PM1toPM3

PM1

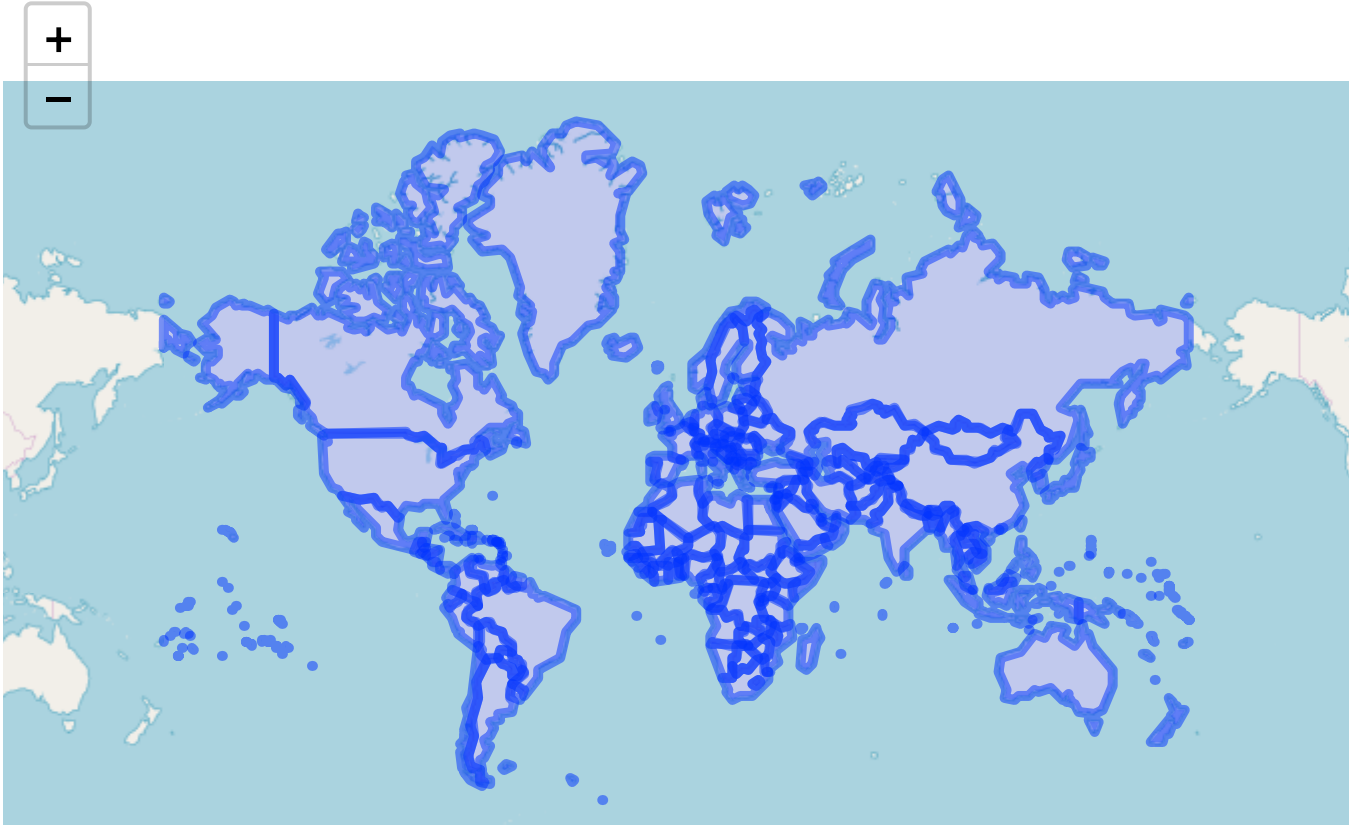
17/04/2019

```
# Get Cait dataset and join to map
cait <-read.csv("~/Downloads/CAIT Country CO2 Emissions.csv",
                skip = 1,stringsAsFactors = FALSE) %>%
  dplyr::rename(CO2 = 3) %>%
  # Shorten the name of the third column.
  filter(Year == "2014", !(Country%in% c("European Union (28)", "World"))) %>%
  mutate(Country =gsub("Micronesia","Federated States of Micronesia",Country),
         Code =countrycode(Country, "country.name", "iso3c"))

spdf <-joinCountryData2Map(cait, nameJoinColumn = "Code") %>%
  subset(continent!="Antarctica")
```

```
## 191 codes from your data successfully matched countries in the map
## 0 codes from your data failed to match with a country code in the map
## 52 codes from the map weren't represented in your data
```

```
# Basicest of leaflets - plot countries as ugly polygons
leaflet(spdf) %>%
  addTiles() %>%
  addPolygons()
```

