Course Code:DATA 601; Homework 4 - Data Visualization

Krishna Chaitanya Rao - MS Data Analytics and Computational Social Sciences

In this homework-4 I have used the same dataset whihch i had used for the homework-3. The objective of this homework is to work on the data visulaization and find the hidden patterns in the dataset (covid vaccination dataset).

library(tinytex)  
#Loading the Data  
#About Dataset: This dataset has the data related to covid19 vaccinations around the world.  
library(tidyverse)

## Warning: package 'tidyverse' was built under R version 3.5.2

## ── Attaching packages ─────────────────────────────────────── tidyverse 1.3.0 ──

## ✓ ggplot2 3.3.3 ✓ purrr 0.3.4  
## ✓ tibble 3.1.0 ✓ dplyr 1.0.5  
## ✓ tidyr 1.1.3 ✓ stringr 1.4.0  
## ✓ readr 1.4.0 ✓ forcats 0.5.1

## Warning: package 'stringr' was built under R version 3.5.2

## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag() masks stats::lag()

library(dplyr)  
testdata <- read\_csv("~/Desktop/MSDACSS/DATA601/country\_vaccinations.csv") #Tibble #reading the data

##   
## ── Column specification ────────────────────────────────────────────────────────  
## cols(  
## country = col\_character(),  
## iso\_code = col\_character(),  
## date = col\_date(format = ""),  
## total\_vaccinations = col\_double(),  
## people\_vaccinated = col\_double(),  
## people\_fully\_vaccinated = col\_double(),  
## daily\_vaccinations\_raw = col\_double(),  
## daily\_vaccinations = col\_double(),  
## total\_vaccinations\_per\_hundred = col\_double(),  
## people\_vaccinated\_per\_hundred = col\_double(),  
## people\_fully\_vaccinated\_per\_hundred = col\_double(),  
## daily\_vaccinations\_per\_million = col\_double(),  
## vaccines = col\_character(),  
## source\_name = col\_character(),  
## source\_website = col\_character()  
## )

#To find the missing values in the dataset; colum wise.  
colSums(is.na(testdata))

## country iso\_code   
## 0 356   
## date total\_vaccinations   
## 0 2200   
## people\_vaccinated people\_fully\_vaccinated   
## 2627 3740   
## daily\_vaccinations\_raw daily\_vaccinations   
## 2810 191   
## total\_vaccinations\_per\_hundred people\_vaccinated\_per\_hundred   
## 2200 2627   
## people\_fully\_vaccinated\_per\_hundred daily\_vaccinations\_per\_million   
## 3740 191   
## vaccines source\_name   
## 0 0   
## source\_website   
## 0

#Counting the missing values in the dataset.  
apply(testdata, MARGIN = 2, function(x) sum(is.na(x)))

## country iso\_code   
## 0 356   
## date total\_vaccinations   
## 0 2200   
## people\_vaccinated people\_fully\_vaccinated   
## 2627 3740   
## daily\_vaccinations\_raw daily\_vaccinations   
## 2810 191   
## total\_vaccinations\_per\_hundred people\_vaccinated\_per\_hundred   
## 2200 2627   
## people\_fully\_vaccinated\_per\_hundred daily\_vaccinations\_per\_million   
## 3740 191   
## vaccines source\_name   
## 0 0   
## source\_website   
## 0

#or   
library(psych)

##   
## Attaching package: 'psych'

## The following objects are masked from 'package:ggplot2':  
##   
## %+%, alpha

describe(testdata)

