Binning Census Tracts by Index Percentiles: Most Vulnerable Census Tracts

Matthew Menon

2025-06-12

Setup

```
library(readr)
library(dplyr)
library(factoextra)
library(exactextractr)
library(tigris)
library(ggplot2)
library(plotly)
library(sf)
library(tibble)
library(GGally)
```

```
setwd("C:/Users/matth/Desktop/Undergraduate-Research/GWU-Bootcamp/Final Project")
data = read_csv("Data/flood_vulnerability_scores.csv",
                          col_types = cols(GEOID = col_character()))
head(data)
```

```
## # A tibble: 6 × 27
##   GEOID       floodplain_500 floodplain_100 tidal_floodplain blue_zone   sso
##   <chr>                <dbl>          <dbl>            <dbl>     <dbl> <dbl>
## 1 11001004001           5.65           4.46                0      2.35     1
## 2 11001004002           0              0                   0      1.85     0
## 3 11001003600           0              0                   0      2.25     0
## 4 11001004201           0              0                   0     10.1      1
## 5 11001004202           0              0                   0     11.2      2
## 6 11001007407           0              0                   0      1.56     8
## # ℹ 21 more variables: ground_elevation <dbl>, base_elevation <dbl>,
## #   one_hundred_year_floodplain_risk <chr>, dist_to_water <dbl>, asthma <dbl>,
## #   diabetes <dbl>, pct_poverty <dbl>, percent_vulnerable <dbl>,
## #   pct_unemp <dbl>, num_fire_stations <dbl>, num_hospitals <dbl>,
## #   num_police_stations <dbl>, num_cross_guards <dbl>, pct_minority <dbl>,
## #   pct_old_housing <dbl>, pct_raster_407m <dbl>, pct_buildings_407m <dbl>,
## #   flood_index <dbl>, social_health_index <dbl>, infra_index <dbl>, …
```

```
# Load DC census tracts shapefile from tigris
dc_tracts <- tracts(state = "DC", year = 2020, class = "sf")
```

```
##   |                                                                      |
|   0%  |                                                                     |========
|  12%  |                                                                     |========
=                                                    |  12%  |
|=========                                                          |  13%  |
|==========                                                         |  14%  |
|============                                                       |  18%  |
|================                                                   |  22%  |
|====================                                               |  29%  |
|============================                                       |  38%  |
|================================                                   |  46%  |
|====================================                               |  52%  |
|========================================                           |  55%  |
|========================================                           |  56%  |
|=============================================                      |  67%  |
|=========================================================          |  86%  |
|======================================================================| 100%
```

Flood Vulnerability Index

Create a new variable called `flood_risk` within the newly created `flood_data` data frame that is a categorical variable of the `flood_index` variable, binned by percentiles. Least Concern represents census tracts not flood susceptible, all the way to Greatest Concern representing census tracts with the highest flood susceptibility.

```
flood_data = data %>%
  mutate(flood_risk = factor(ntile(flood_index, 4),
                             labels = c("Least Concern", "Low Risk", "Moderate Risk", "Greates
t Concern")), tooltip_text = paste0("GEOID: ", GEOID, "<br>",
                             "Flood Susceptibility Index: ", round(flood_index,1), "<br>",
                             "Census Tract Flood Risk: ", flood_risk))
head(flood_data)
```

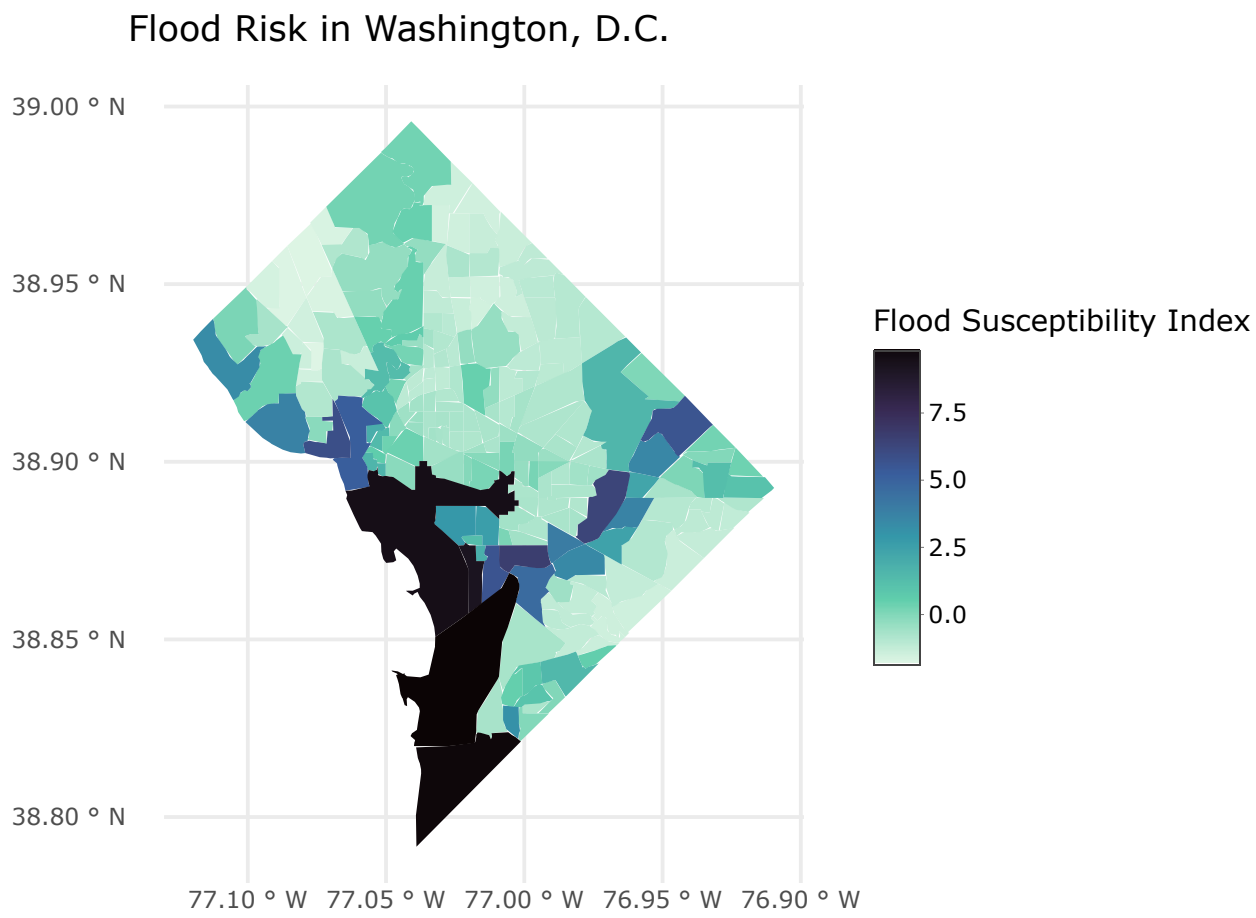
```
## # A tibble: 6 × 29
##   GEOID       floodplain_500 floodplain_100 tidal_floodplain blue_zone   sso
##   <chr>                <dbl>          <dbl>            <dbl>     <dbl> <dbl>
## 1 11001004001           5.65           4.46                0      2.35     1
## 2 11001004002           0              0                   0      1.85     0
## 3 11001003600           0              0                   0      2.25     0
## 4 11001004201           0              0                   0     10.1      1
## 5 11001004202           0              0                   0     11.2      2
## 6 11001007407           0              0                   0      1.56     8
## # ℹ 23 more variables: ground_elevation <dbl>, base_elevation <dbl>,
## #   one_hundred_year_floodplain_risk <chr>, dist_to_water <dbl>, asthma <dbl>,
## #   diabetes <dbl>, pct_poverty <dbl>, percent_vulnerable <dbl>,
## #   pct_unemp <dbl>, num_fire_stations <dbl>, num_hospitals <dbl>,
## #   num_police_stations <dbl>, num_cross_guards <dbl>, pct_minority <dbl>,
## #   pct_old_housing <dbl>, pct_raster_407m <dbl>, pct_buildings_407m <dbl>,
## #   flood_index <dbl>, social_health_index <dbl>, infra_index <dbl>, …
```

