# Interactive Data Maps

## Tidy Data

```
arrest_tab <- read_csv("http://www.hcbravo.org/IntroDataSci/misc/BPD_Arrests.csv")
```

```
## Parsed with column specification:
## cols(
##   arrest = col_integer(),
##   age = col_integer(),
##   race = col_character(),
##   sex = col_character(),
##   arrestDate = col_character(),
##   arrestTime = col_time(format = ""),
##   arrestLocation = col_character(),
##   incidentOffense = col_character(),
##   incidentLocation = col_character(),
##   charge = col_character(),
##   chargeDescription = col_character(),
##   district = col_character(),
##   post = col_integer(),
##   neighborhood = col_character(),
##   `Location 1` = col_character()
## )
```

```
dat <- arrest_tab %>%
  filter(!is.na(`Location 1`)) %>%
  separate(`Location 1`, c("lat","lng"), sep=",") %>%
  mutate(lat=as.numeric(str_replace(lat, "\\(", ""))) %>%
  mutate(lng=as.numeric(str_replace(lng, "\\)", ""))) %>%
  sample_n(2000)
dat
```

```
## # A tibble: 2,000 x 16
##      arrest   age race  sex   arrestDate arrestTime arrestLocation
##       <int> <int> <chr> <chr> <chr>      <time>     <chr>
##  1 11369146    40 B     M     11/15/2011 11:40      100 N Eutaw St
##  2 11151114    41 B     M     02/12/2011 18:00      3400 Clifton Ave
##  3       NA    32 B     M     02/10/2012 00:45      3200 Frederick Ave
##  4 11271310    26 B     M     07/15/2011 15:40      100 N Hilton St
##  5 12494041    21 B     M     06/10/2012 13:34      500 Dolphin St
##  6 12579027    19 B     M     10/27/2012 15:45      2800 Harlem Ave
##  7 12481355    19 B     F     05/18/2012 23:48      0 S Chester St
##  8 11277897    46 B     M     07/22/2011 17:45      800 N Appleton St
##  9 12439262    45 B     F     03/13/2012 01:56      2200 N Fulton Ave
## 10 11317783    38 B     M     09/09/2011 20:45      4200 Reisterstown Rd
## # ... with 1,990 more rows, and 9 more variables: incidentOffense <chr>,
## #   incidentLocation <chr>, charge <chr>, chargeDescription <chr>,
## #   district <chr>, post <int>, neighborhood <chr>, lat <dbl>, lng <dbl>
```

