Hurricane Sandy Maps for New York City

Charles DiMaggio

November 14, 2012

1 Data

1.1 Con Edison Outages

On November 4th, I downloaded outage map info from con ed site under "summary" then "new york city details". The data table was in a file called multipleoutagereportpanel_nyc.html. I created a .csv file called conEdCoordinates.csv. that contains the number of outages ,the number of customers, and the place names that Con Ed uses. There was also a file for place name coordinates from which I created a file called conEdCoordinates.csv

1.2 FEMA Housing Damage Data

The google crisis map for Sandy had damage assessments referenced to FEMA. But getting information from that site requires multiple queries for each version, a laborious, time-consuming process I went through only to come up against the wall of trying to convert the resulting KMZ files to KML to shapefiles. (see the appendix to this note)

Fortunately Jonathan Sury, a colleague at Columbia's National Center for Disaster Preparedness, was able to track down shapefiles posted by Google. From there is was a simple matter of downloading and unzipping the files.

Following is the description of the point data set from the FEMA site:

"Point dataset representing individual analysis at the building level. Attributes show presence or absence of flood water ("INUNDATED") and level of damage ("DAMAGE"). The field "DAMAGETYPE" shows a combination of damage types and can be used for thematic mapping. Explanation of observation-based DAMAGE values: Affected: Generally superficial damage to solid structures (loss of tiles or roof shingles); some mobile homes and light structures damaged or displaced. Minor: Solid structures sustain exterior damage (e.g., missing roofs or roof segments); some mobile homes and light structures are destroyed, many are damaged or displaced. Major: (Wind) Some solid structures are destroyed; most sustain exterior and interior damage (roofs missing, interior walls exposed); most mobile homes and light structures are destroyed. (Storm Surge) Extensive structural damage and/or partial collapse due to surge effects. Partial collapse of exterior bearing walls. Destroyed: (Wind) Most solid and all light or mobile home structures destroyed. (Storm Surge) The structure has been completely destroyed or washed away by surge effects. Assessments are made from NOAA aerial imagery, Civil Air Patrol photographs, and media images captured 29-Oct, 30-Oct, 31-Oct, 01-Nov, 02-Nov, 03-Nov, 04-Nov 2012. Version 14 - 2012-11-04 15:00 PST "

The levels of damage are described as:

"

Field 1: DAMAGETYPE Field 2: DAMAGE

Value: DAMAGE AND INUNDATION, Destroyed Label: DAMAGE AND INUNDATION, Destroyed Description: Building is marked has having both damage and inundation. Destroyed: (Wind) Most solid and all light or mobile home structures destroyed. (Storm Surge) The structure has been completely destroyed or washed away by surge effects.

Value: DAMAGE AND INUNDATION, Major Label: DAMAGE AND INUNDATION, Major Description: Building is noted has having both damage and inundation. Major: (Wind)

Some solid structures are destroyed; most sustain exterior and interior damage (roofs missing, interior walls exposed); most mobile homes and light structures are destroyed. (Storm Surge) Extensive structural damage and/or partial collapse due to surge effects. Partial collapse of exterior bearing walls.

Value: DAMAGE AND INUNDATION, Minor Label: DAMAGE AND INUNDATION, Minor Description: Building is noted has having both damage and inundation. Minor: Solid structures sustain exterior damage (e.g., missing roofs or roof segments); some mobile homes and light structures are destroyed, many are damaged or displaced.

Value: DAMAGE AND INUNDATION, Affected Label: DAMAGE AND INUNDATION, Affected Description: Building is noted has having both damage and inundation. Affected: Generally superficial damage to solid structures (loss of tiles or roof shingles); some mobile homes and light structures damaged or displaced.

Value: DAMAGE ONLY, Destroyed Label: DAMAGE ONLY, Destroyed Description: Building is noted has having damage. Destroyed: (Wind) Most solid and all light or mobile home structures destroyed. (Storm Surge) The structure has been completely destroyed or washed away by surge effects.

