In Class Work: Webscraping

Complete the following steps

1. Using the `acs` package, download data on mean transportation time ("MEANS OF TRANSPORTATION TO WORK BY TRAVEL TIME TO WORK FOR WORKPLACE GEOGRAPHY") by county for individuals who live in California.

(you can find table information here: https://www.census.gov/programs-surveys/acs/technical-documentation/table-shells.html)

```
library(tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 4.0.5
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --
```

```
## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.4     v dplyr   1.0.7
## v tidyr   1.1.3     v stringr 1.4.0
## v readr   2.0.1     v forcats 0.5.1
```

```
## Warning: package 'ggplot2' was built under R version 4.0.5
```

```
## Warning: package 'tibble' was built under R version 4.0.5
```

```
## Warning: package 'tidyr' was built under R version 4.0.5
```

```
## Warning: package 'readr' was built under R version 4.0.5
```

```
## Warning: package 'purrr' was built under R version 4.0.5
```

```
## Warning: package 'dplyr' was built under R version 4.0.5
```

```
## Warning: package 'stringr' was built under R version 4.0.5
```

```
## Warning: package 'forcats' was built under R version 4.0.5
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(rvest)
```

```
## Warning: package 'rvest' was built under R version 4.0.5
```

```
##
## Attaching package: 'rvest'
```

```
## The following object is masked from 'package:readr':
##
##     guess_encoding
```

```
library(tigris)
```

```
## Warning: package 'tigris' was built under R version 4.0.5
```

```
## To enable
## caching of data, set `options(tigris_use_cache = TRUE)` in your R script or .Rprofile.
```

```
library(lubridate)
```

```
## Warning: package 'lubridate' was built under R version 4.0.5
```

```
##
## Attaching package: 'lubridate'
```

```
## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
```

```
library(gridExtra)
```

```
## Warning: package 'gridExtra' was built under R version 4.0.5
```

```
##
## Attaching package: 'gridExtra'
```

```
## The following object is masked from 'package:dplyr':
##
##     combine
```

```
library(tidycensus)
```

```
## Warning: package 'tidycensus' was built under R version 4.0.5
```

```
##
## Attaching package: 'tidycensus'
```

```
## The following object is masked from 'package:tigris':
##
##     fips_codes
```

```
# Get your own key and save as my_acs_key.txt
#my_acs_key<-readLines("my_acs_key.txt",warn = FALSE)
#acs_key<-my_acs_key

acs_key<-"a0f3f8cc65205f8040f93b4e9168f0f09a4cfdbb"

census_api_key(acs_key,install=FALSE,overwrite =TRUE)
```

```
## To install your API key for use in future sessions, run this function with `install = TRUE`.
```

```
# OR just paste it here.
```

Below, I submit a request using my key to get table B08534, which contains information on MEANS OF TRANSPORTATION TO WORK BY TRAVEL TIME TO WORK FOR WORKPLACE GEOGRAPHY.

```
## Educ Characteristics by County for CA

travel_vars<-get_acs(geography = "county",state="CA",
                     table="B08534",geometry = TRUE)
```

```
## Getting data from the 2015-2019 5-year ACS

## Downloading feature geometry from the Census website.  To cache shapefiles for use in future sessions

## Loading ACS5 variables for 2019 from table B08534. To cache this dataset for faster access to ACS tab

##   |                                                                              |
```

