Texas Blackout Analysis

Lucian Scher

2025-11-11

GitHub repository: https://github.com/lucianbluescher/Texas-impacts-of-extreme-weather.git

Texas Blackouts Data Analysis

Image Credits: Anthony Cappelletti, Texas Winter Storm 2021: Accounting for Subsequent Events

Environmental justice geospatial data analysis in R of Texas's blackouts (Feburary 2021)

Author: Lucian Scher

The purpose of this project is to identify the environmental justice impacts of the series of storms in Febuary of 2021 that devastated the Houston Texas power grid. The project acomplishes this by estimating the number of homes in the Houston metropolitan area that lost power and investigating whether impacts were disproportionately felt by low-income communities and non-white areas. It also compares night light intensities before and after the first two storms and mapping the homes that lost power.

The entire project can be found and ran in the texas-blackout.qmd. The repository also includes this README and project files. Filepaths can be found in the tree below.

Data

All data can be found in the /data folder and is read in the second code chunk of the qmd file. This project uses four separate data files accessed 11/1/2025 through a data folder provided by course instruction.

- Night Lights VIIRS (Visible Infrared Imaging Radiometer Suite) data from NASA's LAADS DAAC captures nighttime light emissions. We used cloud-free imagery from February 7, 2021 (before the storm) and February 16, 2021 (during the blackout) covering Houston-area tiles h08v05 and h08v06 in sinusoidal projection.
- Roads Highway data from OpenStreetMap via Geofabrik, filtered to the Houston metropolitan area. Used to exclude areas where reduced nighttime lights may indicate lower traffic rather than power outages.
- Buildings Residential building footprints from OpenStreetMap via Geofabrik, filtered to houses in the Houston metropolitan area. Source:
- Socioeconomic Data Census tract-level data from the U.S. Census Bureau's American Community Survey (2019 5-year estimates) for Texas. Stored as an ArcGIS File Geodatabase with geometry and attributes in separate layers.

```
EDS223-HW3
└──README.md
└──Rmd/Proj files
└── texas_blackout.qmd
└──.gitignore
   └──data
      └──gis_osm_buildings_a_free_1.gpkg     # Buildings
      └──gis_osm_roads_free_1.gpk            # Roads
      └──ACS_2019_5YR_TRACT_48_TEXAS.gdb     # Socioeconomic
         └──census tract gdb files
      └──VNP46A1                             #Night Lights
         └──VIIRS data files
```

Acknowledgments

This project was a homework assignment for the fall 2025 course, Geospatial Analysis & Remote Sensing (EDS 223). A part of the Master of Environmental Data Science (MEDS) program at UCSB's Bren school of Environmental Science & Management. The assigment was created by Ruth Oliver and administered by Annie Adams. I would like to thank them both for this insightful and enjoyable project, and Annie for providing feedback on my analysis.

1. Load required packages

```
library(stars)  # For working with raster data
library(sf)  # For loading data with SQL query subset
library(tmap)  # For maps
library(ggplot2)  # For maps
library(dplyr)
library(ggspatial) # compass and scalebar to ggplot
library(viridis) # Cool colors
```

2. Read in all necessary data

Night Lights:

NASA VIIRS Houston night time light coverage data from sinusoidal equal-area projection tiles h08v05 and h08v06 on 2021-02-07 and 2021-02-16 Key: lights_TILE_DAY

```
# tile h08v05, collected on 2021-02-07 (VNP1038)
lights_05_07 <- read_stars('data/VNP46A1/VNP46A1.A2021038.h08v05.001.2021039064328.tif')

# tile h08v06, collected on 2021-02-07 (VNP1038)
lights_06_07 <- read_stars('data/VNP46A1/VNP46A1.A2021038.h08v06.001.2021039064329.tif')

# tile h08v05, collected on 2021-02-16 (VNP1038)
lights_05_16 <- read_stars('data/VNP46A1/VNP46A1.A2021047.h08v05.001.2021048091106.tif')

# tile h08v06, collected on 2021-02-16 (VNP1038)
lights_06_16 <- read_stars('data/VNP46A1/VNP46A1.A2021047.h08v06.001.2021048091105.tif')
```

