MPLS Charter Amendment

Ryan Larson

2022-11-01

2020 Decennial Census Data

```
census_20 <- get_decennial(
  geography = "voting district",
   variables = c("P4_001N", "P4_002N","P4_005N",
                 "P4_006N", "P4_007N", "P4_008N",
                 "P4_009N", "P4_010N", "P4_011N"),
  year = 2020,
  state = "MN",
  county = "Hennepin",
  geometry = TRUE,
  output = "wide"
 ) %>%
  rename(total_pop = P4_001N,
         hisp_pop = P4_002N,
         white_pop = P4_005N,
         black_pop = P4_006N,
         na_pop = P4_007N,
         asian_pop = P4_008N,
         nhpi_pop = P4_009N,
         other_pop = P4_010N,
         two_pop = P4_011N) %>%
  filter(str_detect(NAME, "Minneapolis")) %>%
  mutate(NAME = gsub("\\,.*", "", NAME))
```

```
## Getting data from the 2020 decennial Census

## Downloading feature geometry from the Census website.  To cache shapefiles for use in future sessions

## Using the PL 94-171 Redistricting Data summary file

##   |                                                                              |
```

```
url_data <- "https://electionresults.sos.state.mn.us/Results/Index?ersElectionId=142&scenario=ResultsBy
```

```
css_selector <- "#dblscroll > table"

precinct <- url_data %>%
  read_html() %>%
  html_element(css = css_selector) %>%
  html_table() %>%
  slice(-n()) %>%
  rename(precinct = `County: Precinct`,
         yes_votes = `NP YES`,
         no_votes = `NP NO`) %>%
  mutate(precinct = str_remove(precinct, pattern = "Hennepin: "),
         yes_votes = str_remove(yes_votes, pattern = ","),
         no_votes = str_remove(no_votes, pattern = ",")) %>%
  mutate_at(c("yes_votes", "no_votes"), as.numeric) %>%
  mutate(perc_yes = yes_votes/(yes_votes+no_votes)*100) %>%
  mutate(precinct = str_replace_all(precinct, "MINNEAPOLIS", "Minneapolis"),
         precinct = str_replace_all(precinct, "-0", "-")) %>%
  mutate_all(~ifelse(is.nan(.), NA, .))



#join to census extract

precinct_sf <- census_20 %>%
  left_join(precinct, by = c("NAME"="precinct"))
```