Merges `flood_data` data frame with geographical information from the `dc_tracts` data frame.

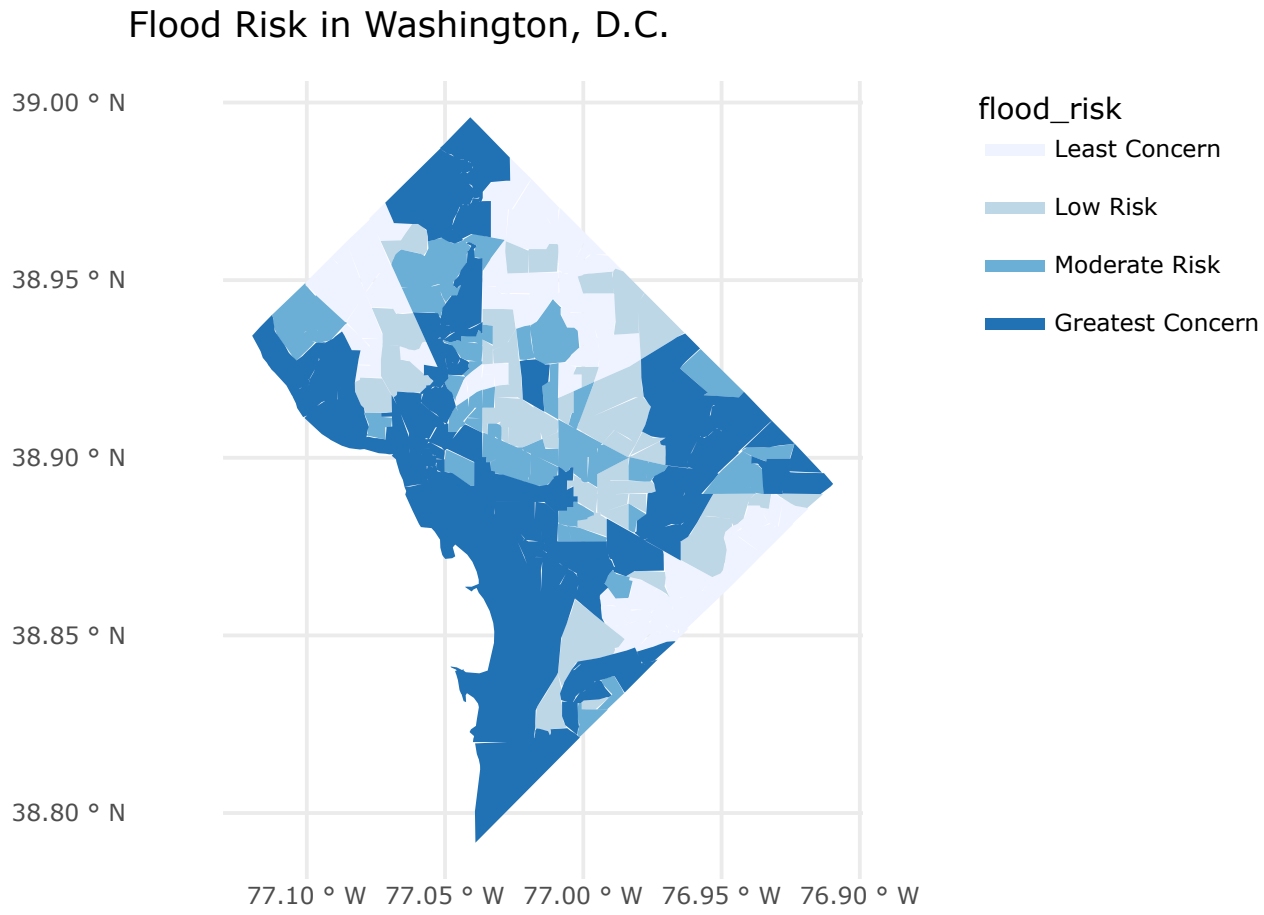
```
# Merge flood data with spatial data
flood_map <- dc_tracts %>%
  left_join(flood_data, by = "GEOID")
```

