Trends in Maryland Crime Rates

Kamil Ismailov

5/8/2020

1. Maryland crime rate data

Crime is an extremely negative mass natural and deeply structured social phenomenon, distributed in time and space, characteristic of all currently known social systems and associated with a huge number of other social phenomena. The identification of the dependence of crimes on the socio-economic problems existing in society can be a driver of reducing the criminological situation in the country.

According to Wikipedia, Baltimore city of the Maryland state in third place in the amount of total violent crimes per 100,000 people per year. Cause of this high crime rate in cities like Baltimore, it is very important for policymakers to understand how crime rate changes across time and how population affects these changes. This information might help them to reform their policies more efficiently. The purpose of this project is to study the effect of crime rates changing from place to place and from time to time not only in Baltimore City but also in all counties of the Maryland state.

To retrieve this kind of information we perform hierarchical modeling to capture and explore crime statistics collected by the Maryland Statistical Analysis Center to see if there is any linear trend in violent crime across the state between 1975 and 2017. By the end of the project, we will compare populations and trends in crime rates of each county.

2. Data preparation and exploratory analysis

As a dataset contains a lot of diverse information, it is important to clean our data first and select variables that need analysis. Thus, the dataset which we will analyze contains counties, year, population, and crime rate.

```
# Load the packages
library(tidyverse)

# load the crime data
crime_raw <- read_csv(url("https://opendata.maryland.gov/api/views/jwfa-
fdxs/rows.csv"))

# Select and mutate columns the needed columns
crime_use <- crime_raw %>%
    select(JURISDICTION, YEAR, POPULATION, crime_rate = `VIOLENT CRIME RATE
PER 100,000 PEOPLE`)
```

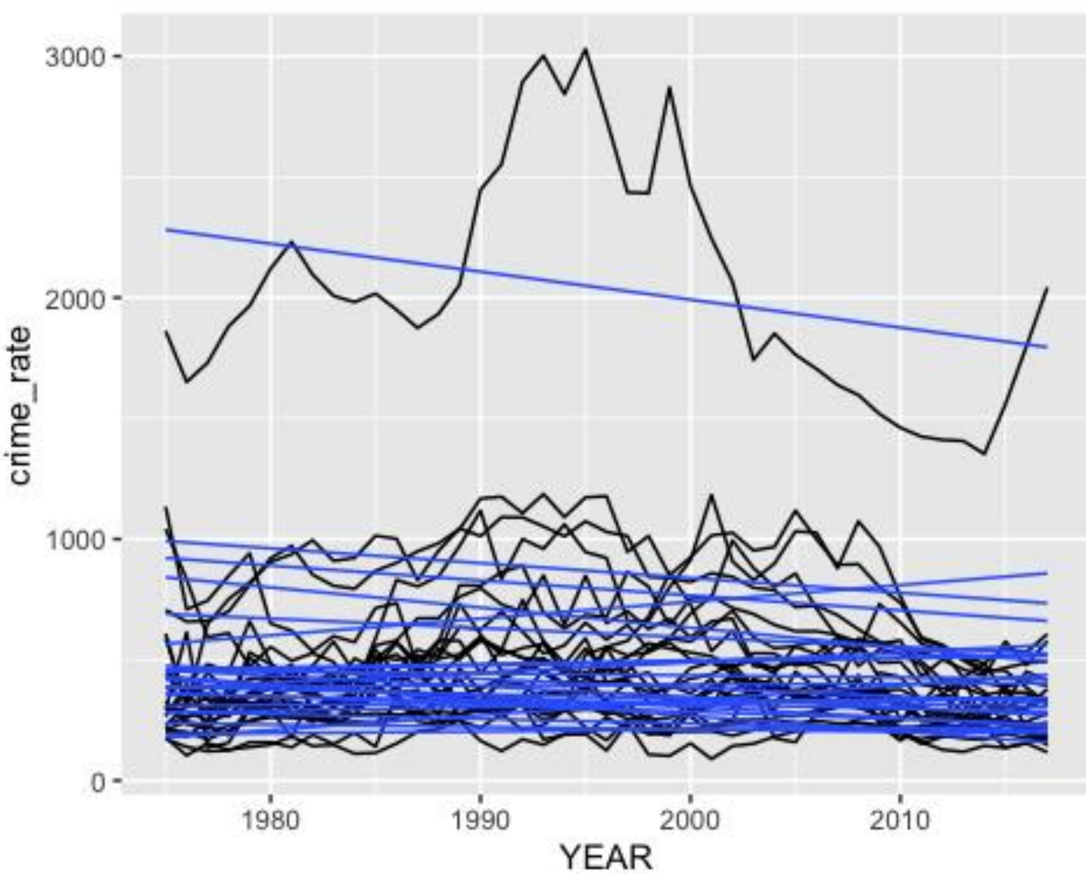
```r
head(crime_use)
```

```
## # A tibble: 6 x 4
##   JURISDICTION     YEAR POPULATION crime_rate
##   <chr>           <dbl>      <dbl>      <dbl>
## 1 Allegany County  1975      79655       178.
## 2 Allegany County  1976      83923       104.
## 3 Allegany County  1977      82102       155.
## 4 Allegany County  1978      79966       128.
## 5 Allegany County  1979      79721       138 
## 6 Allegany County  1980      80461       148.
```

