Maps in R: Exercises

These exercises accompany the Maps in R tutorial: http://rpubs.com/NateByers/Maps. These exercises use daily No2 data for the United States in 2014. Run the following code to clean out your global environment and load the data you need:

rm(list = ls())  
library(dplyr)  
library(region5air)  
data(no2\_2014)

If you are curious about where this data came from, see the [Download](#download) section below.

## Exercises

1. Use dplyr to create a data frame called no2\_max. It should be a filtered subset of the no2\_2014 data frame so that each monitor has one record with the highest 1-hour NO2 maximum for the entire year. **Hint:** Take a look at the column names, names(no2\_2014). Use group\_by() to group by state, county, and site. Use filter() to find which 1st-max-value is equal to the maximum of the 1st-max-value column (use ==).

[Solution 1](#ex1)

1. Make a map of NO2 monitors. Use map(database = 'usa') to make a map of the US first. Then use points() to add the monitors from the no2\_max data frame.

[Solution 2](#ex2)

1. Use filter() to make a new data frame of NO2 motors that had a 1 hour max value above 100 ppb. Call the new data frame no2\_violations and map those monitors.

[Solution 3](#ex3)

1. Use the leaflet package to make an interactive map of the monitors in the no2\_violations data frame.

[Solution 4](#ex4)

#### Advanced Exercise

1. Use the choroplethr package to make a heat map of the number of NO2 monitors in each state. **Hint:** You will need to make a new data frame called no2\_monitors. Start with the no2\_max data frame and group by state. Use the summarize() function to get the total number of monitors in each group. The n() function will count the total number of rows in a group.

[Solution 5](#ex5)

## Solutions

#### Solution 1

Here are the columns.

names(no2\_2014)

## [1] "State.Code" "County.Code" "Site.Num"   
## [4] "Parameter.Code" "POC" "Latitude"   
## [7] "Longitude" "Datum" "Parameter.Name"   
## [10] "Sample.Duration" "Pollutant.Standard" "Date.Local"   
## [13] "Units.of.Measure" "Event.Type" "Observation.Count"   
## [16] "Observation.Percent" "Arithmetic.Mean" "X1st.Max.Value"   
## [19] "X1st.Max.Hour" "AQI" "Method.Code"   
## [22] "Method.Name" "Local.Site.Name" "Address"   
## [25] "State.Name" "County.Name" "City.Name"   
## [28] "CBSA.Name" "Date.of.Last.Change"

Now we know the column names for state, county, and site. Let's group by those columns.

no2\_max <- group\_by(no2\_2014, State.Name, County.Name, Site.Num)

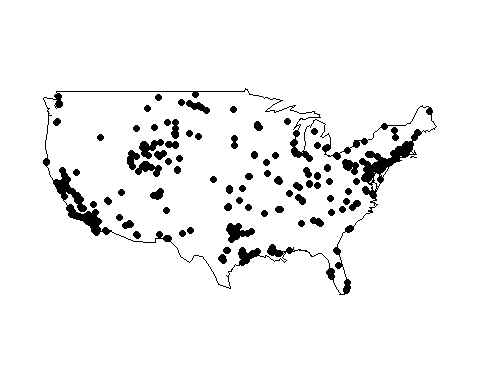
Now we'll filter the data frame so that the "X1st.Max.Value" is equivalent to the maximum for each group.

no2\_max <- filter(no2\_max, X1st.Max.Value == max(X1st.Max.Value, na.rm = TRUE))

[Back to exercises](#ex)

#### Solution 2

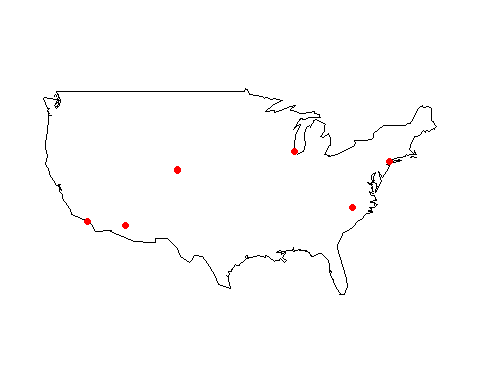
library(maps)  
map(database = 'usa')  
points(x = no2\_max$Longitude, y = no2\_max$Latitude, pch = 19)



