Are Farmer's Markets in Philadelphia Completely Spatially Random?

Akira Di Sandro, Sofia Fasullo, Amy Solano

2024-12-16

Contents

Introduction

```
farmers_market <- st_read("HW 4/Philadelphia_Farmers_Markets201302.shp", quiet = T)

philly_zips <- st_read("HW 4/Philadelphia_ZipCodes.shp", quiet = T)

philly <- st_read("HW 4/Philadelphia.shp", quiet = T)

# load hydro data for philly
# source: https://opendataphilly.org/datasets/hydrology/
hydro <- st_read("https://services.arcgis.com/fLeGjb7u4uXqeF9q/arcgis/rest/services/Hydrographic_Feature
    st_transform(crs = st_crs(farmers_market))
```

Access to healthy, locally grown food remains a challenge for many American cities, including Philadelphia. To address this, the Philadelphia Food Trust has established numerous farmers markets across the city,

providing benefits such as fresher, seasonal, and healthier foods; diverse offerings like organic produce and heritage meats; and opportunities for community interaction and outdoor activity. However, large parts of South, North, and all of Northeast Philadelphia lack farmers markets, leaving residents without these advantages. In this study, we analyze the spatial distribution of farmers markets in Philadelphia to determine whether they are randomly placed, dispersed, or clustered.

```
# create map
ggplot() +

  # philly data
  geom_sf(data = philly, fill = "transparent", color = "lightgrey", lwd = 2) +

  # philly_zips data
  geom_sf(data = philly_zips, fill = "darkgrey", color = "lightgrey", lwd = 0.5) +

  # farmers market data
  geom_sf(data = farmers_market) +

  # hydro data
  geom_sf(data = hydro, fill = "#96dbe3", color = "transparent") +

  # map limits
  coord_sf(xlim = c(2660000, 2749276),
           ylim = c(208915, 310000)) +

  labs(title = "Philadelphia Farmers Market Locations") +
  theme_void() +
  theme(plot.title = element_text(hjust = 0.5),
        panel.background = element_rect(fill = "#555555"))
```

Figure 1 shows the distribution of farmers markets in Philadelphia. It is evident from this map that there are pockets of Northeast, South, and North Philadelphia that do not have easy access to farmers markets.

Methods

Complete Spatial Randomness (CSR)

Complete spatial randomness, or CSR, occurs when a set of points meets two crucial conditions. The first is that the probability that the point is in a spatially bounded area, or "cell" is directly proportional to how big that cell is. The latter is that the placing of one point has no effect on the placing of another point.

If neither conditions are met, the points could be clustered rather than completely spatially random (CSR). This means that if you drew two small cells of the same size, one may contain a great number of close points, and the other very few scattered points. If the points were CSR, then two cells of the same size would have the same probability for having points in them, and should have a similar number of points inside. This would also violate the second condition because the placement of one point implies there will be other points clustered nearby, which does not show independence.

If the second condition is not met, the points are likely dispersed rather than CSR. Dispersion is a spatial pattern where points are almost all equally spaced from each other, leading to an organized appearance. With dispersed points, the placement of one point means the next point is likely a set distance away, which violates the CSR condition of independence of point placements. With CSR points, given a point placement, the next point could be close or far, yet the zoomed-out effect is not of clustering.

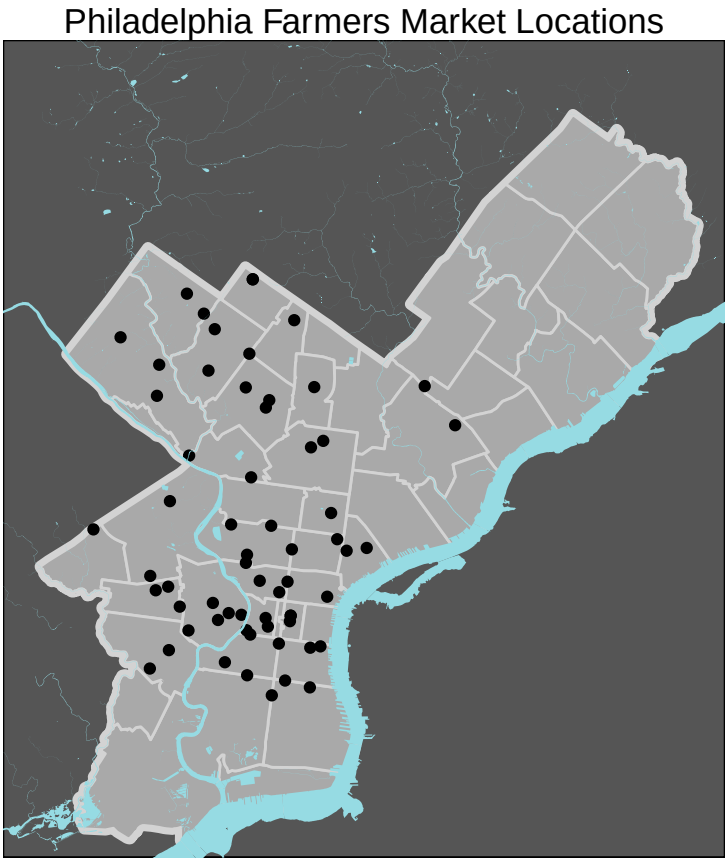

Figure 1: Fig 1. Philadelphia Farmers Markets

Null and Alternative Hypotheses

The null hypothesis in point pattern analyses is that the points are completely spatially random (CSR), and the alternate hypothesis is that the points are not CSR, whether they be clustered or dispersed. In the context of this study, this means that the locations of farmers markets in Philadelphia will have a spatial pattern that is not statistically significantly different from a random pattern under the null hypothesis, meaning that they are equally likely to occur across the city and the placement of one does not affect the placement of another. The alternate hypothesis is farmers markets have some non-random spatial pattern in Philadelphia (clustered or dispersed).