Creates two plots, the first is a simple mapping of the flood susceptibility index for each census tract in Washington D.C. as a raw value. The second is mapping the categorical variable representing four percentiles of the flood susceptibility index for each census tract. Note the census tracts of greatest concern and the census tracts of least concern.

```
# Flood plot
flood_plot <- ggplot(flood_map, aes(text = tooltip_text)) +
  geom_sf(aes(fill = flood_index), color = NA) +
  scale_fill_viridis_c(option = "mako", name = "Flood Susceptibility Index", direction = -1) +
  labs(title = "Flood Risk in Washington, D.C.",
       subtitle = "Census Tracts (2020)",
       caption = "Source: Open Data DC") +
  theme_minimal()

ggplotly(flood_plot, tooltip = "text")
```



```
flood_plot <- ggplot(flood_map, aes(text = tooltip_text)) +
  geom_sf(aes(fill = flood_risk), color = NA) +
  scale_fill_brewer(palette = "Blues") +
  labs(title = "Flood Risk in Washington, D.C.",
       subtitle = "Census Tracts (2020)",
       caption = "Source: Open Data DC") +
  theme_minimal()

# Step 3: Convert to interactive
ggplotly(flood_plot, tooltip = "text")
```



Social and Health Vulnerability Index

Create a new variable called `social_health_risk` within the newly created `social_health_data` data frame that is a categorical variable of the `social_health_index` variable, binned by percentiles. Least Vulnerable represents census tracts not socially vulnerable or susceptible to negative health outcomes, all the way to Greatest Vulnerability representing census tracts with the highest health vulnerability and percentages of individuals in socially vulnerable groups.

```
social_health_data = data %>%
  mutate(social_health_risk = factor(ntile(social_health_index, 4),
                                     labels = c("Least Vulnerable", "Low Vulnerability", "Moderate Vul
nerability", "Greatest Vulnerability")), tooltip_text = paste0("GEOID: ", GEOID, "<br>",
                                     "Social and Health Vulnerability Index: ", round(social_health_in
dex,1), "<br>",
                                     "Census Tract Social and Health Vulnerability: ", social_health_r
isk))
head(social_health_data)
```

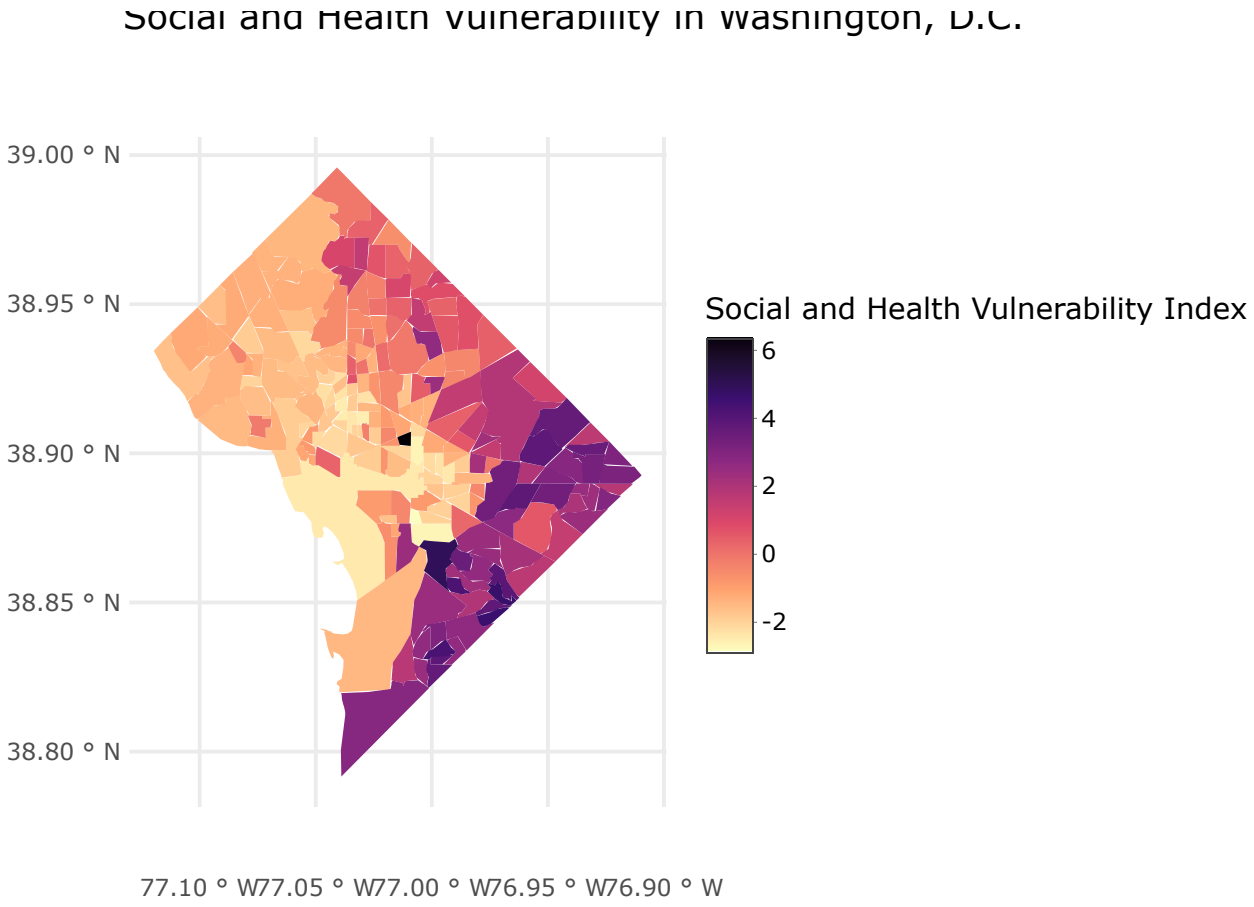
```
## # A tibble: 6 × 29
##   GEOID       floodplain_500 floodplain_100 tidal_floodplain blue_zone   sso
##   <chr>                <dbl>          <dbl>            <dbl>     <dbl> <dbl>
## 1 11001004001           5.65           4.46                0      2.35     1
## 2 11001004002           0              0                   0      1.85     0
## 3 11001003600           0              0                   0      2.25     0
## 4 11001004201           0              0                   0     10.1      1
## 5 11001004202           0              0                   0     11.2      2
## 6 11001007407           0              0                   0      1.56     8
## # ℹ 23 more variables: ground_elevation <dbl>, base_elevation <dbl>,
## #   one_hundred_year_floodplain_risk <chr>, dist_to_water <dbl>, asthma <dbl>,
## #   diabetes <dbl>, pct_poverty <dbl>, percent_vulnerable <dbl>,
## #   pct_unemp <dbl>, num_fire_stations <dbl>, num_hospitals <dbl>,
## #   num_police_stations <dbl>, num_cross_guards <dbl>, pct_minority <dbl>,
## #   pct_old_housing <dbl>, pct_raster_407m <dbl>, pct_buildings_407m <dbl>,
## #   flood_index <dbl>, social_health_index <dbl>, infra_index <dbl>, …
```

Merges `social_health_data` data frame with geographical information from the `dc_tracts` data frame.

