Leaflet Basics

Raymond Looney

4/24/2018

Leaflet:

Leaflet is a way to create interactive maps. leaflet() creates a map widget that can store variables in order to modify the map later on

Installation & Preliminaries

```
#install.packages("leaflet"); comment out in order to knit
library(leaflet)
library(ggplot2)
library(maps)
library(TeachingDemos)
char2seed("Professor Looney")
dF <- read.csv("leafletData30.csv")
dF2<- read.csv("leafletData500.csv")
cities<- read.csv("cities.csv")
```

How to create a simple map of the earth & piping

```
# addTiles() adds mapping data from Open Street Map
# %>% takes the output my_map,and adds or "pipes" to the next command addTiles() as the first argument, and rea
ssigns it to that variable
my_map <- leaflet() %>%
  addTiles()
my_map
```

/by-sa/2.0/)

+

−

```
#Without piping notation.
my_map=leaflet()
my_map= addTiles(my_map)
my_map
```

+

−

Different types of maps, and how to incorportate them into your data.

```
#Link:
#http://leaflet-extras.github.io/leaflet-providers/preview/index.html

#Satellite
#my_map %>% addProviderTiles(providers$Esri.WorldImagery)

#Topography Map
#my_map %>% addProviderTiles(providers$OpenTopoMap)

#National Geographic World Map
#my_map %>% addProviderTiles(providers$Esri.NatGeoWorldMap)

### Adding map to my_map
my_map <- leaflet() %>%
  addTiles() %>%
  addProviderTiles(providers$Esri.WorldImagery)
my_map
```

Adding Markers and Pop-up Text

This can be used to label certain areas that samples were collected for an in-field experiment.

```
#Guess who!
map <- my_map %>%
  addMarkers(lat=44.4764, lng=-73.1955)
map
```

```
#Give her a label
map <- my_map %>%
  addMarkers(lat=44.4764, lng=-73.1955,
             popup="Bio381 Classroom") # adds a pip up
map
```

