Census Tract Visualizations

2025-02-26

```
# Load libraries
library(ggplot2)
library(sf)
```

```
## Linking to GEOS 3.8.0, GDAL 3.0.4, PROJ 6.3.1; sf_use_s2() is TRUE
```

```
library(tigris)
```

```
## To enable caching of data, set `options(tigris_use_cache = TRUE)`
## in your R script or .Rprofile.
```

```
library(paletteer)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(fuzzyjoin)
```

```
##
## Attaching package: 'fuzzyjoin'
```

```
## The following object is masked from 'package:tigris':
##
##     geo_join
```

```
# Load Census tracts for Lee County
lee_county_tracts <- tracts(state = "FL", county = "Lee", year = 2019, cb = TRUE)
```

```
##   |                                                                              |
```

```
lee_county_tracts <- lee_county_tracts %>%
  mutate(NAME = as.numeric(NAME)) %>%
  rename(tract = NAME)

lee_county_tracts_df <- st_drop_geometry(lee_county_tracts)

# Load bus stops and tract data
tract_data <- read.csv("r_tract.csv")
tract_cordinates <- tract_data[, c("centlat", "centlong", "intptlat", "intptlong", "tract", "distance_fi
missing_tract <- read.csv("missing_tract.csv")
```

```r
# Identify tracts in missing_tract that are not in tract_cordinates
missing_in_cordinates <- missing_tract %>%
  filter(!tract %in% tract_cordinates$tract)

# Combine the original data with the new missing tracts
tract_cordinates <- bind_rows(tract_cordinates, missing_tract)

new_tract_coordinates <- tract_cordinates %>%
  dplyr::select(-intptlat, -intptlong, -distance_from_closest_stop_in_miles)

# Join new tract coordinates with lee_county_tracts_df
lee_county_tracts_joined <- lee_county_tracts_df %>%
  left_join(
    new_tract_coordinates %>%
      select(tract, distance_from_closest_stop_feet, poverty_prct, centlat, centlong),
    by = "tract"  # Specify the column to join by
  )

# Join geometry from original lee_county_tracts
lee_county_tracts_joined <- lee_county_tracts_joined %>%
  left_join(
    lee_county_tracts %>%
      select(tract, geometry),
    by = "tract"  # Specify the column to join by
  )

# Convert to sf object
lee_county_tracts_joined <- st_as_sf(lee_county_tracts_joined)

# Fill missing geometries using centlat and centlong
missing_geom_indices <- is.na(lee_county_tracts_joined$geometry)

# Create point geometry for rows with missing geometries
lee_county_tracts_joined$geometry[missing_geom_indices] <-
  st_sfc(mapply(function(long, lat) st_point(c(long, lat)),
                lee_county_tracts_joined$centlong[missing_geom_indices],
                lee_county_tracts_joined$centlat[missing_geom_indices],
                SIMPLIFY = FALSE))

# Remove centlat and centlong columns
lee_county_tracts_joined <- lee_county_tracts_joined %>%
  select(-centlat, -centlong)

# Calculate miles and remove NA values for further analysis
lee_county_tracts_joined <- lee_county_tracts_joined %>%
  mutate(miles = distance_from_closest_stop_feet / 5280) %>%
  filter(!is.na(miles)) # Remove rows with NA in miles
```

```r
# Define breaks and colors for plotting
breaks <- c(0, 0.25, 0.50, 0.75, 1, max(lee_county_tracts_joined$miles, na.rm = TRUE))
colors <- c("#1e3e70", "#419c33", "#6bcf00", "#f0b514", "#be533e")

# Create the plot
distance_plot <- ggplot(data = lee_county_tracts_joined) +
```

