Benches Installed by the New York City Department of Transportation (NYCDOT)

General

Packages

```
library(tidyverse, warn.conflicts = FALSE) # for ggplot, mutate, case_when, etc.
library(lubridate, warn.conflicts = FALSE) # for date conversions w ymd
library(RSocrata) # for accessing NYC Socrata Open Data API (SODA)
library(sf) # for reading shapefiles with st_read()
library(viridis) # for viridis color scales "mako"
library(scales) # for percent scale on legend and comma y labels
library(ggiraph) # for interactive ggplots
library(shiny) # for creating freestanding interactive apps
library(shinybusy) # for "busy indicator" on large maps
library(ggformula) # for geom_spline function on ggplot
library(ggnewscale) # for having two "fill" scales on the Seniors graph
library(rio) # for reading an Excel file of demographic data

token <- Sys.getenv("NYC_SODA_KEY") # API key for NYC open data; stored in my .Renviron file

setwd("C:/Users/Nate/Desktop/CUSP-GX 6006/Term_Project/app")

options(stringsAsFactors = FALSE)
```

Load and adjust data

```
benches <- read.socrata("https://data.cityofnewyork.us/resource/kuxa-tauh.json",
    app_token = token)

# generates problems when exporting
benches$the_geom.coordinates <- NULL

# export to csv, import into QGIS and create shapefile using lat, long
# write.csv(benches, "data/benches.csv")

city <- st_read("data/gis/boros.shp")
# https://data.cityofnewyork.us/City-Government/Borough-Boundaries/tqmj-j8zm
districts <- st_read("data/gis/districts.shp") # community districts
# https://www1.nyc.gov/site/planning/data-maps/open-data/districts-download-metadata.page
benches <- st_read("data/gis/benches.shp")
# same data as before but with geometry added (add'l column)

# remove unnecessary add'l columns
remove <- c(
    "the_geom.t", # not useful
    "borough", # duplicate column, technically wrong (said Marble Hill in BX)
```

```
    "field_1" # duplicate of row number that R generates
)
benches <- benches[ , !(names(benches) %in% remove)]

benches[benches$bid == "NA", "bid"] <- "Not Applicable"
# for BID analysis later; consistent labeling

benches$color <- "brown"
# used much later in maps with scale_color_identity
```

Add the borough of each district

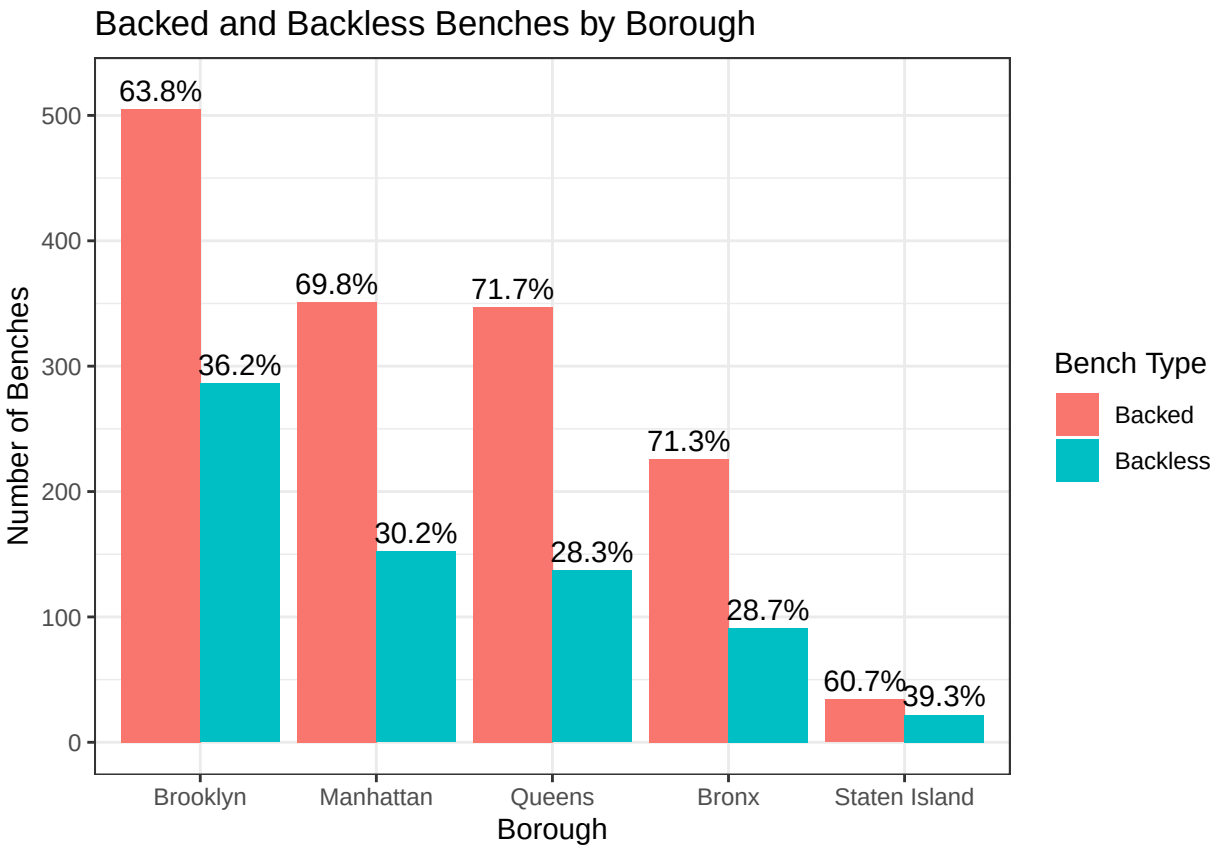
```
# match boro_code in "city" and borocode in "benches"
districts <- districts %>%
    select(everything()) %>%
        mutate(Borough =
            case_when(
                substr(BoroCD, 1, 1) == 1 ~ "Manhattan",
                substr(BoroCD, 1, 1) == 2 ~ "Bronx",
                substr(BoroCD, 1, 1) == 3 ~ "Brooklyn",
                substr(BoroCD, 1, 1) == 4 ~ "Queens",
                substr(BoroCD, 1, 1) == 5 ~ "Staten Island"
            )
        )
```