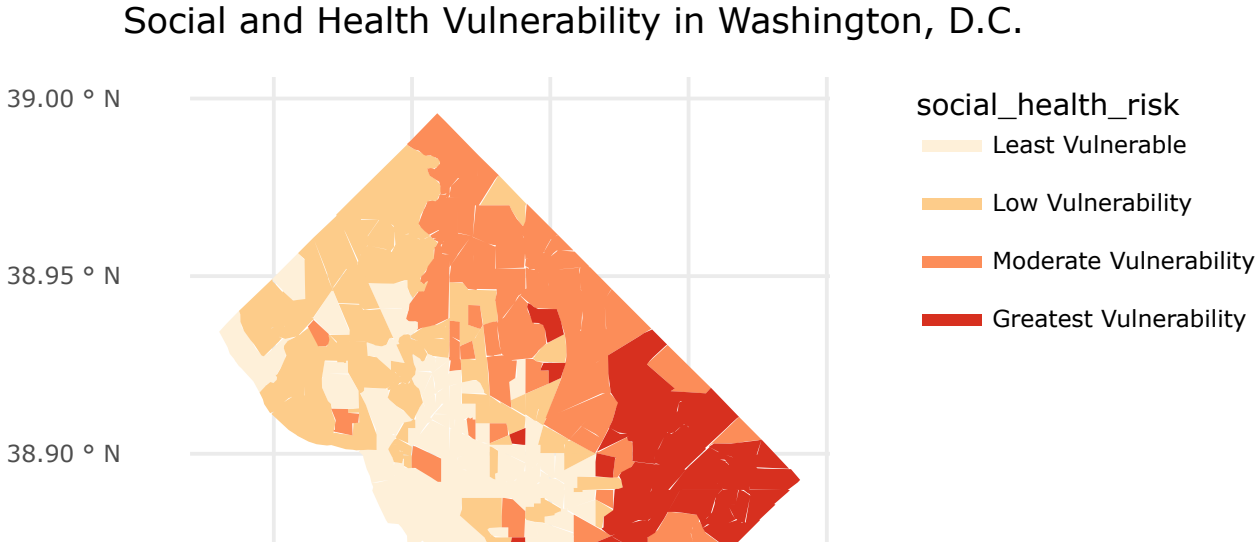
```
# Merge flood data with spatial data
social_health_map <- dc_tracts %>%
  left_join(social_health_data, by = "GEOID")
```

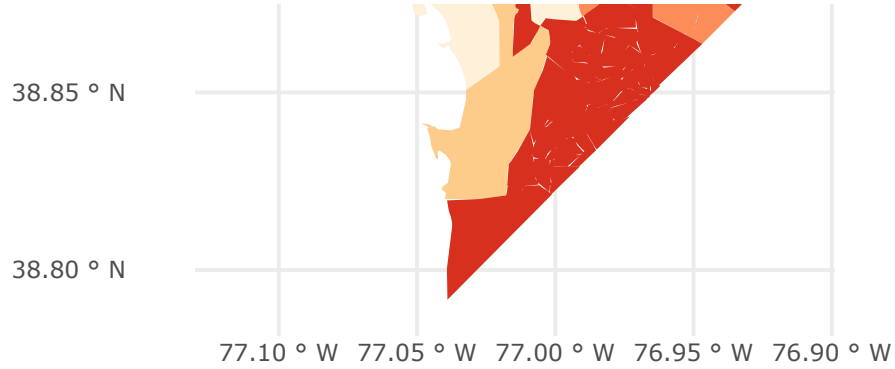
Creates two plots, the first is a simple mapping of the social and health vulnerability index for each census tract in Washington D.C. as a raw value. The second is mapping the categorical variable representing four percentiles of the social and health vulnerability index for each census tract. Note the most vulnerable and least vulnerable census tracts.

```
# Social Health plot
social_health_plot <- ggplot(social_health_map, aes(text = tooltip_text)) +
  geom_sf(aes(fill = social_health_index), color = NA) +
  scale_fill_viridis_c(option = "magma", name = "Social and Health Vulnerability Index", directi
on = -1) +
  labs(title = "Social and Health Vulnerability in Washington, D.C.",
       subtitle = "Census Tracts (2020)",
       caption = "Source: Open Data DC") +
  theme_minimal()

ggplotly(social_health_plot, tooltip = "text")
```




```
social_health_plot <- ggplot(social_health_map, aes(text = tooltip_text)) +
  geom_sf(aes(fill = social_health_risk), color = NA) +
  scale_fill_brewer(palette = "OrRd") +
  labs(title = "Social and Health Vulnerability in Washington, D.C.",
       subtitle = "Census Tracts (2020)",
       caption = "Source: Open Data DC") +
  theme_minimal()

# Step 3: Convert to interactive
ggplotly(social_health_plot, tooltip = "text")
```





Infrastructure Index

Create a new variable called `infrastructure_presence` within the newly created `infrastructure_data` data frame that is a categorical variable of the `infra_index` variable, binned by percentiles. Least Concern represents census tracts not flood susceptible, all the way to Greatest Concern representing census tracts with the highest flood susceptibility.

```
infrastructure_data = data %>%
  mutate(infrastructure_presence = factor(ntile(infra_index, 4),
                                labels = c("Greatest Presence of Infrastructure", "Presence of In
frastructure", "Lack of Infrastructure", "Greatest Lack of Infrastructure")), tooltip_text = pas
te0("GEOID: ", GEOID, "<br>",
                                "Infrastructure Index: ", round(infra_index,1), "<br>",
                                "Census Tract Infrastructure Presence: ", infrastructure_presenc
e))
head(infrastructure_data)
```

