

police-data-eda

margaret

3/19/2021

```
library(tidyverse)

## -- Attaching packages ----- tidyverse 1.3.0 --

## v ggplot2 3.3.3     v purrr    0.3.4
## v tibble   3.1.0     v dplyr    1.0.5
## v tidyverse 1.1.3     v stringr  1.4.0
## v readr    1.4.0     vforcats  0.5.1

## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()   masks stats::lag()

library(sf)

## Linking to GEOS 3.8.1, GDAL 3.1.4, PROJ 6.3.1

police <- read_csv("../data/Raleigh_Police_Incidents_(NIBRS).csv")

## 
## -- Column specification -----
## cols(
##   .default = col_character(),
##   X = col_double(),
##   Y = col_double(),
##   OBJECTID = col_double(),
##   reported_year = col_double(),
##   reported_month = col_double(),
##   reported_day = col_double(),
##   reported_hour = col_double(),
##   latitude = col_double(),
##   longitude = col_double()
## )
## i Use `spec()` for the full column specifications.

police_shape <- st_read("../data/Raleigh_Police_Incidents_(NIBRS).shp")
```

```

## Reading layer 'Raleigh_Police_Incidents_(NIBRS)' from data source '/Users/margaretreed/Projects/Dathathon2021/police_incidents.shp'
## replacing null geometries with empty geometries
## Simple feature collection with 337780 features and 21 fields (with 86244 geometries empty)
## geometry type:  POINT
## dimension:      XY
## bbox:           xmin: -78.9848 ymin: 35.56671 xmax: -78.32503 ymax: 36.00489
## geographic CRS: WGS 84

nc <- st_read(system.file("shape/nc.shp", package = "sf"), quiet = TRUE)
wake <- nc %>% filter(NAME == "Wake")
zipcodes <- st_read("../Zip_Codes/Zip_Codes.shp")

## Reading layer 'Zip_Codes' from data source '/Users/margaretreed/Projects/Dathathon2021/Zip_Codes/Zip_Codes.shp'
## Simple feature collection with 67 features and 8 fields
## geometry type:  MULTIPOLYGON
## dimension:      XY
## bbox:           xmin: 2001465 ymin: 644117.9 xmax: 2221323 ymax: 847042
## projected CRS: NAD83 / North Carolina (ftUS)

sv <- st_read("../social_vuln/Overall_2014_Tracts.shp") %>%
  filter(STATE == "North Carolina",
         COUNTY == "Wake")

