Project 4. Spatial Pattern Analysis of Urban Points of Interest

Abstract

This project conducted a spatial pattern analysis based on Chicago's Points of Interest (POI). Initially, we practiced reading in the data provided by the professor and properly displaying the map. Subsequently, a density map was created to analyze the pattern of all POIs. Afterward, the K functions, along with hypothesis tests on Complete Spatial Randomness (CSR), were conducted to determine whether the distribution of the POIs was random. In the second part of the assignment, population data were integrated for further analysis. Finally, the project compared the results between the built-in K-function and the one conducted independently.

1. Overall Objective

The initial section of the project predominantly concentrates on introducing Points of Interest (POI). The POI data, furnished by the professor include Hospitals, Clinics, Restaurants, Supermarkets, Groceries, and Parks. For the purpose of this research, additional data pertaining to Police Stations and CTA Rail Stations were selected and downloaded from the City of Chicago Data Portal.

2. Spatial point pattern exploration (Using R built-in Packages/functions)

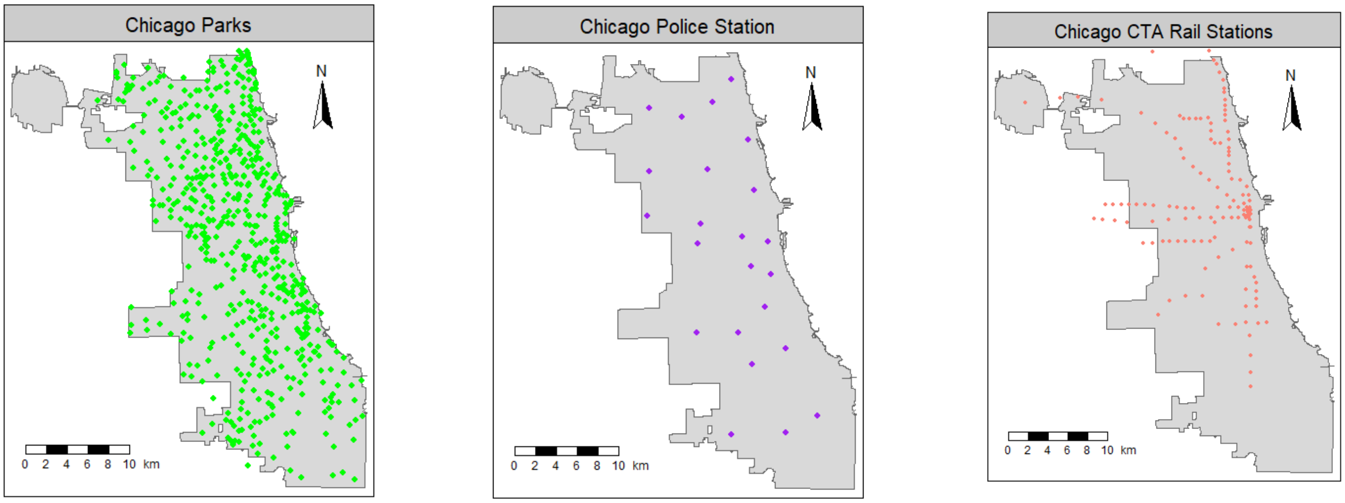
The initial procedure within this section of the project involves importing the provided data into R and accurately displaying it on an appropriate base map and coordinate system (2.1). During this phase, CSV files and shapefiles will be read into R, necessitating the cleansing of empty data (NA). Specifically, for the restaurant data, which pertains to food inspection, it is imperative to first isolate "restaurant" entries within the facility type column and subsequently select "pass" within the result column. Following this, data entries with empty latitude and longitude fields will also be excised. An exemplar of this process can be observed in the subsequent screenshot.

```
#Food Inspection
food_inspection <- read.csv("./Chicago Data/Food_Inspections.csv", header = TRUE)

#chose the restaurant and the passed only
food_inspection <- food_inspection %>%filter(Facility.Type == "Restaurant", Results == "Pass")

#clear missing data
food_inspection <- food_inspection[complete.cases(food_inspection$Longitude, food_inspection$Latitude), ]
resturant <- st_as_sf(x = food_inspection, coords = c("Longitude", "Latitude"), crs = 4236)
```

After importing the data, all coordinates will be converted to the NAD 83/UTM 16N system to optimize further analysis. The initial mapping series in this section aims to plot all the POIs within the Chicago area. Figure 1 illustrates an example of parks scattered around the Chicago area. For additional data sources, I selected the Police Station and CTA Rail Station for analysis (refer to Figure 01). Regarding the Rail Station, it is apparent that most stations align linearly, vividly illustrating the routes of the CTA trains through this pattern. The rest of the maps are listed in the references .

Figure 01

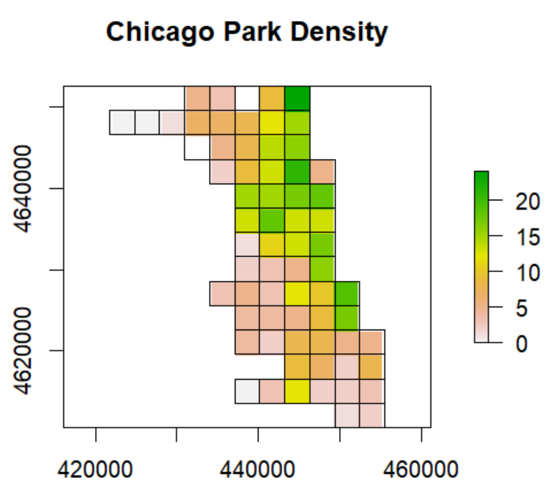
Park, Police Station, and CTA Rail Station at Chicago Area



The next step of this project is to create density map of each POI and conduct analysis. For this part of the project, we would first need to convert the sf files into the sp files (Chicago) for the further use, and then we need to rasterize the Chicago boundary into grids and then do the plotting. In this process, the cell size of Chicago was set to 3000 meters. Once the POIs are implemented, the density of each cell with respect to the number of POIs can be visualized. Figure 2 shows an example of the density map of parks in Chicago area.

Figure 02

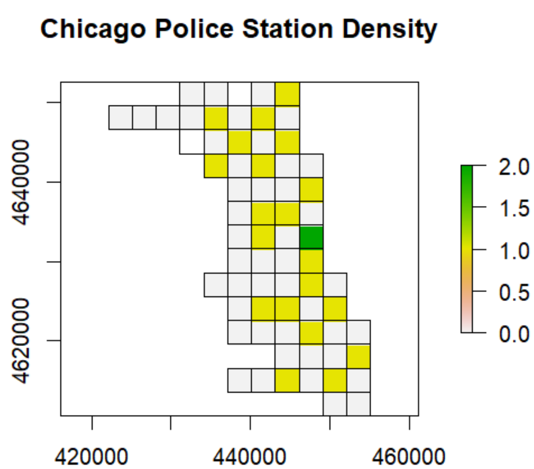
Density Map of Chicago Parks



It is clear to see that most of the Chicago areas have parks; however, the northeastern side of Chicago has relatively more parks than other areas. Another iconic density map was the police density shown in Figure 3. Clearly, the police stations (departments) are more “equally” distributed around the city area. This probably refers to the demand for public safety. Another interesting thing I found is that the northwest tail, where the Chicago O’Hare International Airport is located, has no police station indicated. Although it is known that the airport has its own police department, one possible reason is that the data excluded the airport police department.

Figure 03

Density map of Chicago Police Stations



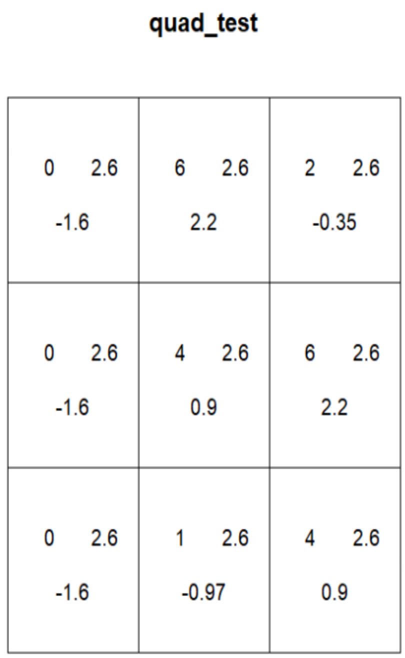
The next step of this project involves conducting a K function analysis along with a hypothesis test on Complete Spatial Randomness (CSR). The hypothesis test can be executed using a Quadrat Test with a Chi-squared distribution. The first step involves creating a point pattern object using the "ppp" function. In the "ppp" function, 'x' and 'y' represent the coordinates. Before establishing 'x' and 'y', since the project employs NAD 83, it is necessary to set up the northing and easting, and subsequently create the bounding box for the Chicago sp files. The following screenshot demonstrates the code used to establish the northing and easting for the coordinate setting before configuring the 'x' and 'y' coordinates.

```
#quadrant test

eastnorth <- st_coordinates(parks)
parks$Easting <- eastnorth[,1]
parks$Northing <- eastnorth[,2]
#bbox
bbx <- bbox(chicago_sp)
xmin <- bbx[1, 1]
xmax <- bbx[1, 2]
ymin <- bbx[2, 1]
ymax <- bbx[2, 2]
park_ppp <- ppp(parks$Easting, parks$Northing, owin(c(xmin, xmax), c(ymin, ymax)))
```

After this step, the next step is to run the “ppp” function and conduct the quadrat test. The results of the Chi-square hypothesis test are listed in the reference. Figure 04 shows an example of the results of the quad test and hypothesis screenshot.