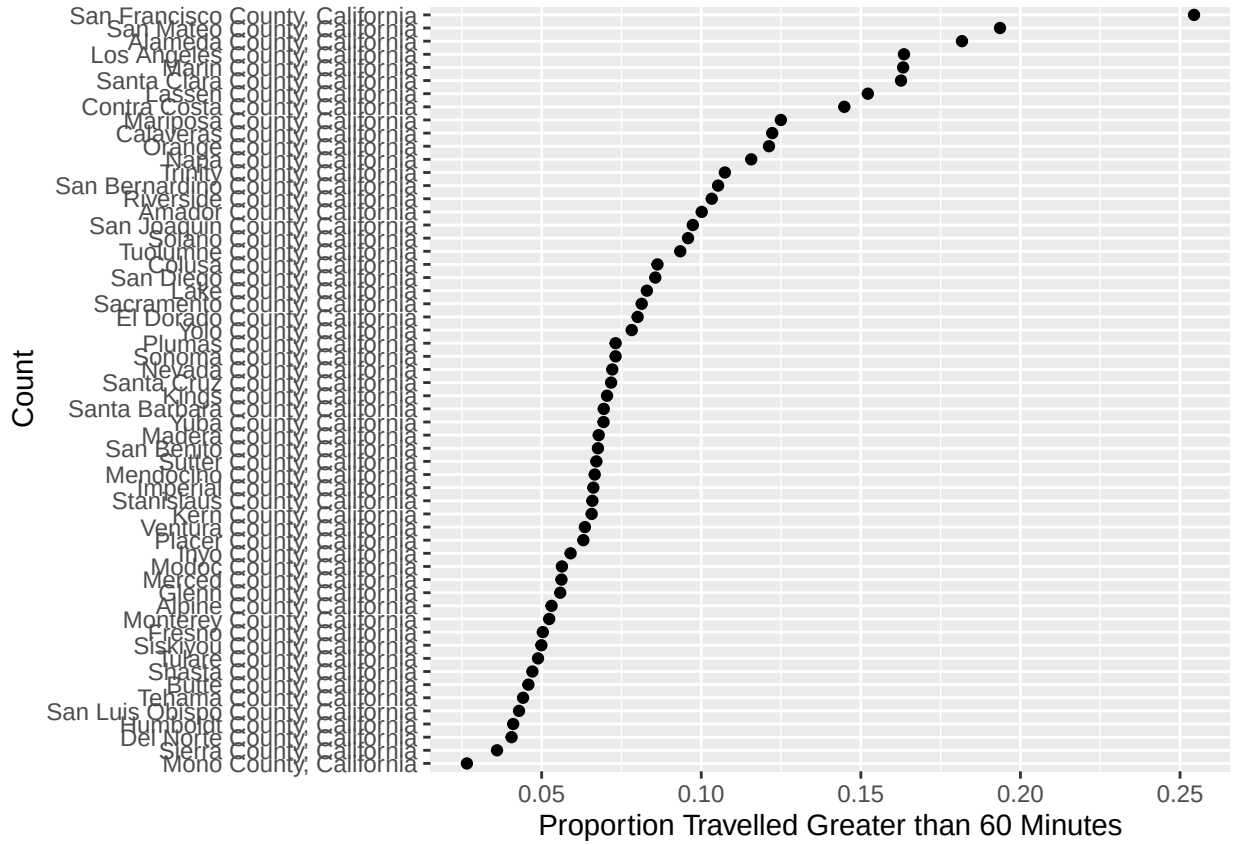
```
## Spread, so that each level of education gets its own column
travel_vars<-travel_vars%>%
  select(GEOID,NAME,variable,estimate)%>%
  spread(key=variable,value = estimate)

## rename to be all lower case
names(travel_vars)<-str_to_lower(names(travel_vars))
```

