## Project 1 Requirements

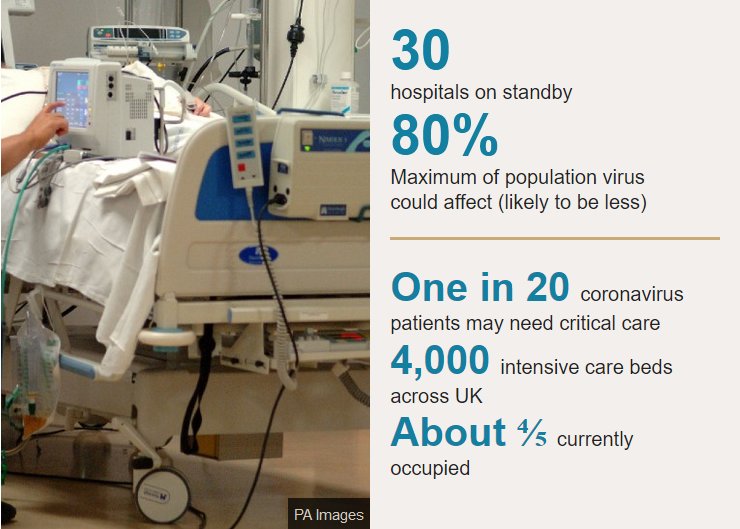


Image from: <https://www.bbc.com/news/health-51714498>

One of the unresolved questions from the ongoing Covid-19 pandemic is why different countries have such different proportions of deaths relative to the number of disease cases. There are several hypotheses about this - including differences in medical facilities (countries with more hospitals should see fewer deaths), age demographics (countries with a larger proportion of older people will see more deaths), and existing mortality rates between countries (indicating pre-existing differences in overall health between populations). The goal of this project is to use linear modeling to quantify some of the variation in mortality from Covid-19 in different countries due to differences in age demographics and hospital instructure.

### Data

1. <https://github.com/CSSEGISandData/COVID-19/tree/master/csse_covid_19_data/csse_covid_19_time_series>[[1]](#footnote-0)

You will only use the numbers of Confirmed and Deaths (not Recovered) for this project. Note that you can read the “raw” CSV file from a URL (the URL for the “raw” file) directly, like so:

read\_csv("https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse\_covid\_19\_data/csse\_covid\_19\_time\_series/time\_series\_covid19\_confirmed\_global.csv")

1. Hospital beds data
   * Download the Hospital bed density Data by country from the World Health Organization (WHO):
     + <http://apps.who.int/gho/data/view.main.HS07v>
   * This is the number of Hospital beds (per 10,000 population) in different countries.
2. demographics.csv[[2]](#footnote-1)
   * This gives the proportion of a country’s population in different age groups and some other demographic data such as mortality rates and expected lifetime. [On Titanium]

### Hints

There are two major steps in this project:

1. **Data preparation/wrangling** to get all the data into **one table** that can then be used for linear modeling
   1. reading the data using read\_csv()

**> deaths <- read\_csv("**[**https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse\_covid\_19\_data/csse\_covid\_19\_time\_series/time\_series\_covid19\_deaths\_global.csv**](https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/csse_covid_19_time_series/time_series_covid19_deaths_global.csv)**")**

**> confirmed <- read\_csv("**[**https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse\_covid\_19\_data/csse\_covid\_19\_time\_series/time\_series\_covid19\_confirmed\_global.csv**](https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/csse_covid_19_time_series/time_series_covid19_confirmed_global.csv)**")**

* 1. tidying tables if needed (to column-centric format). For example, the Covid-19 data is not tidy.