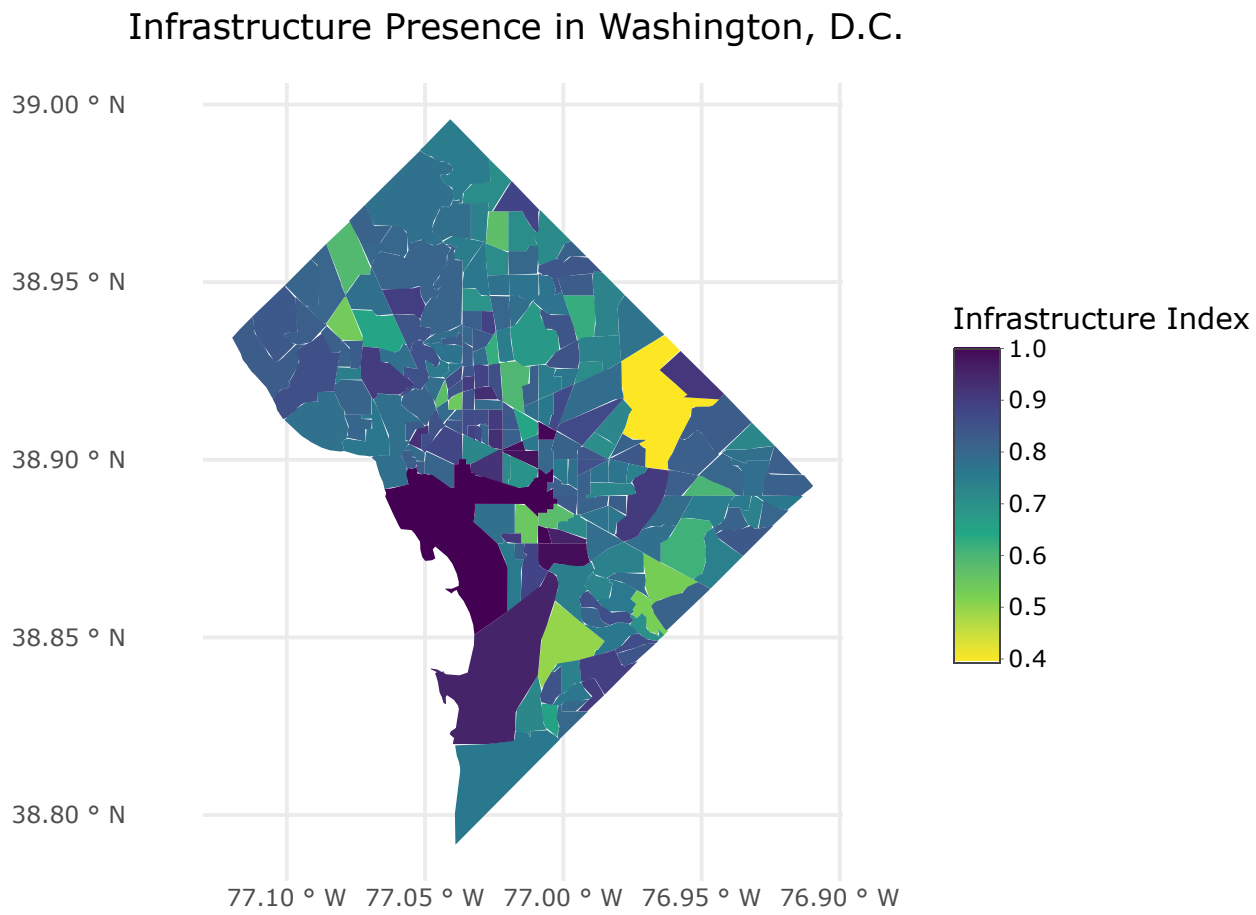
```
## # A tibble: 6 × 29
##   GEOID       floodplain_500 floodplain_100 tidal_floodplain blue_zone   sso
##   <chr>                <dbl>          <dbl>            <dbl>     <dbl> <dbl>
## 1 11001004001           5.65           4.46                0      2.35     1
## 2 11001004002           0              0                   0      1.85     0
## 3 11001003600           0              0                   0      2.25     0
## 4 11001004201           0              0                   0     10.1      1
## 5 11001004202           0              0                   0     11.2      2
## 6 11001007407           0              0                   0      1.56     8
## # ℹ 23 more variables: ground_elevation <dbl>, base_elevation <dbl>,
## #   one_hundred_year_floodplain_risk <chr>, dist_to_water <dbl>, asthma <dbl>,
## #   diabetes <dbl>, pct_poverty <dbl>, percent_vulnerable <dbl>,
## #   pct_unemp <dbl>, num_fire_stations <dbl>, num_hospitals <dbl>,
## #   num_police_stations <dbl>, num_cross_guards <dbl>, pct_minority <dbl>,
## #   pct_old_housing <dbl>, pct_raster_407m <dbl>, pct_buildings_407m <dbl>,
## #   flood_index <dbl>, social_health_index <dbl>, infra_index <dbl>, …
```

Merges `infrastructure_data` data frame with geographical information from the `dc_tracts` data frame.

