Final Project - ACS Analysis

CODE ▾

Le_Linh

HIDE

```
#Read in the csv files
#Create a vector consisting of selected variables
variables <- c("SERIALNO", "REGION", "ST", "ADJINC",  "PWGTP", "AGEP", "CIT",
               "COW", "MAR", "SCHL","SEX", "ESR", "PINCP", "POVPIP", "RAC1P")
#Read in 'psam_pusa.csv'
a <- fread("psam_pusa.csv", header = TRUE, select = variables, data.table = FALSE,
           stringsAsFactors = FALSE)
#Read in 'psam_pusb.csv'
b <- fread("psam_pusb.csv", header = TRUE, select = variables, data.table = FALSE,
           stringsAsFactors = FALSE)
#Read in 'psam_pusc.csv'
c <- fread("psam_pusc.csv", header = TRUE, select = variables, data.table = FALSE,
          stringsAsFactors = FALSE)
#Read in 'psam_pusd.csv'
d <- fread("psam_pusd.csv", header = TRUE, select = variables, data.table = FALSE,
           stringsAsFactors = FALSE)
#Combine 4 files
df <- rbind(a, b, c, d)
#Save combined file
fwrite(df,'population.csv')
#Remove unnecessary files
rm(a, b, c, d, df)
```

HIDE

```r
# Read in the file used for analysis
population <- fread("population.csv", header = TRUE, data.table = FALSE, stringsAsFactors = FALSE)
#Change the columns' names
colnames(population) <- c('Serial.No', 'Region', 'States', 'Inflation', 'Weight', 'Age', 'Citizenship'
         , 'Worker.Class', 'Marital.Status','Education', 'Sex','Employment','Total.Income', 'Inc.Pov.R
         ate', 'Race')
#Create levels and labels for `Region`
region.levels <- c(1,2,3,4,9)
region.labels <- c('Northeast', 'Midwest', 'South', 'West', 'Puerto Rico')
#Create levels and labels for `States`
state.levels <- c(1, 2, 4, 5, 6, 8, 9, 10, 11, 12, 13, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25,
                  26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 44, 45, 46,
                  47, 48, 49, 50, 51, 53, 54, 55, 56, 72)
state.labels <- c('alabama', 'alaska', 'arizona', 'arkansas', 'california', 'colorado', 'connecticut',
         'delaware', 'district of columbia', 'florida', 'georgia', 'hawaii', 'idaho', 'illinois', 'ind
         iana', 'iowa', 'kansas', 'kentucky', 'louisiana', 'maine', 'maryland',  'massachusetts', 'mic
         higan', 'minnesota', 'mississippi', 'missouri', 'montana', 'nebraska', 'nevada', 'new hampshi
         re', 'new jersey', 'new mexico', 'new york', 'north carolina', 'north dakota', 'ohio', 'oklah
         oma', 'oregon', 'pennsylvania', 'rhode island', 'south carolina', 'south dakota', 'tennessee'
         , 'texas', 'utah', 'vermont', 'virginia', 'washington', 'west virginia', 'wisconsin', 'wyomin
         g', 'puerto rico')
# Create citizenship levels and labels
cit.levels <- c(1,2,3,4,5)
cit.labels <- c( 'US Born', 'PR.Guam.VI.NM Born', 'Abroad Born', 'Naturalized Citizen', 'Non US')
# Create class of workers' levels and labels
worker.levels <- c(1,2,3,4,5,6,7,8,9)
worker.labels <-c('Private, for Profit', 'Private, Non-Profit', 'Local','State','Federal',
                  'Self-employed, Not Incor','Self-employed, Incor',
                  'Family work, No Payment','Unemployed')
#Create levels and labels for Marital Status
MS.levels <- c(1,2,3,4,5)
MS.labels <- c('Married','Widowed','Divorced','Separated','Never married/<15')
# Create levels and labels for Education
edu.levels <- c(1,2,3,4,5,6,7,8,9,10, 11,12,13,14,15,16,17,18,19,20,21,22,23,24)
edu.labels <- c( 'No schooling', 'Preschool', 'Kindergarten', 'Grade 1', 'Grade 2', 'Grade 3',
                 'Grade 4', 'Grade 5', 'Grade 6', 'Grade 7', 'Grade 8', 'Grade 9', 'Grade 10',
                 'Grade 11', 'Grade 12, No Diploma','HS Diploma', 'GED/alternative credential',
                 'College, 1y less','College, 1y over',  'Associate', 'Bachelor','Master',
                 'Professional, over bachelor','Doctorate')
# Create levels and labels for 'Employment status'
emp.levels <- c(1,2,3,4,5,6)
emp.labels <- c( 'Civilian employed, at work', 'Civilian employed, not at work','Unemployed',
                 'Armed forces, at work','Armed forces, not at work', 'Not in labor force')
# Create levels and labels for 'Race'
race.levels <- c(1,2,3,4,5,6,7,8,9)
race.labels <- c('White', 'Black/African American', 'American Indian', 'Alaska Native',
                 'American Indian and/or Alaska Native', 'Asian',
                 'Native Hawaiian & Other Pacific Islander', 'Some Other Race',
                 'Two/More Races')
# Create factor variables and factor level names
factor.pop <- population %>%
  mutate(Region = factor(Region, levels = region.levels , labels = region.labels),
         States = factor(States, levels = state.levels, labels = state.labels),
         Citizenship = factor(Citizenship, levels = cit.levels, labels = cit.labels),
         Worker.Class = factor(Worker.Class, levels = worker.levels, labels = worker.labels),
         Marital.Status= factor(Marital.Status, levels = MS.levels, labels = MS.labels),
         Education = factor(Education, levels = edu.levels, labels = edu.labels),
         Sex = factor(Sex, levels = c(1,2), labels = c('Male', 'Female')),
         Employment = factor(Employment, levels = emp.levels, labels = emp.labels),
         Race = factor(Race, levels = race.levels, labels = race.labels))
# Save factorized data into a .rds file
write_rds(factor.pop, "population.rds")
```

HIDE

```
# Load the data
factor.pop <-read_rds("population.rds")
# Adjust income value with inflation rate
adjust.pop <- factor.pop %>%
  mutate(Total.Income = (Inflation / 1000000) * Total.Income)
```

HIDE

```
# Collapse some categorical variables into more meaningful groups
clean.pop <- adjust.pop %>%
  mutate(Citizenship = fct_collapse(Citizenship,
            US = c('US Born', 'PR.Guam.VI.NM Born', 'Abroad Born', 'Naturalized Citizen'),
            Non.US = 'Non US'),
        Worker.Class = fct_collapse(Worker.Class,
            Private = c('Private, for Profit', 'Private, Non-Profit'),
            Government = c('Local', 'State', 'Federal'),
            Self.Employed = c('Self-employed, Not Incor','Self-employed, Incor',
                              'Family work, No Payment'),
            Unemployed = 'Unemployed'),
        Education = fct_collapse(Education,
            No.school = 'No schooling',
            Pre.Primary = c('Preschool', 'Kindergarten'),
            Primary = c('Grade 1', 'Grade 2', 'Grade 3', 'Grade 4', 'Grade 5', 'Grade 6'),
            Secondary = c('Grade 7', 'Grade 8', 'Grade 9', 'Grade 10', 'Grade 11',
                          'Grade 12, No Diploma','HS Diploma', 'GED/alternative credential'),
            College = c('College, 1y less', 'College, 1y over'),
            Bachelor = c('Associate', 'Bachelor'),
            Master = c('Master', 'Professional, over bachelor'),
            PhD = 'Doctorate'),
        Employment = fct_collapse(Employment,
            Civil.employed = c('Civilian employed, at work', 'Civilian employed, not at work'),
            Unemployed = 'Unemployed',
            Arm.force = c('Armed forces, at work','Armed forces, not at work'),
            Not.Labor.Force = 'Not in labor force'),
        Race = fct_collapse(Race,
            White = 'White',
            Black = 'Black/African American',
            Asian = 'Asian',
            Others = c( 'American Indian', 'Alaska Native', 'American Indian and/or Alaska Native',
                       'Native Hawaiian & Other Pacific Islander', 'Some Other Race','Two/More Races'
         )),
        Age.Group = case_when(Age <= 15 ~ 'Children',
                              Age >=16 & Age <=23 ~ "Late Gen Z",
                              Age >= 24 & Age <= 39 ~ 'Millennial',
                              Age >=40 & Age <= 55 ~ 'Baby Boomer',
                              Age > 55 ~ 'Senior'),
        Income.Group = case_when (Total.Income < 20000 ~ 'Very Low',
                                  Total.Income >= 20000 & Total.Income < 44999 ~ 'Low',
                                  Total.Income >= 45000 & Total.Income < 139999 ~ 'Middle',
                                  Total.Income >= 140000 & Total.Income < 149999 ~ 'Upper Middle',
                                  Total.Income >= 150000 ~ 'High')
          )
```

```r
# Create levels for Age Group and Income Group
age.levels <- c('Children','Late Gen Z', 'Millennial', 'Baby Boomer','Senior')
inc.levels <- c('Very Low', 'Low', 'Middle', 'Upper Middle','High')
clean.pop <- clean.pop %>%
  mutate(Age.Group = factor(Age.Group, levels = age.levels),
         Income.Group = factor(Income.Group, levels = inc.levels))
```

Data Summary

The data come from the 2018 American Community Survey (ACS) 5-year Public Use Microdata Samples (PUMS), which is a sample of the actual responses collected by the American Community Survey between 2014-2018 and split into population and household characteristics. This research would focus on the population dataset only. Population dataset is originally a dataset with 183 columns and 15,840,681 rows on approximately five percent of the United States population (one percent/year). To protect strict confidentiality of survey respondents, many variables in the dataset have been modified, for instance, high incomes are replaced with top-coded value. The PUMS are microdata, so each record represents a sample of the children population and individuals are not sampled with equal probability, that is, people who have a greater chance of being sampled have a lower weight to reflect this fact. Therefore, the dataset contains a PUMS person weights (PWGTP) to account for the fact. The original dataset would be filtered into 15 columns and rows would be filtered out all the missing values of each variable and outliers of numerical variables to reduce the size of the big dataset. There are 2 new columns are created which are `Age.Group` and `Income.Group` .

Some variables are collapsed into more general groups for meaningful comparison as below:
- Citizenship is collapsed into 2 groups `US` (including ‘US Born’, ‘PR.Guam.VI.NM Born’, ‘Abroad Born’, ‘Naturalized Citizen’), who have been US citizens, and `Non.Us` who have not become US citizens yet.
- Class of Worker is collapsed in to 4 groups: `Private` (including ‘Private, for Profit’, ‘Private, Non-Profit’), `Government` (including ‘Local’, ‘State’, ‘Federal’), `Self.Employed` (including Self-employed, Not Incor’,‘Self-employed, Incor’, ‘Family work, No Payment’), and `Unemployed` (‘Unemployed’).
- Education is collapsed into 8 groups: `No.school` (including ‘No schooling’), `Pre.Primary` (including ‘Preschool’ and ‘Kindergarten’), `Primary` (including ‘Grade 1’, ‘Grade 2’, ‘Grade 3’, ‘Grade 4’, ‘Grade 5’, ‘Grade 6’), `Secondary` (including ‘Grade 7’, ‘Grade 8’, ‘Grade 9’, ‘Grade 10’, ‘Grade 11’, ‘Grade 12, No Diploma’,‘HS Diploma’, ‘GED/alternative credential’), `College` (including ‘College, 1y less’, ‘College, 1y over’), `Bachelor` (including ‘Associate’, ‘Bachelor’), `Master` (including Master’, ‘Professional, over bachelor’), and `PhD` (including ‘Doctorate’).
- Employment is collapse into 4 groups : `Civil.employed` (including ‘Civilian employed, at work’, ‘Civilian employed, not at work’), `Unemployed` (including ‘Unemployed’), `Arm.force` (including ‘Armed forces, at work’,‘Armed forces, not at work’), and `Not.Labor.Force` (including ‘Not in labor force’).
- Race is collapsed into 4 groups : `White` including ‘White’, `Black` (including ‘Black/African American’), `Asian` (including ‘Asian’), and `Others` (including ‘American Indian’, ‘Alaska Native’, ‘American Indian and/or Alaska Native’, ‘Native Hawaiian & Other Pacific Islander’, ‘Some Other Race’,‘Two/More Races’)
- 2 new columns are created:

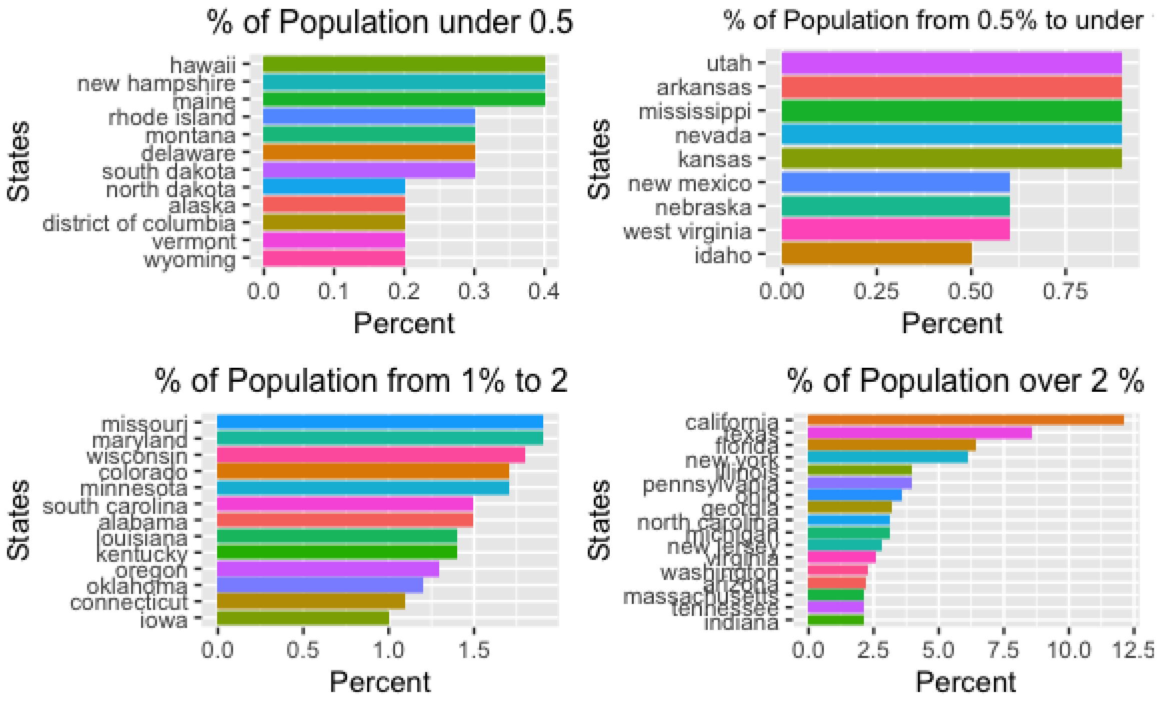
+ `Age.Group` consists of 5 groups: `Children` (Under 15), `Late Gen Z` (from 16 to 23), `Millennial` (from 24 to 39),

`Baby Boomer` (from 40 to 55), and `Senior` (over 55).
+ `Income.Group` contains 5 groups: `Very Low` (under $20,000), `Low` (from $20,000 to $44,999), `Middle` (from $45,000 to $139,999), `Upper Middle` (from $140,000 to $149,999),and `High` (over $150,000)

Overview of population by states

HIDE

```
# Generate proportion of the population by States
state.tbl <- clean.pop %>%
  count(States, wt = Weight) %>%
  mutate(region = States, # Create 'region' column to match with map data
         Percent = 100*round(n/sum(n), digits = 3)) %>%
  arrange(desc(n))

# Proportion of the population under 1%
s1 <- state.tbl %>%
  filter(Percent < 0.5) %>%
  ggplot(aes(x= reorder(States, n), y = Percent, fill = States)) +
  geom_col() +
  labs(x = 'States',title = "% of Population under 0.5%") +
  theme(plot.title = element_text(hjust = 0.5, size = 12), legend.position = 'none') +
  coord_flip()

s2 <- state.tbl %>%
  filter(Percent >= 0.5, Percent < 1) %>%
  ggplot(aes(x= reorder(States, n), y = Percent, fill = States)) +
  geom_col() +
  labs(x = 'States',title = " % of Population from 0.5% to under 1%") +
  theme(plot.title = element_text(hjust = 0.5, size = 10), legend.position = 'none') +
  coord_flip()
# Proportion of the population from 1% to 2%
s3 <- state.tbl %>%
  filter(Percent >= 1, Percent <= 2) %>%
  ggplot(aes(x= reorder(States, n), y = Percent, fill = States)) +
  geom_bar(stat = "identity") +
  labs(x = 'States',title = "% of Population from 1% to 2 %") +
  theme(plot.title = element_text(hjust = 0.5, size = 12), legend.position = 'none') +
  coord_flip()

# Proportion of the population over 2%
s4 <- state.tbl %>%
  filter(Percent > 2) %>%
  ggplot(aes(x= reorder(States, n), y = Percent, fill = States)) +
  geom_bar(stat = "identity") +
  labs(x = 'States',title = "% of Population over 2 %") +
  theme(plot.title = element_text(hjust = 0.5, size = 12), legend.position = 'none') +
  coord_flip()

# Combine 3 plots
  plot_grid(s1, s2, s3, s4)
```



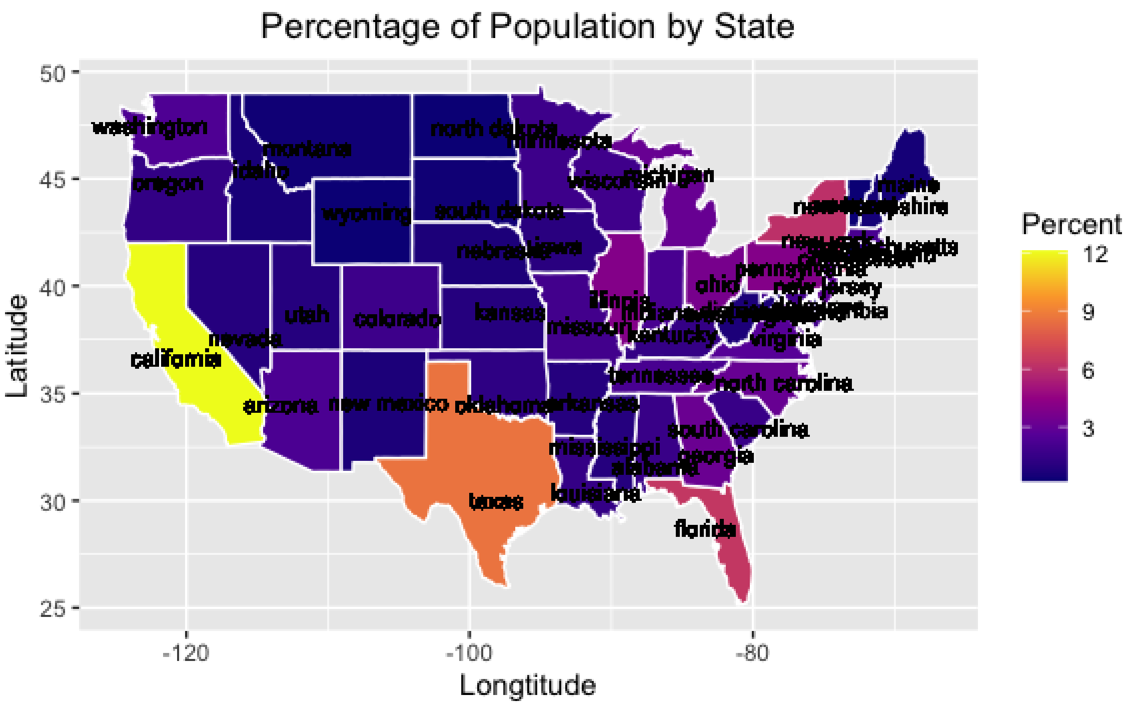
HIDE

```
# Create in-line code
  n1 <- nrow(state.tbl%>%filter(Percent < 0.5))
  n2 <- nrow(state.tbl%>%filter(Percent >= 0.5, Percent < 1))
  n3 <- nrow(state.tbl%>%filter(Percent >= 1, Percent <= 2))
  n4 <- nrow(state.tbl%>%filter(Percent >2))
  min.st <- min(state.tbl$Percent)
  lo.st <- as.vector(filter(state.tbl,Percent == min(state.tbl$Percent))$States)
  avg.low <- floor(mean(filter(state.tbl,Percent == min(state.tbl$Percent))$n))
  hi.st <- as.vector(filter(state.tbl,Percent == max(state.tbl$Percent))$States)
  max.st <- max(state.tbl$Percent)
  p <- filter(state.tbl,Percent == max(state.tbl$Percent))$n
  options(scipen = 999)
```

As can be seen from the graphs, there are 12 states with proportion of population less than 0.5%, 9 states with proportion of population from 0.5% to less than 1%, 13 states with proportion of population from 1% to 2%, and 17 states with proportion of population more than 2%. The states with the lowest percentage of population are **north dakota, alaska, district of columbia, vermont, wyoming** with only 0.2 %, equivalent to approximately 676405 people while the state with the highest percentage is **california**, with 12.1%, equivalent to 39148760 people.

