Spatial Effects of Nonprofit Services in Impoverished Communities

Matt McKnight

May 11, 2017

Create NPO data.frame

```
# Import subset from NCCS BMF Data on nonprofits in Syr MSA
load("C:/Users/mlmck/Documents/analyzing-nonprofit-service-areas/DATA/syr_orgs.Rda")

# Subset - Extract poverty orgs
pov_orgs <- syr_orgs[which (  syr_orgs$NTEECC == "B61" |
                              syr_orgs$NTEECC == "B82" |
                              syr_orgs$NTEECC == "B92" |
                            syr_orgs$nteeFinal1 == "E" |
                            syr_orgs$nteeFinal1 == "F" |
                            syr_orgs$nteeFinal1 == "I" |
                              syr_orgs$NTEECC == "K30" |
                              syr_orgs$NTEECC == "K31" |
                              syr_orgs$NTEECC == "K34" |
                              syr_orgs$NTEECC == "K35" |
                              syr_orgs$NTEECC == "K36" |
                            syr_orgs$nteeFinal1 == "L" |
                            syr_orgs$nteeFinal1 == "O" |
                            syr_orgs$nteeFinal1 == "P" |
                            syr_orgs$nteeFinal1 == "R" |
                            syr_orgs$nteeFinal1 == "S" |
                            syr_orgs$nteeFinal1 == "T"), ]

#DROP Vars with NAs
pov_orgs$PFFRCD <- pov_orgs$TAXPER <- NULL
```