## Warning in FUN(newX[, i], ...): no non-missing arguments to min; returning Inf

## Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning -Inf

## vars n mean sd median  
## country\* 1 6108 65.73 39.08 63.00  
## iso\_code\* 2 5752 60.95 37.31 59.00  
## date 3 6108 NaN NA NA  
## total\_vaccinations 4 3908 2137405.14 7429940.97 238646.00  
## people\_vaccinated 5 3481 1757375.82 5602468.60 222005.00  
## people\_fully\_vaccinated 6 2368 680537.78 2731855.19 84892.50  
## daily\_vaccinations\_raw 7 3298 82918.43 243166.17 13382.50  
## daily\_vaccinations 8 5917 57919.61 189358.01 6637.00  
## total\_vaccinations\_per\_hundred 9 3908 8.63 15.84 3.24  
## people\_vaccinated\_per\_hundred 10 3481 6.71 10.93 2.88  
## people\_fully\_vaccinated\_per\_hundred 11 2368 2.95 6.61 1.19  
## daily\_vaccinations\_per\_million 12 5917 2549.64 4299.56 1217.00  
## vaccines\* 13 6108 11.92 6.17 10.00  
## source\_name\* 14 6108 45.12 21.28 54.00  
## source\_website\* 15 6108 57.69 37.03 50.00  
## trimmed mad min max  
## country\* 65.35 53.37 1 133.00  
## iso\_code\* 60.18 48.93 1 129.00  
## date NaN NA Inf -Inf  
## total\_vaccinations 675036.39 341700.01 0 101128005.00  
## people\_vaccinated 543536.33 310468.30 0 65965305.00  
## people\_fully\_vaccinated 218341.76 120024.62 1 35000159.00  
## daily\_vaccinations\_raw 31929.53 18582.17 0 2924112.00  
## daily\_vaccinations 18218.08 9249.94 1 2302844.00  
## total\_vaccinations\_per\_hundred 4.94 4.37 0 135.02  
## people\_vaccinated\_per\_hundred 4.11 3.69 0 85.99  
## people\_fully\_vaccinated\_per\_hundred 1.44 1.51 0 49.02  
## daily\_vaccinations\_per\_million 1647.77 1423.30 0 54264.00  
## vaccines\* 11.40 5.93 1 25.00  
## source\_name\* 46.46 14.83 1 79.00  
## source\_website\* 55.86 45.96 1 129.00  
## range skew kurtosis se  
## country\* 132.00 0.07 -1.28 0.50  
## iso\_code\* 128.00 0.15 -1.24 0.49  
## date -Inf NA NA NA  
## total\_vaccinations 101128005.00 8.00 78.03 118852.44  
## people\_vaccinated 65965305.00 6.51 51.65 94957.09  
## people\_fully\_vaccinated 35000158.00 8.70 84.31 56139.28  
## daily\_vaccinations\_raw 2924112.00 6.49 50.30 4234.26  
## daily\_vaccinations 2302843.00 6.82 55.30 2461.69  
## total\_vaccinations\_per\_hundred 135.02 3.88 18.58 0.25  
## people\_vaccinated\_per\_hundred 85.99 3.16 12.36 0.19  
## people\_fully\_vaccinated\_per\_hundred 49.02 4.58 22.38 0.14  
## daily\_vaccinations\_per\_million 54264.00 4.95 39.14 55.90  
## vaccines\* 24.00 0.53 -0.90 0.08  
## source\_name\* 78.00 -0.64 -0.67 0.27  
## source\_website\* 128.00 0.31 -1.18 0.47

class(testdata) #Type of data

## [1] "spec\_tbl\_df" "tbl\_df" "tbl" "data.frame"

dim(testdata) #Chcek the dimensions of the data. It returns number of rows \* Coloumns

## [1] 6108 15

head(testdata, n = 8) #It displays first 8 rows of the data set

## # A tibble: 8 x 15  
## country iso\_code date total\_vaccinati… people\_vaccinat… people\_fully\_va…  
## <chr> <chr> <date> <dbl> <dbl> <dbl>  
## 1 Albania ALB 2021-01-10 0 0 NA  
## 2 Albania ALB 2021-01-11 NA NA NA  
## 3 Albania ALB 2021-01-12 128 128 NA  
## 4 Albania ALB 2021-01-13 188 188 NA  
## 5 Albania ALB 2021-01-14 266 266 NA  
## 6 Albania ALB 2021-01-15 308 308 NA  
## 7 Albania ALB 2021-01-16 369 369 NA  
## 8 Albania ALB 2021-01-17 405 405 NA  
## # … with 9 more variables: daily\_vaccinations\_raw <dbl>,  
## # daily\_vaccinations <dbl>, total\_vaccinations\_per\_hundred <dbl>,  
## # people\_vaccinated\_per\_hundred <dbl>,  
## # people\_fully\_vaccinated\_per\_hundred <dbl>,  
## # daily\_vaccinations\_per\_million <dbl>, vaccines <chr>, source\_name <chr>,  
## # source\_website <chr>