Analysis/results

Number by borough, with bench type

```
# remove NA benchtype (what does this mean?)
# remove diff btw backed/backless and backed 2/backless 2
benches <- benches[benches$benchtype != "NA", ] # not a true NA, but character
benches <- benches %>%
    select(everything()) %>%
        mutate(benchtype =
            case_when(
                benchtype == 'backed 2' ~ 'backed',
                benchtype == 'backless 2' ~ 'backless',
                TRUE ~ benchtype
            )
        )

by_boro <- aggregate(benchid ~ boroname + benchtype,
    data = benches,
    FUN = length
)

by_boro_wide <- pivot_wider(by_boro,
    names_from = benchtype,
    values_from = benchid
)

by_boro_wide <- by_boro_wide %>%
    select(everything()) %>%
```

```
        mutate(total = backed + backless,
            pct_backed = backed / total)

by_boro <- left_join(
    x = by_boro,
    y = by_boro_wide[, c("boroname", "pct_backed")],
    by = "boroname"
)

rm(by_boro_wide)

by_boro[by_boro$benchtype == "backless", "pct_backed"] <- 1 -
    by_boro[by_boro$benchtype == "backless", "pct_backed"]

ggplot(by_boro, aes(x = reorder(boroname, -benchid),
        y = benchid, fill = benchtype)) +
    geom_bar(stat = "identity", position = "dodge") +
    scale_fill_discrete(labels = c("Backed", "Backless")) +
    geom_text(
        aes(y = benchid + 15), stat = "identity",
        position = position_dodge(width = 0.9),
        label = paste0(
            round(by_boro$pct_backed, 3) * 100, "%"
        )
    ) +
    xlab("Borough") +
    ylab("Number of Benches") +
    labs(
        title = "Backed and Backless Benches by Borough",
        fill = "Bench Type"
    ) +
    theme_bw()
```



By community district with bench type

```
by_cd <- aggregate(benchid ~ borocd + benchtype,
    data = benches, FUN = length
)

by_cd_wide <- pivot_wider(by_cd,
    names_from = benchtype,
    values_from = benchid
)

# some districts have only one type of bench, resulting in NA for the other
# which messes up the calculations
by_cd_wide[is.na(by_cd_wide$backed), "backed"] <- 0
by_cd_wide[is.na(by_cd_wide$backless), "backless"] <- 0

by_cd_wide <- by_cd_wide %>%
    select(everything()) %>%
        mutate(total = backed + backless,
            pct_backed = backed / total)

# join to districts shapefile df
districts <- left_join(
    x = districts,
    y = by_cd_wide,
    by = c("BoroCD" = "borocd")
)
```

```r
# some CDs do not appear in "benches" and give NA, replace w 0s
districts[is.na(districts$backed), c("backed", "backless", "total")] <- 0
```

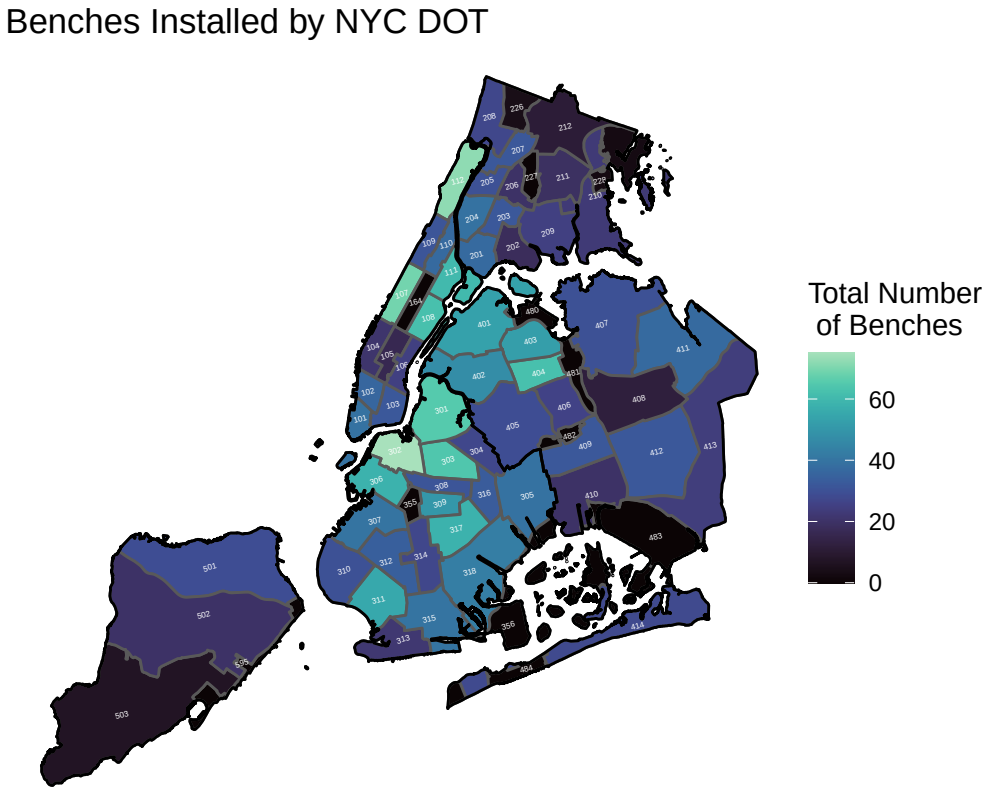
Brooklyn Community District 302 has highest total number of benches (75)

Lowest number is CDs with 0 benches

```r
# to add labels to heat map below
districts_points <- st_point_on_surface(districts)
```

```
## Warning in st_point_on_surface.sf(districts): st_point_on_surface assumes
## attributes are constant over geometries of x
```

```r
# generates X,Y coordinates
districts_coords <- as.data.frame(st_coordinates(districts_points))

# add CD #
districts_coords$BoroCD <- districts$BoroCD

# "heat map" of results
ggplot() +
    geom_sf(data = districts, aes(fill = total)) + # district boundaries
    scale_fill_viridis(
        end = 0.9, option = "mako", # don't go all the way to the
        # lightest color or the white labels won't show up
        "Total Number \n of Benches"
    ) +
    geom_text(
        data = districts_coords, aes(X, Y, label = BoroCD),
        colour = "white", size = 1, angle = 15
    ) +
    geom_sf(data = city, color = "black", fill = NA) + # boros outline
    labs(
        title = "Benches Installed by NYC DOT",
        caption = "(Labels are Community District numbers)"
    ) +
    theme_void()
```