Quadrat Method

One example of point pattern analysis is the quadrat method. The Quadrat method is completed by dividing a spatial area by straight gridded lines of uniform distance, thus slicing the space into squares with equal area. We take the total sum of points for each cell and use the mean and variance of sum as well as VMR (variance-mean ratio, aka variance divided by mean). If the mean and variance are nearly equal (VMR of about 1), we fail to reject the null hypothesis that the points are CSR. If the variance is close to zero, then we reject the null hypothesis in favor of the alternative. The points are considered to be dispersed, because nearly every cell has the same number of points. If the variance is much larger than the mean, then we reject the null hypothesis and the points are considered to be clustered, because high variance indicates a large number of points in some cells and small number of points in others.

Limitations of Quadrat Method

The Quadrat method has significant limitations. For one, the size of the cells used to divide the area can change the results of the test. The larger the cell, the more likely it is to summarize a similar total number of points per cell and overlook more fine-grain variation. Meanwhile, smaller cell sizes are more likely to capture gaps between points and lead to a high number of cells without points and thus a higher variance between cells with points. Thus, it is difficult to understand the optimal cell size in order to understand tests if points are CSR, because tests with smaller cells are more likely to report clustering and tests with bigger cells are more likely to report dispersion.

Another limitation of the Quadrat method is that the sum of points in a cell is an aggregation which does not detail the spatial distribution of the points in that cell. For example, two adjacent cells that each contain one point could have those points right next to each other or far away from each other but would display the same sum value for the Quadrat analysis. This can exclude important data to understand whether there may be CSR in a set of points or not.

Nearest Neighbor Analysis (NNA)

Rather than comparing points within an arbitrarily determined grid cell, Nearest Neighbor analysis compares every point to its closest nearby point, no matter how close or far that point is.

The Nearest Neighbor analysis method first records n-many distances, one measure of the distance for each i^{th} point and its closest neighboring point. The average of these distances is referred to as the **observed average distance** ($\bar{D}_O$). This can be calculated by the formula below:

$$\bar{D}_O = \frac{\sum_n^{i=1} D_i}{n}$$

Where n is the number of points, or observations, and D_i is the distance between the i^{th} point and its nearest neighbor.

In order to put the observed average distance value into context, this number is compared to the average difference between a set of points that were distributed randomly without spatial pattern. This is called the **expected average distance** ($\bar{D}_E$). Mathematically, this number can be calculated with the following formula:

$$\bar{D}_E = \frac{0.5}{\sqrt{\frac{n}{A}}}$$

Where n is the number of points, or observations, and A is the total area of the space the points are in.

Together, these two statistics are used to calculate the **nearest neighbor index (NNI)**, the statistic used in nearest neighbor analysis. The formula for NNI is as follows:

$$\text{NNI} = \frac{\bar{D}_O}{\bar{D}_E}$$

The NNI value provides a simple way to compare a set of points' spatial pattern in comparison to a random spatial pattern. An NNI value of 1 indicates spatial randomness. An NNI value of 0 is a perfect single clustering of all points at 1 point, so NNI values close to 0 indicate clustering. An NNI value close to 2 indicates dispersion, with the maximum possible dispersion being a perfect hexagonal pattern of points that yield an NNI value of 2.149.

We test the significance of this statistic using a z-test, which is very similar to a T-test but uses a standard normal (z) distribution rather than a T distribution. The following statistic comprised of observed and expected average distances follows a standard normal distribution:

$$z = \frac{\bar{D}_O - \bar{D}_E}{\text{SE}_{\bar{D}_O}}$$

Where $\text{SE}_{\bar{D}_O}$ is the standard error among all n observed distances between each point and its nearest neighbor.

The standard error of all observed distances, $\text{SE}_{\bar{D}_O}$, is calculated using the following simplified formula:

$$\text{SE}_{\bar{D}_O} = \frac{0.26136}{\sqrt{\frac{n^2}{A}}}$$

where n is the number of observations and A is the area of the study region. We use the standard normal table to get a p-value from our calculated z statistic and reject the null hypothesis when $p < 0.05$.

Therefore, using this statistic we can calculate the probability of obtaining the observed value of this statistic from our sample under the null hypothesis. In this case, our null hypothesis, H_0, is that the sample has no spatial pattern and is random, and our alternate hypothesis, H_a, is that there *is* a significant non-random spatial pattern, either clustering or dispersion. Specifically, when $z > 1.96$, we reject the null hypothesis in favor of the alternative hypothesis, saying we have significant dispersion (since the average observed distance is greater than the average expected distance). On the other hand, when $z <$ -1.96, we reject the null hypothesis in favor of the alternative hypothesis, saying we have significant clustering (since the average observed distance is less than the average expected distance).

Limitations of NNA

While nearest neighbor analysis is a more thorough method of investigating spatial pattern among points than the Quadrat Method, it still has a number of limitations. For one, the method only considers the *first* nearest neighbor. A point in a cluster of two points vs a point in a cluster of 100 points is intuitively very different, but may yield the same distance between its first neighbor. For this reason, nearest neighbor analysis misses some nuance in spatial pattern.

Another large setback of nearest neighbor analysis is its heavy dependence on the area that contains the points. The smallest possible area that contains all points in a study would yield the smallest possible $\bar{D}_E$,

or average expected distance, and any increase in area is directly related to the resulting calculation, yet $\bar{D}_O$, or average observed distance is not affected. In addition, different shapes used to bound the points may have different areas. This greatly influences the conclusions drawn from such values as NNI and the z-distributed statistics. In softwares, such as ArcGIS Pro developed by ESRI, the boundary cannot even be determined but is automatically set to the smallest possible rectangle. This makes it cumbersome to analyze a set of points that do not cover an entire space, because the software will reduce that space to a smaller one, essentially changing your study area.

