Project 2. Spatial Exploration of Tweets

Haoruo Fu

Outline

1. Data Preparation
2. Spatial object I/O and conversion
3. Spatial object's attribute manipulation
4. Mapping overall tweets distribution
5. Mapping area of interest

1.1 Data Preparation

```r
#1.1
#read the given xls file csv
pu2014 <- read.csv('Twitter Data File_pu2014.csv', header = TRUE)

#create 'sp'(SpatialPointsDataFrame)
coords_sp <- cbind(pu2014$longitude, pu2014$latitude) #assign the cord
latlon_sp <- CRS("+proj=longlat +datum=WGS84") # Using WGS84 datum
sp <- SpatialPointsDataFrame(coords_sp, pu2014, proj4string = latlon_sp)

#create 'sf' (simple features object)
sf <- st_as_sf(pu2014, coords = c('longitude', 'latitude'), crs = 4326)
```

```r
#creat the bounding box
# Find the minimum and maximum latitude and longitude coordinates of tweets
xmin <- min(pu2014$longitude)
ymin <- min(pu2014$latitude)
xmax <- max(pu2014$longitude)
ymax <- max(pu2014$latitude)

#made a bit extra space for the coordinates, 0.1 for latitude and longitude
extra_latlong <- 0.1

#expand the latitude and longitude for 'sp' object
#calculate new bounding box coordinates
new_xmin_sp <- xmin - extra_latlong
new_ymin_sp <- ymin - extra_latlong
new_xmax_sp <- xmax + extra_latlong
new_ymax_sp <- ymax + extra_latlong

#set the expanded bounding box for 'sp' object
new_bbox_sp <- matrix(c(new_xmin_sp, new_ymin_sp, new_xmax_sp, new_ymax_sp), nrow = 2, ncol = 2)
attr(sp, "bbox") <- new_bbox_sp

#expand the latitude and longitude for 'sf' object
#calculate new bounding box coordinates
new_xmin_sf <- xmin - extra_latlong
new_ymin_sf <- ymin - extra_latlong
new_xmax_sf <- xmax + extra_latlong
new_ymax_sf <- ymax + extra_latlong
```

```r
#set the expanded bounding box for 'sf' object
new_sf_bb <- c('xmin' = new_xmin_sf, 'ymin' = new_ymin_sf, 'xmax' = new_xmax_sf, 'ymax' = new_ymax_sf)
attr(new_sf_bb, "class") <- "bbox"
attr(sf, "bbox") <- new_sf_bb
```

1.2 Spatial object I/O and conversion

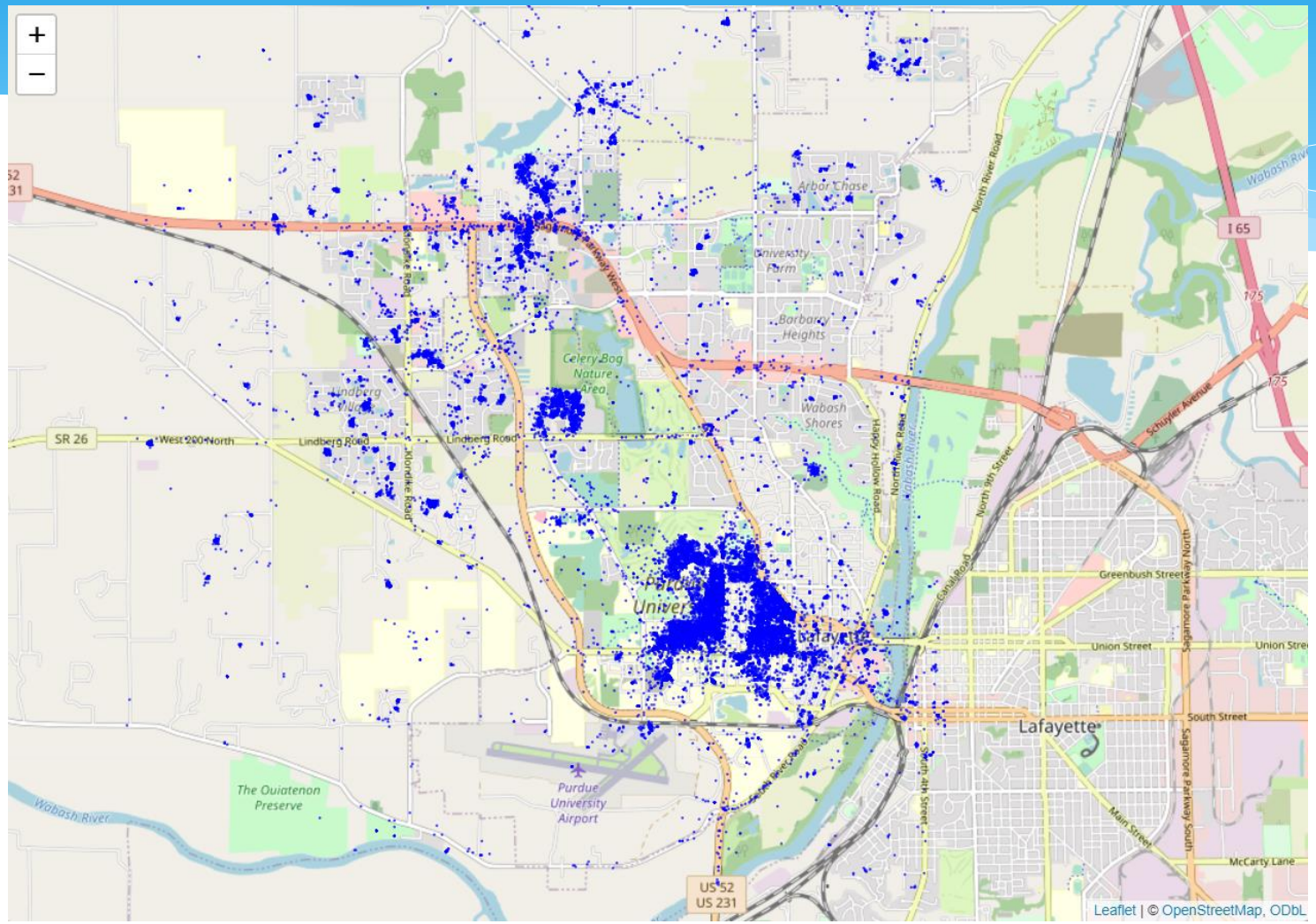
```
#read the shapefiles back from working folder
sf_read <- st_read("sf_shapefile.shp")
sp_read <- readOGR(dsn = ".", layer = "sp_shapefile")

#convert to geojson file
st_write(sf_read, "sf_geojson.geojson")  #sf file to geojson
writeOGR(obj = sp_read, dsn = ".", layer = "sp_geojson", driver = "GeoJSON") #sp files to geojson

#create 'sf' or'sp' object from read-in data if needed
sf_from_geojson <- st_read("sf_geojson.geojson")
sp_from_geojson <- readOGR(dsn = ".", layer = "sp_geojson")

#create KML file
st_write(sf_read, "sf_kml.kml", driver = "KML")
writeOGR(obj = sp_read, dsn = ".", layer = "sp_kml", driver = "KML")

#create object sf or sp same as above
sp_from_kml <- readOGR(dsn = ".", layer = "sp_kml")
sf_from_kml <- st_read("sf_kml.kml")

#create GeoPackage file
st_write(sf_read, "sf_geopackage.gpkg", driver = "GPKG")
writeOGR(obj = sp_read, dsn = ".", layer = "sp_geopackage", driver = "GPKG")

#create object sf sp same
sp_from_geopackage <- readOGR(dsn = ".", layer = "sp_geopackage")
sf_from_geopackage <- st_read("sf_geopackage.gpkg")
```

