Mapping with TIGER/FILE - Madison

Table of Contents

Note: this presentation relies on the work of Kyle Walker (https://www.github.com/walkerke) who developed the tigris and tidycensus package, it is adapted from the file linked at the bottom of this document.

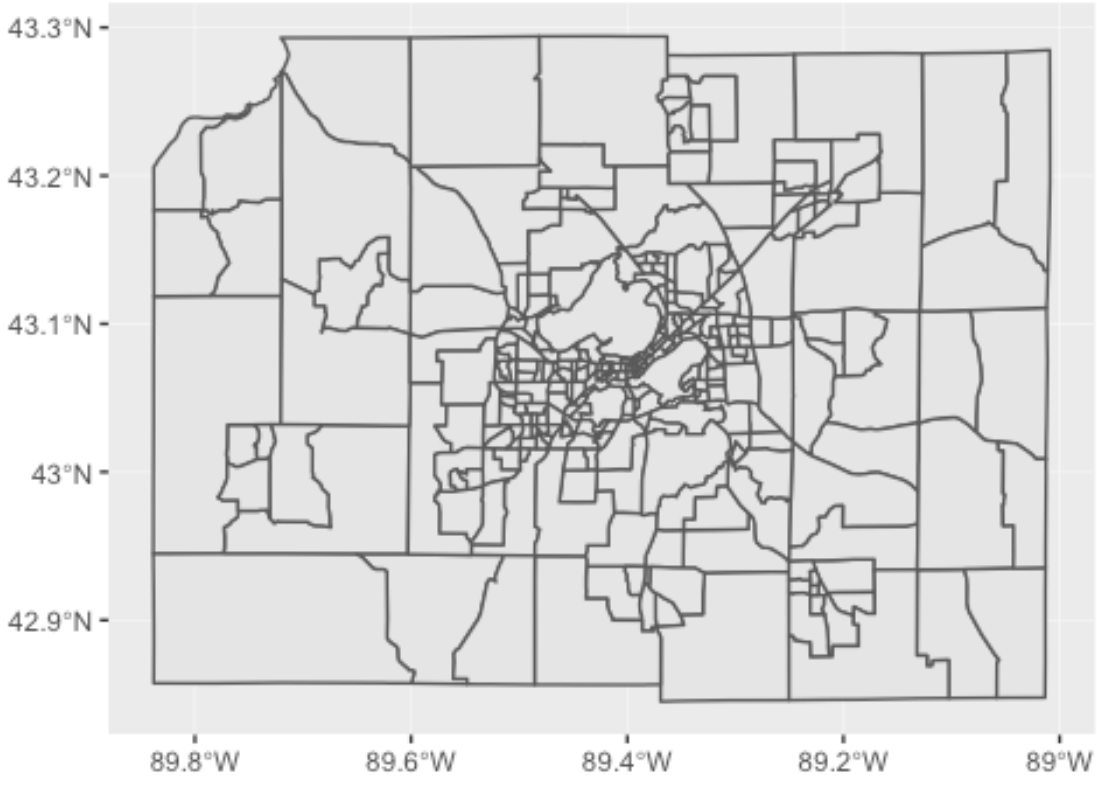
Loading packages

```
library(sf)
library(tidyverse)
library(tigris)
library(tidycensus)
options(tigris_class = "sf")
options(tigris_use_cache = TRUE)
```