2. Calculate the proportion of individuals who have commutes of more than one hour.

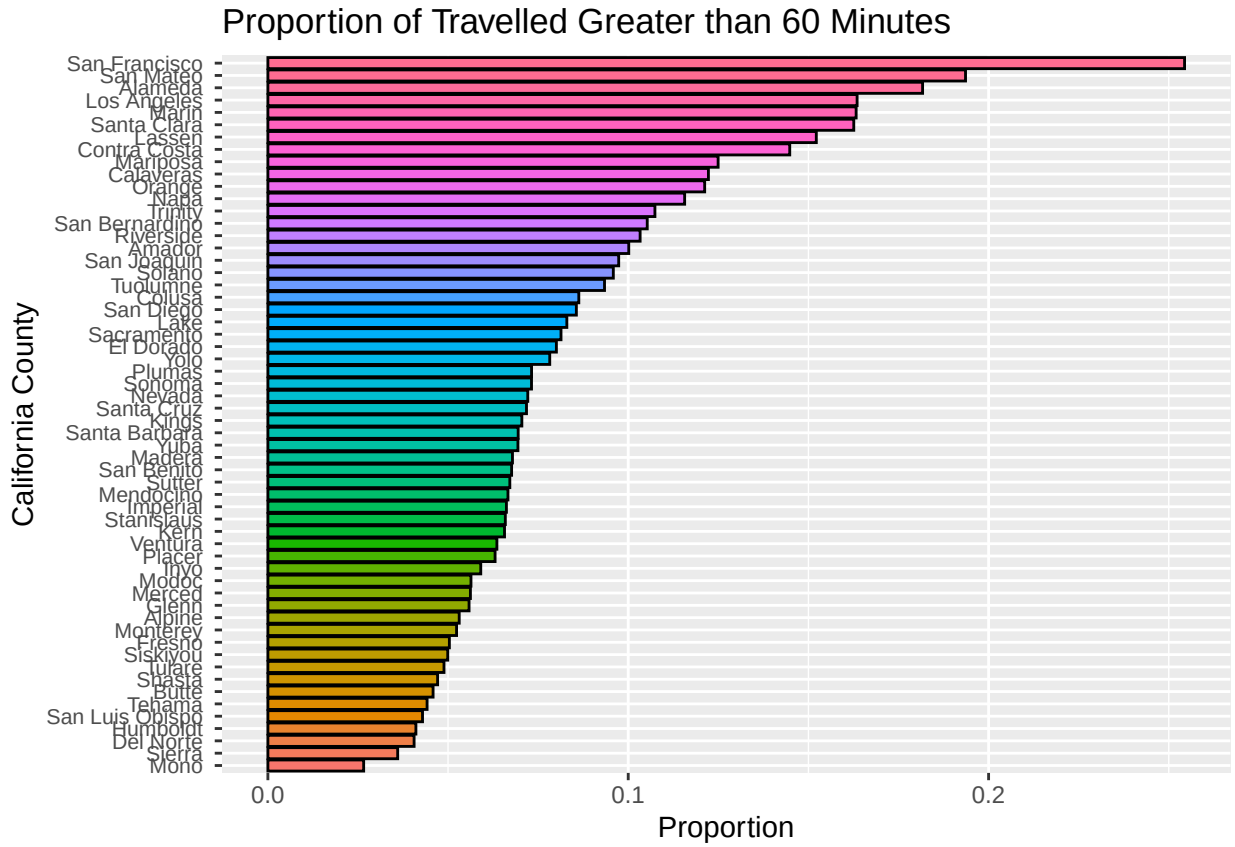
```
travel_vars<-travel_vars%>%
  mutate(trav_hour=(b08534_010)/b08534_001)

## simplify to just proportion
travel_vars<-travel_vars%>%
  select(geoid,name,trav_hour,geometry)
```

3. Plot the results by county, ordered from highest proportion to lowest.

```
gg<-ggplot(travel_vars,aes(y=fct_reorder(name,trav_hour),x=trav_hour))
gg<-gg+geom_point()
gg<-gg+xlab("Proportion Travelled Greater than 60 Minutes")+ylab("Count")
gg
```




```
################This is a modified graph that Looks a little prettier.
travel_vars$name <- gsub('.{18}$', '', travel_vars$name)
#deletes the last 18 characters . = any char, 18 number of char, $ is the end

gg<-ggplot(travel_vars,aes(y=fct_reorder(name,trav_hour),x=trav_hour))
gg<-gg+geom_bar(stat="Identity", aes(fill=fct_reorder(name,trav_hour)), color="black")
#Needs identity to plot the already calculated proportions.aes here, just makes the varied colors
gg<-gg+xlab("Proportion")+ylab("California County")
gg<-gg+ggtitle("Proportion of Travelled Greater than 60 Minutes")
gg<-gg+theme(axis.text.y = element_text(size = 8))# shrink y-axis text size
gg<-gg+theme(legend.position = "none")
gg
```



This is as far as we got in class

4. Plot the proportion of individuals with commutes of more than an hour as a function of the proportion of the population with a bachelor's degree.

```
#### Copied directly from main async lecture with CA substituting TX
## Educ Characteristics by County for CA

educ_vars<-get_acs(geography = "county",state="CA",
                   table="B15003",geometry = TRUE)
```