(Labels are Community District numbers)

```
b_max <- max(districts$total, na.rm = TRUE)
cd_b_max <- districts[[which.max(districts$total), "BoroCD"]]

print(
    paste0(
        "Community District ", cd_b_max, " has the largest number ",
            "of benches (", b_max, ")"
    )
)
```

```
## [1] "Community District 302 has the largest number of benches (75)"
```

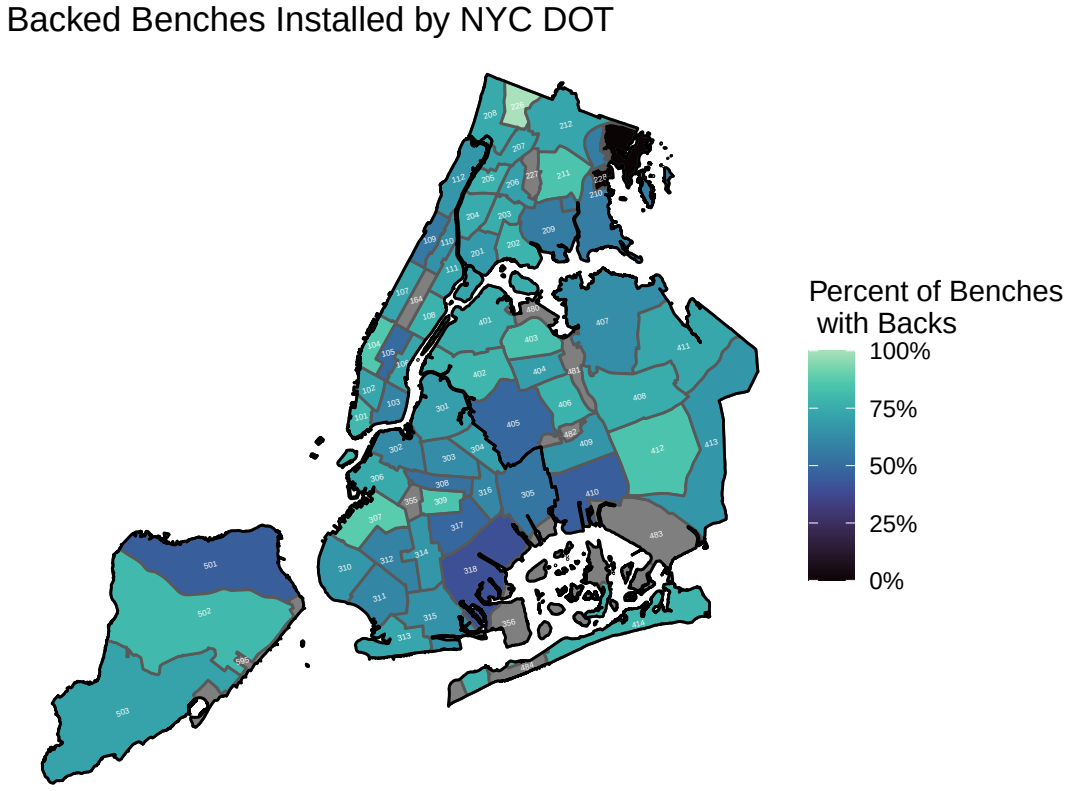
Repeat with percent of benches with backs

All benches in Bronx Community District 226 have backs

Bronx Community District 228 has lowest percentage of benches with backs (though this is basically part of Pelham Bay Park?)

```
# "heat map" of pct_backed
ggplot() +
    geom_sf(data = districts, aes(fill = pct_backed)) + # district boundaries
    scale_fill_viridis(
        end = 0.9, option = "mako",
        "Percent of Benches \n with Backs", labels = percent
    ) +
    geom_text(
        data = districts_coords, aes(X, Y, label = BoroCD),
```

```
        colour = "white", size = 1, angle = 15
    ) +
    geom_sf(data = city, color = "black", fill = NA) + # boros outline
    labs(
        title = "Backed Benches Installed by NYC DOT",
        caption = "(Labels are Community District numbers)"
    ) +
    theme_void()
```




```
pct_max <- max(districts$pct_backed, na.rm = TRUE)
cd_pct_max <- districts[[which.max(districts$pct_backed), "BoroCD"]]

pct_min <- min(districts$pct_backed, na.rm = TRUE)
cd_pct_min <- districts[[which.min(districts$pct_backed), "BoroCD"]]

print(
    paste0(
        "Community District ", cd_pct_max, " has the greatest percentage ",
            "of benches with backs (", round(pct_max, 3)*100, "%)"
    )
)
```

```
## [1] "Community District 226 has the greatest percentage of benches with backs (100%)"
```

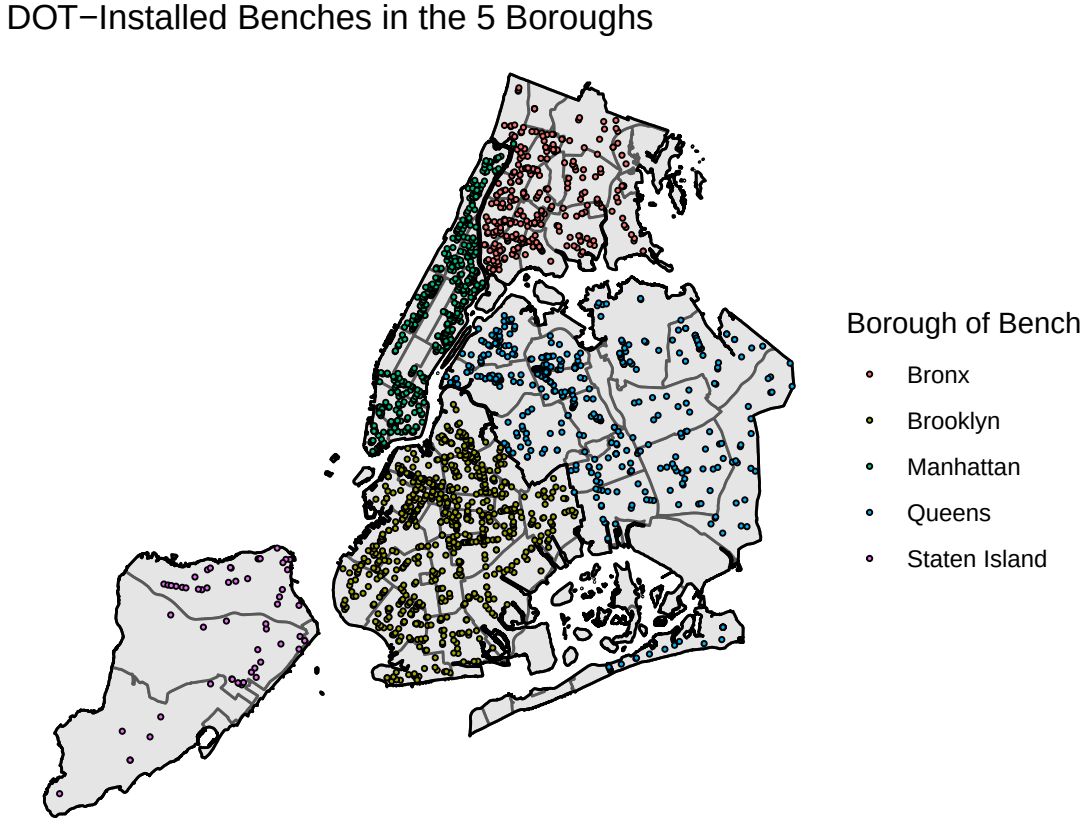
```
print(
    paste0(
        "Community District ", cd_pct_min, " has the smallest percentage ",
```

```
            "of benches with backs (", round(pct_min, 3)*100, "%)"
    )
)
```

```
## [1] "Community District 228 has the smallest percentage of benches with backs (0%)"
```