tail(testdata, n = 9) #It displays last 9 rows of the data set

## # A tibble: 9 x 15  
## country iso\_code date total\_vaccinati… people\_vaccinat… people\_fully\_va…  
## <chr> <chr> <date> <dbl> <dbl> <dbl>  
## 1 Zimbab… ZWE 2021-03-04 30658 30658 NA  
## 2 Zimbab… ZWE 2021-03-05 31325 31325 NA  
## 3 Zimbab… ZWE 2021-03-06 32014 32014 NA  
## 4 Zimbab… ZWE 2021-03-07 32240 32240 NA  
## 5 Zimbab… ZWE 2021-03-08 35518 35518 NA  
## 6 Zimbab… ZWE 2021-03-09 35761 35761 NA  
## 7 Zimbab… ZWE 2021-03-10 35901 35901 NA  
## 8 Zimbab… ZWE 2021-03-11 36019 36019 NA  
## 9 Zimbab… ZWE 2021-03-12 36283 36283 NA  
## # … with 9 more variables: daily\_vaccinations\_raw <dbl>,  
## # daily\_vaccinations <dbl>, total\_vaccinations\_per\_hundred <dbl>,  
## # people\_vaccinated\_per\_hundred <dbl>,  
## # people\_fully\_vaccinated\_per\_hundred <dbl>,  
## # daily\_vaccinations\_per\_million <dbl>, vaccines <chr>, source\_name <chr>,  
## # source\_website <chr>

#Displaying the coloumn names  
colnames(testdata)

## [1] "country" "iso\_code"   
## [3] "date" "total\_vaccinations"   
## [5] "people\_vaccinated" "people\_fully\_vaccinated"   
## [7] "daily\_vaccinations\_raw" "daily\_vaccinations"   
## [9] "total\_vaccinations\_per\_hundred" "people\_vaccinated\_per\_hundred"   
## [11] "people\_fully\_vaccinated\_per\_hundred" "daily\_vaccinations\_per\_million"   
## [13] "vaccines" "source\_name"   
## [15] "source\_website"

#Viewing Country code wise   
table(testdata$iso\_code)

##   
## AGO AIA ALB AND ARE ARG AUS AUT   
## 4 23 60 45 67 74 26 75   
## AZE BEL BGD BGR BHR BLR BLZ BMU   
## 55 74 45 74 80 53 13 60   
## BOL BRA BRB CAN CHE CHL CHN CIV   
## 44 56 25 89 46 79 76 10   
## COL CRI CYM CYP CZE DEU DMA DNK   
## 24 79 70 61 76 76 19 75   
## DOM DZA ECU EGY ESP EST FIN FLK   
## 25 22 51 7 67 74 72 16   
## FRA FRO GBR GGY GHA GIB GRC GRD   
## 75 43 89 38 10 62 75 16   
## GRL GTM GUY HKG HND HRV HUN IDN   
## 34 15 12 20 1 69 75 60   
## IMN IND IRL IRN ISL ISR ITA JEY   
## 51 57 69 10 73 84 76 43   
## JOR JPN KAZ KEN KHM KOR KWT LBN   
## 45 24 25 7 32 16 71 28   
## LCA LIE LKA LTU LUX LVA MAC MAR   
## 17 33 44 76 72 71 33 44   
## MCO MDA MDV MEX MLT MMR MNE MNG   
## 20 8 39 79 54 8 21 5   
## MSR MUS MYS NLD NOR NPL NZL OMN   
## 16 24 18 50 75 26 21 73   
## OWID\_NCY PAK PAN PER PHL POL PRT PRY   
## 9 20 53 31 11 74 76 20   
## QAT ROU RUS RWA SAU SEN SGP SHN   
## 76 75 88 25 66 17 57 1   
## SLV SMR SRB SVK SVN SWE SYC TCA   
## 9 13 64 68 75 76 62 30   
## THA TTO TUR UKR URY USA VEN ZAF   
## 12 8 59 17 14 83 16 25   
## ZWE   
## 23

#List the summary of each attribute   
summary(testdata)