HIDE

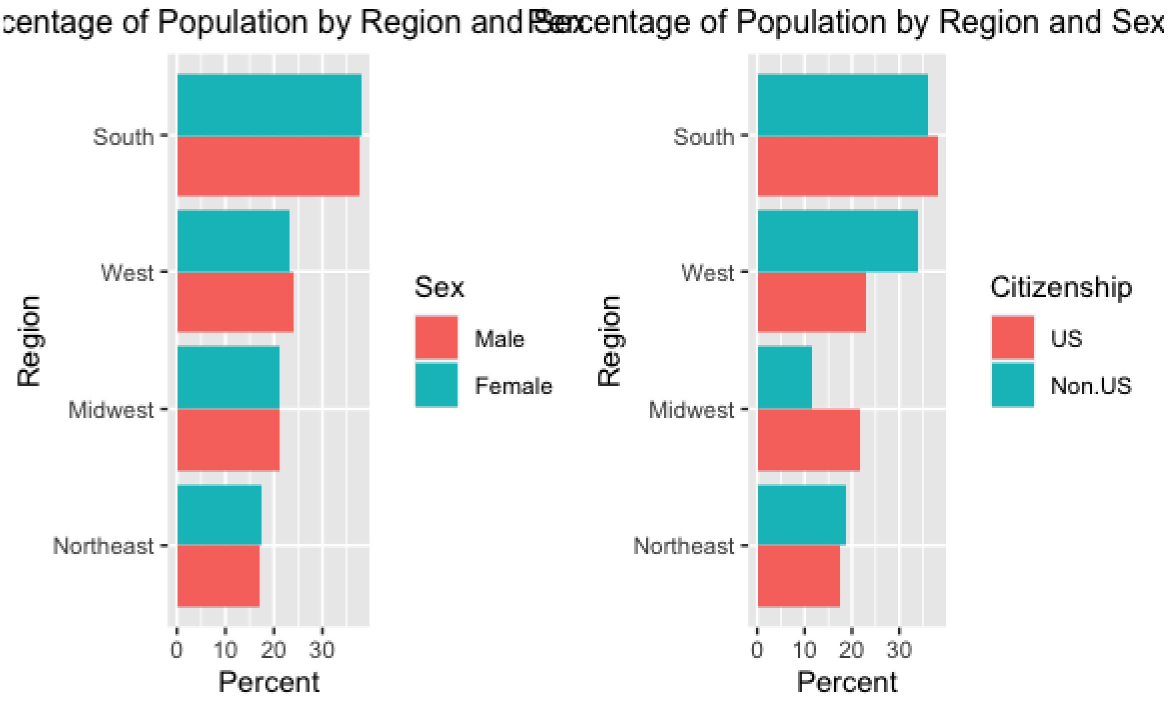
```r
# Create map plot for states
states_map <- map_data("state")
# Calculate average longitude and latitude
state.map <- state.pop %>%
  group_by(region) %>%
  summarise(long = mean(long), lat = mean(lat), group = group)
# Join 2 dataframe by region
state.pop <- left_join(states_map, state.tbl, by = 'region')
# Plot the % of population by state
ggplot(state.pop , aes(long, lat, group = group)) +
  geom_polygon(aes(fill = Percent), color = "white")+
  geom_text(aes(label = region), data = state.map,  size = 3, hjust = 0.5) +
  scale_fill_viridis_c(option = "C") +
  labs( x = 'Longtitude', y = 'Latitude', title = "Percentage of Population by State") +
  theme(plot.title = element_text(hjust = 0.5))
```



Overview of population by Region, Sex and Citizenship

HIDE

```r
pop.region <- clean.pop %>%
  count(Region, wt = Weight) %>%
  mutate(Percent = 100*round(n/sum(n), digits = 3))
# Create in-line code
hi.reg <- max(pop.region$Percent)
lo.reg <- min(pop.region$Percent)
# Calculate the percentage of population by region
r1 <- clean.pop %>%
  group_by(Sex) %>%
  count(Region, wt = Weight) %>%
  mutate(Percent = 100*round(n/sum(n), digits = 3)) %>%
  ggplot(aes(x= reorder(Region, n), y = Percent, fill = Sex)) +
  geom_col(position = "dodge") +
  labs(x = 'Region',title = "Percentage of Population by Region and Sex") +
  theme(plot.title = element_text(hjust = 0.5, size = 12)) +
  coord_flip()
# Calculate the percentage of population by region and race
r2 <- clean.pop %>%
  group_by(Citizenship) %>%
  count(Region, wt = Weight) %>%
  mutate(Percent = 100*round(n/sum(n), digits = 3)) %>%
  ggplot(aes(x= reorder(Region, n), y = Percent, fill = Citizenship)) +
  geom_col(position = "dodge") +
  labs(x = 'Region',title = "Percentage of Population by Region and Sex") +
  theme(plot.title = element_text(hjust = 0.5, size = 12)) +
  coord_flip()
plot_grid(r1,r2)
```

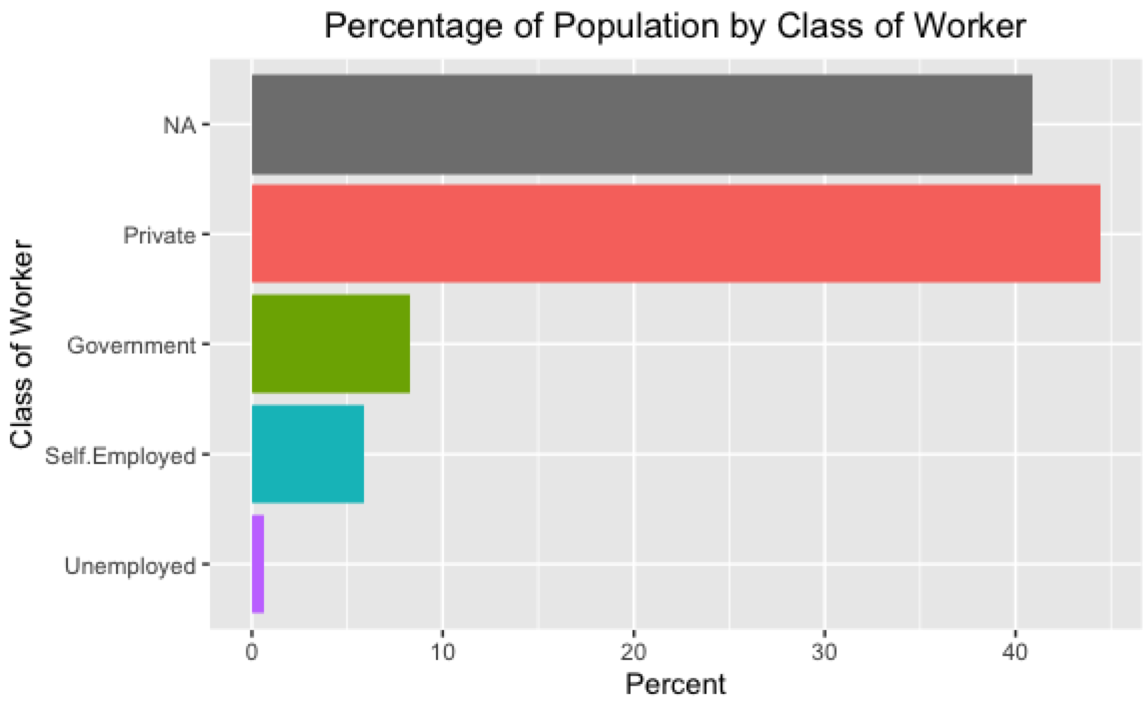


As can be seen from the graphs, US residents populated mostly in the South of USA with 37.9%, and fewest in the Northeast with 17.4% and the distribution of male and female across region quite equal. However, US citizens tend to habitat in the South and Midwest, while US residents without citizenship prefer to live in the West and Northeast. The difference in the number of US and Non Us are significant in the West and Midwest, yet there is a slight discrepancy in that number in the South and Northeast.

Overview of population by Class of Worker

HIDE

```
# Generate the percentage of population by Class of Worker
class <- clean.pop %>%
  count(Worker.Class, wt = Weight) %>%
  mutate(Percent = 100*round(n/sum(n), digits = 3)) %>%
  arrange(desc(Percent))

# Create in-line code
pri <- filter(class, Worker.Class == 'Private')$Percent
gov <- filter(class, Worker.Class == 'Government')$Percent
self <- filter(class, Worker.Class == 'Self.Employed')$Percent
unem <- filter(class, Worker.Class == 'Unemployed')$Percent
na.cla <- filter(class, is.na(Worker.Class))$Percent

# Plot the percentage of population by Class of Worker
class %>%
  ggplot(aes(x= reorder(Worker.Class, n), y = Percent, fill = Worker.Class)) +
  geom_bar(stat = "identity") +
  labs(x = 'Class of Worker',title = "Percentage of Population by Class of Worker") +
  theme(plot.title = element_text(hjust = 0.5), legend.position = 'none') +
  coord_flip()
```

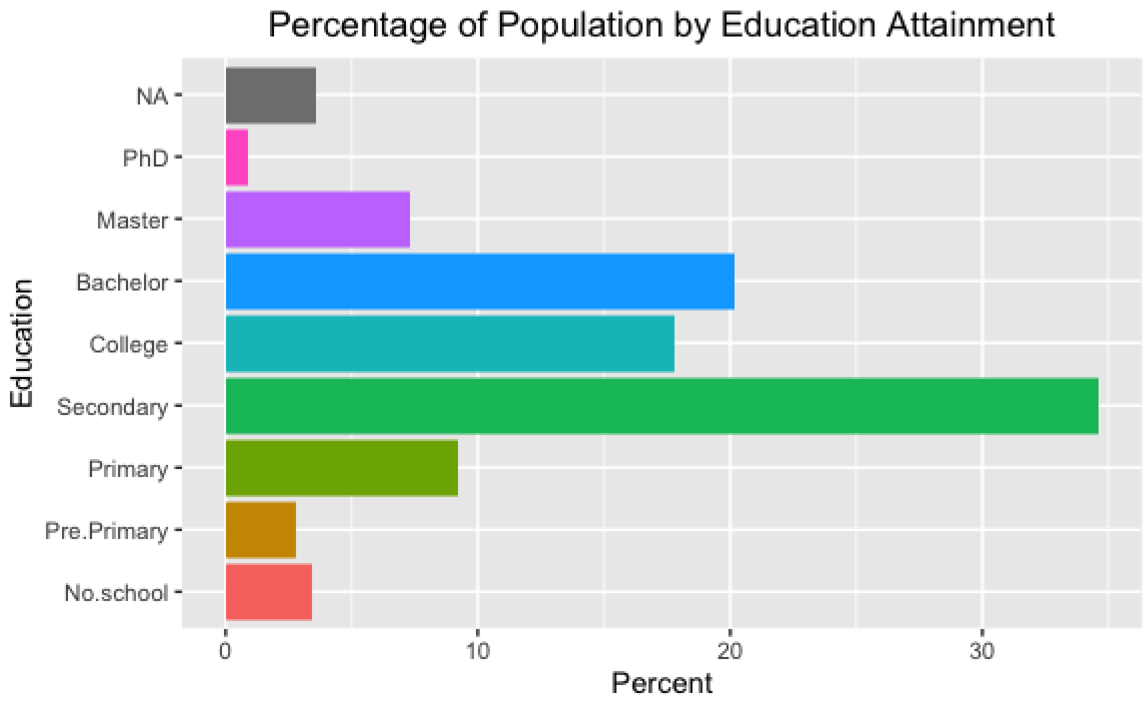


The bar chart above illustrates that US people work mostly in private enterprises with 44.4%, while 8.3 % work for government, and only 5.9% run their own businesses. Unemployed rate is 0.6%. However, it’s worth noting that there is 40.9 % of total population that was not recorded in the data, so those numbers might be not a good representative of the whole population in terms of class of worker.

Overview of population by Education Attainment

HIDE

```r
# Generate the percentage of population by Education Attainment
edu <- clean.pop %>%
  count(Education, wt = Weight) %>%
  mutate(Percent = 100*round(n/sum(n), digits = 3))

# Create in-line code
ba <- filter(edu, Education == 'Bachelor')$Percent
co <- filter(edu, Education == 'College')$Percent
dip <- filter(edu, Education == 'HS Diploma')$Percent
ma <- filter(edu, Education == 'Master')$Percent
PhD <- filter(edu, Education == 'PhD')$Percent
no <- filter(edu, Education == 'No.school')$Percent
sec <- filter(edu, Education == 'Secondary')$Percent
na.edu <- filter(edu, is.na(Education))$Percent

# Plot the percentage of population by Education Attainment
edu %>%
  ggplot(aes(fct_inorder(Education), Percent, fill = Education)) +
  geom_bar(stat = "identity") +
  labs(x = 'Education',title = "Percentage of Population by Education Attainment") +
  theme(plot.title = element_text(hjust = 0.5), legend.position = "none") +
  coord_flip()
```

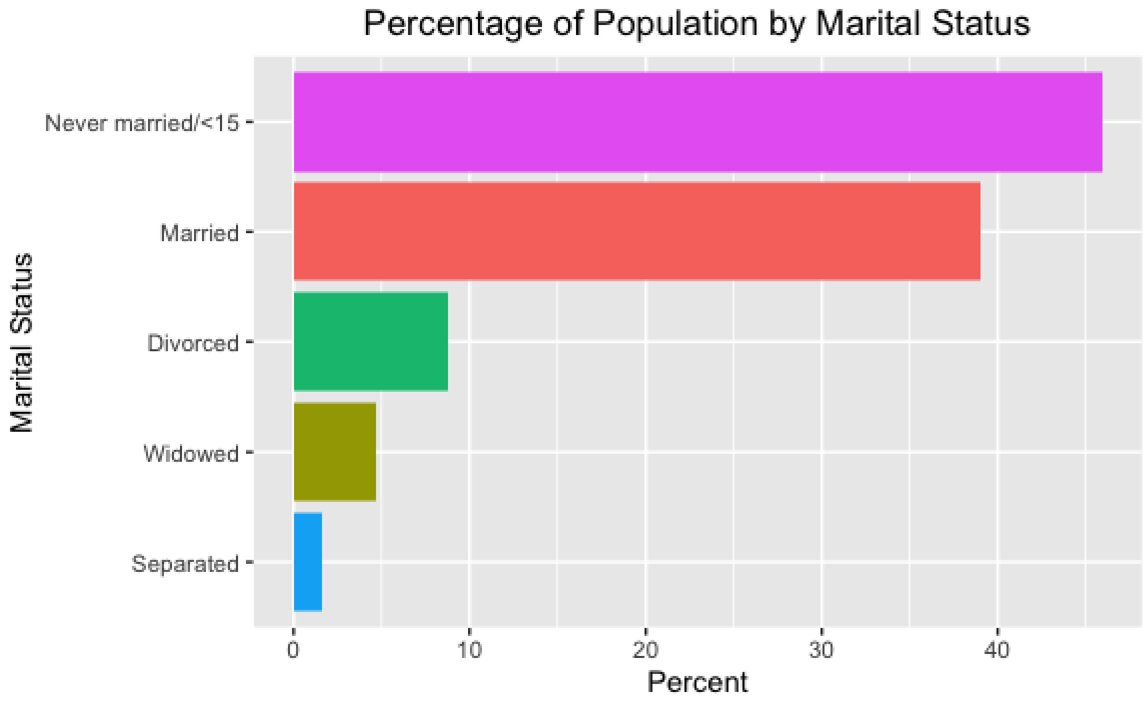


As can be seen from the graph above, the majority of US people achieved secondary attainment with 34.6%, followed by a bachelor's degree with 20.2% and a college diploma with 17.8%. There is 3.4% of population uneducated at school. In terms of post-secondary education, the proportion of Master degree holders is significantly higher than that of PhD holders (7.3% and 0.9% respectively), but less than half of the Bachelor's proportion.

Overview of population by Marital Status

HIDE

```r
# Generate the percentage of population by Marital Status
marital <- clean.pop %>%
  count(Marital.Status, wt = Weight) %>%
  mutate(Percent = 100*round(n/sum(n), digits = 3))

# Create in-line code
never <- filter(marital, Marital.Status == 'Never married/<15')$Percent
mar <- filter(marital, Marital.Status == 'Married')$Percent
div <- filter(marital, Marital.Status == 'Divorced')$Percent
sum <- filter(marital, Marital.Status == 'Separated')$Percent +
       filter(marital, Marital.Status == 'Widowed')$Percent

# Plot the percentage of population by Marital Status
marital %>%
  ggplot(aes(x= reorder(Marital.Status, n), y = Percent, fill = Marital.Status)) +
  geom_bar(stat = "identity") +
  labs(x = 'Marital Status',title = "Percentage of Population by Marital Status") +
  theme(plot.title = element_text(hjust = 0.5), legend.position = "none") +
  coord_flip()
```

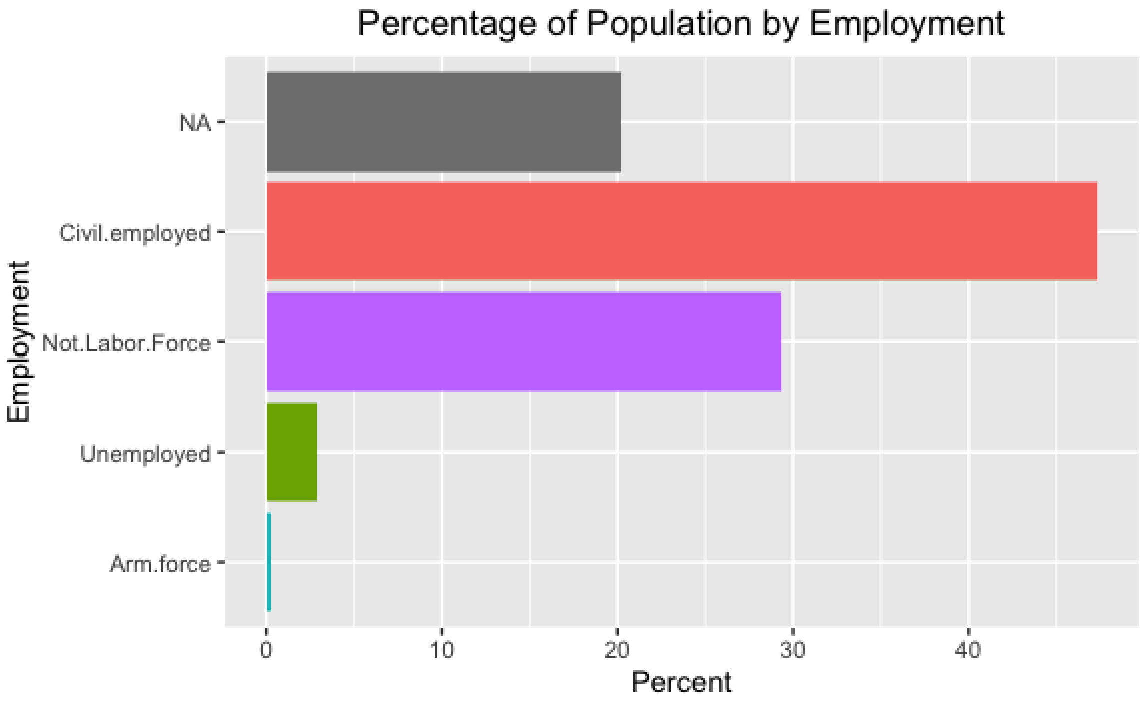


The graph shows that most of US residents are under 15 years old or never married, which accounts for 45.9%, 39% has married, whereas 8.8% accounts for divorced population, 6.3% accounts for both separated and widowed percentage in total.