Roads:

Road data from Open Street Map redistributed by Geofabriks and prepared for Houston Specific analysis by teaching team

```
# Read road data using SQL query specifying highways only
roads_sf <- st_read( "data/gis_osm_roads_free_1.gpkg",
  query = "SELECT *
  FROM gis_osm_roads_free_1
```

```
  WHERE fclass='motorway'", quiet = TRUE
)
```

Houses:

House data also from Open Street Map redistributed by Geofabriks and prepared for Houston Specific analysis by teaching team

```
# Read Houston home data using SQL query specifying only buildings with residents
homes_sf <- st_read("data/gis_osm_buildings_a_free_1.gpkg",
  query = "SELECT *
  FROM gis_osm_buildings_a_free_1
  WHERE (type IS NULL AND name IS NULL)
  OR type in ('residential', 'apartments', 'house', 'static_caravan', 'detached')", quiet = T
)
```

Socioeconomic:

Socioeconomic data from U.S. Census Bureau's American Community Surveys 2019 census tracts

```
#Read in income layer
socio_sf_inc <- st_read('data/ACS_2019_5YR_TRACT_48_TEXAS.gdb',
                    layer = 'X19_INCOME', quiet = TRUE)

# Read in race layer
socio_sf_race <- st_read('data/ACS_2019_5YR_TRACT_48_TEXAS.gdb',
                         layer = 'X02_RACE', quiet = TRUE)
# Read in geometry layer
socio_sf_geo <- st_read('data/ACS_2019_5YR_TRACT_48_TEXAS.gdb',
                    layer = 'ACS_2019_5YR_TRACT_48_TEXAS', quiet = TRUE)
```

```
# Check available layers
#st_layers(dsn = 'data/ACS_2019_5YR_TRACT_48_TEXAS.gdb')
```

3. Workflow

Create Blackout Mask

```
# Combine night light data by day using st_mosaic

feb_7_comb <- st_mosaic(lights_05_07, lights_06_07)
feb_16_comb <- st_mosaic(lights_05_16, lights_06_16)

rm(lights_05_07, lights_05_16, lights_06_07, lights_06_16)
# Subtract into a new object to show differences
comb_change <- feb_16_comb - feb_7_comb

# Reclassify data when change is less than a -200 drop (greater than 200 drop) set to 1, if l
blackout_vals <- ifelse(comb_change[[1]] < -200, 1, NA)

# Wrap back into stars data
blackout_sf <- st_as_stars(blackout_vals, dimensions = st_dimensions(comb_change))

# Convert raster to polygons (sf)
blackout_sf <- st_as_sf(blackout_sf, as_points = FALSE, merge = TRUE, na.rm = TRUE)

# Fix invalid geometries
blackout_sf <- st_make_valid(blackout_sf)

# Set bounding box of Houston Area from given coordinates
houston_bbox <- st_bbox(c(xmin = -96.5, xmax = -94.5, ymin = 29, ymax = 30.5), crs = 4326)

# Crop SF by set bbox
blackout_sf <- st_crop(blackout_sf, houston_bbox)

# Transform CRS of sf to EPSG:3083 (NAD83 / Texas Centric Albers Equal Area)
blackout_sf <- st_transform(x = blackout_sf, crs = 3083)
```

Exclude Highways from blackout mask

```
# Set CRS
if (st_crs(roads_sf)$epsg != 3083) {
  warning("CRS is not EPSG:3083 - transforming to EPSG:3083.")
```

```
  roads_sf <- st_transform(roads_sf, crs = 3083)
} else {
  message("CRS is already EPSG:3083.")
}

# Crop roads by set bbox
#roads_sf <- st_crop(roads_sf, houston_bbox)

# Make valid for speed
#roads_sf <- st_make_valid(roads_sf)

# Use union to combine geometries into one polygon
roads_union <- st_union(roads_sf)

# Create buffer of 200m around roads
road_buffer <- st_buffer(roads_union, dist = 200)

# Use union again for simplification
road_buffer <- st_union(road_buffer)

# Remove points from blackout_sf that are within the 200m buffer around roads
blackout_dif <- st_difference(x = blackout_sf, y = road_buffer)
```