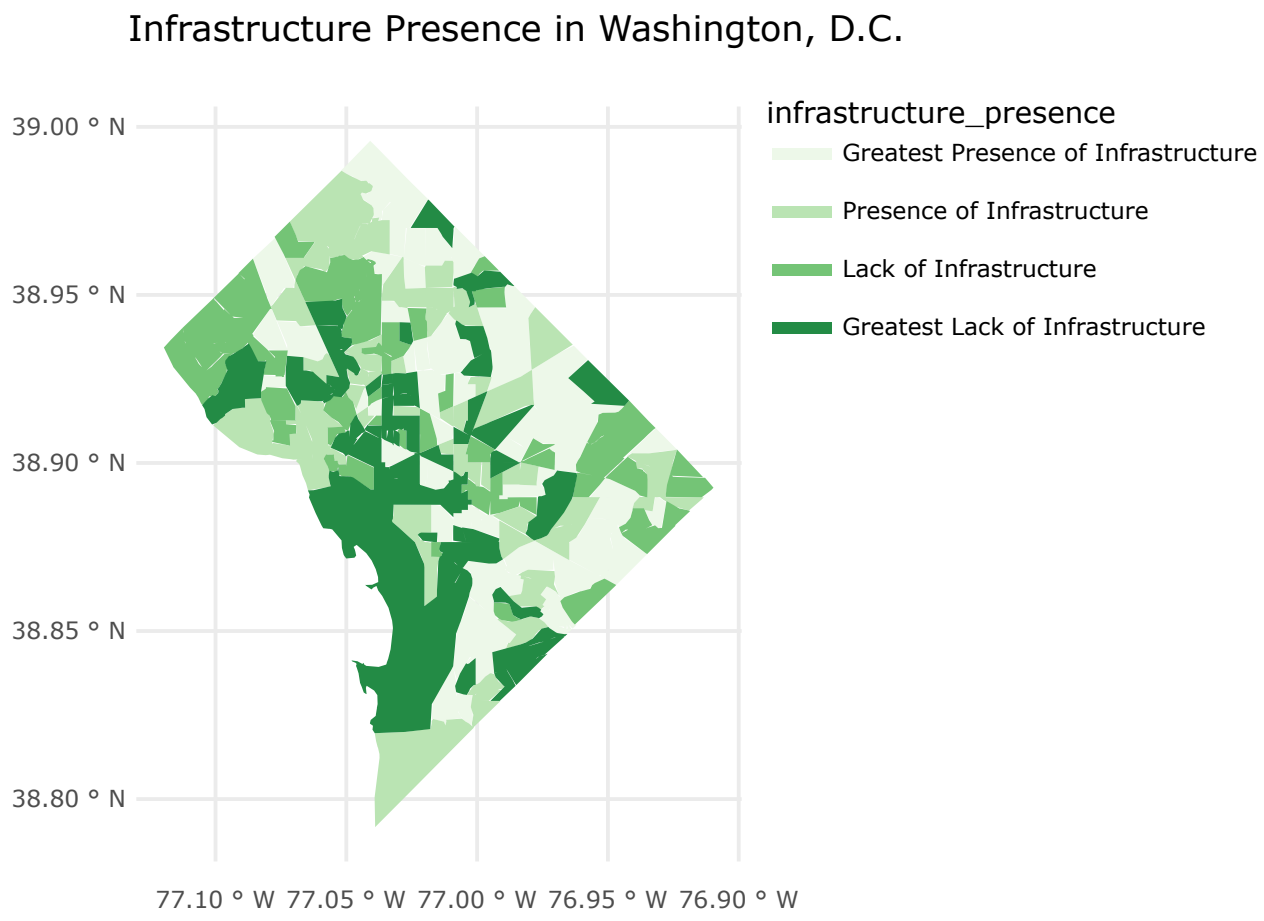
```
# Merge flood data with spatial data
infrastructure_map <- dc_tracts %>%
  left_join(infrastructure_data, by = "GEOID")
```

Creates two plots, the first is a simple mapping of the infrastructure index for each census tract in Washington D.C. as a raw value. The second is mapping the categorical variable representing four percentiles of the infrastructure index for each census tract. Note the tracts with highest and lowest presence of infrastructure.

```
# Infrastructure plot
infrastructure_plot <- ggplot(infrastructure_map, aes(text = tooltip_text)) +
  geom_sf(aes(fill = infra_index), color = NA) +
  scale_fill_viridis_c(option = "viridis", name = "Infrastructure Index", direction = -1) +
  labs(title = "Infrastructure Presence in Washington, D.C.",
       subtitle = "Census Tracts (2020)",
       caption = "Source: Open Data DC") +
  theme_minimal()

ggplotly(infrastructure_plot, tooltip = "text")
```



```
infrastructure_plot <- ggplot(infrastructure_map, aes(text = tooltip_text)) +
  geom_sf(aes(fill = infrastructure_presence), color = NA) +
  scale_fill_brewer(palette = "Rd") +
  labs(title = "Infrastructure Presence in Washington, D.C.",
       subtitle = "Census Tracts (2020)",
       caption = "Source: Open Data DC") +
  theme_minimal()

# Step 3: Convert to interactive
ggplotly(infrastructure_plot, tooltip = "text")
```



Composite Index

Create a new variable called `composite_rank` within the newly created `composite_data` data frame that is a categorical variable of the `composite_index` variable, binned by percentiles. Least Overall Vulneraibility represents census tracts with the lowest composite vulnerability based on flood susceptibility, social and health vulnerability, and infrastructure presence, all the way to Greatest Overall Vulnerability representing census tracts with highest composite vulnerability based on the same factors.

```
composite_data = data %>%
  mutate(composite_rank = factor(ntile(composite_index, 4),
                                 labels = c("Least Overall Vulnerability", "Low Overall Vulnerabl
e", "Moderate Overall Vulnerability", "Greatest Overall Vulnerability")), tooltip_text = paste0
("GEOID: ", GEOID, "<br>",
                                 "Composite Vulnerability Index: ", round(composite_index,1), "<br
>",
                                 "Census Tract Infrastructure Index: ", composite_rank))
head(composite_data)
```