get census block groups for Dane County

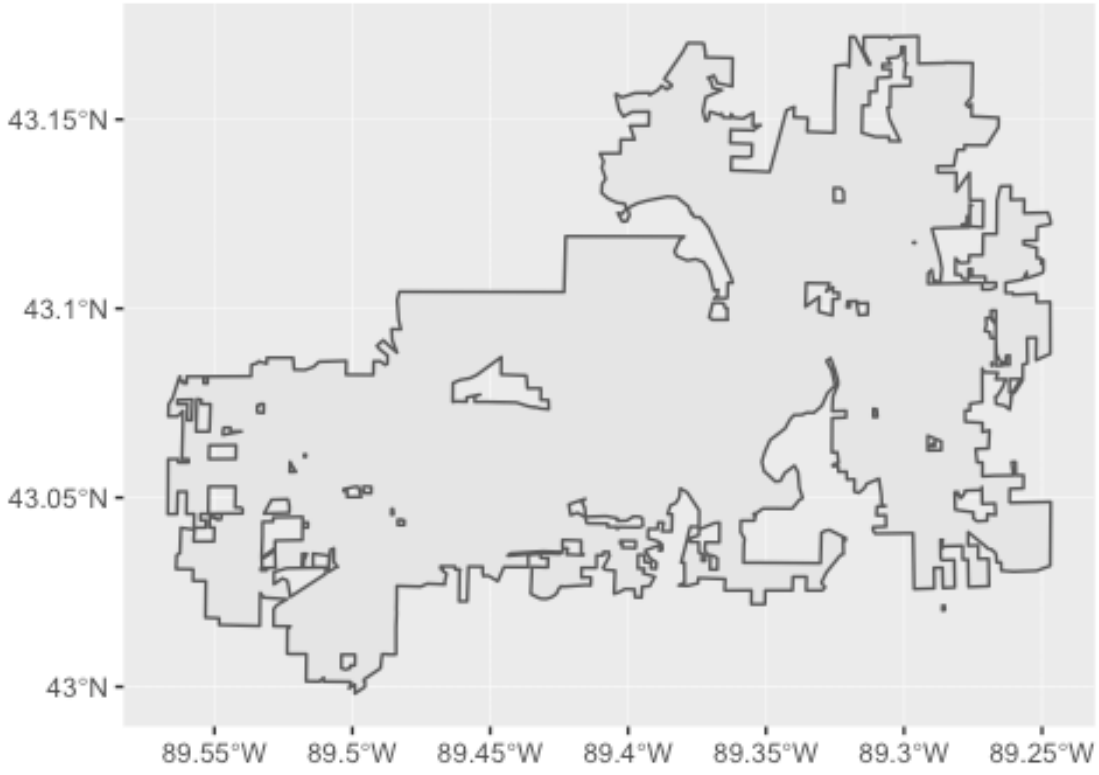
Block groups are subunits of census tracts made up from 39 blocks on average containing a population of 600 to 3,000 people. This is the smallest unit for which the Census will publish sample data.

```
daneblockg <- block_groups("WI", county = "Dane", cb= TRUE)
ggplot() +
  geom_sf(data = daneblockg)
```



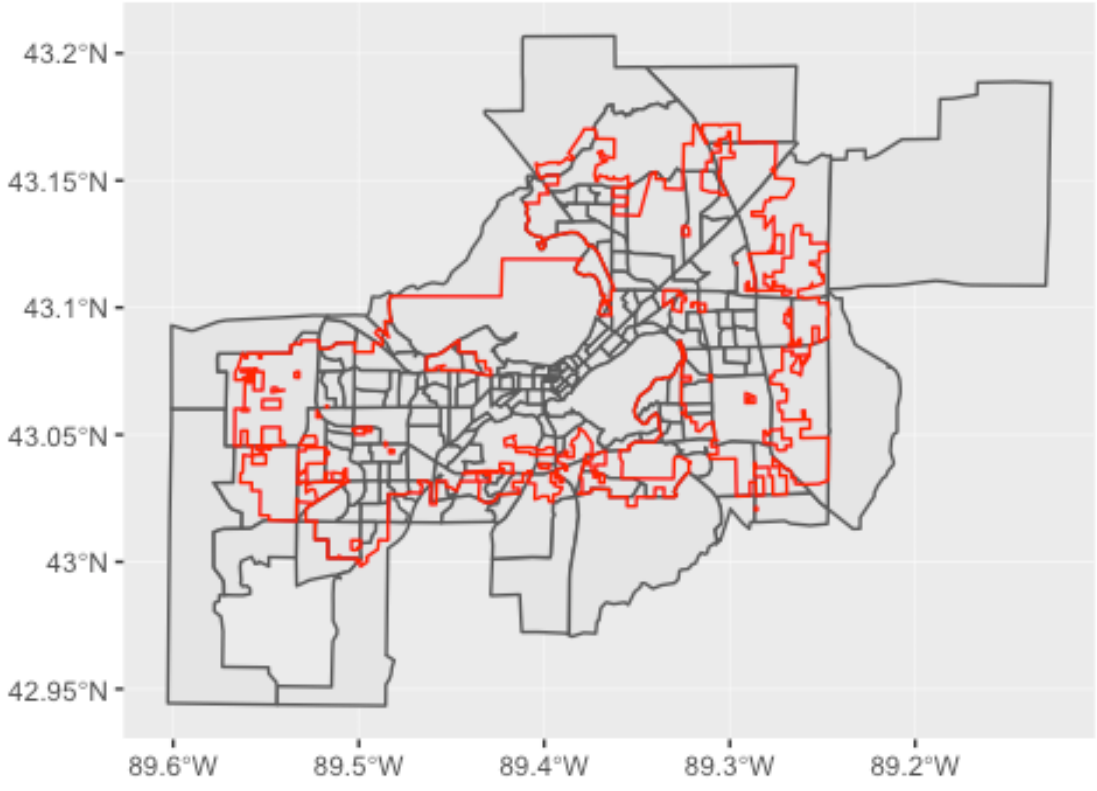
subset City of Madison and plot its shapefile.

```
cs <- county_subdivisions(55, county= 025, cb = TRUE)
madison <- cs[3,]
ggplot(madison) + geom_sf()
```



