

# Leaflet

The Johns Hopkins Data Science Lab

August 26, 2016

# Introduction

Leaflet is one of the most popular Javascript libraries for creating interactive maps. The leaflet R package allows you to create your own leaflet maps without needing to know any Javascript!

## Installation

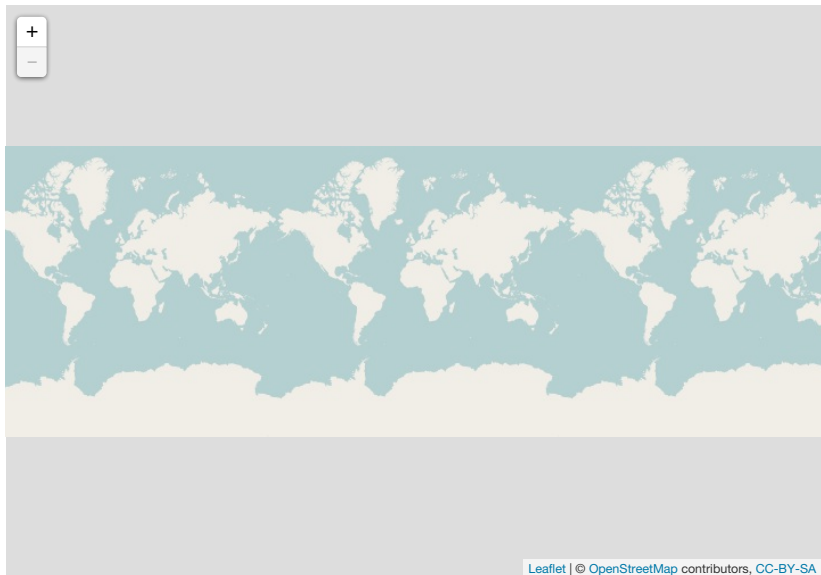
```
install.packages("leaflet")
```

# Your First Map

Getting started with leaflet is easy. The `leaflet()` function creates a map widget that you can store in a variable so that you can modify the map later on. You can add features to the map using the pipe operator (`%>%`) just like in dplyr. The `addTiles()` function adds mapping data from Open Street Map.

```
library(leaflet)
my_map <- leaflet() %>%
  addTiles()
my_map
```

# Your First Map



## Adding Markers

You can add markers to your map one at a time using the `addMarkers()` function by specifying the longitude and latitude. (Here's a tip if you tend to mix them up.) You can specify popup text for when you click on the marker with the `popup` argument.

```
library(leaflet)
my_map <- my_map %>%
  addMarkers(lat=39.2980803, lng=-76.5898801,
            popup="Jeff Leek's Office")
my_map
```

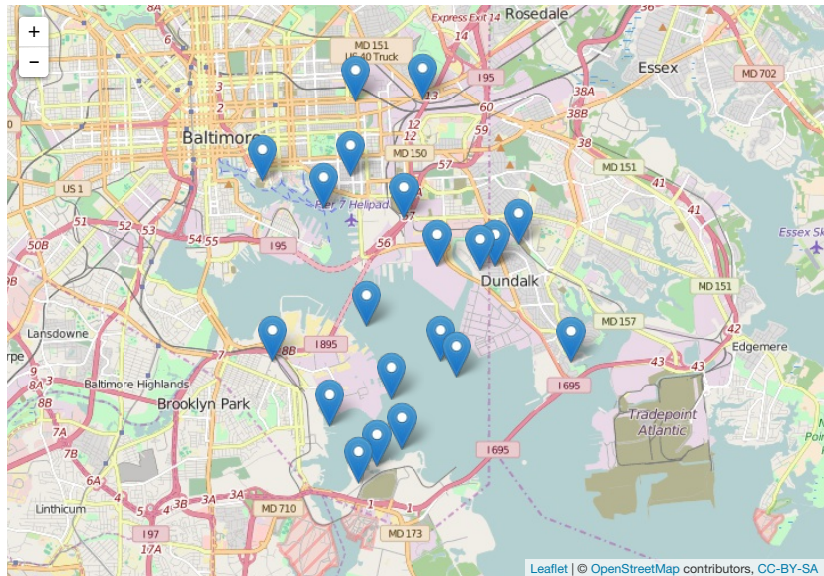


## Adding Many Markers

Adding one marker at a time is often not practical if you want to display many markers. If you have a data frame with columns `lat` and `lng` you can pipe that data frame into `leaflet()` to add all the points at once.

```
set.seed(2016-04-25)
df <- data.frame(lat = runif(20, min = 39.2, max = 39.3),
                 lng = runif(20, min = -76.6, max = -76.5))
df %>%
  leaflet() %>%
  addTiles() %>%
  addMarkers()
```

