

Visualizing FEMA NRI x ACS Data

Aakriti Poudel

2026-02-05

Table of contents

Visualizing FEMA NRI x ACS Data	1
Set up	1
Import and read data	1
Data cleaning and wrangling	2
Data visualization	4
Answer some questions	7

Visualizing FEMA NRI x ACS Data

Create a data viz that helps to answer the question, How does climate hazard risk exposure vary across racial and ethnic groups in California?

Set up

```
# Load all necessary libraries
library(tidyverse)
library(here)
library(janitor)
library(ggplot2)
```

Import and read data

```

# #....Step 1a: see all available ACS variables + descriptions.....
# acs_vars <- tidyacensus::load_variables(year = 2023,
#                                         dataset = "acs1")
#
# #.....Step 1b: import race & ethnicity data.....
# race_ethnicity <- tidyacensus::get_acs(
#   geography = "county",
#   survey = "acs1",
#   # NOTE: you may not end up using all these variables
#   variables = c("B01003_001", "B02001_002", "B02001_003",
#               "B02001_004", "B02001_005", "B02001_006",
#               "B02001_007", "B02001_008", "B03002_012",
#               "B03002_002"),
#   state = "CA",
#   year = 2023) |>
#   # join variable descriptions (so we know what's what!)
#   dplyr::left_join(acs_vars, by = dplyr::join_by(variable == name))
#
# #.....Step 2: write ACS data to file.....
# readr::write_csv(race_ethnicity, here::here("data", "ACS-1yr-2023-county-race-ethnicity.csv"))

#.....Step 3: read in your CSV file.....
race_ethnicity <- readr::read_csv(here::here("data", "ACS-1yr-2023-county-race-ethnicity.csv"),
  clean_names())

# Read data set
nri <- read.csv(here("data", "National_Risk_Index_Counties_807384124455672111.csv")) %>%
  clean_names()

```

Data cleaning and wrangling

```

# Filter `nri` data for California and select necessary columns only
california_nri <- nri %>%
  filter(state_name == "California") %>%
  select(state_county_fips_code,
         county_name,
         national_risk_index_score_composite,
         population_2020)

# Wrangle ACS data `race_ethnicity` to get race/ethnicity

```

```

race_ethnicity_clean <- race_ethnicity %>%
  # Extract just the race/ethnicity categories we need
  filter(variable %in% c("B01003_001", # Total population
                        "B02001_002", # White alone
                        "B02001_003", # Black or African American alone
                        "B02001_004", # American Indian and Alaska Native alone
                        "B02001_005", # Asian alone
                        "B02001_006", # Native Hawaiian and Other Pacific Islander alone
                        "B02001_007", # Some Other Race alone
                        "B02001_008", # Two or More Races
                        "B03002_012")) %>% # Hispanic or Latino
  # Clean `label` column to extract just the race/ethnicity name
  mutate(race_ethnicity = case_when(
    variable == "B01003_001" ~ "Total Population",
    variable == "B02001_002" ~ "White",
    variable == "B02001_003" ~ "Black or African American",
    variable == "B02001_004" ~ "American Indian and Alaska Native",
    variable == "B02001_005" ~ "Asian",
    variable == "B02001_006" ~ "Native Hawaiian and Other Pacific Islander",
    variable == "B02001_007" ~ "Some Other Race",
    variable == "B02001_008" ~ "Two or More Races",
    variable == "B03002_012" ~ "Hispanic or Latino")) %>%
  select(geoid, name, race_ethnicity, estimate)

# Pivot wider to calculate percentages
race_ethnicity_wide <- race_ethnicity_clean %>%
  pivot_wider(names_from = race_ethnicity,
              values_from = estimate) %>%
  # Rename the race/ethnicity to calculate percentages
  rename(total_population = `Total Population`,
         white = White,
         black_or_african_american = `Black or African American`,
         american_indian_and_alaska_native = `American Indian and Alaska Native`,
         asian = Asian,
         native_hawaiian_and_other_pacific_islander = `Native Hawaiian and Other Pacific Islander`,
         some_other_race = `Some Other Race`,
         two_or_more_races = `Two or More Races`,
         hispanic_or_latino = `Hispanic or Latino`) %>%
  # Calculate percentages of race/ethnicity
  mutate(white_pct = white / total_population * 100,
         black_or_african_american_pct = black_or_african_american / total_population * 100,
         american_indian_and_alaska_native_pct = american_indian_and_alaska_native / total_p

