Small Stories in Big Data: Gaining Insights From Large Spatial Point Pattern Datasets

Ate Poorthuis
Matthew Zook
University of Kentucky

Abstract

With the onset of big data, it is now relatively easy to gain access to a wide variety and great magnitude of data sources. Data, however, do not necessarily equate to useful insights and meaningful analysis. In this article, we outline a specific step-by-step approach to gaining insight into the spatial footprint of online, point-based data—in this particular case, data from the popular social media service Twitter.

Introduction

A key aspect of current research directions in urban studies is that researchers are inundated by both a flurry of "big" datasets and persistent writing about the importance of that data deluge. From cellphone records to open government datasets and online social media shared using application programming interfaces (APIs), the topic of big data—both in terms of possible applications and critique—is ever more present.

The relative ease with which a researcher can gain access to a variety of new data sources, however, does not necessarily mean that insights from those data can be achieved as easily. Putting aside key questions of what an indicator actually measures (an issue present in various forms across datasets), researchers are also confronted with the fact that many trusted analytical and mapping methods cannot be directly applied in standard ways to spatial *big* data. To partly alleviate this issue, we outline a step-by-step approach to gaining insight into the spatial footprint of online big data—in this particular case, the popular social media service Twitter. As with most geosocial online data (for example, Foursquare check-ins, geotagged Flickr photos), tweets are stored as spatial points with a longitude and latitude coordinate pair and a variety of other metadata (timestamp, text, images, activity records, etc.) that can be leveraged to gain useful insight on the spatial distribu-

tion of daily life (Poorthuis and Zook, 2014). It should be noted that the approach outlined in this article would also be applicable to large spatial point pattern datasets generated from other, more traditional, sources.

So Much Data… Now What?

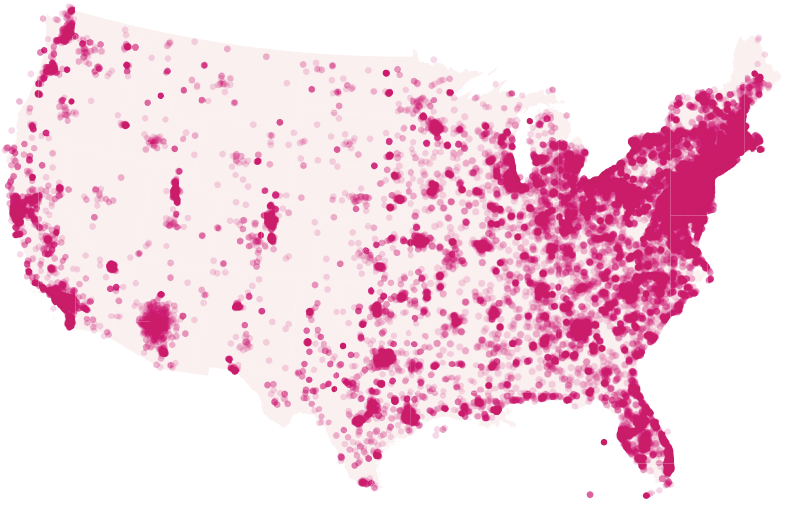
In principle, anyone with an interest in geosocial online data and some programming skills can access a wide range of APIs made available by almost every major social media platform. Although this article does not address the techniques for API access, tutorials and code are widely available. Moreover, those without the prerequisite technical skills can acquire ready-made datasets through third party vendors, such as Gnip. In stark contrast with the situation facing social science research for most of the 20th century, today we certainly do not suffer from a lack of data. For example, the Dolly project at the University of Kentucky has been collecting all geotagged tweets in the world since June 2012—totaling more than 9 billion data points, and counting. Even small subsets of such data, from people talking about receiving a flu shot to people tweeting about their favorite beer brands, yield many thousands of data points.

