councilverse

Data Example

Install Package

```
# replace councilverse with councildown if you only want councildown
remotes::install_github("newyorkcitycouncil/councilverse")
```

Load Package

```
library(tidyverse)
library(leaflet)
# load last; councilverse contains councildown
library(councilverse)
```

Example Project Using CouncilVerse, CouncilVerse (CouncilCount) Functions

Load CouncilCount Data

CouncilCount data is a part of the CouncilVerse package. We will use this data as an example of how to make plots and maps.

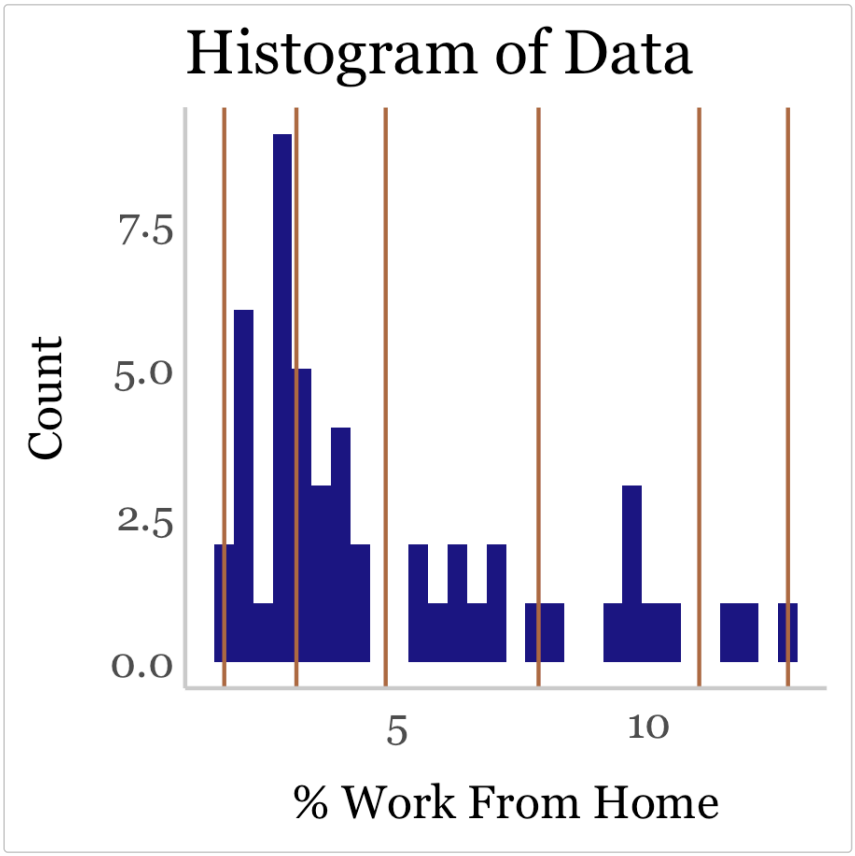
```
# councilverse::get_census_variables to check available variables
census_vars <- get_census_variables()
# councilverse::get_geo_estimates to read in council district data for % work from home using
        2013 council boundaries
# borough, communitydist, councildist, nta, policeprct, schooldist geographies are available
council_geo <- get_geo_estimates("councildist", var_codes = "DP03_0024E", boundary_year =
        "2013")
```

Visualize Data and Determine Map Breaks

```
# check distribution of data (no councildown additions yet)
wfh_hist <- ggplot(data = council_geo, aes(percent_work_from_home_workers_16_and_older)) +
        geom_histogram(bins = 30)
# use fisher-jenks natural breaks after visualizing data
nat_intvl = classInt::classIntervals(council_geo$percent_work_from_home_workers_16_and_older, n
        = 5, style = 'fisher')
```

```
# visualize breaks using councilcount functions and styling
ggplot(data = council_geo, aes(percent_work_from_home_workers_16_and_older)) +
  # use first color from nycc_palette "main" for the histogram bars
  geom_histogram(bins = 30, aes(fill=pal_nycc("main")[1])) +
  # use last color from nycc_palette "main" for the breaks
  geom_vline(xintercept = nat_intvl$brks, color = pal_nycc("main")[6]) +
  labs(
    x = "% Work From Home",
    y = "Count",
    title = "Histogram of Data",
    color = ""
  ) +
  # councildown::theme_nycc() for styling
  theme_nycc() +
  theme(legend.position="none")
#> Warning: The `scale_name` argument of `discrete_scale()` is deprecated as of ggplot2
```

```
#> 3.5.0.
#> This warning is displayed once every 8 hours.
#> Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
#> generated.
```