## Reading layer 'Overall_2014_Tracts' from data source '/Users/margaretreed/Projects/Dathathon2021/social_vuln/Overall_2014_Tracts.shp'
## Simple feature collection with 72842 features and 129 fields
## geometry type:  MULTIPOLYGON
## dimension:      XYZ
## bbox:           xmin: -179.1489 ymin: 18.91036 xmax: 179.7785 ymax: 71.36516
## z_range:        zmin: 0 zmax: 0
## geographic CRS: NAD83

census <- st_read("../Census_Tracts_2010/Census_Tracts_2010.shp")

## Reading layer 'Census_Tracts_2010' from data source '/Users/margaretreed/Projects/Dathathon2021/Census_Tracts_2010.shp'
## Simple feature collection with 187 features and 17 fields
## geometry type:  POLYGON
## dimension:      XY
## bbox:           xmin: 2001470 ymin: 643974.7 xmax: 2220895 ymax: 846791.5
## projected CRS: NAD83 / North Carolina (ftUS)

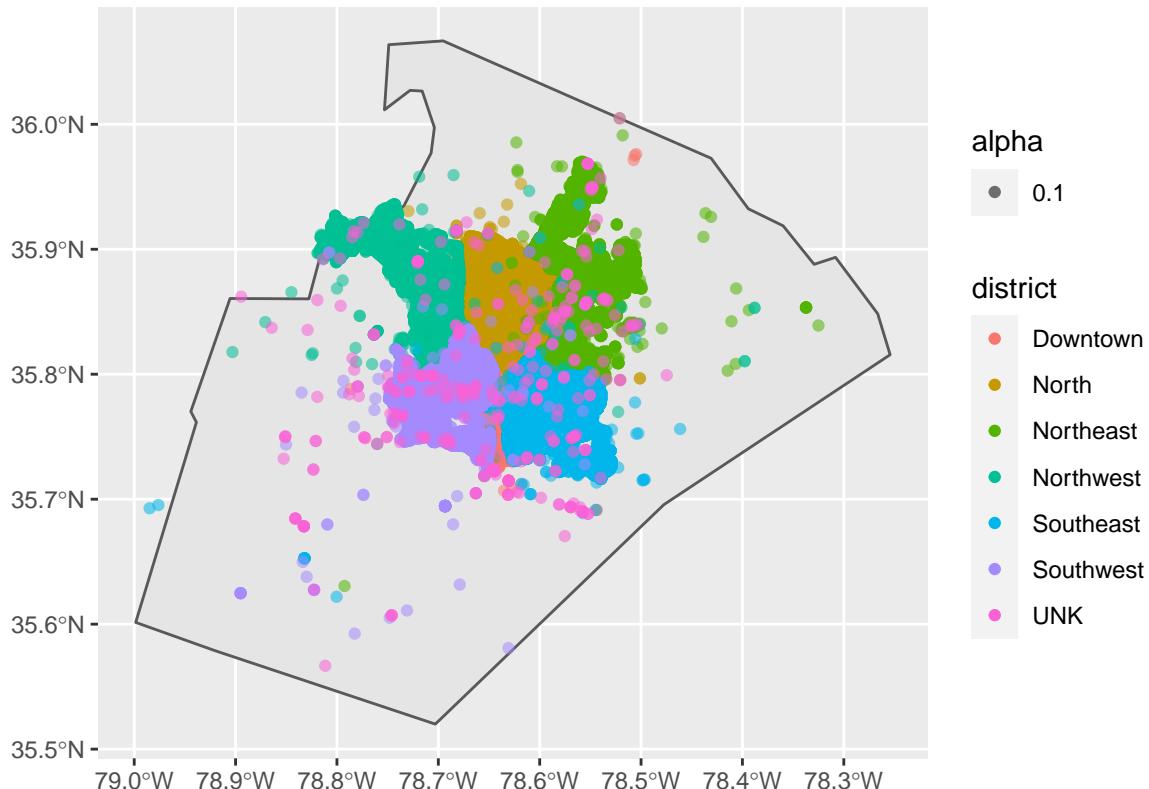
police_shape <- police_shape %>%
  filter(!st_is_empty(geometry))

nc <- st_transform(nc, crs = st_crs(police_shape))

police_shape %>%
  ggplot() +
  geom_sf(data = wake) +
  geom_sf(aes(alpha = 0.1, color = district)) +
  labs(title = "police inspections")

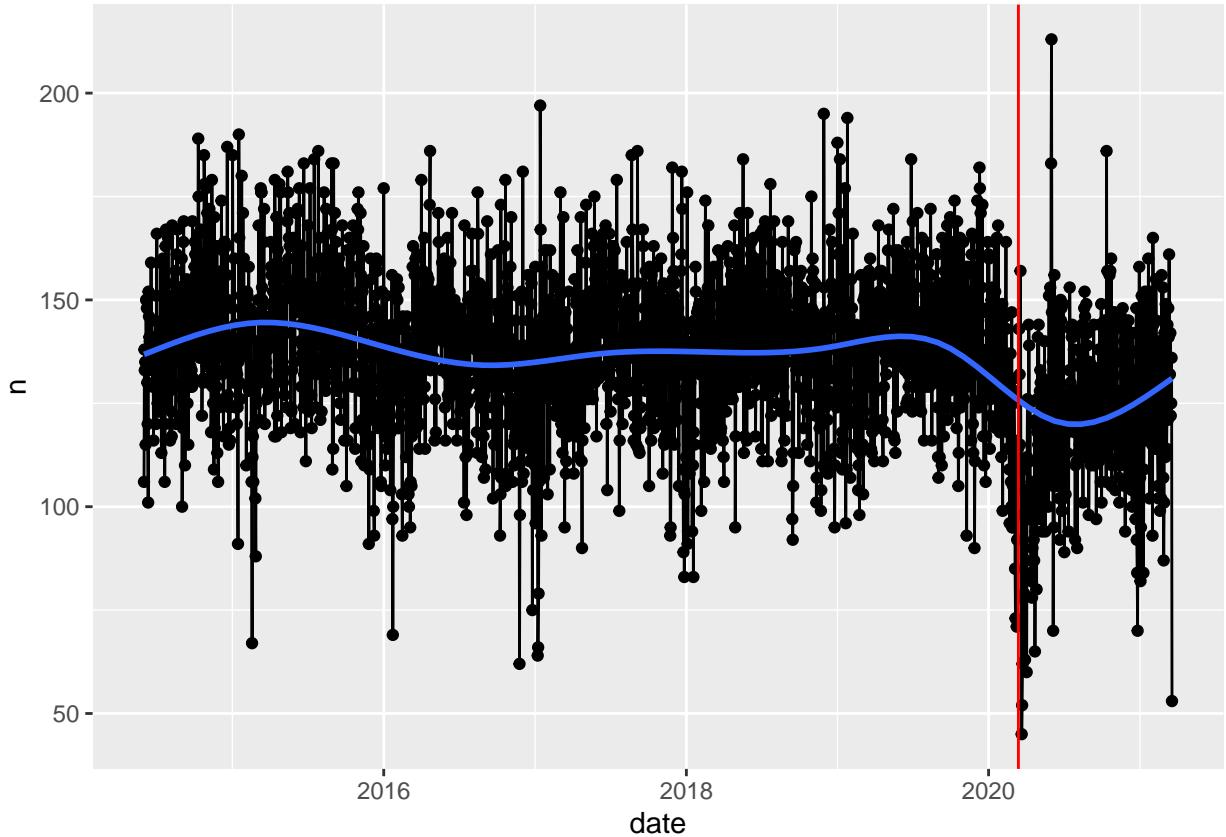
```

