```
# install.packages
library(dplyr)
library(tmap)
library(tigris)
library(ggplot2)
library(sf)

`%notin%` <- Negate(`%in%`)

options(tigris_use_cache = TRUE)


### Load and Process EAGLE-I Data

eaglei.2021 <- readr::read_csv("eaglei_outages/eaglei_outages_2021.csv") %>%
rename(datetime = run_start_time)

uri <- eaglei.2021  %>%
    filter(datetime < lubridate::dmy("19-2-2021") &
        datetime > lubridate::dmy("12-2-2021"))

rm(eaglei.2021)
gc()

#UCS edited the Brelsford et al. 2024 original "Uri_Map.R" code to correct errors
uri.slice <- uri %>%
 dplyr::group_by(fips_code) %>%
 dplyr::mutate(sum = dplyr::if_else(is.na(customers_out), 0, customers_out)) %>%
 dplyr::summarise(max_out = max(customers_out, na.rm = TRUE),
            cust_hours_out = sum(customers_out, na.rm = TRUE)*0.25)

uri.slice <- uri.slice %>% dplyr::filter(substr(fips_code,1,2) %notin% c("60", "66", "69", "78"))

### Load and Process Modeled County Customer Data
mcc <-readr::read_csv("data/MCC.csv") %>%
 dplyr::rename(GEOID = County_FIPS) %>%
 dplyr::group_by(GEOID) %>%
 dplyr::summarise(mcc = sum(Customers, na.rm = TRUE)) %>%
 mutate(GEOID = stringr::str_pad(as.character(GEOID), width = 5, pad = "0", side = "left"))

mcc <- mcc %>% filter(GEOID != "Grand Total") %>% mutate(has.mcc = 1)
```

```
### Load Geographic information data

usa.counties = tigris::counties(year = 2019, resolution = "500k", cb = TRUE) %>%
          dplyr::select(one_of("GEOID", "NAME")) %>%
          dplyr::filter(substr(GEOID,1,2) %notin% c("60", "66", "69", "78")) %>%
          tigris::shift_geometry(position = "below")

### Merge Data

counties <- dplyr::left_join(usa.counties, mcc)

counties <- dplyr::full_join(counties,uri.slice, by = c("GEOID" = "fips_code")) %>%
        dplyr::mutate(max_out = dplyr::if_else(is.na(max_out), 0, max_out),
                  cust_hours_out = dplyr::if_else(is.na(cust_hours_out), 0, cust_hours_out),
                  max_pct_out = dplyr::if_else(max_out/mcc> 1, 1, max_out/mcc),
                  max_pct_out = dplyr::if_else(is.na(max_pct_out), 0, max_pct_out))


### produce national scale map

#ensure spatial reference is set before exporting to .shp file
class(counties)
counties <- st_make_valid(counties)

#export .shp file for use in mapping software
st_write(
  counties, "C:/...counties.shp",
  driver = "ESRI Shapefile",
  delete_dsn = TRUE
)

#keeping the original code here, but we didn't use it:
#tmap::tmap_mode("plot")
#tmap::tm_shape(counties)+tmap::tm_fill("max_pct_out", style = "pretty", n = 10, palette =
"seq", title = "Max #Percentage of Customers Out") +
  tm_layout(aes.palette = list(seq = "-PiYG"))
```