## country iso\_code date total\_vaccinations   
## Length:6108 Length:6108 Min. :2020-12-13 Min. : 0   
## Class :character Class :character 1st Qu.:2021-01-22 1st Qu.: 35432   
## Mode :character Mode :character Median :2021-02-10 Median : 238646   
## Mean :2021-02-06 Mean : 2137405   
## 3rd Qu.:2021-02-25 3rd Qu.: 1112548   
## Max. :2021-03-12 Max. :101128005   
## NA's :2200   
## people\_vaccinated people\_fully\_vaccinated daily\_vaccinations\_raw  
## Min. : 0 Min. : 1 Min. : 0   
## 1st Qu.: 33633 1st Qu.: 16035 1st Qu.: 2618   
## Median : 222005 Median : 84892 Median : 13382   
## Mean : 1757376 Mean : 680538 Mean : 82918   
## 3rd Qu.: 911312 3rd Qu.: 425709 3rd Qu.: 56643   
## Max. :65965305 Max. :35000159 Max. :2924112   
## NA's :2627 NA's :3740 NA's :2810   
## daily\_vaccinations total\_vaccinations\_per\_hundred  
## Min. : 1 Min. : 0.000   
## 1st Qu.: 1122 1st Qu.: 0.680   
## Median : 6637 Median : 3.240   
## Mean : 57920 Mean : 8.627   
## 3rd Qu.: 28857 3rd Qu.: 8.762   
## Max. :2302844 Max. :135.020   
## NA's :191 NA's :2200   
## people\_vaccinated\_per\_hundred people\_fully\_vaccinated\_per\_hundred  
## Min. : 0.000 Min. : 0.000   
## 1st Qu.: 0.690 1st Qu.: 0.310   
## Median : 2.880 Median : 1.190   
## Mean : 6.712 Mean : 2.949   
## 3rd Qu.: 6.850 3rd Qu.: 2.610   
## Max. :85.990 Max. :49.020   
## NA's :2627 NA's :3740   
## daily\_vaccinations\_per\_million vaccines source\_name   
## Min. : 0 Length:6108 Length:6108   
## 1st Qu.: 355 Class :character Class :character   
## Median : 1217 Mode :character Mode :character   
## Mean : 2550   
## 3rd Qu.: 2682   
## Max. :54264   
## NA's :191   
## source\_website   
## Length:6108   
## Class :character   
## Mode :character   
##   
##   
##   
##

#To get to know more than or equal to 100   
filter(testdata, people\_vaccinated >= 100)

## # A tibble: 3,416 x 15  
## country iso\_code date total\_vaccinations people\_vaccinated  
## <chr> <chr> <date> <dbl> <dbl>  
## 1 Albania ALB 2021-01-12 128 128  
## 2 Albania ALB 2021-01-13 188 188  
## 3 Albania ALB 2021-01-14 266 266  
## 4 Albania ALB 2021-01-15 308 308  
## 5 Albania ALB 2021-01-16 369 369  
## 6 Albania ALB 2021-01-17 405 405  
## 7 Albania ALB 2021-01-18 447 447  
## 8 Albania ALB 2021-01-19 483 483  
## 9 Albania ALB 2021-01-20 519 519  
## 10 Albania ALB 2021-01-21 549 549  
## # … with 3,406 more rows, and 10 more variables: people\_fully\_vaccinated <dbl>,  
## # daily\_vaccinations\_raw <dbl>, daily\_vaccinations <dbl>,  
## # total\_vaccinations\_per\_hundred <dbl>, people\_vaccinated\_per\_hundred <dbl>,  
## # people\_fully\_vaccinated\_per\_hundred <dbl>,  
## # daily\_vaccinations\_per\_million <dbl>, vaccines <chr>, source\_name <chr>,  
## # source\_website <chr>

arrange(testdata, date, source\_name)