Geocode selected Poverty Organization data

```
print("Don't run me")

# download specific ggmap, register Google key
devtools::install_github("dkahle/ggmap")
register_google(key = 'AIzaSyAMrECdXqpOt443ms6nzh118uUsqC2lE6M',
                account_type = "premium", day_limit = 15000)

# subset orgs for gecoding address information
pov_orgs_gps <- subset(pov_orgs, select=c(EIN:INCOME))

pov_orgs_gps$whole_address <- do.call(paste, c(pov_orgs_gps[
                                      c("ADDRESS", "CITY","STATE",
                                        "ZIP5")], sep = ", "))
```

```
pov_orgs_gps <- subset(pov_orgs_gps, select=c(whole_address, EIN:INCOME))

for (i in 1:nrow(pov_orgs_gps)) {
  latlon = geocode(pov_orgs_gps[i,1])
  pov_orgs_gps$lon[i] = as.numeric(latlon[1])
  pov_orgs_gps$lat[i] = as.numeric(latlon[2])
}

# Make sure you find a way to then take any PO Boxes and place them in the
# center of centroids, and only the ones that are NAs.
```

```
# Dropping missing values
head(syr_orgs)
```

```
##            EIN  FIPS NTEECC FILER ZFILER
## 3756 010556504 36053    B12     Y      N
## 4256 010587946 36067    L25     Y      Y
## 4469 010601834 36067    T20     Y      Y
## 4477 010602290 36075   I032     Y      Y
## 5290 010661700 36067    W70     Y      N
## 5434 010672153 36067   E039     Y      Y
##                                                        NAME
## 3756              ONEIDA CITY SCHOOL DISTRICT FOUNDATION INC
## 4256                             MIGHTYMEN MINISTRIES INC
## 4469                   EAST SYRACUSE RECREATION COMMITTEE INC
## 4477 ASSOCIATION OF NEW YORK STATE YOUTH COURTS INCORPORATED
## 5290                       EDUCATION LEADERSHIP INSTITUTE INC
## 5434            INTERNATIONAL ASSOCIATION OF FORENSIC NURSES
##                      ADDRESS          CITY STATE  ZIP5  GEN SUBSECCD
## 3756             317 MAIN ST        ONEIDA    NY 13421 0000       03
## 4256          108 SUMMIT AVE      SYRACUSE    NY 13207 0000       03
## 4469 204 NORTH CENTER STREET EAST SYRACUSE    NY 13057 0000       03
## 4477            70 BUNNER ST        OSWEGO    NY 13126 0000       03
## 5290           6390 FLY ROAD EAST SYRACUSE    NY 13057 0000       03
## 5434 4886 HILLOCK MEADOWS DR      SYRACUSE    NY 13215 5479       06
##      RULEDATE FNDNCD TAXPER FRCD PFFRCD ACCPER ASSETS INCOME
## 3756   200206     16 201512   02   <NA>     12  82052  39921
## 4256   200206     15 201412   02   <NA>     12      0      0
## 4469   200405     15 201412   02   <NA>     12      0      0
## 4477   200401     15 201412   02   <NA>     12      0      0
## 5290   200304     04 201506   00   <NA>     06   5694      6
## 5434   200807     00 201512   02   <NA>     12      0      0
##                     SEC_NAME NTEE1 LEVEL1 LEVEL2 LEVEL3 LEVEL4 MAJGRPB
## 3756                    <NA>     B     PC      S     ZB      B       B
## 4256                    <NA>     L     PC      O     HS      L       L
## 4469                    <NA>     T     PC      S     ZC      T       T
## 4477                    <NA>     I     PC      O     HS      I       I
## 5290                    <NA>     W     PF      O     PB      W       W
## 5434 NEW YORK STATE CHAPTER     E      O      O     HE      E       E
##      OUTNCCS OUTREAS NTEESRC ntmaj10 ntmaj12 ntmaj5 nteeFinal nteeFinal1
## 3756      IN    <NA>    <NA>      ED      ED     ED       B12          B
## 4256      IN    <NA>    <NA>      HU      HU     HU       L25          L
## 4469      IN    <NA>    <NA>      PU      PU     OT       T20          T
```

```
## 4477      IN    <NA>    <NA>      HU      HU     HU     I0321          I
## 5290      IN    <NA>    <NA>      PU      PU     OT       W70          W
## 5434      IN    <NA>    <NA>      HE      HE     HE     E0390          E
##      RandNum nteeConf MSA_NECH PMSA cFiler czFiler cTaxPer cAssets cTotRev
## 3756 0.14553        A     8160 <NA>      Y       N  201512   82052   29338
## 4256 0.84984        B     8160 <NA>      Y       Y  201412      NA      NA
## 4469 0.89895        B     8160 <NA>      Y       Y  201412      NA      NA
## 4477 0.56087        A     8160 <NA>      Y       Y  201412      NA      NA
## 5290 0.57753        A     8160 <NA>      Y       N  201406    6114     670
## 5434 0.65598        B     8160 <NA>      Y       Y  201512      NA      NA
##            cFinSrc EPOST FISYR_IMAGE IRS990n  NAICS
## 3756     bmf1608c3     Z        2014       0 813219
## 4256 submaster990n     Z          NA       1 624229
## 4469 submaster990n     Z        2009       1 813211
## 4477 submaster990n     Z        2003       1 813920
## 5290    core2013pf     Z        2015       0 813410
## 5434 submaster990n     Z          NA       1 813920
```

```
load("F:/Spring 2017/R Course/Data Project/R Files/pov_orgs_gps.Rda")
pov_orgs_gps_nona <- na.omit(pov_orgs_gps)
```

Create Service Catchment Area data

```
library( sp )
library( rgeos )
library( spatialEco )
library( ggmap )
library( rgdal )
```

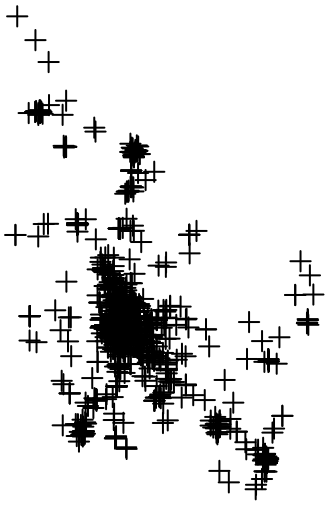
```
class(pov_orgs_gps_nona)
```

```
## [1] "data.frame"
```

```
coordinates(pov_orgs_gps_nona) = ~lat+lon
class( pov_orgs_gps_nona )
```