4. Outputs

Night light intensities before and after the first two storms

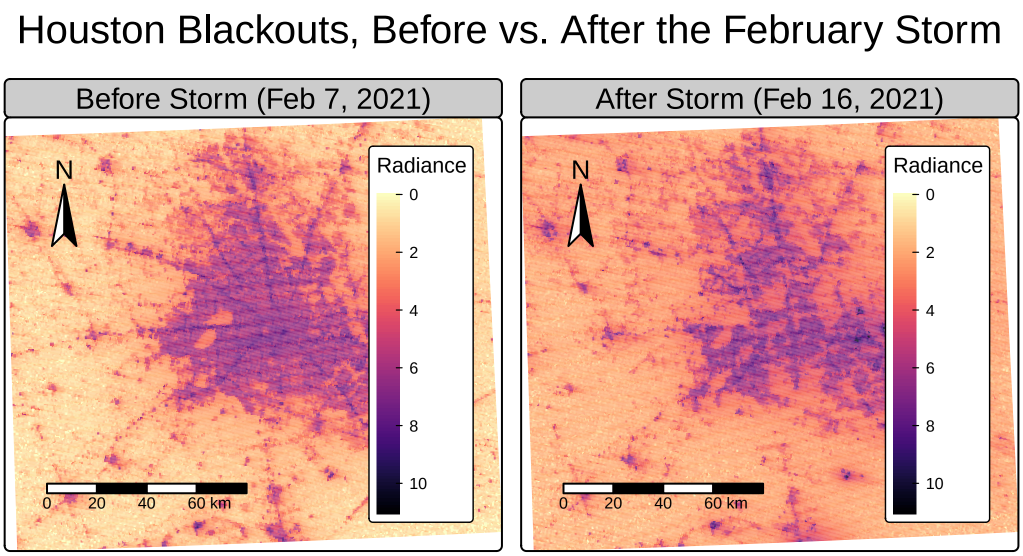
```
# st_crs(feb_7_comb)
# st_crs(feb_16_comb)

# Transform the mosaics to geographic CRS
feb_7_geo <- st_warp(feb_7_comb, crs = 4326)  # WGS84
feb_16_geo <- st_warp(feb_16_comb, crs = 4326)

# Set to Houston area bounding box
feb_7_houston <- st_crop(feb_7_geo, houston_bbox)
feb_16_houston <- st_crop(feb_16_geo, houston_bbox)

# Transform to Texas Centric Albers Equal Area for analysis
```

```r
feb_7_houston <- st_transform(feb_7_houston, crs = 3083)
feb_16_houston <- st_transform(feb_16_houston, crs = 3083)

# Calculate difference
light_change <- feb_16_houston - feb_7_houston
```

Create map showing before and after storm

```r
# Create function to apply log to radiance in order to dramatize color difference
log_raster <- function(st) {
  out <- st_apply(st, c("x", "y"), log1p)
  names(out) <- "log_radiance"
  out
}

# Apply function
feb_7_log  <- log_raster(feb_7_houston)
feb_16_log <- log_raster(feb_16_houston)

# Create stacked raster
stacked <- c(feb_7_log, feb_16_log, along = "time")
stacked <- st_set_dimensions(stacked, "time",
                             values = c("Before Storm (Feb 7, 2021)",
                                        "After Storm (Feb 16, 2021)"))
# Map
tm_shape(stacked) +
  tm_raster(
    col        = "log_radiance",
    col.scale  = tm_scale_continuous(values = viridis::magma(120, direction = -1)),
    col.legend = tm_legend(title = "Radiance")
  ) +
  tm_facets(by = "time", ncol = 2) +
  tm_title("Houston Blackouts, Before vs. After the February Storm") +
  tm_layout(
    legend.position = c("right", "center"),
    legend.outside = TRUE
  ) +
  tm_scalebar(position = c("left", "bottom")) +
  tm_compass(type = "arrow",
             position = c("top", "left"))
```