Before we start to build a regression model, it is better to explore the data first to look at any possible errors, outliers, or other interesting observations. There is a graph of the crime rate over time with linear trend lines grouped by counties.

```r
# Plot the data as lines and linear trend lines
ggplot(crime_use, aes(x = YEAR, y = crime_rate, group = JURISDICTION)) +
  geom_line() +
  stat_smooth(method = "lm", se = FALSE, size = 0.5)
```

```
## `geom_smooth()` using formula 'y ~ x'
```



According to the graph above, we see that the crime rate of Baltimore city significantly higher than the other counties. Nevertheless, the trend line of Baltimore City is decreasing but it doesn't tell us anything now.

Considering the regression model works more efficient when the intercept is close to zero, it is preferable to start counting the year variable from zero. Therefore, the re-scaling of the year variable allows us to avoid the failure of the model converge.

```
# Mutate data to create another year column, YEAR_2
crime_use <-
  crime_use %>% mutate(YEAR_2 = YEAR - min(YEAR))

head(crime_use)

## # A tibble: 6 x 5
##   JURISDICTION     YEAR POPULATION crime_rate YEAR_2
##   <chr>           <dbl>      <dbl>      <dbl>  <dbl>
## 1 Allegany County  1975      79655       178.      0
## 2 Allegany County  1976      83923       104.      1
## 3 Allegany County  1977      82102       155.      2
## 4 Allegany County  1978      79966       128.      3
## 5 Allegany County  1979      79721       138       4
## 6 Allegany County  1980      80461       148.      5
```

3. Linear mixed-effects regression model (LMER)

In this dataset, a crime rate variable nested within counties. They, in turn, nested within year variable. These nested data represents a hierarchical multi-level relationship. Therefore, a simple linear regression model is not suitable for this type of data. To build a hierarchical multi-level model, we can use ***lmerTest*** package in R, which also known as a linear mixed-effects regression. ***Lmer()*** function on this package use the same syntax as a ***lm()*** function. The main difference is that a linear mixed-effects regression model requires a random-effect argument:

```
lmer(y ~ x + (1|randomGroup), data = myData)
```

In this formula a quantitative response **y** (Crime_rate) is predicted by a fixed-effect slope **x** (Year) and a random-effect intercept **randomGroup** (Jurisdiction). Typically, it's better to fill full a random-effect structure by varying both intercept and slope. That assumes they correlate within each county for the random-effect estimates. Thus, *x* also may be included in a random-effect slope. As a result, our model is:

```
# Load the lmerTest package
library(lmerTest)

# Build a lmer model
lmer_crime <- lmer(crime_rate ~ YEAR_2 + (YEAR_2|JURISDICTION), crime_use)
```

Using this formula, we try to predict the trend in violent crime rate in the entire state, which sets as a fixed-effect slope, and for each county, which sets as a random-effect slope.

Considering each county as a random-effect, we infer that each county's trend comes from the distribution of the whole state.

4. Model assessment and output interpretations

```
summary(lmer_crime)

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: crime_rate ~ YEAR_2 + (YEAR_2 | JURISDICTION)
##    Data: crime_use
##
## REML criterion at convergence: 13461
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -4.4418 -0.4827 -0.0538  0.4393  6.4565
##
## Random effects:
##  Groups       Name        Variance  Std.Dev. Corr
##  JURISDICTION (Intercept) 179502.58 423.677
##               YEAR_2          15.93   3.991  -0.68
##  Residual                  23323.87 152.722
## Number of obs: 1032, groups:  JURISDICTION, 24
##
## Fixed effects:
##             Estimate Std. Error       df t value Pr(>|t|)
## (Intercept) 533.5250    86.9862  23.3456   6.133 2.77e-06 ***
## YEAR_2       -1.7534     0.9003  23.1973  -1.948   0.0637 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##        (Intr)
## YEAR_2 -0.655
## optimizer (nloptwrap) convergence code: 0 (OK)
## Model failed to converge with max|grad| = 0.122593 (tol = 0.002, component
1)
```

According to the summary statistics, the p-value of the predictor variable is 0.0637, which infers that we cannot reject the alternative hypothesis that a relationship between a crime rate and the year variable is statistically significant. The estimated coefficient for the predictor variable of the model ≈ -1.75. This assumes that the crime rate approximately declines 1,75 points each year in the entire state.