Plot by location

```
ggplot() +
    geom_sf(data = districts) + # district boundaries
    geom_sf(data = city, color = "black", fill = NA) + # boros outline
    geom_sf(
        data = benches, aes(fill = boroname),
        color = "black", shape = 21, size = 0.6
    ) + # shape = 21 necessary to have fill AND outline
    labs(
        title = "DOT-Installed Benches in the 5 Boroughs",
        fill = "Borough of Bench"
    ) +
    theme_void()
```

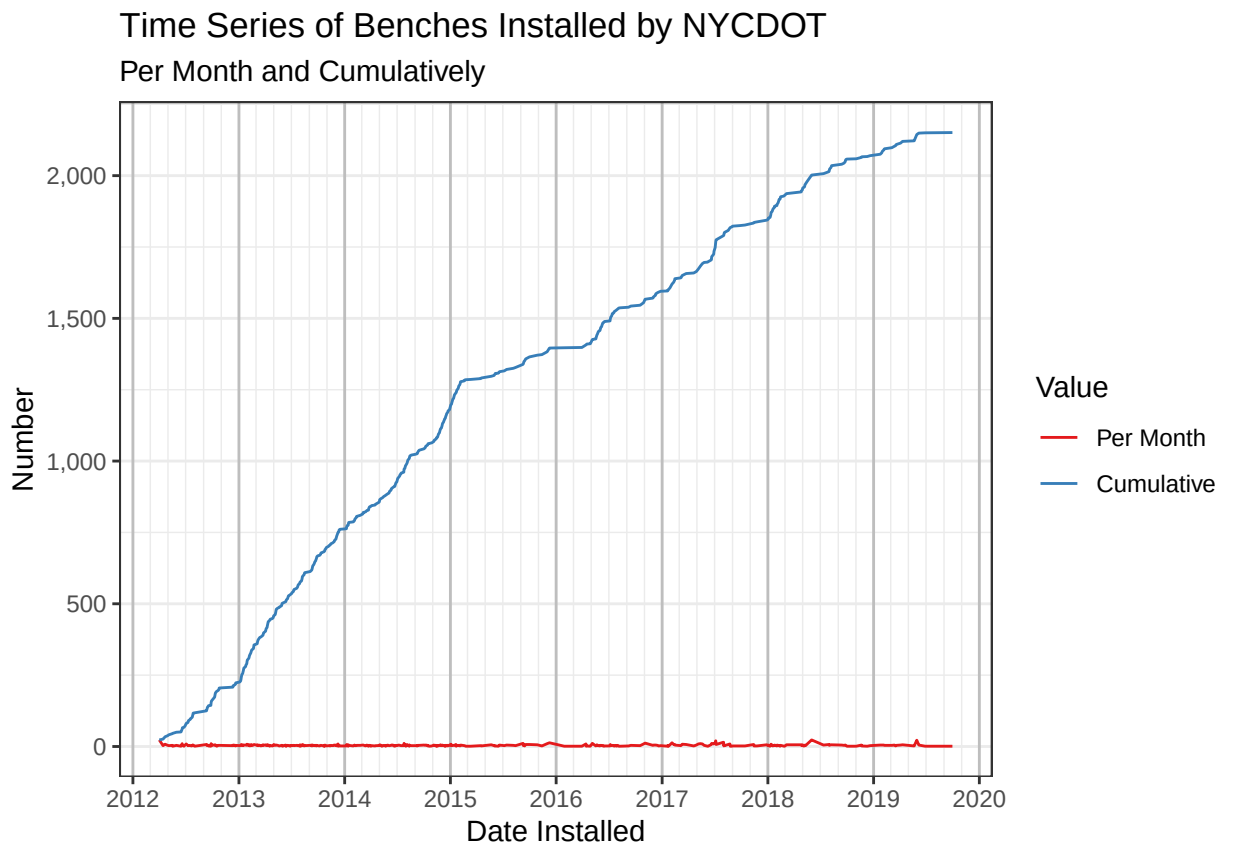


When were benches installed?