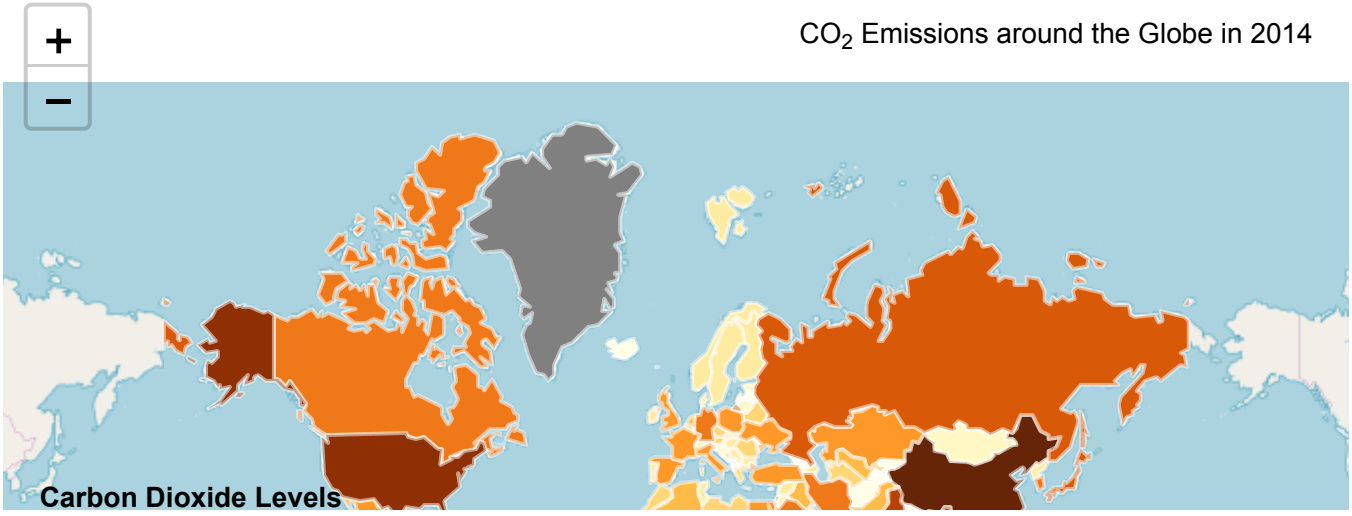


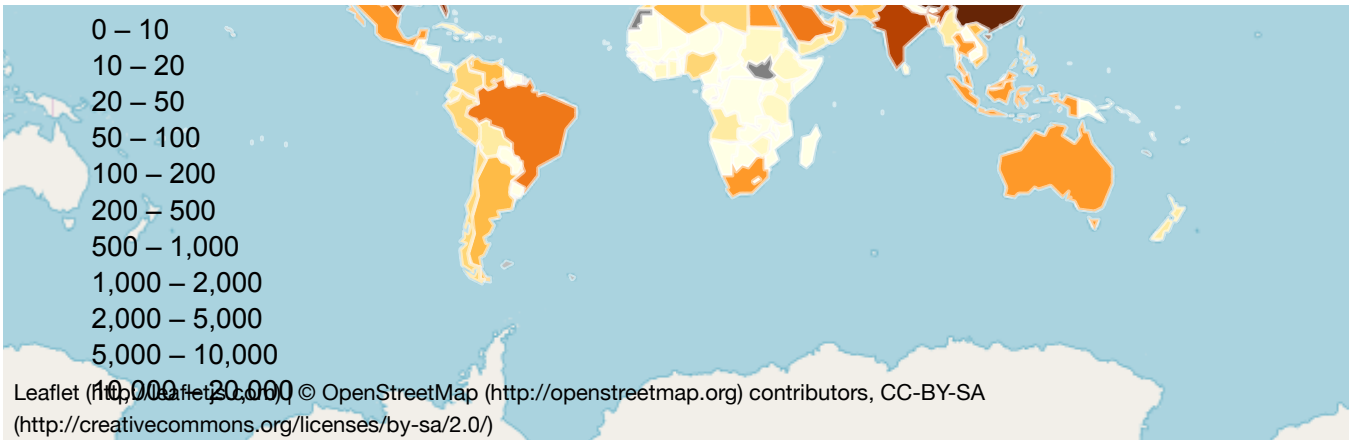

```
# prettify the polygons - show the CO2 level info with
# nice breaks
breaks <-c(0, 10, 20, 50, 100, 200, 500, 1000, 2000, 5000, 10000, 20000)
crp <-colorRampPalette(brewer.pal(9, "YlOrBr"))
binpal <- colorBin(crp(length(breaks)-1), spdf$CO2, bins = breaks, pretty = FALSE)

# HTML labels to enable subscripts, boldface, and new lines
labels <-  sprintf("<strong>%s</strong> <br/> CO<sub>2</sub> levels: %s",
                   cait$Country, cait$CO2) %>%
  lapply(htmltools::HTML)


header <- tags$div(
  HTML('CO<sub>2</sub> Emissions around the Globe in 2014')
)

# Plot: change borders, add highlight, labels, and legend
co2map <- leaflet(spdf) %>%
  addTiles() %>%
  addPolygons(fillColor = ~binpal(CO2),
              fillOpacity = 1,
              color = "white",
              dashArray = "1",
              weight = 2,
              label = labels,
              highlight = highlightOptions(color = "black",
                                           bringToFront = TRUE,
                                           fillOpacity = 0.7)) %>%
  addLegend(pal = binpal, values = CO2,
            title = "Carbon Dioxide Levels",
            position = "bottomleft") %>%
  addControl(header, position = "topright")
co2map
```