Say you wished to analyze whether hospitals in Philadelphia exhibit a significant spatial pattern. The City of Philadelphia follows an irregular, non-rectangular shape, and the hospitals are located primarily in the center of the city. Doing this calculation with software such as ArcGIS Pro, which automatically uses the smallest bounding rectangle, you are not actually analyzing the presence of spatial patterns with accuracy to the scale of the entire city. To do this, you would have to specify the shape, but even that is difficult, so you would have to create a shape of equal area to the city of Philadelphia which contains the points. Overall, this method is clunky and not ideal for efficient analysis.

NNA also does not take into consideration the spatial patterns present at different scales. The next method we describes tackles this problem better.

K-Function Analysis

K-function analysis outperforms both Quadrat Analysis and Nearest Neighbor Analysis. It avoids dependence on quadrat size, allows analysis at varying distances or scales, accounts for population density, and accommodates irregularly shaped study areas without assuming a rectangular layout.

K-Function

K-functions are a set of iterative processes that follow the following steps:

1. Calculate the overall point density for the entire study area. This would be $\frac{n}{A}$, or n many points divided by study area A.
2. Draw n-many circles, each with radius d around each point.
3. Sum the number of points or events, in each circle minus one. This is the total number of other events in that circle, excluding the original event at the center of the circle.
4. Average all total other events.
5. The K-function at this set distance, d equals the average other events at radius d divided by the overall point density.

$$K(d) = \frac{\frac{1}{n}\sum_{i=1}^{n} S_i}{\frac{n}{A}}$$

Where K(d) is the K-function at distance d, n is the number of total points or events, S_i is the number of total *other* points or events at the circle around point i, each circle having radius d, and A is the total area of the study region.

The Kest() funciton in the spatstat.explore package we use to estimate the K-function in R, the K-function is defined as the following:

$$\hat{K}(r) = \frac{a}{n(n-1)} \sum_i \sum_j I(d_{ij} \leq r) e_{ij}$$

where a is the area of the window (study region), n is the number of observations, $I(d_{ij} \leq r)$ is the indicator that equals 1 when the distance between any two points i and j is less than or equal to r, the radius of the circle, and e_{ij} is the edge correction weight which depends on the edge correction type we specify. In our

analyses, we specified the edge correction to be "best" which selects the best edge correction available for the geometry of our study region.

We then repeat the K-function process (steps 1-5) for any number of distance values, d.

K-functions are incredibly useful because they provide detailed information that can capture the nuance of spatial point patterns at different scales. For example, a dataset of bees on a honey farm would show bees clustered at each hive (when zoomed into one or a few hives), but zooming out would show the hives dispersed among their rows and columns on the farm. The iteration of K-functions over a variety of distances collects this change in spatial pattern interpretation.

At each scale d:

- $K(d) = \pi d^2$ when there exists complete spatial randomness (CSR);
- $K(d) > \pi d^2$ when there exists clustering;
- and $K(d) < \pi d^2$ when there exists dispersion.

L-functions

$L(d)$ functions are manipulated K-functions at distance d that provide a more easily interpretable answer by simplifying the output of $K(d)$.

$$L(d) = \sqrt{\frac{K(d)}{\pi}} - d$$

At scale d:

- $L(d) = 0$ when there exists complete spatial randomness (CSR);
- $L(d) > 0$ when there exists clustering;
- and $L(d) < 0$ when there exists dispersion.

It should be noted that in the ArcGIS Pro software, $L(d)$ has a different formula: $L(d) = \sqrt{\frac{K(d)}{\pi}}$, so CSR is at d, not 0. We are not using ArcGIS for our analysis and will use the first formula to calculate $L(d)$.

We use the Lest() function from the spatstat.explore package in R where the L-function is defined as follows:

$$L(r) = \sqrt{\frac{K(r)}{\pi}}$$

where $K(r)$ is the K-function and r is radius of the circle.

Beginning and Incremental Distances

When performing K-function and L-function analysis, researchers need to identify the range of distances that will be evaluated through the functions. Often the functions are iterated over 10-20 values of d, and researchers can pick the beginning value for d and set the increments at which it should be increased, and/or designate the maximum value.

Usually, the maximum distance is calculated by dividing the maximum distance between two points in the dataset by 2 (max distance = half of the distance between the two farthest points). In R, the spatstat.explore manual tells us that there is a "sensible default", so it suffices to specify the rmax value (maximum distance).

Significance Test and Envelopes

We can conduct a significance test for a set distance d, with the null hypothesis, H_0, being that CSR exists so $L(d) = 0$, and the alternate hypothesis, H_a, being that there is some spatial pattern so $L(d) \neq 0$. The alternate hypothesis is then further broken down into two subcategories, with H_{a1} being that there is clustering at d (aka when $L(d) > 0$) and H_{a2} being that there is uniformity, or dispersion at d (aka when $L(d) < 0$).

There is no statistic with a particular distribution used for this hypothesis test. Rather, some amount of (i.e. 99) random rearrangements of the data (which always contains n points and the same area A) are made. Then each of these data rearrangements will have $L(d)$ evaluated. These values of $L(d)$ can be used as a **"confidence envelope"** to test the hypothesis. The highest value among all those produced is referred to as the *Upper Envelope* value, denoted $L^+(d)$, and the lowest value produced is the *Lower Envelope* value, denoted $L^-(d)$. Our observed $L(d)$ value at d with the original spatial configuration of the data is denoted $L^{obs}(d)$.