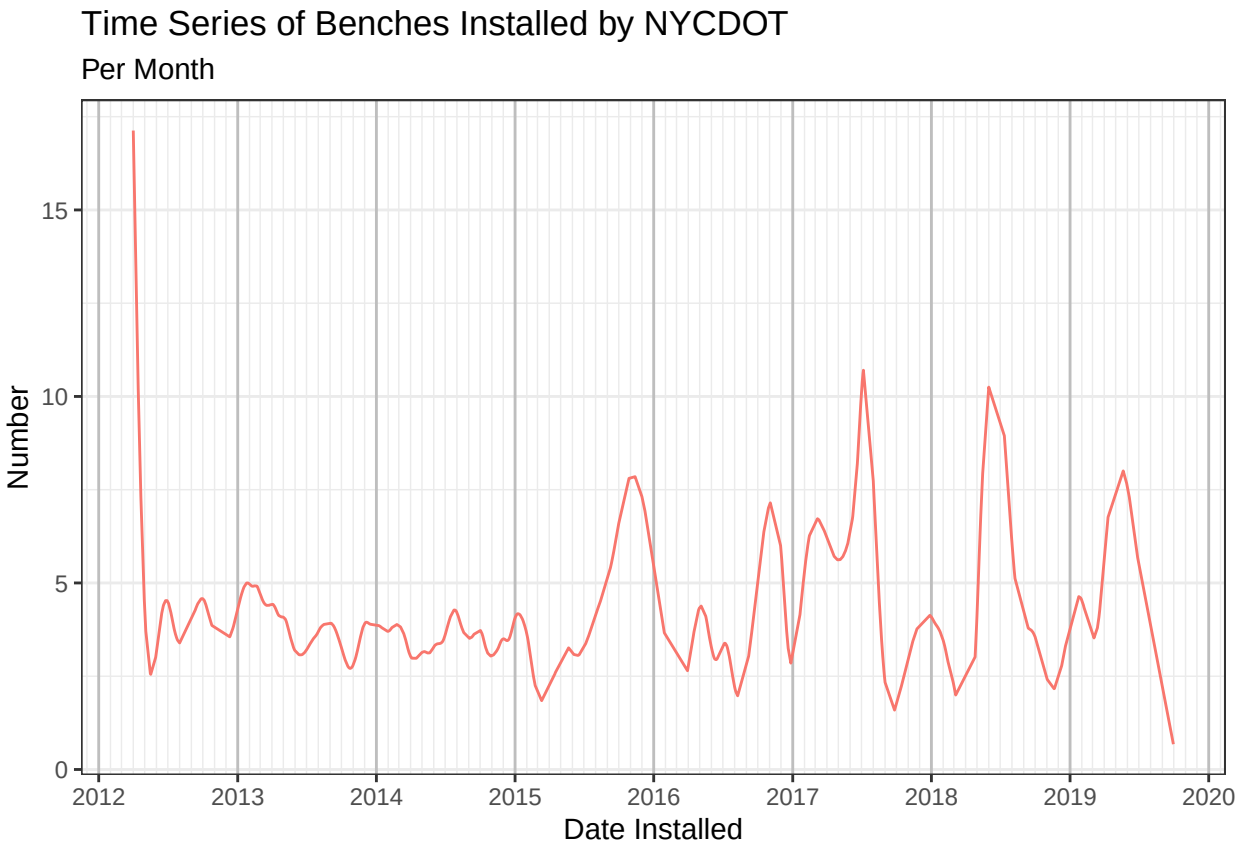
```
# fix column name "installati"? -> installed
names(benches)[names(benches) == "installati"] <- "installed"

# date of installation is a character, convert to POSIX date
benches$installed <- ymd(benches$installed)
```

```
## Warning: 13 failed to parse.
```

```
# some benches have NA installation date, replace with earliest date
benches <- benches %>%
    select(everything()) %>%
        mutate(installed =
            case_when(
                is.na(installed) ~ min(benches$installed, na.rm = TRUE),
                TRUE ~ installed
            )
        )

by_date <- aggregate(benchid ~ installed,
    data = benches, FUN = length
)

by_date$cumulative <- cumsum(by_date$benchid)

# make data "tidy" to make plotting easier
by_date_long <- pivot_longer(by_date,
    cols = benchid:cumulative,
    names_to = "type"
)

ggplot(data = by_date_long) +
    geom_line(aes(x = installed, y = value, color = type)) +
    xlab("Date Installed") +
    ylab("Number") +
    labs(
        title = "Time Series of Benches Installed by NYCDOT",
        subtitle = "Per Month and Cumulatively"
    ) +
    scale_color_brewer(
        palette = "Set1",
        name = "Value",
        labels = c("Per Month", "Cumulative")
    ) +
    scale_x_date(
        breaks = "1 year", minor_breaks = "2 months",
        date_labels = "%Y"
    ) +
    scale_y_continuous(label = comma) +
    theme_bw() +
    theme(panel.grid.major.x = element_line(colour = "grey"))
```




```r
# now to hightlight monthly values
monthly <- by_date_long %>%
    select(everything()) %>%
        filter(type == "benchid")

ggplot(monthly, aes(x = installed, y = value)) +
    geom_spline(color = "#F8766D") +
    # geom_spline gets rid of jagged lines that obscure the trends
    xlab("Date Installed") +
    ylab("Number") +
    labs(
        title = "Time Series of Benches Installed by NYCDOT",
        subtitle = "Per Month"
    ) +
    scale_x_date(
        breaks = "1 year", minor_breaks = "1 month",
        date_labels = "%Y"
    ) +
    theme_bw() +
    theme(panel.grid.major.x = element_line(colour = "grey"))
```



