ndi\_analysis

Harris Policy Lab

2/16/2021

Merging datasets

When reviewing the data, a helpful link for issues: <https://github.com/ropensci/skimr/issues/606>

# Controls for COVID outcomes  
df\_covid\_controls <- df\_all\_yr %>%  
 arrange(country\_standard, year) %>%  
 group\_by(country\_standard) %>%  
 # These variables don't have values in 2020. Filling in from most recent year  
 fill(  
 c(gdp, pub\_health\_exp\_percent\_gdp, percap\_domestic\_health\_expenditure)) %>%  
 filter(year==2020) %>%  
 select(  
 country\_standard,   
 gdp,   
 pub\_health\_exp\_percent\_gdp,   
 percap\_domestic\_health\_expenditure)  
   
df\_covid\_controls <- df\_all\_yr\_2020 %>%  
 select(  
 country\_standard, gdp\_per\_capita, gini\_2020, median\_age, aged\_65\_older) %>%  
 full\_join(df\_covid\_controls, by = "country\_standard") %>%   
 select(  
 country\_standard, gdp, gdp\_per\_capita, gini\_2020,   
 # we discussed using "public health expenditure as a proportion of GDP"  
 # as an indicator, but it seemed very similar to the next indicator  
 # below, which is "public health expenditure per capita"  
 #pub\_health\_exp\_percent\_gdp,   
 percap\_domestic\_health\_expenditure,   
 median\_age, aged\_65\_older)  
  
write\_csv(df\_covid\_controls, "data/covid-19\_controls.csv")

# Creating an index for COVID-19 health factors

Our C index comes from the country-level amounts since the start of the pandemic, used in the following equation: (1 / COVID-19 cases per million) \* (1 / COVID-19 deaths per million) \* (total COVID-19 tests per thousand)

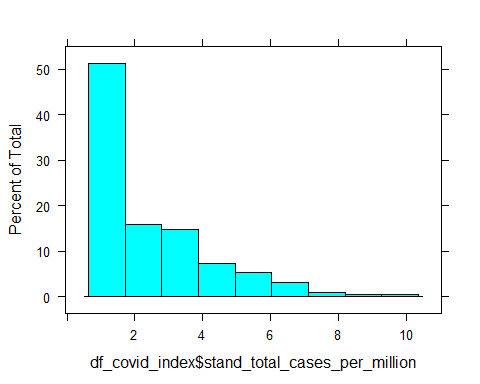
The intuition is that countries will have a higher index score if they have fewer cases, fewer deaths, and more tests. The index is scaled from 0 (worst) to 1 (best).

We decided on this index after testing it among six considered models. We regressed each index on our list of control variables and compared the F-statistic of each model. The chosen model had the second highest F-Statistic, but was chosen because a review of the country rankings of the model with the highest F-statistic revealed it to be a poor index.

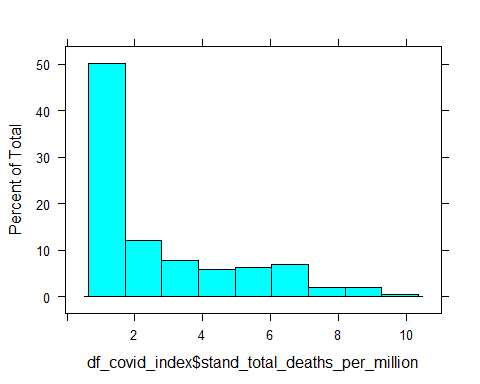
To see the code for the process above, see reference\_dode/covid\_index\_select.r

For countries that were missing data on COVID-19 tests, they are assigned a score of “1” on the scale of 1 to 10 used for countries’ testing rates.