```

```

asian_pct = asian / total_population * 100,
native_hawaiian_and_other_pacific_islander_pct = native_hawaiian_and_other_pacific_islander / total_population * 100,
some_other_race_pct = some_other_race / total_population * 100,
two_or_more_races_pct = two_or_more_races / total_population * 100,
hispanic_or_latino_pct = hispanic_or_latino / total_population * 100)

# Pivot back to long format for combining both data sets
race_ethnicity_long <- race_ethnicity_wide %>%
  select(geoid, name, ends_with("_pct")) %>%
  pivot_longer(cols = ends_with("_pct"),
               names_to = "race_ethnicity",
               values_to = "percentage") %>%
  mutate(race_ethnicity = str_remove(race_ethnicity, "_pct"))

# Combine NRI and ACS data
nri_acs <- california_nri %>%
  # Use str_pad() to ensure the geoid always has 5 digits with leading zeros
  mutate(geoid = str_pad(as.character(state_county_fips_code),
                        width = 5,
                        side = "left",
                        pad = "0")) %>%
  left_join(race_ethnicity_long, by = "geoid")

# Calculate weighted mean risk score by race/ethnicity
risk_by_race <- nri_acs %>%
  group_by(race_ethnicity) %>%
  summarize(weighted_avg_risk = weighted.mean(national_risk_index_score_composite,
                                                w = percentage,
                                                na.rm = TRUE),
            .by_group = TRUE),
  # Calculate the sum of percentages across counties
  pct_sum = sum(percentage, na.rm = TRUE),
  n_counties = n(),
  # Remove NA values
  filter(!is.na(race_ethnicity)) %>%
  # Arrange by risk score for better visualization
  arrange(desc(weighted_avg_risk))

```

Data visualization

```

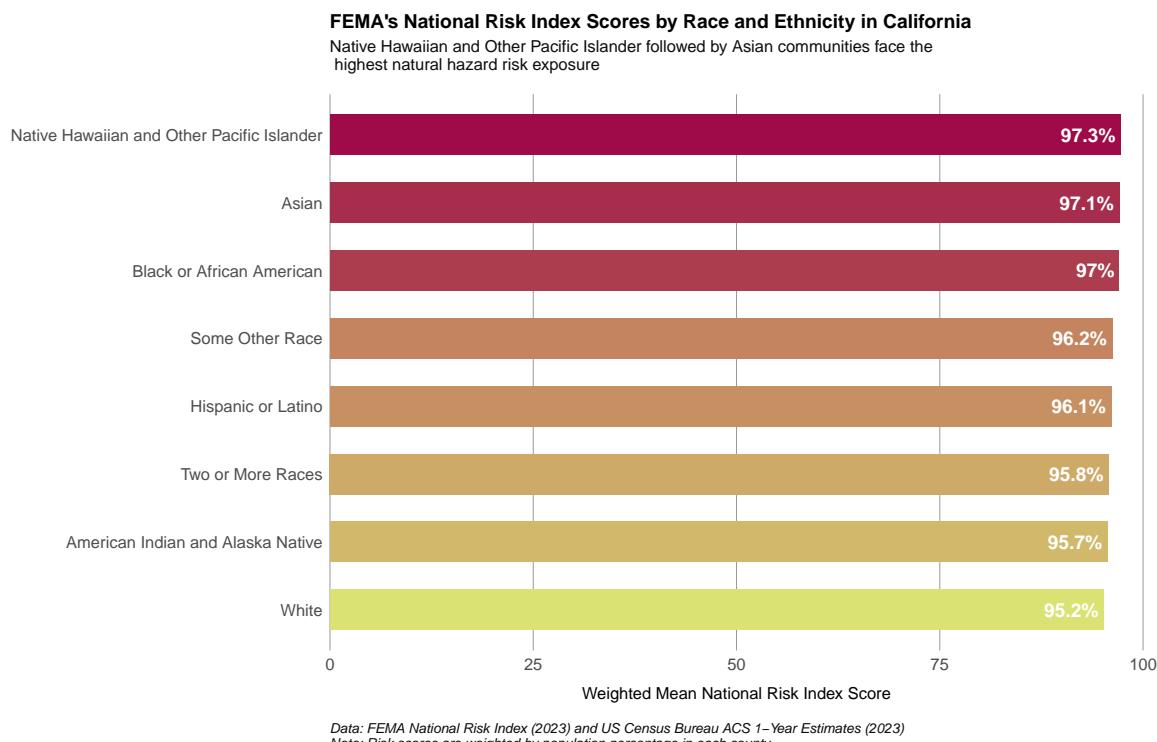
# Create the visualization
ggplot(risk_by_race,
       aes(x = reorder(race_ethnicity, weighted_avg_risk),
           y = weighted_avg_risk,
           fill = weighted_avg_risk)) +
  geom_col(width = 0.6) +
  # Add percentage labels inside the bars
  geom_text(aes(label = paste0(round(weighted_avg_risk, 1), "%")),
            hjust = 1.1,
            size = 5,
            color = "white",
            fontface = "bold") +
  coord_flip() +
  # Use custom color gradient from light green to dark blue
  scale_fill_gradient(high = "#9f0a48",
                      low = "#dae274",
                      guide = "none") +
  # Set y-axis limits and breaks
  scale_y_continuous(expand = c(0, 0),
                     limits = c(0, 100),
                     breaks = seq(0, 100, 25)) +
  # Rename race/ethnicity labels for better readability
  scale_x_discrete(labels = c(
    "white" = "White",
    "black_or_african_american" = "Black or African American",
    "american_indian_and_alaska_native" = "American Indian and Alaska Native",
    "asian" = "Asian",
    "native_hawaiian_and_other_pacific_islander" = "Native Hawaiian and Other Pacific Islander",
    "some_other_race" = "Some Other Race",
    "two_or_more_races" = "Two or More Races",
    "hispanic_or_latino" = "Hispanic or Latino")) +
  labs(
    title = "FEMA's National Risk Index Scores by Race and Ethnicity in California",
    subtitle = "Native Hawaiian and Other Pacific Islander followed by Asian communities face the highest risk",
    x = NULL,
    y = "Weighted Mean National Risk Index Score",
    caption = "Data: FEMA National Risk Index (2023) and US Census Bureau ACS 1-Year Estimates")
  theme_minimal(base_size = 12) +
  theme(plot.title = element_text(face = "bold",
                                 size = 14,
                                 margin = margin(b = 6)),
        plot.subtitle = element_text(face = "bold",
                                 size = 12,
                                 margin = margin(b = 6)))