Overview of population by Employment status

HIDE

```r
# Generate the percentage of population by Employment status
emp <- clean.pop %>%
  count(Employment, wt = Weight) %>%
  mutate(Percent = 100*round(n/sum(n), digits = 3))

# Create in-line code
cil <- filter(emp, Employment == 'Civil.employed')$Percent
not.lf <- filter(emp, Employment == 'Not.Labor.Force')$Percent
unempl <- filter(emp, Employment == 'Unemployed')$Percent
arm <- filter(emp, Employment == 'Arm.force')$Percent

# Plot the percentage of population by Marital Status
emp %>%
  ggplot(aes(x= reorder(Employment, n), y = Percent, fill = Employment)) +
  geom_bar(stat = "identity") +
  labs(x = 'Employment',title = "Percentage of Population by Employment") +
  theme(plot.title = element_text(hjust = 0.5), legend.position = "none") +
  coord_flip()
```

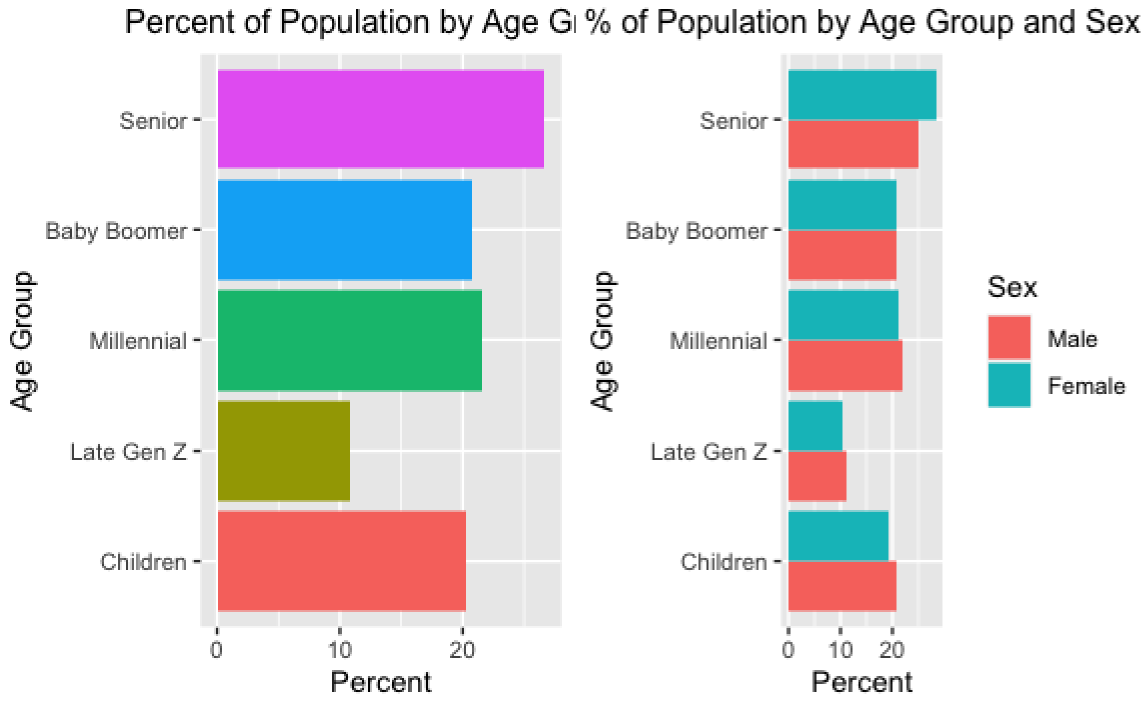


As can be seen from the plot, the majority of US people are working, which accounts for 47.3%, while 2.9% are unemployed, which means they are actively looking for a job but they have not got one. It is noticeable that 29.3% are not in the labor force, who are mainly students, homemakers, retired workers, seasonal workers and become dependents of civilian workers. Only a small proportion (0.3%) of the total population work in the army force.

Overview of population by Age Group

HIDE

```
# Summary statistics of Age
mean.age <- round(wtd.mean(clean.pop$Age, weights= clean.pop$Weight, normwt = FALSE), digits = 0)
qtl.age <- wtd.quantile(clean.pop$Age,weights= clean.pop$Age, normwt = FALSE)
q1.age <- qtl.age[2]
q3.age <- qtl.age[4]
sd.age <- round(sqrt(wtd.var(clean.pop$Age, weights= clean.pop$Weight, normwt = FALSE)), digits = 0)

# Generate the percentage of population by Age Group
age.grp <- clean.pop %>%
  count(Age.Group, wt = Weight) %>%
  mutate(Percent = 100*round(n/sum(n), digits = 3))

# Generate the percentage of population by Age Group and Sex
age.grp.sex <- clean.pop %>%
  group_by(Sex) %>%
  count(Age.Group, wt = Weight) %>%
  mutate(Percent = 100*round(n/sum(n), digits = 3))

# Create in-line code
sen <- filter(age.grp, Age.Group == 'Senior')$Percent
f.sen <- filter(age.grp.sex, Age.Group == 'Senior', Sex == 'Female')$Percent
m.sen <- filter(age.grp.sex, Age.Group == 'Senior', Sex == 'Male')$Percent
f.chi <- filter(age.grp.sex, Age.Group == 'Children', Sex == 'Female')$Percent
m.chi <- filter(age.grp.sex, Age.Group == 'Children', Sex == 'Male')$Percent

# Plot the percentage of population by Age group
a1 <- age.grp %>%
  ggplot(aes(fct_inorder(Age.Group), Percent, fill = Age.Group)) +
  geom_bar(stat = "identity") +
  labs(x = 'Age Group',title = "Percent of Population by Age Group") +
  theme(plot.title = element_text(hjust = 0.5, size = 12), legend.position = "none") +
  coord_flip()
# Plot the percentage of population by Age group and Sex
a2 <- age.grp.sex %>%
  ggplot(aes(fct_inorder(Age.Group), Percent, fill = Sex)) +
  geom_col(position = 'dodge') +
  labs(x = 'Age Group',title = "% of Population by Age Group and Sex") +
  theme(plot.title = element_text(hjust = 0.5, size = 12)) +
  coord_flip()
# Combine 2 plots
plot_grid(a1,a2)
```



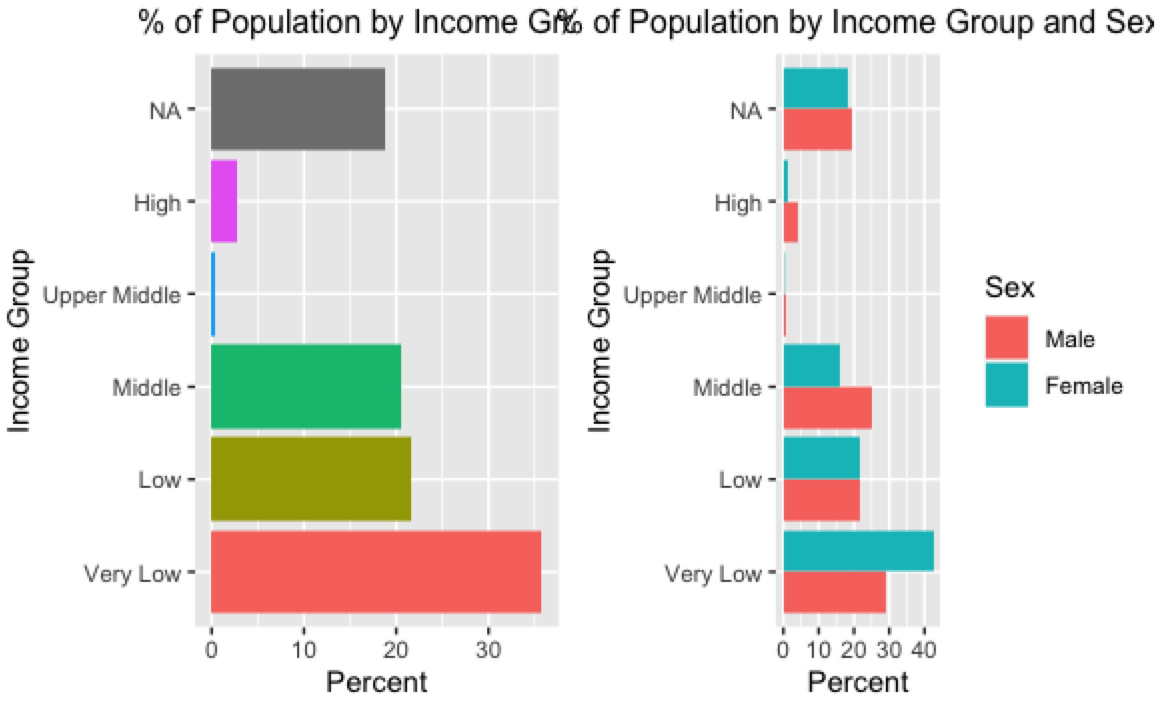
The average age of US residents is 38 years old, and 50% of the total population age is in the range of 42 and 69 years old. The population age deviates from the average age by +/-23.

As can be seen from the plots, the highest percentage of the total population belongs to senior (over 55 years old), with 26.7%, which indicates that the USA is an aging country, and the highest percentage of the population belongs to late Gen Z (from 16 to 23 years old). Moreover, while the proportion of male tends to be higher than that of female across age groups except for senior group, which indicates that although American give birth to boys more than girls (20.9% vs.19.4%), female senior tend to have longer life expectancy than male senior (28.4% vs.24.9%)

Overview of population by Total Income

HIDE

```
# Summary statistics of Income
mean.inc <- round(wtd.mean(clean.pop$Total.Income, weights= clean.pop$Weight,
                           normwt = FALSE, na.rm = TRUE), digits = 0)
qtl.inc <- round(wtd.quantile(clean.pop$Total.Income,weights= clean.pop$Weight,
                              normwt = FALSE, na.rm = TRUE), digits = 0)
q1.inc <- qtl.inc[2]
q3.inc <- qtl.inc[4]
sd.inc <- round(sqrt(wtd.var(clean.pop$Total.Income, weights= clean.pop$Weight,
                         normwt = FALSE, na.rm = TRUE)), digits = 0)
# Number of NA's
na.inc <- nrow(clean.pop %>% filter(is.na(Total.Income)))

# Generate the percentage of population by Age Group
inc.grp <- clean.pop %>%
  count(Income.Group, wt = Weight) %>%
  mutate(Percent = 100 * round(n/sum(n), digits = 3))

# Generate the percentage of population by Age Group and Sex
inc.grp.sex <- clean.pop %>%
  group_by(Sex) %>%
  count(Income.Group, wt = Weight) %>%
  mutate(Percent = 100*round(n/sum(n), digits = 3))

# Create in-line code
vl <- filter(inc.grp, Income.Group == 'Very Low')$Percent
up <- filter(inc.grp, Income.Group == 'Upper Middle')$Percent
hi <- filter(inc.grp, Income.Group == 'High')$Percent

# Plot the percentage of population by Income group
i1 <- inc.grp %>%
  ggplot(aes(fct_inorder(Income.Group), Percent, fill = Income.Group)) +
  geom_bar(stat = "identity") +
  labs(x = 'Income Group',title = "% of Population by Income Group") +
  theme(plot.title = element_text(hjust = 0.5, size = 12), legend.position = "none") +
  coord_flip()

i2 <- inc.grp.sex %>%
  ggplot(aes(fct_inorder(Income.Group), Percent, fill = Sex)) +
  geom_col(position = 'dodge') +
  labs(x = 'Income Group',title = "% of Population by Income Group and Sex") +
  theme(plot.title = element_text(hjust = 0.5, size = 12)) +
  coord_flip()
plot_grid(i1,i2)
```



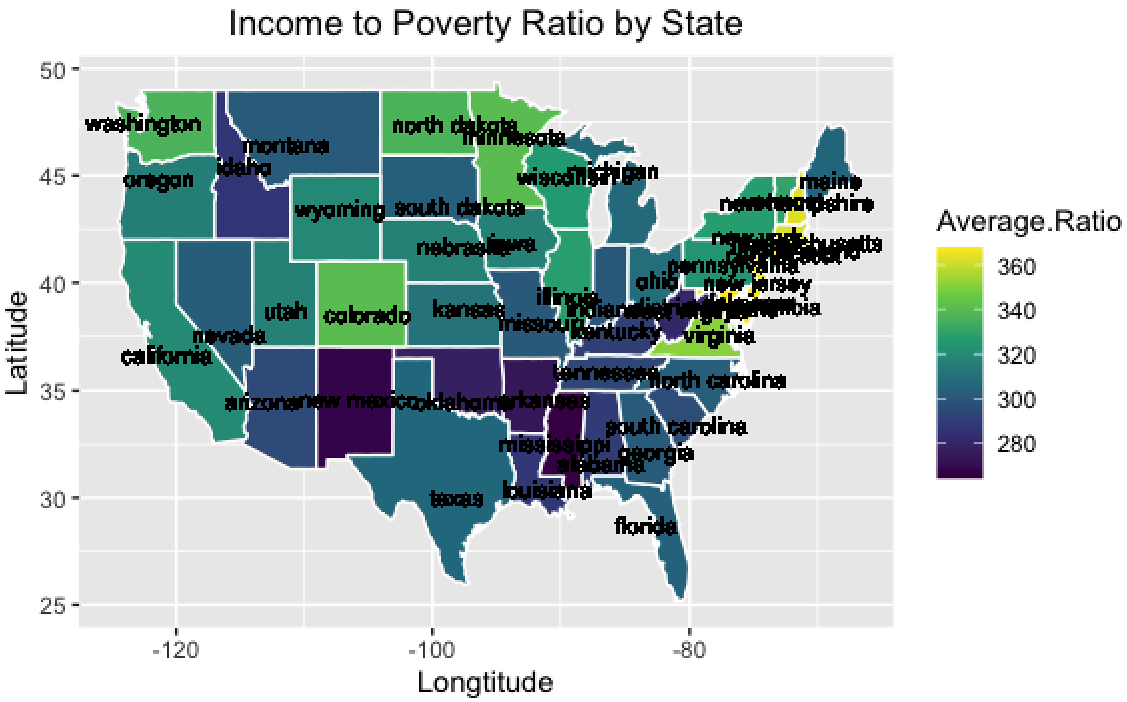
The average income of US people is $40106 a month, and 50% of the total population have an average income ranging from $7976 to $51392 a month. The average income might vary from its mean by +/- $59878.

The plots show that majority of US people have an average income at the very low level (under $20,000), which accounts for 35.8%, only 0.4% has an average income at upper middle class (from $140,000 to $149,999), while the one with high average income (from $150,000) account for 2.8%. The distribution of average income between gender tells us that men tend to earn more money than women across income groups except for the one at very low level.

Overview of population by Income to Poverty Ratio

HIDE

```r
# Summary statistics of Income to Poverty Ratio
mean.pov <- round(wtd.mean(clean.pop$Inc.Pov.Rate, weights= clean.pop$Weight,
                           normwt = FALSE, na.rm = TRUE), digits = 0)
qtl.pov <- round(wtd.quantile(clean.pop$Inc.Pov.Rate,weights= clean.pop$Weight,
                              normwt = FALSE, na.rm = TRUE), digits = 0)
q1.pov <- qtl.pov[2]
q3.pov <- qtl.pov[4]
sd.pov <- round(sqrt(wtd.var(clean.pop$Inc.Pov.Rate, weights= clean.pop$Weight,
                         normwt = FALSE, na.rm = TRUE)), digits = 0)
# Number of NA's
na.pov <- nrow(clean.pop %>% filter(is.na(Inc.Pov.Rate)))

avg.inc.pov <- clean.pop %>%
  group_by(States) %>%
  summarise(Average.Ratio = mean(Inc.Pov.Rate, na.rm = TRUE))

# Create map plot of all states in USA and change the name of `region` column into ` States`
states_map <- map_data("state") %>%
              mutate(States = region)

# Calculate average longitude and latitude
state.map <- state.unem %>%
  group_by(region) %>%
  summarise(long = mean(long), lat = mean(lat), group = group)

# Join two dataframe by state
state.unem <- left_join(states_map, avg.inc.pov, by = 'States')

# Plot Income to Poverty Ratio by State
ggplot(state.unem , aes(long, lat, group = group)) +
  geom_polygon(aes(fill = Average.Ratio), color = "white")+
  geom_text(aes(label = region), data = state.map,  size = 3, hjust = 0.5) +
  scale_fill_viridis_c(option = "D") +
  labs( x = 'Longtitude', y = 'Latitude', title = "Income to Poverty Ratio by State") +
  theme(plot.title = element_text(hjust = 0.5))
```



HIDE

```
# Create inline codes
  min.pov <- round(min(avg.inc.pov$Average.Ratio), digits = 0)
  lo.pov <- as.vector(filter(avg.inc.pov, Average.Ratio == min(avg.inc.pov$Average.Ratio))$States)
  hi.pov <- as.vector(filter(avg.inc.pov, Average.Ratio == max(avg.inc.pov$Average.Ratio))$States)
  max.pov <- round(max(avg.inc.pov$Average.Ratio), digits = 0)
```

The average income to poverty ratio of US people is 307%, and 50% of the total population have an average income to poverty ratio ranging from 160% to 501%. The average income to poverty ratio might vary from its mean by +/- 166%. The lower income to poverty ratio a person has, the poorer he or she is in comparison with others.
As can be seen from the graph, the state with the lowest average income to poverty ratio is **mississippi** with 264 %, while the state with the highest ratio is **connecticut**, with 368%.

Methodology

Dealing with missing values

This dataset consists of plenty of missing values (NA). There are a number of ways to deal with missing values, such as delete them or replace them with the mean (for continuous variables) or the mode (for categorical variables) of the variable containing missing values. However, replacing missing values with its mean/mode would reduce the variance of the data. In this research, missing values of five variables, which are Income and Inc.Pov.Rate , Education, Worker.Class and Employment , would be deleted listwise (i.e. an entire record is excluded from analysis if any single value is missing). This listwise deletion could potentially introduce bias to the model. However, we have a large dataset and the percentage of missing values for each of the aforementioned variables is not over 50%, so listwise deletion in this case might be justified and not impact tremendously on the generalizability of the resulting results. Moreover, the variables with the missing values over 10% would be not utilized in statistical analysis in this research to avoid biased results.

Dealing with outliers

Among continuous variables chosen, only `Total.Income` contains outliers, with `r p.inc.out` %. This percentage is not too much, and the dataset is big, so removal of outliers could be done with less impact on the generalizability of the resulting results.

Dealing with weights and income adjustment values

- `Weight` variable would be taken into account for any count within groups. `Weight` would be also applied to any graphs, and for any statistical calculation, functions that account for weights would be used in order to produce accurate and meaningful results.

- Because the data is collected over 5 years, to be completely accurate, Total Income would be adjusted for inflation by dividing `Inflation` (ADJINC) column by 1,000,000 to obtain the inflation adjustment factor and multiply it to the `Total.Income` value to adjust it to 2018 dollars.

Data Analysis

- The main objectives of this research are to look into unemployment status and poverty level of the US people. Therefore, a new variable called `Unemploy.rate` was created. The unemployment rate represents the number of unemployed people as a percentage of the civilian labor force so the unemployment rate is equal to the ratio between the number of 'Unemployed' and `Civil.employed` .

- The relationship between Unemployment status with other variables would be examined visually. And the average unemployment rate would be statistically compared among races, and among citizenship by conducting weighted independent t-test.

- The relationship between income to poverty ratio and race and citizenship would be identified through a weighted multiple regression model.After that, diagnostic plots would be utilized to examine the assumptions for the regression model and identify issues with the model such as trends in residuals, variance issues, outliers, etc. Finally, anova function would be used to compare the multiple regression model with a weighted linear regression model to test the hypothesis that whether including the `citizenship` variable significantly improves the fit of the simple model or not.