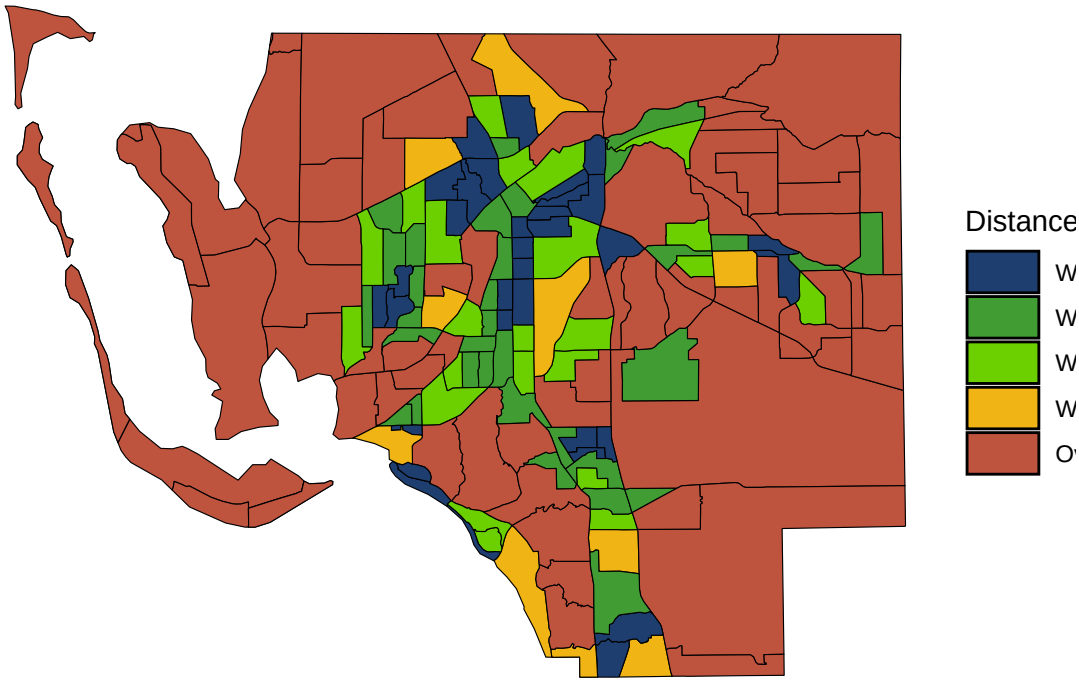
```
  geom_sf(aes(fill = cut(miles, breaks = breaks, include.lowest = TRUE)),
          color = "black", size = 0.1) +
  scale_fill_manual(values = colors,
                    name = "Distance (miles)",
                    labels = c("Within 1/4 Mile", "Within 1/2 Mile", "Within 3/4 Mile", "Within 1 Mile"
  labs(title = "Distance to Closest Bus Stop by Census Tract",
       subtitle = "Lee County, FL") +
  theme_minimal() +
  theme(
    legend.position = c(1, 0.5), # Move legend further to the right
    legend.justification = c("left", "center"), # Align legend to the left
    legend.key.size = unit(0.6, "cm"), # Reduce size of legend keys
    legend.key.width = unit(1, "cm"), # Adjust width of legend keys
    legend.box.margin = margin(10, 10, 10, 10),
    panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(),
    axis.title.x = element_blank(),
    axis.title.y = element_blank(),
    axis.text.x = element_blank(),
    axis.text.y = element_blank()
  ) +
  guides(fill = guide_legend(ncol = 1)) # Set legend to be a vertical list
```

```
## Warning: A numeric `legend.position` argument in `theme()` was deprecated in ggplot2
## 3.5.0.
## i Please use the `legend.position.inside` argument of `theme()` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

```
print(distance_plot)
```

Distance to Closest Bus Stop by Census Tract

Lee County, FL

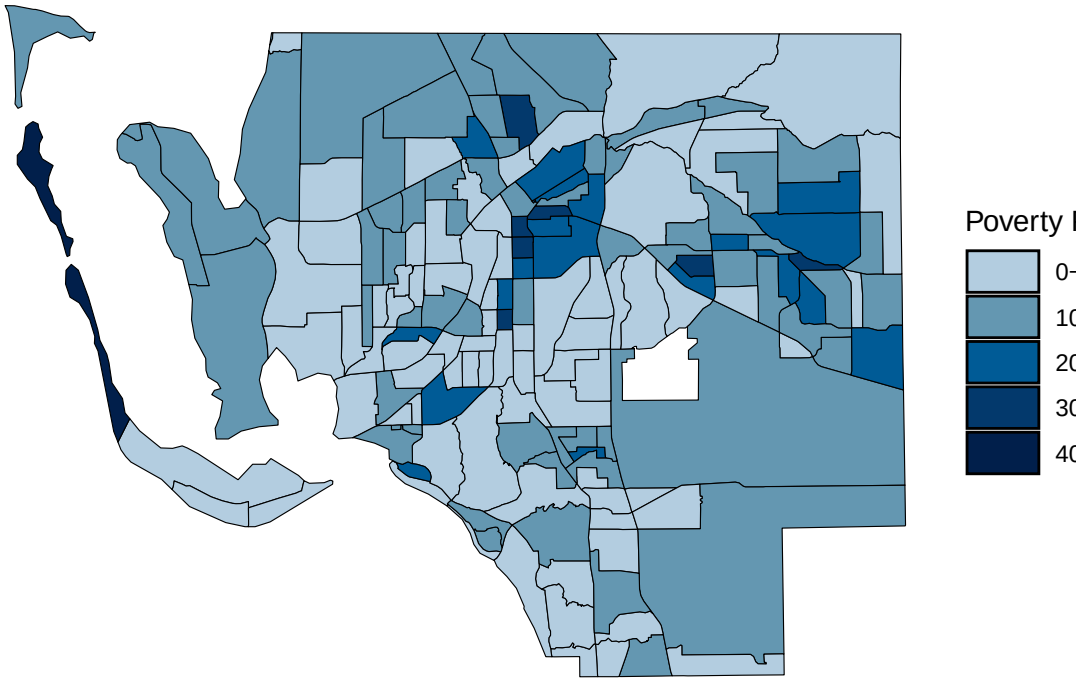