Value: DAMAGE ONLY, Major Label: DAMAGE ONLY, Major Description: Building is noted has having damage. Major: (Wind) Some solid structures are destroyed; most sustain exterior and interior damage (roofs missing, interior walls exposed); most mobile homes and light structures are destroyed. (Storm Surge) Extensive structural damage and/or partial collapse due to surge effects. Partial collapse of exterior bearing walls.

Value: DAMAGE ONLY, Minor Label: DAMAGE ONLY, Minor Description: Building is noted has having damage. Minor: Solid structures sustain exterior damage (e.g., missing roofs or roof segments); some mobile homes and light structures are destroyed, many are damaged or displaced.

Value: DAMAGE ONLY, Affected Label: DAMAGE ONLY, Affected Description: Building is noted has having damage. Affected: Generally superficial damage to solid structures (loss of tiles or roof shingles); some mobile homes and light structures damaged or displaced.

Value: INUNDATION ONLY, No Damage Label: INUNDATION ONLY, No Damage Description: Building is noted has having inundation.

Value: INUNDATION ONLY, Unknown Label: INUNDATION ONLY, Unknown Description: Building is noted has having inundation.

"

2 Mapping Housing Damage

2.1 Preparing the Data

```
> library(maptools)
> library(rgdal)
> fema.points<-readOGR("/Users/charlie/Dropbox/miseryMap/data/femaPoints/", "femaPoint
OGR data source with driver: ESRI Shapefile
Source: "/Users/charlie/Dropbox/miseryMap/data/femaPoints/", layer: "femaPoints"
with 107086 features and 15 fields
Feature type: wkbPoint with 2 dimensions
> boros<-readOGR("/Users/charlie/Dropbox/srtsAnalysis/srtsAnalysisDataSets/nybb/", "ny
OGR data source with driver: ESRI Shapefile
Source: "/Users/charlie/Dropbox/srtsAnalysis/srtsAnalysisDataSets/nybb/", layer: "nybb"
with 5 features and 4 fields
Feature type: wkbMultiPolygon with 2 dimensions
```

```
> fema.points<-spTransform(fema.points, CRS("+proj=longlat +datum=NAD83"))
> boros<-spTransform(boros, CRS("+proj=longlat +datum=NAD83"))
```

This syntax can be used for some simple plots.

```
> plot(fema.points,col="red", pch=20, cex=.1)
> plot(boros, add=T, lty=1, lwd=.5)
```

This syntax overlays (actually indexes) the FEMA data to NYC boundaries.

```
> vignette("over")
> plot(fema.points[boros,], col="red", pch=20, cex=.3)
> plot(boros, add=T, lty=1, lwd=.2)
> title(main="Hurricane Sandy Housing Damage", sub="FEMA Flyover Data November 2012")
```

2.2 Density Map

The following bits of code demonstrate how to plot the points, and then develop a density map from points. First you load ggplot2, create a spatial object restricted to NYC, and (because ggplot doesn't play nicely with spatial objects) convert the spatial object to a data frame for plotting with ggplot2.

```
> library(ggplot2)
> fema.nyc<-fema.points[boros,]
> fema.nyc.df<-as.data.frame(fema.nyc)
> # print(qplot(fema.nyc.df$LONGITUDE, fema.nyc.df$LATITUDE))
```

The qplot() function can return a quick plot.

```
> print(qplot(LONGITUDE,LATITUDE, data=fema.nyc.df, geom='density2d'))
```

The next series of code snippets builds up a density map layer by layer. First, the base map.

```
> p<-ggplot() +
+ geom_density2d(aes(x=LONGITUDE, y=LATITUDE), data=fema.nyc.df) +
+ geom_point(aes(x=LONGITUDE, y=LATITUDE, col="red"), data=fema.nyc.df, size=.9) +
+ theme(axis.text.x = element_blank(), axis.text.y = element_blank(), axis.ticks = ele
+ theme(panel.background = element_rect(colour = NA)) +
+ xlab("")+ylab("") +
+ theme(legend.position="none") +
+ ggtitle("Hurricane Sandy Housing Damage, New York City, Nov 2012")
> # print(p)
```

