Group Members:

Wesley Chok

Project 1

1. Data Wrangling

The current steps completed are: a, c, d, f and g. For step a, to read the csv data, we need to ensure to use the read_csv function. With this the following csv hyper link or file are placed into the parameters. To save the csv I have used the leftwards assignment operator to save it to a variable. For step b, I have removed the rows that contains an NA value, vaccination rates that are 0 and the confirmed shots that are 0. Touching on step c, the tables that I have tidied are the covid and demographic tables. For the covid data, the reason it is messy is because it contains column names as dates and the variables as vaccination shots. To tidy this, I have filtered out all the column names except for the dates, set the name of the column to “dates” and then set the values to “confirmed”. Instead of having an abundance of columns, now dates are formatted to rows and the shots relating to that date are placed to the next column. For the demographics table, YR2015 contains multiple values from other columns. To tidy this, I have pivoted wider to have Series Code as the column name and the values as YR2015. Moving on to step d, to get the vaccination rate I have mutated the data. The variables I have used are the population variable and the confirmed variable that I have tidied. For step e, to get the number of days since the first non-zero vaccination number I had to use the mutate(). The value will be whatever the current row number is. In order to reset the days so it does not bleed into other countries, I used the group_by() function and grouped by Combined_Key. To reset the value, I’ve used the cumsum() function. Going to step f, to get only the number of hospital beds from the most recent year which was 2019 according to the data table. For step g I have used the mutate and select function to combine the male/female population. For each serial code group, I have added the “FE” and ‘MA”

together. Lastly for step h, I've merged all the tables through the inner_join() function. As suggested, I've compared all the country names and merged all the tables together.

2. Data Modeling

For the modeling combinations, I have compared the population with 5 of the demographic series codes. Since there were multiple series code in the demographic file, I thought it would be interesting to see the differences with population. SP.URB.TOTL, SP.POP.TOTL, SP.POP.80UP and SP.POP.1564.IN all had similar R-squared values in comparison to the population. The only outlier that existed was SP.DYN.LE00.IN which had a R-squared value of 0.06027.