Figure 04

Quadrat Test and Hypothesis Screenshot of Police Station

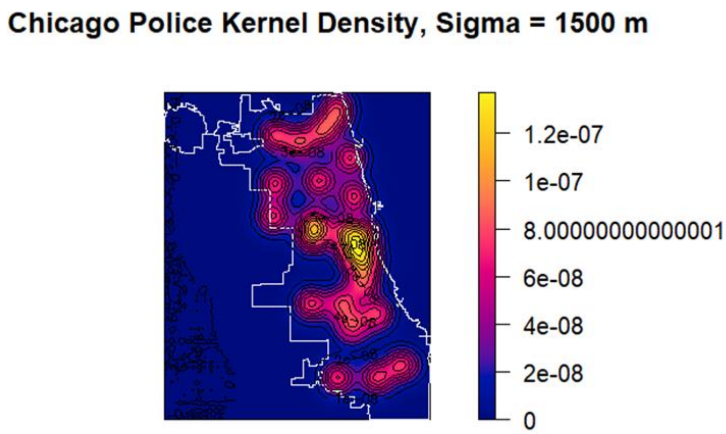



```
> quad_test

        Chi-squared test of CSR using quadrat counts

data:  police_ppp
X2 = 19.652, df = 8, p-value = 0.02347
alternative hypothesis: two.sided

Quadrats: 3 by 3 grid of tiles
>
```

In the context of utilizing quadrat analysis to evaluate Complete Spatial Randomness (CSR) and applying the Chi-square test, the p-values provide information about the probability that the number of points in each quadrat would occur if the null hypothesis of CSR were true. All the p-values for Points of Interest (POIs) in the appendix were less than 0.05, indicating that non-random processes might influence the distribution of points. In other words, some POIs, such as restaurants and groceries, are likely to have a higher concentration in areas with larger populations. However, the result for the Police Station, presented in Figure 04, is around 0.02, indicating that the distribution of police stations does not significantly deviate from a random distribution. This aligns with the pragmatic necessity to have police stations dispersed throughout the area, regardless of population density. The Kernel Density graph also shows that the distribution of the police stations is located in every area of Chicago.

Figure 05

Kernel Density Police Station Chicago

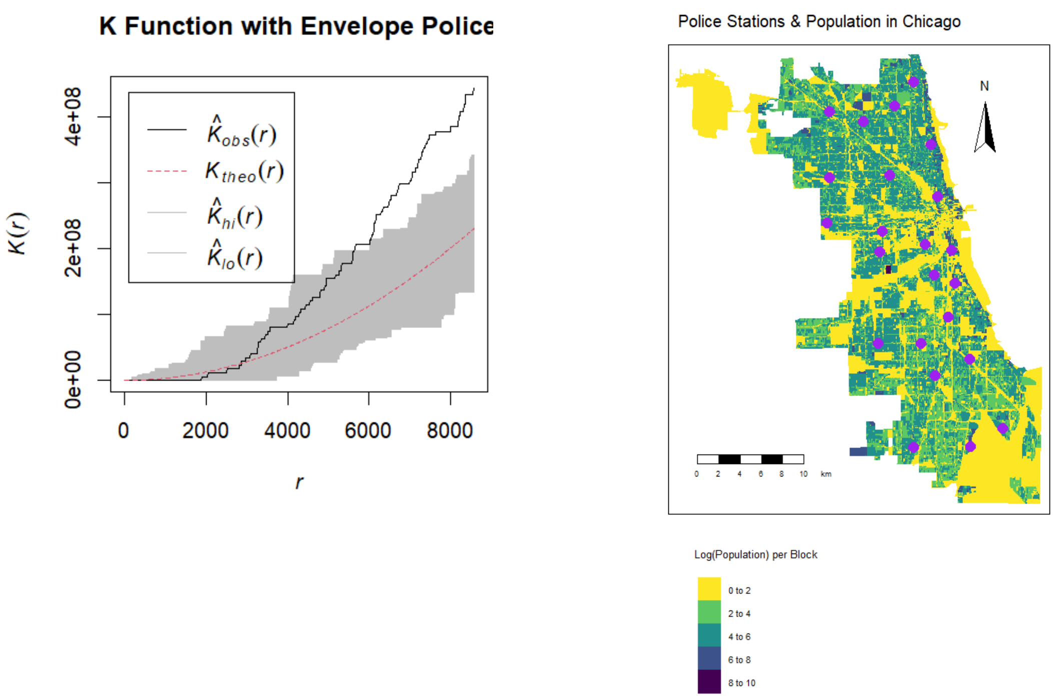


3. Urban Structure Analysis

The K function will be implemented for this part of the analysis. The screenshots of the figures containing POIs are listed in the appendix. The figure contains four different lines. The observed K-function represents the actual data and shows how the expected number of additional points within a distance r changes with varying r in the dataset. The theoretical K represents the null hypothesis obtained from the previous CSR part. The high and the low represent the upper and lower bounds of the envelope. Considering statistical variability, these bounds provide a range within which the K-function for a truly random pattern would be expected to fall. If we can see the observed point outside the envelope (boundary), we can determine that the observed pattern significantly departs from the null hypothesis at the scale r. The results show that the POIs, including restaurants, hospitals, parks, groceries, and supermarkets, are clustered more closely than expected (theoretical). It means that the result is affected by other non-random variables. One possible reason can be related to the population distribution within the area. To conduct further analysis. The Census population was implemented for better analysis. I used Census Block data for the population implementation. After inputting the shapefile, and the Census Block population, two sets of data were combined using the “left_join” function following the GEOID10 column. After plotting the data, it is clear to see that the northern side of Chicago has more population than the southern side of Chicago. This can be the reason that most POIs are clustered. In contrast, Police stations are close to non-clustering (or close to not being affected by non-randomized variants). Figure 06 shows the figure of police station’s K-function with the envelope. Within certain r, the location of the police station is not affected by non-random variant (population). The left side of Figure 06 shows the population density map of Chicago.

Figure 06

Police K-function Results and Population Density



4. Implementation of K function

The main objective of this section is to create our own K function. The K function is defined as

$$K(r) = \lambda^{-1}E[(r)]$$

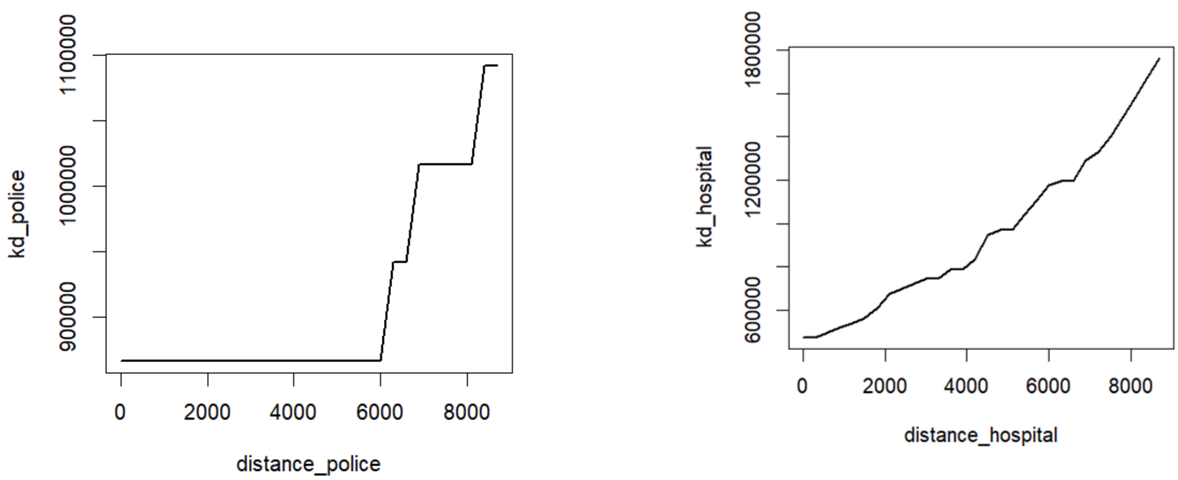
For this part, I first find out the columns representing the coordinates, Subsequently, a data frame of coordinates is constructed, and a key is assigned to facilitate future joins. This ensures the organized storage of spatial points in a structured format. Then, I compute the pairwise distances between all points using the Euclidean Distance formula. After the Euclidean Distance between two points in a plane is determined, a matrix is generated to encapsulate all the computed distance data, ensuring that these vital calculations are stored systematically for subsequent analyses. The subsequent step involves enumerating the number of point pairs that are located within each specified distance. Then, normalization is executed by adjusting the counts based on the area and the number of distances examined. The following screenshot shows an example of the self K-function programming.

```
x_police = police$x
y_police = police$y
df_police = data.frame(x_police, y_police)

#set up a fake key
df_police <- df_police %>% mutate (k = 1)

#perform the join, remove the key, then create the distance
df_police = df_police %>%
  full_join(df_police, by = "k") %>%
  mutate(dist = sqrt((x_police.x - x_police.y)^2 + (y_police.x - y_police.y)^2))

#Euclidean calculation
dis_police = as.matrix(df_police$dist)


m_police = matrix(0, nrow = nrow(df_police), ncol = nrow(df_police))
for(i in 1:nrow(m_police)){
  for(j in 1:nrow(m_police)){
    m_police[i, j] = dis_police[(i-1)*nrow(m_police) + j]
  }
}
```

Two POIs were selected to compare the result with the built-in R functions. As for the hospital result, there isn't a significant difference compared to the K function result. However, Figure 07 shows that there are differences while plotting the K-function. One possible reason can be the initial setting during the Euclidean distance formula setup.

