Assignment 2

Prof. Kamis

Introduction

You should use this code as part of your Assignment 2 assignment. You will re-save this Notebook under a different name and modify it to complete the assignment. For example, you should delete all of the text in this section and replace it with your own introduction.

The first chunk calls the packages that you will need in this assignment.

The second code chunk automatically retrieves the latest data from the World Development Indicators database, for use in the assignment.

install.packages("leaflet") install.packages("WDI") install.package("date") ## Step 1: library calls to load packages

```
library(tidyverse)
library(leaflet)
library(WDI)
library(lubridate)
library(reshape2)
library(readr)
library(leaflet)
```

Step 2: Call package `WDI` to retrieve most updated figures available.

In this assignment, we will fetch ten data series from the WDI:

Tableau Name	WDI Series
Birth Rate	SP.DYN.CBRT.IN
Infant Mortality Rate	SP.DYN.IMRT.IN
Internet Usage	IT.NET.USER.ZS
Life Expectancy (Total)	SP.DYN.LE00.IN
Forest Area (% of land)	AG.LND.FRST.ZS
Mobile Phone Usage	IT.CEL.SETS.P2
Population Total	SP.POP.TOTL
International Tourism receipts (current US$)	ST.INT.RCPT.CD
Import value index (2000=100)	TM.VAL.MRCH.XD.WD
Export value index (2000=100)	TX.VAL.MRCH.XD.WD

The next code chunk will call the WDI API and fetch the years 1998 through 2018, as available. You will find that only a few variables have data for 2018. The dataframe will also contain the longitude and latitude of the capital city in each country.

Note This notebook will take approximately 2 minutes to run. The WDI call is time-consuming as is the process of knitting the file. Be patient.

The World Bank uses a complex, non-intuitive scheme for naming variables. For example, the Birth Rate series is called `SP.DYN.CBRT,IN` . The code assigns variables names that are more intuitive than the codes assigned by the World Bank, and converts the geocodes from factors to numbers.

In your code, you will use the data frame called `countries` .

```
birth <- "SP.DYN.CBRT.IN"
infmort <- "SP.DYN.IMRT.IN"
net <-"IT.NET.USER.ZS"
lifeexp <- "SP.DYN.LE00.IN"
forest <- "AG.LND.FRST.ZS"
mobile <- "IT.CEL.SETS.P2"
pop <- "SP.POP.TOTL"
tour <- "ST.INT.RCPT.CD"
import <- "TM.VAL.MRCH.XD.WD"
export <- "TX.VAL.MRCH.XD.WD"

# create a vector of the desired indicator series
indicators <- c(birth, infmort, net, lifeexp, forest,
                mobile, pop, tour, import, export)

countries <- WDI(country="all", indicator = indicators,
     start = 1998, end = 2018, extra = TRUE)

## rename columns for each of reference
countries <- rename(countries, birth = SP.DYN.CBRT.IN,
       infmort = SP.DYN.IMRT.IN, net  = IT.NET.USER.ZS,
       lifeexp = SP.DYN.LE00.IN, forest = AG.LND.FRST.ZS,
       mobile = IT.CEL.SETS.P2, pop = SP.POP.TOTL,
       tour = ST.INT.RCPT.CD, import = TM.VAL.MRCH.XD.WD,
       export = TX.VAL.MRCH.XD.WD)

# convert geocodes from factors into numerics

countries$lng <- as.numeric(as.character(countries$longitude))
countries$lat <- as.numeric(as.character(countries$latitude))

# Remove groupings, which have no geocodes
countries <- countries %>%
   filter(!is.na(lng))
```