We would fail to reject the null hypothesis at d if $L^{obs}(d)$ fell between the upper and lower envelopes, and we would reject the null hypothesis at d if $L^{obs}(d)$ fell outside envelope. Specifically, we would reject H_0 for H_{a1} if $L^{obs}(d) > L^+(d)$ and we would reject H_0 for H_{a2} if $L^{obs}(d) < L^-(d)$. The number of rearrangements of n data points would roughly equate to the confidence level of the derived conclusions. For example, a test based on an envelope made with 9 rearrangements would result in a conclusion at a confidence level of 90%, and one with 99 rearrangements would correspond to 99% confidence, etc.

This hypothesis test would then be iterated at all values of d to assess whether significant spatial patterns, specifically clustering or dispersion, occurred at different scales of distance.

Edge Correction

It should be noted that points near the edge of a study area, specifically points whose distance to the edge is less than distance d, create an inconsistency in the interpretation. A circle with radius d around such a point would have part of its area outside the study area, and would have a lower chance of having other points, or events, within the circle. The closer a point is to the edge (once already closer than distance d), the greater this phenomenon. This undermines the legitimacy of the comparison between the circles around all other points.

The **Ripley's Edge Correction** accounts for this inconsistency among circles in K-functions by weighing each circle centered at the i^{th} point of a dataset with n points by the percentage of its area that falls within the study area. In other words, all points that are greater than d distance from the edge of the study area are weighted by a factor of 1 when calculating $K(d)$, because the circle around them falls completely within the study area. But a point that lies on the edge of a study area and subsequently was the center of a circle with only half its area in the study area would be weighted by a factor of 0.5. Ripley's Edge Correction accounts for inconsistencies between circles at the edge of a study area and circles further from the edge so that they can be compared, however, if used in ArcGIS Pro, it has the limitation of only working for rectangular study areas. Otherwise, the only limitation of the edge correction is that we do not know which points would have existed beyond the boundary, and it is essentially an inference with some degree of uncertainty.

The **Simulate Outer Values Boundary Correction** offers a different method for edge correction. Rather than deduct from the weight of circles that lie partly outside the study area during K-function calculations, this method extrapolates new data points. It essentially mirrors the points within a circle to the area of the circle outside the study area in attempts to infer what the data would have been there. Like Ripley's Edge Correction, this method provides a more interpretable result of K-functions, infers data without perfect certainty, and in the end, tends to yield a similar result.

In our study we used both Ripley's Edge Correction and the "best" method to correct for K-function circles that lay partly outside the study area.

Inhomogeneous K-functions

Just because a variable exhibits a significant spatial pattern in the study area does not mean it is intuitively significant. For example, one may see that hospitals are clustered in a certain area of the city or state, and seek to draw conclusions that this displays inequity in health access, only to see that hospitals are clustered where population is clustered. This is an example of when **inhomogeneous K-functions** are appropriate to use. Inhomogeneous K-functions are K-functions that compare the study variable to another variable that has some significant spatial pattern when performing the K-function analysis.

This can be done by creating a "probability map" of sorts encoding the probability that a point would fall into an area through a continuous raster, using the comparison variable (i.e. population). Normally, when computing the permutations to create the confidence envelope to test a hypothesis based on observed K-function value, these random rearrangements of n points assume that there is equal probability that any point should fall in any area. When conducting a inhomogeneous K-functions test, the observed K-functions value is compared to an envelope comprised of a set of permutations that *did* take into account existing spatial patterns, by inputting more points randomly where there was higher probability of points existing due to the comparison variable.

Results

```
# get coords for farmers markets
FM_pts <- data.frame(farmers_market %>%
  st_coordinates())

# got a warning about duplicated points, and figured out that row 25 and 41 are duplicates
# checked the farmer_market data and they are in fact both at RTM
# removing the second point to have all unique points
FM_pts_nodup <- FM_pts[-41,]

# boundary
philly_bound <- as.owin(philly)

# define ppp object
FM_ppp <- ppp(FM_pts_nodup$X, FM_pts_nodup$Y, window = philly_bound)
```

Nearest Neighbors Analysis (NNA)

```
# Computes the distance from each point to its nearest neighbor in a point pattern.
nnd <- nndist.ppp(FM_ppp)

# calculate Mean Observed Distance, Mean Expected Distance and SE.
MeanObsDist <- mean(nnd)
MeanExpDist <- 0.5 / sqrt(nrow(farmers_market) / area.owin(philly_bound))
SE <- 0.26136 / sqrt(nrow(farmers_market)^2 / area.owin(philly_bound))

# calculate z-statistic
zscore <- (MeanObsDist - MeanExpDist)/SE                    # z = -3.106
pval <- ifelse(zscore > 0, 1 - pnorm(zscore), pnorm(zscore)) # p = 0.0009
```

```r
# calculating the NNI
NNI <- MeanObsDist / MeanExpDist     # NNI = 0.794
```

The Nearest Neighbor Analysis results in a nearest neighbor index (NNI) of 0.794 with a z-score of -3.106 and $p < 0.001$ ($p = 0.0009$). Though the NNI is closer to 1 than it is to zero, our z-score is less than -1.96, which means the average observed distance is smaller than the average expected distance, meaning we see significant clustering in the distribution of farmers markets across Philadelphia. We can reject the null hypothesis in favor for the alternative hypothesis stating that the observed point pattern is not random, and rather clustered.

We also conducted the NNA with the nni() function in the spatialEco package. This method gave us a higher NNI of 1.136 with a z-score of 2.028 and a p-value of 0.043. In this case, NNI > 1 and z-score > 1.96, so we would reject the null hypothesis to favor the alternative hypothesis that we observe statistically significant dispersion.