## # A tibble: 6,108 x 15  
## country iso\_code date total\_vaccinations people\_vaccinated  
## <chr> <chr> <date> <dbl> <dbl>  
## 1 England <NA> 2020-12-13 55576 55576  
## 2 Northern Ireland <NA> 2020-12-13 3623 3623  
## 3 Scotland <NA> 2020-12-13 19009 19009  
## 4 United Kingdom GBR 2020-12-13 86465 86465  
## 5 Wales <NA> 2020-12-13 8257 8257  
## 6 England <NA> 2020-12-14 NA NA  
## 7 Northern Ireland <NA> 2020-12-14 NA NA  
## 8 Scotland <NA> 2020-12-14 NA NA  
## 9 United Kingdom GBR 2020-12-14 NA NA  
## 10 Wales <NA> 2020-12-14 NA NA  
## # … with 6,098 more rows, and 10 more variables: people\_fully\_vaccinated <dbl>,  
## # daily\_vaccinations\_raw <dbl>, daily\_vaccinations <dbl>,  
## # total\_vaccinations\_per\_hundred <dbl>, people\_vaccinated\_per\_hundred <dbl>,  
## # people\_fully\_vaccinated\_per\_hundred <dbl>,  
## # daily\_vaccinations\_per\_million <dbl>, vaccines <chr>, source\_name <chr>,  
## # source\_website <chr>

summarytest <- group\_by(testdata, date, vaccines, source\_name)  
summarise(summarytest, mean\_testing = mean(people\_vaccinated, na.rm = TRUE))

## `summarise()` has grouped output by 'date', 'vaccines'. You can override using the `.groups` argument.

## # A tibble: 4,617 x 4  
## # Groups: date, vaccines [1,573]  
## date vaccines source\_name mean\_testing  
## <date> <chr> <chr> <dbl>  
## 1 2020-12-13 Oxford/AstraZeneca, Pfi… Government of the United Ki… 34586  
## 2 2020-12-14 Moderna, Pfizer/BioNTech Official data from province… NaN  
## 3 2020-12-14 Oxford/AstraZeneca, Pfi… Government of the United Ki… NaN  
## 4 2020-12-15 EpiVacCorona, Sputnik V Official data from local go… 28500  
## 5 2020-12-15 Moderna, Pfizer/BioNTech Official data from province… NaN  
## 6 2020-12-15 Oxford/AstraZeneca, Pfi… Government of the United Ki… NaN  
## 7 2020-12-15 Sinopharm/Beijing, Sino… National Health Commission NaN  
## 8 2020-12-16 EpiVacCorona, Sputnik V Official data from local go… NaN  
## 9 2020-12-16 Moderna, Pfizer/BioNTech Official data from province… NaN  
## 10 2020-12-16 Oxford/AstraZeneca, Pfi… Government of the United Ki… NaN  
## # … with 4,607 more rows

newtestdata <- testdata %>%  
 select(country, iso\_code, date, total\_vaccinations) %>%  
 group\_by(country) %>%  
 filter(!is.na(total\_vaccinations)) %>%  
 filter(total\_vaccinations == max(total\_vaccinations))  
  
head(testdata, n = 15)

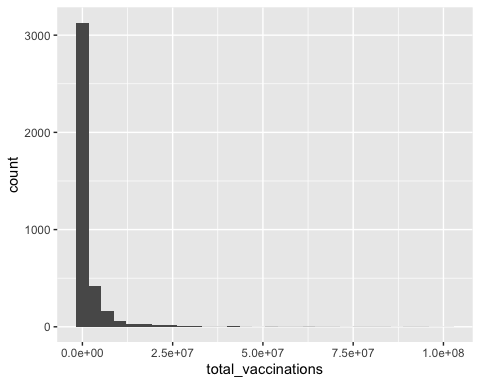
## # A tibble: 15 x 15  
## country iso\_code date total\_vaccinations people\_vaccinated  
## <chr> <chr> <date> <dbl> <dbl>  
## 1 Albania ALB 2021-01-10 0 0  
## 2 Albania ALB 2021-01-11 NA NA  
## 3 Albania ALB 2021-01-12 128 128  
## 4 Albania ALB 2021-01-13 188 188  
## 5 Albania ALB 2021-01-14 266 266  
## 6 Albania ALB 2021-01-15 308 308  
## 7 Albania ALB 2021-01-16 369 369  
## 8 Albania ALB 2021-01-17 405 405  
## 9 Albania ALB 2021-01-18 447 447  
## 10 Albania ALB 2021-01-19 483 483  
## 11 Albania ALB 2021-01-20 519 519  
## 12 Albania ALB 2021-01-21 549 549  
## 13 Albania ALB 2021-01-22 NA NA  
## 14 Albania ALB 2021-01-23 NA NA  
## 15 Albania ALB 2021-01-24 NA NA  
## # … with 10 more variables: people\_fully\_vaccinated <dbl>,  
## # daily\_vaccinations\_raw <dbl>, daily\_vaccinations <dbl>,  
## # total\_vaccinations\_per\_hundred <dbl>, people\_vaccinated\_per\_hundred <dbl>,  
## # people\_fully\_vaccinated\_per\_hundred <dbl>,  
## # daily\_vaccinations\_per\_million <dbl>, vaccines <chr>, source\_name <chr>,  
## # source\_website <chr>