[Back to exercises](#ex)

#### Solution 3

no2\_violations <- filter(no2\_max, X1st.Max.Value > 100)  
map(database = 'usa')  
points(x = no2\_violations$Longitude, y = no2\_violations$Latitude, pch = 19,  
 col = "red")



[Back to exercises](#ex)

#### Solution 4

library(leaflet)  
m <- leaflet()  
m <- addTiles(m)  
m <- addMarkers(m, lng=no2\_violations$Longitude, lat=no2\_violations$Latitude,   
 popup=no2\_violations$CBSA.Name)  
m

[Back to exercises](#ex)

#### Solution 5

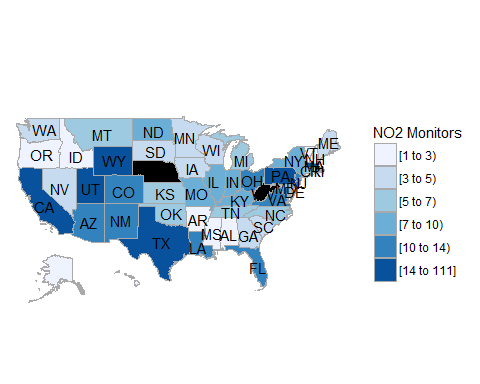
no2\_monitors <- group\_by(no2\_max, State.Name)  
no2\_monitors <- summarize(no2\_monitors, value = n())

The choroplethr package wants full state names with lower case. So we need to remove Puerto Rico, turn the state values to all lower case, and rename the state column to "region"

no2\_monitors <- filter(no2\_monitors, State.Name != "Puerto Rico")  
no2\_monitors <- mutate(no2\_monitors, State.Name = tolower(State.Name))  
no2\_monitors <- rename(no2\_monitors, region = State.Name)

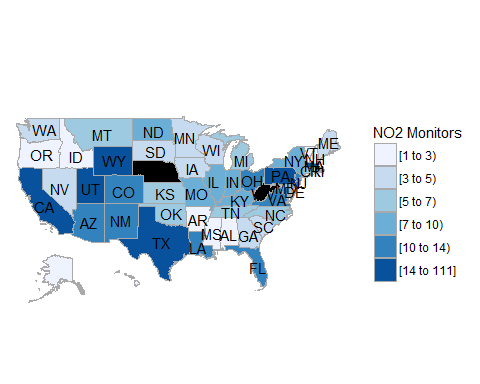
Now we can make the map.

library(choroplethr)  
no2\_map <- state\_choropleth(no2\_monitors, legend = "NO2 Monitors")  
no2\_map



Just for fun, let's see that with pipes.

no2\_map\_piped <- no2\_max %>%  
 group\_by(State.Name) %>%  
 summarize(value = n()) %>%  
 filter(State.Name != "Puerto Rico") %>%  
 mutate(State.Name = tolower(State.Name)) %>%  
 rename(region = State.Name) %>%  
 state\_choropleth(legend = "NO2 Monitors")  
  
no2\_map\_piped



[Back to exercises](#ex)

## Download

Below is the code for retrieving the no2\_2014 data frame from the EPA website.

# create a temporary file  
temp <- tempfile()  
  
# download the .zip file to a temporary file--this might take a minute or two  
download.file('http://aqsdr1.epa.gov/aqsweb/aqstmp/airdata/daily\_42602\_2014.zip', temp)  
  
# read the .csv file into R  
no2 <- read.csv(unz(temp, 'daily\_42602\_2014.csv'), stringsAsFactors = FALSE)  
  
# delete the temporary file  
unlink(temp)