```
#join for next graphs
lee_county_tracts_joined <- lee_county_tracts_joined %>%
  left_join(tract_data %>% select(tract, total_pop), by = "tract") %>%
  filter(!is.na(poverty_prct)) # Keep filtering for NA in poverty_prct

# Define breaks for poverty percentage buckets
poverty_breaks <- c(0, 10, 20, 30, 40, 50, Inf)  # Adjust this as needed
poverty_colors <- c("#b3cde0", "#6497b1", "#005b96", "#03396c", "#011f4b")


# Create the plot
poverty_plot <- ggplot(data = lee_county_tracts_joined) +
  geom_sf(aes(fill = cut(poverty_prct, breaks = poverty_breaks, include.lowest = TRUE)),
          color = "black", size = 0.1) + # Fill based on poverty percentage buckets
  scale_fill_manual(values = poverty_colors,
                    name = "Poverty Percentage",
                    labels = c("0-10%", "10-20%", "20-30%", "30-40%", "40-50%")) + # Custom labels for
  labs(title = "Poverty Percentage by Census Tract",
       subtitle = "Lee County, FL") +
  theme_minimal() +
  theme(
    legend.position = c(1, 0.5), # Move legend further to the right
    legend.justification = c("left", "center"), # Align legend to the left
    legend.key.size = unit(0.6, "cm"), # Reduce size of legend keys
    legend.key.width = unit(1, "cm"), # Adjust width of legend keys
    legend.box.margin = margin(10, 10, 10, 10),
    panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(),
```

```
    axis.title.x = element_blank(),
    axis.title.y = element_blank(),
    axis.text.x = element_blank(),
    axis.text.y = element_blank()
  ) +
  guides(fill = guide_legend(ncol = 1)) # Set legend to be a vertical list

# Display the plot
print(poverty_plot)
```

Poverty Percentage by Census Tract

Lee County, FL




```
pop_breaks <- c(0, 500, 1500, 3500, 6500, 10000, Inf)  # Adjust this as needed
pop_colors <- c("#fefdce", "#cae8b4", "#a7e1a1", "#7ecdba", "#41b6c6", "#2e7fb9")  # Adding a color for

# Create the plot
pop_plot <- ggplot(data = lee_county_tracts_joined) +
  geom_sf(aes(fill = cut(total_pop, breaks = pop_breaks, include.lowest = TRUE)),
          color = "black", size = 0.1) + # Fill based on population buckets
  scale_fill_manual(values = pop_colors,
                    name = "Total Population",
                    labels = c("0-500", "500-1,500", "1,500-3,500", "3,500-6,500", "6,500-10,000", "Abo
  labs(title = "Total Population by Census Tract",
       subtitle = "Lee County, FL") +
  theme_minimal() +
  theme(
    legend.position = c(1, 0.5), # Move legend further to the right
    legend.justification = c("left", "center"), # Align legend to the left
    legend.key.size = unit(0.6, "cm"), # Reduce size of legend keys
    legend.key.width = unit(1, "cm"), # Adjust width of legend keys
```

```
    legend.box.margin = margin(10, 10, 10, 10),
    panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(),
    axis.title.x = element_blank(),
    axis.title.y = element_blank(),
    axis.text.x = element_blank(),
    axis.text.y = element_blank()
  ) +
  guides(fill = guide_legend(ncol = 1)) # Set legend to be a vertical list

# Display the plot
print(pop_plot)
```