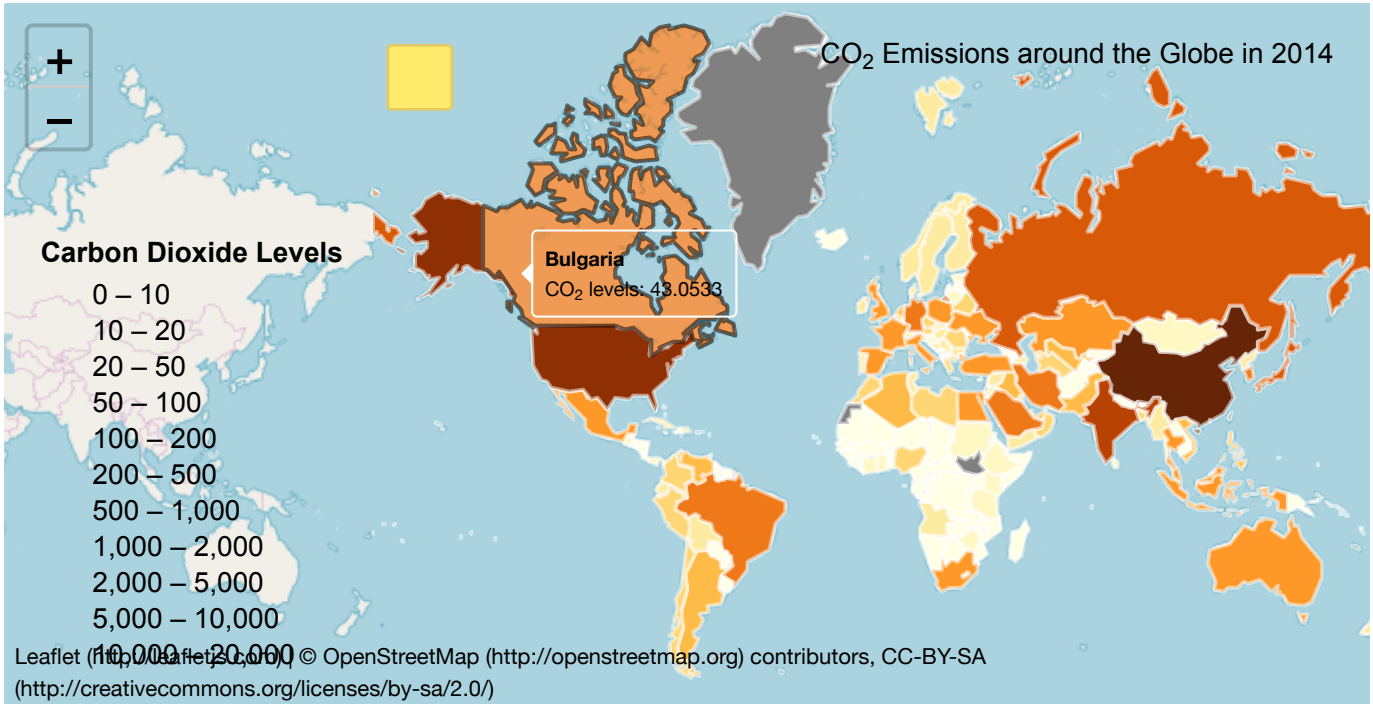

```
# long thing to produce widget with set height.
library(shiny)

ui <- fluidPage(
  leafletOutput("mymap", height=350),
  p(),
  actionButton("recalc", "New points")
)

server <- function(input, output, session) {

  points <- eventReactive(input$recalc, {
    cbind(rnorm(40) * 2 + 13, rnorm(40) + 48)
  }, ignoreNULL = FALSE)

  output$mymap <- renderLeaflet({
    co2map
  })
}
shinyApp(ui, server)
```


CO_2 Emissions around the Globe in 2014
Carbon Dioxide Levels
0 – 10
10 – 20
20 – 50
50 – 100
100 – 200
200 – 500
500 – 1,000
1,000 – 2,000
2,000 – 5,000
5,000 – 10,000
10,000 – 20,000
Bulgaria
CO_2 levels: 43.0533
Leaflet (http://leafletjs.com) | © OpenStreetMap (http://openstreetmap.org) contributors, CC-BY-SA (http://creativecommons.org/licenses/by-sa/2.0/)

New points