# Map

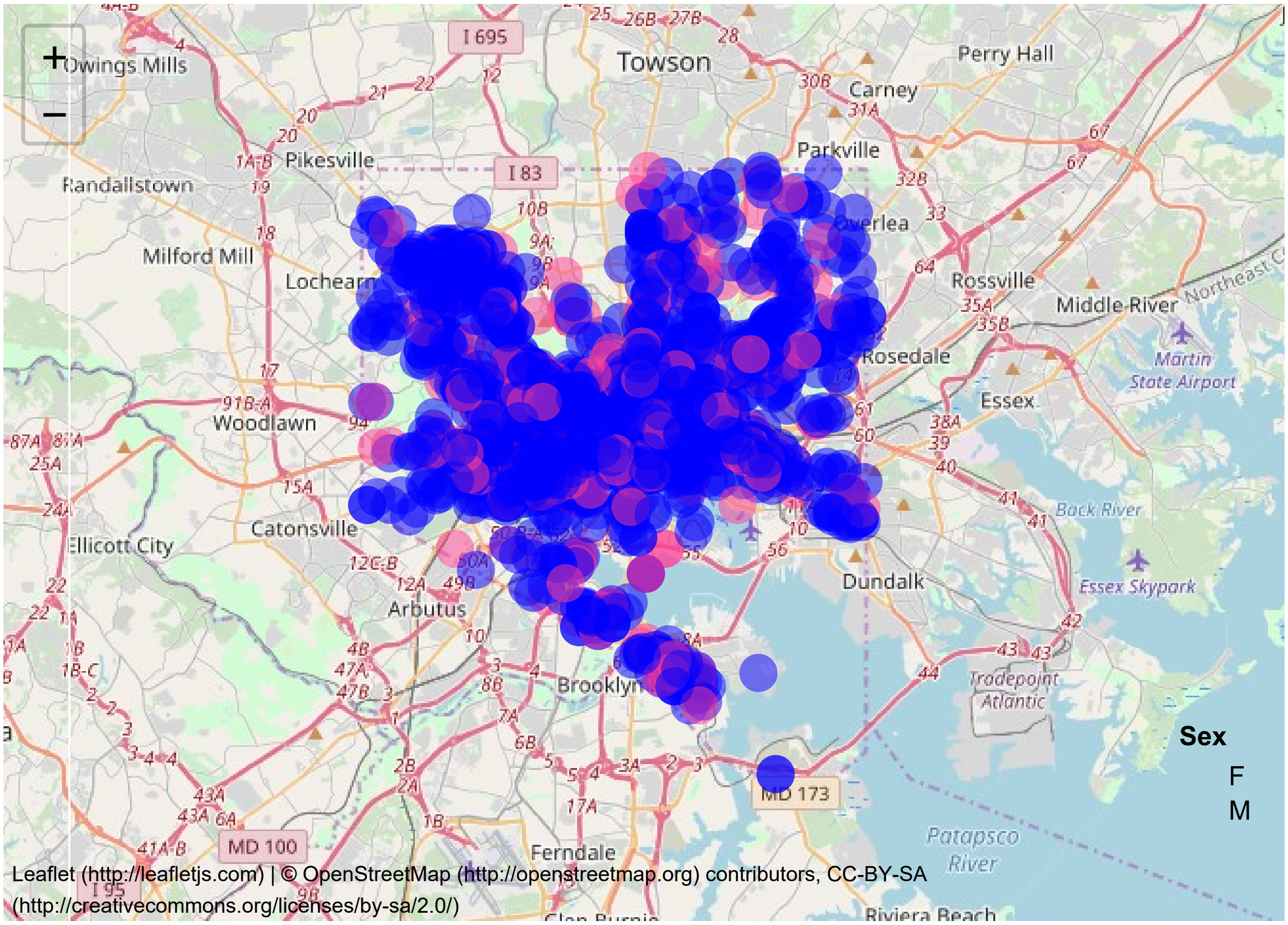
In my interactive map I chose to display three different metrics: Number of offenses, distributions of crimes based on sex, and incident offense. The number of offenses was displayed using the markerClusters function. This is a very cool way to represent data in my opinion because you can see how the sampled data( 2000 offenses) is distributed on the map it self. Also the markers for the clusters are colored so the viewer of the map will easily be able to pickout highly concentrated areas of crime. From this metric we are able to see that crime occurs more frequenctly as you get closer to Baltimore( or atleast more crime was recorded closer towards the city rather than outside of the city). The second metric is sex where the blue color is Male and the pink color is female. A very interesting (admittedly unplanned) aspect of the map is if there is a incident involving both a Male and a Female (prostitution, offense where offender is a group of both male and female, etc. ) the colored marked would actually show up as purple because the blue dot and pink dot are transparent enough to blend together. This is accomplished through the fillopacity argument to the addCircleMarkers function. Because of the coloring it was evident that the perpetrator is more likely to be a male than a female. Finally, I included popups that ascertained what the offense of a particular marker was. I made these popups be hover over because otherwise, they would clutter the rest of the data and the map would look too bulky with too many things going on.

```
pal <- colorFactor(c("violetred1", "Blue"), dat$sex)

balto_map <- leaflet(dat) %>%
  addTiles() %>%
  addCircleMarkers(
    color = ~pal(sex),
    stroke = FALSE, fillOpacity = 0.5
    ,label= ~incidentOffense
  ) %>%
  addMarkers(clusterOptions = markerClusterOptions()) %>%
  addLegend("bottomright", pal = pal, values = ~sex,
    title = "Sex",
    opacity = 1) %>%
  setView(lat=39.29, lng=-76.61, zoom=11)
```

```
## Assuming "lng" and "lat" are longitude and latitude, respectively
## Assuming "lng" and "lat" are longitude and latitude, respectively
```

```
balto_map
```



Leaflet (http://leafletjs.com) | © OpenStreetMap (http://openstreetmap.org) contributors, CC-BY-SA (http://creativecommons.org/licenses/by-sa/2.0/)