Week 10: Spatial Exercise

Melissa Wickers

12/11/2019

**Objective:** The objective for this exercise is to determine if the parking enforcements in San Francisco are Spatially Random.

**Data:** The data was obtained from <https://sf.connect.socrata.com/#!/view-data> and was filtered between the dates of March 8, 2019 and March 10, 2019. Once the data was imported into R, it was filtered even further to only include the data for “Parking Enforcement”.

**Analysis:** Once the data was filtered, a window was created using the owin() function in the spatstat package. To create the window, the x and y column of the sf.ploynomial dataset was used. Owin() uses the x range and the y range to build a boundary. Then, to form the polygon, you must set the poly attribute to a list of the x and y column. This will allow us to plot the data for the parking enforcements within a polygon that is shaped like the city of San Francisco.

window <- owin(xrange = c(min(shapes$x), max(shapes$x)), yrange = c(min(shapes$y), max(shapes$y)), poly = list(x=shapes$x,y=shapes$y))

Next, we must use ppp() function to create a point pattern for the data. You must pass in x and y, and a winsow that describes the possible range of x and y and the polygon. The window was created above. Now we can plot pp to see what the point pattern looks like.

pp <- ppp(data$Longitude, data$Latitude, window = window)

plot(pp, main = 'Plot of Parking Enforcement in San Francisco')

A close up of a piece of paper

Description automatically generated

By looking at the graph above, it appears that the points may not be spatially random. The red box indicates a group of points that are gathered in a specific location.

To determine if the points are spatial random, we must use the envelope() function to compute and plot the estimated F function for the data.

env <- envelope(pp)

plot(env, main = 'Estimated F Function')

A screenshot of a cell phone

Description automatically generated

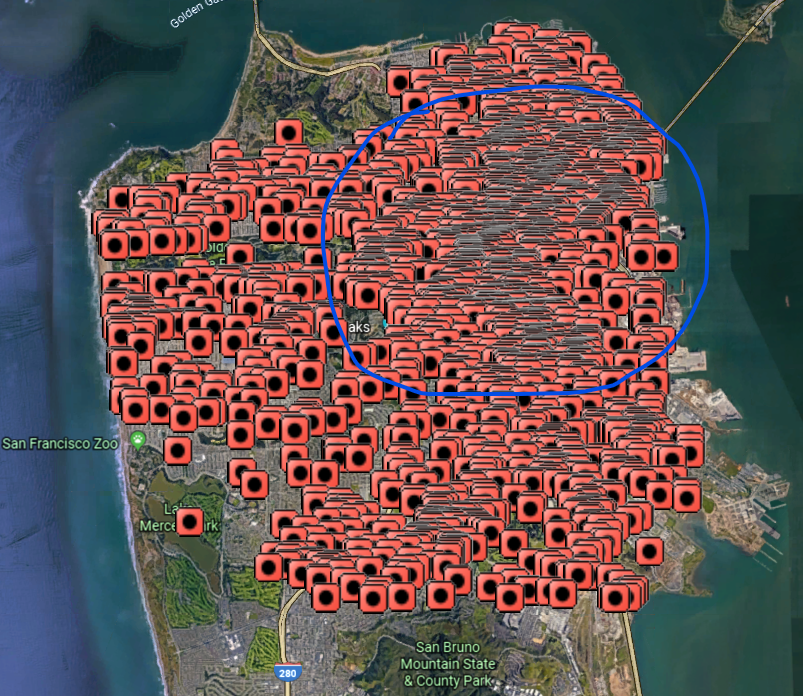
Above, we can see that the observed values and the theoretical values do not come close to each other. The theoretical values are what would be expected if the data was spatially random. Since the black line does not fall close to the red line, we can say that the data is not spatially random.

Now, lets look at a prettier picture of the data. The writekml was provided to us to write the data as a kml. The pp dataset was used to create the file. You must read in the columns that are the latitude and longitude of the data. In this case, x and y.

source('R/writekml.r'

write.kml (pp, "x", "y", outfile = "myfile2.kml", icon.num = 1)

With the kml file, we can now import the data into Google Earth. To do so, navigate to Google Earth via Google Chrome, click on settings to enable kml file import, click on projects, create new project, then select import kml. This will upload and populate your Google Earth web browser with the location that you want to see. The picture is below.



Like the plot of the pp dataset, we can see there is a significant portion of the locations located within the blue circle. This would indicate that the data is not spatially random.

**Conclusion:** Based on the analysis above, we can say that the Parking Enforcement locations in San Francisco are not spatially random.