A Glimpse of the new dataframe

```
glimpse(countries)
```

```
## Observations: 4,410
## Variables: 22
## $ iso2c     <chr> "AD", "AD", "AD", "AD", "AD", "AD", "AD", "AD", "AD", "AD",…
## $ country   <chr> "Andorra", "Andorra", "Andorra", "Andorra", "Andorra", "And…
## $ year      <int> 2018, 2007, 2004, 2005, 2017, 1998, 1999, 2000, 2006, 2002,…
## $ birth     <dbl> 7.200, 10.100, 10.900, 10.700, NA, 11.900, 12.600, 11.300, …
## $ infmort   <dbl> 2.7, 4.5, 5.1, 4.9, 2.8, 6.4, 6.2, 5.9, 4.7, 5.5, 5.3, 4.3,…
## $ net       <dbl> NA, 70.870000, 26.837954, 37.605766, 91.567467, 6.886209, 7…
## $ lifeexp   <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,…
## $ forest    <dbl> NA, 34.042553, 34.042553, 34.042553, NA, 34.042553, 34.0425…
## $ mobile    <dbl> 107.28255, 76.80204, 76.55160, 81.85933, 104.33241, 22.0089…
## $ pop       <dbl> 77006, 82684, 76244, 78867, 77001, 64142, 64370, 65390, 809…
## $ tour      <dbl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA,…
## $ import    <dbl> 136.50668, 190.30053, 174.09246, 178.06349, 146.27331, 105.…
## $ export    <dbl> 268.35043, 332.78037, 271.81148, 314.89205, 264.92993, 127.…
## $ iso3c     <fct> AND, AND, AND, AND, AND, AND, AND, AND, AND, AND, AND, AND,…
## $ region    <fct> Europe & Central Asia, Europe & Central Asia, Europe & Cent…
## $ capital   <fct> Andorra la Vella, Andorra la Vella, Andorra la Vella, Andor…
## $ longitude <fct> 1.5218, 1.5218, 1.5218, 1.5218, 1.5218, 1.5218, 1.5218, 1.5…
## $ latitude  <fct> 42.5075, 42.5075, 42.5075, 42.5075, 42.5075, 42.5075, 42.50…
## $ income    <fct> High income, High income, High income, High income, High in…
## $ lending   <fct> Not classified, Not classified, Not classified, Not classif…
## $ lng       <dbl> 1.5218, 1.5218, 1.5218, 1.5218, 1.5218, 1.5218, 1.5218, 1.5…
## $ lat       <dbl> 42.5075, 42.5075, 42.5075, 42.5075, 42.5075, 42.5075, 42.50…
```

#Graphing and Comments

Beyond this line, you will insert your original code, following the instructions in the assignment.