# Adding Many Markers





## Making Custom Markers

The blue markers that leaflet comes packaged with may not be enough depending on what you're mapping. Thankfully you can make your own markers from .png files.

```
hopkinsIcon <- makeIcon(  
  iconUrl = "http://brand.jhu.edu/content/uploads/2014/06/1  
  iconWidth = 31*215/230, iconHeight = 31,  
  iconAnchorX = 31*215/230/2, iconAnchorY = 16  
)
```

```
hopkinsLatLng <- data.frame(  
  lat = c(39.2973166, 39.3288851, 39.2906617),  
  lng = c(-76.5929798, -76.6206598, -76.5469683))
```

```
hopkinsLatLng %>%  
  leaflet() %>%  
  addTiles() %>%  
  addMarkers(icon = hopkinsIcon)
```



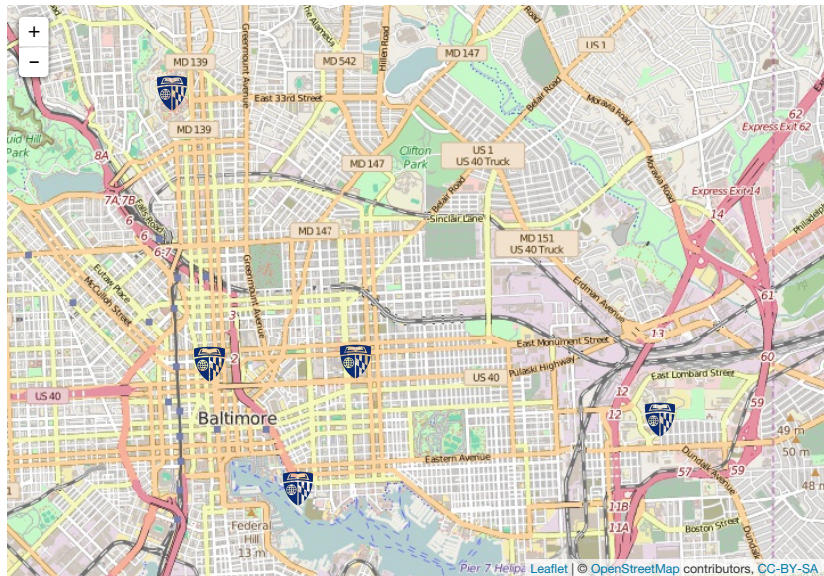
## Adding Multiple Popups

When adding multiple markers to a map, you may want to add popups for each marker. You can specify a string of plain text for each popup, or you can provide HTML which will be rendered inside of each popup.

```
hopkinsSites <- c(
  "<a href='http://www.jhsph.edu/'>East Baltimore Campus</a>"
  "<a href='https://apply.jhu.edu/visit/homewood/'>Homewood</a>"
  "<a href='http://www.hopkinsmedicine.org/johns_hopkins_ba</a>"
  "<a href='http://www.peabody.jhu.edu/'>Peabody Institutes</a>"
  "<a href='http://carey.jhu.edu/'>Carey Business School</a>"
)

hopkinsLatLng %>%
  leaflet() %>%
  addTiles() %>%
  addMarkers(icon = hopkinsIcon, popup = hopkinsSites)
```