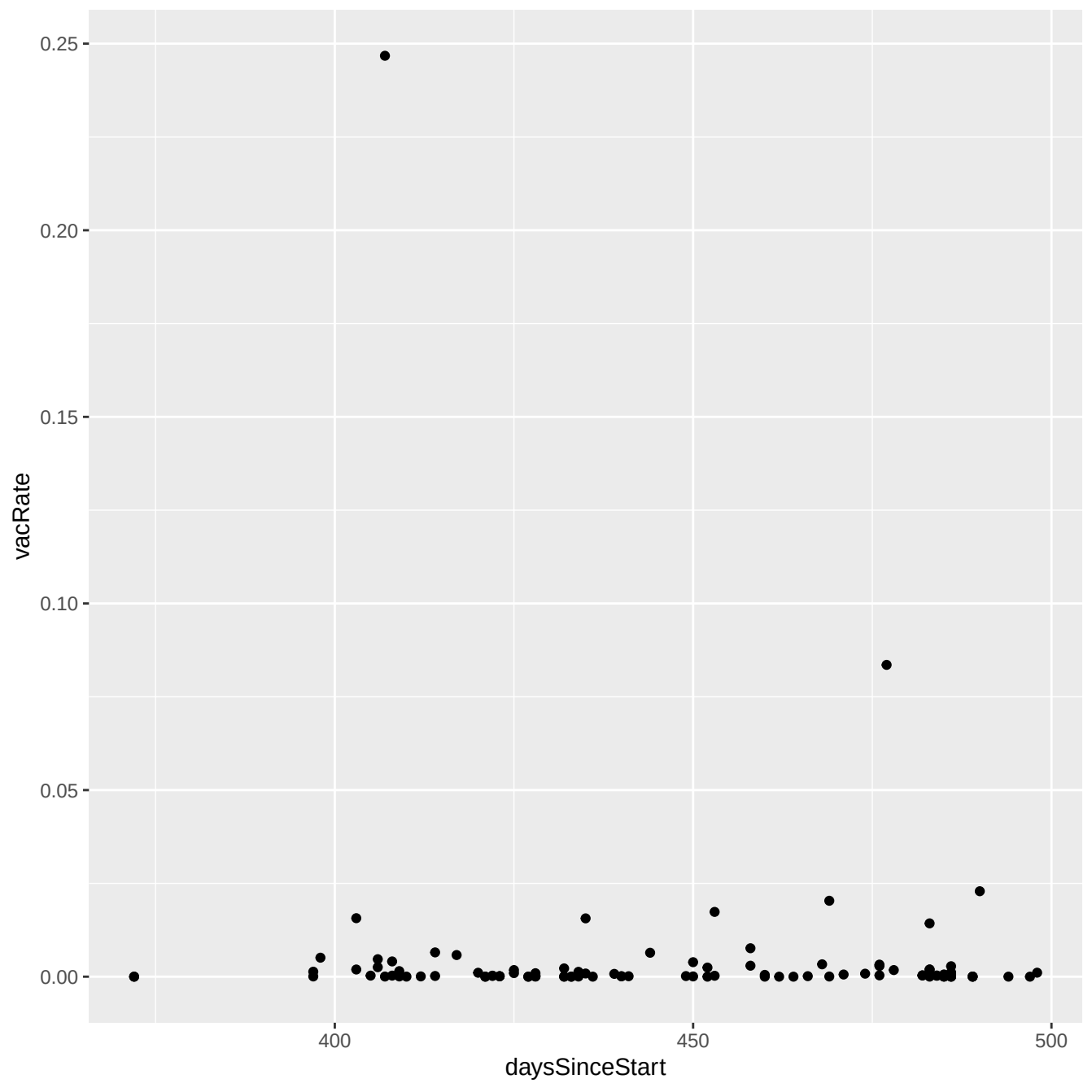
3. Conclusion

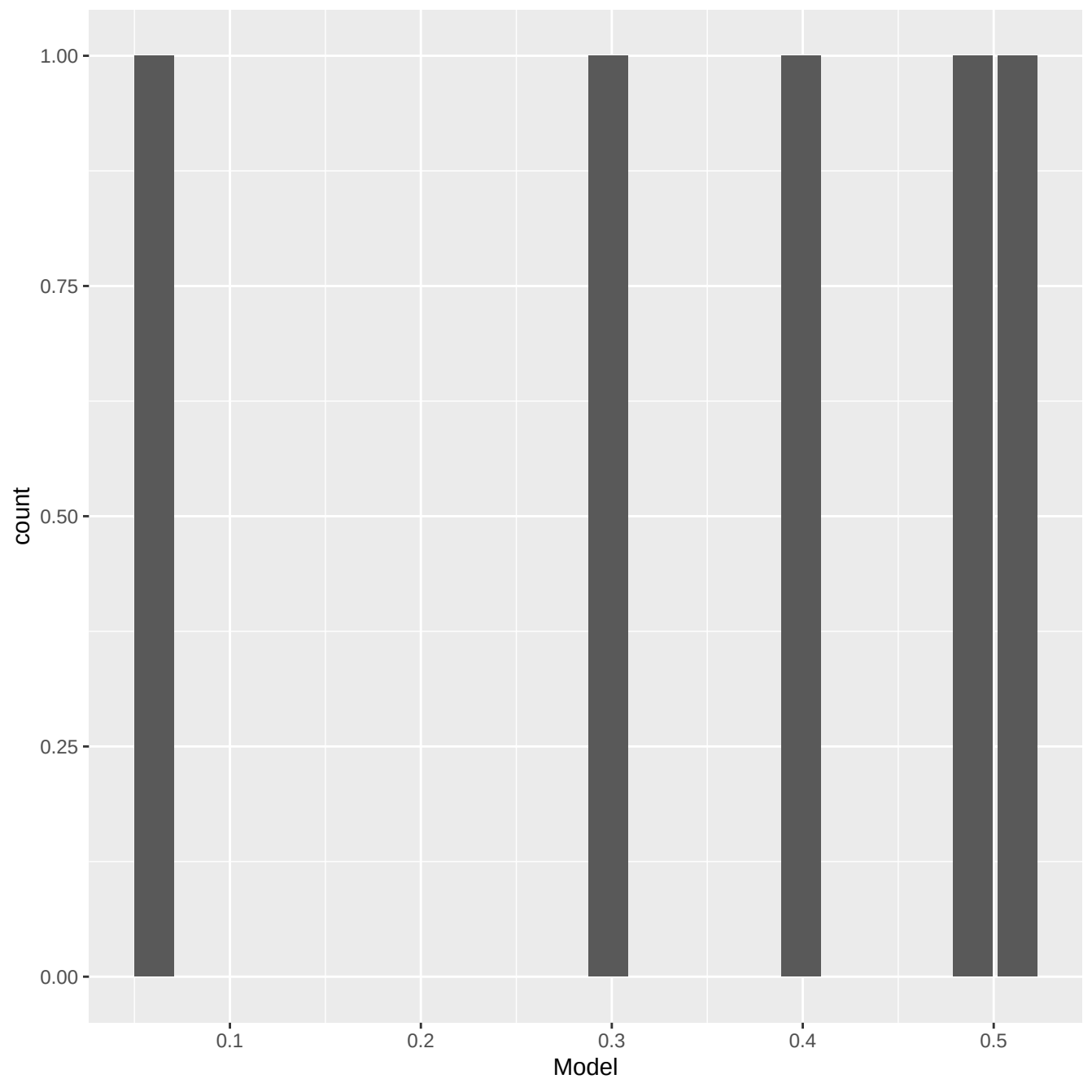
Most of the vaccination rates seem to have a very close margin to 0 vaccinations throughout all the countries. These rates are most likely to be the most impactful data as it's all clumped together the most noticeable outlier out of all of the countries had a vaccination rate of approximately 0.24. The outlier is likely to be the least significant factor as it does not accurately represent the general vaccination rates per country,

```r
library(tidyverse)

covid <- read_csv("https://raw.githubusercontent.com/govex/COVID-19/master/data_tables/vaccine_data/global_data/time_series_covid19_vaccine_doses_admin_global.cs
hospital <- read_csv("data.csv")
demo <- read_csv("demographics.csv")

# Data wrangling for vaccinations data. Created column for dates and
# confirmed vaccinations shots
covid %>%
  pivot_longer(cols = -one_of('UID', 'iso2', 'iso3', 'code3', 'Admin2',
                              'Province_State', 'Country_Region', 'Lat',
                              'Long_', 'Combined_Key', 'Population'),
               names_to='date',values_to= 'confirmed') -> covid
covid

# New variable to calculate the vaccination rate (confirmed/ Population)
vacRate <- covid %>% mutate(vacRate = confirmed / Population)

# Removed unnecessary data
hello <- vacRate %>% filter(!is.na(confirmed), !is.na(vacRate), vacRate > 0,
                            confirmed > 0)

# New variable to calculate number of days since last non-zero vaccination shot
value <- hello %>%
  group_by(Combined_Key,
           group = cumsum(date = lag(date, default = "12/31/2021"))) %>%
  mutate(daysSinceStart = row_number()) %>% ungroup %>% select(-group)

# Completed data wrangling vaccination data
value