Plot from Phase 1

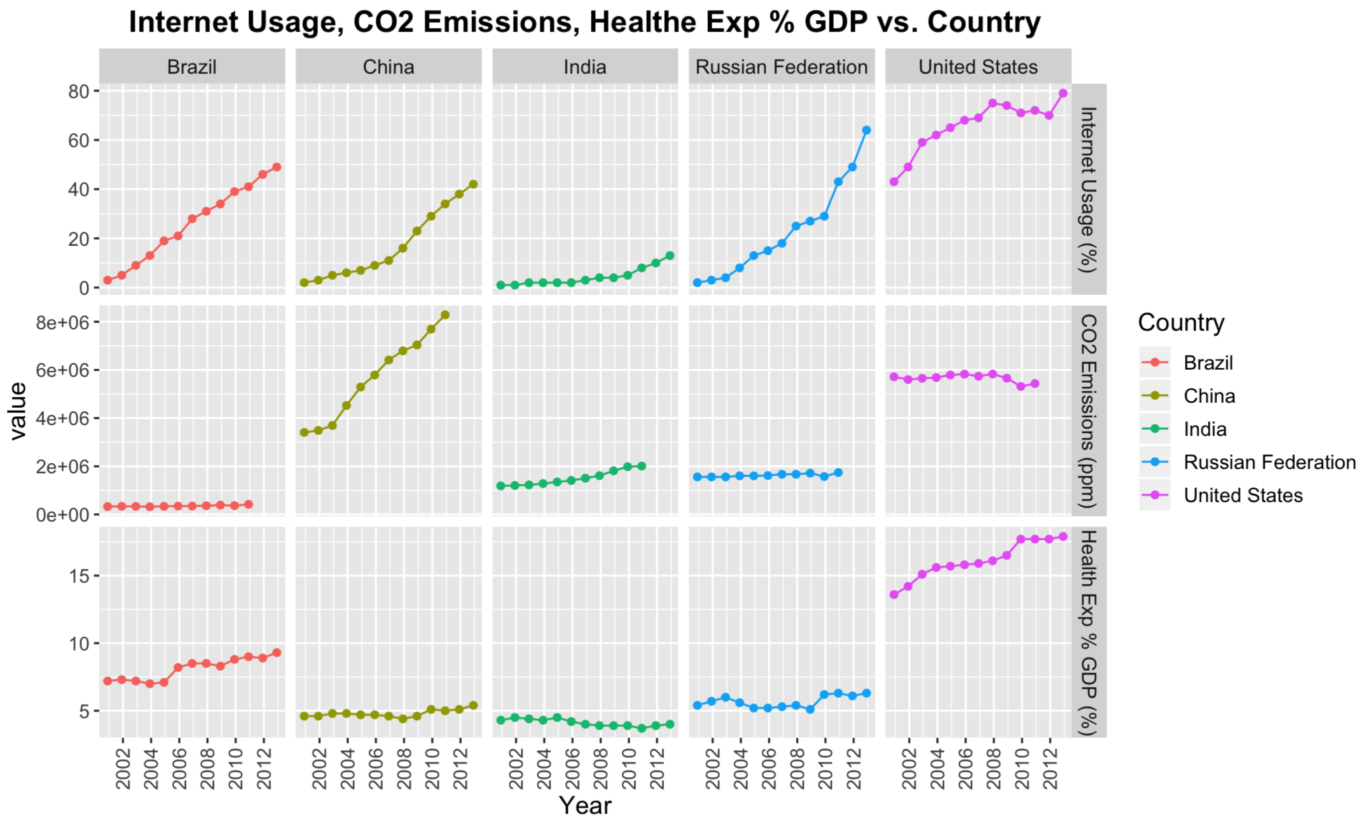
Replace this text chunk with an explanation of what you have done.

```
World_Indicators <- read_csv("World Indicators.csv",
    col_types = cols(`Health Exp % GDP` = col_number(),
        `Internet Usage` = col_number(),
        Year = col_date(format = "%m/%d/%Y")))


colnames(World_Indicators)[which(names(World_Indicators) == "Internet Usage")] <- "In
ternet Usage (%)"
colnames(World_Indicators)[which(names(World_Indicators) == "CO2 Emissions")] <- "CO2
Emissions (ppm)"
colnames(World_Indicators)[which(names(World_Indicators) == "Health Exp % GDP")] <-
"Health Exp % GDP (%)"


Phase1_data<- World_Indicators %>% filter(Country %in% c("United States", "Brazil","R
ussian Federation", "India", "China")) %>% melt( measure.vars = c( "Internet Usage
 (%)", "CO2 Emissions (ppm)", "Health Exp % GDP (%)"))


ggplot(Phase1_data, aes(x= Year, y = value, color= Country))+
  geom_point()+
  geom_line()+
  facet_grid(variable~Country, scale = "free") +
  theme(axis.text.x =element_text(size = 10, angle = 90, hjust = 1, vjust =.5  ),
        axis.text.y = element_text(size =10), text = element_text(size =13),
        plot.title = element_text(face="bold",hjust = 0.5))+
  ggtitle("Internet Usage, CO2 Emissions, Healthe Exp % GDP vs. Country")
```