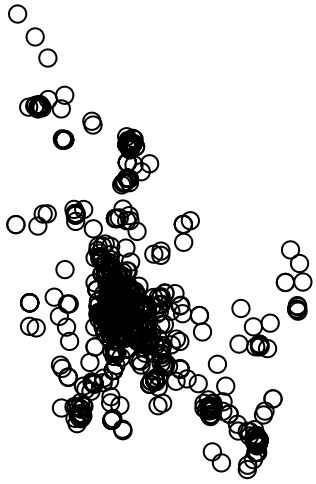
```
## [1] "SpatialPointsDataFrame"
## attr(,"package")
## [1] "sp"
```

```
plot( pov_orgs_gps_nona )
```



```
# Create buffer zone around each point

pov_orgs_gps_nona_buff <- gBuffer( pov_orgs_gps_nona, width=.025, byid=TRUE )
plot( pov_orgs_gps_nona_buff, main="NPO service buffer zone (1.5 miles?)" )
```

NPO service buffer zone (1.5 miles?)



```
class( pov_orgs_gps_nona_buff )
```

```
## [1] "SpatialPolygonsDataFrame"
## attr(,"package")
## [1] "sp"
```

Download & clean Census ACS 2011-2015 5-yr data

```
# Census API
# install.packages("devtools")
devtools::install_github("hrecht/censusapi")
censuskey <- "f8064a17832b439e690be95ab0e0ef9a78617584"


# Us Census API Extraction
library(devtools)
library(censusapi)
library(plyr)
library(gdata)
```

```
acs_2015 <- getCensus(name="acs5",
                      vintage = 2015,
                      key = censuskey,
                      vars=c("NAME",
                             "B25077_001E", "B25003_001E", "B25003_002E",
```

```
                              "B25003_003E", "B25002_001E", "B25002_002E",
                              "B25002_003E",
                              "B01003_001E", "B02001_001E", "B02001_002E",
                              "B02001_003E", "B02001_004E", "B02001_005E",
                              "B02001_006E", "B02001_007E", "B02001_008E",
                              "B03001_003E",
                              "GEOID"),
                        region="tract:*", regionin="state:36")

# subset to Syracuse
# sum(acs_2015$county == "067")
# sum(acs_2015$county == "075")
# sum(acs_2015$county == "053")

acs_2015_syr <- acs_2015[which (acs_2015$county == "067" |
                                acs_2015$county == "075" |
                                acs_2015$county == "053"), ]
```

```
rename(acs_2015_syr,
       c("B25077_001E" = "mdn_hous_val", "B25003_001E" = "tenure_tot",
         "B25003_002E" = "tenure_own", "B25003_003E" = "tenure_rent",
         "B25002_001E" = "tot_occup", "B25002_002E" = "occupied",
         "B25002_003E" = "vacant", "B01003_001E" = "tot_pop",
         "B02001_001E" = "tot_race", "B02001_002E" = "white",
         "B02001_003E" = "black", "B02001_004E" = "am_ind",
         "B02001_005E" = "asian", "B02001_006E" = "islander",
         "B02001_007E" = "other", "B02001_008E" = "mixed",
         "B03001_003E" = "hispanic"
         ))


acs_2015_syr <- rename.vars(acs_2015_syr,
            c("B25077_001E", "B25003_001E", "B25003_002E", "B25003_003E",
              "B25002_001E", "B25002_002E", "B25002_003E", "B01003_001E",
              "B02001_001E", "B02001_002E", "B02001_003E", "B02001_004E",
              "B02001_005E", "B02001_006E", "B02001_007E", "B02001_008E",
              "B03001_003E"),
            c("mdn_hous_val", "tenure_tot", "tenure_own", "tenure_rent",
              "tot_occup", "occupied", "vacant", "tot_pop",
              "tot_race", "white", "black", "am_ind",
              "asian", "islander", "other", "mixed",
              "hispanic"))

names(acs_2015_syr)
```

