library(lehmansociology)

##   
## Attaching package: 'lehmansociology'

## The following object is masked from 'package:stats':  
##   
## frequency

library(sf)

## Linking to GEOS 3.4.2, GDAL 2.4.2, PROJ 4.8.0

library(ggplot2)  
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(readr)  
library(nycgeo)

acs\_data2 <- read\_csv("acs\_data2.csv", skip = 1)

##   
## ── Column specification ────────────────────────────────────────────────────────  
## cols(  
## .default = col\_double(),  
## id = col\_character(),  
## `Geographic Area Name` = col\_character(),  
## `Percent Margin of Error!!EMPLOYMENT STATUS!!Population 16 years and over` = col\_character(),  
## `Percent Margin of Error!!EMPLOYMENT STATUS!!Civilian labor force` = col\_character(),  
## `Estimate!!EMPLOYMENT STATUS!!Civilian labor force!!Unemployment Rate` = col\_character(),  
## `Margin of Error!!EMPLOYMENT STATUS!!Civilian labor force!!Unemployment Rate` = col\_character(),  
## `Percent Margin of Error!!EMPLOYMENT STATUS!!Females 16 years and over` = col\_character(),  
## `Percent Margin of Error!!EMPLOYMENT STATUS!!Own children of the householder under 6 years` = col\_character(),  
## `Percent Margin of Error!!EMPLOYMENT STATUS!!Own children of the householder 6 to 17 years` = col\_character(),  
## `Percent Margin of Error!!COMMUTING TO WORK!!Workers 16 years and over` = col\_character(),  
## `Percent!!COMMUTING TO WORK!!Workers 16 years and over!!Mean travel time to work (minutes)` = col\_character(),  
## `Percent Margin of Error!!COMMUTING TO WORK!!Workers 16 years and over!!Mean travel time to work (minutes)` = col\_character(),  
## `Percent Margin of Error!!OCCUPATION!!Civilian employed population 16 years and over` = col\_character(),  
## `Percent Margin of Error!!INDUSTRY!!Civilian employed population 16 years and over` = col\_character(),  
## `Percent Margin of Error!!CLASS OF WORKER!!Civilian employed population 16 years and over` = col\_character(),  
## `Percent Margin of Error!!INCOME AND BENEFITS (IN 2019 INFLATION-ADJUSTED DOLLARS)!!Total households` = col\_character(),  
## `Percent!!INCOME AND BENEFITS (IN 2019 INFLATION-ADJUSTED DOLLARS)!!Total households!!Median household income (dollars)` = col\_character(),  
## `Percent Margin of Error!!INCOME AND BENEFITS (IN 2019 INFLATION-ADJUSTED DOLLARS)!!Total households!!Median household income (dollars)` = col\_character(),  
## `Percent!!INCOME AND BENEFITS (IN 2019 INFLATION-ADJUSTED DOLLARS)!!Total households!!Mean household income (dollars)` = col\_character(),  
## `Percent Margin of Error!!INCOME AND BENEFITS (IN 2019 INFLATION-ADJUSTED DOLLARS)!!Total households!!Mean household income (dollars)` = col\_character()  
## # ... with 73 more columns  
## )  
## ℹ Use `spec()` for the full column specifications.

#View(acs\_data2)

nyc\_boundaries(geography = "borough")

## Simple feature collection with 5 features and 6 fields  
## geometry type: MULTIPOLYGON  
## dimension: XY  
## bbox: xmin: 913213.6 ymin: 120131.4 xmax: 1067379 ymax: 272798.5  
## CRS: EPSG:2263  
## # A tibble: 5 x 7  
## geoid state\_fips county\_fips county\_name borough\_name borough\_id  
## <chr> <chr> <chr> <chr> <chr> <chr>   
## 1 36061 36 061 New York Manhattan 1   
## 2 36005 36 005 Bronx Bronx 2   
## 3 36047 36 047 Kings Brooklyn 3   
## 4 36081 36 081 Queens Queens 4   
## 5 36085 36 085 Richmond Staten Island 5   
## # … with 1 more variable: geometry <MULTIPOLYGON [US\_survey\_foot]>

