CPSC 375 Project - Summary

Group Members: Matthew Padden, Prateek Gupta, Malka Lazerson

Load necessary libraries

Summary

Load the libraries needed to carry out the commands required to properly organize and analyze the data.

R Code

```
> library(tidyverse)
```

Read and store databases into data frames

Summary

The data we will be working with is stored into three separate data frames. The vaccines data frame holds daily Covid-19 vaccination information per country. The beds data frame holds information on hospital bed density data by country. Finally, the demographics data frame holds the proportion of a country's population in different age groups and some other demographic data such as mortality rates and expected lifetime. We will organize and combine these three data frames into one table where we can work from and perform the necessary data analysis.

R Code

```
> vaccines <-
read_csv("https://raw.githubusercontent.com/govex/COVID-19/master/data_tables/vaccine_data/global_data/time_series_covid19_vaccine_doses_admin_global.csv")

> beds <- read_csv("data.csv")

> demographics <- read_csv("demographics.csv")
```

Data preparation - Vaccines

Summary

The vaccines data frame holds daily Covid-19 vaccination information per country. However, data wrangling is necessary to work with this data properly. To make this data tidy we will need to remove some unnecessary columns then pivot the table longer by making the column names

the previous “Date” values, and the new values to the previous “Shots” values. This will organize the data by vaccinations per day by country and make it much easier to read and work with. After we tidy the data we can replace some of the countries' names so our data can work better with other tables and we don’t lose data when we join them.

Tidy Data - R Code

```
> vaccines <- vaccines %>% select(-c(FIPS, Admin2, Province_State, code3, iso3, iso2, UID, Lat, Long_, Combined_Key), Country = Country_Region)

> vaccines <- vaccines %>% pivot_longer(3:473, names_to = "Date", values_to = "Shots", values_drop_na = TRUE) %>% filter(Shots != 0) %>% group_by(Country) %>% mutate(vacRate = Shots/Population, daysSinceStart = row_number()) %>% ungroup() %>% select(Country, vacRate, Shots, Population, daysSinceStart)
```

Replace Countries - R Code

```
> vaccines <- vaccines %>% mutate(Country = replace(Country, Country == "Korea, South", "South Korea"))

> vaccines <- vaccines %>% mutate(Country = replace(Country, Country == "US", "United States"))
```

Data preparation - Beds

Summary

The beds data frame holds information on hospital bed density data by country. To make this data easier to work with we will group the data by country name and only look at the most recent year as it is the most relevant for our work. Again, we will replace some of the countries' names so our data can work better with other tables and we don’t lose data when we join them.

Tidy Data - R Code

```
> beds <- beds %>% group_by(Country) %>% filter(Year == max(Year)) %>% select(Country, Beds = `Hospital beds (per 10 000 population)`)
```

Replace Countries - R Code

```
> beds <- beds %>% mutate(Country = replace(Country, Country == "Democratic People's Republic of Korea", "South Korea"))

> beds <- beds %>% mutate(Country = replace(Country, Country == "Republic of Korea", "South Korea"))
```

```
> beds <- beds %>% mutate(Country = replace(Country, Country == "United Kingdom of Great Britain and Northern Ireland", "United Kingdom"))

> beds <- beds %>% mutate(Country = replace(Country, Country == "Iran (Islamic Republic of)", "Iran"))

> beds <- beds %>% mutate(Country = replace(Country, Country == "United States of America", "United States"))
```

Data preparation - Demographics

Summary

The demographics data frame holds the proportion of a country's population in different age groups and some other demographic data such as mortality rates and expected lifetime. This data needs to be tidied by removing some unnecessary column names and pivoting wider by taking the names from the previous "Series Code" column and making the values data from the previous "YR2015" column. Once the table is wider, we will combine the female and male information together into new columns with total information. Again, we will replace some of the countries' names so our data can work better with other tables and we don't lose data when we join them.

Tidy Data - R Code

```
> demographics <- demographics %>% pivot_wider(-'Series Name', names_from = `Series Code`, values_from = YR2015) %>% mutate(SP.POP.80UP.TOTL = SP.POP.80UP.FE + SP.POP.80UP.MA, SP.POP.1564.TOTL = SP.POP.1564.MA.IN + SP.POP.1564.FE.IN, SP.POP.0014.TOTL = SP.POP.0014.MA.IN + SP.POP.0014.FE.IN, SP.DYN.AMRT.TOTL = SP.DYN.AMRT.FE + SP.DYN.AMRT.MA, SP.POP.TOTL.TOTL = SP.POP.TOTL.FE.IN + SP.POP.TOTL.MA.IN, SP.POP.65UP.TOTL = SP.POP.65UP.FE.IN + SP.POP.65UP.MA.IN) %>% select(Country = `Country Name`, 3:5, 18:23)
```

Replace Countries in Demographics DF - R Code

```
> demographics <- demographics %>% mutate(Country = replace(Country, Country == "Korea, Dem. People's Rep.", "South Korea"))

> demographics <- demographics %>% mutate(Country = replace(Country, Country == "Korea, Rep.", "South Korea"))

> demographics <- demographics %>% mutate(Country = replace(Country, Country == "Iran, Islamic Rep.", "Iran"))
```

Joining data tables

Summary

Once all of our tables we are going to work with are tidy and organized with the information we need, we will perform an inner join on them to create one new table that we will be working with to perform our linear models and data analysis on.

Joining - R Code

```
> vaxData <- vaccines %>% inner_join(beds) %>% inner_join(demographics)
```