police inspections



```
police %>%
  mutate(date = as.Date(reported_date)) %>%
  count(date) %>%
  ggplot(aes(x=date, y=n)) +
  geom_line() +
  geom_point() +
  geom_smooth(se=F) +
  geom_vline(xintercept=as.Date("03/13/2020", format="%m/%d/%Y"), color = "red")
```

```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```



```

new_po <- police_shape %>%
  st_join(st_transform(census, st_crs(police_shape))) %>%
  filter(!is.na(TRACTCE10))

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

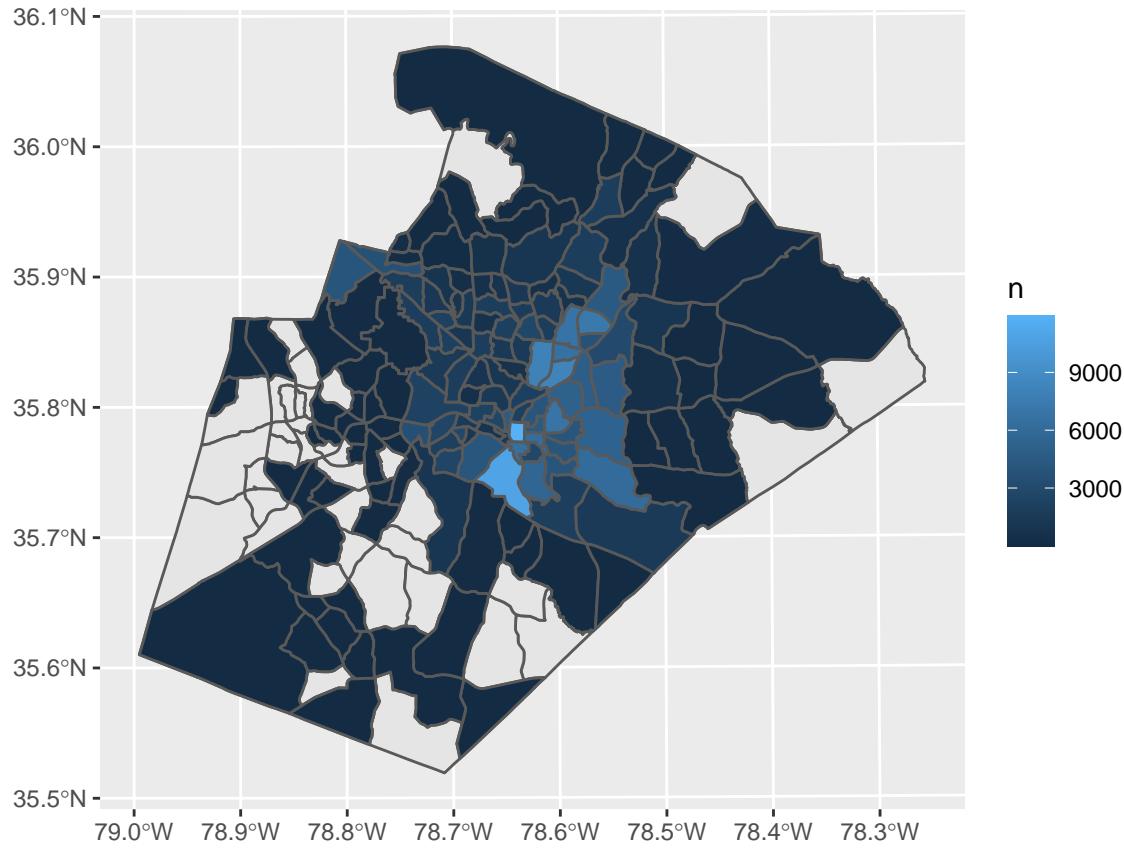
cts <- census %>%
  st_join(st_transform(police_shape, st_crs(census))) %>%
  filter(!is.na(TRACTCE10),
         !is.na(OBJECTID.y)) %>%
  count(TRACTCE10)

top_10 <- cts %>%
  arrange(desc(n)) %>%
  slice(1:10) %>%
  pull(TRACTCE10)

new_cts <- st_transform(cts %>% mutate(st_cast(., "MULTIPOLYGON")),
                        st_crs(sv))