Since summary statistics give an only estimated slope of the average crime rate in Maryland, we can use ***fixef()*** and ***ranef()*** functions to retrieve coefficients of fixed-effect (state) and random-effect slopes for each county respectively.

```
## [1] **** Fixed-effects ****

## (Intercept)      YEAR_2
##  533.524982   -1.753395

## [1] **** Random-effects ****

## $JURISDICTION
##                        (Intercept)      YEAR_2
## Allegany County         -321.77111  6.29215891
## Anne Arundel County     -121.33061  4.55757223
## Baltimore City          1738.35225 -9.65806951
## Baltimore County         373.72965 -3.79100745
## Calvert County          -133.35748 -1.84378674
## Caroline County         -138.10934  1.76573896
## Carroll County          -299.78264  0.97997681
## Cecil County             -64.23568  2.45909856
## Charles County           -56.95240  2.43959271
## Dorchester County        148.83885 -1.78138482
## Frederick County         -69.97837 -1.28007678
## Garrett County          -336.46011  2.30261214
## Harford County          -180.64173  0.30882399
## Howard County           -131.18086 -3.30700570
## Kent County             -216.69378  2.19563676
## Montgomery County       -256.22851  0.21898957
## Prince George's County   446.59661 -3.84951664
## Queen Anne's County     -182.13409 -0.74204521
## Somerset County          -73.98743  0.77597996
## St. Mary's County       -151.87616  0.09036298
## Talbot County            -93.10705 -1.83118389
## Washington County       -240.49311  2.65623111
## Wicomico County           80.53461  6.28718998
## Worcester County         280.26846 -5.24588796
##
## with conditional variances for "JURISDICTION"
```

To implement our model to each county, we need to calculate the difference between the county crime rate and the state average crime rate, which is also called a mixed-effect estimated slope. We simply add coefficients of fixed-effect to each random-effect coefficient. To this data frame, we will add the population of each county of the latest year to make a further comparison.

```
# Add the fixed-effect to the random-effect and save as county_slopes
county_slopes <- fixef(lmer_crime)["YEAR_2"] +
ranef(lmer_crime)$JURISDICTION["YEAR_2"]

# Add a new column with county names and population
pop_2017 <- crime_use %>% select(POPULATION,YEAR) %>% filter (YEAR == 2017)
```

```r
county_slopes <-
  county_slopes %>% rownames_to_column("county") %>% mutate(POPULATION =
pop_2017$POPULATION)

head(county_slopes)
```

```
##                county       YEAR_2 POPULATION
## 1     Allegany County   4.53876366      71616
## 2 Anne Arundel County   2.80417699     572990
## 3      Baltimore City -11.41146475     613217
## 4    Baltimore County  -5.54440270     834727
## 5      Calvert County  -3.59718199      91587
## 6     Caroline County   0.01234372      32795
```

5. Maryland crime rate map visualization

It is difficult to observe trends and make quick comparisons having a bunch of numbers. Especially for policy and lawmakers who want a concise and interpretable report. The visual representation is the best option to understand the outcomes. In this case, we use *usmap* package to apply our results to the map of the Maryland state. The *usmap* package has already had data with names and coordinates of all counties. We only need to add our crime trend data and use *ggplot* package for visualization.

```r
# Load usmap package
library(usmap)

# load and filter map data
county_map <- us_map(regions = "counties", include = "MD")
head(county_map)
```

```
##             x         y order  hole piece   group  fips abbr     full
## 22297 1775008 -387237.8 22297 FALSE     1 24001.1 24001   MD Maryland
## 22298 1779953 -357902.4 22298 FALSE     1 24001.1 24001   MD Maryland
## 22299 1789820 -355286.0 22299 FALSE     1 24001.1 24001   MD Maryland
## 22300 1796777 -353431.8 22300 FALSE     1 24001.1 24001   MD Maryland
## 22301 1824941 -345915.7 22301 FALSE     1 24001.1 24001   MD Maryland
## 22302 1828029 -345083.7 22302 FALSE     1 24001.1 24001   MD Maryland
##                county
## 22297 Allegany County
## 22298 Allegany County
## 22299 Allegany County
## 22300 Allegany County
## 22301 Allegany County
## 22302 Allegany County
```