Subset the Census block groups to include only those that are included in the Cty of Madison geography -see above.

```
p1 <- daneblockg[madison,]

## although coordinates are longitude/latitude, st_intersects assumes
that they are planar

ggplot() +
  geom_sf(data = p1) +
  geom_sf(data = madison, fill = NA, color = "red")
```



The Census bock groups and the City of Madison limits do not fit neatly, with several block groups that include a part of Madison and a part of other county subdivisions

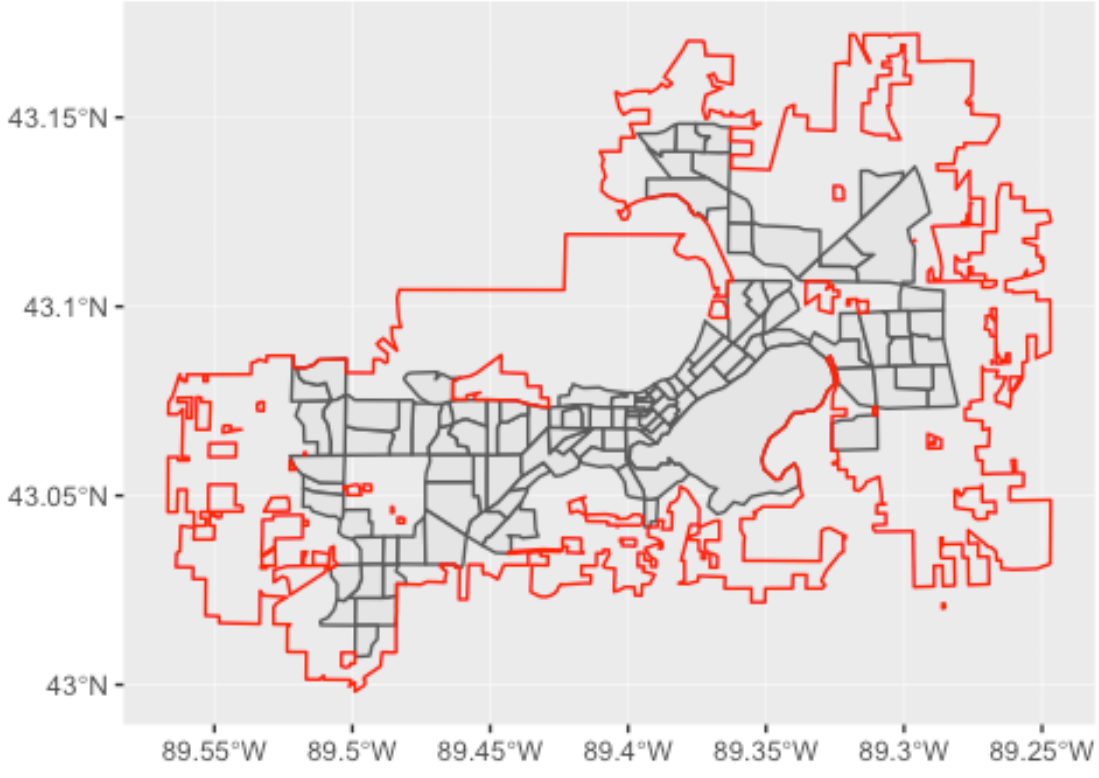
select only the tracts that are within city of Madison