ggplot(cts) +
  geom_sf(data = census) +
  geom_sf(aes(fill = n))

```



data codes: https://svi.cdc.gov/Documents/Data/2014_SVI_Data/SVI2014Documentation.pdf

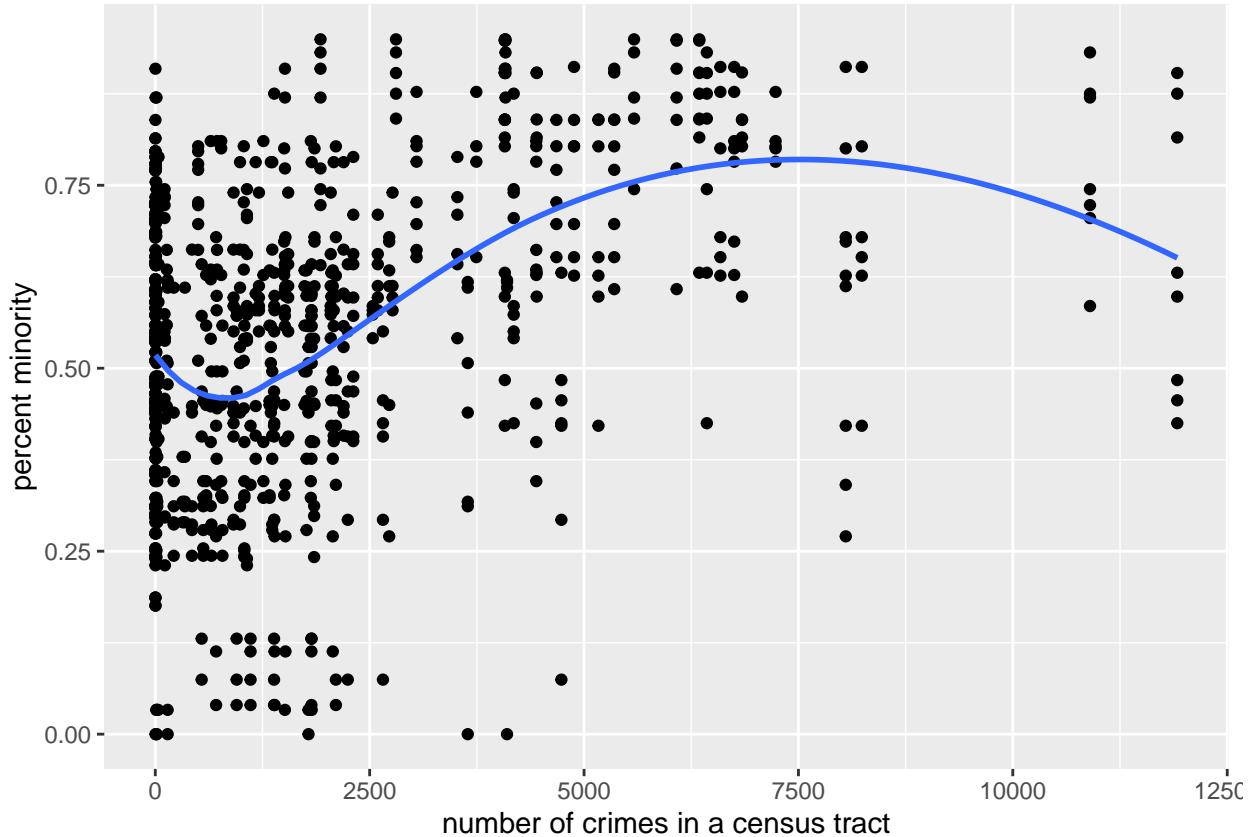
```
sv %>%
  st_zm(drop = TRUE, what = "ZM") %>%
  st_join(new_cts, join = st_overlaps) %>%
  ggplot(aes(x = n, y = EPL_MINRTY)) +
  geom_point() +
  geom_smooth(se = F) +
  labs(x = "number of crimes in a census tract", y = "percent minority")
```

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

## `geom_smooth()` using method = 'loess' and formula 'y ~ x'

## Warning: Removed 4 rows containing non-finite values (stat_smooth).

## Warning: Removed 4 rows containing missing values (geom_point).
```