Minneapolis Police Department Data

```
#Minneapolis Police Department - Use of Force Dashboard
uof_spatial <- read_csv("MPLS Charter Amendment/Data/Police_Use_of_Force.csv")  %>%
  mutate(date=ymd_hms(ResponseDate)) %>%
  filter(date >= as.Date("2020-01-01") & date < as.Date("2021-11-08")) %>%
  select(OBJECTID, X, Y, Race) %>%
  st_as_sf(coords = c("X", "Y"), crs = "NAD83", remove=F) %>%
  st_join(precinct_sf) %>%
  st_drop_geometry() %>%
  filter(!is.na(NAME)) %>%
  group_by(NAME, Race, .drop=F) %>%
  tally(name = "use_of_force") %>%
  filter(!is.na(Race)) %>%
  ungroup() %>%
  complete(NAME=precinct_sf$NAME, Race, fill = list(use_of_force = 0)) %>%
  arrange(NAME, Race) %>%
  mutate(race = str_to_lower(Race)) %>%
  select(-Race) %>%
  pivot_wider(names_from = race,
              values_from = use_of_force,
              values_fill = 0,
              names_glue = "{race}_{.value}") %>%
  mutate(total_use_of_force = asian_use_of_force+black_use_of_force+`native american_use_of_force`+
           `not recorded_use_of_force`+`pacific islander_use_of_force`+unknown_use_of_force+
            white_use_of_force)
```

```
## Rows: 37899 Columns: 30
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (15): CaseNumber, ResponseDate, Problem, Is911Call, PrimaryOffense, Subj...
## dbl (15): X, Y, PoliceUseOfForceID, ForceReportNumber, SubjectRoleNumber, Ev...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
# MPD Crime Data
crime_spatial <- read_csv("MPLS Charter Amendment/Data/Crime_Data.csv")  %>%
  mutate(date=ymd_hms(Occurred_Date)) %>%
  filter(date >= as.Date("2020-01-01") & date < as.Date("2021-11-08") &
           Type=="Crime Offenses (NIBRS)") %>%
  select(OBJECTID, Longitude, Latitude, Crime_Count) %>%
  st_as_sf(coords = c("Longitude", "Latitude"), crs = "NAD83", remove=F) %>%
  st_join(precinct_sf) %>%
  st_drop_geometry() %>%
  filter(!is.na(NAME)) %>%
  group_by(NAME, .drop=F) %>%
  summarize(mpd_crime = sum(Crime_Count, na.rm = T)) %>%
  ungroup() %>%
  complete(NAME=precinct_sf$NAME, fill = list(mpd_crime = 0)) %>%
  arrange(NAME)
```

```
## Rows: 201831 Columns: 24
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (14): Type, Case_Number, Case_NumberAlt, Reported_Date, Occurred_Date, N...
## dbl (10): X, Y, Precinct, Ward, Latitude, Longitude, wgsXAnon, wgsYAnon, Cri...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
#MPD Stop Dashboard
stops_spatial <- read_csv("MPLS Charter Amendment/Data/Police_Stop_Data.csv")  %>%
  mutate(date=ymd_hms(responseDate)) %>%
  filter(date >= as.Date("2020-01-01") & date < as.Date("2021-11-08")) %>%
  select(OBJECTID, long, lat) %>%
  st_as_sf(coords = c("long", "lat"), crs = "NAD83", remove=F) %>%
  st_join(precinct_sf) %>%
  st_drop_geometry() %>%
  filter(!is.na(NAME)) %>%
  group_by(NAME, .drop=F) %>%
  tally(name = "stops") %>%
  ungroup() %>%
  complete(NAME=precinct_sf$NAME, fill = list(stops = 0)) %>%
  arrange(NAME)
```

```
## Rows: 199309 Columns: 19
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (13): masterIncidentNumber, responseDate, reason, problem, callDispositi...
## dbl  (6): OBJECTID, lat, long, x, y, policePrecinct
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
#Officer Involved Shootings - MPD
ois_spatial <- read_csv("MPLS Charter Amendment/Data/Police_Officer_Involved_Shootings.csv") %>%
  mutate(date=ymd_hms(IncidentDate)) %>%
  filter(date >= as.Date("2020-01-01") & date < as.Date("2021-11-08")) %>%
  select(OBJECTID, CenterLatitude, CenterLongitude) %>%
  rename(lat = CenterLatitude,
         long = CenterLongitude) %>%
  st_as_sf(coords = c("long", "lat"), crs = "NAD83", remove=F) %>%
  st_join(precinct_sf) %>%
  st_drop_geometry() %>%
  filter(!is.na(NAME)) %>%
  group_by(NAME, .drop=F) %>%
  tally(name = "ois") %>%
  ungroup() %>%
  complete(NAME=precinct_sf$NAME, fill = list(ois = 0)) %>%
  arrange(NAME)
```

```
## Rows: 90 Columns: 21
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (11): IncidentDate, CaseNumber, Officer, OfficerWeaponUsed, OfficerInjur...
## dbl (10): X, Y, OBJECTID, OfficerYearsOfService, SubjectOfForceAge, CenterLa...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

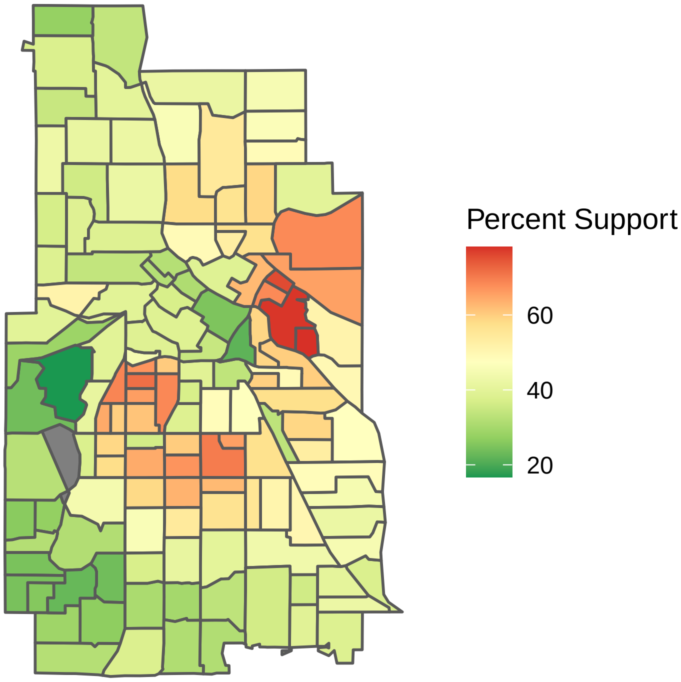
```
precinct_sf <- precinct_sf %>%
  left_join(uof_spatial, by = "NAME") %>%
  left_join(stops_spatial, by = "NAME") %>%
  left_join(ois_spatial, by = "NAME") %>%
  left_join(crime_spatial, by = "NAME") %>%
  mutate(uof_rate = total_use_of_force/total_pop*1000,
         stops_rate = stops/total_pop*1000,
         ois_rate = ois/total_pop*1000,
         crime_rate = mpd_crime/total_pop*1000)
```

Support for Charter Amendment #2