We expect that this inconsistency between the two methods comes from the fact that our latter method does not accept a boundary argument and assumes a convex hull for study area. As a result, this method used an expected mean distance of 2,797 instead of 4,001 that we calculated for the former method. This is a result of one of the limitations of NNA that we discussed in the Methods section. Since the question we are trying to answer with this question is whether farmers markets are clustered in Philadelphia, our study area boundary is central to that question. The former method takes into consideration study area, so we focus on the results of that method.

```r
# prof did this with the base plot functions
plot(density(FM_ppp),
     main = "Kernel Density Plot of Farmers Markets in Philadelphia")
contour(density(FM_ppp), add = T)
plot(FM_ppp, add = T)
```

Figure 2 shows a kernel density plot of Philadelphia farmers markets which, again, clearly highlight the lack of farmers markets in Northeast and South Philadelphia. It also shows that there is a cluster of farmers markets centered around Center City and another potential cluster around Northwest Philadelphia.

K-Function Analysis

In our K-function analysis, we used a max distance of 28,500 ft since the two points that were furthest apart from each other were about 56,698 ft apart (rounded to 57,000 before dividing by 2). As stated in our methods section, we used the "best" method as the edge correction with 99 simulations (for a 99% confidence envelope).

```r
# plot Ripley's K-function with 90% simulation envelopes
plot(Kenv,
     xlab = "r", ylab = "Khat(r)",
     cex.lab = 1.6, cex.axis = 1.5,
     main = "Ripley's K-Function with Confidence Envelopes",
     cex.main = 1.5, lwd = 2)
```

As shown in figure 3 above, the observed K-function is consistently above the expected K-function and the 99% envelopes. More specifically, until a distance of about 3,000 ft, we observe a random spatial pattern, but for distances greater than 3,000 ft, the observed $\hat{K}$ is greater than the theoretical K, meaning we observe clustering at these scales.