```

sv %>%
  st_zm(drop = TRUE, what = "ZM") %>%
  st_join(new_cts, join = st_overlaps) %>%
  ggplot(aes(x = n, y = E_NOHSDP/E_TOTPOP)) +
  geom_point() +
  geom_smooth(se = F) +
  labs(x = "number of crimes in a census tract",
       y = "percent without highschool diploma")

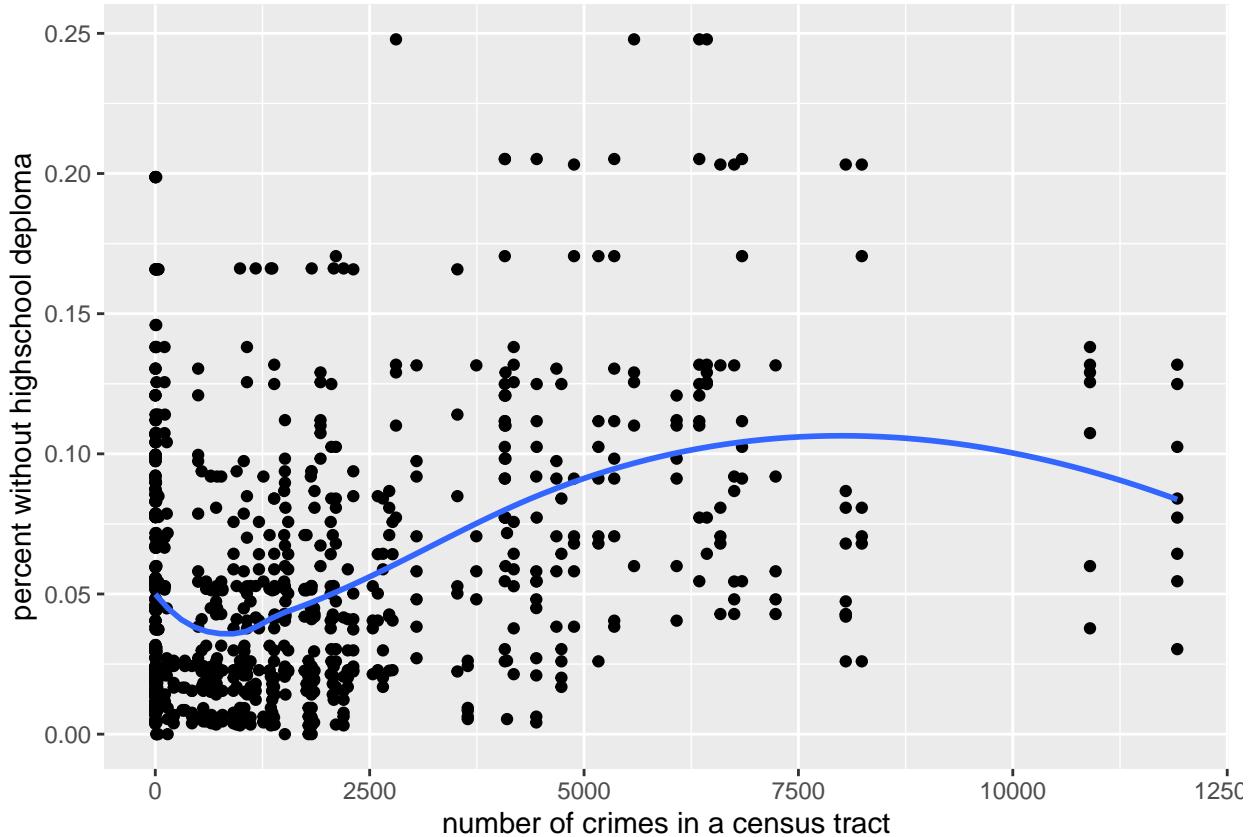
## although coordinates are longitude/latitude, st_overlaps assumes that they are planar

## `geom_smooth()` using method = 'loess' and formula 'y ~ x'

## Warning: Removed 11 rows containing non-finite values (stat_smooth).

## Warning: Removed 11 rows containing missing values (geom_point).