Census Shapefiles

```
# load package: rgdal
library(rgdal)
library( geojsonio )
```

```r
setwd("C:/Users/mlmck/Documents/analyzing-nonprofit-service-areas/")


# Census tracts shapefiles of Syracuse MSA, changed to geojson files
Mad_county <- readOGR(dsn="SHAPEFILES/Madison-ct", layer="tl_2010_36053_tract10")
```

```
## OGR data source with driver: ESRI Shapefile
## Source: "SHAPEFILES/Madison-ct", layer: "tl_2010_36053_tract10"
## with 16 features
## It has 12 fields
## Integer64 fields read as strings:  ALAND10 AWATER10
```

```r
geojson_write( Mad_county, geometry="polygon", file="Mad_county.geojson" )
```

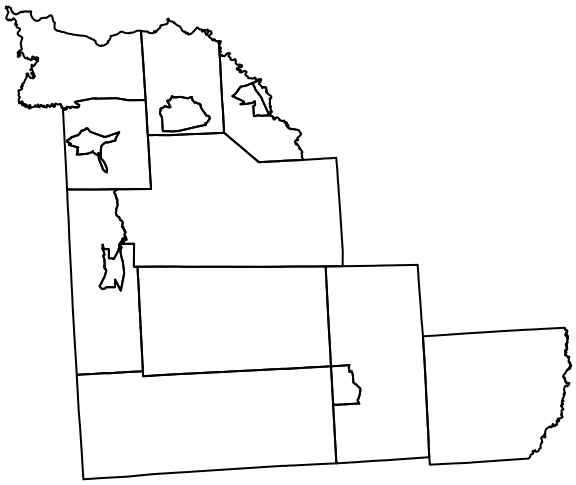
```
## Success! File is at Mad_county.geojson
```

```
## <geojson>
##   Path:       Mad_county.geojson
##   From class: SpatialPolygonsDataFrame
```

```r
class(Mad_county)
```

```
## [1] "SpatialPolygonsDataFrame"
## attr(,"package")
## [1] "sp"
```

```r
plot(Mad_county)
```



```
names(Mad_county)
```

```
##  [1] "STATEFP10"  "COUNTYFP10" "TRACTCE10"  "GEOID10"    "NAME10"
##  [6] "NAMELSAD10" "MTFCC10"    "FUNCSTAT10" "ALAND10"    "AWATER10"
## [11] "INTPTLAT10" "INTPTLON10"
```

```
On_county <- readOGR(dsn="SHAPEFILES/Onondaga-ct", layer="tl_2010_36067_tract10")
```

```
## OGR data source with driver: ESRI Shapefile
## Source: "SHAPEFILES/Onondaga-ct", layer: "tl_2010_36067_tract10"
## with 140 features
## It has 12 fields
## Integer64 fields read as strings:  ALAND10 AWATER10
```

```
geojson_write( On_county, geometry="polygon", file="On_county.geojson" )
```

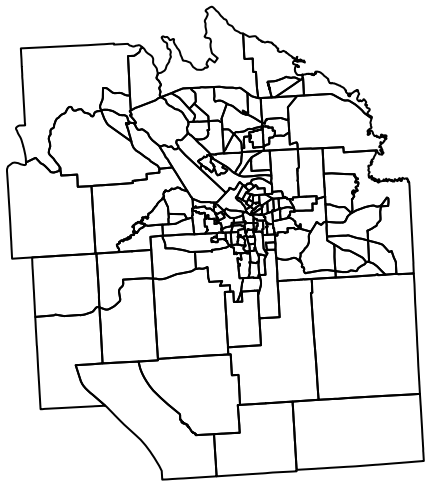
```
## Success! File is at On_county.geojson
```

```
## <geojson>
##   Path:       On_county.geojson
##   From class: SpatialPolygonsDataFrame
```

```
class(On_county)
```

```
## [1] "SpatialPolygonsDataFrame"
## attr(,"package")
## [1] "sp"
```

```
plot(On_county)
```



```
names(On_county)
```

```
##  [1] "STATEFP10"  "COUNTYFP10" "TRACTCE10"  "GEOID10"    "NAME10"
##  [6] "NAMELSAD10" "MTFCC10"    "FUNCSTAT10" "ALAND10"    "AWATER10"
## [11] "INTPTLAT10" "INTPTLON10"
```

```
Osw_county <- readOGR(dsn="SHAPEFILES/Oswego-ct", layer="tl_2010_36075_tract10")
```

```
## OGR data source with driver: ESRI Shapefile
## Source: "SHAPEFILES/Oswego-ct", layer: "tl_2010_36075_tract10"
## with 30 features
## It has 12 fields
## Integer64 fields read as strings:  ALAND10 AWATER10
```

```
geojson_write( Osw_county, geometry="polygon", file="Osw_county.geojson" )
```