**> deaths <- deaths %>% pivot\_longer(-(1:4), names\_to="Day", values\_to="Deaths")**

**> deaths <- deaths %>% rename(`Country`= `Country/Region`)**

**> deaths <- deaths %>% group\_by(Country, Day) %>% summarize(Deaths=sum(Deaths))**

**> deaths <- deaths[order(as.Date(deaths$Day, format="%m/%d/%Y")),]**

**> deaths <- deaths %>% arrange(Country)**

**> confirmed <- confirmed %>% pivot\_longer(-(1:4), names\_to="Day", values\_to="Confirmed")**

**> confirmed <- confirmed %>% rename(`Country`= `Country/Region`)**

**> confirmed <- confirmed %>% group\_by(Country, Day) %>% summarize(Confirmed=sum(Confirmed))**

**> confirmed <- confirmed[order(as.Date(confirmed$Day, format="%m/%d/%Y")),]**

**> confirmed <- confirmed %>% arrange(Country)**

**> covid <- deaths %>% full\_join(confirmed)**

* 1. Discard data that is not needed. For example, only the number of hospital beds from the most recent year is necessary.

**Hospital Beds:**

**> beds <- read\_csv("C:/Users/username/Downloads/hospitalbeds.csv")**

**> beds <- beds %>% group\_by(Country) %>% filter(Year == max(Year))**

**> beds <- beds %>% select(-Year)**

**Demographics:**

**> demographics <- read\_csv("C:/Users/username/Downloads/demographics.csv")**

**> demographics <- demographics %>% select(`Country Name`, `Series Code`, YR2015)**

**> demographics <- demographics %>% pivot\_wider(names\_from = `Series Code`, values\_from = YR2015)**

**> demographics <- demographics %>% rename(`Country`= `Country Name`)**

* 1. Aggregating Covid-19 data to country level. Since the hospital beds and demographics only have country-level data, aggregate all state/province-level data.

**As done in part 1b:**

**> deaths <- deaths %>% group\_by(Country, Day) %>% summarize(Deaths=sum(Deaths))**

**> confirmed <- confirmed %>% group\_by(Country, Day) %>% summarize(Confirmed=sum(Confirmed))**

* 1. Since sex-related differences will not be examined in this project, add the male/female population numbers together.

**> demographics <- demographics %>% mutate("Population (Total)"=SP.POP.TOTL.FE.IN+SP.POP.TOTL.MA.IN, "Population (Urban)"=SP.URB.TOTL, "Population (80+)"=SP.POP.80UP.FE+SP.POP.80UP.MA, "Population (65+)"=SP.POP.65UP.FE.IN+SP.POP.65UP.MA.IN, "Population (15-64)"=SP.POP.1564.MA.IN+SP.POP.1564.FE.IN, "Population (0-14)"=SP.POP.0014.MA.IN+SP.POP.1564.FE.IN)**

**> demographics <- demographics %>% select(-(2:16))**

* 1. At the end of this step, the data should be in one table, in a format ready for linear regression:

**> covid <- covid %>% mutate(Country = replace(Country, Country == "Korea, South", "South Korea")) %>% mutate(Country = replace(Country, Country == "US", "United States")) %>% mutate(Country = replace(Country, Country == "Congo (Brazzaville)", "Congo")) %>% mutate(Country = replace(Country, Country == "Congo (Kinshasa)", "Congo"))**

**> beds <- beds %>% mutate(Country = replace(Country, Country == "Republic of Korea", "South Korea")) %>% mutate(Country = replace(Country, Country == "Iran (Islamic Republic of)", "Iran")) %>% mutate(Country = replace(Country, Country == "United Kingdom of Great Britain and Northern Ireland", "United Kingdom")) %>% mutate(Country = replace(Country, Country == "Bolivia (Plurinational State of)", "Bolivia")) %>% mutate(Country = replace(Country, Country == "United States of America", "United States")) %>% mutate(Country = replace(Country, Country == "Democratic Republic of the Congo", "Congo")) %>% mutate(Country = replace(Country, Country == "Venezuela (Bolivarian Republic of)", "Venezuela"))**