1.3 Spatial object's attribute manipulation

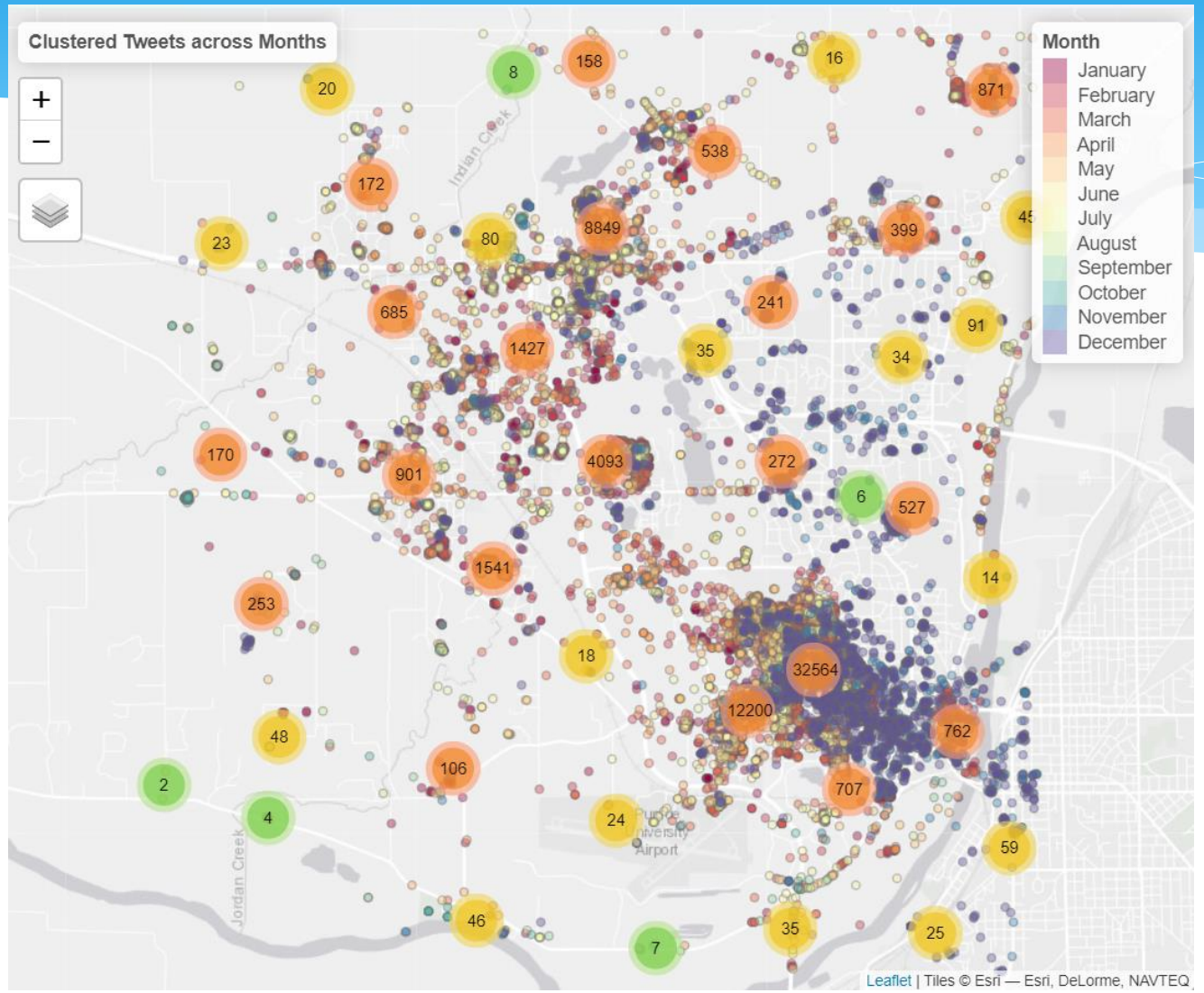
```
#convert 'epoch' column to datetime format
# 1.3 (continued)
sf$datetime <- as.POSIXct(sf$epoch, origin = "1970-01-01", tz = "EST")

#day, month, year
sf$date <- format(sf$datetime, format='%d-%m-%Y')

#day of the week
sf$dayofweek <- format(sf$datetime, "%w")

#subset
sf_subset <- sf[,c('user_id', 'geometry', 'epoch', 'datetime', 'date', 'dayofweek')]
```

```
Geodetic CRS:  WGS 84
    user_id                   geometry      epoch            datetime       date dayofweek
1 174220305 POINT (-86.94425 40.47112) 1388552464 2014-01-01 00:01:04 01-01-2014         3
2  99818152 POINT (-86.94266 40.44576) 1388552467 2014-01-01 00:01:07 01-01-2014         3
3  30137074 POINT (-86.93918 40.47966) 1388552533 2014-01-01 00:02:13 01-01-2014         3
4 458797537  POINT (-86.99292 40.4582) 1388552645 2014-01-01 00:04:05 01-01-2014         3
5  26849093  POINT (-86.9006 40.42621) 1388552648 2014-01-01 00:04:08 01-01-2014         3
6 284247037 POINT (-86.98755 40.43553) 1388552673 2014-01-01 00:04:33 01-01-2014         3
```