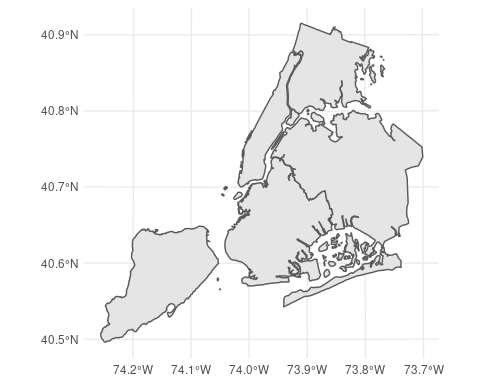
acs\_nyc <- acs\_data2 %>% select("id", "Geographic Area Name",  
 "Percent!!EMPLOYMENT STATUS!!Civilian labor force!!Unemployment Rate",  
 "Estimate!!INCOME AND BENEFITS (IN 2019 INFLATION-ADJUSTED DOLLARS)!!Total households!!Median household income (dollars)"  
 )

names(acs\_nyc)[names(acs\_nyc)== "id"] <- "ID"   
names(acs\_nyc)[names(acs\_nyc)== "Geographic Area Name"] <- "area"   
names(acs\_nyc)[names(acs\_nyc)== "Percent!!EMPLOYMENT STATUS!!Civilian labor force!!Unemployment Rate"] <- "unemployment\_rate"   
names(acs\_nyc)[names(acs\_nyc)== "Estimate!!INCOME AND BENEFITS (IN 2019 INFLATION-ADJUSTED DOLLARS)!!Total households!!Median household income (dollars)"] <- "med\_income"

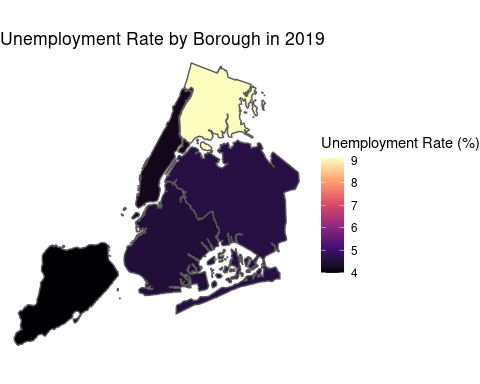
# FOR NYC AS A WHOLE: unemployment = 4.4%, median income = $72,108  
acs\_nyc <-subset(acs\_nyc, ID!= "0400000US36")

acs\_nyc$area[acs\_nyc$area == "Bronx County, New York"] <- "Bronx"  
acs\_nyc$area[acs\_nyc$area == "Kings County, New York"] <- "Brooklyn"  
acs\_nyc$area[acs\_nyc$area == "New York County, New York"] <- "Manhattan"  
acs\_nyc$area[acs\_nyc$area == "Queens County, New York"] <- "Queens"  
acs\_nyc$area[acs\_nyc$area == "Richmond County, New York"] <- "Staten Island"

test\_map <- nyc\_boundaries(  
 geography = "borough",  
 filter\_by = "borough",  
 region = c("brooklyn", "queens", "bronx", "staten island", "manhattan")  
 )  
ggplot(test\_map) +  
 geom\_sf(aes(fill = )) +  
 theme\_minimal()



nyc\_boundaries(geography = "borough") %>%   
 left\_join(acs\_nyc, by = c("borough\_name" = "area")) %>%   
 ggplot() +  
 geom\_sf(aes(fill = unemployment\_rate)) +  
 scale\_fill\_viridis\_c(name = "Unemployment Rate (%)", option = "magma") +  
 theme\_void() +  
 theme(panel.grid = element\_line(color = "transparent")) +  
 labs(title = "Unemployment Rate by Borough in 2019")



nyc\_boundaries(geography = "borough") %>%   
 left\_join(acs\_nyc, by = c("borough\_name" = "area")) %>%   
 ggplot() +  
 geom\_sf(aes(fill = med\_income)) +  
 scale\_fill\_viridis\_c(name = "Median Income ($)", option = "magma") +  
 theme\_void() +  
 theme(panel.grid = element\_line(color = "transparent")) +  
 labs(title = "Median Income by Borough in 2019")