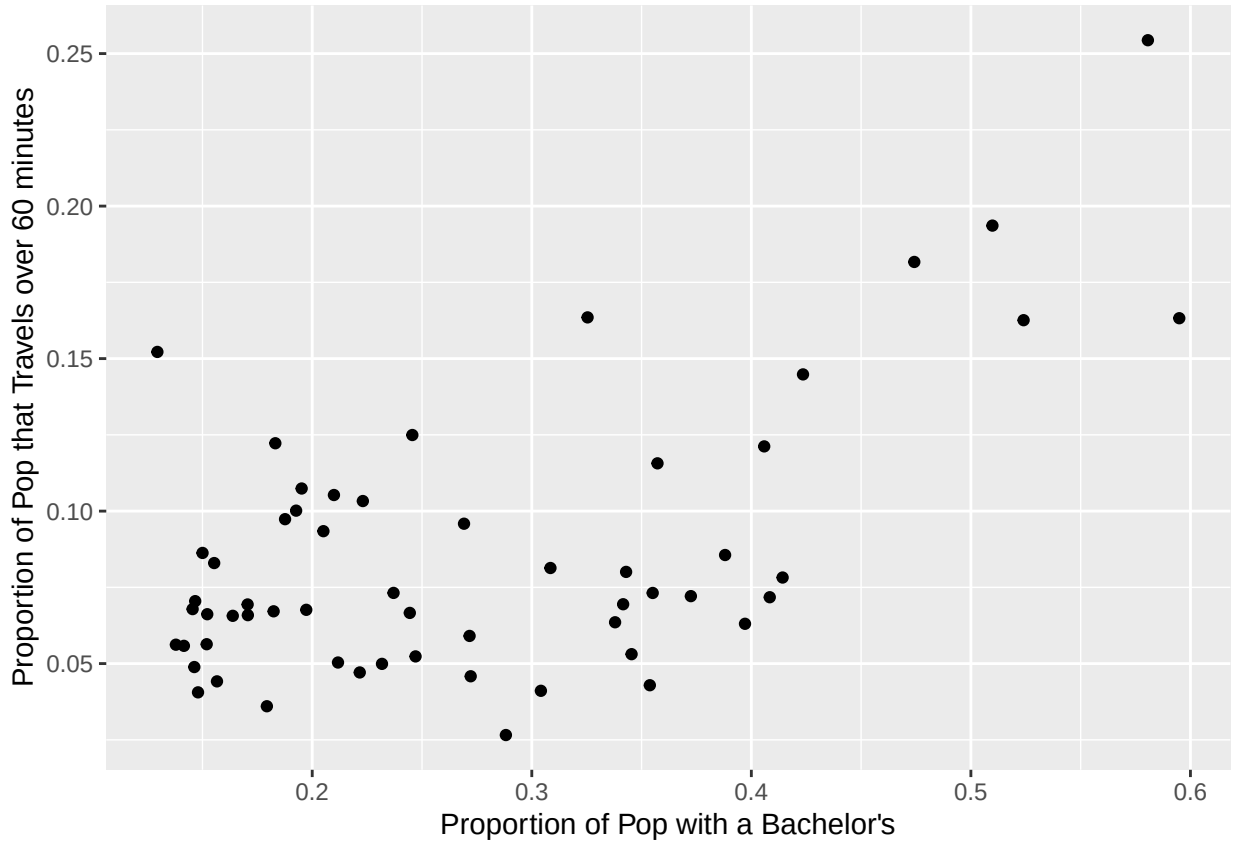
```
## Getting data from the 2015-2019 5-year ACS

## Downloading feature geometry from the Census website.  To cache shapefiles for use in future sessions

## Loading ACS5 variables for 2019 from table B15003. To cache this dataset for faster access to ACS tab
```

```
## Spread, so that each level of education gets its own column
educ_vars<-educ_vars%>%
  select(GEOID,NAME,variable,estimate)%>%
  spread(key=variable,value = estimate)

## rename to be all lower case
names(educ_vars)<-str_to_lower(names(educ_vars))
```