**> demographics <- demographics %>% mutate(Country = replace(Country, Country == "Korea, Dem. People's Rep.", "South Korea")) %>% mutate(Country = replace(Country, Country == "Korea, Rep.", "South Korea")) %>% mutate(Country = replace(Country, Country == "Iran, Islamic Rep.", "Iran")) %>% mutate(Country = replace(Country, Country == "Congo, Dem. Rep.", "Congo")) %>% mutate(Country = replace(Country, Country == "Congo, Rep.", "Congo")) %>% mutate(Country = replace(Country, Country == "Venezuela, RB", "Venezuela"))**

**> mydata <- covid %>% full\_join(beds) %>% full\_join(demographics)**



1. **Linear modeling the number of deaths** in a country. Make a list of all predictor variables that are available. The challenge is to identify which combination of these predictors will give the best predictive model and if transforming some of the variables will increase accuracy.

You should also try transforming some of the variables (e.g., transforming population counts to proportion of total population). Another possible transform is shifting the confirmed and deaths relative to each other (i.e., we know deaths lag cases). Run linear regression with at least 5 different combinations of predictor variables. Note:

* 1. The number of confirmed cases is obviously the most relevant predictor. Adding other variables may or may not give better models.
  2. Each day gives a new data point. For example, all cases upto 2020-03-01, all cases upto 2020-03-02, …
  3. The country name itself should not be a predictor - the goal is to identify what aspects of a country determine the death toll.

**Country names:**

The country names across datasets do not all match (e.g., "Iran" and "Iran, Islamic Rep."). Such mismatches are a common problem in data science and this should be fixed before joining tables. However, since the larger project goal is to learn a model that relates deaths to health-related factors in every country/every day, missing a few countries will not greatly reduce the data to build the model. So it is sufficient to fix only these countries' names which have been severely affected by the pandemic as we don't want to exclude their data, for example:

1. "Iran (Islamic Republic of)", "Iran", "Iran, Islamic Rep."
2. "South Korea", "Republic of Korea", "Korea, Rep."
3. "United Kingdom", "United Kingdom of Great Britain and Northern Ireland"

The names can be fixed by code like this:

mydata[mydata$Country == "Republic of Korea", "Country"] <- "South Korea"

or using the tidyverse style:

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

**USA data:**

Confirmed cases in the USA are no longer included in the global data files (they are in a separate file). So, for simplicity, you do not have to include USA data in your project. (Of course, you are free to include USA data and fix the name mismatches of every country.

**Variable names with spaces:**

The tidyverse functions allow you to use variable names with spaces in them by enclosing them in backquotes, like so: `Country Name`. Alternatively, you may find it more convenient to rename the variables to something more convenient, like so:

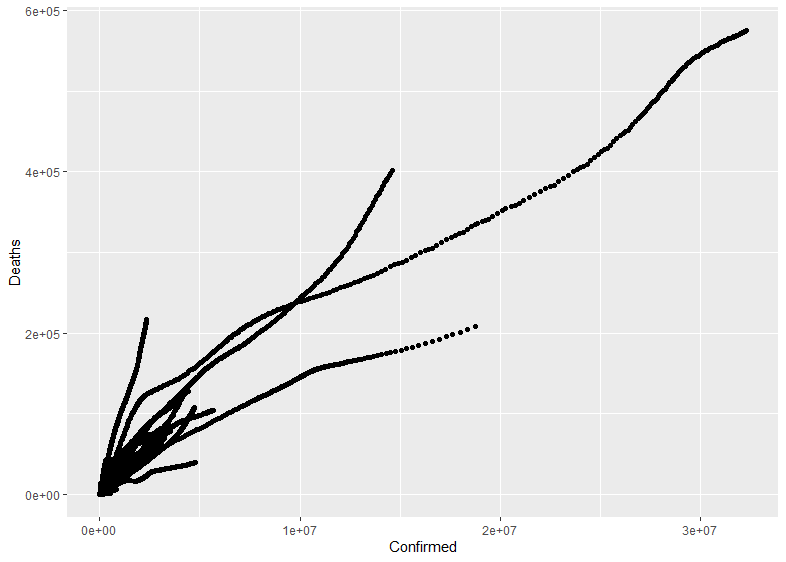
mydata %>% rename(HospitalBeds = `Hospital beds (per 10 000 population)`)