Instead, the pressing problem that presents itself is how to gain *meaningful* insights from such large collections of spatial points. Although research may take any number of approaches (Crampton et al., 2013), an early step is to simply map or visualize these data in ways that reveal the presence (or absence) of underlying spatial processes and distributions.

The easiest approach for visualizing the spatial distribution of data—in this case, tweets from Hurricane Sandy (Shelton et al., 2014)—is, quite literally, putting the points on the map (exhibit 1). This relatively straightforward one-to-one plotting of data points on a map presents two specific

Exhibit 1

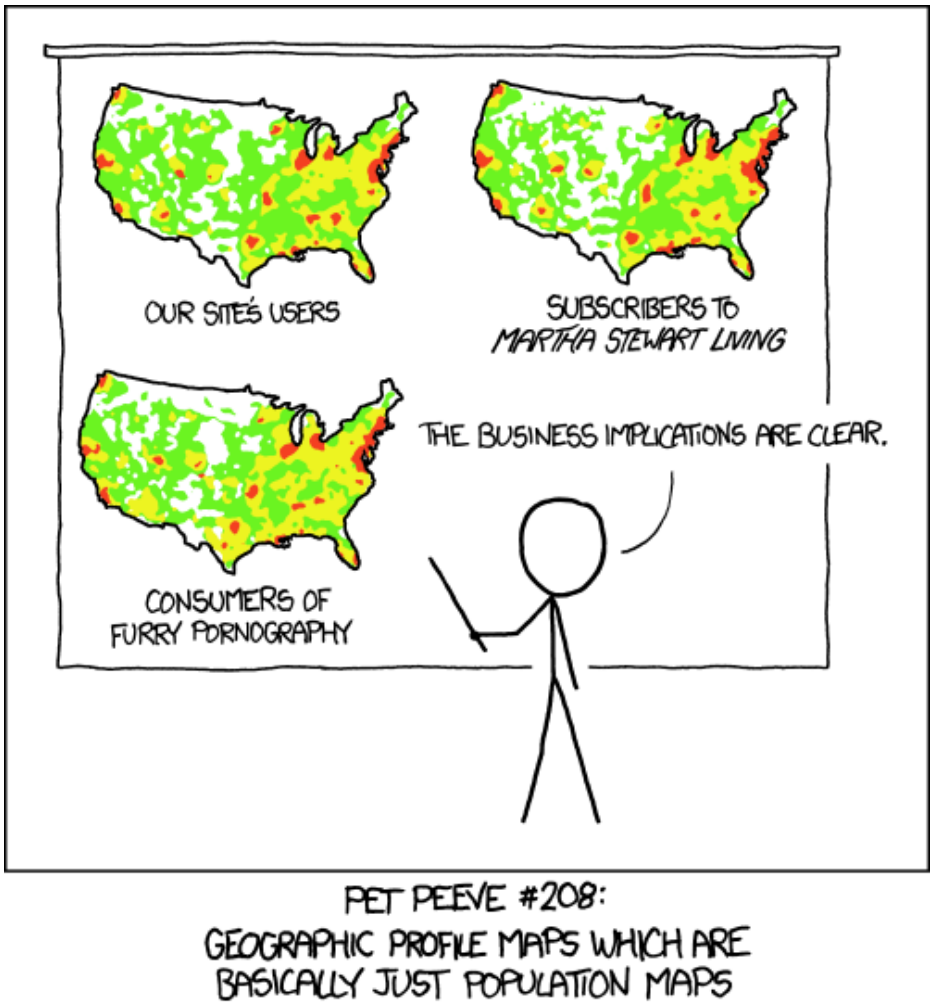
Overplotting—Too Many Points To Distinguish Clear Spatial Patterns



problems. First, such maps suffer from "overplotting": many overlapping points obscure each other and make it difficult to assess the total number of points in each area. Second, even if the problem of overplotting were solved, these patterns largely mimic population density: in the case of Twitter, more tweets are generally sent from densely populated areas because these locations simply have more Twitter users. This second problem is very much prevalent in maps of online phenomena and even reached modest Internet fame after Randall Munroe devoted a popular XKCD comic to it (exhibit 2). In the next sections, we walk through a step-by-step approach that first addresses the problem of overplotting by aggregating individual points to a hexagonal lattice. It subsequently provides a solution to correct for this population density "mirroring" by normalizing raw counts through the calculation of an odds ratio.