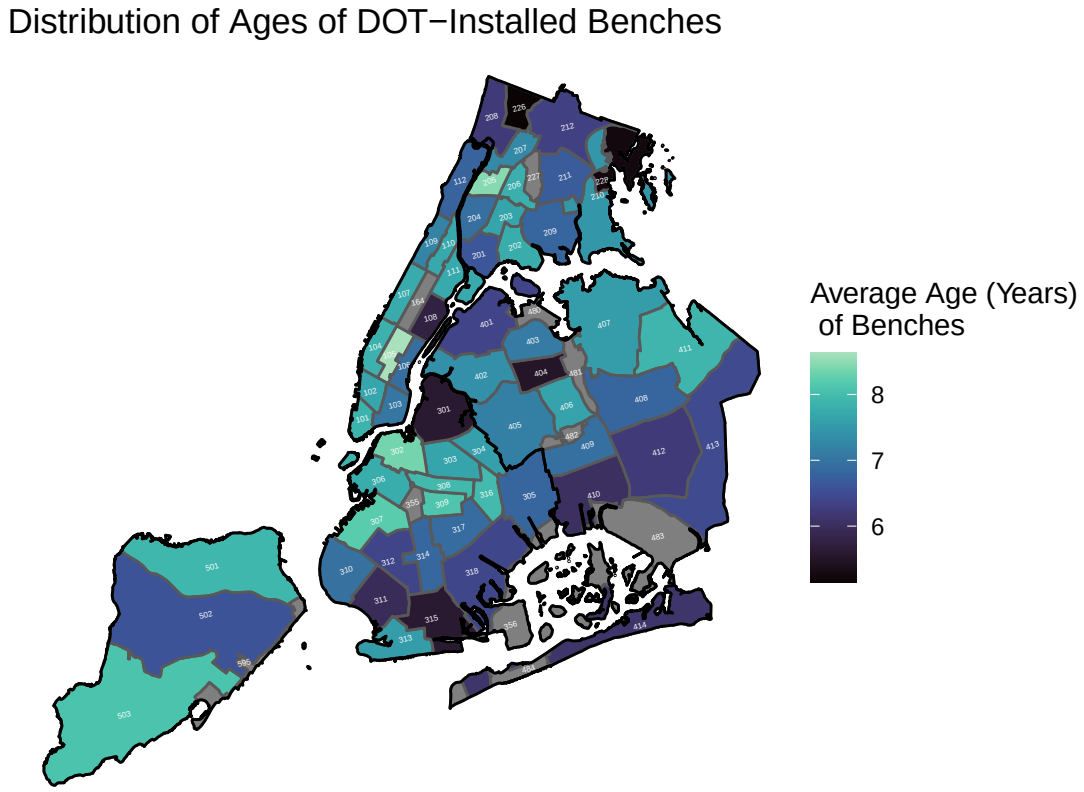
Different way to show data—age of bench

Manhattan Community District 105 has lowest average age (5.14 years)

Bronx Community District 226 has highest average age (8.62 years)

```
benches$age <- as.numeric(
    difftime(as.Date(today()), as.Date(benches$installed),
        units = "days"
    ) / 365
) # age in years

benches_avg_age <- setNames(aggregate(age ~ borocd,
    data = benches, FUN = mean),
    c("borocd", "avg_age")
)

districts <- left_join(
    x = districts,
    y = benches_avg_age,
    by = c("BoroCD" = "borocd")
)

ggplot() +
    geom_sf(data = districts, aes(fill = avg_age)) + # district boundaries
    scale_fill_viridis(
        end = 0.9, option = "mako",
        "Average Age (Years) \n of Benches"
```

```
    ) +
    geom_text(
        data = districts_coords, aes(X, Y, label = BoroCD),
        colour = "white", size = 1, angle = 15
    ) +
    geom_sf(data = city, color = "black", fill = NA) + # boros outline
    labs(
        title = "Distribution of Ages of DOT-Installed Benches",
        caption = "(Labels are Community District numbers)"
    ) +
    theme_void()
```




```
avg_age_max <- max(districts$avg_age, na.rm = TRUE)
cd_avg_age_max <- districts[[which.max(districts$avg_age), "BoroCD"]]

avg_age_min <- min(districts$avg_age, na.rm = TRUE)
cd_avg_age_min <- districts[[which.min(districts$avg_age), "BoroCD"]]
```

Benches in Business Improvement Districts (BIDs)

```
by_bid_num <- setNames(aggregate(benchid ~ bid,
    data = benches, FUN = length),
    c("bid", "num")
)

by_bid_age <- setNames(aggregate(age ~ bid,
    data = benches, FUN = mean),
```

```
     c("bid", "avg_age")
)

by_bid <- left_join(
    x = by_bid_num,
    y = by_bid_age,
    by = "bid"
)
```

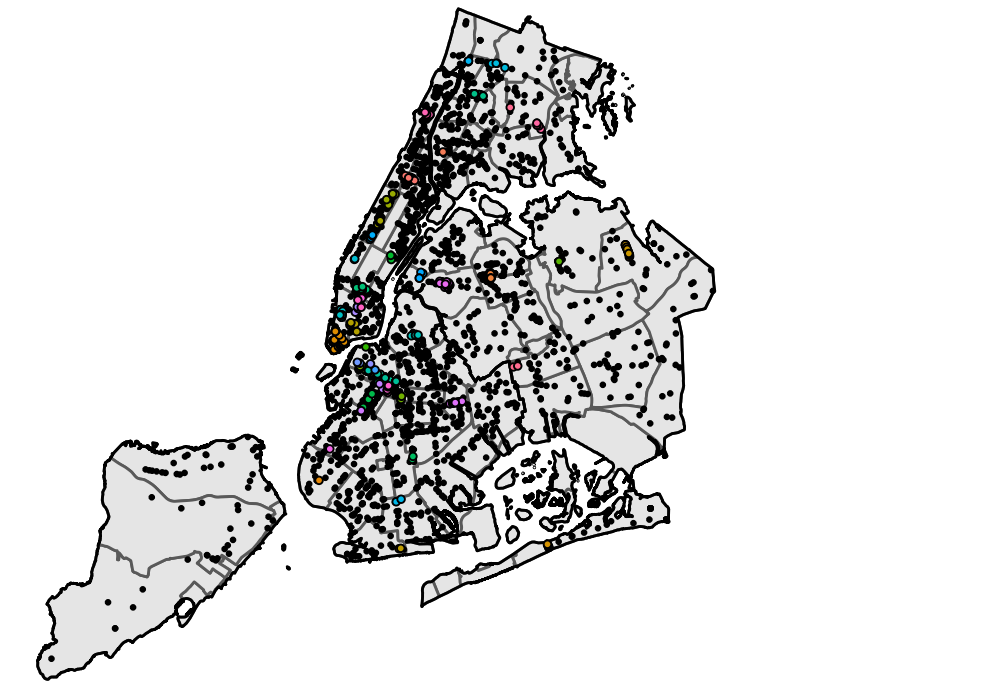
Majority of benches *not* in BID

```
in_bid <- benches[benches$bid != "Not Applicable",]
no_bid <- benches[benches$bid == "Not Applicable",]

print(
    paste0(
        100 * round(
            (dim(in_bid)[1])
            /(dim(in_bid)[1] + dim(no_bid)[1])
        , 3),
        "% of benches are in a BID."
    )
)
```

```
## [1] "12.6% of benches are in a BID."
```

Where are the benches that *are* in BIDs?