**Evaluating Confirmed Cases as a Predictor (to compare to following combinations):**

Dependent Variable: Deaths

Independent Variable: Confirmed Cases

**> ggplot(data=mydata)+ geom\_point(mapping = aes(x=Confirmed, y=Deaths), na.rm = TRUE)**



**> modConfirmed <- lm(data=mydata, formula=Deaths~Confirmed)**

**> summary(modConfirmed)**

**Call:**

**lm(formula = Deaths ~ Confirmed, data = mydata)**

**Residuals:**

**Min 1Q Median 3Q Max**

**-155366 -1224 -1195 -1140 170025**

**Coefficients:**

**Estimate Std. Error t value Pr(>|t|)**

**(Intercept) 1.195e+03 3.094e+01 38.63 <2e-16 \*\*\***

**Confirmed 1.932e-02 2.241e-05 862.09 <2e-16 \*\*\***

**---**

**Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1**

**Residual standard error: 9244 on 91870 degrees of freedom**

**(93 observations deleted due to missingness)**

**Multiple R-squared: 0.89, Adjusted R-squared: 0.89**

**F-statistic: 7.432e+05 on 1 and 91870 DF, p-value: < 2.2e-16**

**# The R2 value is close to one which means that this is a good model**

**# Equation: -2.939e+04 + 4.604e+-1 \* Deaths**

**# We can use the R2 values to compare how good the following combinations’ models are**

**# We can see that obviously the higher amount of confirmed cases there are the higher the death count usually is**

**First Combination:**

Dependent Variable: Deaths

Independent Variables: Confirmed Cases and Beds

**> modBeds <- lm(data=mydata, formula = Deaths~Confirmed + `Hospital beds (per 10 000 population)`)**

**> summary(modBeds)**

**Call:**

**lm(formula = Deaths ~ Confirmed + `Hospital beds (per 10 000 population)`,**

**data = mydata)**

**Residuals:**

**Min 1Q Median 3Q Max**

**-155090 -1426 -1387 -1313 169888**

**Coefficients:**

**Estimate Std. Error**

**(Intercept) 1.394e+03 5.471e+01**

**Confirmed 1.929e-02 2.420e-05**

**`Hospital beds (per 10 000 population)` -3.113e-01 1.553e+00**

**t value Pr(>|t|)**

**(Intercept) 25.48 <2e-16 \*\*\***

**Confirmed 797.27 <2e-16 \*\*\***

**`Hospital beds (per 10 000 population)` -0.20 0.841**

**---**

**Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1**

**Residual standard error: 9898 on 79805 degrees of freedom**

**(12157 observations deleted due to missingness)**

**Multiple R-squared: 0.8885, Adjusted R-squared: 0.8885**

**F-statistic: 3.179e+05 on 2 and 79805 DF, p-value: < 2.2e-16**

**Second Combination:**

Dependent Variable: Deaths

Independent Variables: Confirmed Cases and Urban Population

**> modUrban <- lm(data=mydata, formula = Deaths~Confirmed + `Population (Urban)`)**

**> summary(modUrban)**

**Call:**

**lm(formula = Deaths ~ Confirmed + `Population (Urban)`, data = mydata)**

**Residuals:**

**Min 1Q Median 3Q Max**

**-156444 -1382 -1153 -1112 169433**

**Coefficients:**

**Estimate Std. Error t value Pr(>|t|)**

**(Intercept) 1.118e+03 3.577e+01 31.24 <2e-16**

**Confirmed 1.910e-02 2.529e-05 755.44 <2e-16**

**`Population (Urban)` 1.222e-05 5.108e-07 23.92 <2e-16**

**(Intercept) \*\*\***

**Confirmed \*\*\***

**`Population (Urban)` \*\*\***

**---**

**Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1**

**Residual standard error: 9749 on 81661 degrees of freedom**

**(10301 observations deleted due to missingness)**

**Multiple R-squared: 0.8894, Adjusted R-squared: 0.8894**

**F-statistic: 3.283e+05 on 2 and 81661 DF, p-value: < 2.2e-16**

**Third Combination:**

Independent Variable: Deaths

Predictor Variables: Confirmed Cases, Ages 65+, and Population Total

**> modElderlyTotal <- lm(data=mydata, formula = Deaths~Confirmed + `Population (65+)` +`Population (Total)`)**

**> summary(modElderlyTotal)**

**Call:**

**lm(formula = Deaths ~ Confirmed + `Population (65+)` + `Population (Total)`,**

**data = mydata)**

**Residuals:**

**Min 1Q Median 3Q Max**

**-133541 -1581 -1362 -756 171159**

**Coefficients:**

**Estimate Std. Error t value Pr(>|t|)**

**(Intercept) 1.440e+03 3.585e+01 40.16 <2e-16**

**Confirmed 1.914e-02 2.499e-05 765.65 <2e-16**

**`Population (65+)` 5.795e-04 8.277e-06 70.02 <2e-16**

**`Population (Total)` -4.675e-05 6.702e-07 -69.76 <2e-16**

**(Intercept) \*\*\***

**Confirmed \*\*\***

**`Population (65+)` \*\*\***

**`Population (Total)` \*\*\***

**---**

**Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1**

**Residual standard error: 9647 on 78876 degrees of freedom**

**(13085 observations deleted due to missingness)**

**Multiple R-squared: 0.8952, Adjusted R-squared: 0.8952**

**F-statistic: 2.246e+05 on 3 and 78876 DF, p-value: < 2.2e-16**

**Fourth Combination:**

Dependent Variable: Deaths

Independent Variables: Confirmed Cases and Urban Population to Total Population Ratio