Exhibit 2

Population Density?



Source: Reprinted from http://xkcd.com/1138/

Fixing the Overplotting Problem

A range of common cartographic and geographic information science, or GIScience, approaches can solve the problem of overplotting. The first is to make each data point slightly transparent. This approach is fine with only a small number of overlapping points but, in the case of big geodata, we are often confronted with hundreds of points overlapping in one location, while other locations have only one or two points. Another approach would be to visually "explode" overlapping features, slightly offsetting their position to prevent overlap. Again, this approach works well with smaller datasets (John Snow's classic cholera map is a prime example of this approach [Snow, 1855]) but is not well suited for large datasets.

Another way to address the issue of overplotting is generating what is colloquially called a "heatmap." Techniques such as kernel density estimation or kriging are used to create a (smooth) density surface. A major caveat, however, is that these techniques interpolate or "smooth" values in between actual data points and thus assume that the underlying spatial processes are continuous. This caveat applies to many natural phenomena, such as temperature and precipitation, but is more problematic when applied to social phenomena. This caveat is especially the case on an urban scale in which stark differences in demographics, retailing, and so on, are often present between neighborhoods or even from block to block. Although heatmaps are visually pleasing (and hence popular), they are not necessarily the most appropriate technique for gaining meaningful insight from online social media data.

A more suitable approach is to aggregate individual points to larger areas or polygons. These areas could be administrative regions, such as census tracts or counties, or they could be arbitrary spatial areas, such as rectangles, circles, or hexagons. Unless the final goal of the analysis is to compare the point data under study with other datasets that are available only for certain administrative units, aggregating to a lattice of arbitrary areas (such as hexagons) has two specific advantages from an analytical perspective. First, administrative units often have varying sizes. For example, counties in the western part of the United States, in general, are much larger than their eastern counterparts. The larger counties not only have a higher chance of having more points inside their border, but they also stand out much more visually. Aggregating to a regular lattice of rectangles or hexagons, in which every area has the exact same size, solves this problem. Second, such a lattice enables us to address, although not solve per se, the Modifiable Areal Unit Problem (MAUP) by intentionally modifying the size of the rectangles or hexagons. This can be done to test whether the spatial patterns indeed change due to MAUP or simply to choose the "best" cell size based on the underlying phenomenon (see Wilson, 2013, for an example of the consequences of changing areal units).

Creating a Hexagonal Lattice

Hexagonal lattices have seen a recent surge in popularity within online mapping, but they are more than just the latest fad—they have a few distinct advantages over rectangular grids. First, in cartographical terms, rectangular cells are more distracting. The eye is drawn to the horizontal and vertical grid lines, making it more difficult for the reader of the map to distinguish spatial patterns

(Carr, Olsen, and White, 1992). Second, in analytical terms, hexagonal lattices have a higher representational accuracy than square or rectangular grids (Burt, 1980; Scott, 1988), meaning that they represent the underlying point pattern more closely. The hexagon is the highest sided regular polygon that can still be used to tessellate (that is, cover a surface without gaps or overlap), and, as such, it is closest to the ideal of a circle. The closer a polygon is to a circle, the closer its border points are to its center, which partly explains the higher representational accuracy.