demo_tidy <- demo %>% select(-`Series Name`) %>%
  pivot_wider(names_from = "Series Code", values_from = YR2015)

demo_merged <- demo_tidy %>%
  mutate(SP.POP.80UP=SP.POP.80UP.FE+SP.POP.80UP.MA) %>%
  mutate(SP.POP.1564.IN=SP.POP.1564.MA.IN+SP.POP.1564.FE.IN) %>%
  mutate(SP.POP.0014.IN=SP.POP.0014.MA.IN+SP.POP.0014.FE.IN) %>%
  mutate(SP.DYN.AMRT=SP.DYN.AMRT.MA+SP.DYN.AMRT.FE) %>%
  mutate(SP.POP.TOTL.IN=SP.POP.TOTL.FE.IN+SP.POP.TOTL.MA.IN) %>%
  mutate(SP.POP.65UP.IN=SP.POP.65UP.FE.IN+SP.POP.65UP.MA.IN) %>%
  select(-contains(".FE")) %>% select(-contains(".MA"))

names(demo_merged) <- gsub(" ", "_", names(demo_merged))

# Completed demographics wrangling vaccination data
demo_merged

#hospital <- filter(hospital, Year == 2019)

# Completed hospital wrangling vaccination data
hospital


# Changed name to South Korea
demo_merged <- demo_merged %>%
```

```
  mutate(Country_Name = replace(Country_Name , Country_Name  ==
                                  "Korea, Dem. People’s Rep.", "South Korea"))
demo_merged <- demo_merged %>%
  mutate(Country_Name = replace(Country_Name, Country_Name == "Korea, Rep.",
                                "South Korea"))

# Changed name to Bahamas
demo_merged <- demo_merged %>%
  mutate(Country_Name = replace(Country_Name, Country_Name == "Bahamas, The",
                                "Bahamas"))

# Changed name to Iran
demo_merged <- demo_merged %>%
  mutate(Country_Name = replace(Country_Name,
                                Country_Name == "Iran, Islamic Rep.", "Iran"))

# Changed name to Hong Kong
demo_merged <- demo_merged %>%
  mutate(Country_Name = replace(Country_Name,
                                Country_Name == "Hong Kong SAR, China",
                                "Hong Kong"))