To provide some reference, I overlay the borders of the NYC boros. Here I use the fortify method to convert the shape files to a data frame for use with ggplot2.

```
> gpclibPermit()
[1] TRUE
> boros_df <- fortify(boros, region='BoroCode')
```

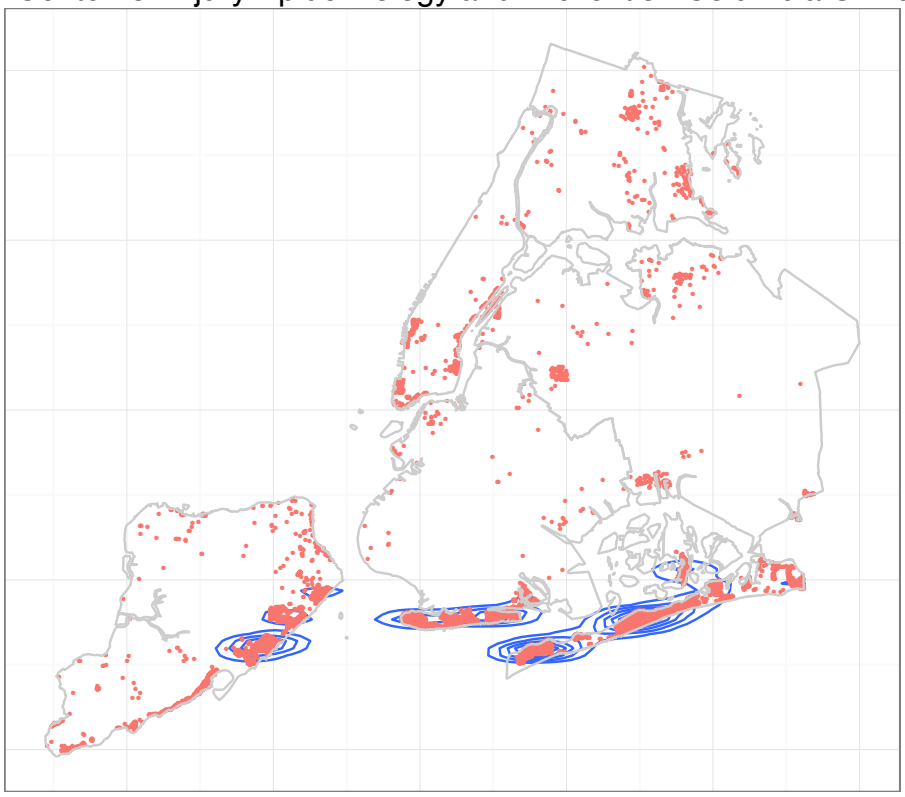
Now overlay the boro borders

```
> p1<-p+geom_polygon(data=boros_df,aes(long, lat,group=group),fill="NA", color="#CDCDC
> # print(p1)
```

Then some cleaning up and plotting.

```
> p2<-p1+theme_bw()+
+ theme(axis.text.x = element_blank(), axis.text.y = element_blank(), axis.ticks = ele
+ theme(panel.background = element_rect(colour = NA)) +
+ xlab("")+ylab("") +
+ theme(legend.position="none") +
+ ggtitle("Density Map Hurricane Sandy Housing Damage, \n New York City FEMA Data Nov
> print(p2)
```

Density Map Hurricane Sandy Housing Damage,
New York City FEMA Data Nov 2012
(from Center for Injury Epidemiology and Prevention Columbia Univers



2.3 "Weather Map" Effects

As an alternative that may be less informative, but perhaps easier to interpret, I recast the density map as with smoothed "weather map" effects. I again build up a map layer by layer. First, I establish the base layer of points. I then add a layer of density plots.

```
> p<-ggplot(fema.nyc.df, aes(x=LONGITUDE, y=LATITUDE))
> p1<-p+stat_density2d(aes(alpha=..level..), geom="polygon")
> # print(p1)
```

Now I smooth the levels of the density polygons using the alpha function.

```
> p2<-p1+scale_alpha_continuous(limits=c(0,500),breaks=seq(0,500,by=50))
> # print(p2)
```