#Data Visualization

library(tidyverse)  
ggplot(testdata, aes(total\_vaccinations)) + geom\_histogram()

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

## Warning: Removed 2200 rows containing non-finite values (stat\_bin).

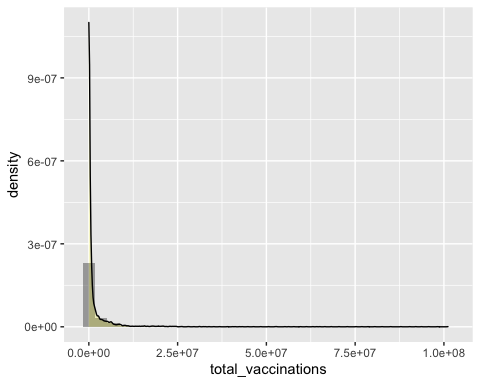


#Plotting density   
  
ggplot(testdata, aes(total\_vaccinations)) + geom\_histogram(aes(y= ..density..), alpha = 0.5) + geom\_density(alpha = 0.2, fill = 'yellow')

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

## Warning: Removed 2200 rows containing non-finite values (stat\_bin).

## Warning: Removed 2200 rows containing non-finite values (stat\_density).



library(pacman)

## Warning: package 'pacman' was built under R version 3.5.2

library(tidyverse)  
library(lubridate)

##   
## Attaching package: 'lubridate'

## The following objects are masked from 'package:base':  
##   
## date, intersect, setdiff, union

#Let me store my original dataset which is in "testdata" to a temporary dataframe.  
tempdata <- testdata  
  
## Used na.omit command to remove any NA values in the dataset   
newtempdata <- na.omit(tempdata)  
head(newtempdata)

## # A tibble: 6 x 15  
## country iso\_code date total\_vaccinati… people\_vaccinat… people\_fully\_va…  
## <chr> <chr> <date> <dbl> <dbl> <dbl>  
## 1 Albania ALB 2021-02-18 3049 2438 611  
## 2 Argent… ARG 2021-01-21 265724 249372 16352  
## 3 Argent… ARG 2021-01-22 279602 254456 25146  
## 4 Argent… ARG 2021-01-23 288064 258876 29188  
## 5 Argent… ARG 2021-01-24 292023 260036 31987  
## 6 Argent… ARG 2021-01-25 292386 260122 32264  
## # … with 9 more variables: daily\_vaccinations\_raw <dbl>,  
## # daily\_vaccinations <dbl>, total\_vaccinations\_per\_hundred <dbl>,  
## # people\_vaccinated\_per\_hundred <dbl>,  
## # people\_fully\_vaccinated\_per\_hundred <dbl>,  
## # daily\_vaccinations\_per\_million <dbl>, vaccines <chr>, source\_name <chr>,  
## # source\_website <chr>

## Now I am interested in from the larger dataset and named this new table 'data1'. Lets try to see the relationship over time between countries and daily vaccinations.  
  
data1 <- newtempdata %>%  
 select(country, date, daily\_vaccinations)  
head(data1)

## # A tibble: 6 x 3  
## country date daily\_vaccinations  
## <chr> <date> <dbl>  
## 1 Albania 2021-02-18 254  
## 2 Argentina 2021-01-21 11704  
## 3 Argentina 2021-01-22 11263  
## 4 Argentina 2021-01-23 11124  
## 5 Argentina 2021-01-24 10342  
## 6 Argentina 2021-01-25 9046