# Merged demographics data into value
value_joined <- value %>%
  inner_join(demo_merged, by=c(Country_Region = "Country_Name"))

# Merged hospital data into value
value_joined <- value_joined %>%
  inner_join(hospital, by=c(Country_Region = "Country"))

# Completed merge of all 3 tables
value_joined

# Linear modeling data, Population vs SP.DYN.LE00.IN
modPopulationDyn <- lm(data=value_joined, formula=Population~SP.DYN.LE00.IN)
summary(modPopulationDyn)

# Linear modeling data, Population vs SP.URB.TOTL
modPopulationUrb <- lm(data=value_joined, formula=Population~SP.URB.TOTL)
summary(modPopulationUrb)

# Linear modeling data, Population vs SP.POP.TOTL
modPopulationUrbPop <- lm(data=value_joined, formula=Population~SP.POP.TOTL)
summary(modPopulationUrbPop)

# Linear modeling data, Population vs SP.POP.80UP
modPopulationUrb80UP <- lm(data=value_joined, formula=Population~SP.POP.80UP)
summary(modPopulationUrb80UP)

# Linear modeling data, Population vs SP.POP.1564.IN
modPopulationUrb80In <- lm(data=value_joined, formula=Population~SP.POP.1564.IN)
summary(modPopulationUrb80In)


# Calculates the most recent vaccination rate per country
valueTable <- value_joined %>% group_by(Country_Region) %>%
  summarize(daysSinceStart = max(daysSinceStart), vacRate = first(vacRate))
```

```
114
115 valueTable
116
117
118 # Scatterplot
119 ggplot(data=valueTable) + geom_point(mapping = aes(x=daysSinceStart, y=vacRate))
120
121 ggsave("scatterplot.pdf")
122
123 Model <- c(0.06027, 0.3993, 0.5123, 0.2982, 0.4891)
124
125 df <- data.frame(Model)
126
127 # Bar graph
128 ggplot(data=df) + geom_bar(mapping = aes(x=Model))
129
130 ggsave("bar_graph.pdf")
```


0.25
0.20
0.15
0.10
0.05
0.00
vacRate
400
450
500
daysSinceStart


1.00
0.75
0.50
0.25
0.00
count
0.1
0.2
0.3
0.4
0.5
Model