Figure 07

Self K Function Police Station



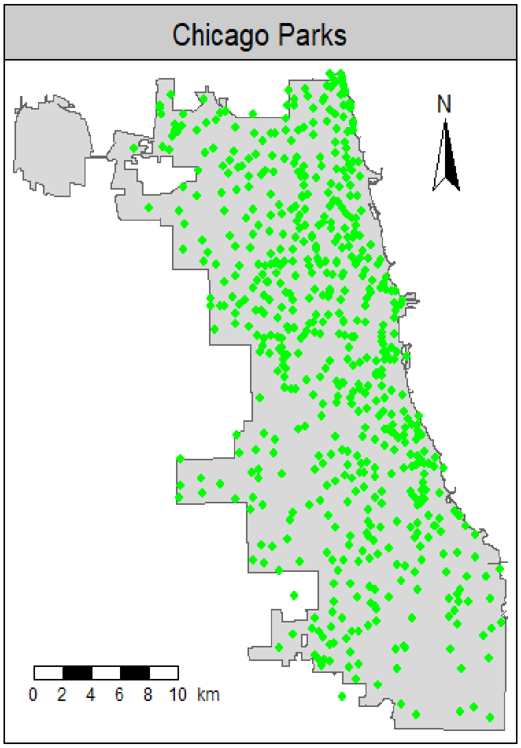
5. Summary/Conclusion/Concluding Remarks

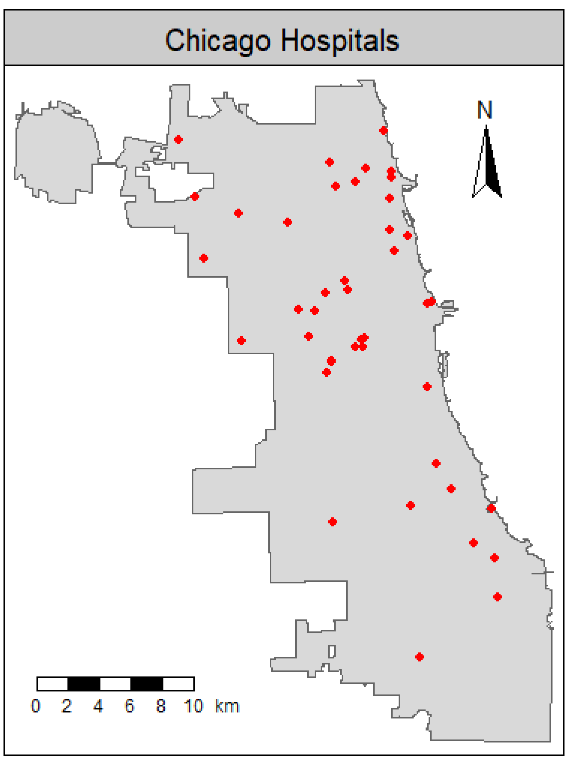
This study conducted the spatial pattern analysis of Points of Interest (POIs) in Chicago. The analysis showed that most of the POIs were not randomly distributed and could be affected by other variants, such as the population. In contrast, some public safety authorities were not significantly affected by population. Although there are slightly more police stations located in areas with a larger population, areas with larger populations also have some police stations.