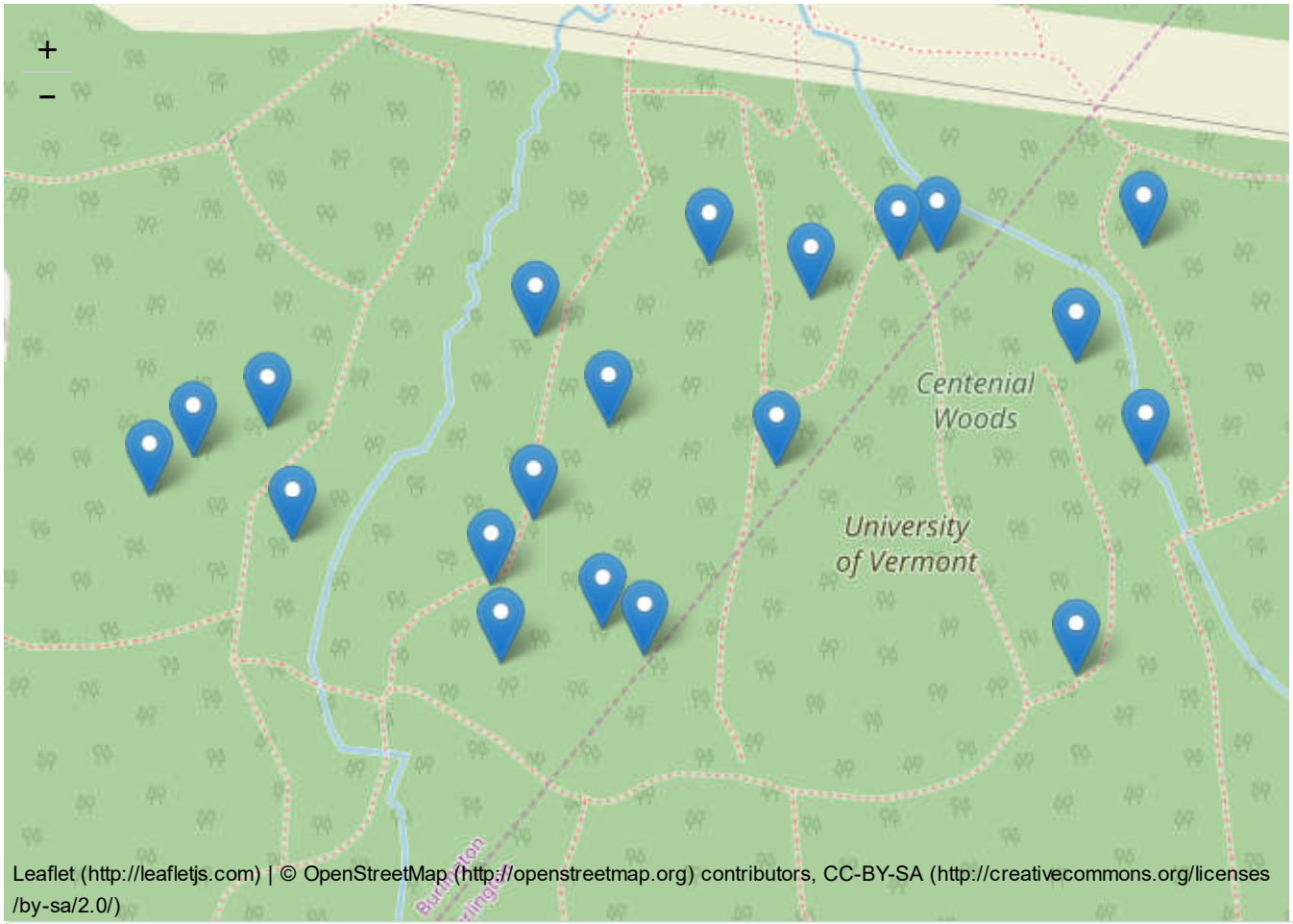
Adding Many Markers

Adding 1 marker at a time isn't practical all the time. Create a random data file, or insert a .csv file with 2 variables: latitude (lat) and longitude (lng)

```
# Let's say I wanted to collect  data randomly in Centennial Woods.
                            # Create a random data frame:
df <- data.frame(lat= runif(20, min=44.4770, max=44.4793),
                 lng= runif(20, min=-73.18788, max=-73.18203))
head(df)
```

```
##        lat       lng
## 1 44.47813 -73.18413
## 2 44.47766 -73.18574
## 3 44.47900 -73.18324
## 4 44.47865 -73.18548
## 5 44.47792 -73.18549
## 6 44.47802 -73.18766
```

```
df %>%
  leaflet() %>% #passes argument to make map
  addTiles() %>% # Adds mapping data from Open Street Map
  addMarkers() #Adds markers
```




```
                          #Importing a .csv file
# dF is a data frame with 30 coordinates in it
markers <- data.frame(lat= dF$lat,
                      lng= dF$lng)
head(markers)
```

```
##        lat       lng
## 1 44.47709 -73.18203
## 2 44.47721 -73.18507
## 3 44.47912 -73.18382
## 4 44.47773 -73.18309
## 5 44.47762 -73.18596
## 6 44.47891 -73.18708
```

```
markers %>%
  leaflet() %>% #passes argument to make map
  addTiles() %>% # Adds mapping data from Open Street Map
  addMarkers() #Adds markers
```