Data merging requires additional processes to check and find misspelling or differences in county's names.

```r
# See which counties are not in both datasets
county_slopes %>% anti_join(county_map, by = "county")

##            county    YEAR_2 POPULATION
## 1 Baltimore City -11.41146     613217

# Rename crime_names county
county_slopes  <- county_slopes  %>%
  mutate(county = ifelse(county == "Baltimore City", "Baltimore city",
county))

# Merge the map and slope data frames
both_data <- full_join(county_map,county_slopes)

## Joining, by = "county"
```

The plotting also requires some esthetical adjustments to make our map more attractive and understandable. Finally, we create the same map for the population to make further comparisons.

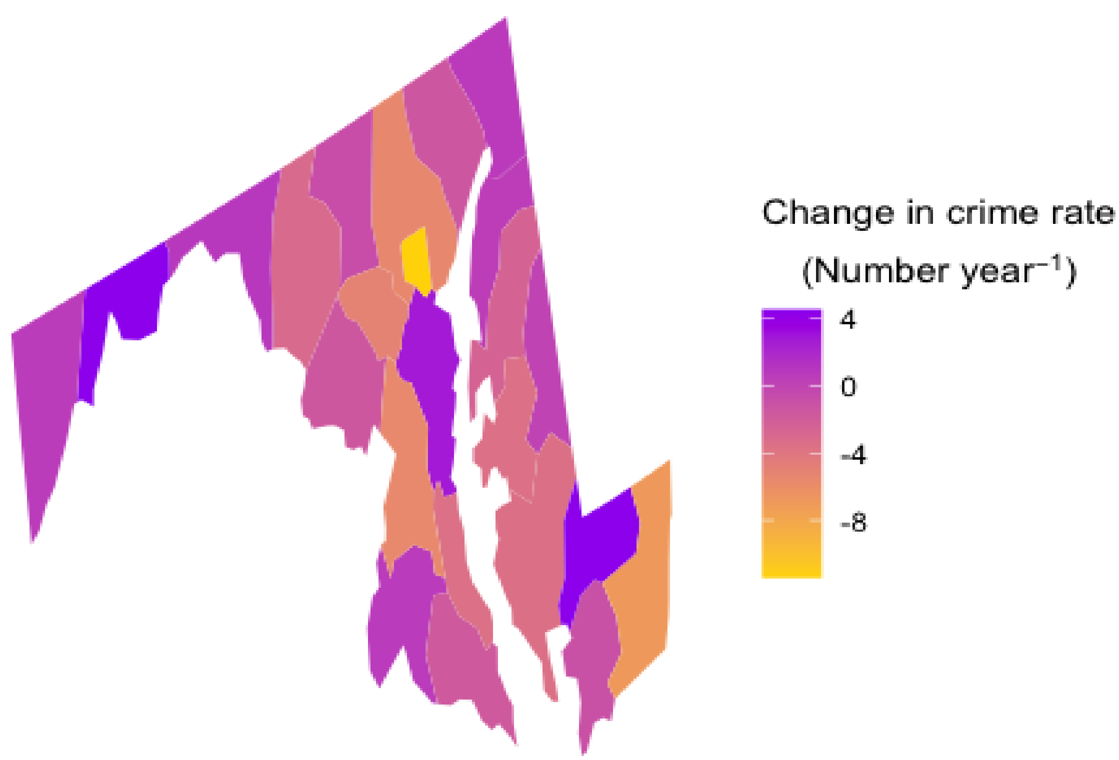
```r
#Plot the crime rate map
crime_map <-
  ggplot(both_data, aes(x,y, group = county, fill = YEAR_2)) +
  geom_polygon() +
  scale_fill_continuous(name = expression(atop("Change in crime
rate","(Number year"^-1*")")),
                        low = "gold", high = "purple")

# Plot options
options(repr.plot.width=10, repr.plot.height=5)

# Polish figure
crime_map_final <- crime_map +
  theme_minimal() + xlab("") + ylab("") +
  theme(axis.line = element_blank(), axis.text = element_blank(),
        panel.grid.major = element_blank(), panel.grid.minor =
element_blank(),
        panel.border = element_blank(), panel.background = element_blank())

# Plot the population map
pop_map <-
  ggplot(both_data, aes(x,y, group = county, fill = POPULATION)) +
  geom_polygon() +
  scale_fill_continuous(name = expression(atop("Population")),
                        low = "skyblue", high = "blue")

# Plot options
options(repr.plot.width=10, repr.plot.height=5)
```

```
# Polish figure
pop_map_final <- pop_map +
  theme_minimal() + xlab("") + ylab("") +
  theme(axis.line = element_blank(), axis.text = element_blank(),
        panel.grid.major = element_blank(), panel.grid.minor =
element_blank(),
        panel.border = element_blank(), panel.background = element_blank())
```