HIDE

```
# Identify outliers for Total Income
inc.outlier <- clean.pop %>%
  identify_outliers(Total.Income)
# Calculate percentage of total income outliers
p.inc.out <- round(length(inc.outlier$Total.Income) / length(clean.pop$Total.Income), digits = 3)*100
# Identify outliers for Poverty
pov.outlier <- clean.pop %>%
  identify_outliers(Inc.Pov.Rate) # No outliers
```

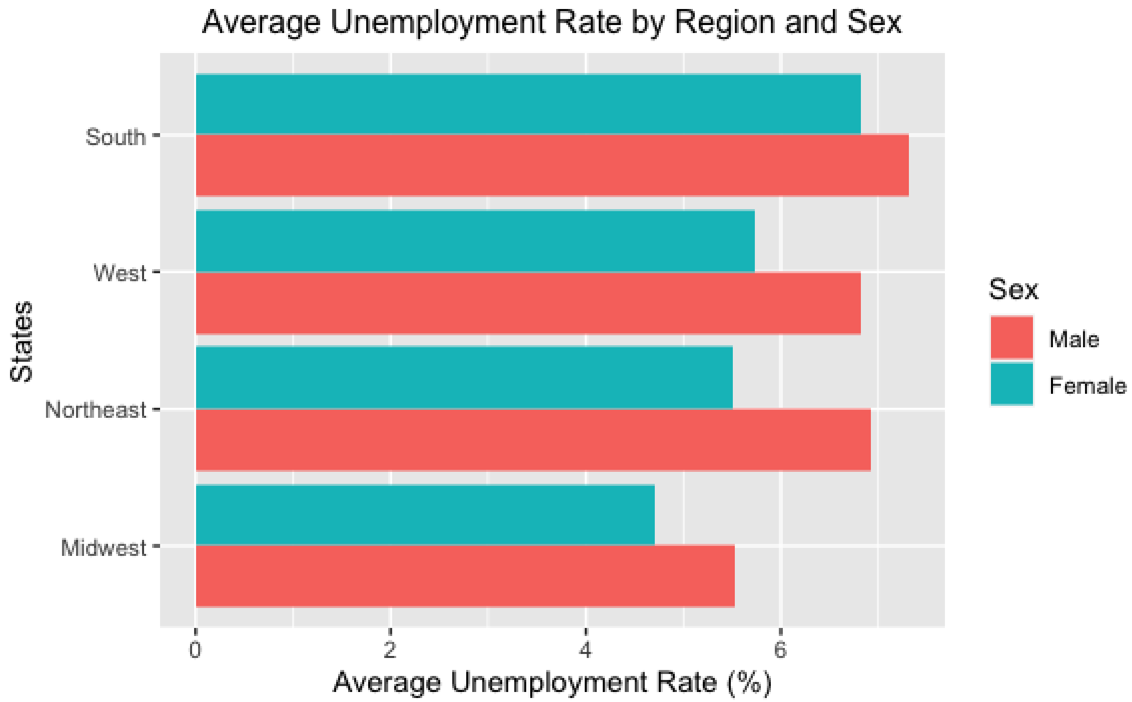
HIDE

```
# Filter out outliers of Total Income, and people under 16 years old and NA values for Income and Inc.Pov.Rate, Education, Worker.Class and Employment
filter.pop <- clean.pop %>%
  filter(Age >= 16,
         !is.na(Total.Income),
         !is.na(Inc.Pov.Rate),
         !is.na(Education),
         !is.na(Worker.Class),
         !is.na(Employment)) %>%
  anti_join(inc.outlier)
```

Analysis of Unemployment rate

HIDE

```
# Unemployment rate between States, Region and Sex
Unem.state <- filter.pop %>%
  group_by(States, Region, Sex) %>%
  count(Employment, wt = Weight) %>%
  spread(Employment, value = n) %>%
  summarise(Unemploy.rate = round(Unemployed/Civil.employed, digits = 3)*100)

# Average Unemployment rate between Region
Unem.state %>%
  group_by(Region, Sex) %>%
  summarise(avg.unem = mean(Unemploy.rate)) %>%
  ggplot(aes(reorder(Region, avg.unem), avg.unem, fill = Sex)) +
  geom_col(position = 'dodge')+
  labs(y = 'Average Unemployment Rate (%)', x = 'States',
       title = 'Average Unemployment Rate by Region and Sex')+
  theme(plot.title = element_text(hjust = 0.5, size = 12)) +
  coord_flip()
```



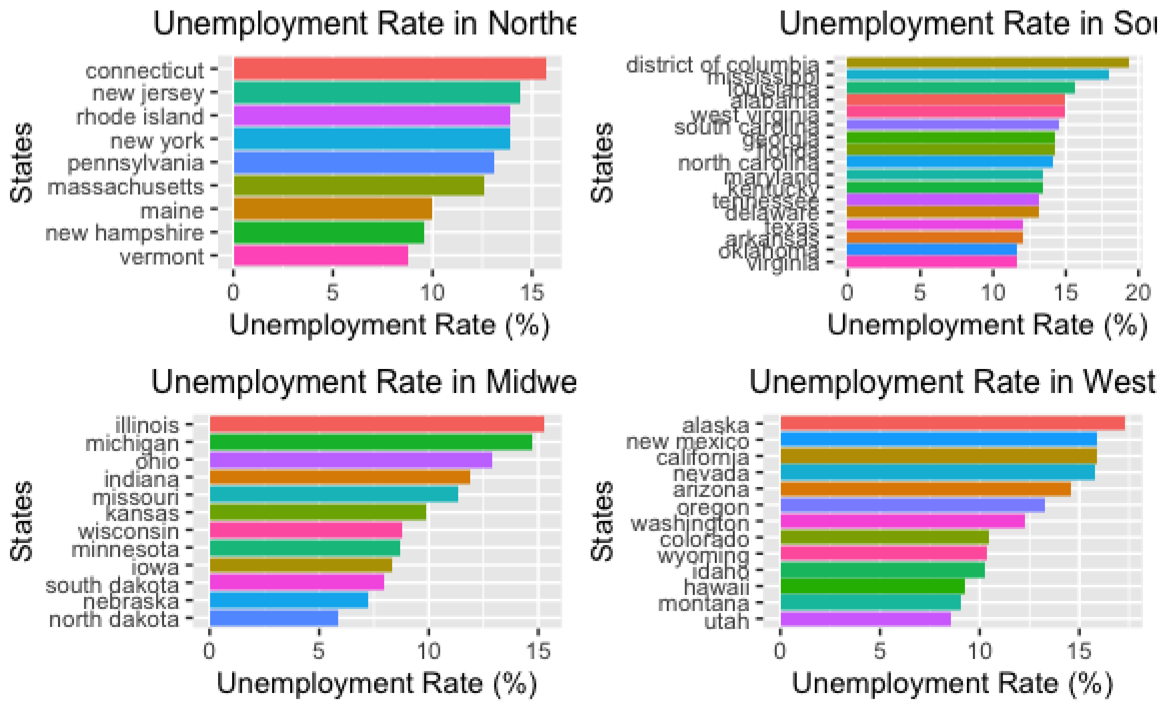
HIDE

```
# Create inline code
Unem.reg <- Unem.state %>%
  group_by(Region) %>%
  summarise(avg.unem = mean(Unemploy.rate))
hi.S <- round(max(Unem.reg$avg.unem), digits = 1)
lo.MW <- round(min(Unem.reg$avg.unem), digits = 1)
bw <- round(Unem.reg$avg.unem[Unem.reg$Region == 'Northeast'], digits = 1)
```

The bar chart shows that the unemployment rate is highest in the South with 10% and lowest in the Midwest with 2.9%. Northeast and West has the similar rate of unemployment of 6.2%. Men are more likely to face with unemployment than women across the region.

HIDE

```
# Create inline codes
## The lowest and highest % of unemployment rate in the Northeast
lo.NE <- min(filter(Unem.state, Region == 'Northeast')$Unemploy.rate)
hi.NE <- max(filter(Unem.state, Region == 'Northeast')$Unemploy.rate)
## The lowest and highest % of unemployment rate in the South
lo.S <- min(filter(Unem.state, Region == 'South')$Unemploy.rate)
hi.S <- max(filter(Unem.state, Region == 'South')$Unemploy.rate)
## The lowest and highest % of unemployment rate in the Midwest
lo.MW <- min(filter(Unem.state, Region == 'Midwest')$Unemploy.rate)
hi.MW <- max(filter(Unem.state, Region == 'Midwest')$Unemploy.rate)
## The lowest and highest % of unemployment rate in the West
lo.W <- min(filter(Unem.state, Region == 'West')$Unemploy.rate)
hi.W <- max(filter(Unem.state, Region == 'West')$Unemploy.rate)
## States' names with the lowest and highest % of unemployment rate in the USA
lo.st.name <- as.vector(filter(Unem.state,Unemploy.rate == min(Unem.state$Unemploy.rate))$States)
lo.st.unem <- min(Unem.state$Unemploy.rate) # States with lowest % of unemployment rate
hi.st.name <- as.vector(filter(Unem.state,Unemploy.rate == max(Unem.state$Unemploy.rate))$States)
hi.st.unem <- max(Unem.state$Unemploy.rate) # States with highest % of unemployment rate

# Unemployment rate between States in Northeast region
p1 <- Unem.state %>%
  filter(Region == 'Northeast') %>%
  ggplot(aes(reorder(States,Unemploy.rate), Unemploy.rate, fill = States)) +
  geom_col()+
  labs(y = 'Unemployment Rate (%)', x = 'States',
       title = 'Unemployment Rate in Northeast')+
  theme(plot.title = element_text(hjust = 0.5, size = 12), legend.position = 'none') +
  coord_flip()

# Unemployment rate between States in Northeast region
p2 <- Unem.state %>%
  filter(Region == 'South') %>%
  ggplot(aes(reorder(States,Unemploy.rate), Unemploy.rate, fill = States)) +
  geom_col()+
  labs(y = 'Unemployment Rate (%)', x = 'States',
       title = 'Unemployment Rate in South')+
  theme(plot.title = element_text(hjust = 0.5, size = 12), legend.position = 'none') +
  coord_flip()

# Unemployment rate between States in Midwest region
p3 <- Unem.state %>%
  filter(Region == 'Midwest') %>%
  ggplot(aes(reorder(States,Unemploy.rate), Unemploy.rate, fill = States)) +
  geom_col()+
  labs(y = 'Unemployment Rate (%)', x = 'States',
       title = 'Unemployment Rate in Midwest') +
  theme(plot.title = element_text(hjust = 0.5, size = 12), legend.position = 'none') +
  coord_flip()

# Unemployment rate between States in West region
p4 <- Unem.state %>%
  filter(Region == 'West') %>%
  ggplot(aes(reorder(States,Unemploy.rate), Unemploy.rate, fill = States)) +
  geom_col()+
  labs(y = 'Unemployment Rate (%)', x = 'States',
       title = 'Unemployment Rate in West')+
  theme(plot.title = element_text(hjust = 0.5, size = 12),legend.position = 'none' ) +
  coord_flip()

# Combine 4 plots
plot_grid(p1,p2,p3, p4)
```



Look at each region more specifically, we can see that:

- Northeast Region has unemployment rate ranging from 3.9% to 8.6%.

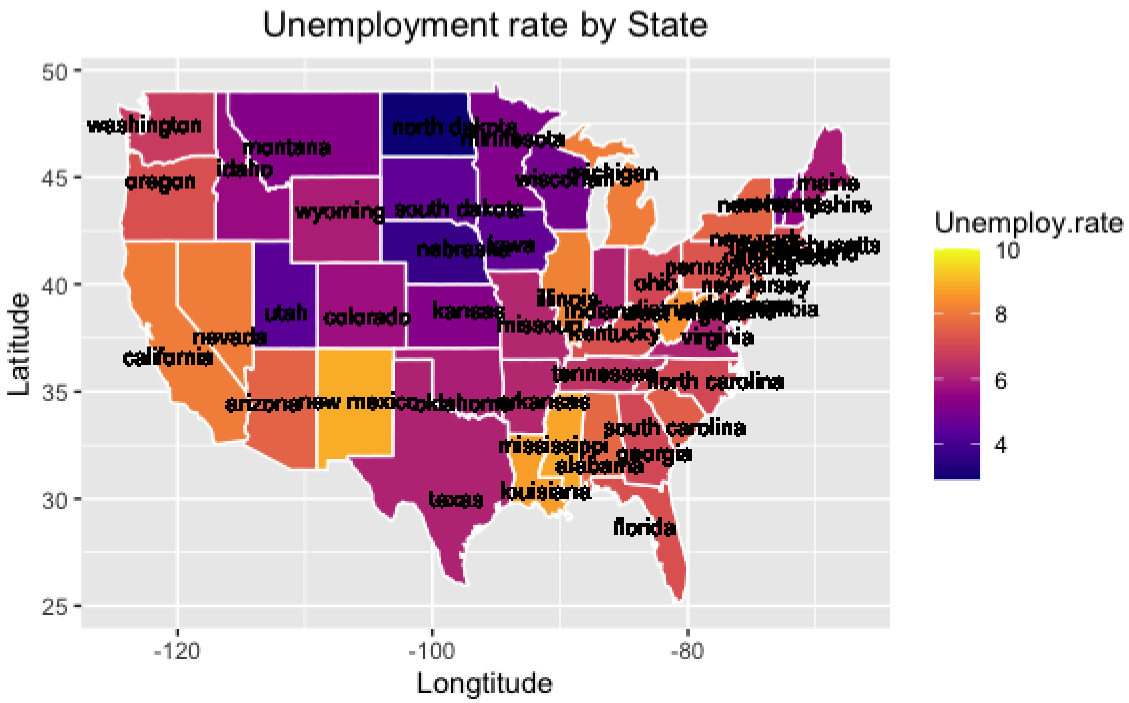
- Midwest Region has unemployment rate ranging from 2.9% to 8.2%.

- South Region has unemployment rate ranging from 5.5% to 10%.

- West Region has unemployment rate ranging from 3.9% to 10.3%.

HIDE

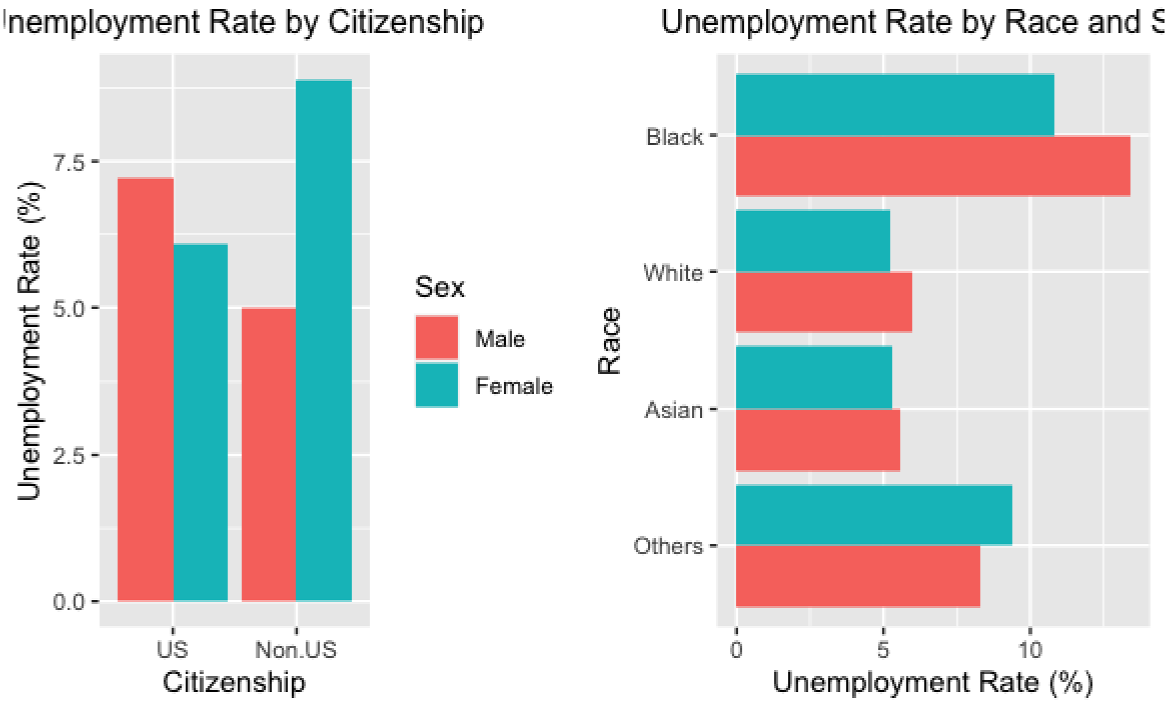
```
# Create map plot of all states in USA and change the name of `region` column into ` States`
states_map <- map_data("state") %>%
              mutate(States = region)
# Calculate average longitude and latitude
state.map <- state.unem %>%
  group_by(region) %>%
  summarise(long = mean(long), lat = mean(lat), group = group)

# Join two dataframe by state
state.unem <- left_join(states_map, Unem.state, by = 'States')

# Plot unemployment rate by State
ggplot(state.unem , aes(long, lat, group = group)) +
  geom_polygon(aes(fill = Unemploy.rate), color = "white")+
  geom_text(aes(label = region), data = state.map,  size = 3, hjust = 0.5) +
  scale_fill_viridis_c(option = "C") +
  labs( x = 'Longtitude', y = 'Latitude', title = "Unemployment rate by State") +
  theme(plot.title = element_text(hjust = 0.5))
```



The state with the lowest unemployment rate 2.9% is **north dakota**, whereas **alaska** is the one with the highest rate of unemployment of 10.3% across state.

HIDE

```
# Unemployment rate by Race
Unem.race <- filter.pop %>%
  group_by(Race) %>%
  count(Employment, wt = Weight) %>%
  spread(Employment, value = n) %>%
  summarise(Unemploy.rate = round(Unemployed/Civil.employed, digits = 3)*100)

# Unemployment rate by Race and Sex
Unem.race.sex <- filter.pop %>%
  group_by(Race, Sex) %>%
  count(Employment, wt = Weight) %>%
  spread(Employment, value = n) %>%
  summarise(Unemploy.rate = round(Unemployed/Civil.employed, digits = 3)*100)
#Unemployment rate by Race and Sex
rac.sex <- Unem.race.sex %>%
  ggplot(aes(fct_relevel(reorder(Race,Unemploy.rate), 'Others'), Unemploy.rate, fill =  Sex)) +
  geom_col(position = 'dodge') +
  labs(y = 'Unemployment Rate (%)', x = 'Race', title = 'Unemployment Rate by Race and Sex') +
  theme(plot.title = element_text(hjust = 0.5, size = 12), legend.position = 'none') +
  coord_flip()

# Unemployment rate by Citizenship
Unem.cit <- filter.pop %>%
  group_by(Citizenship) %>%
  count(Employment, wt = Weight) %>%
  spread(Employment, value = n) %>%
  summarise(Unemploy.rate = round(Unemployed/Civil.employed, digits = 3)*100)
# Unemployment rate by Citizenship and Sex
Unem.cit.sex <- filter.pop %>%
  group_by(Citizenship, Sex) %>%
  count(Employment, wt = Weight) %>%
  spread(Employment, value = n) %>%
  summarise(Unemploy.rate = round(Unemployed/Civil.employed, digits = 3)*100)
# Plot Unemployment rate by Citizenship and Sex
cit.sex <- Unem.cit.sex %>%
  ggplot(aes(reorder(Citizenship,Unemploy.rate), Unemploy.rate, fill =  Sex)) +
  geom_col(position = 'dodge')+
  labs(y = 'Unemployment Rate (%)', x = 'Citizenship', title = 'Unemployment Rate by Citizenship')+
  theme(plot.title = element_text(hjust = 0.5, size = 12))
#Combine 2 plots
plot_grid(cit.sex, rac.sex)
```



HIDE

```
# Create inline codes for citizenship
US.unem <- filter(Unem.cit, Unem.cit$Citizenship == 'US')$Unemploy.rate
non.unem <- filter(Unem.cit, Unem.cit$Citizenship == 'Non.US')$Unemploy.rate
# Create inline codes for citizenship
unem.Blk <- filter(Unem.race, Unem.race$Race == 'Black')$Unemploy.rate
unem.Wt <- filter(Unem.race, Unem.race$Race == 'White')$Unemploy.rate
unem.Asi <- filter(Unem.race, Unem.race$Race == 'Asian')$Unemploy.rate
```

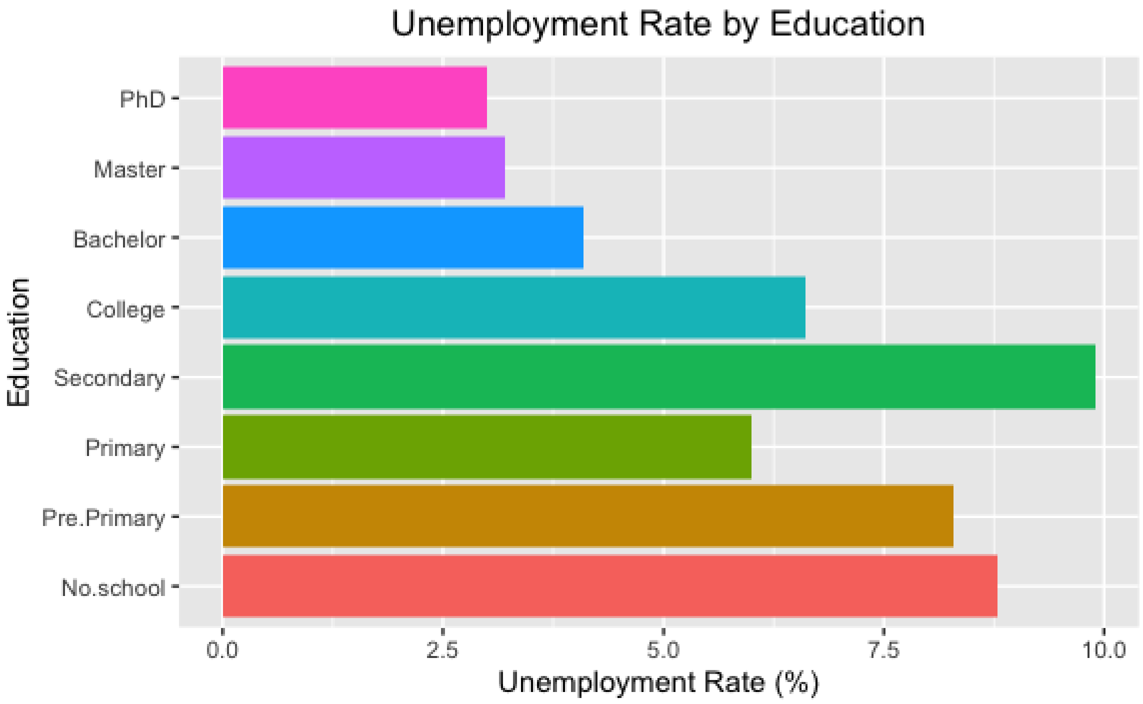
- **Citizenship**:

  The first bar chart shows that unemployment rate of US residents without US citizenship (Non US) is slightly lower than US citizen, with 6.5% vs 6.7%. In terms of gender, women without US citizenship have much higher rate of unemployment than the men, whereas the difference between this rate of male and female US citizens is in the reverse direction, that is more male are unemployed than female.
- **Race**:

  The second bar chart indicates that the Black has the highest rate of unemployment in comparison with other races, with 12%, while the White and the Asian people has a slightly difference in this rate, with 5.6% vs 5.4%. Men are more likely to face up with unemployment than women among Black, White and Asian races, while other races are in reverse direction.