```

```

plot.subtitle = element_text(size = 12,
                             color = "grey5",
                             margin = margin(b = 15)),
plot.caption = element_text(size = 10,
                           color = "grey5",
                           hjust = 0,
                           face = "italic",
                           margin = margin(t = 15)),
axis.text.y = element_text(size = 12),
axis.text.x = element_text(size = 12),
axis.title.x = element_text(size = 12,
                             margin = margin(t = 10)),
panel.grid.major.y = element_blank(),
panel.grid.minor = element_blank(),
panel.grid.major.x = element_line(color = "grey60",
                                   linewidth = 0.3),
plot.margin = margin(20, 20, 20, 20))

```



Answer some questions

- 1. What are your variables of interest and what kinds of data (e.g. numeric, categorical, ordered, etc.) are they (a bullet point list is fine)?**

Answer: My main variables are race and ethnicity groups and natural hazard risk scores.

Race and ethnicity is a categorical variable with 8 groups such as White, Black or African American, American Indian and Alaska Native, Asian, Native Hawaiian and Other Pacific Islander, Some Other Race, Two or More Races, and Hispanic or Latino. The NRI score is a numeric variable measuring natural hazard risk for each California county, ranging from 0 to over 100 (higher = more risk). The population percentage is a numeric variable showing what percent of each county belongs to each racial/ethnic group. I use this to calculate a weighted mean risk score, and it gives me the risk exposure for each group across California. I also use county codes (GEOID) to join the data sets together.

- 2. How did you decide which type of graphic form was best suited for answering the question? What alternative graphic forms could you have used instead? Why did you settle on this particular graphic form?**

Answer: I chose a horizontal bar chart because it's the simplest way to compare a numeric value (risk score) across different categories (racial and ethnic groups).

I considered vertical bar chart, but the long group names would be hard to read.

I settled on horizontal bars for three main reasons. First, the long category names fit better when they're horizontal. Second, I could arrange the bars from highest to lowest risk score. Third, I could add the exact percentages inside each bar and use color to show which groups face the most risk. This makes both the numbers and the overall trend easy to understand.

- 3. Summarize your main finding in no more than two sentences.**

Answer: Native Hawaiian and Other Pacific Islander populations face the highest climate hazard risk exposure in California with a weighted average score of 97.3%, followed closely by Asian populations at 97.1%. White populations face the lowest risk exposure at 95.2%, revealing disparities in climate risk across racial and ethnic groups. These differences reflect where different racial and ethnic groups live in California. The communities of color, particularly Native Hawaiian and Other Pacific Islander and Asian populations, are more concentrated in counties with higher exposure to climate hazards like coastal flooding, wildfires and earthquakes. This raises important environmental justice concerns about who bears the greatest burden of natural hazards.

- 4. What modifications did you make to this visualization to make it more easily readable?**

Answer: I made several modifications to improve readability. First, I arranged the bars from highest to lowest instead of alphabetical order, making the pattern immediately clear. Second,

I used a color gradient from dark magenta to light olive, with darker colors highlighting the higher risk groups. Third, I added percentage labels in white text inside each bar. Fourth, I made the chart horizontal instead of vertical so the long racial and ethnic group names are easy to read. Finally, I removed unnecessary grid lines. These changes help to understand both the trend and specific numbers easily.

5. Is there anything you wanted to implement, but didn't know how? If so, please describe.

Answer: I was able to implement everything I wanted for this visualization. The main challenge was figuring out how to calculate the weighted mean properly to account for population differences across counties, but once I understood the concept, the code worked as intended.