As such, the first practical step in generating a hexagonal lattice is to aggregate up from the original point pattern. This aggregation can be done quite easily in Arcmap[1] and QGIS[2] or using R (R Core Team, 2014). Given the power of R and its relative newness to geospatial analysis, we include code snippets used for generating the maps in this article. More extensive (and commented) code with some sample data is available at https://github.com/atepoorthuis/smallstoriesbigdata.

```
library(sp)
tweets <- read.csv("tweets.csv", colClasses=c("latitude"="numeric",
    "longitude"="numeric"))
coordinates(tweets) <- c("longitude", "latitude")
proj4string(tweets) <- "+proj=longlat +ellps=WGS84 +datum=WGS84 +no_defs"
hex <- HexPoints2SpatialPolygons(spsample(tweets, n=3000, "hexagonal"))
```

At this point we read a dataset of tweets from a .csv file, point to the longitude and latitude columns for the spatial coordinates, set a projection, and then generate a hexagonal grid over the same spatial extent. A key variable in this code is the number of cells in the lattice (3,000 in this example), but this number can be readily changed to explore how changes in cell size affect the resulting visualization. After we have produced a hexagonal lattice, we can then simply spatially join each individual tweet to a grid cell.

```
tweets$hex <- over(tweets, hex)$id
```

We take this intermediate step of adding the identifier of the corresponding grid cell to each individual tweet, because it also allows for the flexibility of sampling down power users (Poorthuis and Zook, 2014). Within most online social media, a power law, or close approximation, can be found in which a few users contribute by far the most content (Clauset, Shalizi, and Newman, 2009). If we wanted to correct for that effect, we could, for example, randomly sample down active users to a maximum of 5 data points (or some other selected value) per grid cell.

```
library(data.table)
tweets.dt <- data.table(tweets)
tweets.dt[,sample(.I, if(.N > 5) 5 else .N), by=list(u_id, hex)]
```

After that, it is just a matter of counting the number of tweets per grid cell and visualizing the results.

```
tweets.dt[,tweets=.N, by=list(hex)]
```

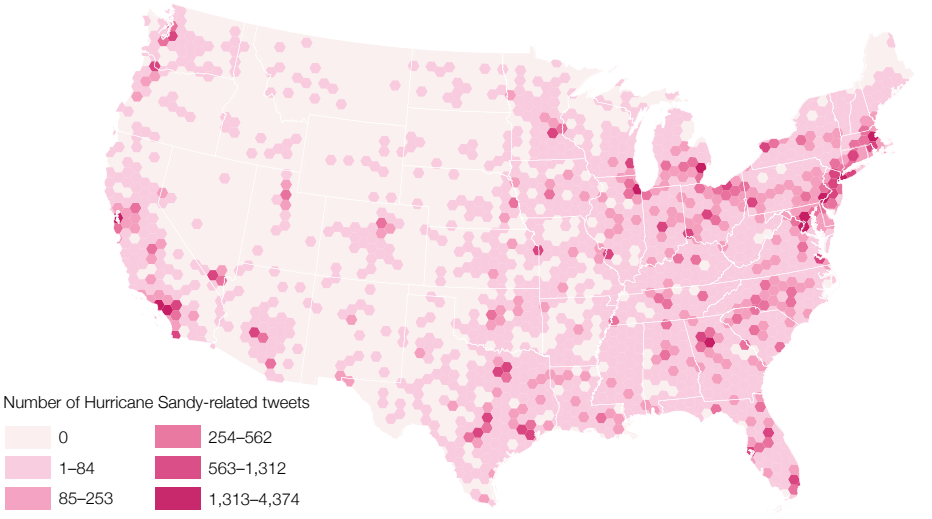
[1] http://www.arcgis.com/home/item.html?id=03388990d3274160afe240ac54763e57.

[2] http://michaelminn.com/linux/mmqgis/.

An example of this step in the process is provided in exhibit 3, which shows the spatial pattern of tweets related to Hurricane Sandy sent in October 2012 (Shelton et al., 2014).