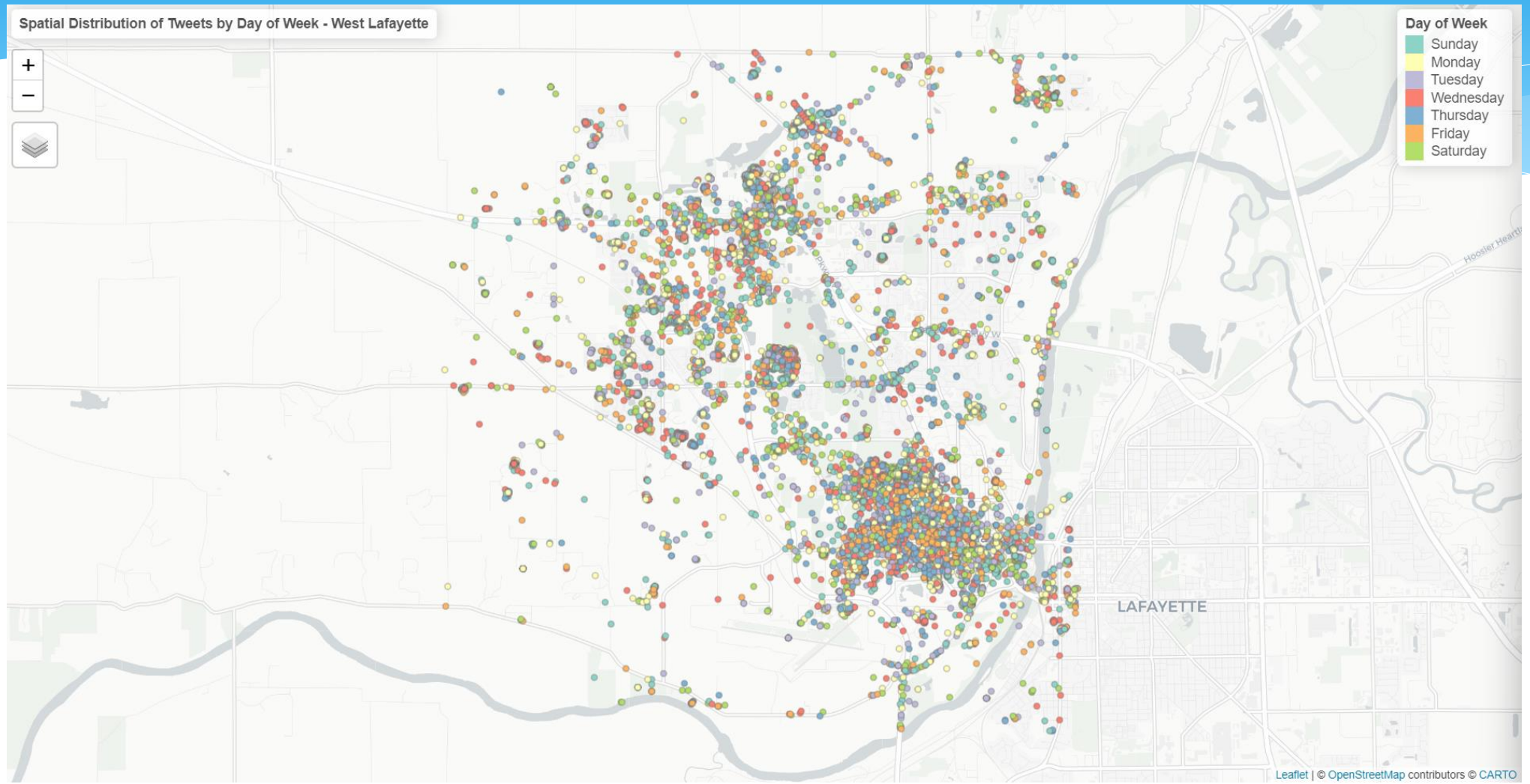
Mapping the overall tweets distribution



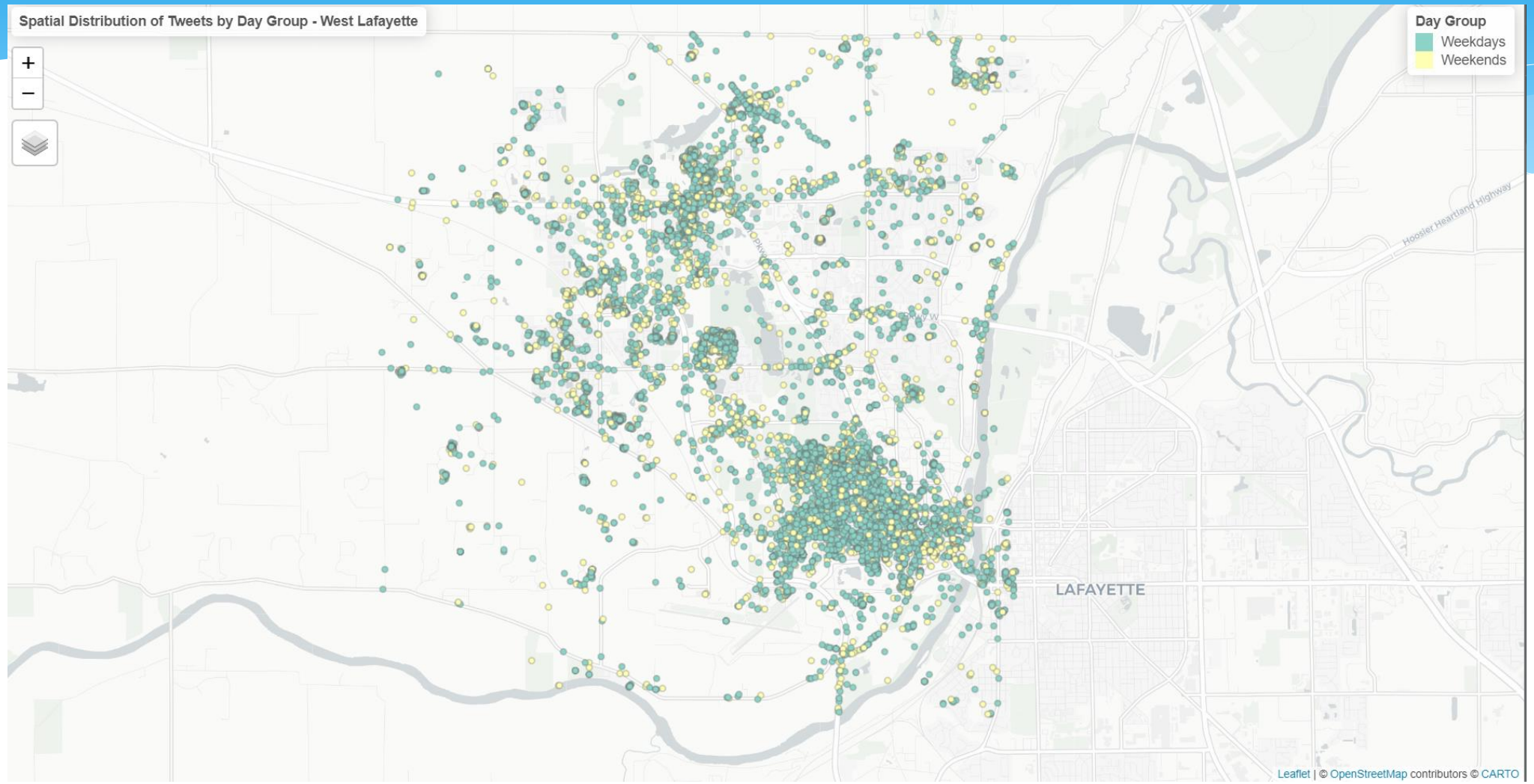