We will try the method that we used with Census tracts and core based statistical areas and plot

we convert those that have integer(1) to a logical vector function by Kyle Walker

```
withinmad <- st_within(daneblockg, madison)

## although coordinates are longitude/latitude, st_within assumes that
they are planar

withinmad2 <- map_lgl(withinmad, function(x) {
  if (length(x) == 1) {
    return(TRUE)
```

```
  } else {
    return(FALSE)
  }
})
tractmad2 <- daneblockg[withinmad2,]

ggplot() +
  geom_sf(data = tractmad2) +
  geom_sf(data = madison, fill = NA, color = "red")
```



Using Tidycensus for to obtain a table for American Community Survey variables the acs package relies on the 2015 tables. Acs was unable to access newly added variables such as the ones under code B28002 that relate to internet access and computer ownership. The tidycensus package returns a data frame with statistical information along with a list of coordinates for the geography that is requested.

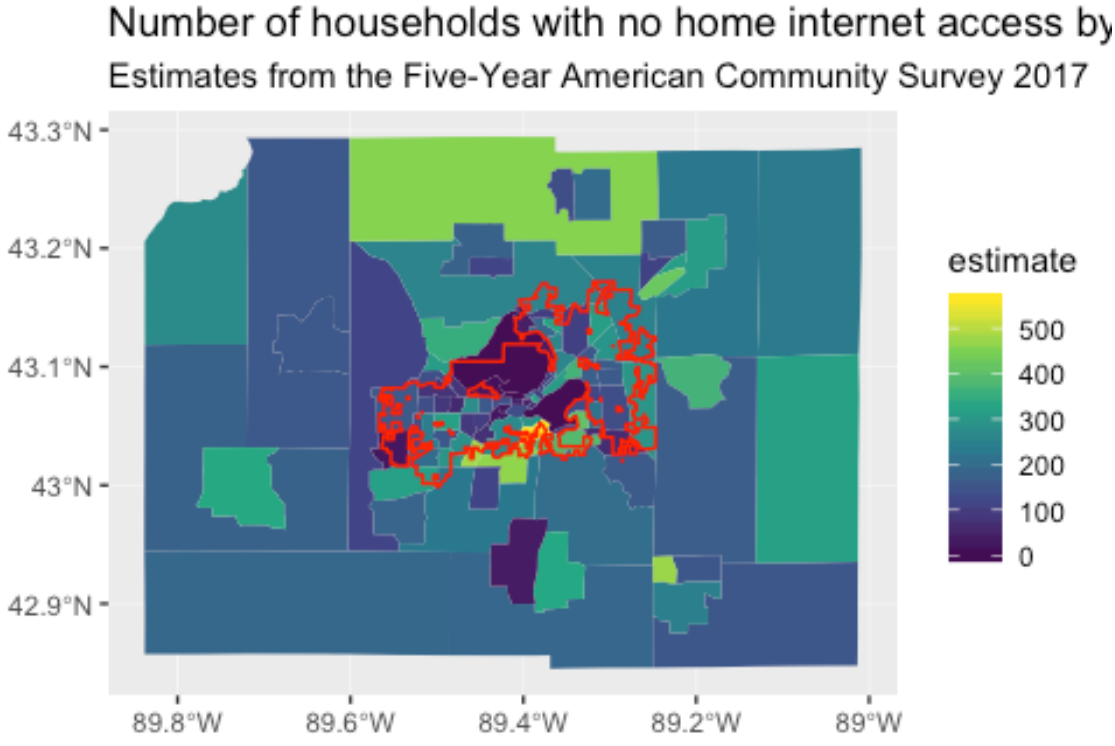
```
v17 <- load_variables(2017, "acs5", cache = TRUE)
```