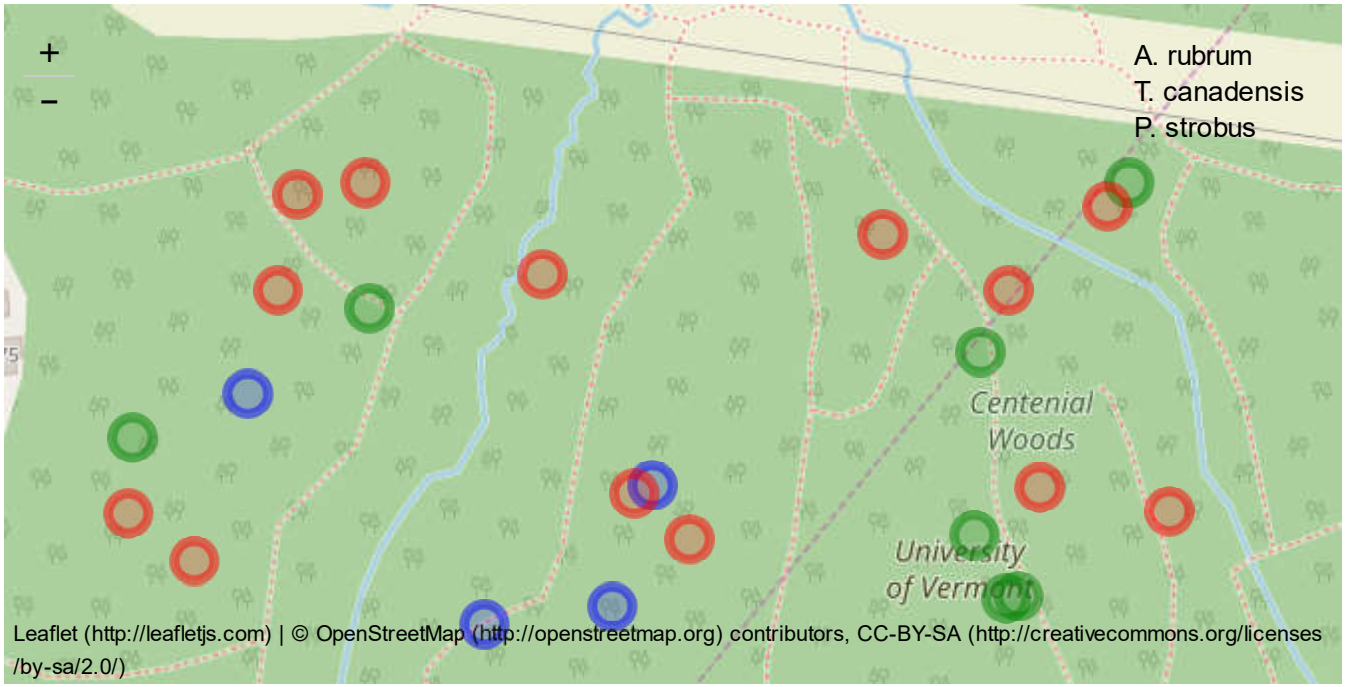
Adding Legends:

A. rubrum= Red Maple

T. canadensis= Eastern Hemlock

P. strobus= White Pine

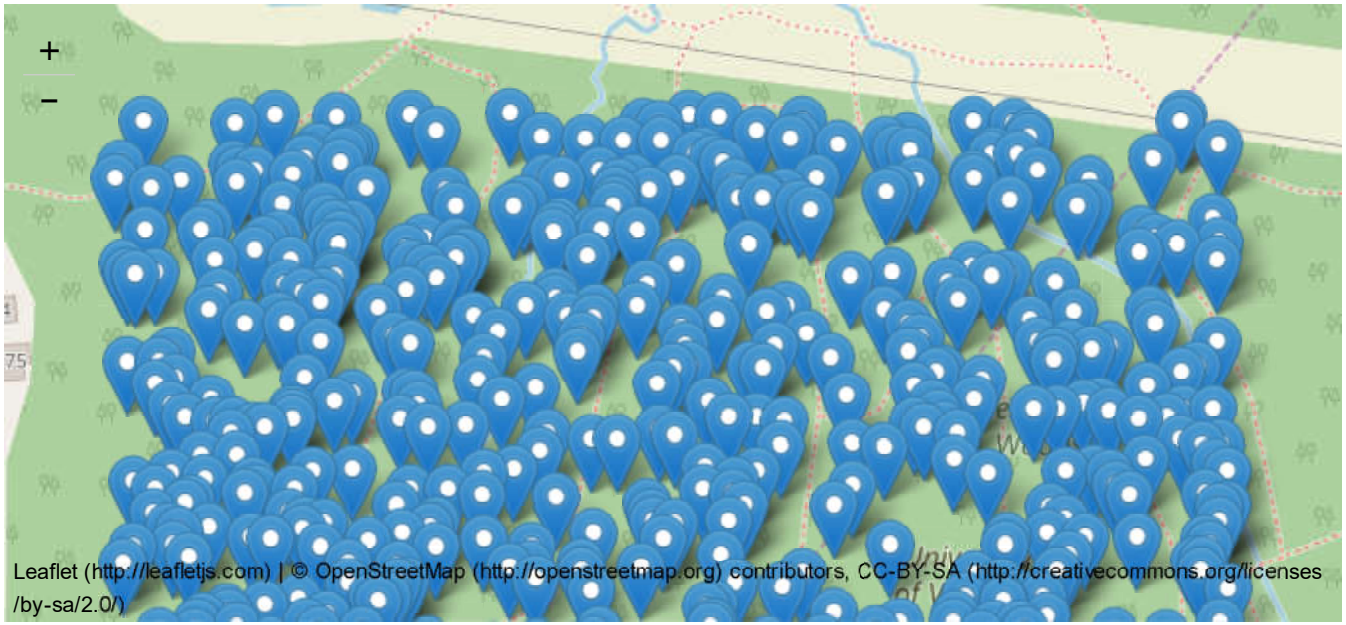
```
# add what colors you want. If this doesnt appear you get shadows on the map.
df <- data.frame( col = sample(c("red", "blue", "green"),
                               20, replace = TRUE),
                  stringsAsFactors = FALSE)

markers %>%
  leaflet() %>%
  addTiles() %>%
  addCircleMarkers(color = df$col) %>%
  addLegend(labels = c("A. rubrum", "T. canadensis", "P. strobus"), colors = c("blue", "red", "green"))
```