Mapping the overall tweets distribution Month



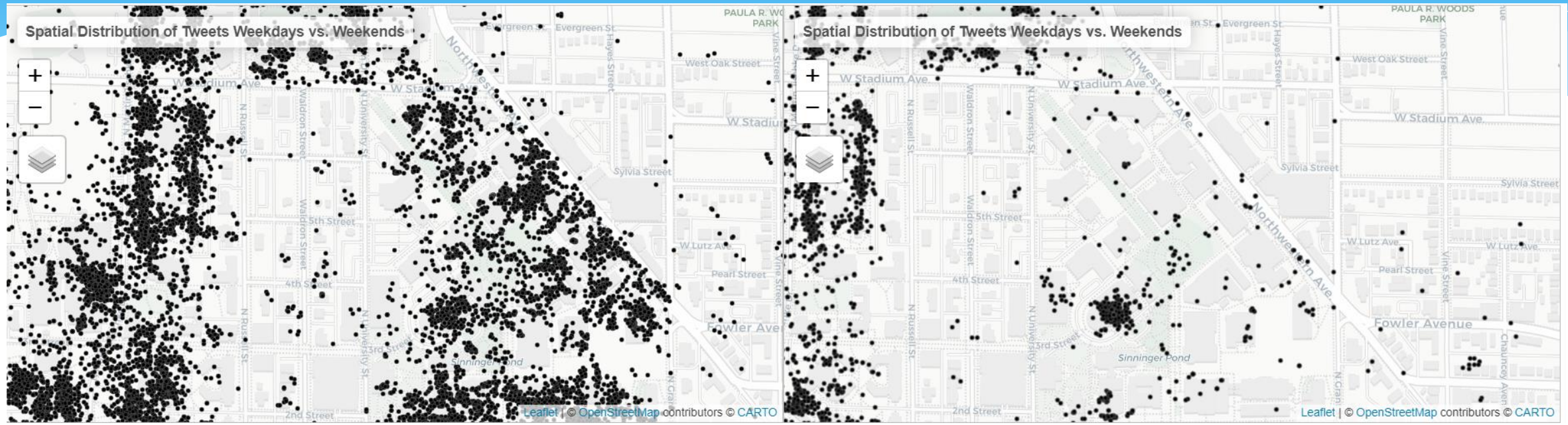
Mapping the overall tweets distribution Week of Day