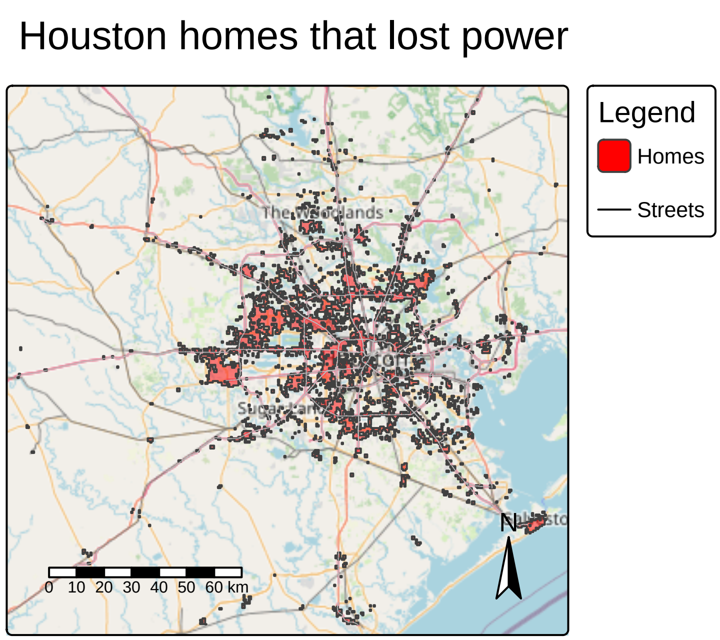
```
[plot mode] fit legend/component: Some legend items or map compoments do not
fit well, and are therefore rescaled.
i Set the tmap option `component.autoscale = FALSE` to disable rescaling.
```



Map of the homes in Houston that lost power

```
# Map homes in Houston that lost power
tm_shape(blackout_sf) + # Add blacked out homes
  tm_polygons(
    fill = "red",
    fill_alpha = 0.5
  ) +
tm_shape(roads_sf) + # Add roads
  tm_lines(
    col = "grey",
    lwd = 0.5,
    col_alpha = 0.5
  ) +
tm_basemap("OpenStreetMap") + # Add base map
tm_add_legend(
  type = "polygons",
```

```
  labels = "Homes",
  fill = "red",
  title = "Legend"
) +
tm_add_legend(
  type = "lines",
  labels = "Streets",
  lwd = 1
) +
tm_title("Houston homes that lost power") +
tm_layout(legend.outside = TRUE) +
tm_compass(position = c("right", "bottom")) +
tm_scalebar(position = c("left", "bottom"))
```



FInd total number of homes in Houston that lost power

```
# Check CRS of homes
if (st_crs(homes_sf)$epsg != 3083) {
  warning("CRS is not EPSG:3083 - transforming to EPSG:3083.")
  homes_sf <- st_transform(homes_sf, crs = 3083)
```

```
} else {
  message("CRS is already EPSG:3083.")
}
```

```
Warning: CRS is not EPSG:3083 - transforming to EPSG:3083.
```

```
# Total number of homes
total_homes <- nrow(homes_sf)

# Number of homes in blackout area (spatial join)
homes_in_blackout <- st_intersection(homes_sf, blackout_sf)
num_blackout <- nrow(homes_in_blackout)

# Number of homes NOT in blackout area
num_no_blackout <- total_homes - num_blackout

# Percentage affected
pct_blackout <- num_blackout / total_homes * 100
pct_no_blackout <- 100 - pct_blackout

# Print summary
cat("Total homes:", total_homes, "\n")
```

```
Total homes: 475941
```

```
cat("Homes in blackout areas:", num_blackout, sprintf("(%.1f%%)", pct_blackout), "\n")
```

```
Homes in blackout areas: 164874 (34.6%)
```

```
cat("Homes not in blackout areas:", num_no_blackout, sprintf("(%.1f%%)", pct_no_blackout), "
```

```
Homes not in blackout areas: 311067 (65.4%)
```