I overlay the locations of the points themselves, coloring them red, and blurring overlaid points (again, with the alpha function)

```
> p3<-p2+geom_point(colour="red",alpha=0.009)
> # print(p3)
```

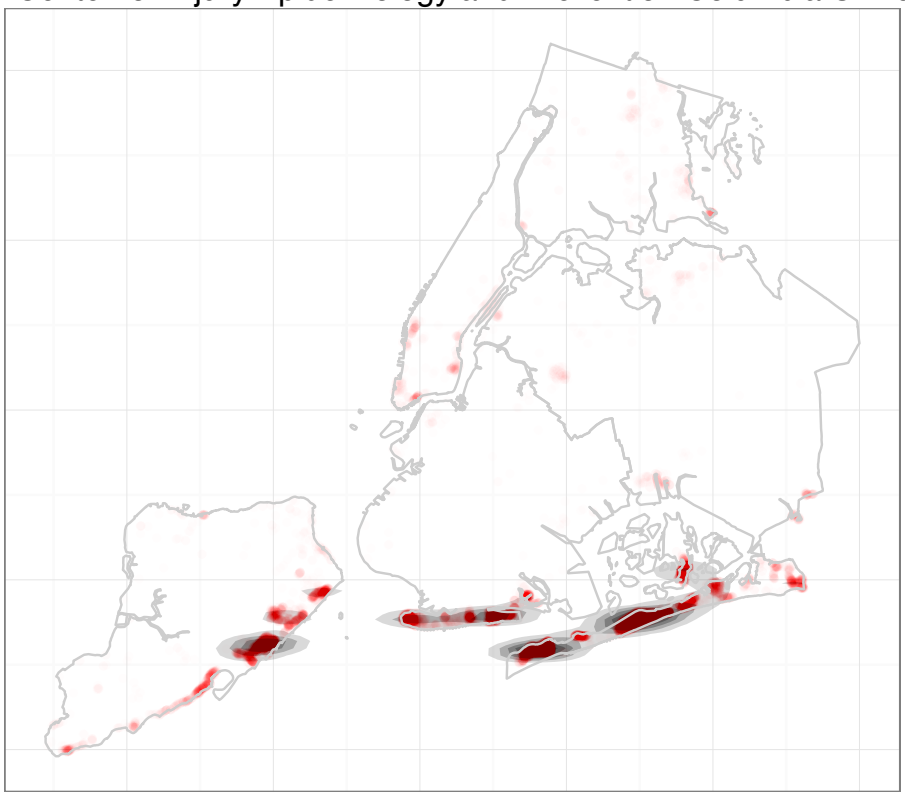
Then I overlay the boro borders.

```
> p4<-p3+geom_polygon(data=boros_df,aes(long, lat,group=group),fill="NA", color="#CDCD
> # print(p4)
```

And add a title and clean up the theme elements.

```
> p5<-p4+theme_bw()+
+ theme(axis.text.x = element_blank(), axis.text.y = element_blank(), axis.ticks = ele
+ theme(panel.background = element_rect(colour = NA)) +
+ xlab("")+ylab("") +
+ theme(legend.position="none") +
+ ggtitle("Hurricane Sandy Housing Damage, New York City, Nov 2012 \n Based on FEMA Da
> print(p5)
```