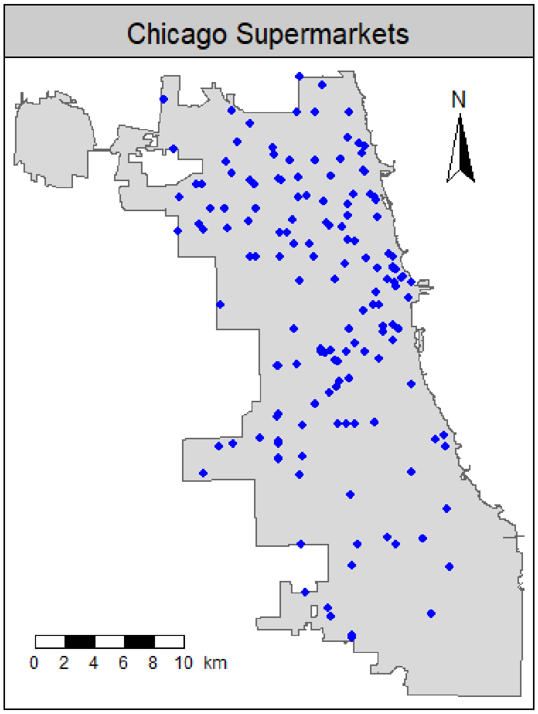
Acknowledgment

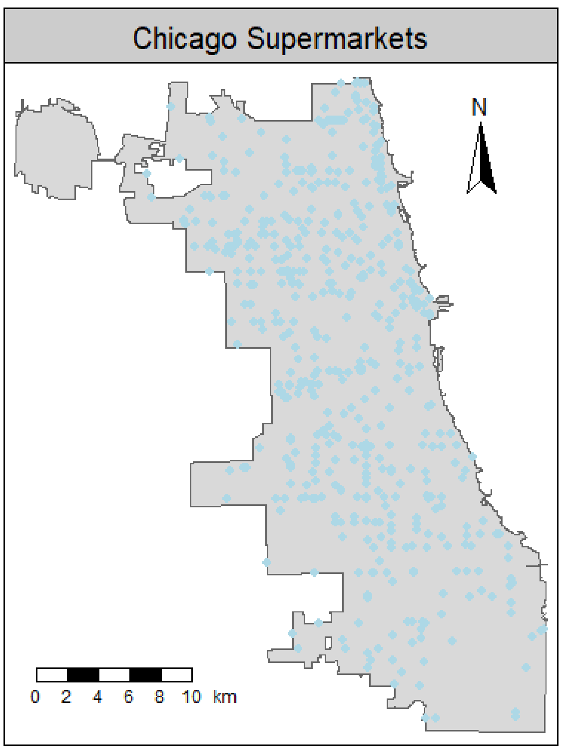
References

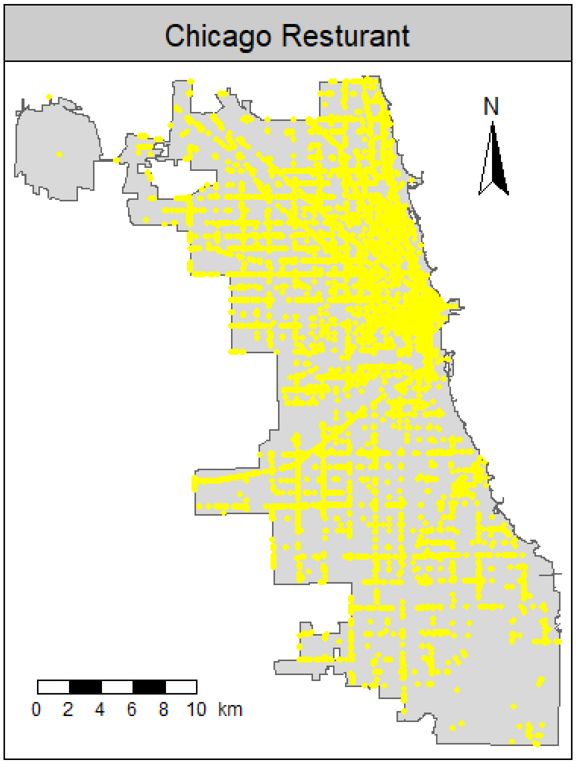
POIs

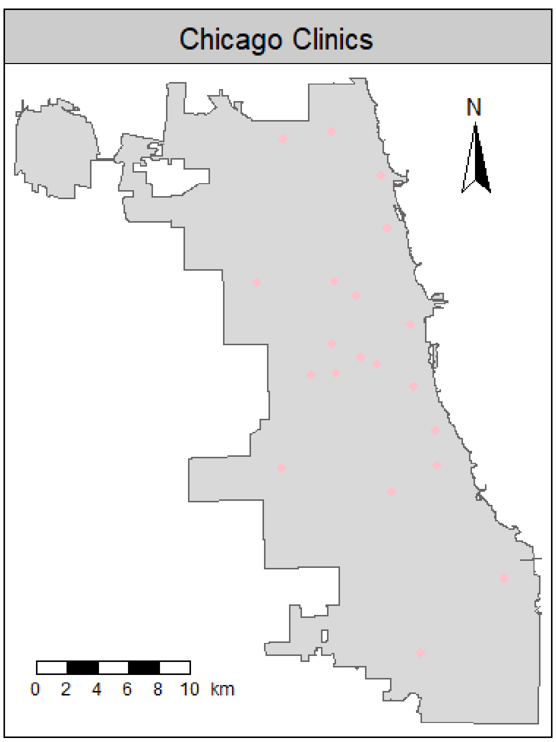


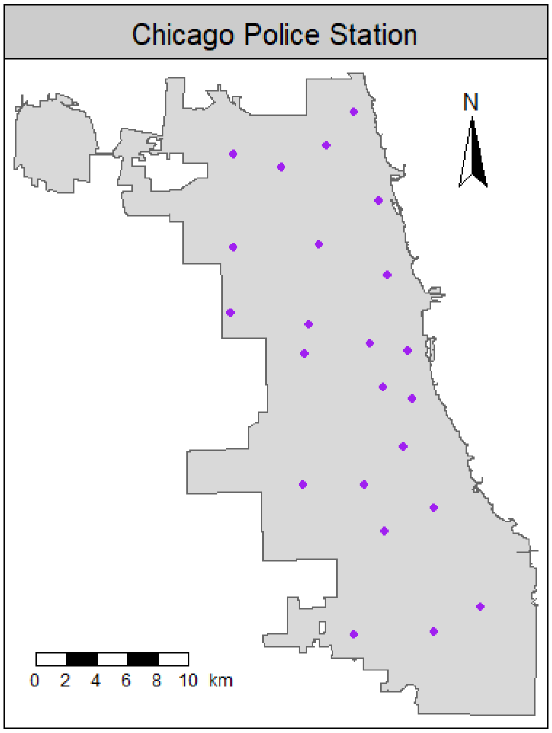


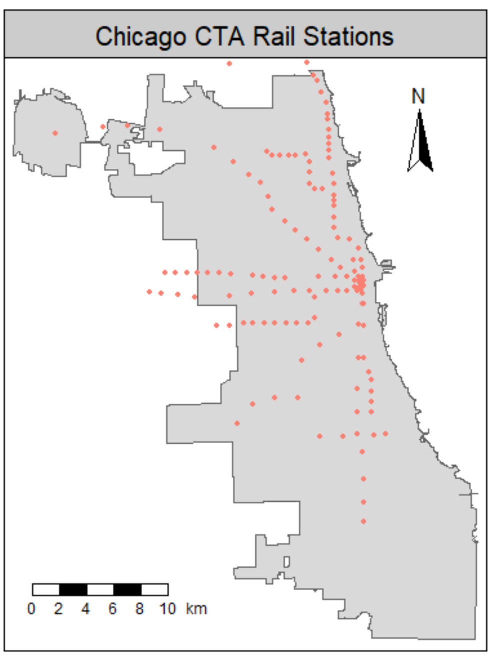

Chicago Supermarkets
N
0 2 4 6 8 10 km

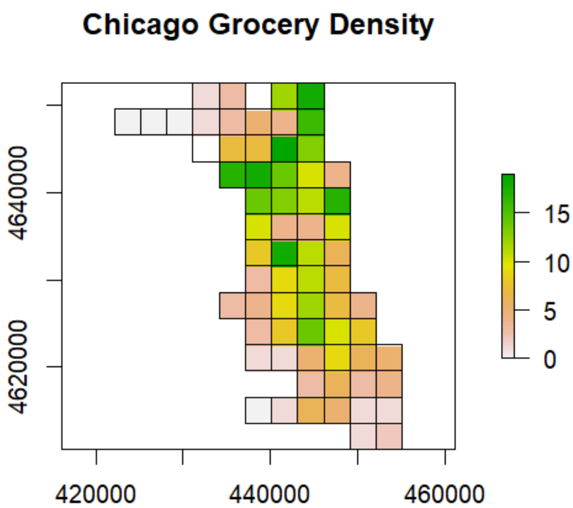

Chicago Supermarkets
N
0 2 4 6 8 10 km

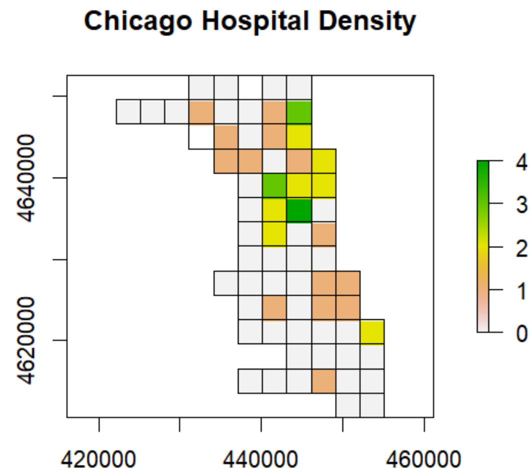

Chicago Resturant
N
0 2 4 6 8 10 km

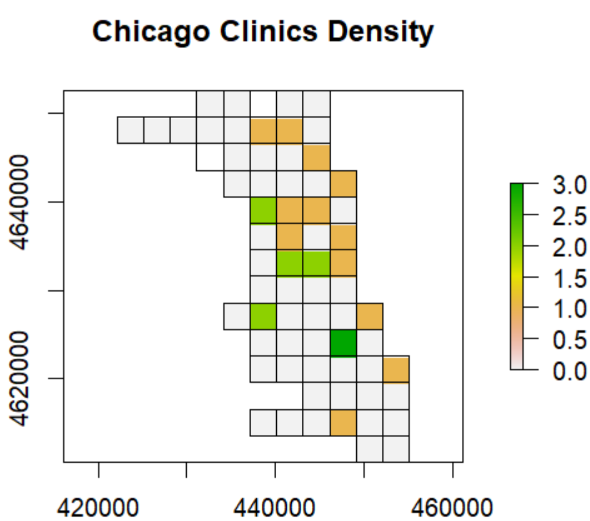

Chicago Clinics
N
0 2 4 6 8 10 km

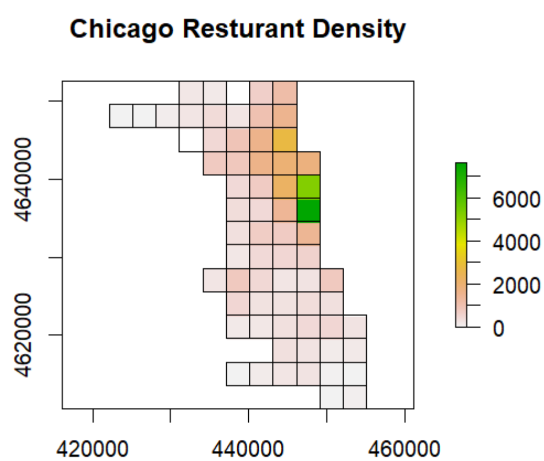

Chicago Police Station
N
0 2 4 6 8 10 km

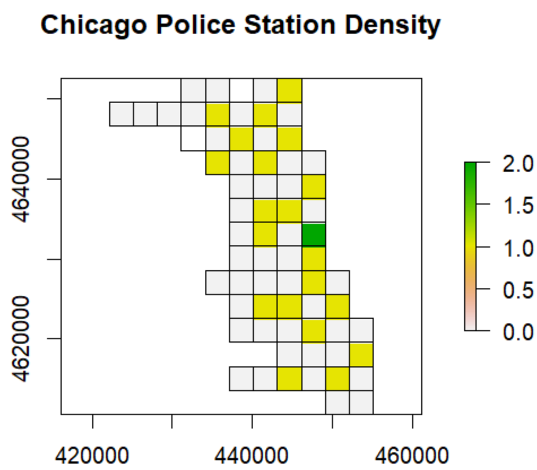

Chicago CTA Rail Stations
N
0 2 4 6 8 10 km

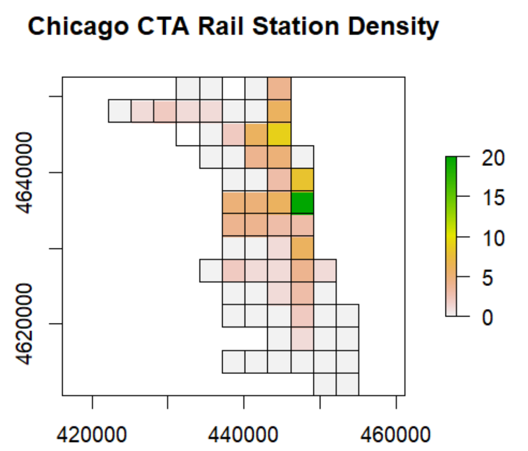

Chicago Grocery Density
4640000
4620000
420000
440000
460000
15
10
5
0

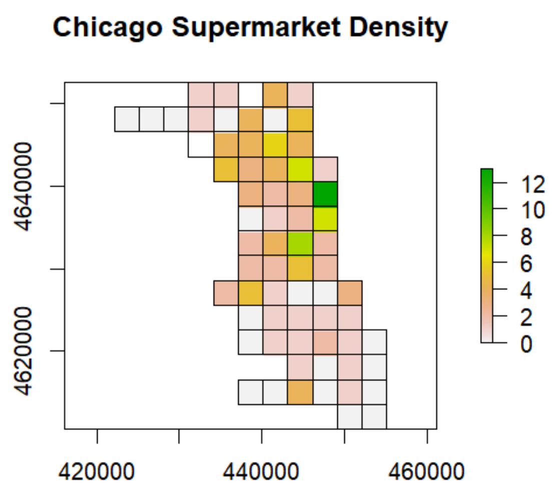

Chicago Hospital Density
4640000
4620000
420000
440000
460000
4
3
2
1
0


Chicago Clinics Density
4640000
4620000
420000
440000
460000
3.0
2.5
2.0
1.5
1.0
0.5
0.0

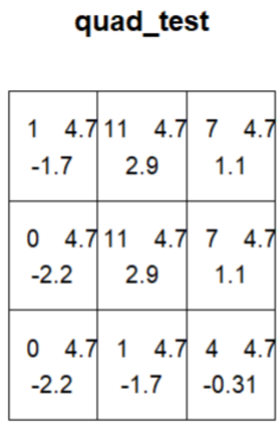

Chicago Resturant Density
4640000
4620000
420000
440000
460000
6000
4000
2000
0


Chicago Police Station Density
4640000
4620000
420000
440000
460000
2.0
1.5
1.0
0.5
0.0

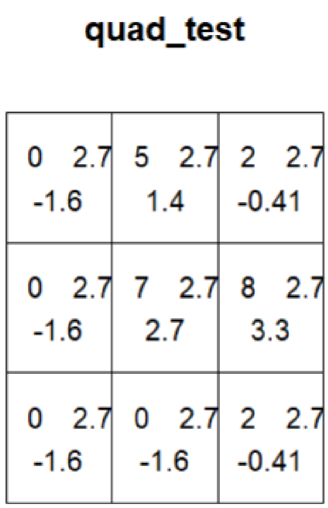

Chicago CTA Rail Station Density
4640000
4620000
420000
440000
460000
20
15
10
5
0


Chicago Supermarket Density
4640000
4620000
420000
440000
460000
12
10
8
6
4
2
0

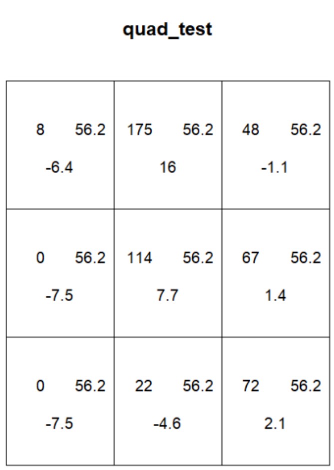
Quad CSR Chi-Square



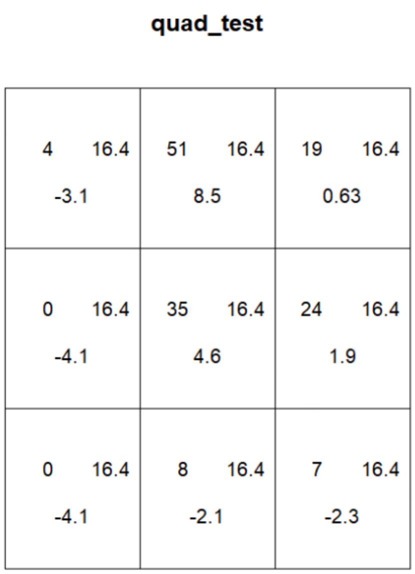

```
 Chi-squared test of CSR using quadrat counts

 park_ppp
28.99, df = 8, p-value < 2.2e-16
ative hypothesis: two.sided

ts: 3 by 3 grid of tiles
```



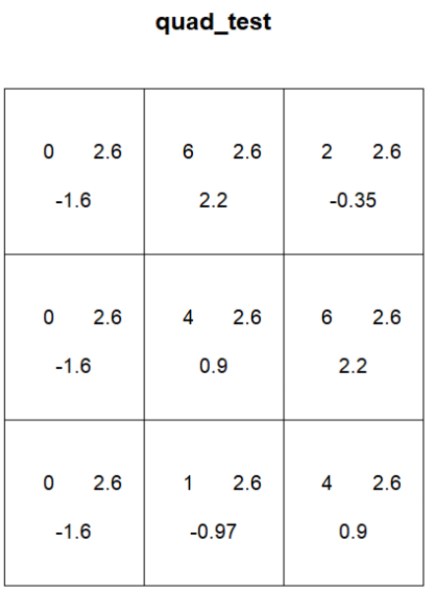

```
> quad_test

        Chi-squared test of CSR using quadrat counts

data:  clinics_ppp
X2 = 30.75, df = 8, p-value = 0.0003113
alternative hypothesis: two.sided

Quadrats: 3 by 3 grid of tiles
>
```