**> urbanratiodata <- mydata %>% mutate(UrbanTotalRatio = `Population (Urban)`/`Population (Total)`)**

**> modUrbanTotal <- lm(data=urbanratiodata, formula = Deaths~Confirmed + UrbanTotalRatio)**

**> summary(modUrbanTotal)**

**Call:**

**lm(formula = Deaths ~ Confirmed + UrbanTotalRatio, data = urbanratiodata)**

**Residuals:**

**Min 1Q Median 3Q Max**

**-151350 -2304 -1073 533 168668**

**Coefficients:**

**Estimate Std. Error t value Pr(>|t|)**

**(Intercept) -2.550e+03 9.647e+01 -26.43 <2e-16 \*\*\***

**Confirmed 1.919e-02 2.419e-05 793.29 <2e-16 \*\*\***

**UrbanTotalRatio 6.830e+03 1.559e+02 43.82 <2e-16 \*\*\***

**---**

**Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1**

**Residual standard error: 9832 on 78877 degrees of freedom**

**(13085 observations deleted due to missingness)**

**Multiple R-squared: 0.8912, Adjusted R-squared: 0.8912**

**F-statistic: 3.229e+05 on 2 and 78877 DF, p-value: < 2.2e-16**

**Fifth Combination:**

Dependent Variable: Deaths

Independent Variable: Confirmed + Ages 0-14

**> modYoung <- lm(data=mydata, formula = Deaths~Confirmed + `Population (0-14)`)**

**> summary(modYoung)**

**Call:**

**lm(formula = Deaths ~ Confirmed + `Population (0-14)`, data = mydata)**

**Residuals:**

**Min 1Q Median 3Q Max**

**-152736 -1509 -1466 -1265 169964**

**Coefficients:**

**Estimate Std. Error t value Pr(>|t|)**

**(Intercept) 1.494e+03 3.695e+01 40.44 <2e-16**

**Confirmed 1.940e-02 2.545e-05 762.10 <2e-16**

**`Population (0-14)` -7.199e-06 5.435e-07 -13.24 <2e-16**

**(Intercept) \*\*\***

**Confirmed \*\*\***

**`Population (0-14)` \*\*\***

**---**

**Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1**

**Residual standard error: 9940 on 78877 degrees of freedom**

**(13085 observations deleted due to missingness)**

**Multiple R-squared: 0.8888, Adjusted