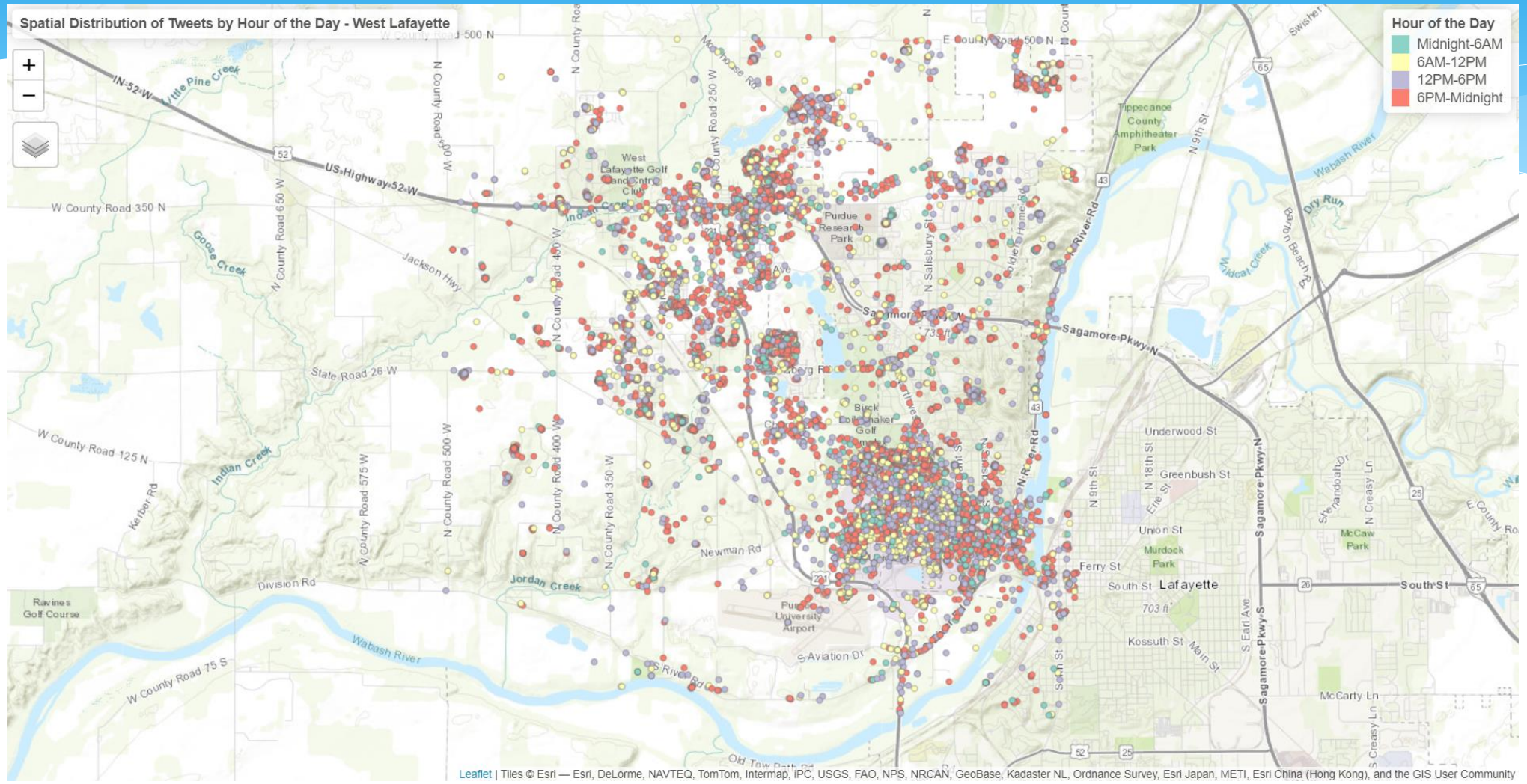
Mapping the overall tweets distribution Week of Day



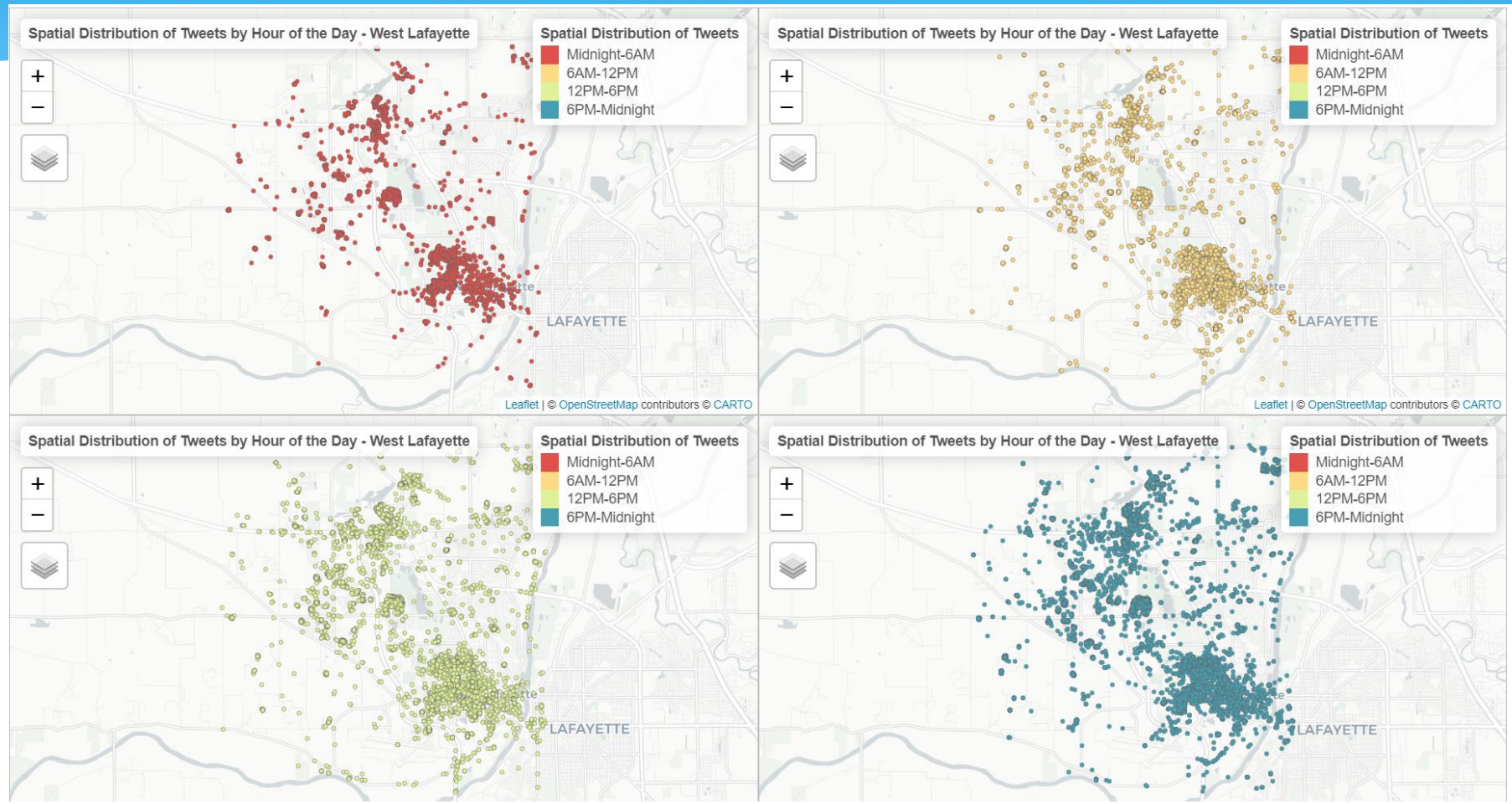
Mapping the overall tweets distribution Week of Day