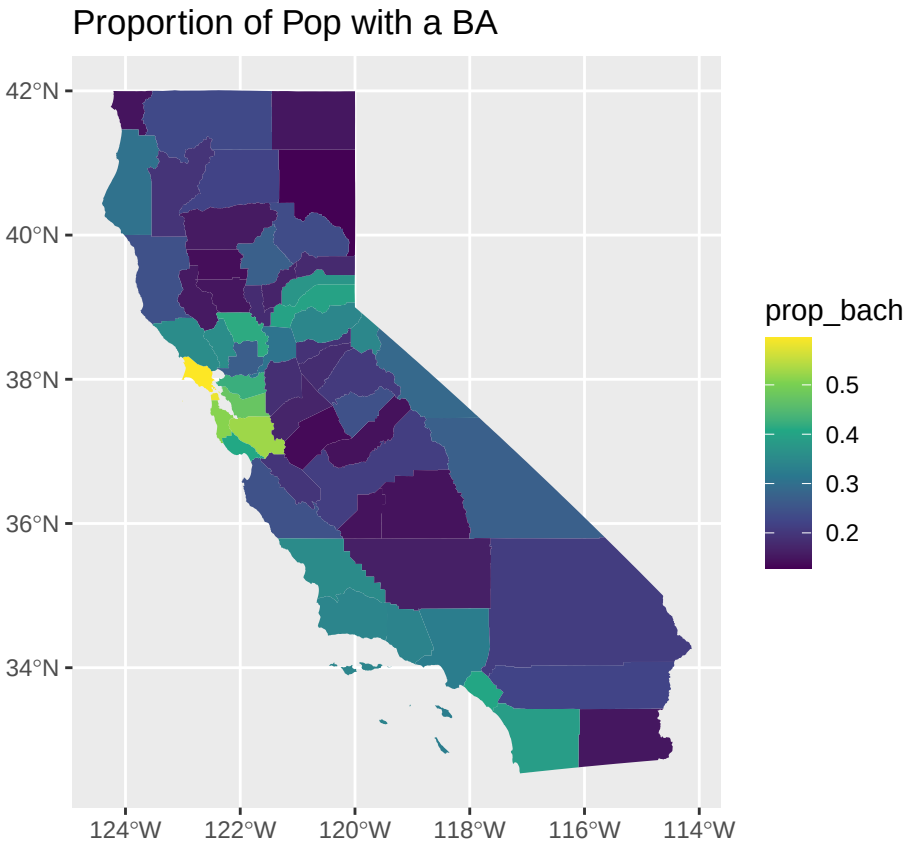
```
## Calculate prop with at least bachelor's for every county
educ_vars<-educ_vars%>%
  mutate(prop_bach=(b15003_022+
                    b15003_023+
                    b15003_024+
                    b15003_025)/b15003_001)

## simplify to just proportion
educ_vars<-educ_vars%>%
  select(geoid,name,prop_bach,geometry)

educ_vars$name <- gsub('.{18}$', '', educ_vars$name)
```

```
educ_vars_2<-educ_vars%>%as_tibble()%>%select(geoid,name,prop_bach)

travel_vars_2<-travel_vars%>%as_tibble()%>%select(geoid,name,trav_hour)

educ_travel<-left_join(educ_vars_2,travel_vars_2,by=c("geoid","name"))
```

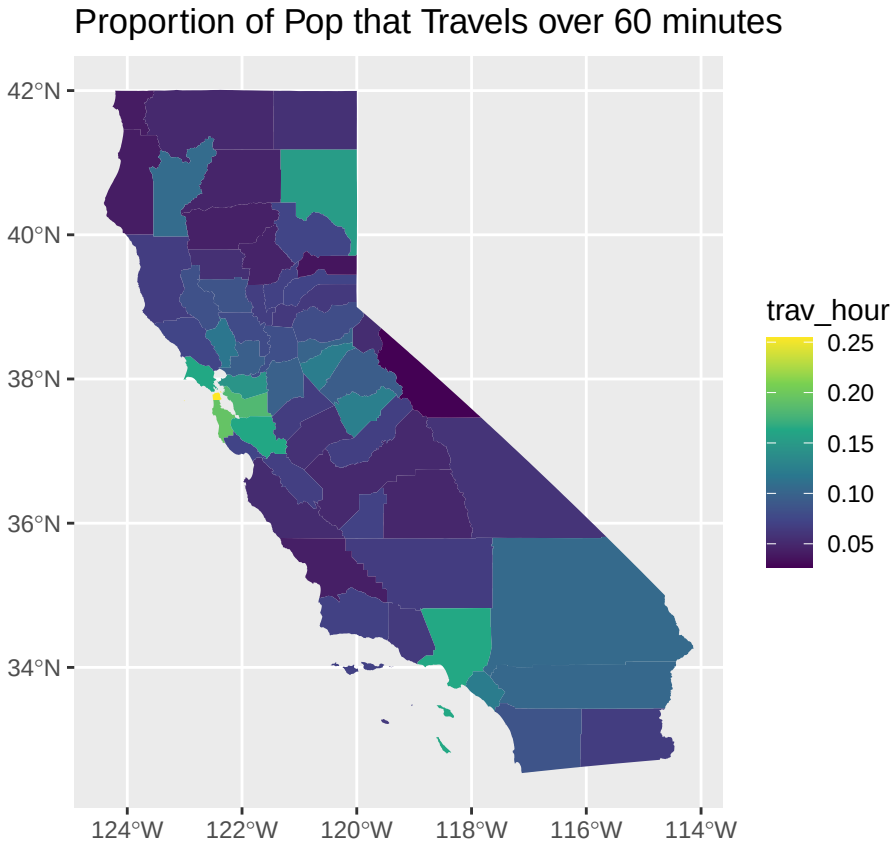
```
gg<-ggplot(educ_travel,aes(y=trav_hour,x=prop_bach))
gg<-gg+geom_point()
gg<-gg+xlab("Proportion of Pop with a Bachelor's")+ylab("Proportion of Pop that Travels over 60 minutes
gg
```