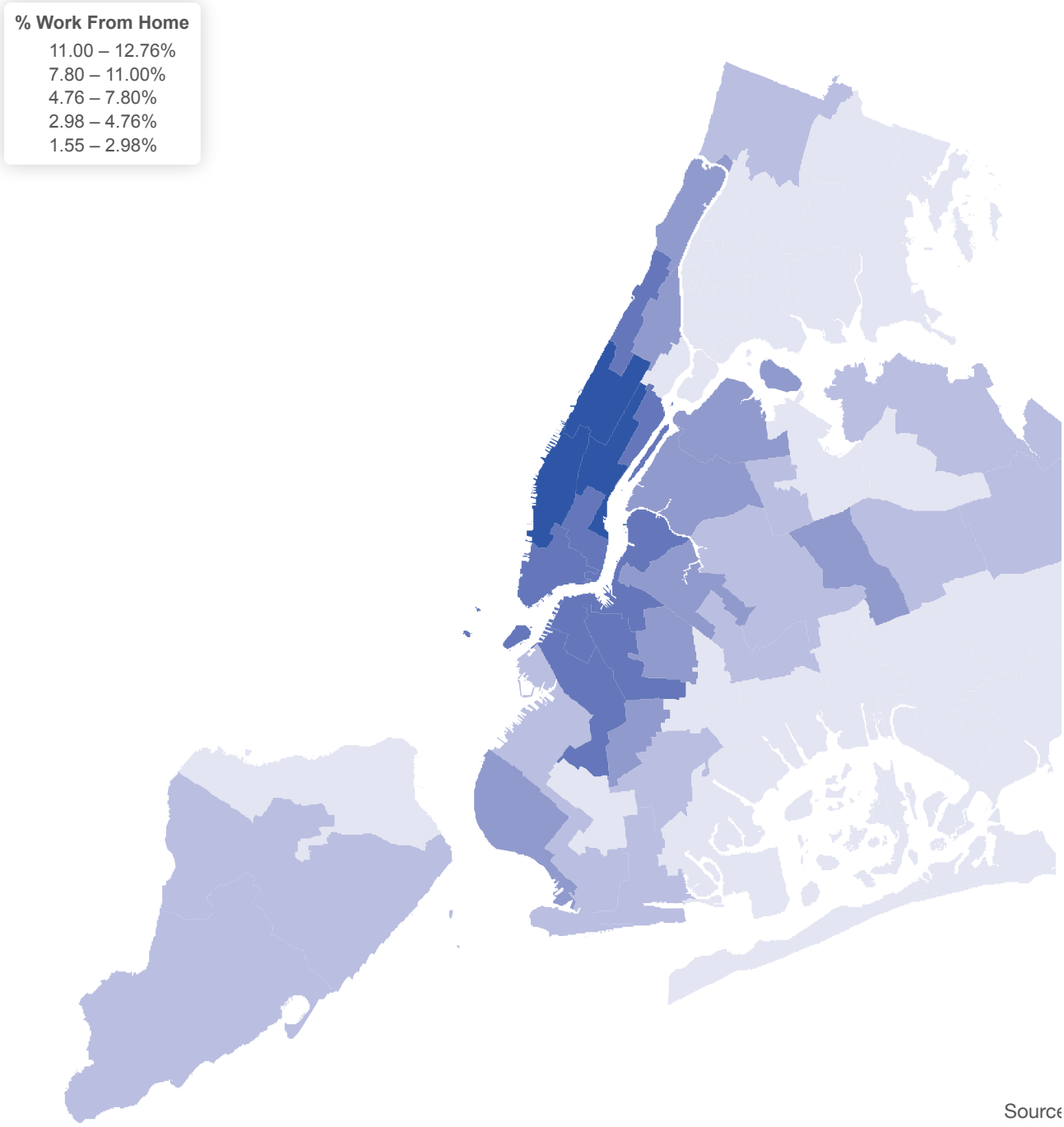
Make Palette for Map

```
# councildown::colorBin has default options for palette, bins, and na.color
pal_council = colorBin(
  # use nycc_blue, which is the default palette for councildown::colorBin
  palette = "nycc_blue",
  # note: make sure rounding doesn't leave out any data in the bottom or top bin
  bins = round(nat_intvl$brks,2),
  domain = council_geo$percent_work_from_home_workers_16_and_older,
  # use the default NA color for councildown::colorBin
  na.color = "#FFFFFF"
)
```