6. Populations and crime rates comparison

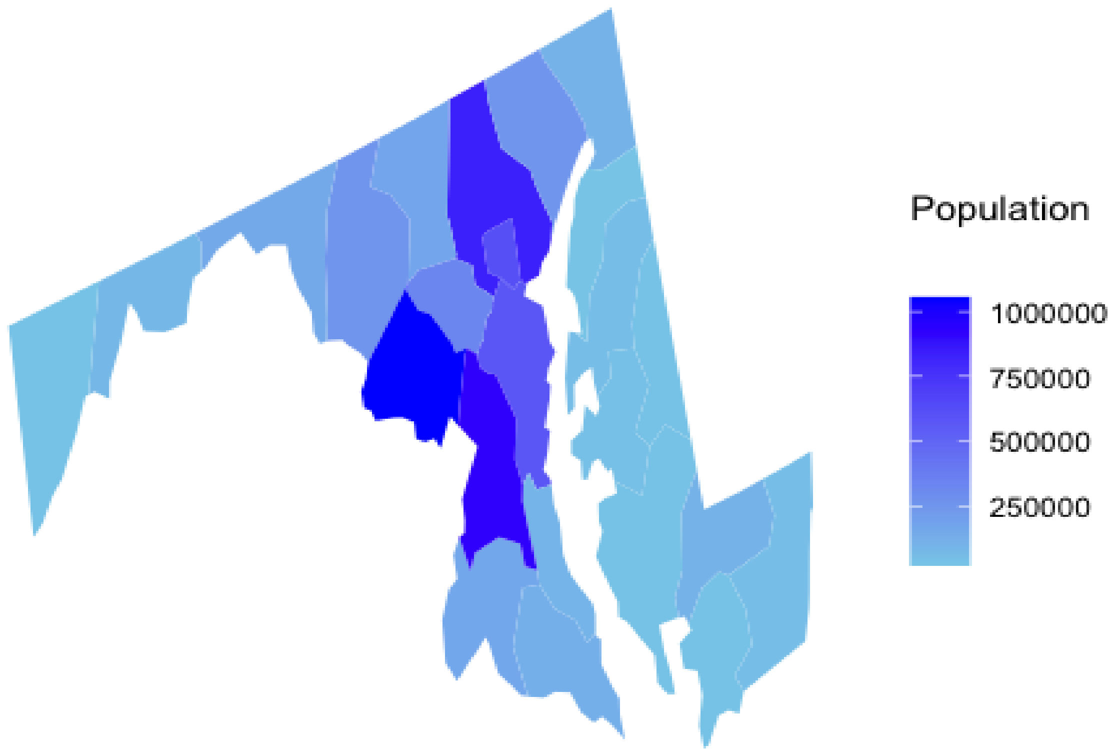
According to the map above, we can emphasize several counties with a fast-growing crime rate, and what is more important all of them located in various parts of the state. It tells us that violent crimes don't connect with a particular side or area of the state. Although Baltimore is considered a city with a high crime rate, it has the lowest value (yellow area in the middle) in the crime rate changes. Thus, it assumes that the crime rate in Baltimore City falls faster than in any other county each year.

```
# Crime rate map
print(crime_map_final)
```



Comparing the crime rate map with the population map, we can assume that there is no obvious correlation between crime rate changes and number of population of each county. For instance, 5 counties with the lowest population have rotating positive and negative changes in a crime rate. On the other hand, 4 counties with the highest population have a tendency to decline the crime rate. Eventually, according to our model, we cannot claim with certainty that the population effects on increasing or decreasing the crime rate.

```r
# population map
print(pop_map_final)
```



To sum up, our analysis might question some prejudgment and bias. As we used only one predictor variable, there are many ways to improve our model to make our results more robust by adding more explanatory variables from other data sources. Another method is decreasing the scope of the observation to analyze data for the past 10 or 20 years to make predictions more relevant.