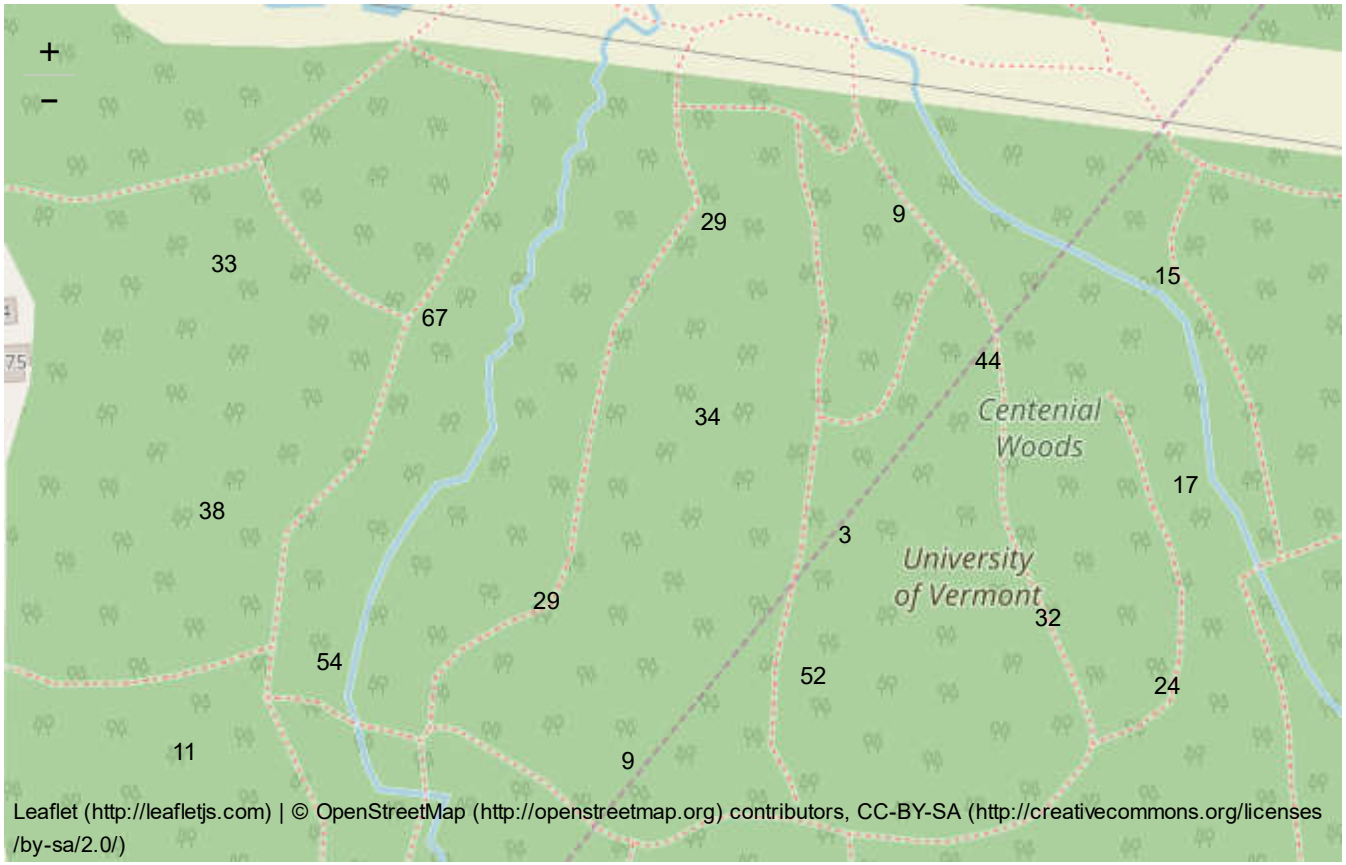
```
#assign color to tree species
```

Making Clusters

```
#dF2 is a data frame with 500 coordinates
# say you want to work with A LOT of data.. just adding markers to the map isnt going to be very useful.
cluster <- data.frame(lat= dF2$lat,
                      lng= dF2$lng)
cluster%>%
  leaflet()%>%
  addTiles()%>%
  addMarkers()
```