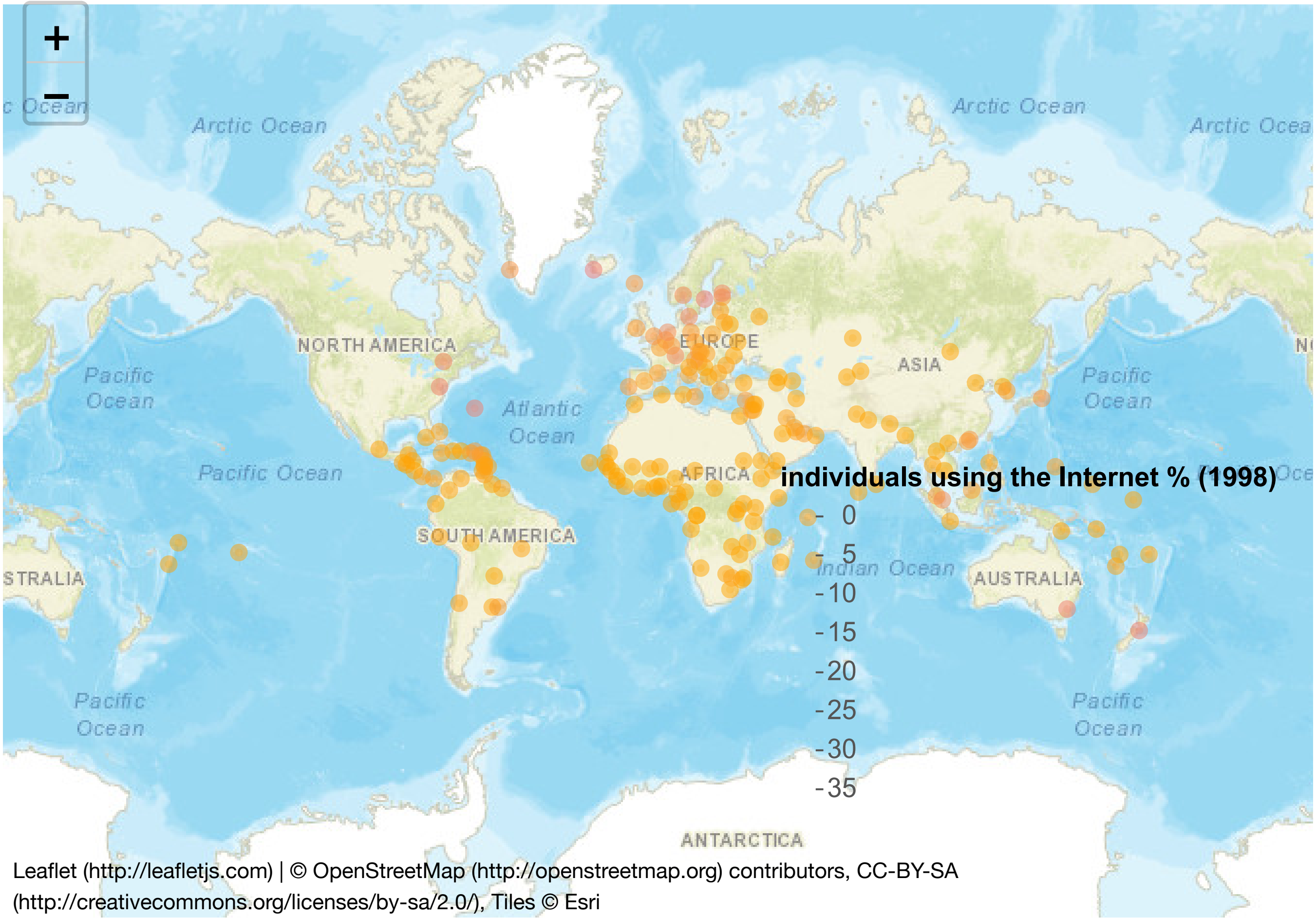
## World map showing a variable in 1998

Replace this text chunk with an explanation of what you have done, and what you notice in this map.

```
#1998
countries <- countries %>% filter(!is.na(net))

pal = colorNumeric(palette = c('orange', 'purple'), domain = c(min(countries$net) : m
ax(countries$net)))

leaflet(data = countries[countries$year==1998,]) %>%
   addTiles() %>%
   addProviderTiles('Esri') %>%
   addCircleMarkers(lng = ~lng, lat = ~lat,
                    radius = 2,
                    color = ~pal(net),
                    popup = ~paste0('<b> Country: </b>', as.character(country), '<br/>
<b> Internet Usage: </b>', round(net, 2), '%'),
                    labelOptions = labelOptions(noHide = T, textOnly = TRUE)) %>%
  addLegend(
     pal = pal,
     values = ~net,
     opacity = 1,
     title = "individuals using the Internet % (1998)",
     position = "bottomright")
```