# Adding Multiple Popups

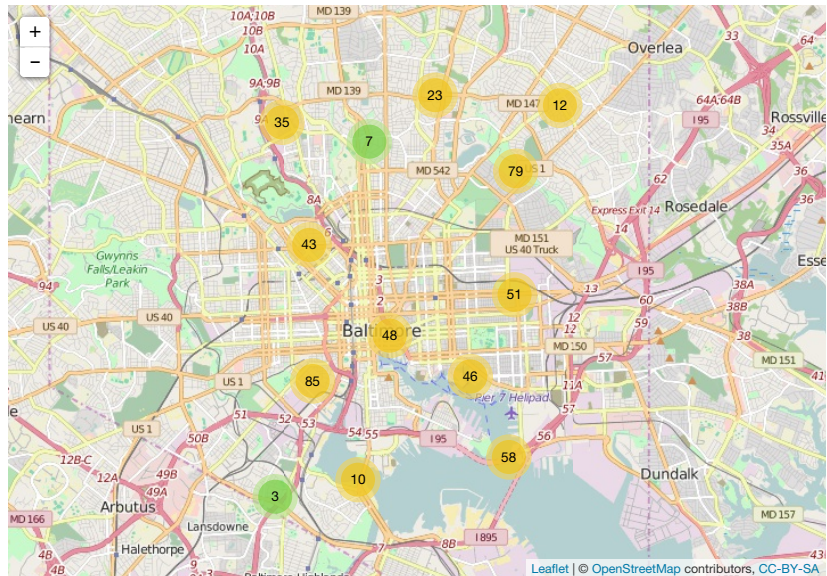


## Mapping Clusters

Sometimes you might have so many points on a map that it doesn't make sense to plot every marker. In these situations leaflet allows you to plot clusters of markers using `addMarkers(clusterOptions = markerClusterOptions())`. When you zoom in to each cluster, the clusters will separate until you can see the individual markers.

```
df <- data.frame(lat = runif(500, min = 39.25, max = 39.35),
                 lng = runif(500, min = -76.65, max = -76.5))
df %>%
  leaflet() %>%
  addTiles() %>%
  addMarkers(clusterOptions = markerClusterOptions())
```

# Mapping Clusters

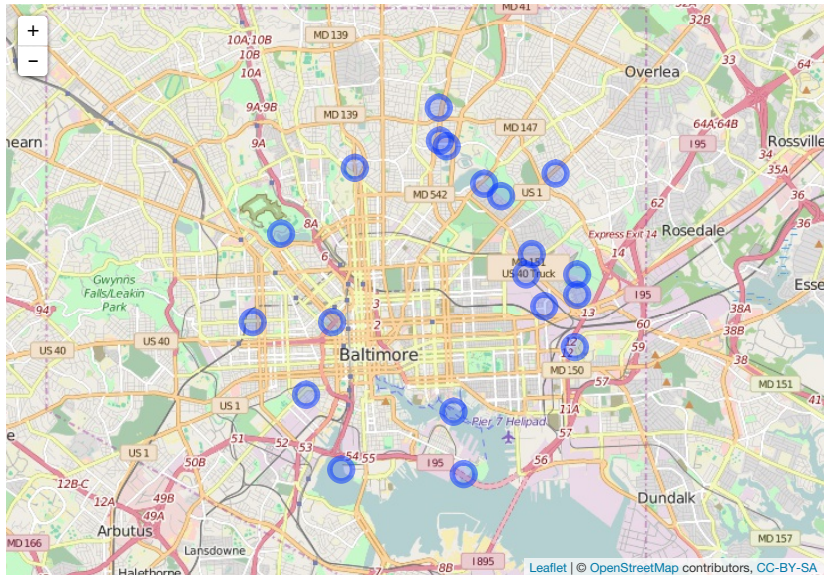


## Mapping Circle Markers

Instead of adding markers or clusters you can easily add circle markers using `addCircleMarkers()`.

```
df <- data.frame(lat = runif(20, min = 39.25, max = 39.35),  
                 lng = runif(20, min = -76.65, max = -76.55))  
df %>%  
  leaflet() %>%  
  addTiles() %>%  
  addCircleMarkers()
```

# Mapping Circle Markers





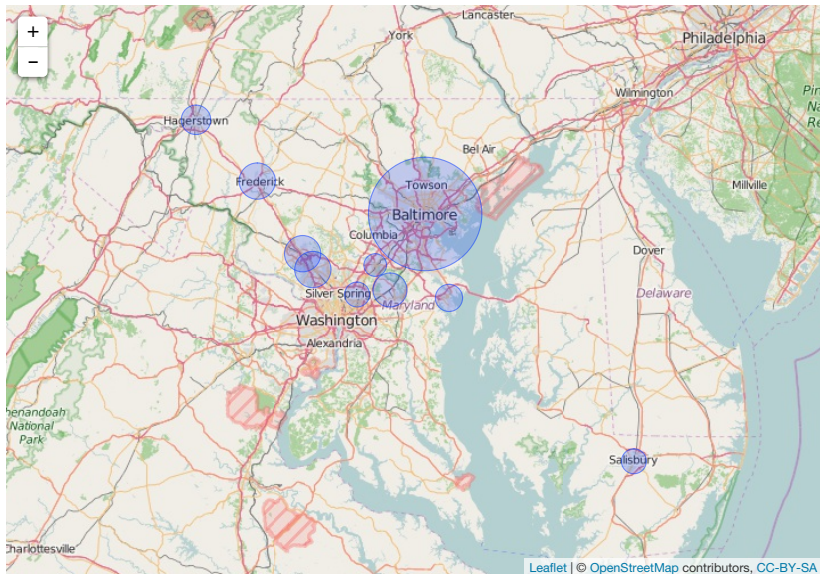
## Drawing Circles

You can draw arbitrary shapes on the maps you create, including circles and squares. The code below draws a map where the circle on each city is proportional to the population of that city.

```
md_cities <- data.frame(name = c("Baltimore", "Frederick",  
                                "Bowie", "Hagerstown", "Annapolis"),  
                        pop = c(619493, 66169, 62334, 61045,  
                                39890, 38880, 30587, 30484),  
                        lat = c(39.2920592, 39.4143921, 39.2857143,  
                                39.3321429, 39.2857143),  
                        lng = c(-76.6077852, -77.4204875, -76.8018521,  
                                -77.7142857, -76.8018521))  
  
md_cities %>%  
  leaflet() %>%  
  addTiles() %>%  
  addCircles(weight = 1, radius = sqrt(md_cities$pop) * 30)
```

