likely that new variants emerge. These new variants, as currently the Omikron variant, are often more infectious and might be able to bypass the immune protection created by the vaccines.

That is why it is essential to share the vaccines with the whole world - in the end it is not helping the rich countries if they hoard the vaccines and, in worst case, let them expire.