Mapping Distribution of Hours



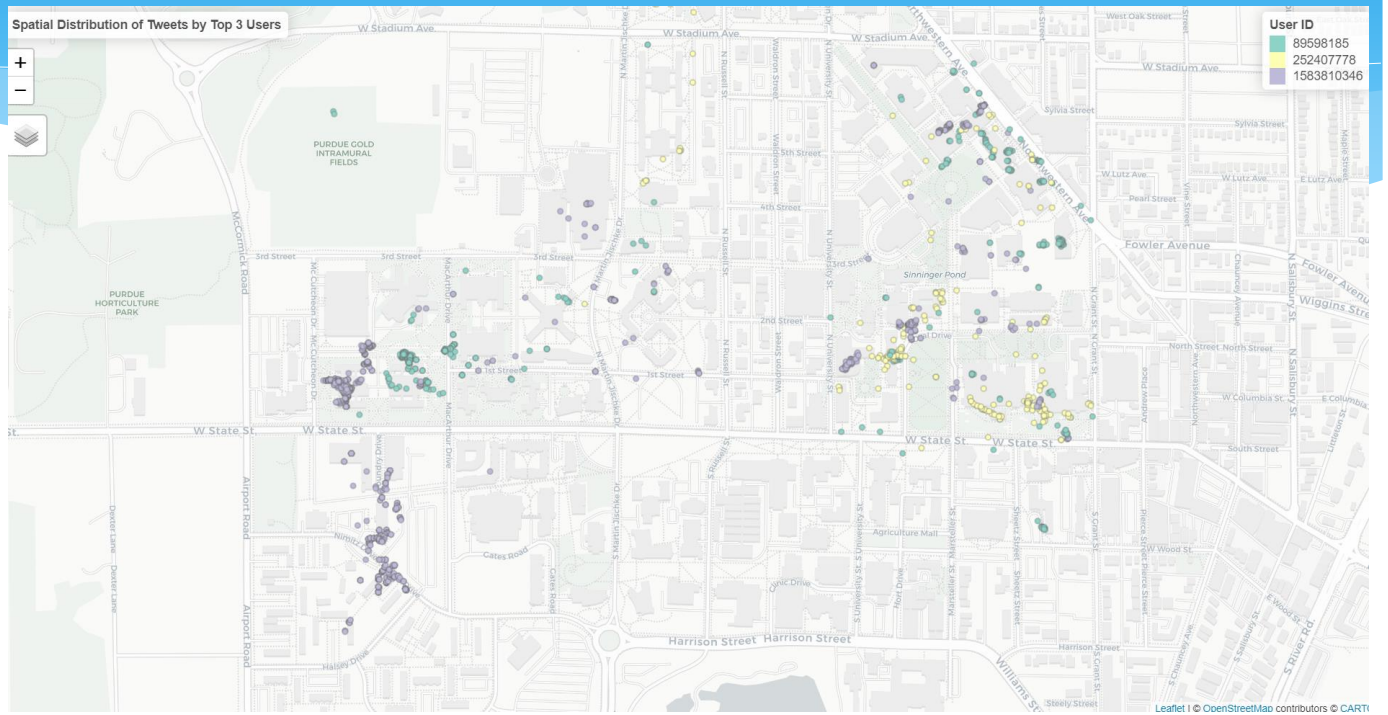
Mapping Distribution of Hours

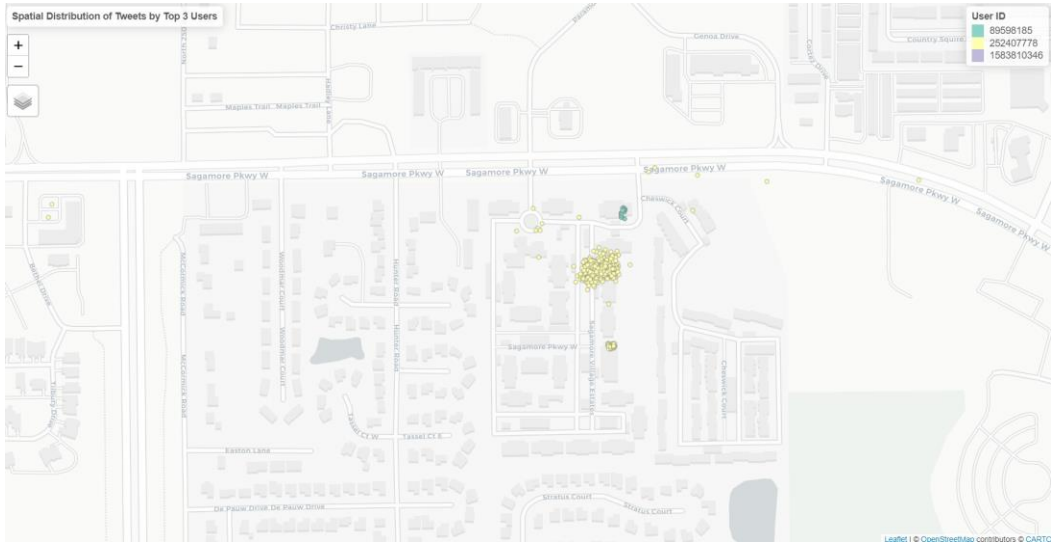


Top Three Users.