R-squared: 0.8887**

**F-statistic: 3.151e+05 on 2 and 78877 DF, p-value: < 2.2e-16**

**Results:**

**> myresults <- tibble(model=c("Deaths~Confirmed", "Deaths~Confirmed+Beds", "Deaths~Confirmed+Urban", "Deaths~Confirmed+65UP+Total", "Deaths~Confirmed+Urban/Total", "Deaths~Confirmed+0-14"), R2 = c(0.89, 0.8885, 0.8894, 0.8952, 0.8912, 0.8888))**

**> myresults**

**# A tibble: 6 x 2**

**model R2**

**<chr> <dbl>**

**1 Deaths~Confirmed 0.89**

**2 Deaths~Confirmed+Beds 0.888**

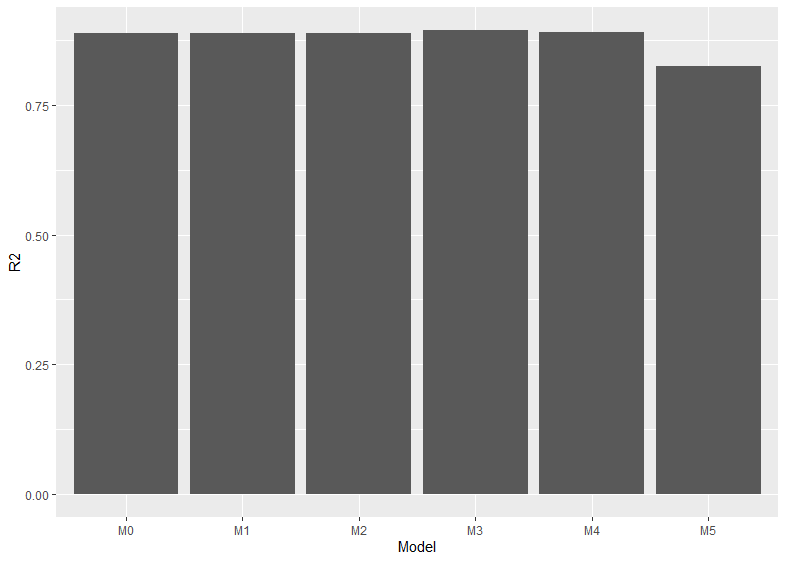
**3 Deaths~Confirmed+Urban 0.889**

**4 Deaths~Confirmed+65UP+Total 0.895**

**5 Deaths~Confirmed+Urban/Total 0.891**

**6 Deaths~Confirmed+0-14 0.889**

**> ggplot(data=myresults) + geom\_col(mapping = aes(x=model, y=R2)) + labs(x="Model")**

****

**# M0-M5 follow the the myresults tibble names respectively**

### Submission

1. Write a short report describing your data wrangling steps and the different combinations of predictor variables you tried, and any variable transforms. The report should include a summary plot that shows the evaluation metric of the different models (i.e., a comparison of the R2 values). There should be a conclusion that describes in words the implication of your most accurate model. [A PDF file]
2. A listing of your R code [.R file]

### Due date

Friday 4/23, 7pm on Titanium. Submit two files: the PDF report, full code (.R).

### Group work

You may work in groups of 1-3. Include all group member names in the PDF file. Only one person in the group needs to submit to Titanium.

### Acknowledgement

Idea based on discussions with Brian Chung.

1. More information: <https://github.com/CSSEGISandData/COVID-19> [↑](#footnote-ref-0)
2. Original dataset: <https://databank.worldbank.org/source/population-estimates-and-projections/Type/TABLE/preview/on#> [↑](#footnote-ref-1)