# Drawing Circles

## Assuming 'lng' and 'lat' are longitude and latitude, res

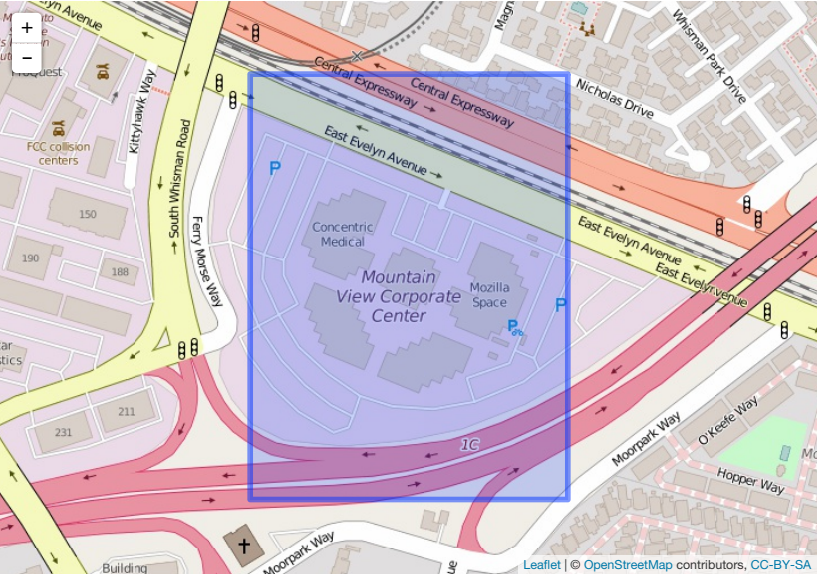


# Drawing Rectangles

You can add rectangles on leaflet maps as well:

```
leaflet() %>%  
  addTiles() %>%  
  addRectangles(lat1 = 37.3858, lng1 = -122.0595,  
                lat2 = 37.3890, lng2 = -122.0625)
```

# Drawing Rectangles



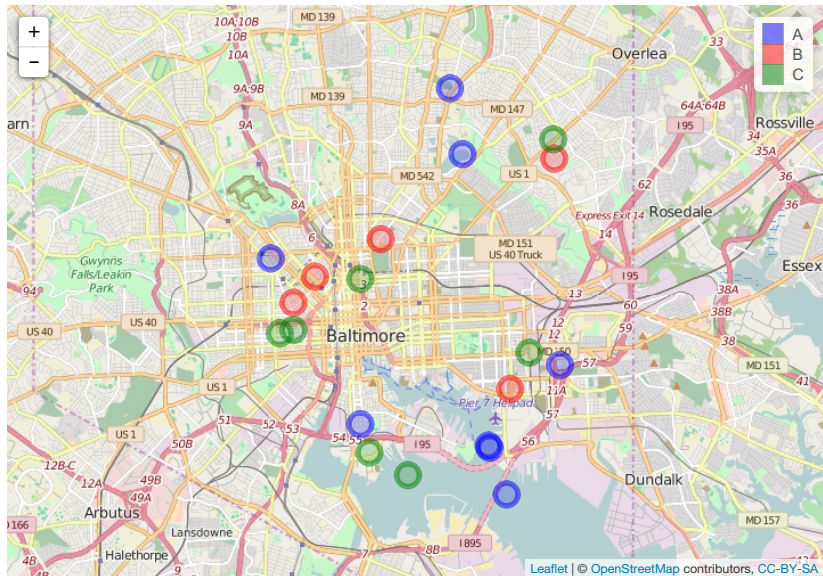
## Adding Legends

Adding a legend can be useful if you have markers on your map with different colors:

```
df <- data.frame(lat = runif(20, min = 39.25, max = 39.35),  
                 lng = runif(20, min = -76.65, max = -76.55),  
                 col = sample(c("red", "blue", "green"), 20,  
                             stringsAsFactors = FALSE)
```

```
df %>%  
  leaflet() %>%  
  addTiles() %>%  
  addCircleMarkers(color = df$col) %>%  
  addLegend(labels = LETTERS[1:3], colors = c("blue", "red", "green"))
```

# Adding Legends



# Conclusion

For more details about the leaflet package for R visit  
<http://rstudio.github.io/leaflet/>.