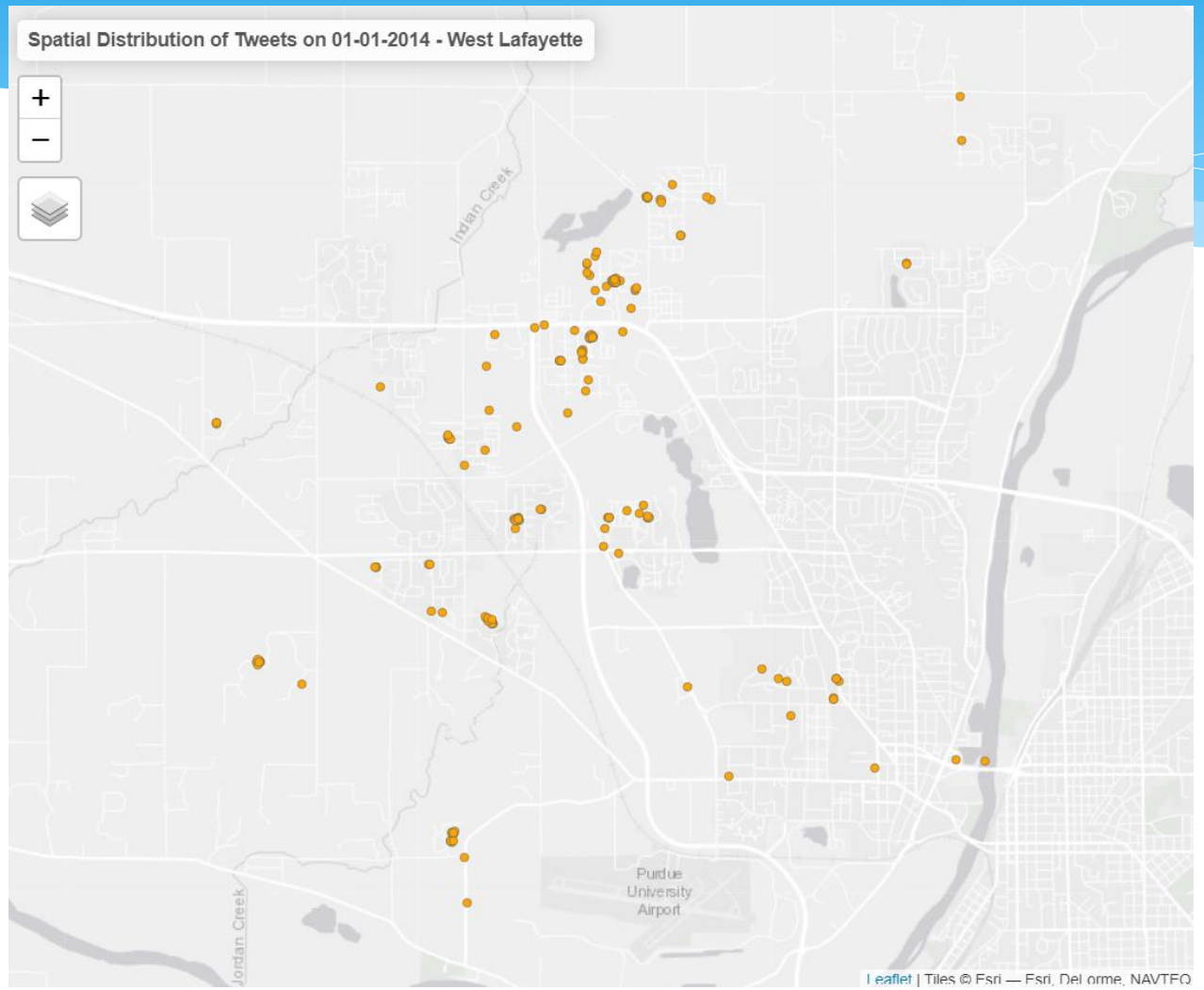
	user_id	tweet_count	
	<dbl>	<int>	
1	1583810346	2566	((-86
2	252407778	1633	((-86
3	89598185	1496	((-86