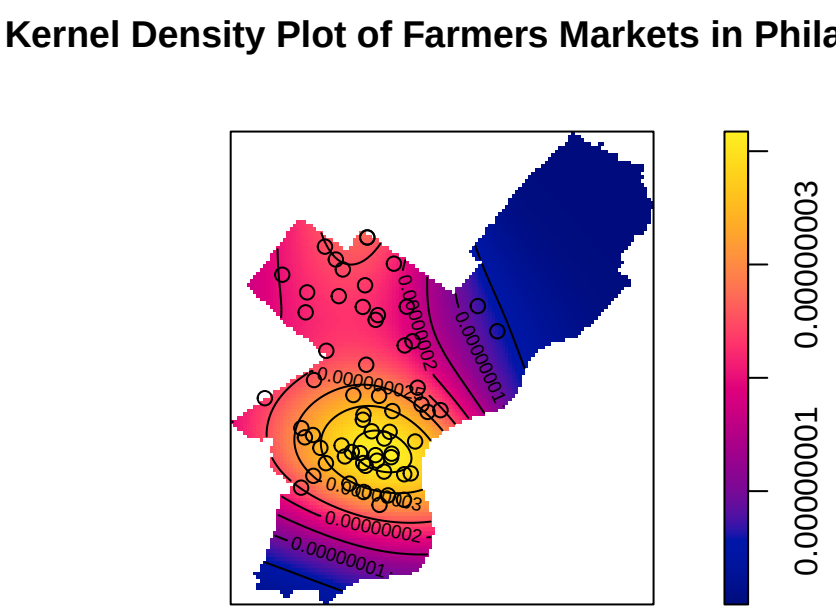

Figure 2: Fig 2. Kernel Density Plot of Philadelphia Farmers Markets

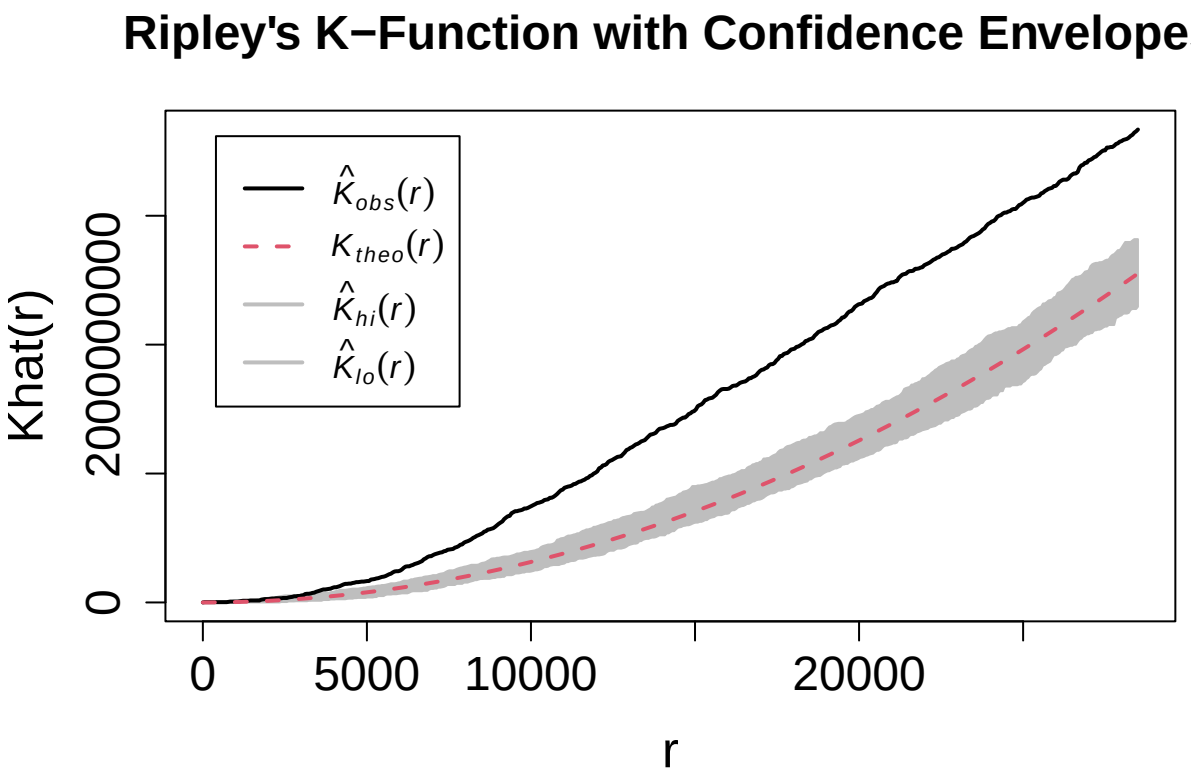

Figure 3: Fig 3. Ripley's K-Function with Confidence Envelopes

```r
L2 <- Lenv
L2$obs <- L2$obs-L2$r
L2$theo <- L2$theo-L2$r
L2$lo <- L2$lo-L2$r
L2$hi <- L2$hi-L2$r

# look at where obs > hi, obs < lo, and where lo < obs < hi
L_cutoff <- data.frame(L2) %>%
    mutate(spatial = case_when(obs > hi ~ "clustering",
                               obs < lo ~ "dispersed",
                               .default = "CSR"))

plot(L2,
     xlab = "r", ylab = "Lhat(r)",
     cex.lab = 1.6, cex.axis = 1.5,
     main = "Ripley's L-function with Confidence Envelopes",
     cex.main = 1.5, lwd = 2)
```

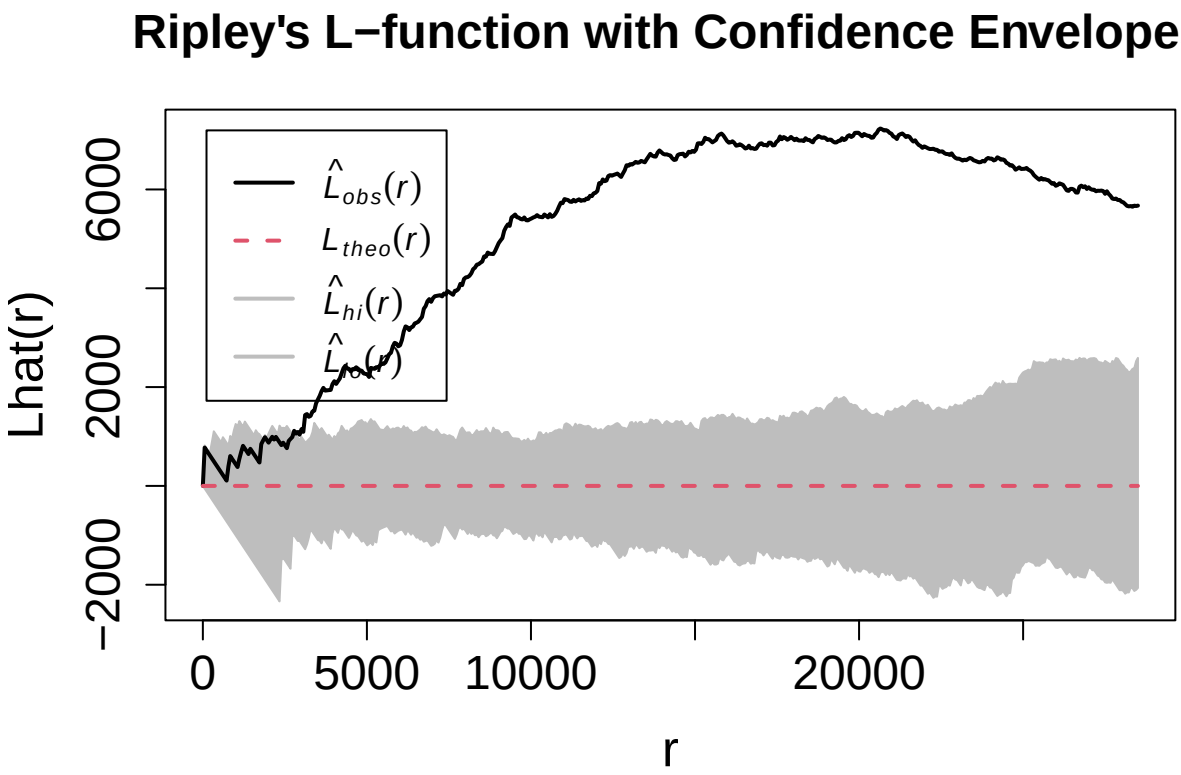

Figure 4: Fig 4. Ripley's L-Function with Confidence Envelopes

As shown in figure 4 above, the L-function plot shows us the same results as the K-function plot with the observed $\hat{L}(r)$ being within the 99% confidence envelope until $r >$ 2,700 ft at which $\hat{L}(r) > L^{+}(r)$, meaning there is statistically significant clustering at those larger scales. Note that there are also some distances $<$ 2,700 ft at which $\hat{L}(r) > L^{+}(r)$, but for the most part, $\hat{L}(r)$ is the 99% confidence envelope at this range of r