HIDE

```
# Unemployment rate by Education
Unem.edu <- filter.pop %>%
  group_by(Education) %>%
  count(Employment, wt = Weight) %>%
  spread(Employment, value = n) %>%
  summarise(Unemploy.rate = round(Unemployed/Civil.employed, digits = 3)*100)

# Plot unemployment rate by Education
Unem.edu %>%
  ggplot(aes(Education, Unemploy.rate, fill =  Education)) +
  geom_col(position = 'dodge')+
  labs(y = 'Unemployment Rate (%)', x = 'Education', title = 'Unemployment Rate by Education')+
  theme(plot.title = element_text(hjust = 0.5), legend.position = 'none') +
  coord_flip()
```



HIDE

```
# Create inline codes
hi.edu <-max(Unem.edu$Unemploy.rate) # Highest % of unemployment for education attainment
lo.edu <-min(Unem.edu$Unemploy.rate) # Lowest % of unemployment for education attainment
```

As can be seen from the graph, people with secondary education attainment are the most unlikely to be employed with 9.9%, and people with PhD degree are the least with 3%. Among people with post-secondary education attainment, the higher education background, the higher chance they are able to get a job.

Compare average unemployment rate between male and female

- Calculate the variance of each group

HIDE

```
# Calculate the variance of each group
aggregate(Unemploy.rate ~ Sex, data=Unem.state, var)
```

Sex <fctr>	Unemploy.rate <dbl>
Male	2.411765
Female	2.169749

2 rows

As can be seen from the table, the variance of male was higher than female
- Test for normality of Unemployment rate distribution

HIDE

```
# Test for normality of Unemployment rate distribution
sha.test <- shapiro.test(Unem.state$Unemploy.rate)
sha.test
```

```


	Shapiro-Wilk normality test


data:  Unem.state$Unemploy.rate
W = 0.98809, p-value = 0.4999
```

HIDE

```
p.unem.s <- sha.test$p.value
```

Shapiro-Wilk test demonstrates that the distribution of unemployment rate is normal because p_value (= 0.4998835) > 0.05.
- Test for normality of Unemployment rate distribution grouped by Sex

HIDE

```
# Test for normality of Unemployment rate distribution grouped by Sex
sha.test.f <- shapiro.test(Unem.state$Unemploy.rate[Unem.state$Sex == "Female"]) # For Female
sha.test.f
```

```


    Shapiro-Wilk normality test


data:  Unem.state$Unemploy.rate[Unem.state$Sex == "Female"]
W = 0.96147, p-value = 0.09632
```

HIDE

```
p.unem.f <- sha.test.f$p.value # P-value for Female
sha.test.m <- shapiro.test(Unem.state$Unemploy.rate[Unem.state$Sex == "Male"]) # For Male
sha.test.f
```

```


    Shapiro-Wilk normality test


data:  Unem.state$Unemploy.rate[Unem.state$Sex == "Female"]
W = 0.96147, p-value = 0.09632
```

HIDE

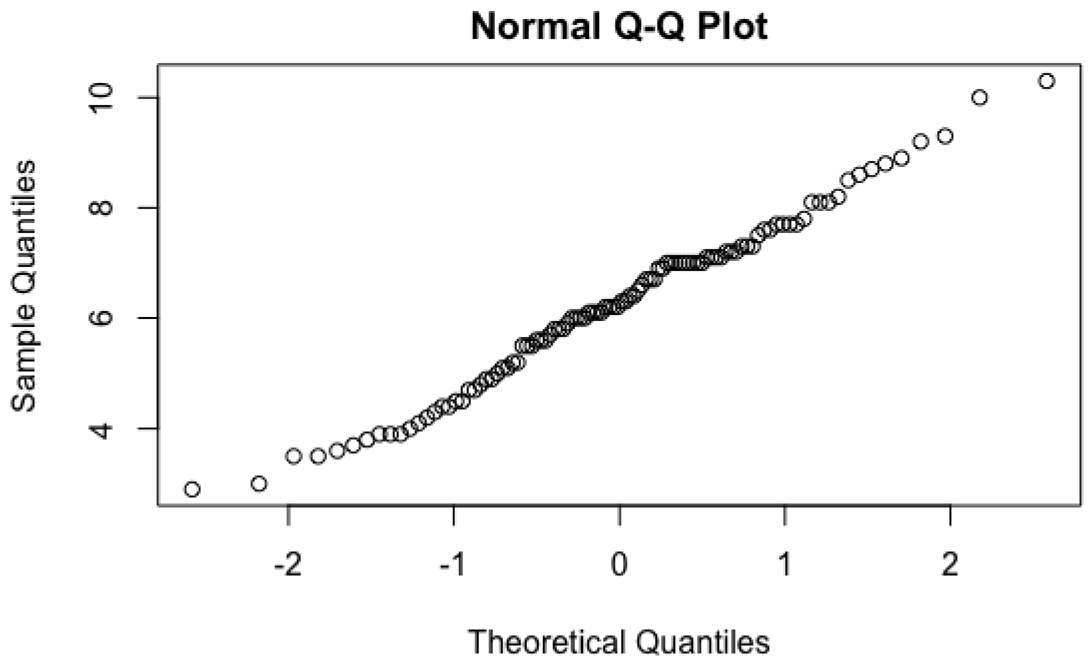
```
p.unem.m <-sha.test.m$p.value # P-value for Female
```

Shapiro-Wilk test demonstrates that the distribution of unemployment rate is normal for both female (p_value = 0.0963179) and male (p_value = 0.9713773) because their p_value > 0.05.

- Visualize normality of Unemployment rate distribution

HIDE

```
# Visualize normality of Unemployment rate distribution
qqnorm(Unem.state$Unemploy.rate)
```

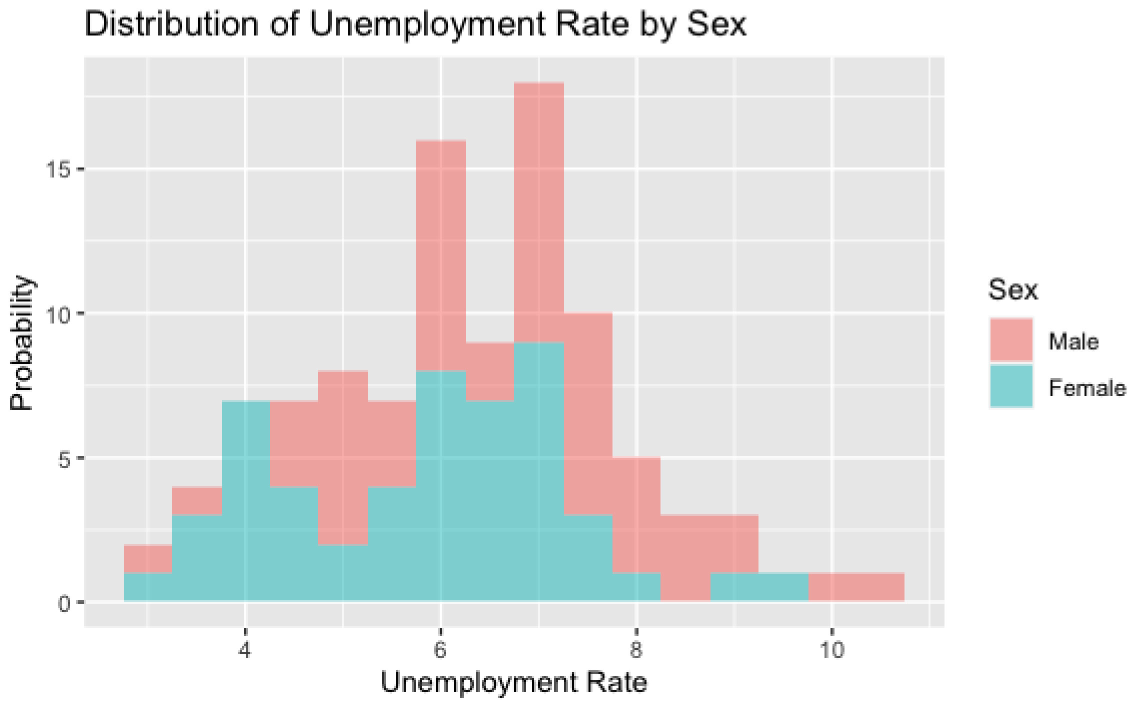


The Q-Q plot also illustrates that the distribution of unemployment rate is quite normal.

- Visualize normality of Unemployment rate distribution between group

HIDE

```
# Visualize normality of Unemployment rate distribution between group
ggplot(Unem.state, aes(x = Unemploy.rate, fill = Sex)) +
geom_histogram(binwidth=.5, alpha=1/2) +
labs(x = 'Unemployment Rate', y = 'Probability', title = 'Distribution of Unemployment Rate by Sex')
```



The graph also shows that the distribution of unemployment rate is approximately normal for both female and male.

HIDE

```
# Weighted Independent t-test
x.sex <- Unem.state$Unemploy.rate[Unem.state$Sex == "Female"]
y.sex <- Unem.state$Unemploy.rate[Unem.state$Sex == "Male"]
t.mdl <- wtd.t.test(x = x.sex, y = y.sex, weight = filter.pop$Weight,
                    weighty = filter.pop$Weight, mean1 = FALSE)
t.mdl
```

```
$test
[1] "Two Sample Weighted T-Test (Welch)"

$coefficients
  t.value        df   p.value
 -13.4011 2369.0333    0.0000

$additional
 Difference      Mean.x      Mean.y    Std. Err
-0.80209380  5.82973199  6.63182580  0.05985283
```

HIDE

```
# Create inline code
t.sex <-  round(t.mdl$coefficients[1], digits = 2)  # t statistics
f.mean <- round(t.mdl$additional[2], digits = 2)    # Average unemployment rate for female
m.mean <- round(t.mdl$additional[3], digits = 2)    # Average unemployment rate for male
se.unem <-round(t.mdl$additional[4],digits = 2)     # Standard error
df.unem <-round(t.mdl$coefficients[2], digits = 2)  # Degree of freedom
p.unem <- round(t.mdl$coefficients[3], digits = 2)  # p_value
```

The result of t-test:

Independent t-test demonstrates that the average unemployment rate is statistically different between male and female. t(2369.03) = -13.4 (p_value = 0). The dataset provides enough evidence to conclude that the average employment rate for female is significantly lower than male. (5.83% +/-0.06 vs 6.63% +/-0.06)

Compare unemployment rate between US and Non US people

HIDE

```
# Unemployment rate by States and Citizenship
unem.st.cit <- filter.pop %>%
             group_by(States, Citizenship) %>%
             count(Employment, wt = Weight) %>%
             spread(Employment, value = n) %>%
             summarise(Unemploy.rate = round(Unemployed/Civil.employed, digits = 3)*100)
```

- Calculate the variance of each group

HIDE

```
# Calculate the variance of each group
aggregate(Unemploy.rate ~ Citizenship, data = unem.st.cit, var)
```

Citizenship <fctr>	**Unemploy.rate** <dbl>
US	2.375137
Non.US	2.548871

2 rows

As can be seen from the table, the variance of `Non Us` was higher than `US` - Test normality of Unemployment rate distribution for Citizenship

HIDE

```
# Test normality of Unemployment rate distribution for Citizenship
sha.test.cit <-shapiro.test(unem.st.cit$Unemploy.rate)
sha.test.cit
```

```


	Shapiro-Wilk normality test


data:  unem.st.cit$Unemploy.rate
W = 0.98923, p-value = 0.5884
```

HIDE

```
p.unem.cit <- sha.test.cit$p.value
```

Shapiro-Wilk test demonstrates that the distribution of unemployment rate is normal because p_value (= 0.5884415) > 0.05.

- Test for normality of Unemployment rate distribution grouped by Citizenship

HIDE

```
# Test for normality of Unemployment rate distribution grouped by Citizenship
sha.test.us <- shapiro.test(unem.st.cit$Unemploy.rate[unem.st.cit$Citizenship == "US"])
sha.test.us
```

```

    Shapiro-Wilk normality test

data:  unem.st.cit$Unemploy.rate[unem.st.cit$Citizenship == "US"]
W = 0.98399, p-value = 0.7167
```

HIDE

```
p.unem.us <- sha.test.us$p.value # P-value for US
sha.test.non <- shapiro.test(unem.st.cit$Unemploy.rate[unem.st.cit$Citizenship == "Non.US"])
sha.test.non
```

```

    Shapiro-Wilk normality test

data:  unem.st.cit$Unemploy.rate[unem.st.cit$Citizenship == "Non.US"]
W = 0.9857, p-value = 0.792
```

HIDE

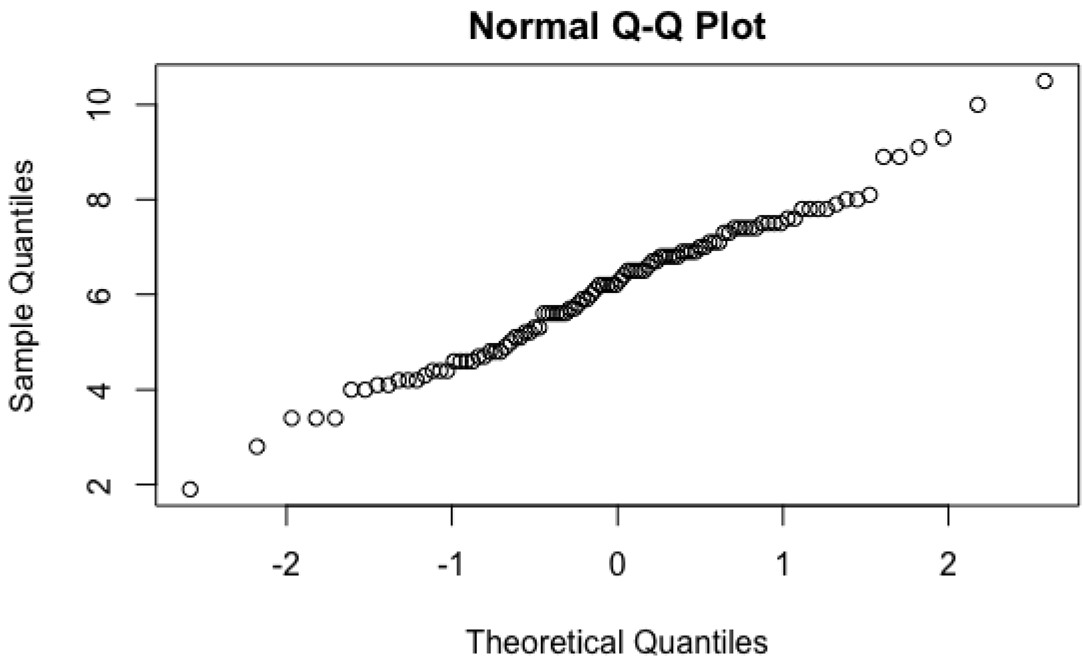
```
p.unem.non <- sha.test.non$p.value # P-value for US
```

Shapiro-Wilk test demonstrates that the distribution of unemployment rate is normal for both US citizens (p = 0.71671) and Non US citizens (p = 0.7919675) because their p_value > 0.05.

- Check normality of distribution visually using Q-Q Plot

HIDE

```
# Check normality of distribution
qqnorm(unem.st.cit$Unemploy.rate)
```

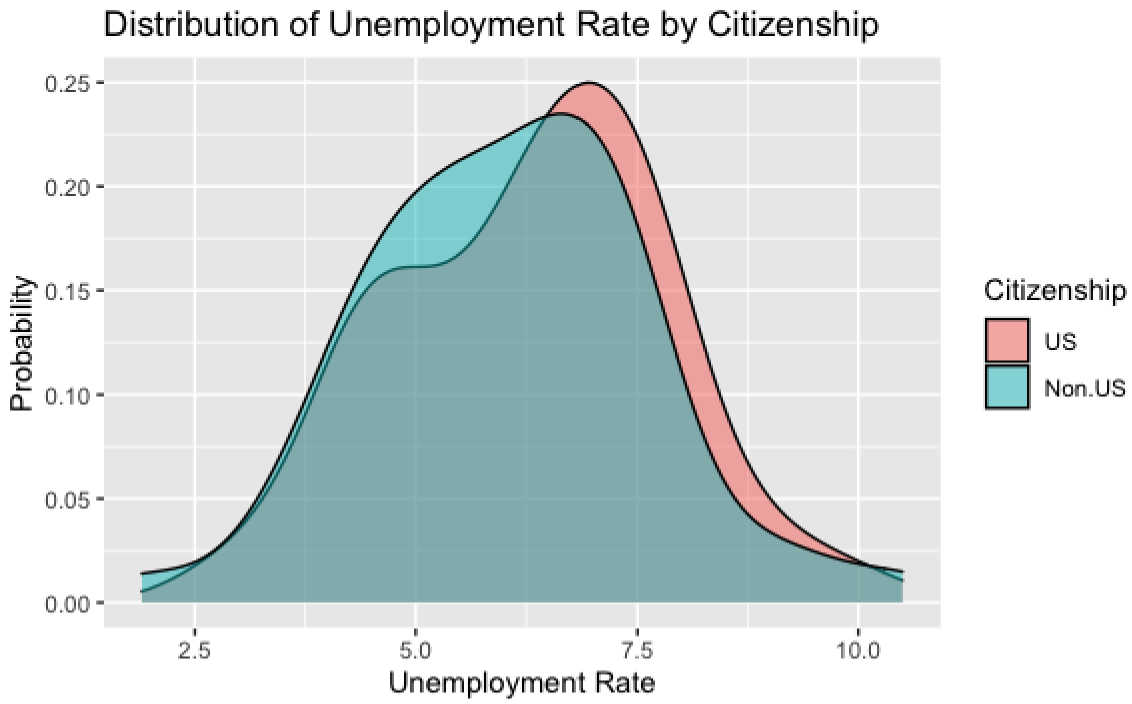


The Q-Q plot also illustrates that the distribution of unemployment rate is quite normal.

- Visualize the difference between groups:

HIDE

```
# Visualize the difference between groups
ggplot(unem.st.cit, aes(x = Unemploy.rate, fill = Citizenship)) +
  geom_density(alpha=1/2)+
  labs(x = 'Unemployment Rate', y = 'Probability',
       title = 'Distribution of Unemployment Rate by Citizenship')
```



The graph also shows that the distribution of unemployment rate is approximately normal for both US and Non US citizens.