```
ggplot() +
  geom_sf(data = precinct_sf, aes(geometry = geometry, fill = perc_yes)) +
  scale_fill_distiller(palette = "RdYlGn")+
  labs(title = "Figure 1: Support for MPLS Charter Amendment #2, 2021",
       subtitle = "MN Secretary of State",
       fill = "Percent Support")+
  theme(axis.text.x = element_blank(),
        axis.text.y = element_blank(),
  axis.line = element_blank(),
  axis.ticks = element_blank(),
  panel.border = element_blank(),
  panel.grid = element_blank(),
  axis.title = element_blank(),
  panel.background = element_blank(),
  panel.grid.major = element_line(colour="transparent"),
  plot.subtitle = element_text(face="italic"),
  strip.background = element_rect(fill = "white",
                colour = "black"))
```

Figure 1: Support for MPLS Charter Amendment #2, 2021

MN Secretary of State



Preliminary Spatial Autoregressive Model

```
m1 <- lm(perc_yes~white_pop+black_pop+na_pop+hisp_pop+asian_pop+uof_rate+stops_rate+ois_rate+crime_rate

summary(m1)
```

```
##
## Call:
## lm(formula = perc_yes ~ white_pop + black_pop + na_pop + hisp_pop +
##     asian_pop + uof_rate + stops_rate + ois_rate + crime_rate,
##     data = precinct_sf)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -23.889  -9.135  -2.287   8.687  30.475
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 36.207831   3.958660   9.146 1.46e-15 ***
## white_pop    0.001356   0.001683   0.806  0.42187
## black_pop   -0.009276   0.002997  -3.095  0.00243 **
## na_pop      -0.041688   0.031427  -1.327  0.18711
## hisp_pop     0.032559   0.007216   4.512 1.47e-05 ***
## asian_pop    0.007876   0.005272   1.494  0.13777
## uof_rate    -0.034248   0.097508  -0.351  0.72601
## stops_rate  -0.015265   0.020913  -0.730  0.46681
## ois_rate     1.229120   5.246648   0.234  0.81516
## crime_rate   0.024121   0.016073   1.501  0.13598
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.47 on 124 degrees of freedom
##   (2 observations deleted due to missingness)
## Multiple R-squared:  0.2312, Adjusted R-squared:  0.1754
## F-statistic: 4.143 on 9 and 124 DF,  p-value: 0.0001095
```

```
library(spdep)
```

```
## Loading required package: sp

## Loading required package: spData

## To access larger datasets in this package, install the spDataLarge
## package with: `install.packages('spDataLarge',
## repos='https://nowosad.github.io/drat/', type='source')`
```