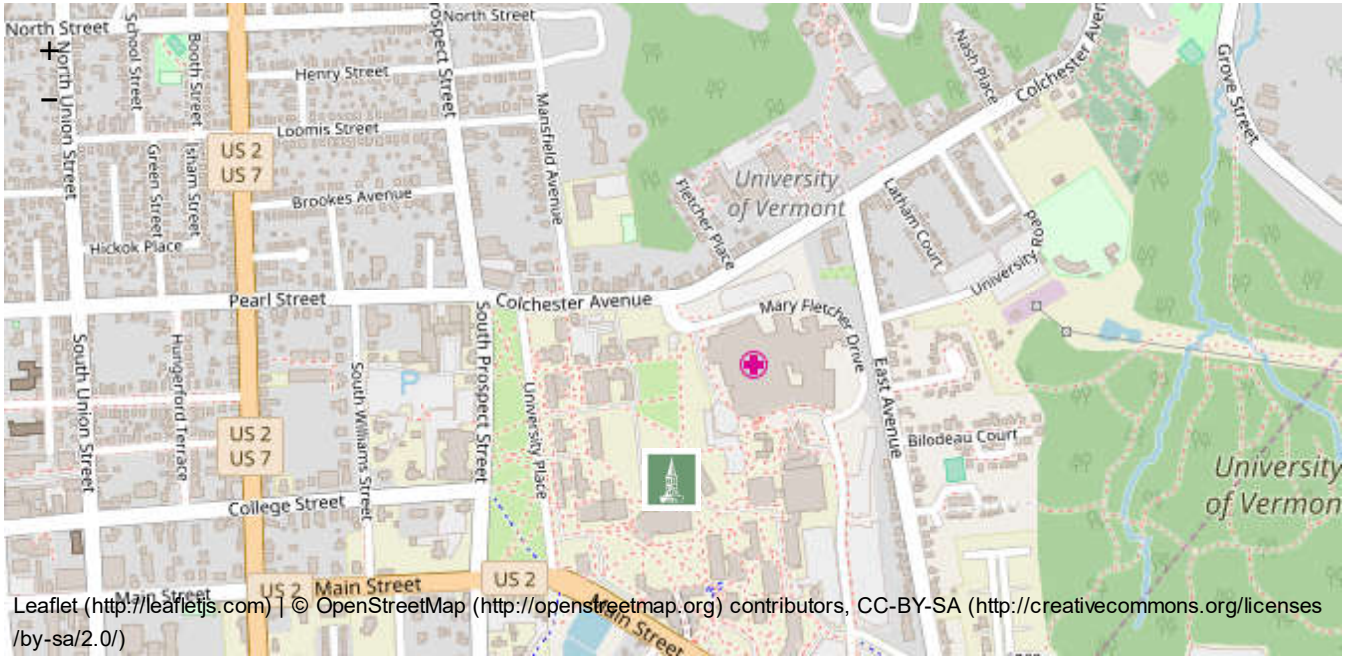


```
# boom problem solved with clusters!
cluster%>%
  leaflet()%>%
  addTiles()%>%
  addMarkers(clusterOptions=markerClusterOptions())
```