after looking at the v17, and filtering down statistics by keyword to internet to select B28002_013 houses wihout internet in Dane County

```
dane.no.web <- get_acs(state = "WI", county = "dane", geography =
"tract",
                       variables = "B28002_013", geometry = TRUE)

## Getting data from the 2013-2017 5-year ACS

head(dane.no.web)

## Simple feature collection with 6 features and 5 fields
## geometry type:  MULTIPOLYGON
## dimension:      XY
## bbox:           xmin: -89.52327 ymin: 43.06049 xmax: -89.45112
ymax: 43.09459
## epsg (SRID):    4269
## proj4string:    +proj=longlat +datum=NAD83 +no_defs
##         GEOID                                          NAME   variable
## 1 55025000100    Census Tract 1, Dane County, Wisconsin B28002_013
## 2 55025000201 Census Tract 2.01, Dane County, Wisconsin B28002_013
## 3 55025000202 Census Tract 2.02, Dane County, Wisconsin B28002_013
## 4 55025000204 Census Tract 2.04, Dane County, Wisconsin B28002_013
## 5 55025000205 Census Tract 2.05, Dane County, Wisconsin B28002_013
## 6 55025000300    Census Tract 3, Dane County, Wisconsin B28002_013
##   estimate moe                       geometry
## 1       83  47 MULTIPOLYGON (((-89.48777 4...
## 2       59  37 MULTIPOLYGON (((-89.48651 4...
## 3       86  49 MULTIPOLYGON (((-89.50256 4...
## 4      133  77 MULTIPOLYGON (((-89.52327 4...
## 5       80  48 MULTIPOLYGON (((-89.52163 4...
## 6      276 116 MULTIPOLYGON (((-89.47234 4...
```

We can now plot the estimate column onto the map

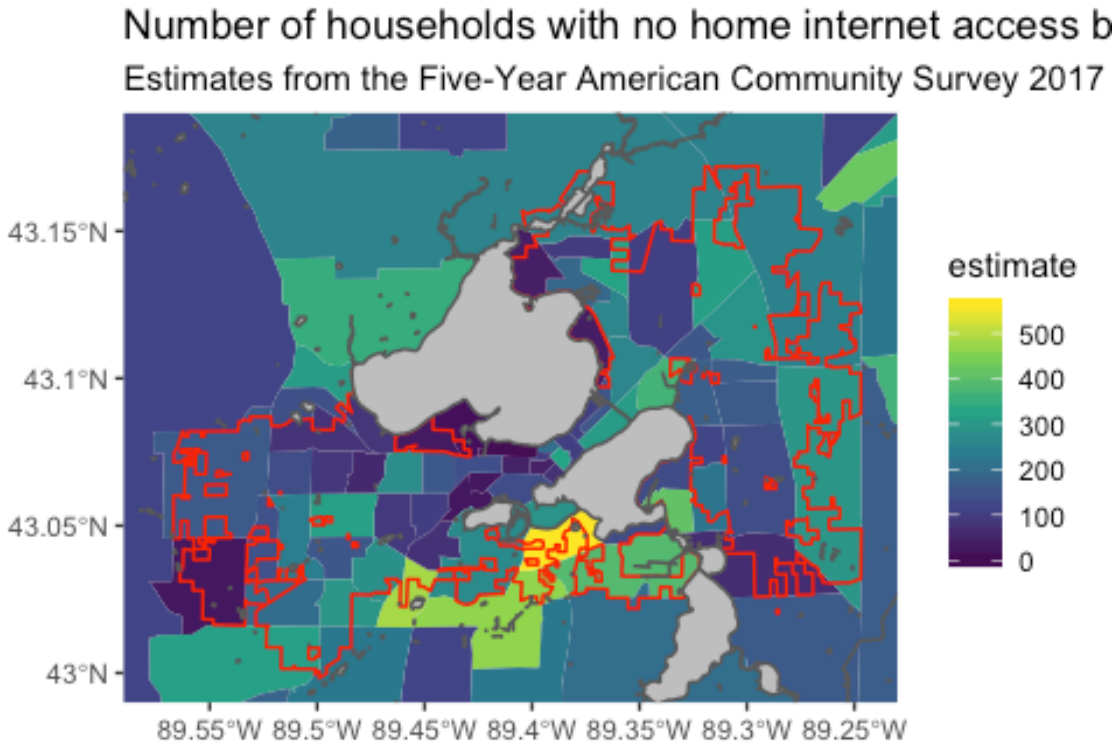
```
dane.no.web %>%
  ggplot(aes(fill = estimate)) +
  geom_sf(color = NA) +
  scale_fill_viridis_c(option = "viridis") +
  geom_sf(data = madison, fill = NA, color = "red") +
  labs(title = "Number of households with no home internet access by
Census tract", subtitle = "Estimates from the Five-Year American
Community Survey 2017")
```