#Since there are lot of countries, So I have decided to play with specific countries where covid is growing drastically.   
data2 <- data1 %>%  
 filter(country %in% c("Germany", "United Kingdom", "United States", "Canada", "France", "Israel", "Chile", "Brazil"))  
head(data2)

## # A tibble: 6 x 3  
## country date daily\_vaccinations  
## <chr> <date> <dbl>  
## 1 Brazil 2021-02-06 199739  
## 2 Brazil 2021-02-07 211375  
## 3 Brazil 2021-02-08 211604  
## 4 Brazil 2021-02-09 218237  
## 5 Brazil 2021-02-10 228375  
## 6 Brazil 2021-02-11 190540

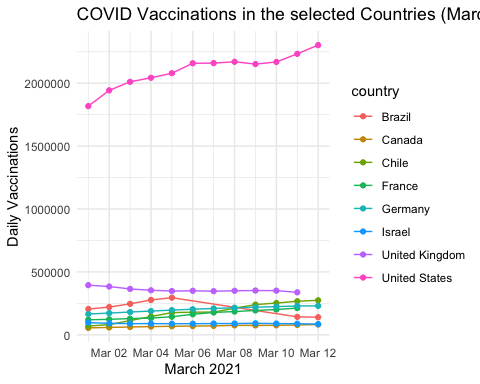
str(data2)

## tibble [437 × 3] (S3: tbl\_df/tbl/data.frame)  
## $ country : chr [1:437] "Brazil" "Brazil" "Brazil" "Brazil" ...  
## $ date : Date[1:437], format: "2021-02-06" "2021-02-07" ...  
## $ daily\_vaccinations: num [1:437] 199739 211375 211604 218237 228375 ...  
## - attr(\*, "na.action")= 'omit' Named int [1:4207] 1 2 3 4 5 6 7 8 9 10 ...  
## ..- attr(\*, "names")= chr [1:4207] "1" "2" "3" "4" ...

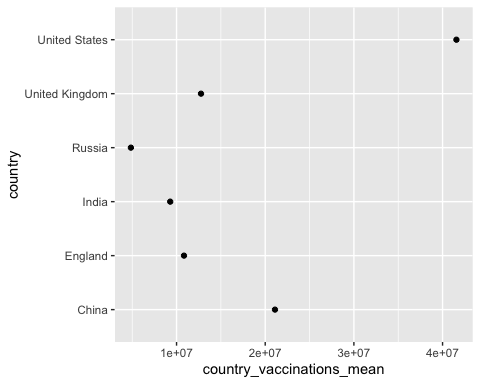
#I found this pacman library very useful while formating the date as "year", month, day to make it easier to work with when I go to filter the dates.  
data2$date <- data2$date %>%  
 ymd()

## Used dplyr 'filter' command in order to choose only March 2021 dates, so as to make the final visualization less daunting also i am very intrested to see march because its a Covid Anniversary :)   
  
march\_vacc <- data2 %>%  
 filter(month(date)==3)

plot1<-ggplot(march\_vacc, aes(x=date, y=daily\_vaccinations, group=country)) +  
 geom\_line(aes(color=country))+  
 geom\_point(aes(color=country)) +  
 ggtitle("COVID Vaccinations in the selected Countries (March 2021)")+  
 xlab("March 2021")+  
 ylab("Daily Vaccinations")+  
 theme\_minimal()  
plot1



by\_country <- group\_by(tempdata, country)  
summarise <- summarise(by\_country, count = n(),  
country\_vaccinations\_mean = mean(total\_vaccinations, na.rm = TRUE))  
by\_country <-head(summarise %>% arrange(desc(country\_vaccinations\_mean)))  
  
#ggplot  
ggplot(by\_country, aes(x=country\_vaccinations\_mean, y=country)) + geom\_point()



ggplot(data=by\_country, aes(x=(reorder(country, country\_vaccinations\_mean)), y = country\_vaccinations\_mean))+  
 geom\_bar(stat="identity", fill= 'skyblue')+ coord\_flip()+  
 labs(title="Average of country vaccinations", x= "Country", y = "Country Vaccinations Mean")+  
 geom\_text(aes(label=round(country\_vaccinations\_mean, digits = 2)))+  
 theme(plot.title=element\_text(hjust=0.5)) + theme\_minimal()