HIDE

```
# Independent t-test
x <- unem.st.cit$Unemploy.rate[unem.st.cit$Citizenship == "US"]
y <- unem.st.cit$Unemploy.rate[unem.st.cit$Citizenship == "Non.US"]
t.cit <- wtd.t.test(x = x, y = y, weight = filter.pop$Weight, weighty = filter.pop$Weight,
                    mean1 = FALSE)
t.cit
```

```
$test
[1] "Two Sample Weighted T-Test (Welch)"

$coefficients
         t.value               df          p.value
   4.06770228122 2362.65365398421    0.00004903532

$additional
Difference     Mean.x     Mean.y   Std. Err
0.25996650 6.25644891 5.99648241 0.06390991
```

HIDE

```
# Create inline code
t.ci <- round(t.cit$coefficients[1], digits = 2)    # t statistics
us.cit <-  round(t.cit$additional[2], digits = 2)   # Average unemployment rate for US
non.cit <-  round(t.cit$additional[3], digits = 2)  # Average unemployment rate for Non US
se.cit <- round(t.cit$additional[4], digits = 2)    # Standard error
df.cit <-  round(t.cit$coefficients[2], digits = 2) # Degree of freedom
p.cit <-  round(t.cit$coefficients[3], digits = 2)  # p_value
```

The result of t-test:

Independent t-test demonstrates that the average unemployment rate is statistically different between US and Non US citizens. t(2362.65) = 4.07 (p = 0). The dataset provides enough evidence to conclude that the average employment rate for US citizens is significantly higher than Non US citizens. (6.26% +/- 0.06 vs 6% +/-0.06)

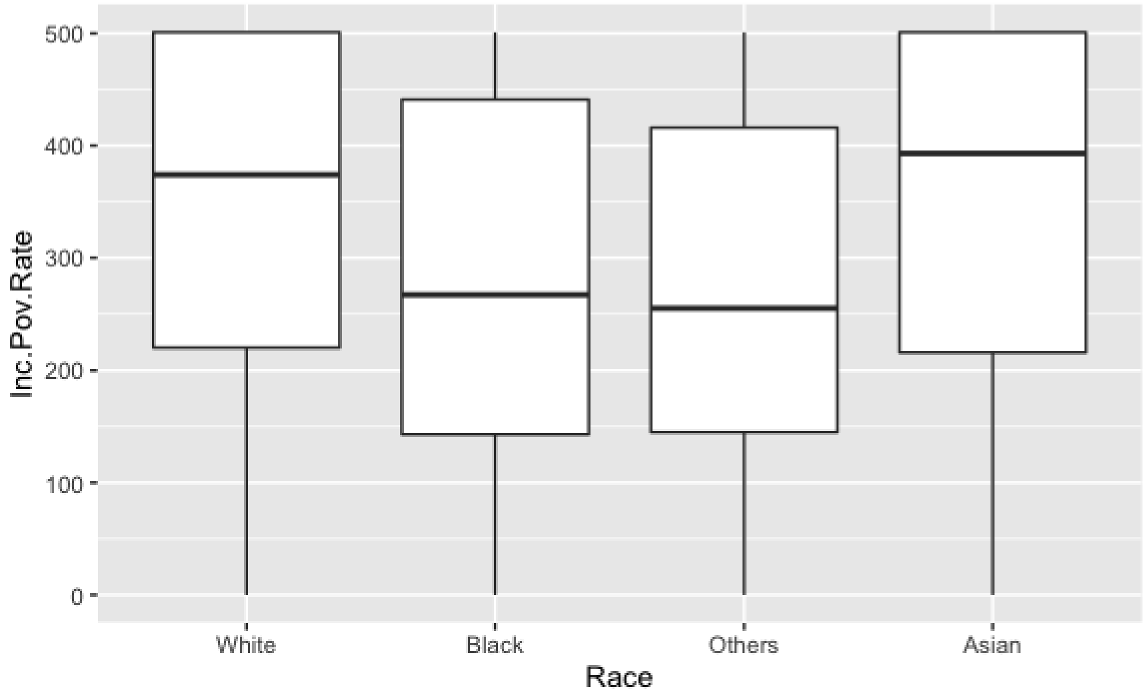
Explore the relationship between Income to Poverty Ratio and Race and Citizenship

Visualize Income to Poverty Ratio by Race

- Create boxplot for income to poverty ratio grouped by race

HIDE

```
# Create boxplot for income to poverty ratio grouped by race
qplot(Race, Inc.Pov.Rate, geom = "boxplot", data = filter.pop)
```

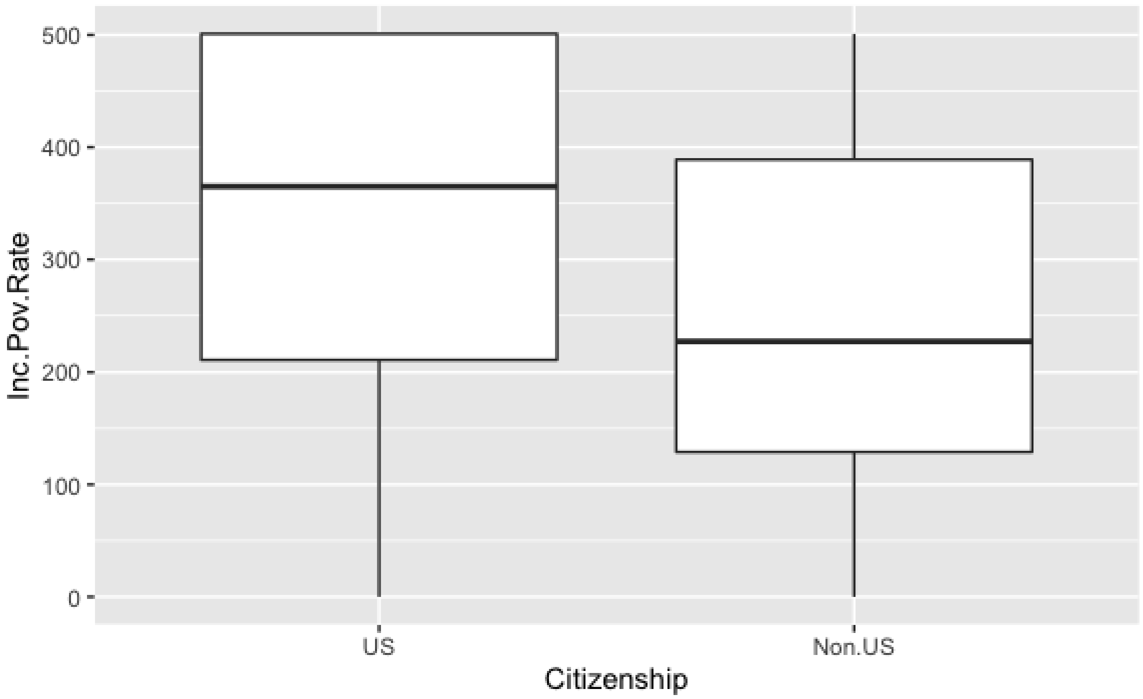


Looking at the plot above, we can see that the average income to poverty ratio appears to be different between White or Asian and Black or Other races because their medians are not overlapped, while there is no difference in the ratio between White and Asian or between Black and Other races because their medians seem to be equivalent.

- Create boxplot for income to poverty ratio grouped by Citizenship

HIDE

```
# Create boxplot for income to poverty ratio grouped by Citizenship
qplot(Citizenship, Inc.Pov.Rate, geom = "boxplot", data = filter.pop)
```



The boxplot above shows that the average income to poverty ratio appears to be different between US and Non US people because their medians are not overlapped.

Perform multiple regression to confirm whether those differences are statistically significant

- Fit the complex model with race and citizenship as the predictor variables
  Summary of regression model output:

HIDE

```
# Multiple regression model
pov.lm <- lm(Inc.Pov.Rate ~ Race + Citizenship, data = filter.pop, weights = Weight)
#Summary of regression model
summary(pov.lm)
```

```
Call:
lm(formula = Inc.Pov.Rate ~ Race + Citizenship, data = filter.pop,
    weights = Weight)

Weighted Residuals:
    Min      1Q  Median      3Q     Max
-6190.0  -458.5    98.6   572.2  5482.8

Coefficients:
                  Estimate Std. Error t value            Pr(>|t|)
(Intercept)       342.3082     0.0633  5408.0 <0.0000000000000002 ***
RaceBlack         -61.1976     0.1641  -372.9 <0.0000000000000002 ***
RaceOthers        -52.5111     0.1982  -265.0 <0.0000000000000002 ***
RaceAsian          26.9602     0.2446   110.2 <0.0000000000000002 ***
CitizenshipNon.US -81.2930     0.1994  -407.7 <0.0000000000000002 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 705.3 on 8323509 degrees of freedom
Multiple R-squared:  0.0466,    Adjusted R-squared:  0.0466
F-statistic: 1.017e+05 on 4 and 8323509 DF,  p-value: < 0.00000000000000022
```

HIDE

```
ti.lm <- tidy(pov.lm)
ti.lm
```

term <chr>	estimate <dbl>	std.error <dbl>	statistic <dbl>	p.value <dbl>
(Intercept)	342.30821	0.06329711	5407.9594	0
RaceBlack	-61.19760	0.16409606	-372.9376	0
RaceOthers	-52.51110	0.19818031	-264.9663	0
RaceAsian	26.96025	0.24458356	110.2292	0
CitizenshipNon.US	-81.29296	0.19937859	-407.7317	0

5 rows

HIDE

```r
gla.lm <- glance(pov.lm)
gla.lm
```

r.squared	adj.r.squared	sigma	statistic	p.value	df	logLik	AIC	BIC	devian
<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<db
0.04660412	0.04660366	705.2691	101717.9	0	4	-54832723	109665457	109665541	41401507048

1 row | 1-10 of 12 columns

HIDE

```r
# Create inline codes:
p.intercept <- ti.lm$p.value[1] # p value of White people
p.black <- ti.lm$p.value[2] # p value of Black people
p.others<- ti.lm$p.value[3] # p value of Other people
p.asian <- ti.lm$p.value[4] # p value of Asian people
p.non.us <- ti.lm$p.value[5] # p value of Non US people
p.value <- gla.lm$p.value # p value of the model
df1 <- floor(gla.lm$df) # degree of freedom between groups
F_stat <- floor(gla.lm$statistic) # F Statistics
df2 <- floor(gla.lm$df.residual) # degree of freedom within groups
R.squa.per <- 100*round(gla.lm$adj.r.squared, digits = 3) # Adjusted R squared (%)
R.squared <- round(gla.lm$adj.r.squared, digits = 3) # Adjusted R squared
sigma <- round(gla.lm$sigma,1)
est.black <-  round(ti.lm$estimate[2], digits = 1) # Difference between White and Black
est.others <- round(ti.lm$estimate[3], digits = 1) # Difference between White and Others
est.asian <-  round(ti.lm$estimate[4], digits = 1) # Difference between White and Asian
est.non.us <- round(ti.lm$estimate[5], digits = 1) # Difference between US and Non US
```

The statistically significant regression model was found, F(4, 8323509) = 101717, p = 0. Therefore, we can reject the null hypothesis and conclude that there is a relationship between the income to poverty ration and race and citizenship. Coefficients of intercept (p= 0), `black` (p = 0), `others` (p = 0), `asian` (p = 0), and `Non Us` (p = 0) were also statistically significant. 4.7% of the variance in the income to poverty ratio can be explained by the regression model, R2 = 0.047.

Among US residents of the same citizenship, the income to poverty ratio of Black US residents and other races are -61.2% and -52.5% respectively lower than the ratio of white US residents on average, whereas the income to poverty ratio of Asian US residents is 27% higher than the ratio of White US residents on average. In other words, the White people tend to be wealthier than the Black and the people of the other race but less wealthy than the Asian people on average.

Among US residents of the same race, the income to poverty ratio of Non US residents is -81.3% +/- lower than that of the US citizens on average, that is, Us residents without citizenship incline to be poorer than US citizens.

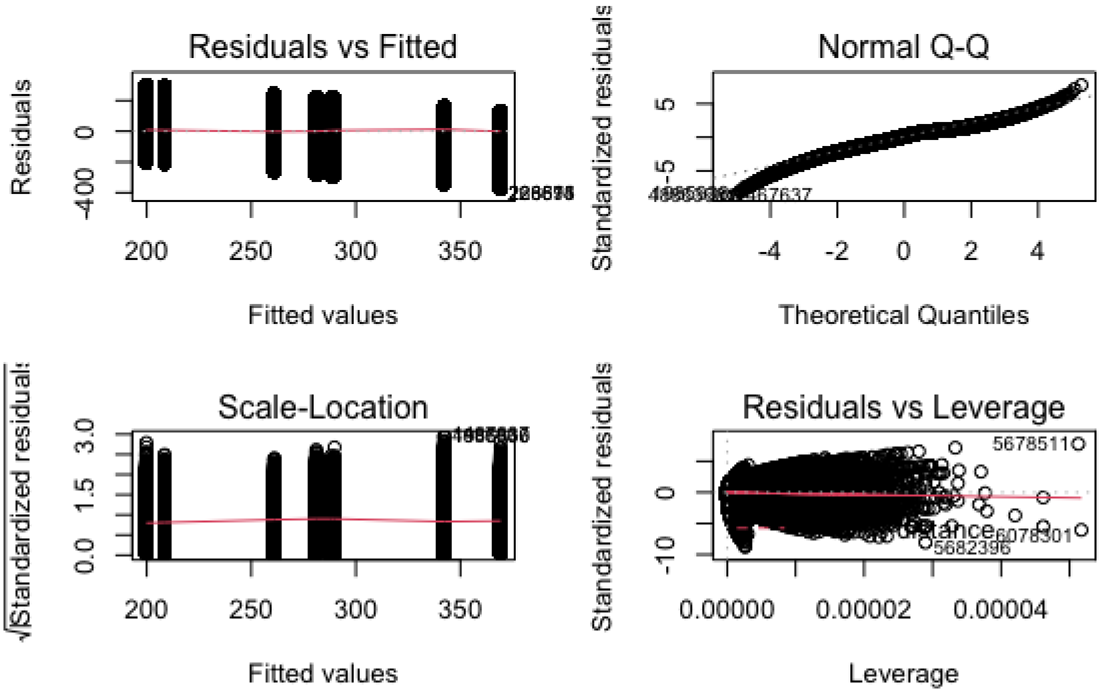
Reliability of the model:

Just a small amount of the variance (4.7%) in the income to poverty ratio can be explained by the regression model, which means its reliability is in doubt. Moreover, the residual standard error (= 705.3) is much higher than the estimate of all the parameters which indicates that the model is not reliable and race and citizenship might be not good predictors for predicting the income to poverty ratio.

Using diagnostic Plots to examine the assumptions for the regression model

HIDE

```
# Using diagnostic Plots to examine the assumptions for the regression model
par(mfrow = c(2,2))
plot(lm(Inc.Pov.Rate ~ Race + Citizenship, data = filter.pop, weights = Weight))
```



- ‘Residuals vs Fitted’ plot shows us a horizontal red line, which indicates that the expected value of the errors is 0 and the pattern of residuals is linear.
- QQ plot shows that the residuals are normally distributed because they line up quite well on the straight dashed line (although the lower tail slightly deviates from the straight line, it’s acceptable).
- ’Scale-Location` plot provides us a horizontal line with equally spread points, which indicates that the errors have the same variance.
- ‘Residuals vs Leverage’ plot show us that there are no cases outside of Cook’s distance, which indicates that there are no observations with high influence on the regression result and none of the points appear to be outliers.

As a result, all the assumptions are met.

Hypothesis testing: Compare the simple regression to the more complex model (original model)

- Fit the simpler model with race as the only predictor variable. Summary of simple regression model output:

HIDE

```
# Fit the simpler model with race as the only predictor variable
pov.lm.simple <- lm(Inc.Pov.Rate ~ Race, data = filter.pop, weights = Weight)
summary(pov.lm.simple)
```

```


Call:
lm(formula = Inc.Pov.Rate ~ Race, data = filter.pop, weights = Weight)

Weighted Residuals:
    Min      1Q  Median      3Q     Max 
-6108.1  -463.4   103.4   588.5  4036.4 

Coefficients:
             Estimate Std. Error t value            Pr(>|t|)    
(Intercept) 337.77826    0.06293  5367.2 <0.0000000000000002 ***
RaceBlack   -60.89242    0.16572  -367.4 <0.0000000000000002 ***
RaceOthers  -68.28045    0.19630  -347.8 <0.0000000000000002 ***
RaceAsian     6.92128    0.24198    28.6 <0.0000000000000002 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 712.3 on 8323510 degrees of freedom
Multiple R-squared:  0.02756,	Adjusted R-squared:  0.02756 
F-statistic: 7.864e+04 on 3 and 8323510 DF,  p-value: < 0.00000000000000022
```

- Compare the simple regression to the more complex model (original model)

HIDE

```
# Compare the simple regression to the more complex model (original model)
ano <- anova(pov.lm.simple, pov.lm)
# Output the result
ano
```

```
Analysis of Variance Table

Model 1: Inc.Pov.Rate ~ Race
Model 2: Inc.Pov.Rate ~ Race + Citizenship
   Res.Df           RSS Df   Sum of Sq      F                Pr(>F)    
1 8323510 4222841763663                                                
2 8323509 4140150704845  1 82691058818 166245 < 0.00000000000000022 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

HIDE

```
p.ano <- ano$`Pr(>F)`[2] # Create inline code for p value
```

The `citizenship` variable is statistically significant because p-value = 0 (< 0.05). Therefore, we have enough evidence to conclude that including the `citizenship` variable significantly improves the fit of the simple model. In other words, it is unlikely that the improvement in fit when adding the `citizenship` variable is simply due to random fluctuations in the data.

Discussion

Summary of findings:

Unemployment rate:
Unemployment rate is highest in the South and lowest in the Midwest. Northeast and West had the similar rate of unemployment. Men were more likely to face with unemployment than women across the region. North Dakota is the state with the lowest unemployment rate, whereas Alaska is the one with the highest rate of unemployment.
- In terms of Citizenship, unemployment rate of Non-US citizens is slightly lower than US citizen. Women without US citizenship have much higher rate of unemployment than the men, whereas the difference between this rate of male and female US citizens is in the reverse direction, that is, more male are unemployed than female.
- In terms of Race, the Black people has the highest rate of unemployment in comparison with other races, while the White and the Asian people has a slightly difference in this rate. Men are more likely to face up with unemployment than women among Black, White and Asian races, while other races are in reverse direction.
- In terms of People with education attainment, secondary education attainment are the most unlikely to be employed, while people with PhD degree are the least . Among people with post-secondary education attainment, the higher education background, the higher chance they are able to get a job.

The average employment rate for female is significantly lower than male. The average employment rate for US citizens is significantly higher than Non-US citizens.

Income to poverty ratio vs. race and citizenship:

There is a relationship between the income to poverty ratio and race and citizenship. In particular, among US residents of the same citizenship, the income to poverty ration of the Black US residents and other races are significantly lower than the ratio of white US residents on average, whereas the income to poverty ration of the Asian US residents is significantly higher than the ratio of the White US residents on average. In other words, the White people tend to be wealthier than the Black and the people of the other race but less wealthy than the Asian people on average.

Among US residents of the same race, the income to poverty ratio of Non US residents is significantly lower than that of the US citizens on average, that is, Us residents without citizenship incline to be poorer than US citizens.

Reliability of the model:

Just a small amount of the variance (4.7%) in the income to poverty ratio can be explained by the regression model, which means its reliability is in doubt. Moreover, the residual standard error (= 705.3) is much higher than the estimate of all the parameters which indicates that the model is not reliable and race and citizenship might be not good predictors for predicting the income to poverty ratio.

Limitation of this analysis and findings:

In this research, I have not run a model that takes into account all the variables to check whether those variables improve the fit of the model or not. It's better to run a stepwise or backward model to get the optimal model with. Cross-validation should also be applied to confirm the reliability and generalizability of the model. Besides, the listwise deletion of the missing values might impact the results of the analysis so it's recommended to compare the model without missing values with the model with the missing values or with the model with missing values replaced with the mean or mode of each variable containing the missing value. Any model that leads to big discrepancy should not be applied for the analysis.

Confounding variables:

There are some potential confounding variables that may affect the relationship between income to poverty ratio and race and citizenship that I did not control for, such as gender, employment status, marital status, education attainment, family structure, age, etc. In addition, as I mentioned above about the reliability of the model, the model fit should be justified. Therefore, it is not reliable enough to present the findings of this analysis to the policy makers, or present these findings to them with a caution.

Appendix: All codes

HIDE