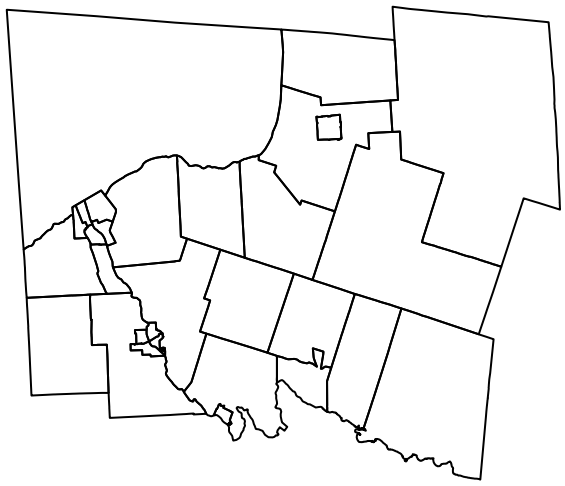
```
## Success! File is at Osw_county.geojson
```

```
## <geojson>
##   Path:       Osw_county.geojson
##   From class: SpatialPolygonsDataFrame
```

```
class(Osw_county)
```

```
## [1] "SpatialPolygonsDataFrame"
## attr(,"package")
## [1] "sp"
```

```
plot(Osw_county)
```



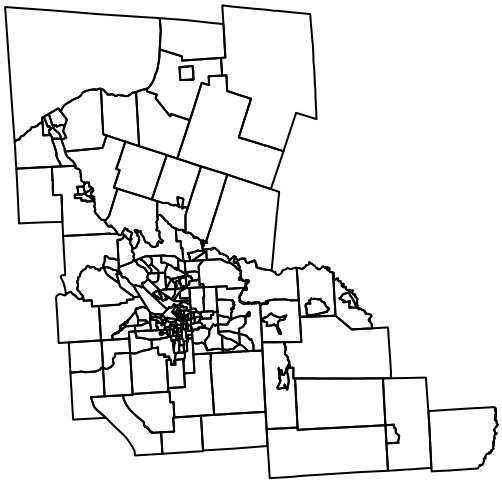
```
names(Osw_county)
```

```
##  [1] "STATEFP10"  "COUNTYFP10" "TRACTCE10"  "GEOID10"    "NAME10"
##  [6] "NAMELSAD10" "MTFCC10"    "FUNCSTAT10" "ALAND10"    "AWATER10"
## [11] "INTPTLAT10" "INTPTLON10"
```

Add Syracuse MSA census tract data to census tract shp files

```
library( acs )
library( tigris )
library( spatialEco )
library( sp )
library( leaflet )
library( raster )
```