Map of the census tracts in Houston that lost power

Preparing census tract data

```r
# Join income and race data with geometry
houston_tracts_inc <- socio_sf_geo |>
  left_join(socio_sf_inc, by = c("GEOID_Data" = "GEOID"))

houston_tracts_race <- socio_sf_geo |>
  left_join(socio_sf_race, by = c("GEOID_Data" = "GEOID"))

# Transform to EPSG:3083
houston_tracts_inc <- st_transform(houston_tracts_inc, crs = 3083)
houston_tracts_race <- st_transform(houston_tracts_race, crs = 3083)

# Crop to Houston area
houston_bbox <- st_bbox(blackout_sf)
houston_tracts_inc <- st_crop(houston_tracts_inc, houston_bbox)
houston_tracts_race <- st_crop(houston_tracts_race, houston_bbox)

# Identify which census tracts experienced blackouts
tracts_with_blackout <- st_intersects(houston_tracts_inc, blackout_sf, sparse = FALSE)
houston_tracts_inc$blackout <- rowSums(tracts_with_blackout) > 0


# Define low income (e.g., below median)
median_income_threshold <- median(houston_tracts_inc$B19013e1, na.rm = TRUE)
houston_tracts_inc$low_income <- houston_tracts_inc$B19013e1 < median_income_threshold

# First make sure low_income is a factor with proper labels
houston_tracts_inc <- houston_tracts_inc %>%
  mutate(low_income = factor(low_income,
                             levels = c(FALSE, TRUE),
                             labels = c("Higher Income","Lower Income")))
```

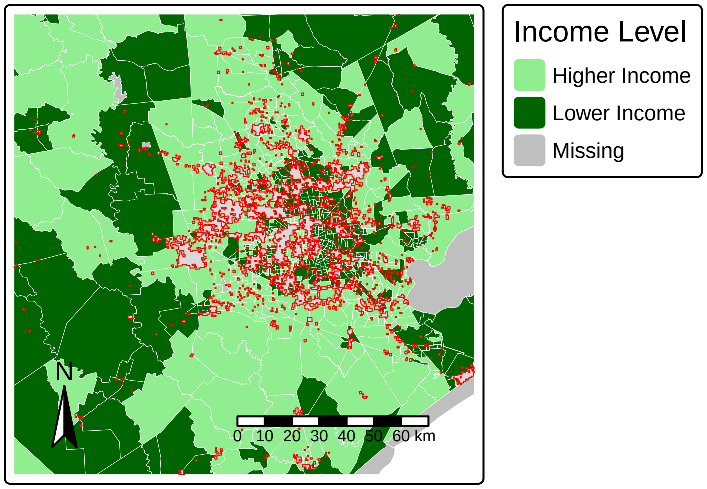
Mapping blackout status on top of income status

```r
# Create map showing both income and blackout status

tm_shape(houston_tracts_inc) + # Add census tracts
  tm_polygons(
    fill = "low_income",
    fill.scale = tm_scale_categorical(values = c("lightgreen", "darkgreen")),
    fill.legend = tm_legend(title = "Income Level"),
```

```r
    col = "white",
    lwd = 0.1
  ) +

tm_shape(blackout_sf) + # Add blackout homes
  tm_polygons(
    fill = NA,
    col = "red",
    lwd = 0.5
  ) +
tm_title("Low Income Census Tracts and Blackout Areas",
         bg.color = "white",
         position = tm_pos_out("center", "top")) +
tm_layout(
  legend.outside = TRUE,
  legend.outside.position = "right",
  outer.margins = 0.02
) +
tm_credits(
  "Red outline shows areas that lost power in February 2021",
  bg.color = "white",
  position = tm_pos_out("center", "bottom"),
  size = 0.7
) +
tm_compass(
  position = tm_pos_in("left", "bottom"),
  size = 2
) +
tm_scalebar(
  position = tm_pos_in("right", "bottom")
)
```

```
[plot mode] fit legend/component: Some legend items or map compoments do not
fit well, and are therefore rescaled.
i Set the tmap option `component.autoscale = FALSE` to disable rescaling.
```