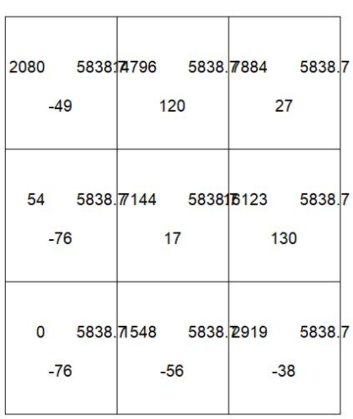

```
> quad_test

        Chi-squared test of CSR using quadrat counts

data:  grocery_ppp
X2 = 492.64, df = 8, p-value < 2.2e-16
alternative hypothesis: two.sided

Quadrats: 3 by 3 grid of tiles
>
```




```
> quad_test

        Chi-squared test of CSR using quadrat counts

data:  supermarket_ppp
X2 = 149.49, df = 8, p-value < 2.2e-16
alternative hypothesis: two.sided

Quadrats: 3 by 3 grid of tiles
>
```



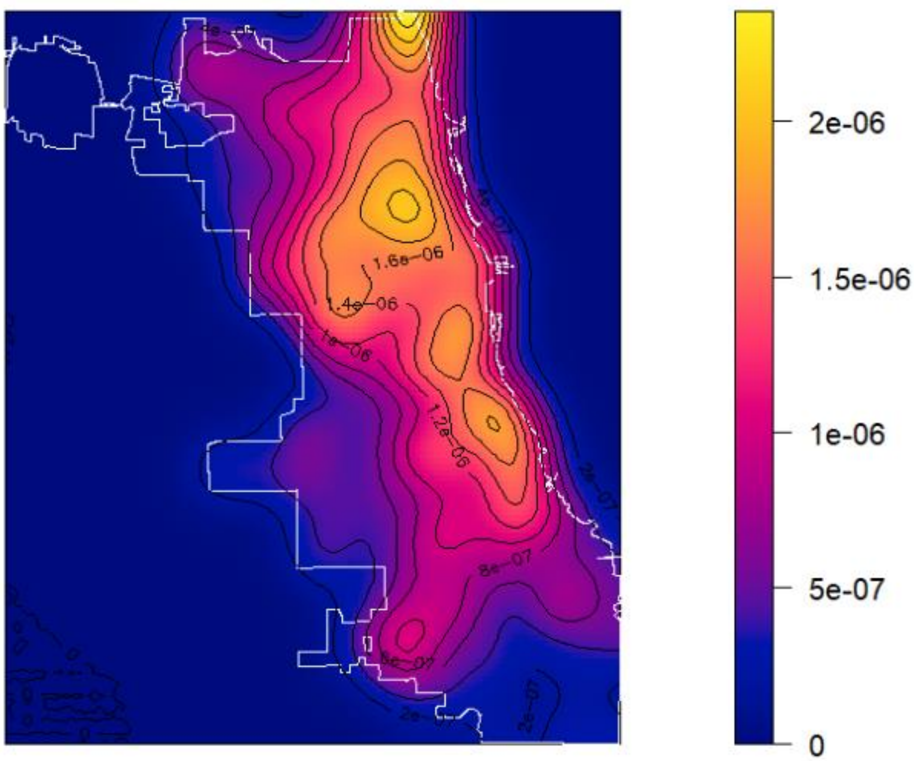

```
> quad_test

        Chi-squared test of CSR using quadrat counts

data:  hospital_ppp
X2 = 34.714, df = 8, p-value = 6.028e-05
alternative hypothesis: two.sided

Quadrats: 3 by 3 grid of tiles
>
```



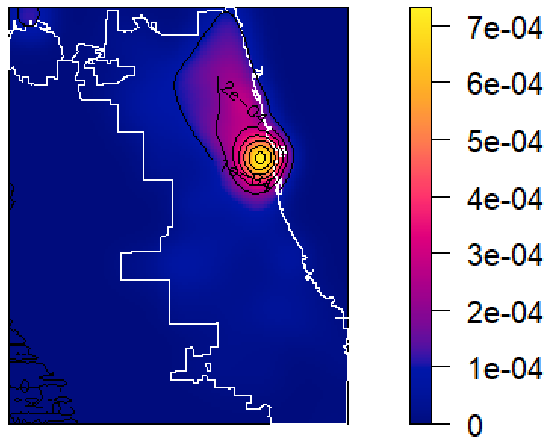

```
> quad_test

        Chi-squared test of CSR using quadrat counts

data:  resturant_ppp
X2 = 51468, df = 8, p-value < 2.2e-16
alternative hypothesis: two.sided

Quadrats: 3 by 3 grid of tiles
>
```

```
> quad_test

        Chi-squared test of CSR using quadrat counts

data:  police_ppp
X2 = 19.652, df = 8, p-value = 0.02347
alternative hypothesis: two.sided

Quadrats: 3 by 3 grid of tiles
>
```

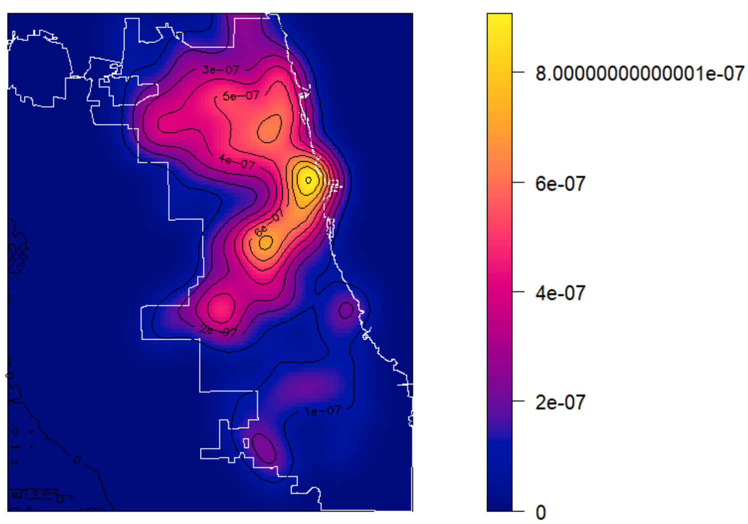
Chicago Park Kernel Density, sigma = 1500 m



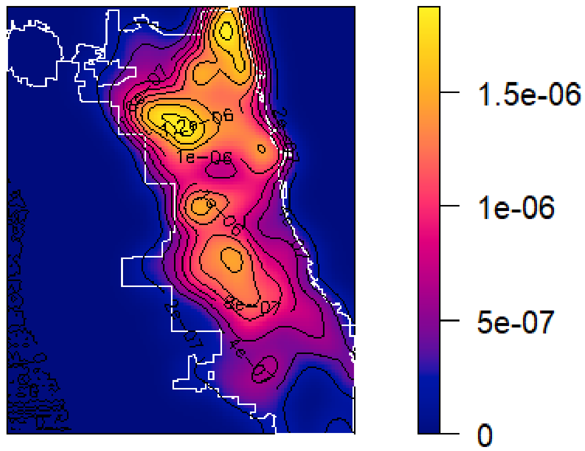
Chicago Resturant Kernel Density, Sigma = 1500 m



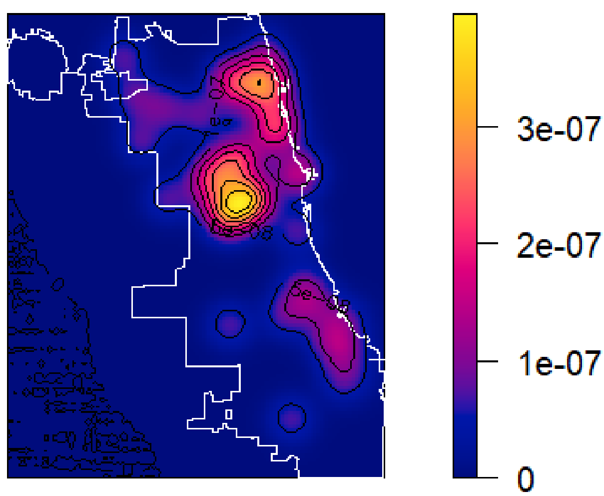
Chicago Supermarket Kernel Density, Sigma = 1500 m



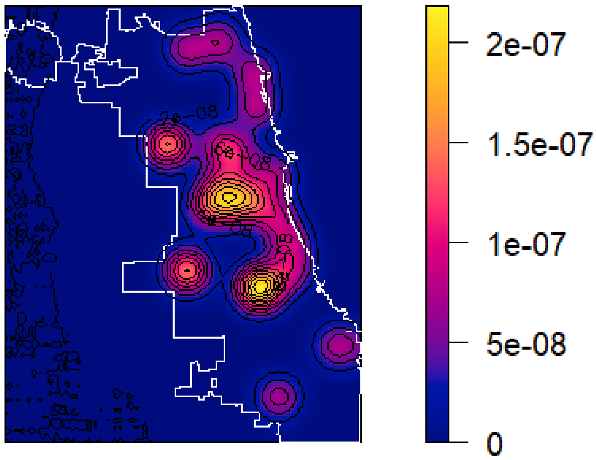
Chicago Grocery Kernel Density, Sigma = 1500 m