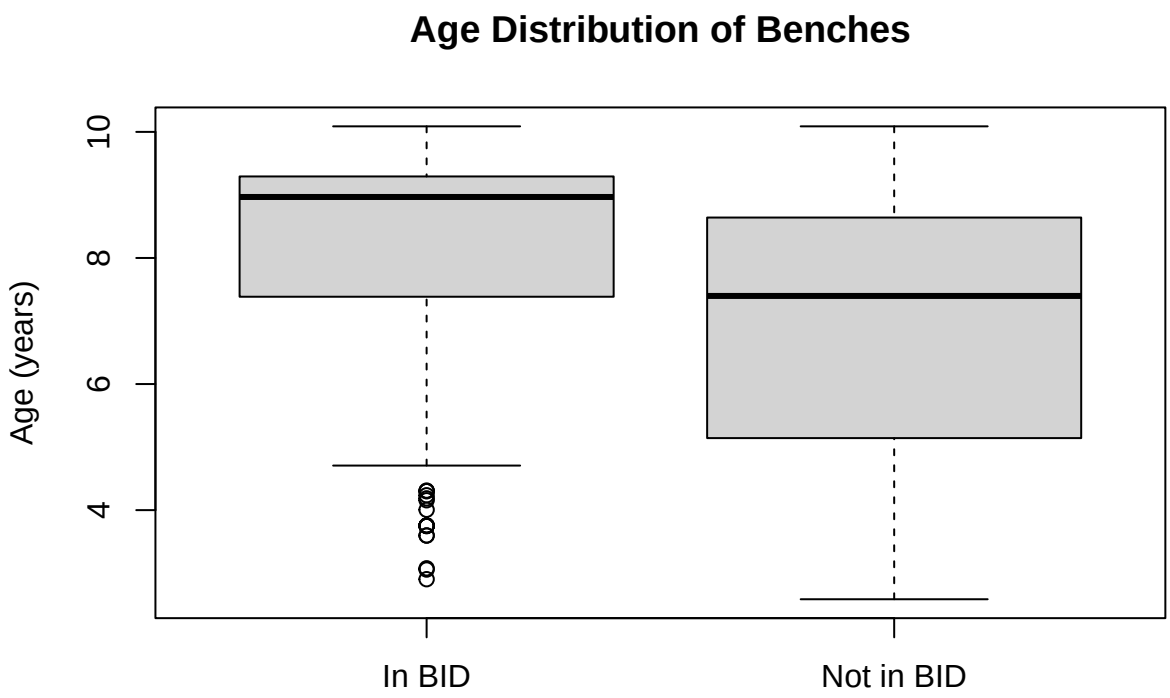
```
ggplot() +
    geom_sf(data = districts) + # district boundaries
    geom_sf(data = city, color = "black", fill = NA) + # boros outline
    geom_sf(
        data = no_bid, fill = "black",
        color = "black", shape = 21, size = 0.5
    ) +
    geom_sf(
        data = in_bid, aes(fill = bid),
        color = "black", shape = 21, size = 1
    ) +
    labs(
        title = "Placement of Benches in Business Improvement Districts (BIDs)",
        subtitle = "Benches in BIDs (Colored) vs Not in BIDs (Black)",
        caption = "(Benches colored according to BID that placed them)"
    ) +
    guides(fill = "none") +
    theme_void()
```



(Benches colored according to BID that placed them)

Significance testing about age of benches in BIDs

```
boxplot(in_bid$age, no_bid$age,
    main = "Age Distribution of Benches",
    ylab = "Age (years)",
    at = c(1, 2),
    names = c("In BID", "Not in BID")
)
```




```
neq <- t.test(in_bid$age, no_bid$age)
print(
    paste0(
        "p-value: ", neq$p.value
    )
)
```

```
## [1] "p-value: 3.09466149442883e-19"
```

```
# p << 0.05 -> can say w/ high certainty mean ages are different

greater <- t.test(in_bid$age, no_bid$age, alternative = "greater")
print(
    paste0(
        "p-value: ", greater$p.value
    )
)
```

```
## [1] "p-value: 1.54733074721441e-19"
```

```
# p << 0.05 -> can say w/ high certainty mean ages in BID
# is greater than mean age not in BID
```

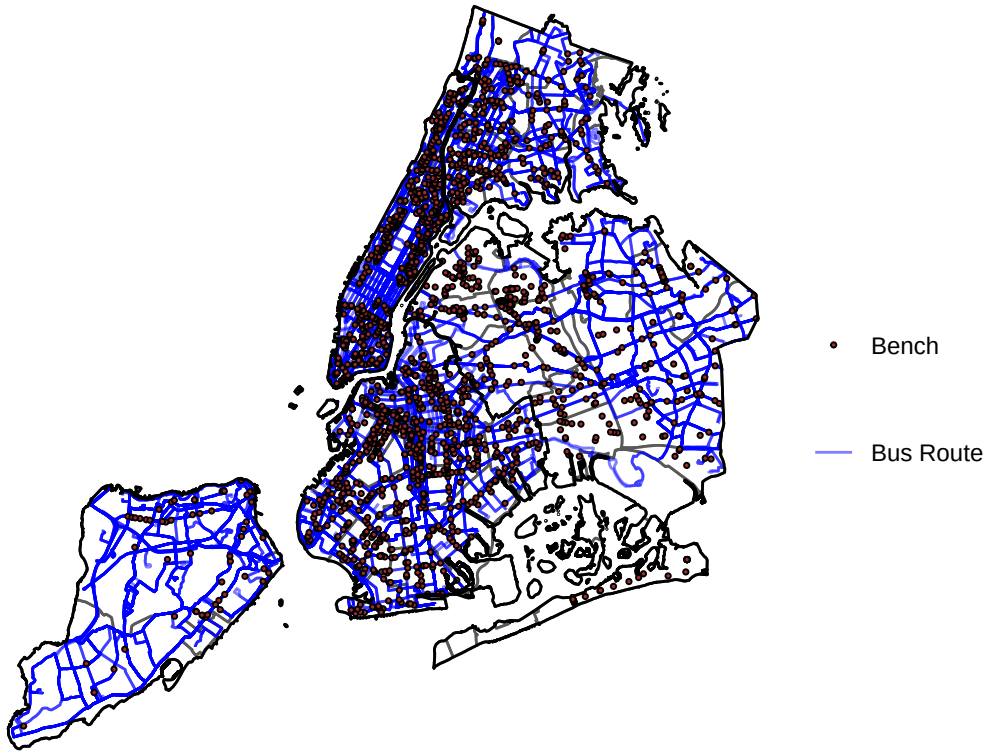
How did DOT do at installing benches near bus routes as they hoped?

```
with_busroute <- benches[benches$busroute != "Not Applicable",]
wout_busroute <- benches[benches$busroute == "Not Applicable",]
```

```r
# compilation of shape.txt files for each borough
# http://web.mta.info/developers/developer-data-terms.html#data
busroutes <-  st_read("data/gtfs/lines_5boros.shp")
```

```
## Reading layer `lines_5boros' from data source
##   `C:\Users\Nate\Desktop\CUSP-GX 6006\Term_Project\app\data\gtfs\lines_5boros.shp'
##   using driver `ESRI Shapefile'
## Simple feature collection with 1329 features and 3 fields
## Geometry type: MULTILINESTRING
## Dimension:     XY
## Bounding box:  xmin: -74.25234 ymin: 40.50287 xmax: -73.70124 ymax: 40.91239
## Geodetic CRS:  WGS 84
```

```r
busroutes$color <- "blue" # used below with scale_color_identity

ggplot() +
    geom_sf(data = districts, fill = "white") + # district boundaries
    geom_sf(data = busroutes, aes(color = color), alpha = 0.5) +
    scale_color_identity(
        name = "",
        labels = "Bus Route",
        guide = "legend"
    ) +
    geom_sf(
        data = benches, aes(fill = color),
        color = "black", shape = 21, size = 0.6
    ) +
    scale_fill_identity(
        name = "",
        labels = "Bench",
        guide = "legend"
    ) +
    ggtitle("Proximity of DOT-Installed Benches to MTA Bus Routes") +
    geom_sf(data = city, color = "black", fill = NA) + # boros outline
    theme_void()
```




```
print(
    paste0(
        100 * round(
            (dim(with_busroute)[1])
            /(dim(with_busroute)[1] + dim(wout_busroute)[1])
        , 3),
        "% of benches are located near a bus route."
    )
)
```