>

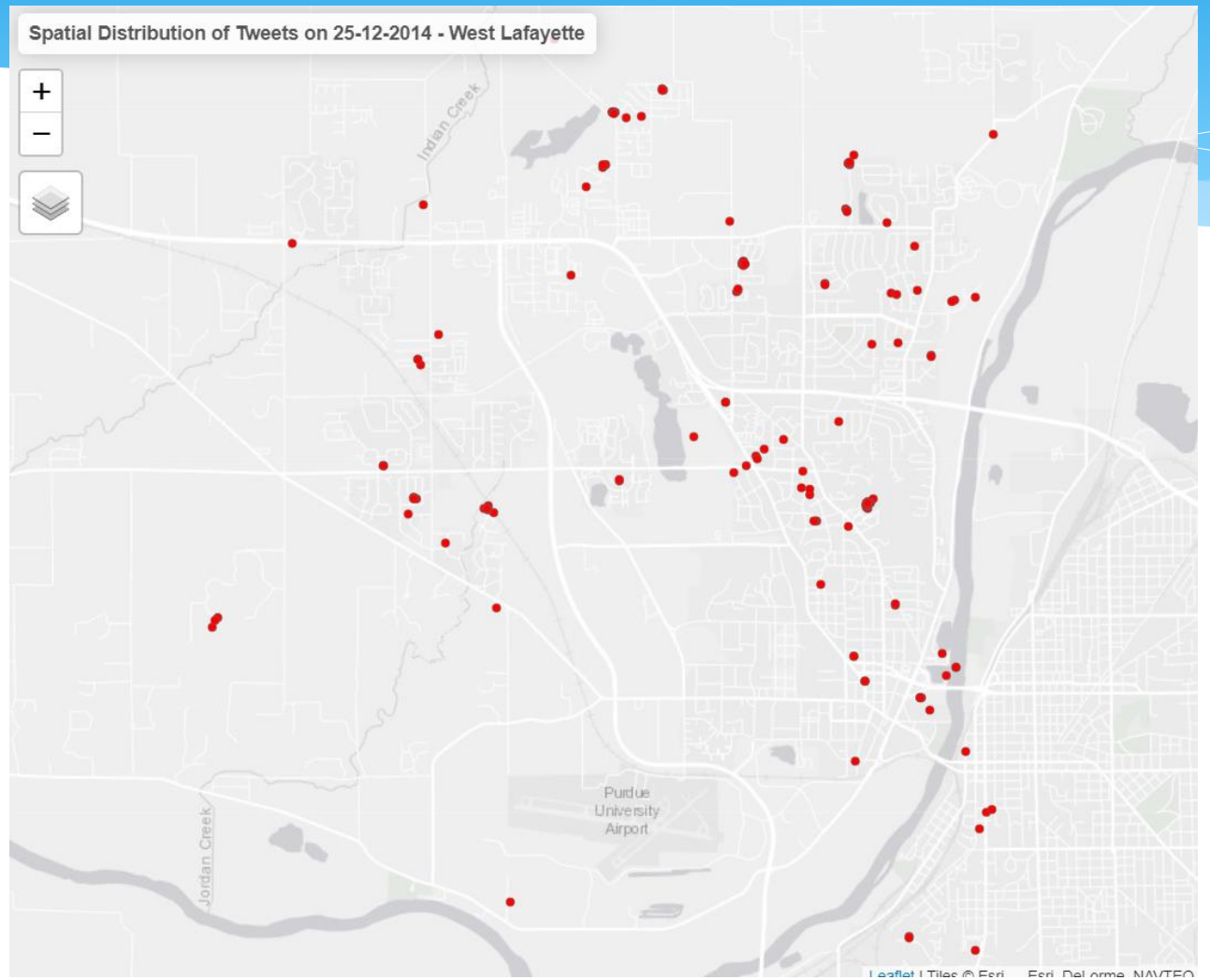




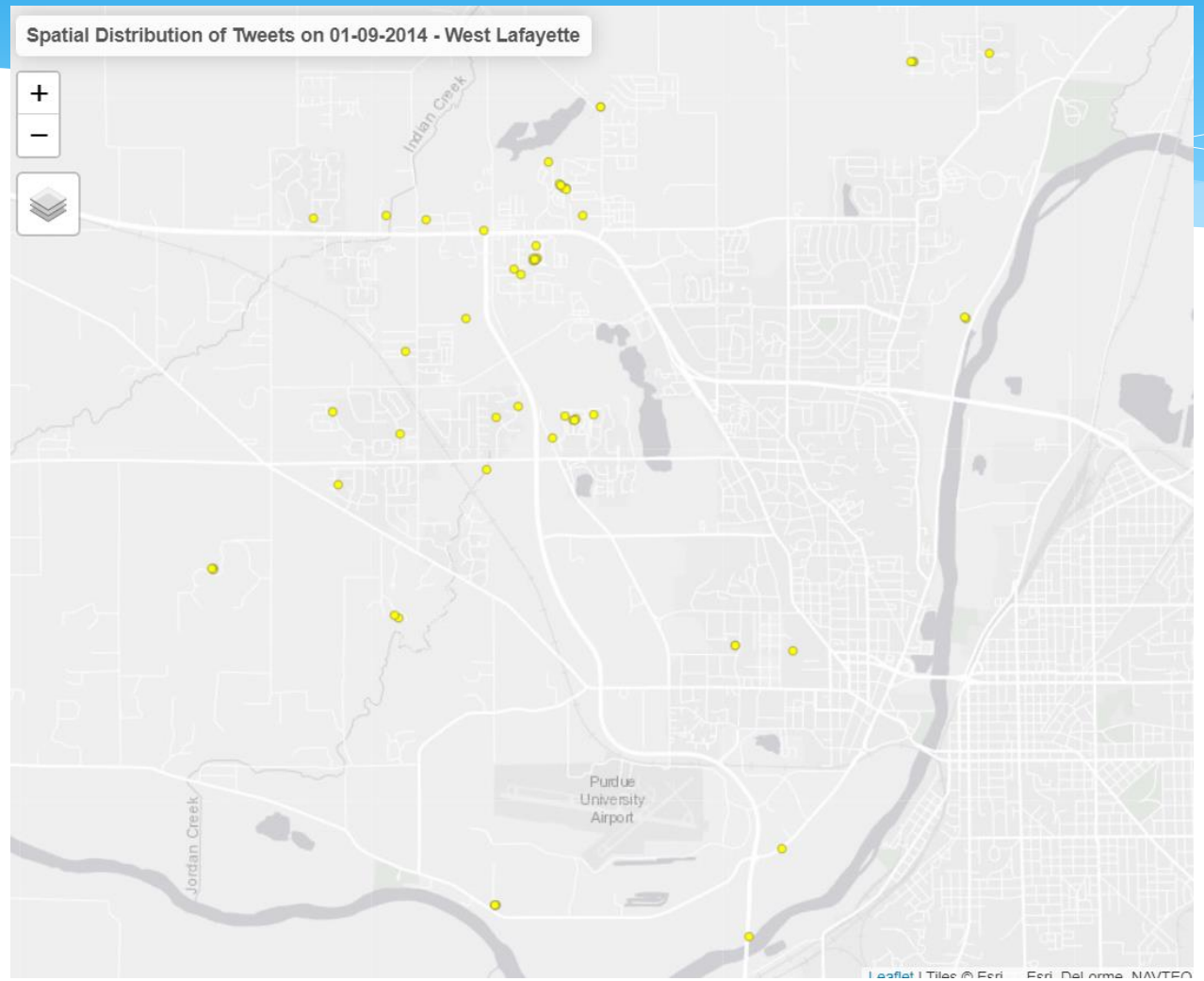
Spatial Distribution Holidays



Spatial Distribution Holidays



Spatial Distribution Holidays



Thank You