Exhibit 3

Aggregation to Hexagons



Normalizing the Cells Using an Odds Ratio

Although aggregating to hexagons solves the problem of overplotting, to a large extent, the resulting spatial pattern still follows very closely the distribution of population. Given that this dataset is derived from social media, this problem is to be expected. We are, after all, still looking at the raw count of the number of tweets, which is heavily influenced by how many people happen to live in each hexagon.

A fortunate side effect of the aggregation to polygons is that it becomes much easier to normalize each raw count. For conventional data, we would likely choose to normalize a phenomenon by simply dividing raw counts by the total population or, for example, the area of each polygon. In the case of online social media data, this approach has two specific disadvantages. First, the approach yields a ratio that becomes difficult to understand; for example, what does 15 tweets per square mile or 100,000 people actually mean? Second, the total population might very well not be the same as total tweeting population.

Instead, we calculate an odds ratio, which is slightly more sophisticated but has the great advantage of allowing us to normalize by any other variable, and the resulting ratios are easy to interpret (Edwards, 1963). In the case of social media data such as Twitter, it often makes sense to normalize

by a random sample of all tweets that stands in as a proxy for the total tweeting population rather than the total population in and of itself. By using the total tweeting population, we can visualize the distribution of a phenomenon within social media use, rather than the popularity of a social media service within the overall population. The formula for the odds ratio is—

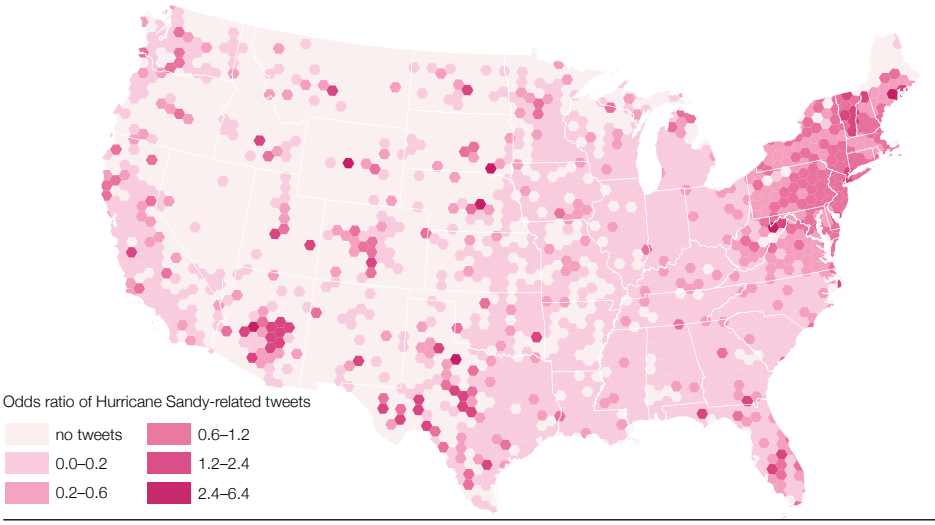
$$OR = \frac{p_i/p}{r_i/r}, \quad (1)$$

where p_i is the number of tweets in hexagon i related to the phenomenon of interest (for example, flu shot tweets or tweets related to a certain beer brand) and p is the sum of all tweets related to that phenomenon in all hexagons. r_i is the number of random tweets in hexagon i and r is the sum of all random tweets in all hexagons. We choose a random sample of all tweets at this point, but one could easily substitute other variables—for example, active Internet users or possibly another point-based phenomenon aggregated to the same hexagonal lattice.

The resulting ratio has a midpoint of 1. At that midpoint, as many data points related to our phenomenon of interest as we would expect are present based on that random sample of all tweets. Values lower than 1 indicate we have fewer points of interest than expected, and vice versa. For example, an odds ratio of 0.5 means that we find only half as many points of interest as we expected, and a value of 2.0 means we find twice as many points as we expected, based on the total population. We can easily calculate this odds ratio in R (see result in exhibit 4).