For more control we can do some adjustments - center the coordinates around Madison (by using the *coord_sf* function in ggplot and add bodies of water so that those areas are not represented incorrectly as part of a statistic.

```
dane.water <- area_water("WI", county= 025)
```

```
dane.no.web %>%
  ggplot(aes(fill = estimate)) +
  geom_sf(color = NA) +
  scale_fill_viridis_c(option = "viridis") +
  geom_sf(data = madison, fill = NA, color = "red") +
  geom_sf(data = dane.water, fill = "gray") +
  labs(title = "Number of households with no home internet access by
Census tract", subtitle = "Estimates from the Five-Year American
Community Survey 2017") + coord_sf(xlim = c(-89.59, -89.23), ylim =
c(42.99, 43.19), expand = FALSE)
```



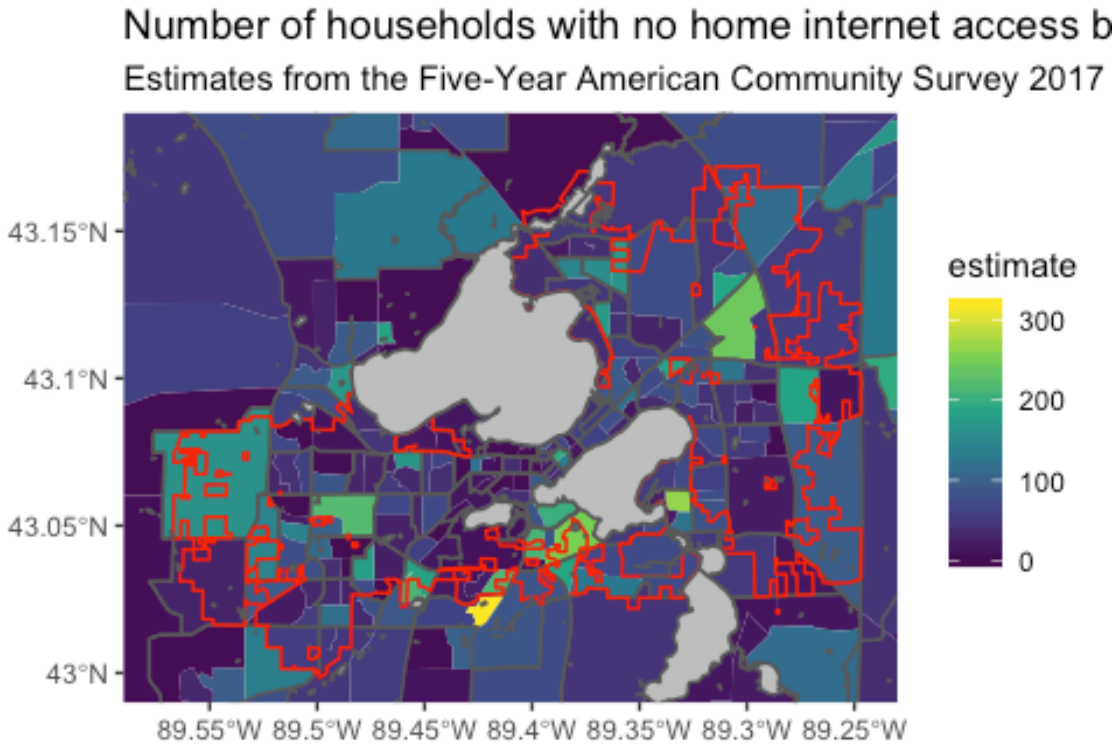
The same statistic, now at the block group level.

```
daneblockgroup.no.web <- get_acs(state = "WI", county = "dane",
geography = "block group", variables = "B28002_013", geometry = TRUE)
```

```
## Getting data from the 2013-2017 5-year ACS
```

We want to still graph the tracts but not fill them and limit those to Madison.

```
danetract <- tracts("WI", county = "Dane", cb= TRUE)
tractmad <- danetract[madison,]

## although coordinates are longitude/latitude, st_intersects assumes
that they are planar

daneblockgroup.no.web %>%
  ggplot(aes(fill = estimate)) +
  geom_sf(color = NA) +
  scale_fill_viridis_c(option = "viridis") +
  geom_sf(data = tractmad, fill = NA) +
  geom_sf(data = madison, fill = NA, color = "red") +
  geom_sf(data = dane.water, fill = "gray") +
  labs(title = "Number of households with no home internet access by
Census Block Group", subtitle = "Estimates from the Five-Year American
Community Survey 2017") +
  coord_sf(xlim = c(-89.59, -89.23), ylim = c(42.99, 43.19), expand =
FALSE)
```



for more information go to https://walkerke.github.io/2017/05/tigris-metros/