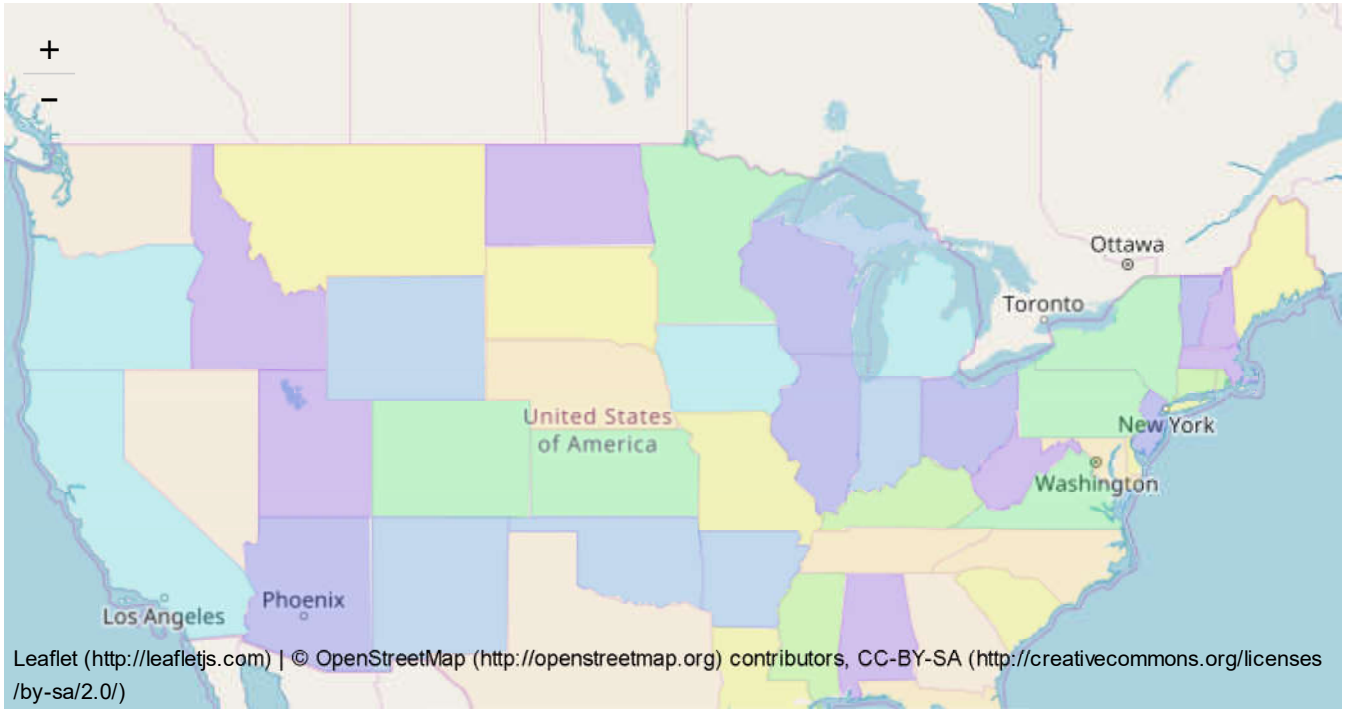
Making Custom Markers

```
# copy and paste!!!
uvmIcon <- makeIcon(iconUrl = "UVM.jpg", # call the image
                    iconWidth = 31*215/230,
                    iconHeight= 31,
  iconAnchorX= 31*215/230/2,
  iconAnchorY= 16
) # what i found to be the best length,height,width for marker
UVMLatLong <- data.frame(
  lat= c(44.4779),
  lng= c(-73.1965)) #lat & lng for your data point
UVMLatLong %>%
  leaflet()%>%
  addTiles()%>%
  addMarkers(icon= uvmIcon) # what icon do u want
```



Adding Shapes

```
# you can add rectangles or polygons with addRectangles() or addPolygons()

#POLYGONS
# simple colored map of the united states
mapStates = map("state", fill = TRUE, plot = FALSE)
leaflet(data = mapStates) %>% addTiles() %>%
  addPolygons(fillColor = topo.colors(10, alpha = NULL), stroke = FALSE)
```



```
#CIRCLES
print(cities) # data set with cities, lng, lat, and population.
```

```
##           city     lat     long     pop
## 1       Boston 42.3601 -71.0589  645966
## 2     Hartford 41.7627 -72.6743  125017
## 3          NYC 40.7127 -74.0059 8406000
## 4 Philadelphia 39.9500 -75.1667 1553000
## 5   Pittsburgh 40.4397 -79.9764  305841
## 6   Providence 41.8236 -71.4222  177994
## 7   Burlington 44.4759 -73.2121   42260
```

```
leaflet(cities) %>% addTiles() %>%
  addCircles(lng = ~long, lat = ~lat, weight = 1,
             radius = ~sqrt(pop) * 30, popup = ~city)
```

```
# radius of circle is sqrt of population size, making it span all areas around the midpoint.
```