Create Map

```
map <- leaflet() %>%
  # councildown::addCouncilStyle() for styling
  addCouncilStyle() %>%
  # councildown::addPolygons has default smoothFactor and weight
  addPolygons(data = council_geo,
              fillColor = ~pal_council(percent_work_from_home_workers_16_and_older),
              fillOpacity = 1,
              # councildown::councilPopup to style popups
              label = ~lapply(councilPopup(
                paste0("<h3>", paste0("CD: ",councildist13), "</h3>",
                       "<p>", paste0("Work From Home: ",
       percent_work_from_home_workers_16_and_older, "%"), "</p>")),
                htmltools::HTML)) %>%
```

```r
  # councildown::addLegend_decreasing to change the default ordering of the legend
  addLegend_decreasing(position = "topleft", pal = pal_council,
                       title = paste0("% Work From Home"),
                       values = c(0,1), opacity = 1, decreasing = T,
                       labFormat = labelFormat(suffix = "%"),
                       na.label = "NA") %>%
  # councildown::addSourceText to add text on bottom left for source
  addSourceText("Source: NYCC Data Team")
map
```




```r
# add districts for 2013 or 2023
map_w_dists <- map %>%
  # councildown::addCouncilStyle has options to add 2013 or 2023 council districts to map
  addCouncilStyle(add_dists = T, dist_year = "2023")
```

```
# note: can access cd shapefiles by themselves
councildown::nycc_cd_13 # 2013 council district lines
councildown::nycc_cd_23 # 2023 council district lines
```

Save Map

```
# councildown::mapshot has default zoom, vwidth, vheight, remove_controls
map_png <- file_name_generator(description = "council_district_percent_work_from_home",
        date_year = 2022, file_extension = ".png")
mapshot(map, file = glue::glue("visuals/{map_png}"))

map_html <- file_name_generator(description = "council_district_percent_work_from_home",
        date_year = 2022, file_extension = ".html")
# save leaflet html (function not a part of councilverse)
htmlwidgets::saveWidget(map, file=glue::glue("visuals/{map_html}"), selfcontained = T)
```

Reference

CouncilVerse

Graphs and Tables

`encodeGraphic()` – Graphics to raw HTML `gt_table()` – gt table to raw HTML helper function

Misc

`file_name_generator()` – Helper function for creating file names using accepted guidelines `unzip_sf()` – Helper function for loading in shapefiles that come zipped

CouncilCount

Population Estimates

`get_bbl_estimates()` – Generates a dataframe that provides population estimates at the point level (there are also columns for various other geographies, like council district)

`get_census_variables()` – Provides information on all of the ACS demographic variables that can be accessed using `get_geo_estimates()` via variable codes

`get_geo_estimates()` – Creates a dataframe that provides population estimates for selected demographic variables along chosen geographic boundaries (e.g. council district, borough, etc.)

CouncilDown

Plots

`theme_nycc()` – A `ggplot2` theme that uses Council fonts and style guidelines to produce plots

`scale_fill_nycc()`, `scale_color_nycc()` – Color and fill scales for `ggplot2` plots. Used to replace `scale_color_discrete()`, `scale_color_continuous()`, `scale_fill_discrete()`, and `scale_fill_continuous()`

Colors

`pal_nycc()` – Makes a color palette using Council colors

`nycc_pal()` DEPRECATED. Use `pal_nycc` instead

Maps

`addCouncilStyle()` – Adds Council styles including tiles, council district outlines, and fonts to a `leaflet` map

`addSourceText()` – Adds Council “Source” note in the bottom right of `leaflet` map

`addPolygons()` – The `leaflet` function `addPolygons` with default `smoothFactor` and `weight`

`colorBin()` – The `leaflet` function `colorBin` with default `palette`, `bins`, `na.color`

`councilPopup()` – Basic styling for `leaflet` popups

`mapshot()` – Saves `leaflet` maps as a static png image and defaults map saves with standardized `width`, `height`, and `zoom`

`addLegend_decreasing()` – The `leaflet` function `addLegend` but with the option for the highest number to be at the top of the legend

Data

`nycc_cd_13` – 2013-2023 NYC Council District sf collection

`nycc_cd_23` – 2023-2033 NYC Council District sf collection

Templates

`council_html()` – Render a Council HTML report `council_pdf()` – Render a Council PDF report

Misc

`pretty_date` – Make a pretty date