Low Income Census Tracts and Blackout Areas



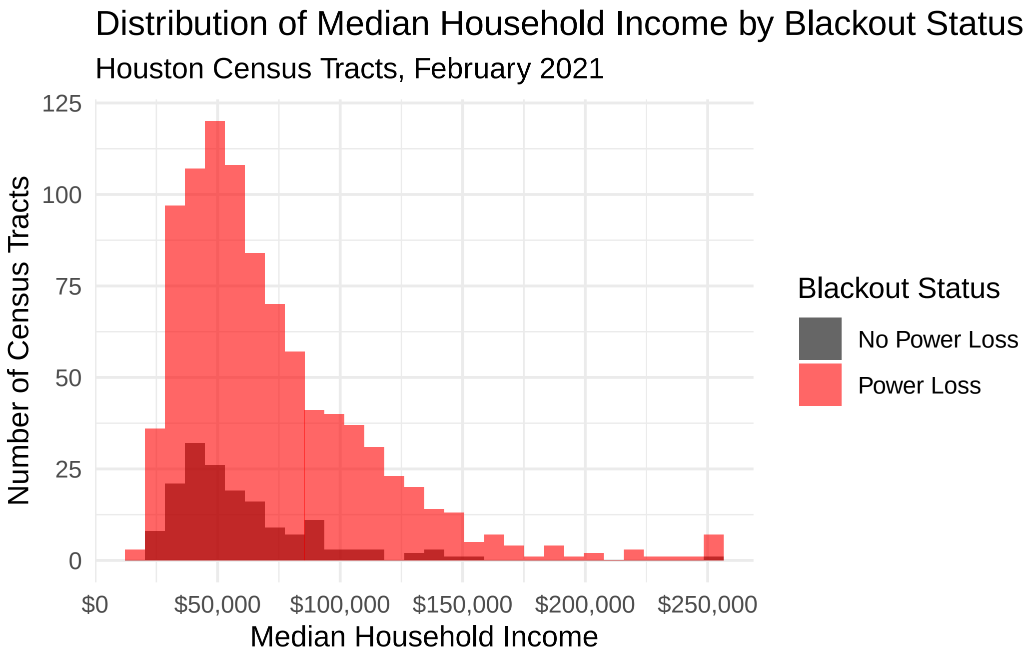
Red outline shows areas that lost power in February 2021

Plot comparison of the distributions of median household income for census tracts that did and did not experience blackouts

Median household income

```r
# Identify blackouts in low income census tracts
tracts_with_blackout <- st_intersects(houston_tracts_inc, blackout_sf, sparse = FALSE)
houston_tracts_inc$blackout <- rowSums(tracts_with_blackout) > 0

# Create the plot
income_data <- houston_tracts_inc |>
  st_drop_geometry() |>
  select(blackout, B19013e1) |>
  rename(median_income = B19013e1) |>
  filter(!is.na(blackout), !is.na(median_income), median_income > 0) |>
  mutate(blackout_status = factor(blackout,
                                  levels = c(FALSE, TRUE),
                                  labels = c("No Power Loss", "Power Loss")))

ggplot(income_data, aes(x = median_income, fill = blackout_status)) +
```