According to these results, we fail to reject the null hypothesis at distances less than or equal to 3,000 ft, meaning that at this smaller scale, we see spatial randomness. At a mid to larger scale (distance $>$ 3,000 ft), we reject the null hypothesis in favor of the alternative hypothesis (specifically, H_{a1}) and conclude that there is statistically significant spatial clustering.

```
# create map
ggplot() +

  # philly data
  geom_sf(data = philly, fill = "transparent", color = "lightgrey", lwd = 2) +

  # philly_zips data
  geom_sf(data = philly_zips, aes(fill = Pop2000), color = "lightgrey", lwd = 0.5) +

  # farmers market data
  geom_sf(data = farmers_market) +

  # hydro data
  geom_sf(data = hydro, fill = "#96dbe3", color = "transparent") +

  # map limits
  coord_sf(xlim = c(2660000, 2749276),
           ylim = c(208915, 310000)) +

  # change themes
  scale_fill_gradient(low = "#2C5223FF",
                      high = "#CEC917FF",
                      breaks = c(0, 14500, 29000, 43500, 58000, 72500),
                      limits = c(0, 72500),
                      labels = c("0", "14,500", "29,000", "43,500", "58,000", "72,500")) +

  labs(title = "Philadelphia Farmers Market Locations and Population by ZIP code",
       fill = "Pop. in 2000") +
  theme_void() +
  theme(plot.title = element_text(hjust = 0.5),
        panel.background = element_rect(fill = "#555555"),
        legend.position = c(.95, .05),
        legend.justification = c("right", "bottom"),
        legend.margin = margin(3, 3, 3, 3))
```

As discussed in the methods section, when performing K-function analysis, we should also keep in mind other variables that may influence spatial patterns of points of interest. In our case, it may be helpful to consider whether population has an influence on farmers market locations – are farmers markets more frequently places in ZIP codes with higher populations? Figure 5 (above) helps us look at this question without performing any additional analyses. Although there are some ZIP codes in South and Northeast Philadelphia where the population is very low, there are many ZIP codes with populations of around 25,000 to 58,000 that don't have any farmers markets in them. There are also areas like Northwest Philadelphia (where we saw a possible cluster of farmers markets in figure 2) with relatively low population. Therefore, it seems as though taking in population at the ZIP code level would not drastically change our results.

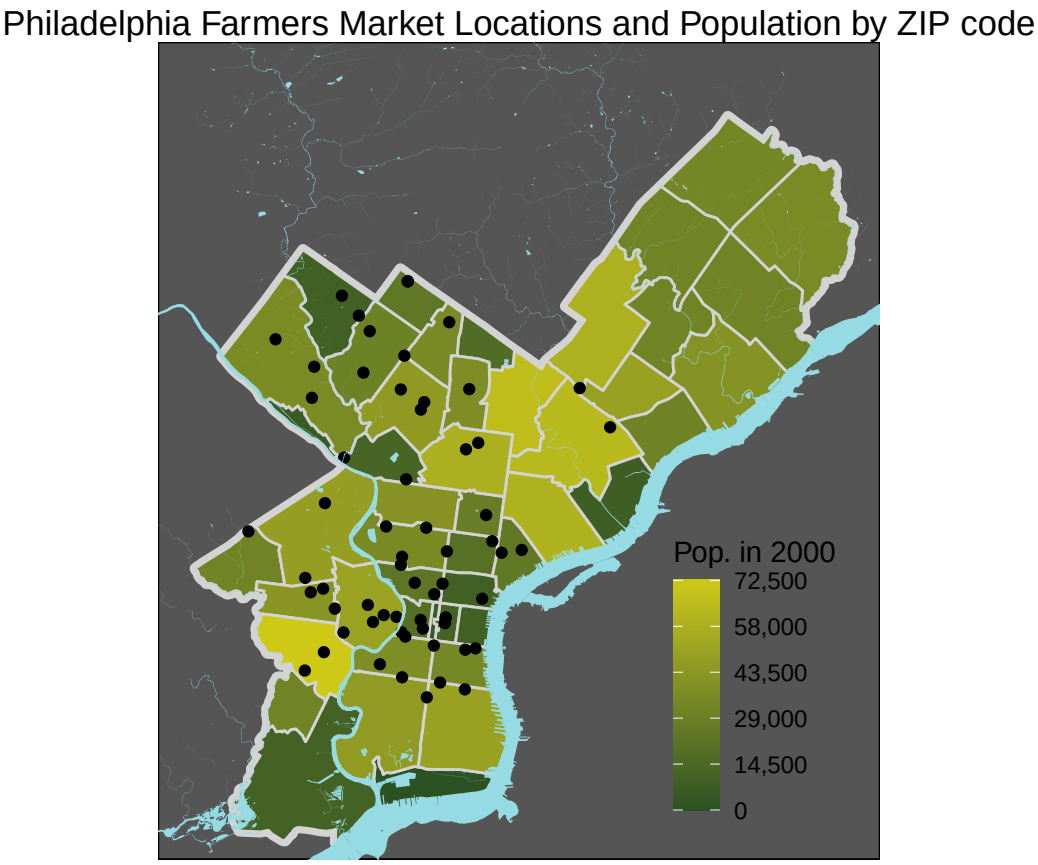

Figure 5: Fig 5. Philadelphia Farmers Markets and Population by ZIP Code

Discussion

In general, both the Nearest Neighbor Analysis and K-Function Analysis lead us to the conclusion that there is some level of clustering in Philadelphia's Farmers Market locations. K-Function Analysis goes a step further than NNA and tells us that we specifically see clustering when zoomed out at a larger scale.

These conclusions are consistent with what we can assume from taking a glance at the map of Philadelphia's farmers markets. From looking at something like figure 1, we can see that it looks like farmers markets are clustered around center city and that there are some clustered in the Northwestern part of Philadelphia as well. If we only zoom in on the convex hull of the observations of farmers markets, we can see that the distributions of markets look relatively random. This corresponds to the results of the K-function analysis at a smaller scale (distance $< 3{,}000$ ft) and of the spatialEco package nni() function that does not take into consideration study area.

```
# create map
ggplot() +

  # philly data
  geom_sf(data = philly, fill = "transparent", color = "lightgrey", lwd = 2) +

  # philly_zips data
  geom_sf(data = philly_zips, aes(fill = MedIncome), color = "lightgrey", lwd = 0.5) +

  # farmers market data
  geom_sf(data = farmers_market) +

  # hydro data
  geom_sf(data = hydro, fill = "#96dbe3", color = "transparent") +

  # map limits
  coord_sf(xlim = c(2660000, 2749276),
           ylim = c(208915, 310000)) +

  # change themes
  scale_fill_gradient(low = "#742C14FF",
                      high = "#F0D77BFF",
                      breaks = c(0, 15000, 30000, 45000, 60000),
                      limits = c(0,60000),
                      labels = c("$0", "$15k", "$30k", "$45k", "$60k")) +

  labs(title = "Philadelphia Farmers Market Locations and Median HH Income by ZIP code",
       fill = "Med. HH Inc.") +
  theme_void() +
  theme(plot.title = element_text(hjust = 0.5),
        panel.background = element_rect(fill = "#555555"),
        legend.position = c(.95, .05),
        legend.justification = c("right", "bottom"),
        legend.margin = margin(3, 3, 3, 3))
```