```



```

sv %>%
  st_zm(drop = TRUE, what = "ZM") %>%
  st_join(new_cts, join = st_overlaps) %>%
  ggplot(aes(x = n, y = E_UNEMP/E_TOTPOP)) +
  geom_point() +
  geom_smooth(se = F) +
  labs(x = "number of crimes in a census tract",
       y = "percent unemployed")

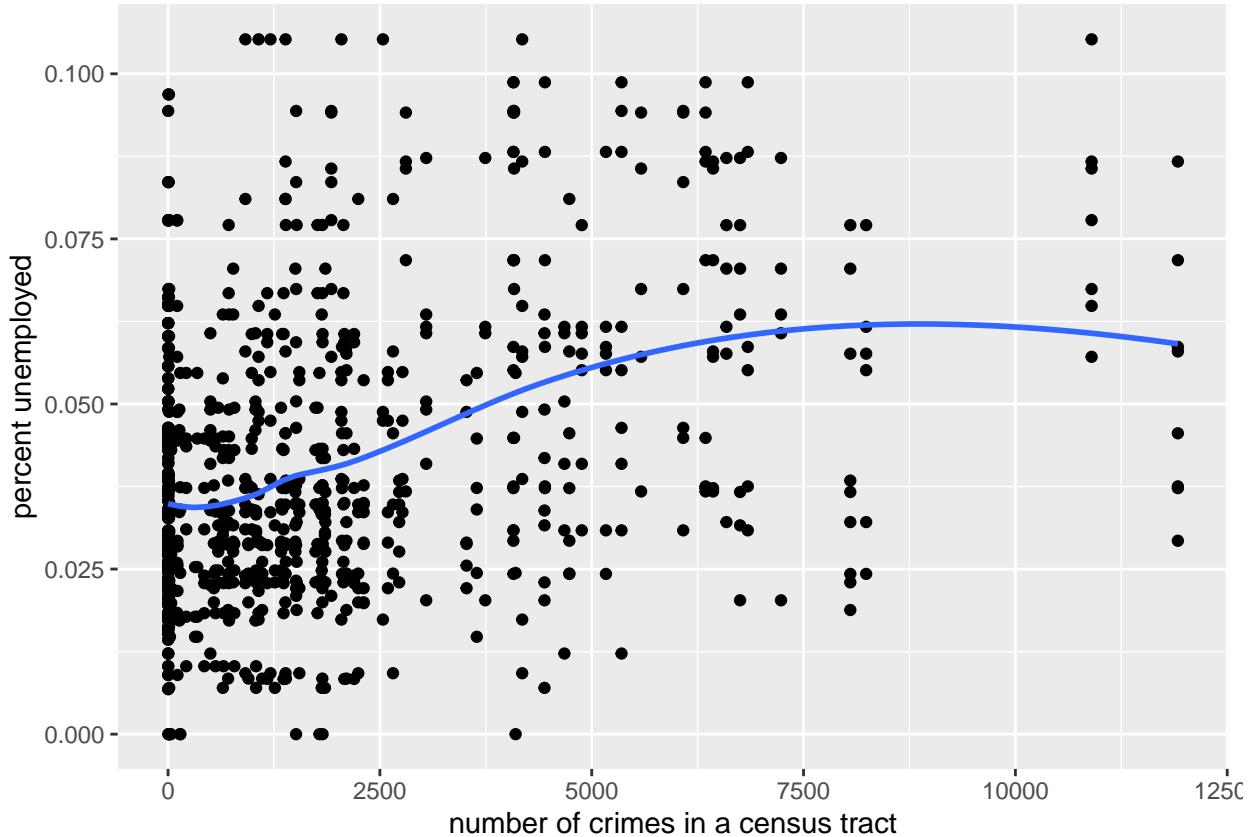
## although coordinates are longitude/latitude, st_overlaps assumes that they are planar

## `geom_smooth()` using method = 'loess' and formula 'y ~ x'

## Warning: Removed 11 rows containing non-finite values (stat_smooth).

## Warning: Removed 11 rows containing missing values (geom_point).