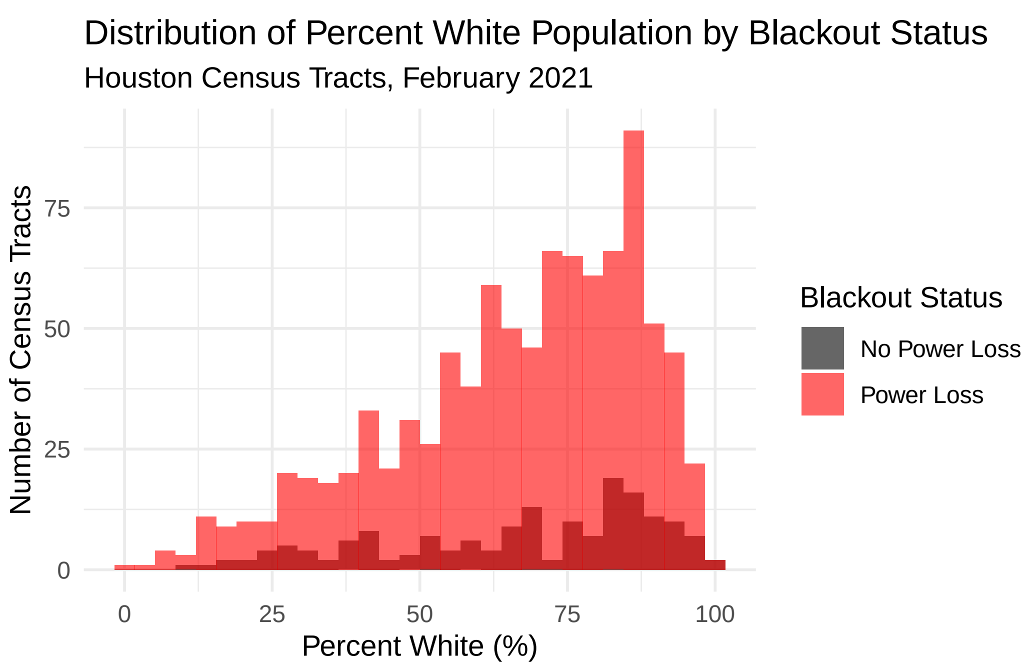
```r
  geom_histogram(alpha = 0.6, position = "identity", bins = 30) +
  scale_fill_manual(values = c("No Power Loss" = "black", "Power Loss" = "red"),
                    name = "Blackout Status") +
  labs(title = "Distribution of Median Household Income by Blackout Status",
       subtitle = "Houston Census Tracts, February 2021",
       x = "Median Household Income",
       y = "Number of Census Tracts") +
  theme_minimal() +
  scale_x_continuous(labels = scales::dollar_format())
```



Percent white

```r
# Identify blackouts in census tracts
tracts_with_blackout <- st_intersects(houston_tracts_race, blackout_sf, sparse = FALSE)
houston_tracts_race$blackout <- rowSums(tracts_with_blackout) > 0

# Calculate percent white for comparison
houston_tracts_race$pct_white <- (houston_tracts_race$B02001e2 / houston_tracts_race$B02001e

# Plot percent white by blackout status
```

```r
race_data <- houston_tracts_race |>
  st_drop_geometry() |>
  select(blackout, pct_white) |>
  filter(!is.na(blackout), !is.na(pct_white), pct_white >= 0) |>
  mutate(blackout_status = factor(blackout,
                                  levels = c(FALSE, TRUE),
                                  labels = c("No Power Loss", "Power Loss")))

ggplot(race_data, aes(x = pct_white, fill = blackout_status)) +
  geom_histogram(alpha = 0.6, position = "identity", bins = 30) +
  scale_fill_manual(values = c("Power Loss" = "red", "No Power Loss" = "black"),
                    name = "Blackout Status") +
  labs(title = "Distribution of Percent White Population by Blackout Status",
       subtitle = "Houston Census Tracts, February 2021",
       x = "Percent White (%)",
       y = "Number of Census Tracts") +
  theme_minimal()
```



Reflection

The results show inequities in the socioeconomic impacts of power loss. Poorer census tracts with a median household income of $40,000 to $80,000 were relatively more affected compared to rich ones. Predominantly white neighborhoods experienced higher rates of power loss, which could simply be a result of Houston's demographics. The limitations of this project could be that the 200-unit threshold for defining blackouts could have missed some variation, especially because I used st_difference instead of st_intersects which was taking to long. In addition, using the spatial overlap to identify tracts likely overestimated impacts. Finally, 2019 Census data may or may not reflect the actual demographics of 2021 when the storms took place.