Figure 6 (above) shows a map of Philadelphia with Farmers Market locations overlaid on a map of median household income (from 2000) by ZIP code. One might expect to see a cluster of farmers markets in more wealthy neighborhoods, but looking at this map, we see that Northeast Philadelphia (which is relatively wealthy) does not have any farmers markets, while some ZIP codes around Center City filled in with a darker color (indicating lower median household income) have several farmers markets. In South Philadelphia, the

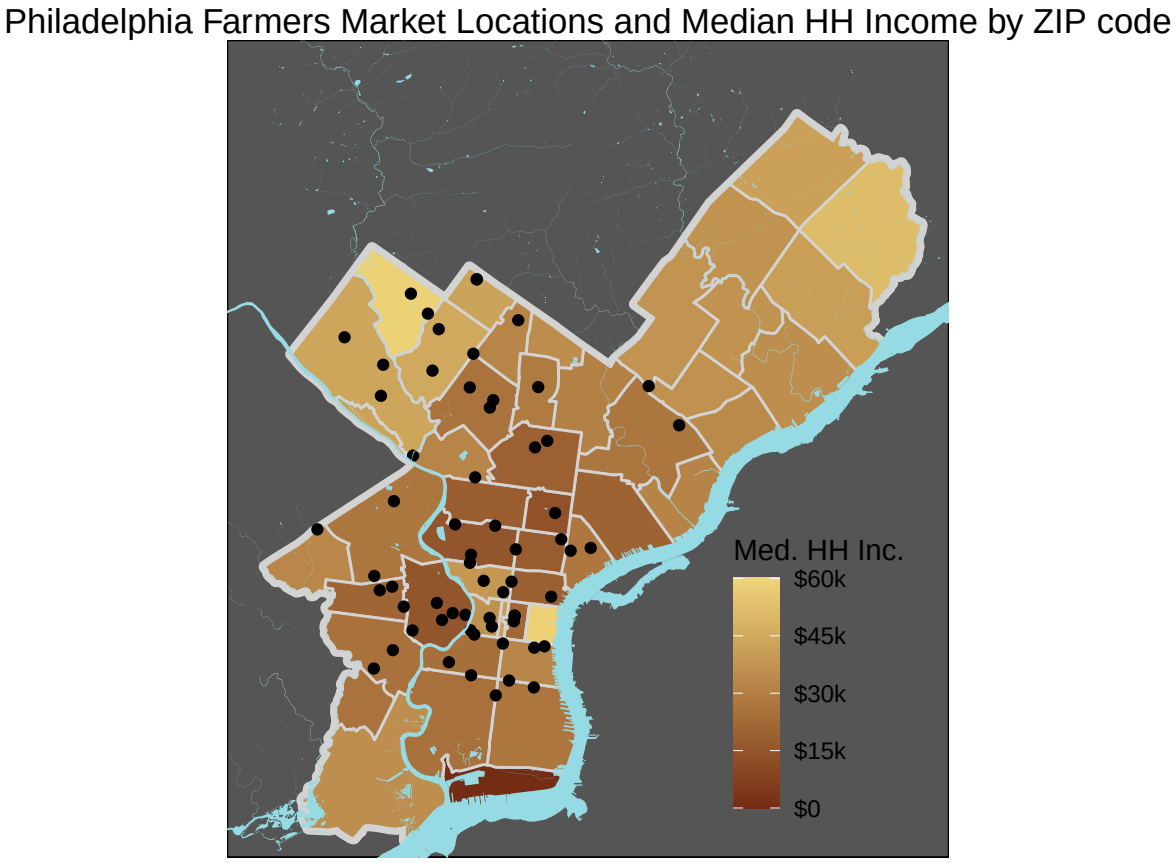

Figure 6: Fig 6. Philadelphia Farmers Markets and Median HH Income by ZIP Code

ZIP codes with no or few farmers markets seem to range from low to middle levels of median household income. In conclusion, it does not seem to be the case that median household income influences farmers market locations (though this may not be the case upon running actual statistical tests).

In conclusion, we can conclude that at a smaller scale, farmers markets are spatially random, but at larger scales, farmers markets seem to be clustered with a node around Center City and another node in Northwestern Philadelphia. These clusters do not seem to be influenced by Population nor Median household income (in the year 2000), but statistical tests need to be run to confirm this. Perhaps farmers markets are concentrated around more touristy and commonly frequented areas of Philadelphia.

This analysis helps provide evidence that there is a disparity in access to farmers markets in Philadelphia. As described in the introduction, farmers markets provide many benefits to its frequenters, not only limited to access to fresh fruits and vegetables, but also a community and often outdoor space to meet and interact with neighbors and a fun activity that helps local economy at the same time. With farmers markets being clustered, this means that there are areas, namely in South, North, and Northeast Philadelphia that are not afforded the same benefits that come with farmers markets. This analysis can provide evidence that the city needs to invest in more farmers markets in areas that currently don't have any.