```



```

sv %>%
  st_zm(drop = TRUE, what = "ZM") %>%
  st_join(new_cts, join = st_overlaps) %>%
  ggplot(aes(x = n, y = E_POV/E_TOTPOP)) +
  geom_point() +
  geom_smooth(se = F) +
  labs(x = "number of crimes in a census tract",
       y = "percent under the poverty line")

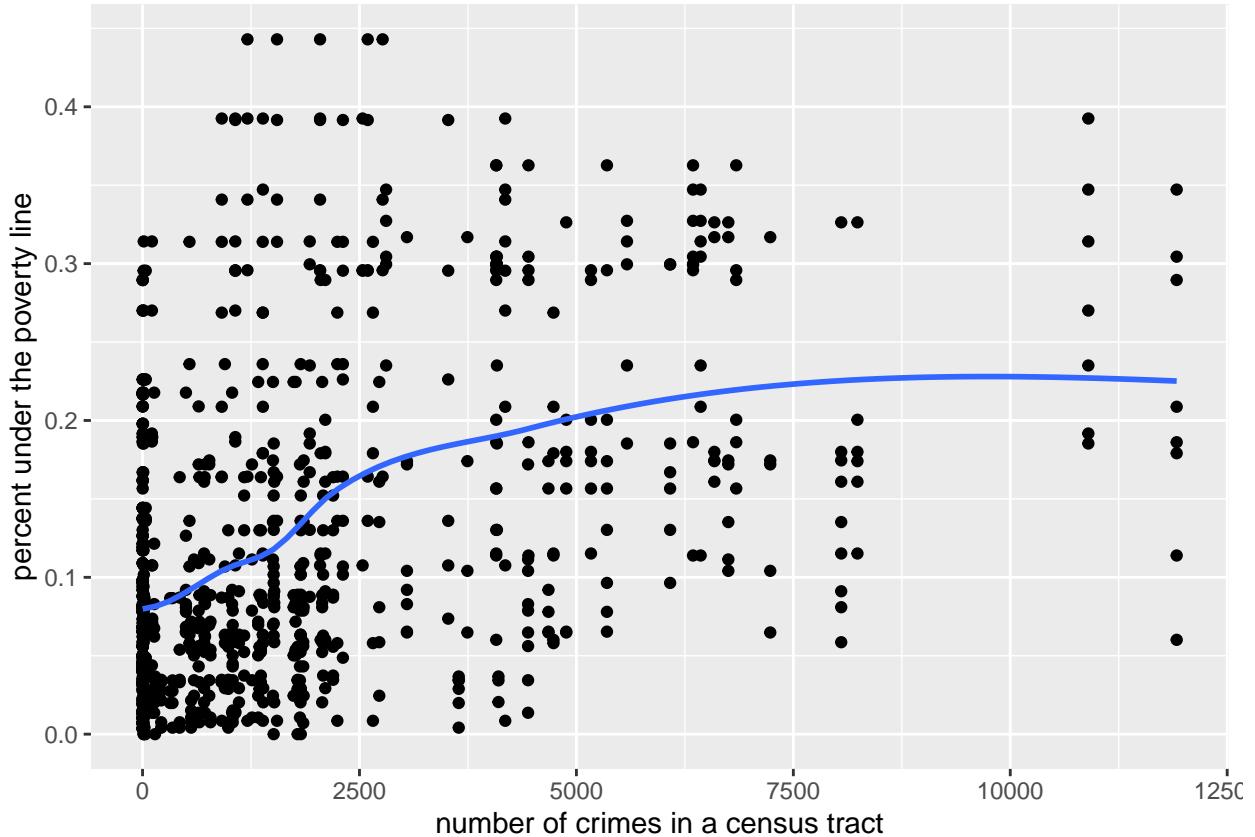
## although coordinates are longitude/latitude, st_overlaps assumes that they are planar

## `geom_smooth()` using method = 'loess' and formula 'y ~ x'

## Warning: Removed 11 rows containing non-finite values (stat_smooth).

## Warning: Removed 11 rows containing missing values (geom_point).

```



```

sv %>%
  st_zm(drop = TRUE, what = "ZM") %>%
  st_join(new_cts, join = st_overlaps) %>%
  ggplot(aes(x = n, y = E_AGE17/E_TOTPOP)) +
  geom_point() +
  geom_smooth(se = F) +
  labs(x = "number of crimes in a census tract",
       y = "percent under age 17")

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

## `geom_smooth()` using method = 'loess' and formula 'y ~ x'

## Warning: Removed 11 rows containing non-finite values (stat_smooth).

## Warning: Removed 11 rows containing missing values (geom_point).

```