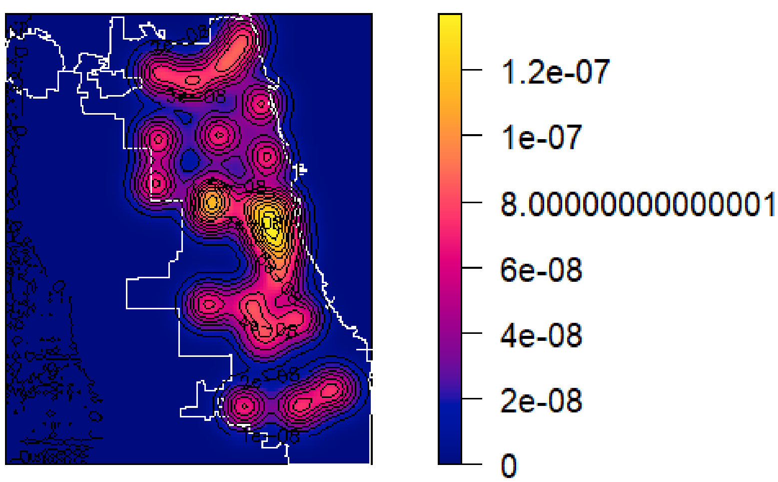
Chicago Hospital Kernel Density, Sigma = 1500 m

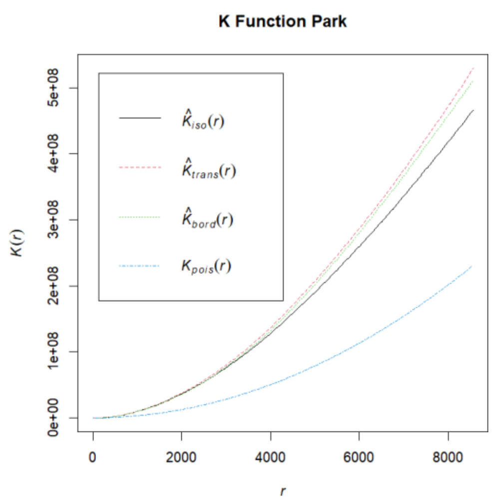


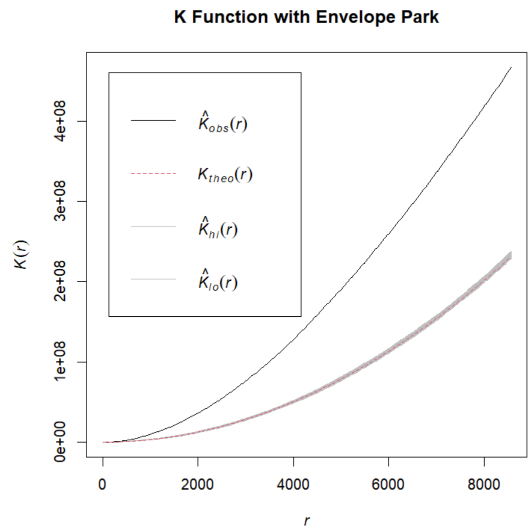
Chicago Clinics Kernel Density, Sigma = 1500 m



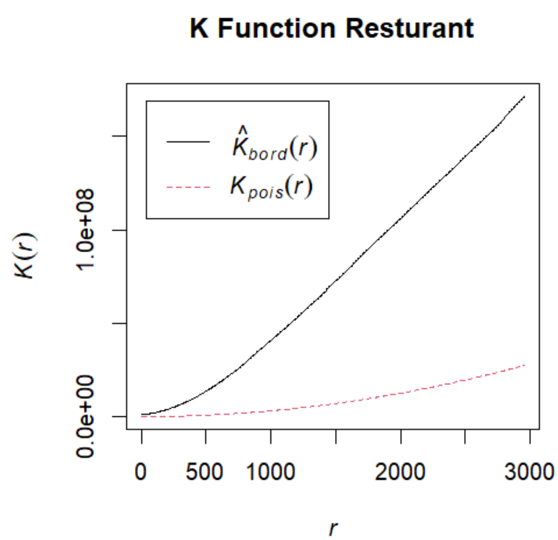
Chicago Police Kernel Density, Sigma = 1500 m

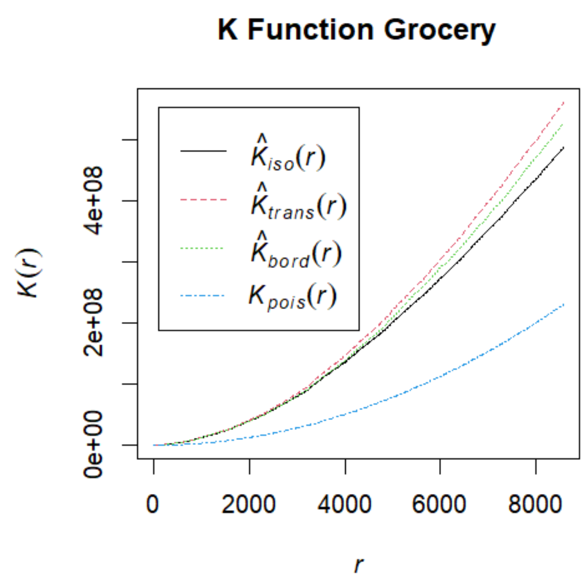


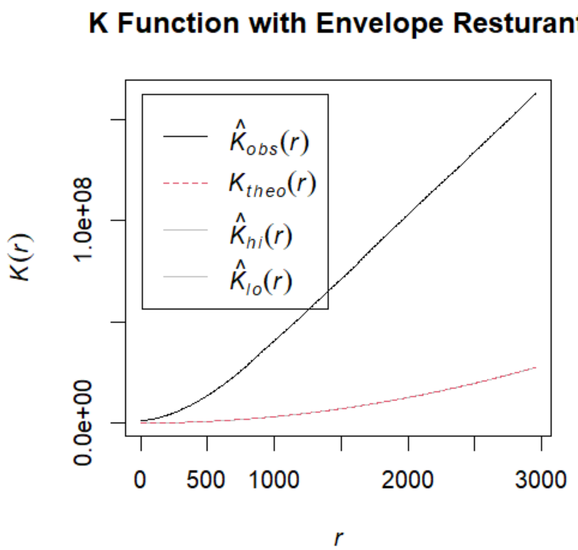


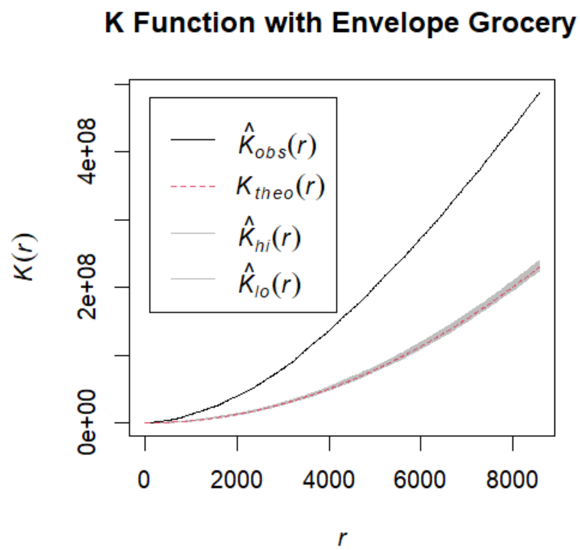


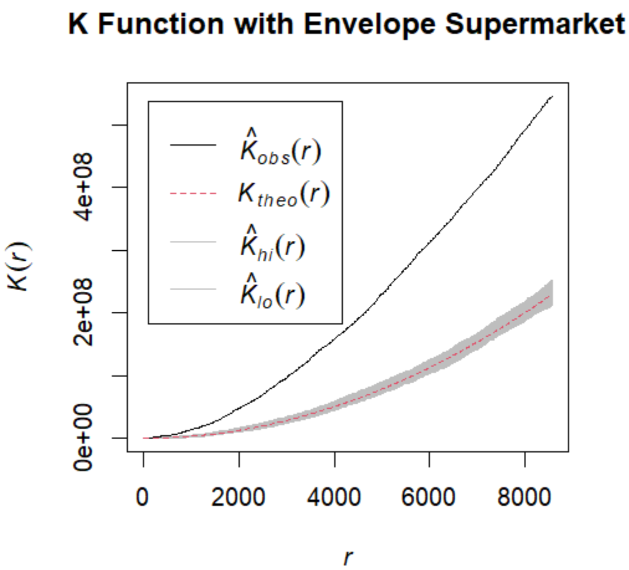

```
> plot(K_enve, main = "K Function with Envelope Park")
> #resturant
> K_func <- Kest(resturant_ppp)
number of data points exceeds 3000 - computing border correction estimate only
> K_enve <- envelope(resturant_ppp, Kest)
Generating 99 simulations of CSR  ...
```

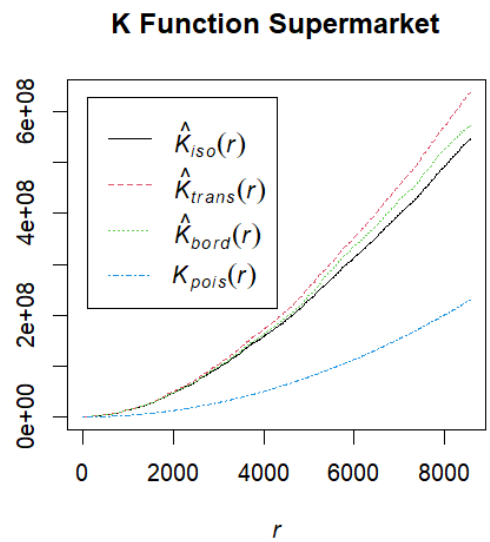


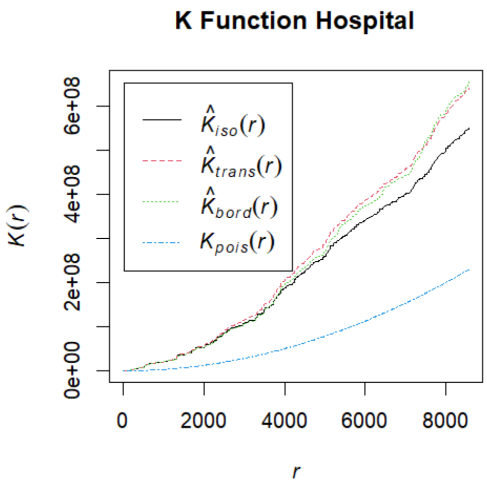


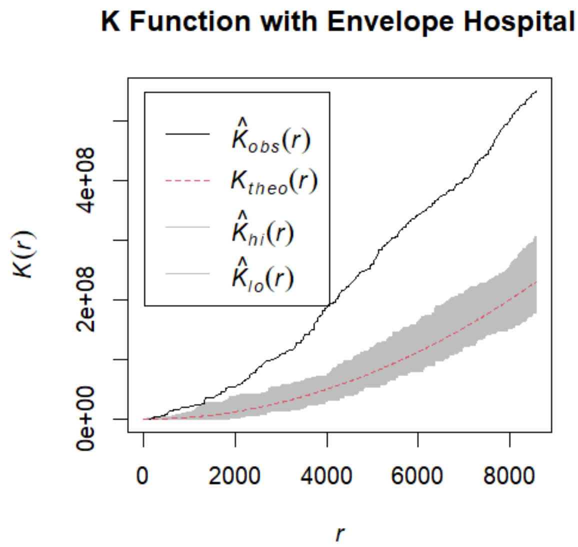

K Function with Envelope Resturant
$\hat{K}_{obs}(r)$
$K_{theo}(r)$
$\hat{K}_{hi}(r)$
$\hat{K}_{lo}(r)$
K(r)
0.0e+00
1.0e+08
0
500
1000
2000
3000
r