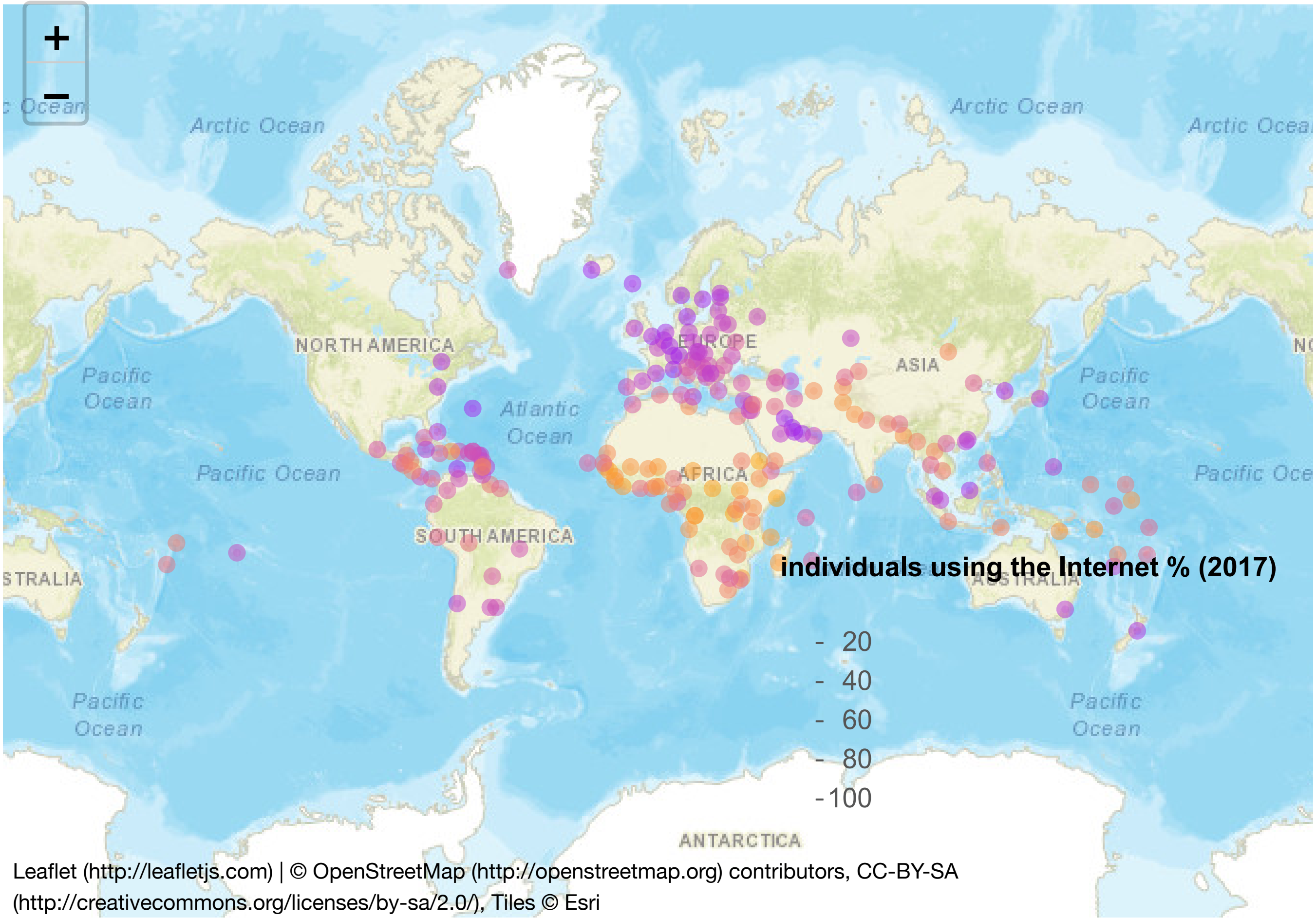


Leaflet (http://leafletjs.com) | © OpenStreetMap (http://openstreetmap.org) contributors, CC-BY-SA (http://creativecommons.org/licenses/by-sa/2.0/), Tiles © Esri

World map showing the same variable recently

Replace this text chunk with an explanation of what you have done, and what you notice between 1998 and the recent year.

```
leaflet(data = countries[countries$year==2017,]) %>%
   addTiles() %>%
   addProviderTiles('Esri') %>%
   addCircleMarkers(lng = ~lng, lat = ~lat,
                    radius = 2,
                    color = ~pal(net),
                    popup = ~paste0('<b> Country: </b>', as.character(country), '<br/>
<b> Internet Usage: </b>', round(net, 2), '%'),
                    labelOptions = labelOptions(noHide = T, textOnly = TRUE)) %>%
  addLegend(
     pal = pal,
     values = ~net,
     opacity = 1,
     title = "individuals using the Internet % (2017)",
     position = "bottomright")
```



Leaflet (http://leafletjs.com) | © OpenStreetMap (http://openstreetmap.org) contributors, CC-BY-SA (http://creativecommons.org/licenses/by-sa/2.0/), Tiles © Esri