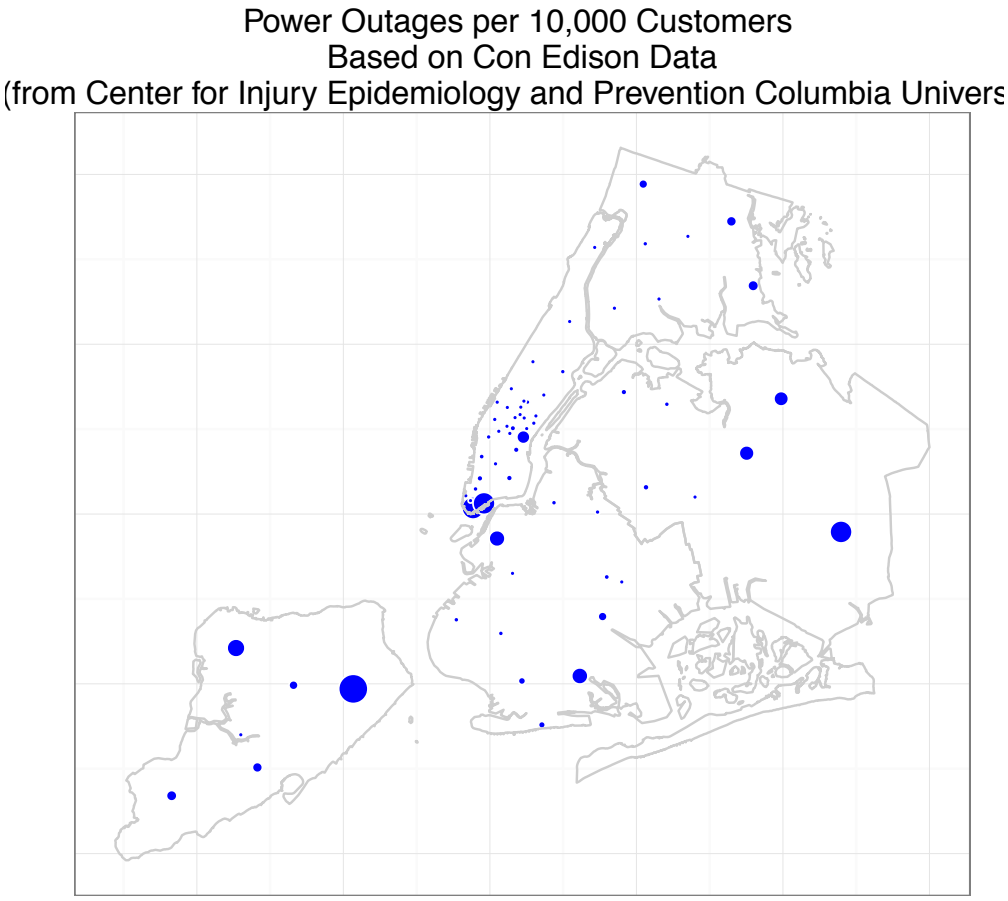
Hurricane Sandy Housing Damage, New York City, Nov 2012
Based on FEMA Data
(from Center for Injury Epidemiology and Prevention Columbia Univers



2.4 Mapping Outages

```
> outages<-read.csv("/Users/charlie/Dropbox/miseryMap/data/conEdOutages.csv",header=T,
> places<-read.csv("/Users/charlie/Dropbox/miseryMap/data/conEdCoordinates.csv",header
> conEd<-merge(outages, places, by="location", all.y=T)
> conEd$outages[is.na(conEd$outages)]<-0
> conEd$customers[is.na(conEd$customers)]<-.1
> conEd$propOut<-conEd$outages/conEd$customers*100
> summary(conEd$propOut)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 0.0000  0.0000  0.0000  1.4890  0.2548 22.1500
> coordinates(conEd)<-~long+lat
> proj4string(conEd)<-CRS("+proj=longlat +datum=NAD83")
> # plot(conEd[boros,],col="blue", pch=20, cex=.9)
>
> conEd.nyc<-conEd[boros,]
> conEd.nyc.df<-as.data.frame(conEd.nyc)
> p6<-ggplot(conEd.nyc.df, aes(x=long, y=lat, size=propOut, label=location, shape=21))
> p7<-p6+geom_point(color="white", fill="blue")
> # print(p7)
>
> p8<-p7+geom_polygon(data=boros_df,aes(long, lat,group=group),fill="NA", color="#CDCD
> p9<-p8+theme_bw()+
+ theme(axis.text.x = element_blank(), axis.text.y = element_blank(), axis.ticks = ele
+ theme(panel.background = element_rect(colour = NA)) +
+ xlab("")+ylab("") +
+ theme(legend.position="none") +
+ ggtitle("Power Outages per 10,000 Customers \n Based on Con Edison Data \n (from Cen
> print(p9)
```



This is not entirely satisfactory, as places without power outages show up as small dots, and (more importantly) the hard hit areas of the Rockaways do not show up at all, because they are serviced by LIPA.