Exhibit 4

Basic Odds Ratio



```
randomTweets <- read.csv("random.csv", colClasses=c("latitude"="numeric",
    "longitude"="numeric"))
coordinates(randomTweets) <- c("longitude", "latitude")
proj4string(randomTweets) <- "+proj=longlat +ellps=WGS84 +datum=WGS84
    +no_defs"
randomTweets$hex <- over(randomTweets, hex)$id
randomTweets.dt <- data.table(randomTweets)
randomTweets.dt[,random=.N, by=list(hex)]
hex.join <- merge(tweets.dt, randomTweet)
hex.join[,OR:=(tweets/sum(tweets))/(random/sum(random)),]
```

Small Numbers Problem: Confidence Intervals

When the odds ratio for each hexagon is visualized, we finally start to gain a meaningful understanding of the spatial pattern. Furthermore, we have now solved the issue around population density that the XKCD comic pointed out so vividly. One last problem remains, however: areas with only a few tweets have wildly varying odds ratios. In exhibit 4, this variation can be clearly seen in sparsely populated states such as Wyoming and Montana. This issue is mostly the result of the small numbers problem. Simply put, areas with only a small number of tweets show a high degree of variance. This makes sense: an odds ratio based on two tweets of interest versus three random "population" tweets is less "reliable" than the same ratio based on 200 tweets of interest versus 300 random tweets. We can thus calculate a confidence interval for each odds ratio to gain an indicator of reliability (Bland and Altman, 2000; Morris and Gardner, 1988).

$$OR_{CI} = e^{\ln(OR) \pm z \times \sqrt{\frac{1}{p_i}+\frac{1}{p}+\frac{1}{r_i}+\frac{1}{r}}}, \quad (2)$$

where z is the z-score of the chosen confidence level (for example, z = 1.96 for 95 percent confidence level).

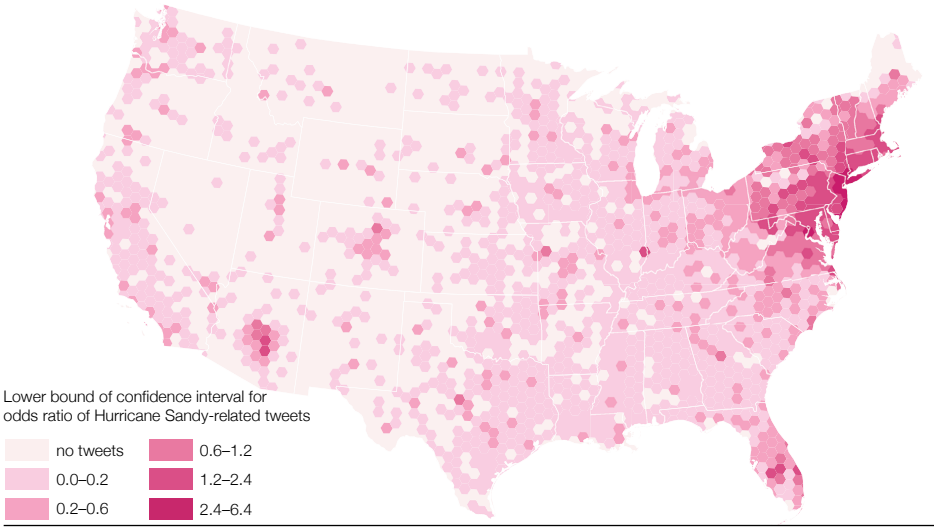
We can use this approach to calculate both the upper and lower bounds of the confidence interval but, if we are interested only in significant instances of higher odds ratios, we can calculate and visualize the lower bound only. This approach would enable one to say, for example, the value in hexagon *I* is *at least* 1.5, with 95 percent confidence. To calculate this value in R, we only have to adapt the formula (the last line of the previous code snippet) a little bit.

```
hex.join[,ORlowerconf:=exp(log(OR)-1.96*sqrt(1/tweets+1/sum(tweets)+1/
    random+1/sum(random))),]
```

When we visualize this lower bound of the confidence interval for the odds ratio, we get to the final step in our approach, seen in exhibit 5, which results in a clear—and in this case, expected—spatial pattern largely following the areas most affected by Hurricane Sandy (see Shelton, 2014, for a more indepth discussion of this pattern).