```r
# Appendix: All codes
#Read in the csv files
#Create a vector consisting of selected variables
variables <- c("SERIALNO", "REGION", "ST", "ADJINC",  "PWGTP", "AGEP", "CIT",
               "COW", "MAR", "SCHL","SEX", "ESR", "PINCP", "POVPIP", "RAC1P")
#Read in 'psam_pusa.csv'
a <- fread("psam_pusa.csv", header = TRUE, select = variables, data.table = FALSE,
           stringsAsFactors = FALSE)
#Read in 'psam_pusb.csv'
b <- fread("psam_pusb.csv", header = TRUE, select = variables, data.table = FALSE,
           stringsAsFactors = FALSE)
#Read in 'psam_pusc.csv'
c <- fread("psam_pusc.csv", header = TRUE, select = variables, data.table = FALSE,
           stringsAsFactors = FALSE)
#Read in 'psam_pusd.csv'
d <- fread("psam_pusd.csv", header = TRUE, select = variables, data.table = FALSE,
           stringsAsFactors = FALSE)
#Combine 4 files
df <- rbind(a, b, c, d)
#Save combined file
fwrite(df,'population.csv')
#Remove unnecessary files
rm(a, b, c, d, df)


# Read in the file used for analysis
population <- fread("population.csv", header = TRUE, data.table = FALSE, stringsAsFactors = FALSE)
#Change the columns' names
colnames(population) <- c('Serial.No', 'Region', 'States', 'Inflation', 'Weight', 'Age', 'Citizenship', 'Worker.Class', 'Marital.Status','Education', 'Sex','Employment','Total.Income', 'Inc.Pov.Rate', 'Race')
#Create levels and labels for `Region`
region.levels <- c(1,2,3,4,9)
region.labels <- c('Northeast', 'Midwest', 'South', 'West', 'Puerto Rico')
#Create levels and labels for `States`
state.levels <- c(1, 2, 4, 5, 6, 8, 9, 10, 11, 12, 13, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25,
                  26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 44, 45, 46,
                  47, 48, 49, 50, 51, 53, 54, 55, 56, 72)
state.labels <- c('alabama', 'alaska', 'arizona', 'arkansas', 'california', 'colorado', 'connecticut', 'delaware', 'district of columbia', 'florida', 'georgia', 'hawaii', 'idaho', 'illinois', 'indiana', 'iowa', 'kansas', 'kentucky', 'louisiana', 'maine', 'maryland',  'massachusetts', 'michigan', 'minnesota', 'mississippi', 'missouri', 'montana', 'nebraska', 'nevada', 'new hampshire', 'new jersey', 'new mexico', 'new york', 'north carolina', 'north dakota', 'ohio', 'oklahoma', 'oregon', 'pennsylvania', 'rhode island', 'south carolina', 'south dakota', 'tennessee', 'texas', 'utah', 'vermont', 'virginia', 'washington', 'west virginia', 'wisconsin', 'wyoming', 'puerto rico')
# Create citizenship levels and labels
```

```
cit.levels <- c(1,2,3,4,5)
cit.labels <- c( 'US Born', 'PR.Guam.VI.NM Born', 'Abroad Born', 'Naturalized Citizen', 'Non US')
# Create class of workers' levels and labels
worker.levels <- c(1,2,3,4,5,6,7,8,9)
worker.labels <-c('Private, for Profit', 'Private, Non-Profit', 'Local','State','Federal',
                  'Self-employed, Not Incor','Self-employed, Incor',
                  'Family work, No Payment','Unemployed')
#Create levels and labels for Marital Status
MS.levels <- c(1,2,3,4,5)
MS.labels <- c('Married','Widowed','Divorced','Separated','Never married/<15')
# Create levels and labels for Education
edu.levels <- c(1,2,3,4,5,6,7,8,9,10, 11,12,13,14,15,16,17,18,19,20,21,22,23,24)
edu.labels <- c( 'No schooling', 'Preschool', 'Kindergarten', 'Grade 1', 'Grade 2', 'Grade 3',
                 'Grade 4', 'Grade 5', 'Grade 6', 'Grade 7', 'Grade 8', 'Grade 9', 'Grade 10',
                 'Grade 11', 'Grade 12, No Diploma','HS Diploma', 'GED/alternative credential',
                 'College, 1y less','College, 1y over',  'Associate', 'Bachelor','Master',
                 'Professional, over bachelor','Doctorate')
# Create levels and labels for 'Employment status'
emp.levels <- c(1,2,3,4,5,6)
emp.labels <- c( 'Civilian employed, at work', 'Civilian employed, not at work','Unemployed',
                 'Armed forces, at work','Armed forces, not at work', 'Not in labor force')
# Create levels and labels for 'Race'
race.levels <- c(1,2,3,4,5,6,7,8,9)
race.labels <- c('White', 'Black/African American', 'American Indian', 'Alaska Native',
                 'American Indian and/or Alaska Native', 'Asian',
                 'Native Hawaiian & Other Pacific Islander', 'Some Other Race',
                 'Two/More Races')
# Create factor variables and factor level names
factor.pop <- population %>%
  mutate(Region = factor(Region, levels = region.levels , labels = region.labels),
         States = factor(States, levels = state.levels, labels = state.labels),
         Citizenship = factor(Citizenship, levels = cit.levels, labels = cit.labels),
         Worker.Class = factor(Worker.Class, levels = worker.levels, labels = worker.labels),
         Marital.Status= factor(Marital.Status, levels = MS.levels, labels = MS.labels),
         Education = factor(Education, levels = edu.levels, labels = edu.labels),
         Sex = factor(Sex, levels = c(1,2), labels = c('Male', 'Female')),
         Employment = factor(Employment, levels = emp.levels, labels = emp.labels),
         Race = factor(Race, levels = race.levels, labels = race.labels))
# Save factorized data into a .rds file
write_rds(factor.pop, "population.rds")

# Load the data
factor.pop <-read_rds("population.rds")
# Adjust income value with inflation rate
```

```
adjust.pop <- factor.pop %>%
  mutate(Total.Income = (Inflation / 1000000) * Total.Income)
# Collapse some categorical variables into more meaningful groups
clean.pop <- adjust.pop %>%
  mutate(Citizenship = fct_collapse(Citizenship,
                                    US = c('US Born', 'PR.Guam.VI.NM Born', 'Abroad Born', 'Naturalize
         d Citizen'),
                                    Non.US = 'Non US'),
         Worker.Class = fct_collapse(Worker.Class,
                                     Private = c('Private, for Profit', 'Private, Non-Profit'),
                                     Government = c('Local', 'State', 'Federal'),
                                     Self.Employed = c('Self-employed, Not Incor','Self-employed, Inco
         r',
                                                       'Family work, No Payment'),
                                     Unemployed = 'Unemployed'),
         Education = fct_collapse(Education,
                                  No.school = 'No schooling',
                                  Pre.Primary = c('Preschool', 'Kindergarten'),
                                  Primary = c('Grade 1', 'Grade 2', 'Grade 3', 'Grade 4', 'Grade 5',
         'Grade 6'),
                                  Secondary = c('Grade 7', 'Grade 8', 'Grade 9', 'Grade 10', 'Grade 1
         1',
                                                'Grade 12, No Diploma','HS Diploma', 'GED/alternative
          credential'),
                                  College = c('College, 1y less', 'College, 1y over'),
                                  Bachelor = c('Associate', 'Bachelor'),
                                  Master = c('Master', 'Professional, over bachelor'),
                                  PhD = 'Doctorate'),
         Employment = fct_collapse(Employment,
                                   Civil.employed = c('Civilian employed, at work', 'Civilian employe
         d, not at work'),
                                   Unemployed = 'Unemployed',
                                   Arm.force = c('Armed forces, at work','Armed forces, not at work'),
                                   Not.Labor.Force = 'Not in labor force'),
         Race = fct_collapse(Race,
                             White = 'White',
                             Black = 'Black/African American',
                             Asian = 'Asian',
                             Others = c( 'American Indian', 'Alaska Native', 'American Indian and/or A
         laska Native',
                                         'Native Hawaiian & Other Pacific Islander', 'Some Other Race'
         ,'Two/More Races')),
         Age.Group = case_when(Age <= 15 ~ 'Children',
                               Age >=16 & Age <=23 ~ "Late Gen Z",
                               Age >= 24 & Age <= 39 ~ 'Millennial',
                               Age >=40 & Age <= 55 ~ 'Baby Boomer',
                               Age > 55 ~ 'Senior'),
         Income.Group = case_when (Total.Income < 20000 ~ 'Very Low',
                                   Total.Income >= 20000 & Total.Income < 44999 ~ 'Low',
                                   Total.Income >= 45000 & Total.Income < 139999 ~ 'Middle',
                                   Total.Income >= 140000 & Total.Income < 149999 ~ 'Upper Middle',
                                   Total.Income >= 150000 ~ 'High')
  )

# Create levels for Age Group and Income Group
age.levels <- c('Children','Late Gen Z', 'Millennial', 'Baby Boomer','Senior')
inc.levels <- c('Very Low', 'Low', 'Middle', 'Upper Middle','High')
clean.pop <- clean.pop %>%
  mutate(Age.Group = factor(Age.Group, levels = age.levels),
         Income.Group = factor(Income.Group, levels = inc.levels))
# Generate proportion of the population by States
state.tbl <- clean.pop %>%
  count(States, wt = Weight) %>%
  mutate(region = States, # Create 'region' column to match with map data
         Percent = 100*round(n/sum(n), digits = 3)) %>%
  arrange(desc(n))

# Proportion of the population under 1%
s1 <- state.tbl %>%
  filter(Percent < 0.5) %>%
  ggplot(aes(x= reorder(States, n), y = Percent, fill = States)) +
  geom_col() +
  labs(x = 'States',title = "% of Population under 0.5%") +
  theme(plot.title = element_text(hjust = 0.5, size = 12), legend.position = 'none') +
  coord_flip()

s2 <- state.tbl %>%
  filter(Percent >= 0.5, Percent < 1) %>%
  ggplot(aes(x= reorder(States, n), y = Percent, fill = States)) +
  geom_col() +
  labs(x = 'States',title = " % of Population from 0.5% to under 1%") +
  theme(plot.title = element_text(hjust = 0.5, size = 10), legend.position = 'none') +
  coord_flip()
# Proportion of the population from 1% to 2%
s3 <- state.tbl %>%
  filter(Percent >= 1, Percent <= 2) %>%
  ggplot(aes(x= reorder(States, n), y = Percent, fill = States)) +
  geom_bar(stat = "identity") +
  labs(x = 'States',title = "% of Population from 1% to 2 %") +
  theme(plot.title = element_text(hjust = 0.5, size = 12), legend.position = 'none') +
  coord_flip()
```

```
# Proportion of the population over 2%
s4 <- state.tbl %>%
  filter(Percent > 2) %>%
  ggplot(aes(x= reorder(States, n), y = Percent, fill = States)) +
  geom_bar(stat = "identity") +
  labs(x = 'States',title = "% of Population over 2 %") +
  theme(plot.title = element_text(hjust = 0.5, size = 12), legend.position = 'none') +
  coord_flip()

# Combine 3 plots
plot_grid(s1, s2, s3, s4)

# Create in-line code
n1 <- nrow(state.tbl%>%filter(Percent < 0.5))
n2 <- nrow(state.tbl%>%filter(Percent >= 0.5, Percent < 1))
n3 <- nrow(state.tbl%>%filter(Percent >= 1, Percent <= 2))
n4 <- nrow(state.tbl%>%filter(Percent >2))
min.st <- min(state.tbl$Percent)
lo.st <- as.vector(filter(state.tbl,Percent == min(state.tbl$Percent))$States)
avg.low <- floor(mean(filter(state.tbl,Percent == min(state.tbl$Percent))$n))
hi.st <- as.vector(filter(state.tbl,Percent == max(state.tbl$Percent))$States)
max.st <- max(state.tbl$Percent)
p <- filter(state.tbl,Percent == max(state.tbl$Percent))$n
options(scipen = 999)

# Create map plot for states
states_map <- map_data("state")
# Calculate average longitude and latitude
state.map <- state.pop %>%
  group_by(region) %>%
  summarise(long = mean(long), lat = mean(lat), group = group)
# Join 2 dataframe by region
state.pop <- left_join(states_map, state.tbl, by = 'region')
# Plot the % of population by state
ggplot(state.pop , aes(long, lat, group = group)) +
  geom_polygon(aes(fill = Percent), color = "white")+
  geom_text(aes(label = region), data = state.map,  size = 3, hjust = 0.5) +
  scale_fill_viridis_c(option = "C") +
  labs( x = 'Longtitude', y = 'Latitude', title = "Percentage of Population by State") +
  theme(plot.title = element_text(hjust = 0.5))

pop.region <- clean.pop %>%
  count(Region, wt = Weight) %>%
  mutate(Percent = 100*round(n/sum(n), digits = 3))
```

```r
# Create in-line code
hi.reg <- max(pop.region$Percent)
lo.reg <- min(pop.region$Percent)
# Calculate the percentage of population by region
r1 <- clean.pop %>%
  group_by(Sex) %>%
  count(Region, wt = Weight) %>%
  mutate(Percent = 100*round(n/sum(n), digits = 3)) %>%
  ggplot(aes(x= reorder(Region, n), y = Percent, fill = Sex)) +
  geom_col(position = "dodge") +
  labs(x = 'Region',title = "Percentage of Population by Region and Sex") +
  theme(plot.title = element_text(hjust = 0.5, size = 12)) +
  coord_flip()
# Calculate the percentage of population by region and race
r2 <- clean.pop %>%
  group_by(Citizenship) %>%
  count(Region, wt = Weight) %>%
  mutate(Percent = 100*round(n/sum(n), digits = 3)) %>%
  ggplot(aes(x= reorder(Region, n), y = Percent, fill = Citizenship)) +
  geom_col(position = "dodge") +
  labs(x = 'Region',title = "Percentage of Population by Region and Sex") +
  theme(plot.title = element_text(hjust = 0.5, size = 12)) +
  coord_flip()
plot_grid(r1,r2)

# Generate the percentage of population by Class of Worker
class <- clean.pop %>%
  count(Worker.Class, wt = Weight) %>%
  mutate(Percent = 100*round(n/sum(n), digits = 3)) %>%
  arrange(desc(Percent))

# Create in-line code
pri <- filter(class, Worker.Class == 'Private')$Percent
gov <- filter(class, Worker.Class == 'Government')$Percent
self <- filter(class, Worker.Class == 'Self.Employed')$Percent
unem <- filter(class, Worker.Class == 'Unemployed')$Percent
na.cla <- filter(class, is.na(Worker.Class))$Percent

# Plot the percentage of population by Class of Worker
class %>%
  ggplot(aes(x= reorder(Worker.Class, n), y = Percent, fill = Worker.Class)) +
  geom_bar(stat = "identity") +
  labs(x = 'Class of Worker',title = "Percentage of Population by Class of Worker") +
  theme(plot.title = element_text(hjust = 0.5), legend.position = 'none') +
  coord_flip()
# Generate the percentage of population by Education Attainment
edu <- clean.pop %>%
  count(Education, wt = Weight) %>%
  mutate(Percent = 100*round(n/sum(n), digits = 3))

# Create in-line code
ba <- filter(edu, Education == 'Bachelor')$Percent
co <- filter(edu, Education == 'College')$Percent
dip <- filter(edu, Education == 'HS Diploma')$Percent
ma <- filter(edu, Education == 'Master')$Percent
PhD <- filter(edu, Education == 'PhD')$Percent
no <- filter(edu, Education == 'No.school')$Percent
sec <- filter(edu, Education == 'Secondary')$Percent
na.edu <- filter(edu, is.na(Education))$Percent

# Plot the percentage of population by Education Attainment
edu %>%
  ggplot(aes(fct_inorder(Education), Percent, fill = Education)) +
  geom_bar(stat = "identity") +
  labs(x = 'Education',title = "Percentage of Population by Education Attainment") +
  theme(plot.title = element_text(hjust = 0.5), legend.position = "none") +
  coord_flip()
# Generate the percentage of population by Marital Status
marital <- clean.pop %>%
  count(Marital.Status, wt = Weight) %>%
  mutate(Percent = 100*round(n/sum(n), digits = 3))

# Create in-line code
never <- filter(marital, Marital.Status == 'Never married/<15')$Percent
mar <- filter(marital, Marital.Status == 'Married')$Percent
div <- filter(marital, Marital.Status == 'Divorced')$Percent
sum <- filter(marital, Marital.Status == 'Separated')$Percent +
  filter(marital, Marital.Status == 'Widowed')$Percent

# Plot the percentage of population by Marital Status
marital %>%
  ggplot(aes(x= reorder(Marital.Status, n), y = Percent, fill = Marital.Status)) +
  geom_bar(stat = "identity") +
  labs(x = 'Marital Status',title = "Percentage of Population by Marital Status") +
  theme(plot.title = element_text(hjust = 0.5), legend.position = "none") +
  coord_flip()
# Generate the percentage of population by Employment status
emp <- clean.pop %>%
  count(Employment, wt = Weight) %>%
  mutate(Percent = 100*round(n/sum(n), digits = 3))

# Create in-line code
cil <- filter(emp, Employment == 'Civil.employed')$Percent
not.lf <- filter(emp, Employment == 'Not.Labor.Force')$Percent
unempl <- filter(emp, Employment == 'Unemployed')$Percent
arm <- filter(emp, Employment == 'Arm.force')$Percent

# Plot the percentage of population by Marital Status
emp %>%
  ggplot(aes(x= reorder(Employment, n), y = Percent, fill = Employment)) +
  geom_bar(stat = "identity") +
  labs(x = 'Employment',title = "Percentage of Population by Employment") +
  theme(plot.title = element_text(hjust = 0.5), legend.position = "none") +
  coord_flip()
# Summary statistics of Age
mean.age <- round(wtd.mean(clean.pop$Age, weights= clean.pop$Weight, normwt = FALSE), digits = 0)
qtl.age <- wtd.quantile(clean.pop$Age,weights= clean.pop$Age, normwt = FALSE)
q1.age <- qtl.age[2]
q3.age <- qtl.age[4]
sd.age <- round(sqrt(wtd.var(clean.pop$Age, weights= clean.pop$Weight, normwt = FALSE)), digits = 0)

# Generate the percentage of population by Age Group
age.grp <- clean.pop %>%
  count(Age.Group, wt = Weight) %>%
  mutate(Percent = 100*round(n/sum(n), digits = 3))

# Generate the percentage of population by Age Group and Sex
age.grp.sex <- clean.pop %>%
  group_by(Sex) %>%
  count(Age.Group, wt = Weight) %>%
  mutate(Percent = 100*round(n/sum(n), digits = 3))

# Create in-line code
sen <- filter(age.grp, Age.Group == 'Senior')$Percent
f.sen <- filter(age.grp.sex, Age.Group == 'Senior', Sex == 'Female')$Percent
m.sen <- filter(age.grp.sex, Age.Group == 'Senior', Sex == 'Male')$Percent
f.chi <- filter(age.grp.sex, Age.Group == 'Children', Sex == 'Female')$Percent
m.chi <- filter(age.grp.sex, Age.Group == 'Children', Sex == 'Male')$Percent

# Plot the percentage of population by Age group
a1 <- age.grp %>%
  ggplot(aes(fct_inorder(Age.Group), Percent, fill = Age.Group)) +
```

```r
  geom_bar(stat = "identity") +
  labs(x = 'Age Group',title = "Percent of Population by Age Group") +
  theme(plot.title = element_text(hjust = 0.5, size = 12), legend.position = "none") +
  coord_flip()
# Plot the percentage of population by Age group and Sex
a2 <- age.grp.sex %>%
  ggplot(aes(fct_inorder(Age.Group), Percent, fill = Sex)) +
  geom_col(position = 'dodge') +
  labs(x = 'Age Group',title = "% of Population by Age Group and Sex") +
  theme(plot.title = element_text(hjust = 0.5, size = 12)) +
  coord_flip()
# Combine 2 plots
plot_grid(a1,a2)

# Summary statistics of Income
mean.inc <- round(wtd.mean(clean.pop$Total.Income, weights= clean.pop$Weight,
                           normwt = FALSE, na.rm = TRUE), digits = 0)
qtl.inc <- round(wtd.quantile(clean.pop$Total.Income,weights= clean.pop$Weight,
                              normwt = FALSE, na.rm = TRUE), digits = 0)
q1.inc <- qtl.inc[2]
q3.inc <- qtl.inc[4]
sd.inc <- round(sqrt(wtd.var(clean.pop$Total.Income, weights= clean.pop$Weight,
                             normwt = FALSE, na.rm = TRUE)), digits = 0)
# Number of NA's
na.inc <- nrow(clean.pop %>% filter(is.na(Total.Income)))