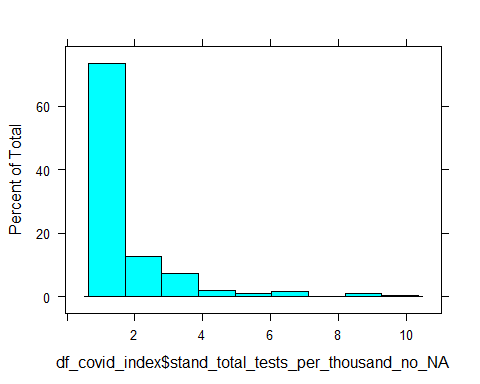
df\_covid\_index <- df\_all\_yr %>%  
 select(  
 country\_standard,   
 cum\_total\_cases\_per\_million,   
 cum\_total\_deaths\_per\_million,  
 cum\_total\_tests\_per\_thousand) %>%  
 filter(  
 if\_any(  
 cum\_total\_cases\_per\_million:cum\_total\_deaths\_per\_million,   
 ~ !is.na(.))) %>%   
 distinct() %>%  
 mutate(  
 # Use a scale of 1 - 10 to avoid having "0" in denominator  
 stand\_total\_cases\_per\_million = rescale(cum\_total\_cases\_per\_million, to = c(1, 10)),  
 stand\_total\_deaths\_per\_million = rescale(cum\_total\_deaths\_per\_million, to = c(1, 10)),  
 stand\_total\_tests\_per\_thousand = rescale(cum\_total\_tests\_per\_thousand, to = c(1, 10)),  
 stand\_total\_tests\_per\_thousand\_no\_NA = ifelse(  
 is.na(stand\_total\_tests\_per\_thousand),   
 1,   
 stand\_total\_tests\_per\_thousand),  
 covid\_index = rescale(  
 (1/stand\_total\_cases\_per\_million) +  
 (1/stand\_total\_deaths\_per\_million) +  
 stand\_total\_tests\_per\_thousand\_no\_NA)  
 ) %>%  
 select(country\_standard, covid\_index, everything())  
  
# Reviewing distribution of the data  
histogram(df\_covid\_index$stand\_total\_cases\_per\_million)



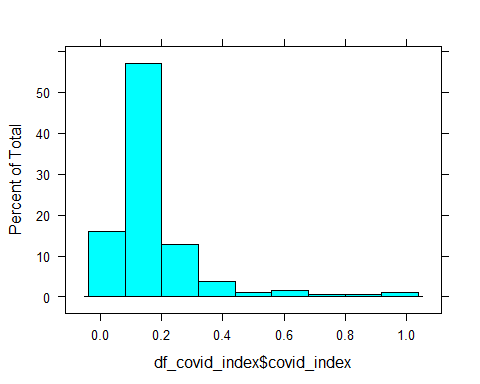
histogram(df\_covid\_index$stand\_total\_deaths\_per\_million)



histogram(df\_covid\_index$stand\_total\_tests\_per\_thousand\_no\_NA)



histogram(df\_covid\_index$covid\_index)



# RUnning a regression of control variables on COVID-19 index  
covid\_index\_mod\_test <- df\_covid\_index %>%   
 select(country\_standard, covid\_index) %>%  
 full\_join(df\_covid\_controls, by = "country\_standard") %>%  
 select(-country\_standard)  
  
 lm(covid\_index ~ ., data = covid\_index\_mod\_test) %>% summary()

##   
## Call:  
## lm(formula = covid\_index ~ ., data = covid\_index\_mod\_test)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.34923 -0.06285 0.00081 0.03053 0.61423   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 3.574e-01 1.275e-01 2.803 0.00573 \*\*  
## gdp -5.506e-15 5.367e-15 -1.026 0.30652   
## gdp\_per\_capita 3.414e-06 1.068e-06 3.197 0.00169 \*\*  
## gini\_2020 -2.266e-01 1.578e-01 -1.436 0.15316   
## percap\_domestic\_health\_expenditure 2.867e-05 1.557e-05 1.842 0.06741 .   
## median\_age -1.851e-03 3.971e-03 -0.466 0.64180   
## aged\_65\_older -3.558e-03 5.721e-03 -0.622 0.53491   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.1299 on 151 degrees of freedom  
## (36 observations deleted due to missingness)  
## Multiple R-squared: 0.2895, Adjusted R-squared: 0.2613   
## F-statistic: 10.26 on 6 and 151 DF, p-value: 1.64e-09

write\_csv(df\_covid\_index, "data/covid-19\_index.csv")

You can do python code, like in example chunk below.

## Notes to self:

We don’t have 2020 GDP data

There are about 20 countries missing Gini index data

From OWID Covid, look at “stringeny\_index”

Additional ideas for controls:

* df\_covid\_owid
  + “population\_density”
  + “gdp\_per\_capita”
  + “extreme\_poverty”
  + “cardiovasc\_death\_rate”
  + “diabetes\_prevalence”
  + “female\_smokers”
  + “male\_smokers”
  + “handwashing\_facilities”
  + “hospital\_beds\_per\_thousand”
  + “life\_expectancy”
  + “human\_development\_index”

## Resources:

## Citations:

This project uses the countrycode R package

Arel-Bundock, Vincent, Nils Enevoldsen, and CJ Yetman, (2018). countrycode: An R package to convert country names and country codes. Journal of Open Source Software, 3(28), 848, <https://doi.org/10.21105/joss.00848>