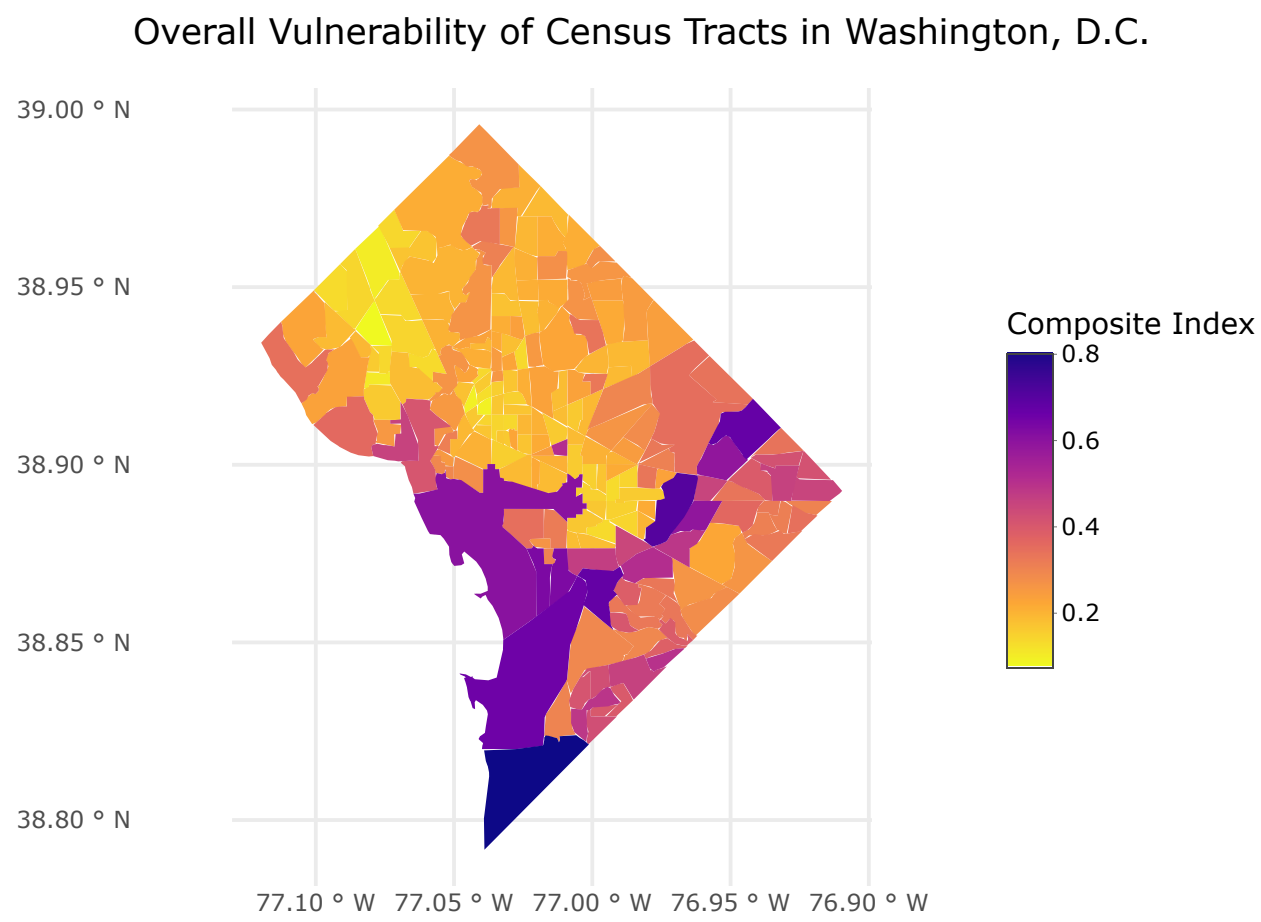
```
## # A tibble: 6 × 29
##   GEOID       floodplain_500 floodplain_100 tidal_floodplain blue_zone   sso
##   <chr>                <dbl>          <dbl>            <dbl>     <dbl> <dbl>
## 1 11001004001           5.65           4.46                0      2.35     1
## 2 11001004002           0              0                   0      1.85     0
## 3 11001003600           0              0                   0      2.25     0
## 4 11001004201           0              0                   0     10.1      1
## 5 11001004202           0              0                   0     11.2      2
## 6 11001007407           0              0                   0      1.56     8
## # ℹ 23 more variables: ground_elevation <dbl>, base_elevation <dbl>,
## #   one_hundred_year_floodplain_risk <chr>, dist_to_water <dbl>, asthma <dbl>,
## #   diabetes <dbl>, pct_poverty <dbl>, percent_vulnerable <dbl>,
## #   pct_unemp <dbl>, num_fire_stations <dbl>, num_hospitals <dbl>,
## #   num_police_stations <dbl>, num_cross_guards <dbl>, pct_minority <dbl>,
## #   pct_old_housing <dbl>, pct_raster_407m <dbl>, pct_buildings_407m <dbl>,
## #   flood_index <dbl>, social_health_index <dbl>, infra_index <dbl>, …
```

Merges `composite_data` data frame with geographical information from the `dc_tracts` data frame.

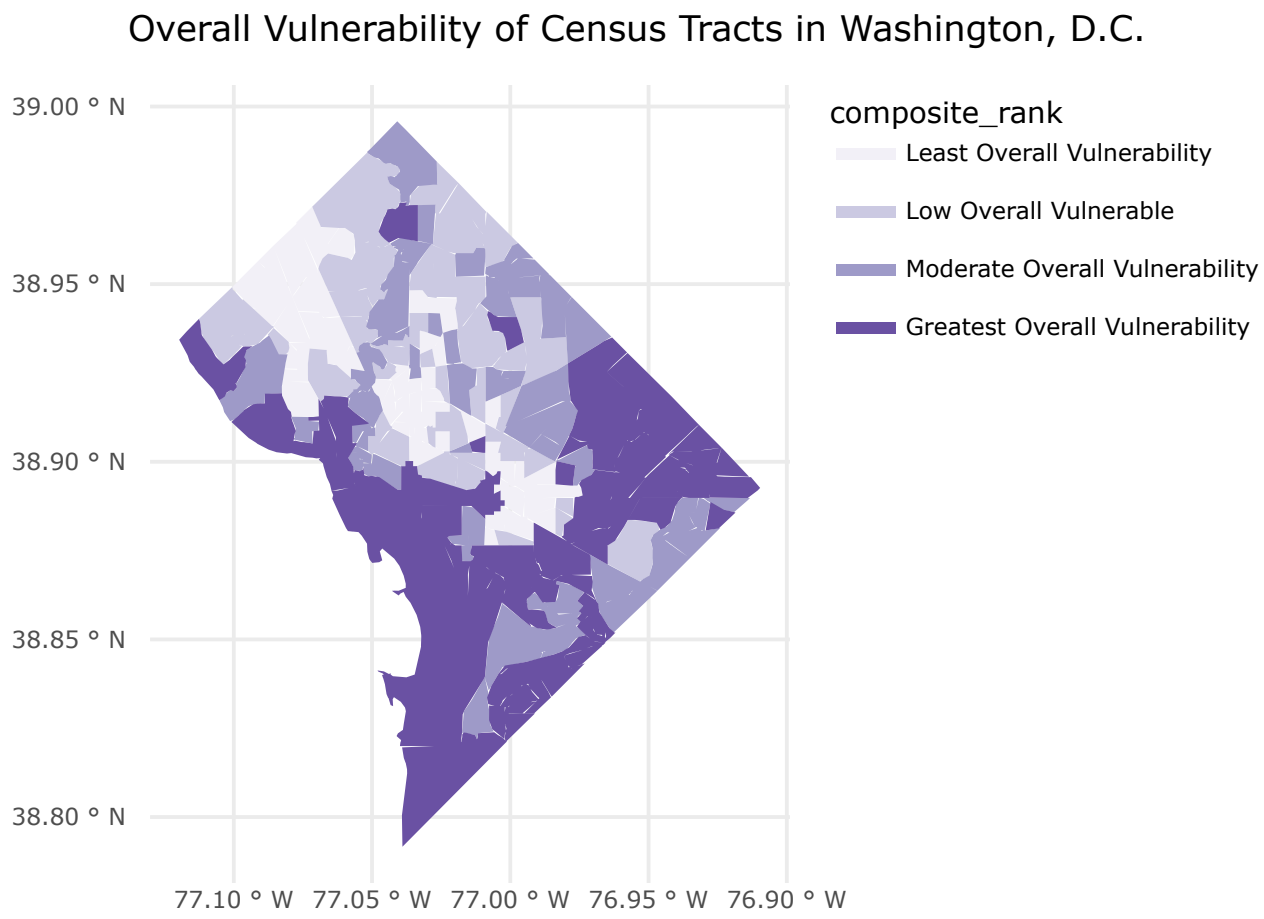
```
# Merge flood data with spatial data
composite_map <- dc_tracts %>%
  left_join(composite_data, by = "GEOID")
```