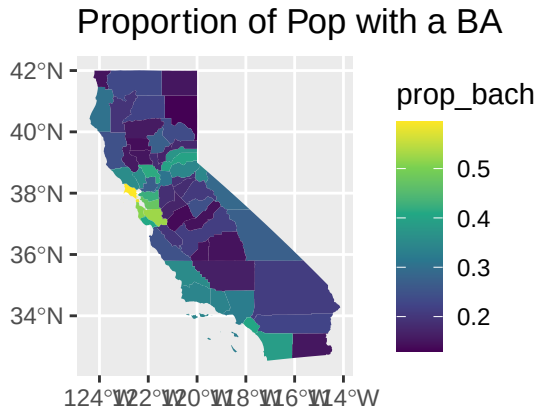
```
gg1<-ggplot(educ_vars,aes(fill=prop_bach))
gg1<-gg1+geom_sf(color=NA)
gg1<- gg1+ scale_fill_viridis_c(option = "viridis")
gg1<-gg1+ggtitle("Proportion of Pop with a BA")
gg1
```

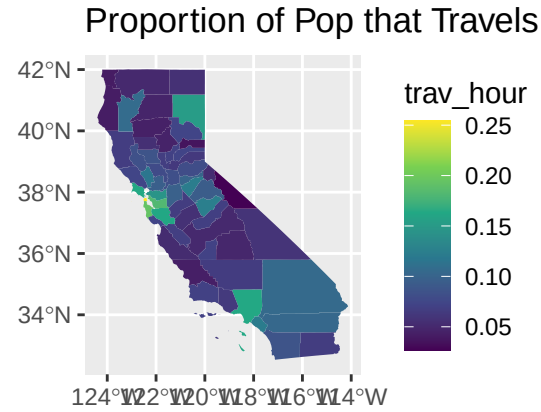



```
gg2<-ggplot(travel_vars,aes(fill=trav_hour))
gg2<-gg2+geom_sf(color=NA)
gg2<- gg2+ scale_fill_viridis_c(option = "viridis")
gg2<-gg2+ggtitle("Proportion of Pop that Travels over 60 minutes")
gg2
```




```
gg_both<-grid.arrange(gg1,gg2)
```






```
gg_both
```

```
## TableGrob (2 x 1) "arrange": 2 grobs
##   z     cells    name           grob
## 1 1 (1-1,1-1) arrange gtable[layout]
## 2 2 (2-2,1-1) arrange gtable[layout]
```

Worked example from class (Doyle notes)

Download data on owner-occupied housing

```
housing_vars<-get_acs(geography = "county",state="CA",
                      table="B25008")
```

```
## Getting data from the 2015-2019 5-year ACS
```

```
## Loading ACS5 variables for 2019 from table B25008. To cache this dataset for faster access to ACS tab
```

```
housing_vars<-housing_vars%>%
  select(GEOID,NAME,variable,estimate)%>%
  spread(key=variable,value = estimate)
```

```
housing_vars<-housing_vars%>%
  mutate(prop_owner_occupied=B25008_002/B25008_001)
```

```
housing_vars%>%
  ggplot(aes(y=prop_owner_occupied,x=fct_reorder(NAME,prop_owner_occupied)))+
  geom_point()+
  coord_flip()
```