Exhibit 5

Lower Bound of Confidence Interval for Odds Ratio



Source: Reprinted from Shelton (2014)

Final Considerations

The approach outlined in this article starts with an arguably noisy and large set of point-level data derived from social media. Using aggregation to a hexagonal lattice and subsequent normalization and calculation of an odds ratio with confidence intervals, we go from a raw view on the data (exhibit 1) to a clear spatial pattern (exhibit 5). Although we have used a random "population" sample to normalize in the example, this approach is flexible; thus, the same approach can be used to directly compare two different point datasets (for example, artists versus bankers) or different time periods of the same dataset.

Furthermore, in many cases, this step will be only the first in a more indepth spatial analysis, especially because the hexagonal lattice is a very suitable input for further analysis with subsequent spatial statistical techniques (for example, cluster detection or spatial regression models).

Acknowledgments

The Department of Geography and the Vice President for Research at the University of Kentucky supported this work.

Authors

Ate Poorthuis is a Ph.D. candidate in the Department of Geography at the University of Kentucky.

Matthew Zook is a professor in the Department of Geography at the University of Kentucky.

References

Bland, J. Martin, and Douglas G. Altman. 2000. "The Odds Ratio," *BMJ* 320 (7247): 1468.

Burt, Peter J. 1980. "Tree and Pyramid Structures for Coding Hexagonally Sampled Binary Images," *Computer Graphics and Image Processing* 14 (3): 271–280.

Carr, Daniel B., Anthony R. Olsen, and Denis White. 1992. "Hexagon Mosaic Maps for Display of Univariate and Bivariate Geographical Data," *Cartography and Geographic Information Systems* 19 (4): 228–236.

Clauset, Aaron, Cosma R. Shalizi, and Mark E. Newman. 2009. "Power-Law Distributions in Empirical Data," *SIAM Review* 51 (4): 661–703.

Crampton, Jeremy W., Mark Graham, Ate Poorthuis, Taylor Shelton, Monica Stephens, Matthew W. Wilson, and Matthew Zook. 2013. "Beyond the Geotag: Situating 'Big Data' and Leveraging the Potential of the Geoweb," *Cartography and Geographic Information Science* 40 (2): 130–139.

Edwards, Anthony W. 1963. "The Measure of Association in a 2 x 2 Table," *Journal of the Royal Statistical Society* Series A (General): 109–114.

Morris, Julie A., and Martin J. Gardner. 1988. "Statistics in Medicine: Calculating Confidence Intervals for Relative Risks (Odds Ratios) and Standardised Ratios and Rates," *British Medical Journal (Clinical Research ed.)* 296 (6632): 1313–1316.

Poorthuis, Ate, and Matthew Zook. 2014. "Artists and Bankers and Hipsters, Oh My! Mapping Tweets in the New York Metropolitan Region," *Cityscape* 16 (2): 169–172.

R Core Team. 2014. *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing. http://www.R-project.org/.

Scott, David W. 1988. "A Note on Choice of Bivariate Histogram Bin Shape," *Journal of Official Statistics* 4 (1): 47–51.

Shelton, Taylor, Ate Poorthuis, Mark Graham, and Matthew Zook. 2014. "Mapping the Data Shadows of Hurricane Sandy: Uncovering the Sociospatial Dimensions of 'Big Data,'" *Geoforum* 52: 167–179.

Snow, John. 1855. *On the Mode of Communication of Cholera.* London, United Kingdom: Churchill.

Wilson, Ron. 2013. "Changing Geographic Units and the Analytical Consequences: An Example of Simpson's Paradox," *Cityscape* 15 (2): 289–304.