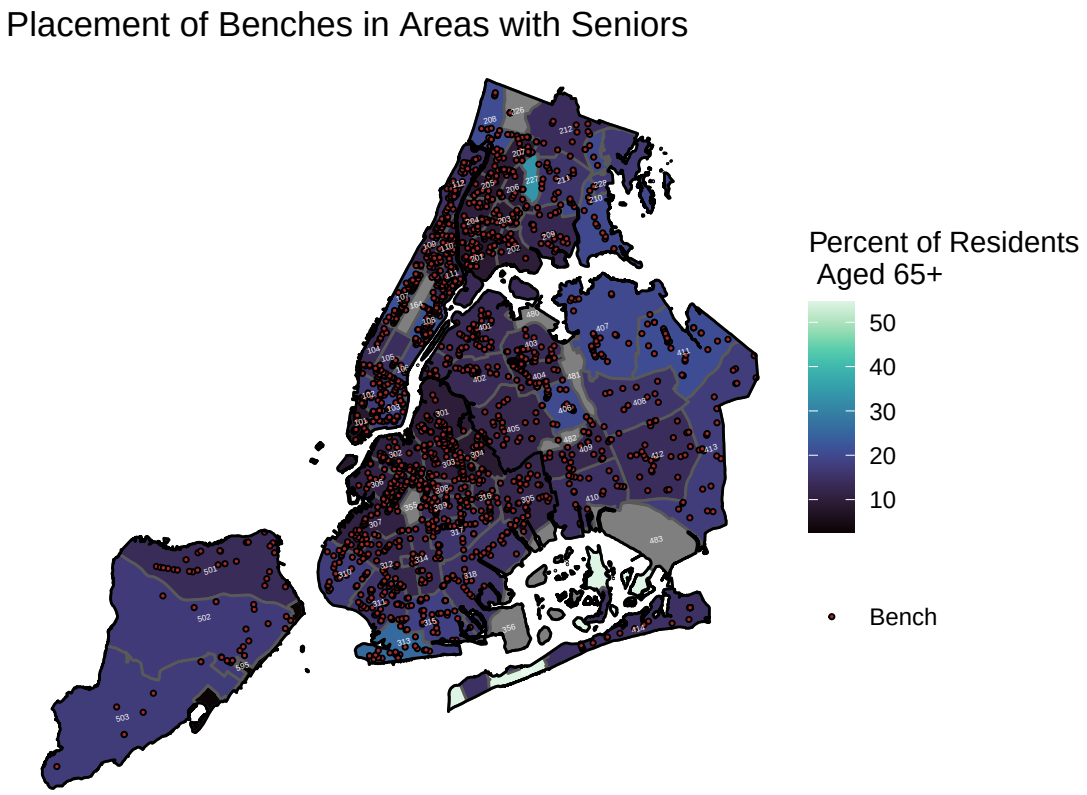
```
## [1] "36.5% of benches are located near a bus route."
```

Static version shown here; interactive version below

```
# NYC demographics data *by Community District*
# (census data is by census tract--not helpful)
url <- "https://www1.nyc.gov/assets/planning/download/office/planning-level/nyc-population/acs/demo_201
demographics <- rio::import(file = url, which = "DemData")

demographics <- demographics[, c(
    "GeoID",
    "Pop65pl1P" # percent
)]

names(demographics) <- c("GeoID", "pct_65_plus")

demographics <- demographics %>%
```

```
    select(everything()) %>%
        mutate(BoroCD =
            case_when(
                substr(GeoID, 1, 2) == "MN" ~
                    as.numeric(paste0(1, substr(GeoID, 3, 4))),
                substr(GeoID, 1, 2) == "BX" ~
                    as.numeric(paste0(2, substr(GeoID, 3, 4))),
                substr(GeoID, 1, 2) == "BK" ~
                    as.numeric(paste0(3, substr(GeoID, 3, 4))),
                substr(GeoID, 1, 2) == "QN" ~
                    as.numeric(paste0(4, substr(GeoID, 3, 4))),
                substr(GeoID, 1, 2) == "SI" ~
                    as.numeric(paste0(5, substr(GeoID, 3, 4)))
            )
        )

districts <- left_join(
    x = districts,
    y = demographics,
    by = "BoroCD"
)
districts$GeoID <- NULL

benches <- left_join(
    x = benches,
    y = demographics,
    by = c("borocd" = "BoroCD")
)
benches$GeoID <- NULL

ggplot() +
    geom_sf(data = districts, aes(fill = pct_65_plus)) + # district boundaries
    scale_fill_viridis(option = "mako",
        "Percent of Residents \n Aged 65+"
    ) +
    new_scale_fill() +
    geom_sf(
        data = benches, aes(fill = color),
        color = "black", shape = 21, size = 0.6
    ) +
    scale_fill_identity(
        name = "",
        labels = "Bench",
        guide = "legend"
    ) +
    geom_text(
        data = districts_coords, aes(X, Y, label = BoroCD),
        colour = "white", size = 1, angle = 15
    ) +
    geom_sf(data = city, color = "black", fill = NA) + # boros outline
    labs(
        title = "Placement of Benches in Areas with Seniors",
        caption = "(Labels are Community District numbers)"
```

```
  ) +
  theme_void()
```



(Labels are Community District numbers)