GEOG 491/891: Special Topics - Spatial Analysis in R

Week 12.01: Interactive Mapping in R

Dr. Bitterman

Today's schedule

- Open discussion
- Intro to Leaflet (https://rstudio.github.io/leaflet/)

Anything to discuss? Questions?

Remaining topics

- Week 12: Interactive mapping (Intro lab 5)
- Week 13: Applications
- Week 14: Thanksgiving week
- Week 15: Applications
- Week 16: Project presentations

What does it mean (to you) if a map is "interactive"?

Today's setup

```
library(tidyverse)
library(leaflet)
```

Leaflet is…

- A Javascript library with an API we can access in R

Features

- Interactive panning/zooming
- Compose maps using arbitrary combinations of:
    - Map tiles
    - Markers
    - Polygons
    - Lines
    - Popups
    - GeoJSON

Features continued

- Create maps right from the R console or RStudio
- Embed maps in knitr/R Markdown documents and Shiny apps
- Easily render spatial objects from the sp or sf packages, or data frames with latitude/longitude columns
- Use map bounds and mouse events to drive Shiny logic
- Display maps in non spherical mercator projections
- Augment map features using chosen plugins from leaflet plugins repository

Let's try it

```
m <- leaflet()

m
```

What do you get?

Let's try it

```r
m <- leaflet() %>%
  addTiles()

m
```

What do you get?

Let's try it

```
m <- leaflet() %>%
  addTiles() %>%  # Add default OpenStreetMap map tiles
  addMarkers(lng = -96.703090, lat = 40.819288, popup="The Burnett Hall GIS Lab")

m
```

What do you get?

Let's create some data to plot

```r
# start with a data frame
df <- data.frame(
  lat = rnorm(100),
  lng = rnorm(100),
  size = runif(100, 5, 20),
  color = sample(colors(), 100)
)

# then add the data frame to a leaflet map
m2 <- leaflet(df) %>% addTiles()
```

What do you get?

How can we interrogate the properties/attributes of an object?

The `$` operator

```
m2$x
```

What do the data look like?

Let's try to visualize it

Break down the code first

```
# first one
m2 %>% addCircleMarkers(radius = ~size, color = ~color, fill = FALSE)


# second one
m2 %>% addCircleMarkers(radius = runif(100, 4, 10), color = c('red'))
```

What happened?

Let's check out some other tiles

```
m <- leaflet() %>% setView(lng = -96.703090, lat = 40.81928, zoom = 14)
m %>% addTiles()

# third party tiles using addProvider() function

m %>% addProviderTiles(providers$Stamen.Toner)
m %>% addProviderTiles(providers$CartoDB.Positron)
m %>% addProviderTiles(providers$CartoDB.DarkMatter)
m %>% addProviderTiles(providers$Esri.NatGeoWorldMap)
```

Give it a shot. What do you like?

Let's use some of our data

```r
parks <- sf::read_sf("./data/State_Park_Locations.shp")

# set up the map, zoom out a bit
mp <- leaflet(data = parks) %>% setView(lng = -96.703090, lat = 40.81928, zoom = 10)
mp %>% addTiles() %>%
  addMarkers(popup = ~AreaName, label = ~AreaName)
```

What's the diff between popup and label?

Lines

```r
streams <- sf::read_sf("./data/Streams_303_d_.shp")
ms <- leaflet(data = streams) %>%
  setView(lng = -96.703090, lat = 40.81928, zoom = 10) %>%
  addTiles() %>%
  addPolylines(., color = "blue",
               popup = ~paste0(Waterbody_, " - ", Impairment))
```

What happened?

Multiple layers

Note, there's a difference here

```
# do multiple layers by not passing the first "leaflet()" call a data argument
m.both <- leaflet() %>%
  setView(lng = -96.703090, lat = 40.81928, zoom = 10) %>%
  addProviderTiles(providers$Esri.NatGeoWorldMap) %>%
  addMarkers(data = parks, popup = ~AreaName, label = ~AreaName) %>%
  addPolylines(data = streams, color = "blue",
               popup = ~paste0(Waterbody_, " - ", Impairment))
```

In the time remaining

- Add the municipal boundaries in Lancaster County to this map
- Give them a "popup" corresponding to their name
- And give them a fill color (you'll have to look this one up)

For this week

- Readings posted on Canvas
- Practice, practice, practice
- Work on your projects