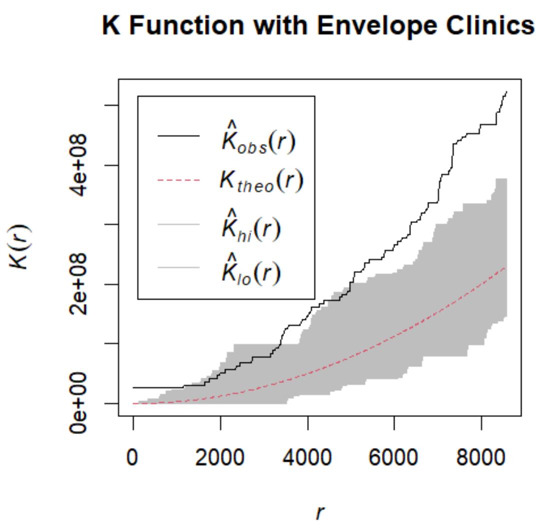

K Function with Envelope Grocery
$\hat{K}_{obs}(r)$
$K_{theo}(r)$
$\hat{K}_{hi}(r)$
$\hat{K}_{lo}(r)$
K(r)
0e+00
2e+08
4e+08
0
2000
4000
6000
8000
r

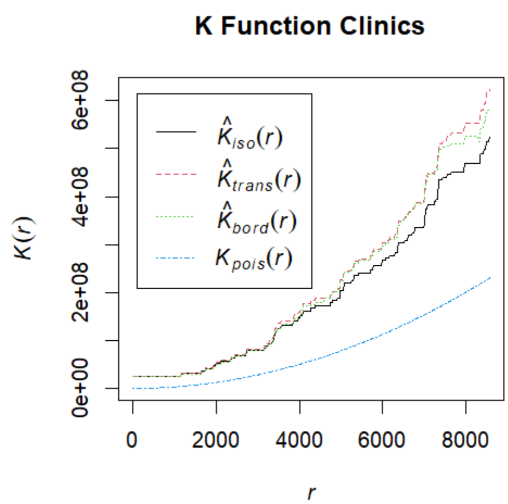

K Function with Envelope Supermarket
$\hat{K}_{obs}(r)$
$K_{theo}(r)$
$\hat{K}_{hi}(r)$
$\hat{K}_{lo}(r)$
K(r)
0e+00
2e+08
4e+08
0
2000
4000
6000
8000
r

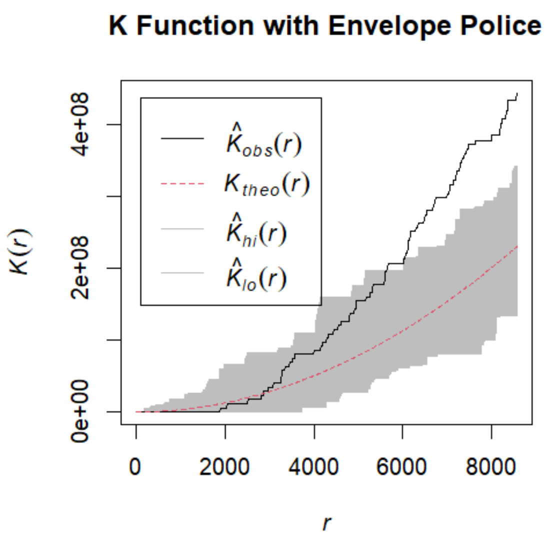

K Function Supermarket
$\hat{K}_{iso}(r)$
$\hat{K}_{trans}(r)$
$\hat{K}_{bord}(r)$
$K_{pois}(r)$
0e+00
2e+08
4e+08
6e+08
0
2000
4000
6000
8000
r

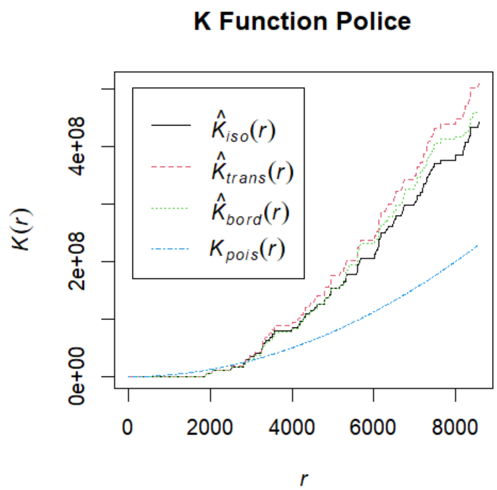

K Function Hospital
$\hat{K}_{iso}(r)$
$\hat{K}_{trans}(r)$
$\hat{K}_{bord}(r)$
$K_{pois}(r)$
K(r)
0e+00 2e+08 4e+08 6e+08
0 2000 4000 6000 8000
r


K Function with Envelope Hospital
$\hat{K}_{obs}(r)$
$K_{theo}(r)$
$\hat{K}_{hi}(r)$
$\hat{K}_{lo}(r)$
K(r)
0e+00 2e+08 4e+08
0 2000 4000 6000 8000
r

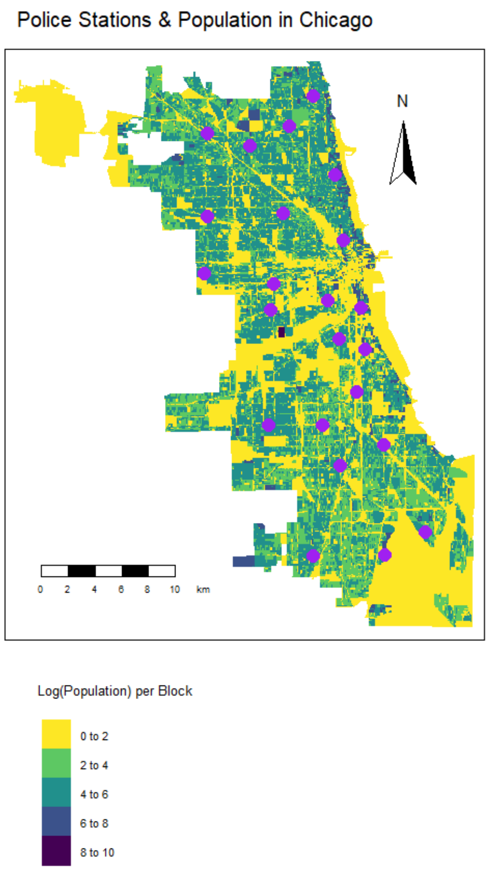

K Function with Envelope Clinics
$\hat{K}_{obs}(r)$
$K_{theo}(r)$
$\hat{K}_{hi}(r)$
$\hat{K}_{lo}(r)$
K(r)
0e+00 2e+08 4e+08
0 2000 4000 6000 8000
r

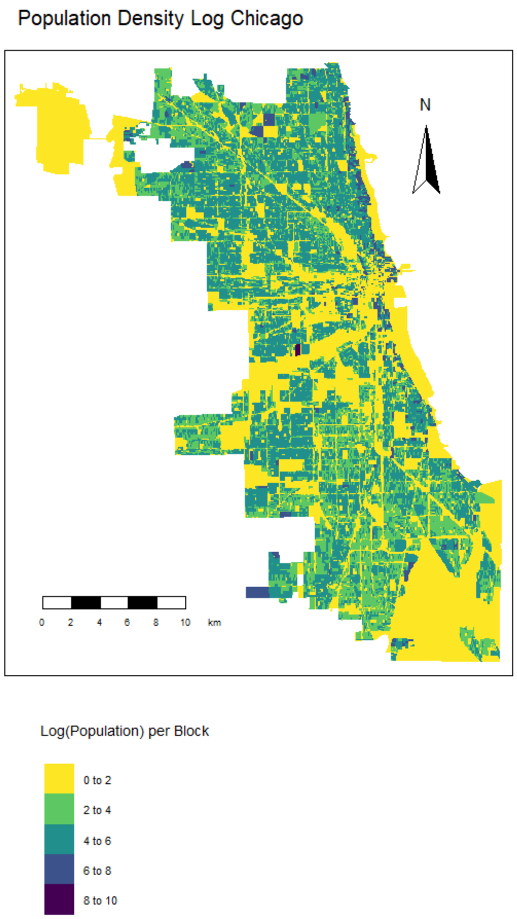

K Function Clinics
$\hat{K}_{iso}(r)$
$\hat{K}_{trans}(r)$
$\hat{K}_{bord}(r)$
$K_{pois}(r)$
K(r)
0e+00 2e+08 4e+08 6e+08
0 2000 4000 6000 8000
r