3 appendix

3.1 Working with FEMA Service Files

A post led me to fellow named Bob, who led me to a post on stackexchange that described the process of getting the data as follows:

”That map service is a ArcGIS Server Dynamic Map service which typically only returns images and specific query results, much like a WMS. Some ArcGIS Server image services allow for data download, but this isn't one of those.

You can get the information you're looking for through the query operation, but it will take multiple queries as the max record return limit on this service is 1000, and there are over 58000 records in the v14 layer alone.

To submit queries, go to the layer endpoint and scroll to the bottom for supported operations, then click Query. To find the total number of records, enter $1\bar{1}$ in the Where field, and select True for Return Count Only, looks like 58919 records. It will only return 1000 records at a time, so you will have to issue 59 queries to get all the records. Seems easiest to filter by OBJECTID, so the first query would be WHERE OBJECTID 1000, Out Fields $^{-}$, Return Geometry $\bar{T}$RUE, Return Count $\bar{F}$alse, and Format is HTML. The service returns all the features 1- 999 and their attributes. You could change the output format to KMZ and convert to desired format later, iterate through and you would have all the data.”

I did this for the 4 different ”versions” or fly overs (11-14), ending up with about 200 separate KMZ files. The KMZ files are zipped version of the KML (keyhole markup) files which can be read into R.

Worked with KMZ files in Google Earth (labor intensive, very pointy clicky) For each of the 50 or so separate KMZ files for each ”version”: Opened the KMZ files into the default ”temporary” place foder; R clicked ”My Places”, created new folder to hold the KML files manually clicked and dragged each opened KMZ file into the newly created folder R clicked the newly created folder that now holds all the KMZ files, chose ”save place as...”, and saved it as a KML file

Then, since it appeared that only one of the components of the 50 or so downloaded queries for each ”version” contained point process data, I selected the 4 query files (one from each newly created KML file) and combined them into a single file called ”femaPoints”

3.2 Reading the KML files into R

Getting the kml file into a shapefile or directly into R as an sp object turned out to be something of an issue. I tried the following approaches:

1) Read into R using OGR and GDAL library

```
> library(rgdal)
> temp<-readOGR("/Users/charlie/Desktop/miseryMap/fema/femaPoints/", "femaPoints")
>  # error in reading file,
> #  advice from roger bivand at https://stat.ethz.ch/pipermail/r-sig-geo/2010-May/008
```

2) Convert to shape file in GRASS by first reading in then using v.out.ogr. But did not retain the attribute data.

3) Convert to kml csv using http://chroonchernlim.com/kmlcsv/, But returned error messages; apparently the kml file is not structured in a regular format

4) Parse the kml file (which is actually an xml file) using the R XML package

```
> library(XML)
> # no luck trying to go directly to a list or data frame as per https://stat.ethz.ch/
> kml.frame<-xmlToList("/Users/charlie/Desktop/miseryMap/fema/femaPoints/femaPoints.km
> kml.frame<-xmlToDataFrame("/Users/charlie/Desktop/miseryMap/fema/femaPoints/femaPoin
> # parsing down the tree to get at nodes and elements was promising, but requires me
> doc<-xmlTreeParse("/Users/charlie/Desktop/miseryMap/fema/femaPoints/femaPoints.kml",
> top<-xmlRoot(doc)
> xmlName(top)
> names(top[[1]][3])
> doc<-top[[1]]["Document"]
> names(doc[[1]][12])
> folder1<-doc[[1]][12]
```

```
> str(folder1)
> getNodeSet(folder1, "//table[tr/td/b/text()='INUNDATED']")
```

5) Using maptools to extract the coordinates without the attributes.

```
> library(maptools)
> fema.points<-getKMLcoordinates("/Users/charlie/Desktop/miseryMap/fema/femaPoints/fema
> str(fema.points)
> head(fema.points)
> df <- data.frame(matrix(unlist(fema.points), nrow=3996, byrow=T))
> head(df)
> write.csv(df, file="/Users/charlie/Desktop/miseryMap/fema/femaPoints/femaPoints.csv"
```