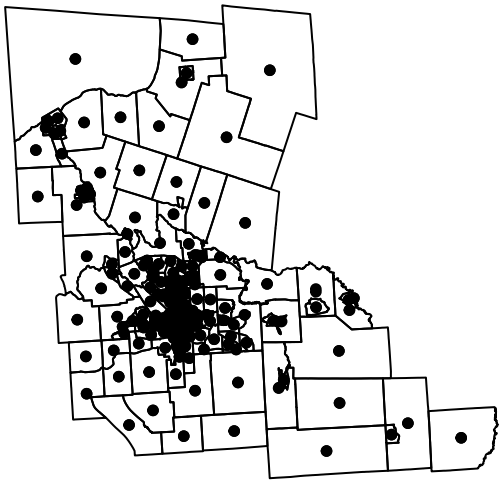
```
# drop NAs
Mad_county <- na.omit(Mad_county)
On_county <- na.omit(On_county)
Osw_county <- na.omit(Osw_county)

#Merge spatial files
syr_msa <- union(Mad_county, On_county)
syr_msa <- union(syr_msa, Osw_county)
plot( syr_msa )

# Merge data with MSA shp file
syr_merged <- geo_join(syr_msa, acs_2015_syr, "GEOID10", "GEOID")
plot( syr_merged )
```



Create centroid points for each census tract

```
library( rgeos )
```

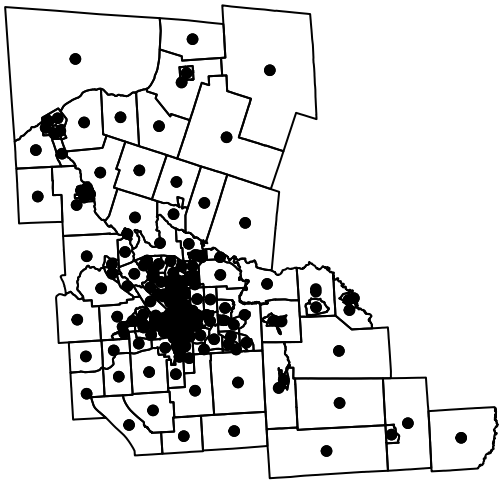
```
syr_merged_cen = gCentroid(syr_merged,byid=TRUE)
plot( syr_merged )
points(coordinates(syr_merged),pch=20)
points(syr_merged_cen,pch=20)
```



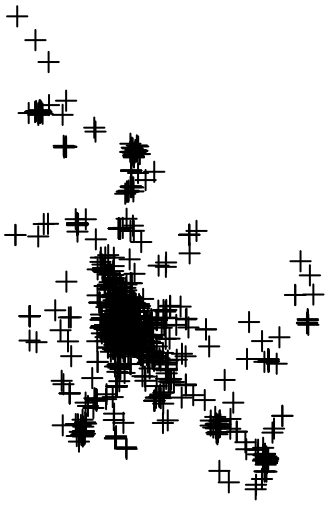
Spatial Projection difficulties

Point shp file for Nonprofit orgs does not show up with census shapefiles, even after placing both in the same projection.

```
## Correct spatial projection ##

proj4string(syr_merged) <- NA_character_
proj4string(syr_merged_cen) <- NA_character_
proj4string(syr_merged) <- CRS("+proj=longlat +datum=WGS84")
proj4string(syr_merged) <- CRS("+proj=longlat +datum=WGS84")

# Check
plot( syr_merged )
points(coordinates(syr_merged),pch=20)
points(syr_merged_cen,pch=20)
```



```
# Not sure why these points dont match up below...
plot( pov_orgs_gps_nona)
points(coordinates(syr_merged),pch=20)
points(syr_merged_cen,pch=20)
```



```
names( pov_orgs_gps_nona )
```

```
##  [1] "whole_address" "EIN"           "FIPS"          "NTEECC"
##  [5] "FILER"         "ZFILER"        "NAME"          "ADDRESS"
##  [9] "CITY"          "STATE"         "ZIP5"          "GEN"
## [13] "SUBSECCD"      "RULEDATE"      "FNDNCD"        "FRCD"
## [17] "ACCPER"        "ASSETS"        "INCOME"
```

Run Analysis on Orgs in wealthier / white places versus poverty / racially diverse neighborhoods

```
## Distance analysis b/w centroid centers and Nonprofit buffers

# neighbor analysis?

## Number of nonprofits location by census tract income or race
```

Final section: Account for all urban census tracts in 25 cities