# Generate the percentage of population by Age Group
inc.grp <- clean.pop %>%
  count(Income.Group, wt = Weight) %>%
  mutate(Percent = 100 * round(n/sum(n), digits = 3))

# Generate the percentage of population by Age Group and Sex
inc.grp.sex <- clean.pop %>%
  group_by(Sex) %>%
  count(Income.Group, wt = Weight) %>%
  mutate(Percent = 100*round(n/sum(n), digits = 3))

# Create in-line code
vl <- filter(inc.grp, Income.Group == 'Very Low')$Percent
up <- filter(inc.grp, Income.Group == 'Upper Middle')$Percent
hi <- filter(inc.grp, Income.Group == 'High')$Percent

# Plot the percentage of population by Income group
i1 <- inc.grp %>%
  ggplot(aes(fct_inorder(Income.Group), Percent, fill = Income.Group)) +
  geom_bar(stat = "identity") +
  labs(x = 'Income Group',title = "% of Population by Income Group") +
  theme(plot.title = element_text(hjust = 0.5, size = 12), legend.position = "none") +
  coord_flip()

i2 <- inc.grp.sex %>%
  ggplot(aes(fct_inorder(Income.Group), Percent, fill = Sex)) +
  geom_col(position = 'dodge') +
  labs(x = 'Income Group',title = "% of Population by Income Group and Sex") +
  theme(plot.title = element_text(hjust = 0.5, size = 12)) +
  coord_flip()
plot_grid(i1,i2)

# Summary statistics of Income to Poverty Ratio
mean.pov <- round(wtd.mean(clean.pop$Inc.Pov.Rate, weights= clean.pop$Weight,
                           normwt = FALSE, na.rm = TRUE), digits = 0)
qtl.pov <- round(wtd.quantile(clean.pop$Inc.Pov.Rate,weights= clean.pop$Weight,
                              normwt = FALSE, na.rm = TRUE), digits = 0)
q1.pov <- qtl.pov[2]
q3.pov <- qtl.pov[4]
sd.pov <- round(sqrt(wtd.var(clean.pop$Inc.Pov.Rate, weights= clean.pop$Weight,
                             normwt = FALSE, na.rm = TRUE)), digits = 0)
# Number of NA's
na.pov <- nrow(clean.pop %>% filter(is.na(Inc.Pov.Rate)))

avg.inc.pov <- clean.pop %>%
  group_by(States) %>%
  summarise(Average.Ratio = mean(Inc.Pov.Rate, na.rm = TRUE))

# Create map plot of all states in USA and change the name of `region` column into ` States`
states_map <- map_data("state") %>%
  mutate(States = region)

# Calculate average longitude and latitude
state.map <- state.unem %>%
  group_by(region) %>%
  summarise(long = mean(long), lat = mean(lat), group = group)

# Join two dataframe by state
state.unem <- left_join(states_map, avg.inc.pov, by = 'States')

# Plot Income to Poverty Ratio by State
ggplot(state.unem , aes(long, lat, group = group)) +
```

```r
  geom_polygon(aes(fill = Average.Ratio), color = "white")+
  geom_text(aes(label = region), data = state.map,  size = 3, hjust = 0.5) +
  scale_fill_viridis_c(option = "D") +
  labs( x = 'Longtitude', y = 'Latitude', title = "Income to Poverty Ratio by State") +
  theme(plot.title = element_text(hjust = 0.5))

# Create inline codes
min.pov <- round(min(avg.inc.pov$Average.Ratio), digits = 0)
lo.pov <- as.vector(filter(avg.inc.pov, Average.Ratio == min(avg.inc.pov$Average.Ratio))$States)
hi.pov <- as.vector(filter(avg.inc.pov, Average.Ratio == max(avg.inc.pov$Average.Ratio))$States)
max.pov <- round(max(avg.inc.pov$Average.Ratio), digits = 0)


# Identify outliers for Total Income
inc.outlier <- clean.pop %>%
  identify_outliers(Total.Income)
# Calculate percentage of total income outliers
p.inc.out <- round(length(inc.outlier$Total.Income) / length(clean.pop$Total.Income), digits = 3)*100
# Identify outliers for Poverty
pov.outlier <- clean.pop %>%
  identify_outliers(Inc.Pov.Rate) # No outliers
# Filter out outliers of Total Income, and people under 16 years old and NA values for Income and Inc.
         Pov.Rate, Education, Worker.Class and Employment
filter.pop <- clean.pop %>%
  filter(Age >= 16,
         !is.na(Total.Income),
         !is.na(Inc.Pov.Rate),
         !is.na(Education),
         !is.na(Worker.Class),
         !is.na(Employment)) %>%
  anti_join(inc.outlier)
# Unemployment rate between States, Region and Sex
Unem.state <- filter.pop %>%
  group_by(States, Region, Sex) %>%
  count(Employment, wt = Weight) %>%
  spread(Employment, value = n) %>%
  summarise(Unemploy.rate = round(Unemployed/Civil.employed, digits = 3)*100)


# Average Unemployment rate between Region
Unem.state %>%
  group_by(Region, Sex) %>%
  summarise(avg.unem = mean(Unemploy.rate)) %>%
  ggplot(aes(reorder(Region, avg.unem), avg.unem, fill = Sex)) +
  geom_col(position = 'dodge')+
  labs(y = 'Average Unemployment Rate (%)', x = 'States',
       title = 'Average Unemployment Rate by Region and Sex')+
  theme(plot.title = element_text(hjust = 0.5, size = 12)) +
  coord_flip()

# Create inline code
Unem.reg <- Unem.state %>%
  group_by(Region) %>%
  summarise(avg.unem = mean(Unemploy.rate))
hi.S <- round(max(Unem.reg$avg.unem), digits = 1)
lo.MW <- round(min(Unem.reg$avg.unem), digits = 1)
bw <- round(Unem.reg$avg.unem[Unem.reg$Region == 'Northeast'], digits = 1)

# Create inline codes
## The lowest and highest % of unemployment rate in the Northeast
lo.NE <- min(filter(Unem.state, Region == 'Northeast')$Unemploy.rate)
hi.NE <- max(filter(Unem.state, Region == 'Northeast')$Unemploy.rate)
## The lowest and highest % of unemployment rate in the South
lo.S <- min(filter(Unem.state, Region == 'South')$Unemploy.rate)
hi.S <- max(filter(Unem.state, Region == 'South')$Unemploy.rate)
## The lowest and highest % of unemployment rate in the Midwest
lo.MW <- min(filter(Unem.state, Region == 'Midwest')$Unemploy.rate)
hi.MW <- max(filter(Unem.state, Region == 'Midwest')$Unemploy.rate)
## The lowest and highest % of unemployment rate in the West
lo.W <- min(filter(Unem.state, Region == 'West')$Unemploy.rate)
hi.W <- max(filter(Unem.state, Region == 'West')$Unemploy.rate)
## States' names with the lowest and highest % of unemployment rate in the USA
lo.st.name <- as.vector(filter(Unem.state,Unemploy.rate == min(Unem.state$Unemploy.rate))$States)
lo.st.unem <- min(Unem.state$Unemploy.rate) # States with lowest % of unemployment rate
hi.st.name <- as.vector(filter(Unem.state,Unemploy.rate == max(Unem.state$Unemploy.rate))$States)
hi.st.unem <- max(Unem.state$Unemploy.rate) # States with highest % of unemployment rate

# Unemployment rate between States in Northeast region
p1 <- Unem.state %>%
  filter(Region == 'Northeast') %>%
  ggplot(aes(reorder(States,Unemploy.rate), Unemploy.rate, fill = States)) +
  geom_col()+
  labs(y = 'Unemployment Rate (%)', x = 'States',
       title = 'Unemployment Rate in Northeast')+
  theme(plot.title = element_text(hjust = 0.5, size = 12), legend.position = 'none') +
  coord_flip()

# Unemployment rate between States in Northeast region
p2 <- Unem.state %>%
  filter(Region == 'South') %>%
  ggplot(aes(reorder(States,Unemploy.rate), Unemploy.rate, fill = States)) +
  geom_col()+
  labs(y = 'Unemployment Rate (%)', x = 'States',
       title = 'Unemployment Rate in South')+
  theme(plot.title = element_text(hjust = 0.5, size = 12), legend.position = 'none') +
  coord_flip()

# Unemployment rate between States in Midwest region
p3 <- Unem.state %>%
  filter(Region == 'Midwest') %>%
  ggplot(aes(reorder(States,Unemploy.rate), Unemploy.rate, fill = States)) +
  geom_col()+
  labs(y = 'Unemployment Rate (%)', x = 'States',
       title = 'Unemployment Rate in Midwest') +
  theme(plot.title = element_text(hjust = 0.5, size = 12), legend.position = 'none') +
  coord_flip()

# Unemployment rate between States in West region
p4 <- Unem.state %>%
  filter(Region == 'West') %>%
  ggplot(aes(reorder(States,Unemploy.rate), Unemploy.rate, fill = States)) +
  geom_col()+
  labs(y = 'Unemployment Rate (%)', x = 'States',
       title = 'Unemployment Rate in West')+
  theme(plot.title = element_text(hjust = 0.5, size = 12),legend.position = 'none' ) +
  coord_flip()

# Combine 4 plots
plot_grid(p1,p2,p3, p4)

# Create map plot of all states in USA and change the name of `region` column into ` States`
states_map <- map_data("state") %>%
  mutate(States = region)
# Calculate average longitude and latitude
state.map <- state.unem %>%
  group_by(region) %>%
  summarise(long = mean(long), lat = mean(lat), group = group)

# Join two dataframe by state
state.unem <- left_join(states_map, Unem.state, by = 'States')

# Plot unemployment rate by State
ggplot(state.unem , aes(long, lat, group = group)) +
  geom_polygon(aes(fill = Unemploy.rate), color = "white")+
  geom_text(aes(label = region), data = state.map,  size = 3, hjust = 0.5) +
  scale_fill_viridis_c(option = "C") +
  labs( x = 'Longtitude', y = 'Latitude', title = "Unemployment rate by State") +
  theme(plot.title = element_text(hjust = 0.5))

# Unemployment rate by Race
Unem.race <- filter.pop %>%
  group_by(Race) %>%
  count(Employment, wt = Weight) %>%
  spread(Employment, value = n) %>%
  summarise(Unemploy.rate = round(Unemployed/Civil.employed, digits = 3)*100)

# Unemployment rate by Race and Sex
Unem.race.sex <- filter.pop %>%
  group_by(Race, Sex) %>%
  count(Employment, wt = Weight) %>%
  spread(Employment, value = n) %>%
  summarise(Unemploy.rate = round(Unemployed/Civil.employed, digits = 3)*100)
#Unemployment rate by Race and Sex
rac.sex <- Unem.race.sex %>%
  ggplot(aes(fct_relevel(reorder(Race,Unemploy.rate), 'Others'), Unemploy.rate, fill =  Sex)) +
  geom_col(position = 'dodge') +
  labs(y = 'Unemployment Rate (%)', x = 'Race', title = 'Unemployment Rate by Race and Sex') +
  theme(plot.title = element_text(hjust = 0.5, size = 12), legend.position = 'none') +
  coord_flip()

# Unemployment rate by Citizenship
Unem.cit <- filter.pop %>%
  group_by(Citizenship) %>%
  count(Employment, wt = Weight) %>%
  spread(Employment, value = n) %>%
  summarise(Unemploy.rate = round(Unemployed/Civil.employed, digits = 3)*100)
# Unemployment rate by Citizenship and Sex
Unem.cit.sex <- filter.pop %>%
  group_by(Citizenship, Sex) %>%
  count(Employment, wt = Weight) %>%
  spread(Employment, value = n) %>%
  summarise(Unemploy.rate = round(Unemployed/Civil.employed, digits = 3)*100)
# Plot Unemployment rate by Citizenship and Sex
cit.sex <- Unem.cit.sex %>%
  ggplot(aes(reorder(Citizenship,Unemploy.rate), Unemploy.rate, fill =  Sex)) +
  geom_col(position = 'dodge')+
  labs(y = 'Unemployment Rate (%)', x = 'Citizenship', title = 'Unemployment Rate by Citizenship')+
  theme(plot.title = element_text(hjust = 0.5, size = 12))
```

```
#Combine 2 plots
plot_grid(cit.sex, rac.sex)

# Create inline codes for citizenship
US.unem <- filter(Unem.cit, Unem.cit$Citizenship == 'US')$Unemploy.rate
non.unem <- filter(Unem.cit, Unem.cit$Citizenship == 'Non.US')$Unemploy.rate
# Create inline codes for citizenship
unem.Blk <- filter(Unem.race, Unem.race$Race == 'Black')$Unemploy.rate
unem.Wt <- filter(Unem.race, Unem.race$Race == 'White')$Unemploy.rate
unem.Asi <- filter(Unem.race, Unem.race$Race == 'Asian')$Unemploy.rate

# Unemployment rate by Education
Unem.edu <- filter.pop %>%
  group_by(Education) %>%
  count(Employment, wt = Weight) %>%
  spread(Employment, value = n) %>%
  summarise(Unemploy.rate = round(Unemployed/Civil.employed, digits = 3)*100)

# Plot unemployment rate by Education
Unem.edu %>%
  ggplot(aes(Education, Unemploy.rate, fill =  Education)) +
  geom_col(position = 'dodge')+
  labs(y = 'Unemployment Rate (%)', x = 'Education', title = 'Unemployment Rate by Education')+
  theme(plot.title = element_text(hjust = 0.5), legend.position = 'none') +
  coord_flip()

# Create inline codes
hi.edu <-max(Unem.edu$Unemploy.rate) # Highest % of unemployment for education attainment
lo.edu <-min(Unem.edu$Unemploy.rate) # Lowest % of unemployment for education attainment

# Calculate the variance of each group
aggregate(Unemploy.rate ~ Sex, data=Unem.state, var)

# Test for normality of Unemployment rate distribution
sha.test <- shapiro.test(Unem.state$Unemploy.rate)
sha.test
p.unem.s <- sha.test$p.value

# Test for normality of Unemployment rate distribution grouped by Sex
sha.test.f <- shapiro.test(Unem.state$Unemploy.rate[Unem.state$Sex == "Female"]) # For Female
sha.test.f
p.unem.f <- sha.test.f$p.value # P-value for Female
sha.test.m <- shapiro.test(Unem.state$Unemploy.rate[Unem.state$Sex == "Male"]) # For Male
sha.test.f
```

```r
p.unem.m <-sha.test.m$p.value # P-value for Female

# Visualize normality of Unemployment rate distribution
qqnorm(Unem.state$Unemploy.rate)

# Visualize normality of Unemployment rate distribution between group
ggplot(Unem.state, aes(x = Unemploy.rate, fill = Sex)) +
  geom_histogram(binwidth=.5, alpha=1/2) +
  labs(x = 'Unemployment Rate', y = 'Probability', title = 'Distribution of Unemployment Rate by Sex')

# Weighted Independent t-test
x.sex <- Unem.state$Unemploy.rate[Unem.state$Sex == "Female"]
y.sex <- Unem.state$Unemploy.rate[Unem.state$Sex == "Male"]
t.mdl <- wtd.t.test(x = x.sex, y = y.sex, weight = filter.pop$Weight,
                    weighty = filter.pop$Weight, mean1 = FALSE)
t.mdl
# Create inline code
t.sex <-  round(t.mdl$coefficients[1], digits = 2)  # t statistics
f.mean <- round(t.mdl$additional[2], digits = 2)    # Average unemployment rate for female
m.mean <- round(t.mdl$additional[3], digits = 2)    # Average unemployment rate for male
se.unem <-round(t.mdl$additional[4],digits = 2)     # Standard error
df.unem <-round(t.mdl$coefficients[2], digits = 2)  # Degree of freedom
p.unem <- round(t.mdl$coefficients[3], digits = 2)  # p_value

# Unemployment rate by States and Citizenship
unem.st.cit <- filter.pop %>%
  group_by(States, Citizenship) %>%
  count(Employment, wt = Weight) %>%
  spread(Employment, value = n) %>%
  summarise(Unemploy.rate = round(Unemployed/Civil.employed, digits = 3)*100)

# Calculate the variance of each group
aggregate(Unemploy.rate ~ Citizenship, data = unem.st.cit, var)

# Test normality of Unemployment rate distribution for Citizenship
sha.test.cit <-shapiro.test(unem.st.cit$Unemploy.rate)
sha.test.cit
p.unem.cit <- sha.test.cit$p.value

# Test for normality of Unemployment rate distribution grouped by Citizenship
sha.test.us <- shapiro.test(unem.st.cit$Unemploy.rate[unem.st.cit$Citizenship == "US"])
sha.test.us
p.unem.us <- sha.test.us$p.value # P-value for US
sha.test.non <- shapiro.test(unem.st.cit$Unemploy.rate[unem.st.cit$Citizenship == "Non.US"])
sha.test.non
p.unem.non <- sha.test.non$p.value # P-value for US

# Check normality of distribution
qqnorm(unem.st.cit$Unemploy.rate)

# Visualize the difference between groups
ggplot(unem.st.cit, aes(x = Unemploy.rate, fill = Citizenship)) +
  geom_density(alpha=1/2)+
  labs(x = 'Unemployment Rate', y = 'Probability',
       title = 'Distribution of Unemployment Rate by Citizenship')

# Independent t-test
x <- unem.st.cit$Unemploy.rate[unem.st.cit$Citizenship == "US"]
y <- unem.st.cit$Unemploy.rate[unem.st.cit$Citizenship == "Non.US"]
t.cit <- wtd.t.test(x = x, y = y, weight = filter.pop$Weight, weighty = filter.pop$Weight,
                    mean1 = FALSE)
t.cit
# Create inline code
t.ci <- round(t.cit$coefficients[1], digits = 2)    # t statistics
us.cit <-  round(t.cit$additional[2], digits = 2)   # Average unemployment rate for US
non.cit <-  round(t.cit$additional[3], digits = 2)  # Average unemployment rate for Non US
se.cit <- round(t.cit$additional[4], digits = 2)    # Standard error
df.cit <-  round(t.cit$coefficients[2], digits = 2) # Degree of freedom
p.cit <-  round(t.cit$coefficients[3], digits = 2)  # p_value

# Create boxplot for income to poverty ratio grouped by race
qplot(Race, Inc.Pov.Rate, geom = "boxplot", data = filter.pop)

# Create boxplot for income to poverty ratio grouped by Citizenship
qplot(Citizenship, Inc.Pov.Rate, geom = "boxplot", data = filter.pop)

# Multiple regression model
pov.lm <- lm(Inc.Pov.Rate ~ Race + Citizenship, data = filter.pop, weights = Weight)
#Summary of regression model
summary(pov.lm)
ti.lm <- tidy(pov.lm)
ti.lm
gla.lm <- glance(pov.lm)
gla.lm
# Create inline codes:
p.intercept <- ti.lm$p.value[1] # p value of White people
p.black <- ti.lm$p.value[2] # p value of Black people
p.others<- ti.lm$p.value[3] # p value of Other people
```

```r
p.asian <- ti.lm$p.value[4] # p value of Asian people
p.non.us <- ti.lm$p.value[5] # p value of Non US people
p.value <- gla.lm$p.value # p value of the model
df1 <- floor(gla.lm$df) # degree of freedom between groups
F_stat <- floor(gla.lm$statistic) # F Statistics
df2 <- floor(gla.lm$df.residual) # degree of freedom within groups
R.squa.per <- 100*round(gla.lm$adj.r.squared, digits = 3) # Adjusted R squared (%)
R.squared <- round(gla.lm$adj.r.squared, digits = 3) # Adjusted R squared
sigma <- round(gla.lm$sigma,1)
est.black <-  round(ti.lm$estimate[2], digits = 1) # Difference between White and Black
est.others <- round(ti.lm$estimate[3], digits = 1) # Difference between White and Others
est.asian <-  round(ti.lm$estimate[4], digits = 1) # Difference between White and Asian
est.non.us <- round(ti.lm$estimate[5], digits = 1) # Difference between US and Non US

# Using diagnostic Plots to examine the assumptions for the regression model
par(mfrow = c(2,2))
plot(lm(Inc.Pov.Rate ~ Race + Citizenship, data = filter.pop, weights = Weight))

# Fit the simpler model with race as the only predictor variable
pov.lm.simple <- lm(Inc.Pov.Rate ~ Race, data = filter.pop, weights = Weight)
summary(pov.lm.simple)

# Compare the simple regression to the more complex model (original model)
ano <- anova(pov.lm.simple, pov.lm)
ano
p.ano <- ano$`Pr(>F)`[2]
```