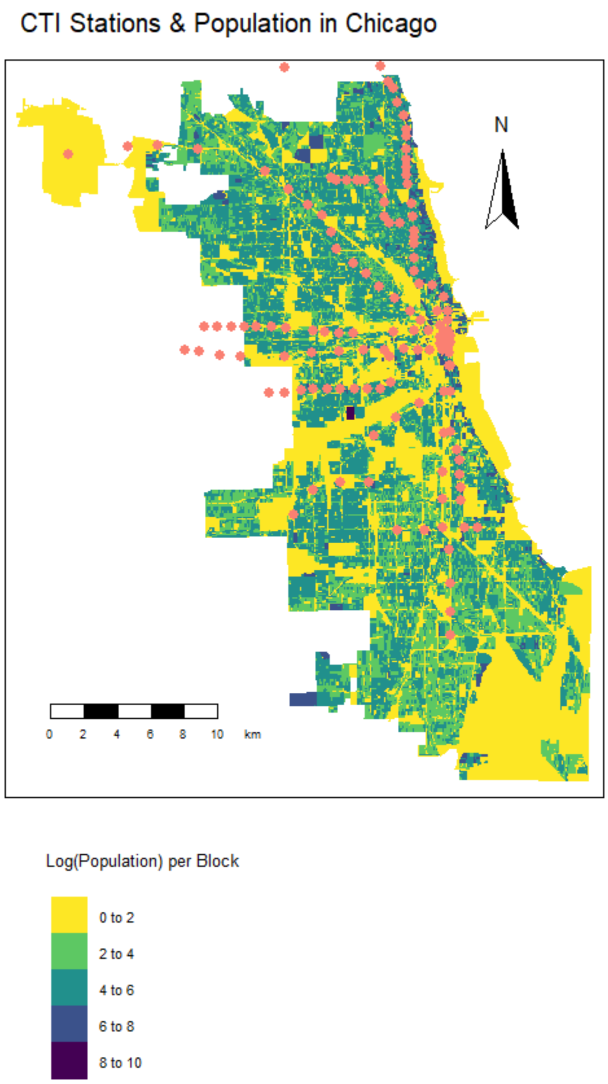


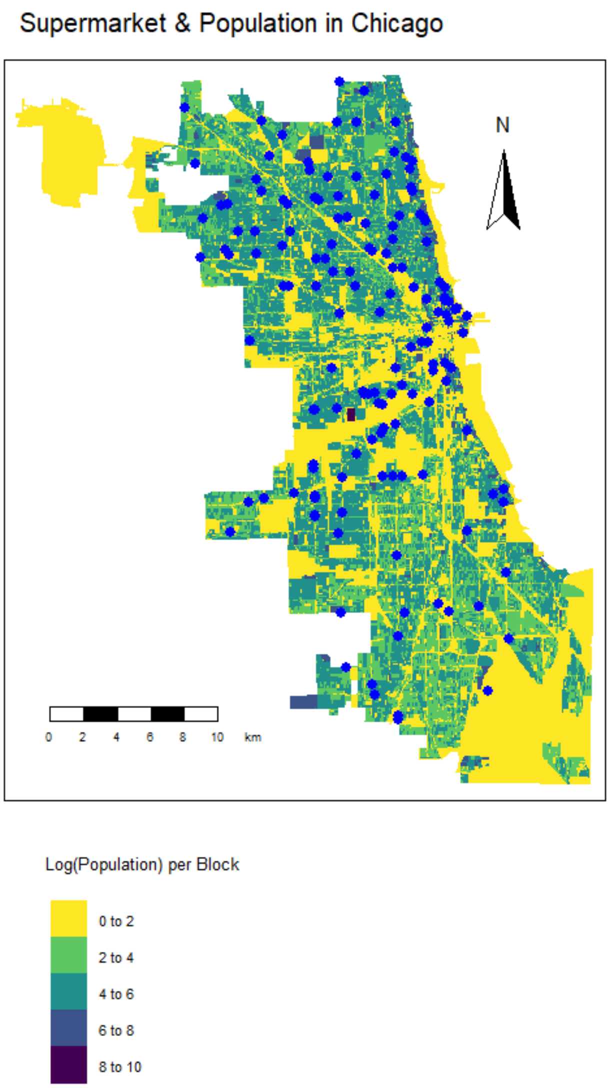


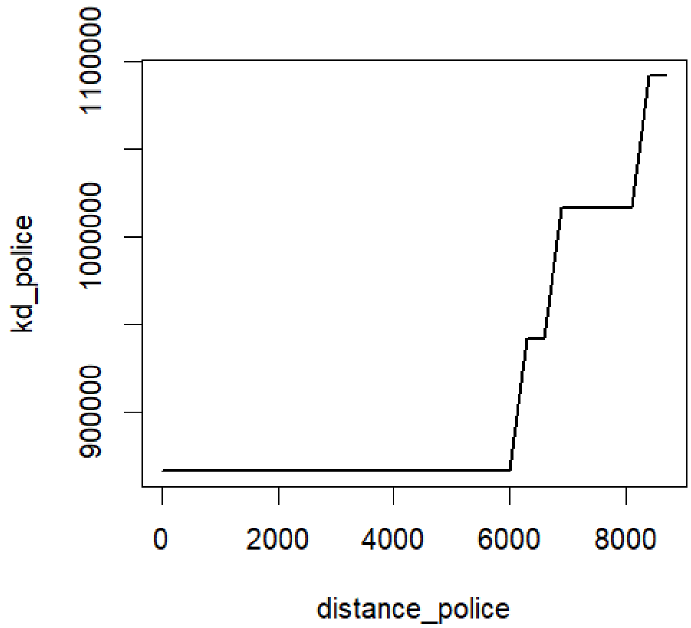
Population Density

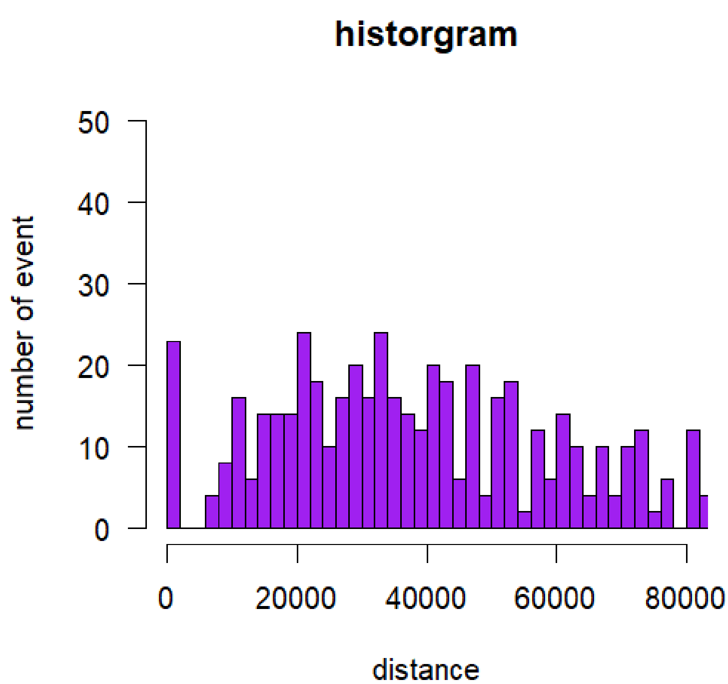


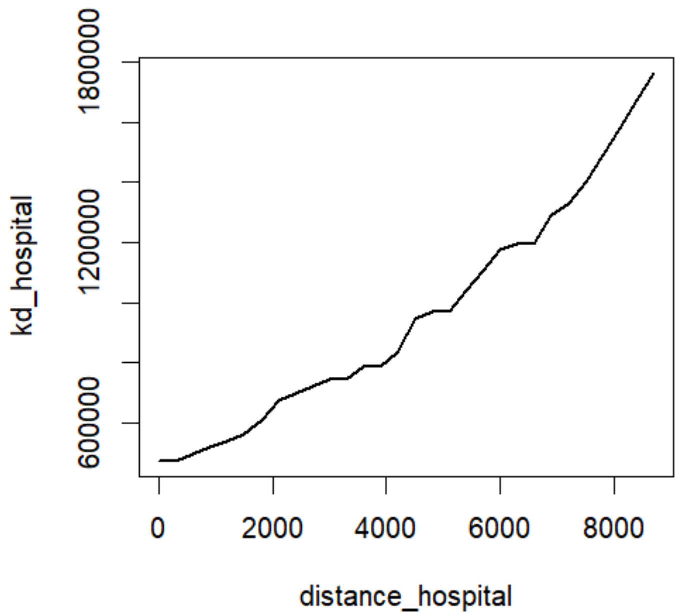

Population Density Log Chicago
N
0 2 4 6 8 10 km
Log(Population) per Block
0 to 2
2 to 4
4 to 6
6 to 8
8 to 10


CTI Stations & Population in Chicago
N
0 2 4 6 8 10 km
Log(Population) per Block
0 to 2
2 to 4
4 to 6
6 to 8
8 to 10

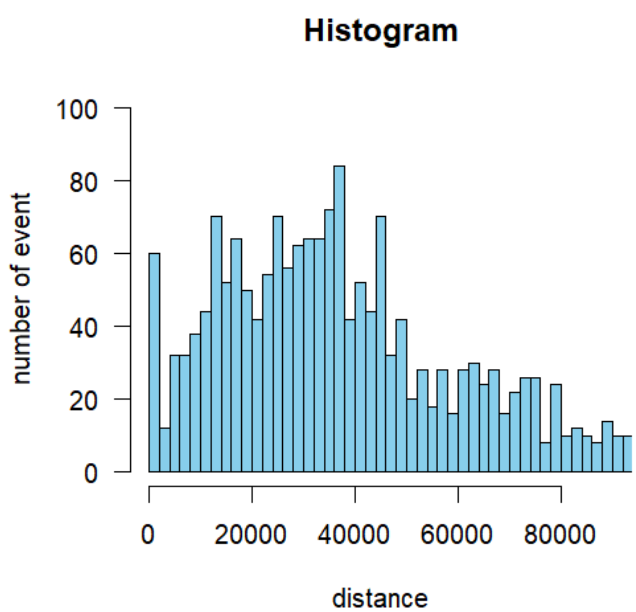

Supermarket & Population in Chicago
N
0 2 4 6 8 10 km
Log(Population) per Block
0 to 2
2 to 4
4 to 6
6 to 8
8 to 10


kd_police
900000
1000000
1100000
0
2000
4000
6000
8000
distance_police


historgram
number of event
0
10
20
30
40
50
0
20000
40000
60000
80000
distance



kd_hospital
600000
1200000
1800000
0
2000
4000
6000
8000
distance_hospital



Histogram
number of event
0
20
40
60
80
100
0
20000
40000
60000
80000
distance