```
library(spatialreg)
```

```
## Loading required package: Matrix

##
## Attaching package: 'Matrix'

## The following objects are masked from 'package:tidyr':
##
##     expand, pack, unpack

##
## Attaching package: 'spatialreg'

## The following objects are masked from 'package:spdep':
##
##     get.ClusterOption, get.coresOption, get.mcOption,
##     get.VerboseOption, get.ZeroPolicyOption, set.ClusterOption,
##     set.coresOption, set.mcOption, set.VerboseOption,
##     set.ZeroPolicyOption
```

```
#neighbors: queen adjacency
mpls_nb <- poly2nb(precinct_sf, queen=T)
summary(mpls_nb)
```

```
## Neighbour list object:
## Number of regions: 136
## Number of nonzero links: 766
## Percentage nonzero weights: 4.141436
## Average number of links: 5.632353
## Link number distribution:
##
##  2  3  4  5  6  7  8  9 10
##  3 10 26 28 25 25 11  7  1
## 3 least connected regions:
## 7 9 125 with 2 links
## 1 most connected region:
## 4 with 10 links
```

```
#neighbor weights
mpls_w <- nb2listw(mpls_nb, style = "W", zero.policy = T)

print(lm.morantest(m1, mpls_w)) #testing for spatial autocorrelation in residuals
```

```
##
##  Global Moran I for regression residuals
##
## data:
## model: lm(formula = perc_yes ~ white_pop + black_pop + na_pop +
## hisp_pop + asian_pop + uof_rate + stops_rate + ois_rate + crime_rate,
## data = precinct_sf)
## weights: mpls_w
##
## Moran I statistic standard deviate = 11.635, p-value < 2.2e-16
## alternative hypothesis: greater
## sample estimates:
## Observed Moran I      Expectation         Variance
##      0.564898411     -0.023879798      0.002560702
```

```
m1_lag <- lagsarlm(perc_yes~white_pop+black_pop+na_pop+hisp_pop+asian_pop+uof_rate+stops_rate+ois_rate+c
                   data = precinct_sf, listw = mpls_w)

summary(m1_lag)
```

```
##
## Call:lagsarlm(formula = perc_yes ~ white_pop + black_pop + na_pop +
##     hisp_pop + asian_pop + uof_rate + stops_rate + ois_rate +
##     crime_rate, data = precinct_sf, listw = mpls_w)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -19.87124  -3.67303  -0.21825   3.80202  18.55735
##
## Type: lag
## Coefficients: (asymptotic standard errors)
##                Estimate  Std. Error z value Pr(>|z|)
## (Intercept)  2.95967375  2.92820900  1.0107 0.312138
## white_pop    0.00123673  0.00094124  1.3139 0.188870
## black_pop   -0.00413182  0.00170121 -2.4288 0.015151
## na_pop      -0.02155905  0.01760574 -1.2245 0.220746
## hisp_pop     0.01321901  0.00417184  3.1686 0.001532
## asian_pop   -0.00103885  0.00295365 -0.3517 0.725050
## uof_rate     0.02893161  0.05456738  0.5302 0.595973
## stops_rate  -0.00850566  0.01175341 -0.7237 0.469265
## ois_rate    -0.89592297  2.93292004 -0.3055 0.760007
## crime_rate   0.00757228  0.00898935  0.8424 0.399586
##
## Rho: 0.84778, LR test value: 117.66, p-value: < 2.22e-16
## Asymptotic standard error: 0.04468
##     z-value: 18.975, p-value: < 2.22e-16
## Wald statistic: 360.04, p-value: < 2.22e-16
##
## Log likelihood: -464.2179 for lag model
## ML residual variance (sigma squared): 48.574, (sigma: 6.9695)
## Number of observations: 134
## Number of parameters estimated: 12
## AIC: 952.44, (AIC for lm: 1068.1)
## LM test for residual autocorrelation
## test value: 3.6994, p-value: 0.054434
```