Creates two plots, the first is a simple mapping of the composite vulnerability index for each census tract in Washington D.C. as a raw value. The second is mapping the categorical variable representing four percentiles of the composite index for each census tract. Note the tracts with highest and lowest overall vulnerability.

```
# Infrastructure plot

composite_map <- composite_map %>%
  mutate(tooltip_text = paste0("Tract: ", GEOID,
                               "<br>Index: ", sprintf("%.2f", composite_index)))

composite_plot <- ggplot(composite_map, aes(text = tooltip_text)) +
  geom_sf(aes(fill = composite_index), color = NA) +
  scale_fill_viridis_c(option = "plasma", name = "Composite Index", direction = -1) +
  labs(title = "Overall Vulnerability of Census Tracts in Washington, D.C.",
       subtitle = "Census Tracts (2020)",
       caption = "Source: Open Data DC") +
  theme_minimal()

ggplotly(composite_plot, tooltip = "text")
```



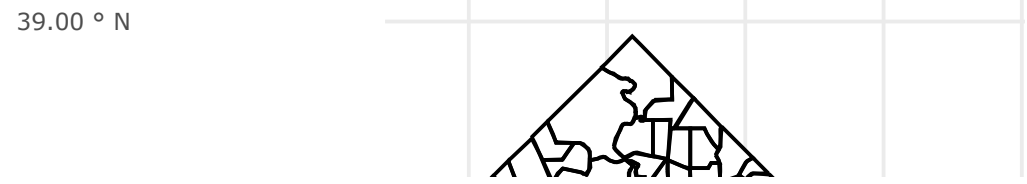
```
composite_plot <- ggplot(composite_map, aes(text = tooltip_text)) +
  geom_sf(aes(fill = composite_rank), color = NA) +
  scale_fill_brewer(palette = "Purples") +
  labs(title = "Overall Vulnerability of Census Tracts in Washington, D.C.",
       subtitle = "Census Tracts (2020)",
       caption = "Source: Open Data DC") +
  theme_minimal()

# Step 3: Convert to interactive
ggplotly(composite_plot, tooltip = "text")
```




```
# Step 0: Confirm column names
names(composite_data)  # Make sure 'GEOID' and 'composite_index' exist here
```

```
##  [1] "GEOID"                           "floodplain_500"
##  [3] "floodplain_100"                  "tidal_floodplain"
##  [5] "blue_zone"                       "sso"
##  [7] "ground_elevation"                "base_elevation"
##  [9] "one_hundred_year_floodplain_risk" "dist_to_water"
## [11] "asthma"                          "diabetes"
## [13] "pct_poverty"                     "percent_vulnerable"
## [15] "pct_unemp"                       "num_fire_stations"
## [17] "num_hospitals"                   "num_police_stations"
## [19] "num_cross_guards"                "pct_minority"
## [21] "pct_old_housing"                 "pct_raster_407m"
## [23] "pct_buildings_407m"              "flood_index"
## [25] "social_health_index"             "infra_index"
## [27] "composite_index"                 "composite_rank"
## [29] "tooltip_text"
```

```
# Ensure GEOID is character in both datasets
composite_data <- composite_data %>%
  mutate(GEOID = as.character(GEOID))

dc_tracts <- dc_tracts %>%
  mutate(GEOID = as.character(GEOID))

# Step 1: Identify and rank the top 15 tracts by composite_index
top15_data <- composite_data %>%
  arrange(desc(composite_index)) %>%
  slice_head(n = 15) %>%
  mutate(priority_rank = row_number())

# Step 2: Merge full composite_data and top15 priority into spatial data
dc_tracts <- dc_tracts %>%
  left_join(composite_data, by = "GEOID") %>%
  left_join(top15_data %>% select(GEOID, priority_rank), by = "GEOID")

# Step 3: Create interactive tooltip
dc_tracts <- dc_tracts %>%
  mutate(
    tooltip_text = paste0(
      "GEOID: ", GEOID,
      "<br>Composite Index: ", ifelse(!is.na(composite_index), sprintf("%.2f", composite_index),
"NA"),
      ifelse(!is.na(priority_rank), paste0("<br>Priority Rank: ", priority_rank), "")
    )
  )

# Step 4: Create ggplot object with ranked fill
priority_plot <- ggplot(dc_tracts, aes(text = tooltip_text)) +
  geom_sf(aes(fill = priority_rank), color = "black", show.legend = FALSE) +
  scale_fill_gradient(
    low = "#fee0d2", high = "#de2d26", na.value = "white"
  ) +
  labs(
    title = "Top 15 Vulnerable Census Tracts in Washington, D.C.",
    subtitle = "Shaded by Priority Rank (1 = Highest Vulnerability)",
    caption = "Source: Open Data DC"
  ) +
  theme_minimal()

# Step 5: Interactive plot
ggplotly(priority_plot, tooltip = "text")
```



Binning Census Tracts by Index Percentiles: Most Vulnerable Census Tracts