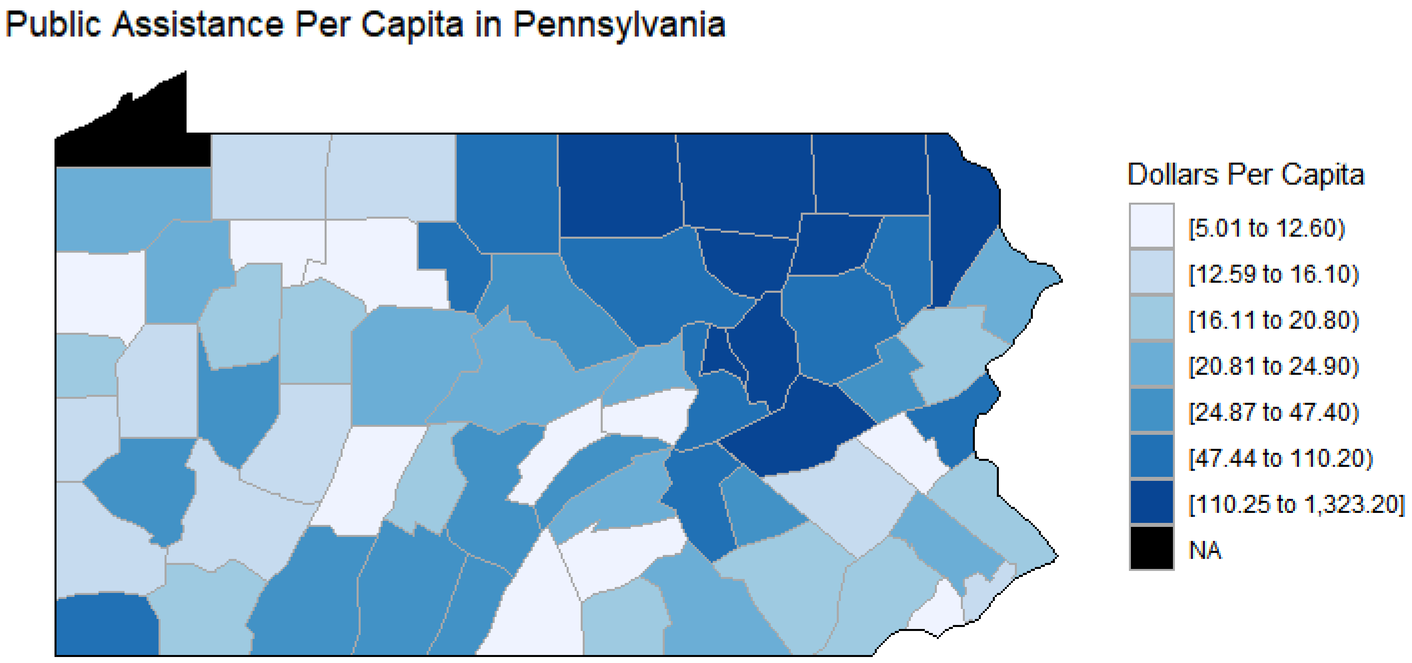
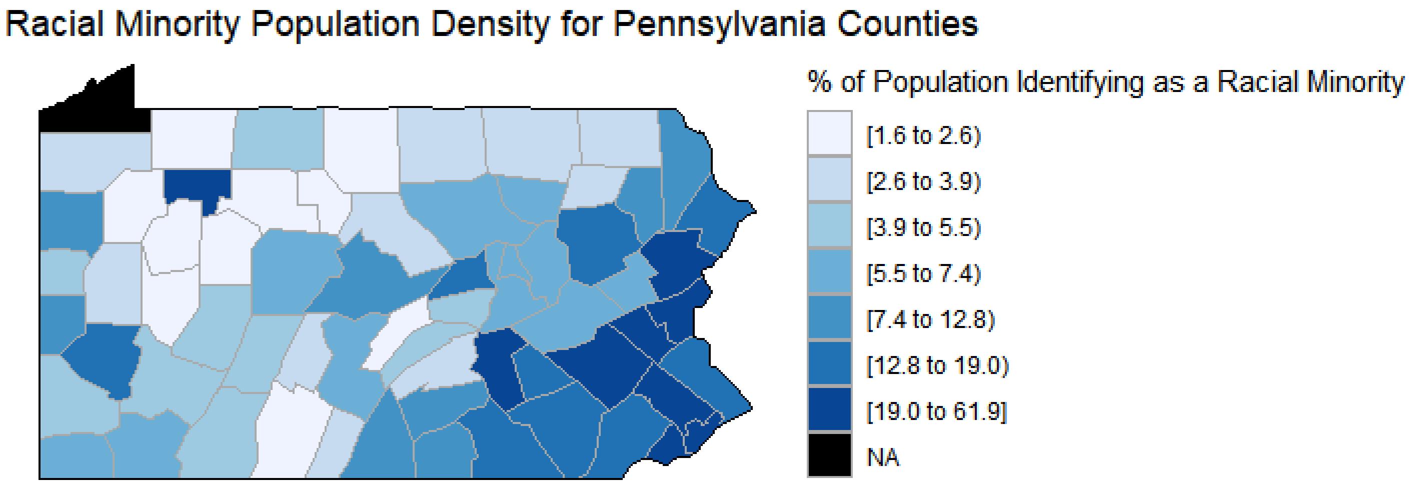
DS 320 FEMA & Census EDA

```
# Front Matter
rm(list = ls())

# Load packages
library(DataComputing)
library(mosaic)
library(ggplot2)
library(openintro)
library(dplyr)
library(stringr)
library(choroplethr)
library(usmap)
library(choroplethrMaps)
library(reshape2)
# Load datasets from FEMA and the United States Census
rawdata <- read.csv("DisasterDeclarationsSummaries.csv")
censusdata <- read.csv("county_census_data.csv")
PublicAssistanceData <- read.csv("PublicAssistanceData.csv")

# Select census variables of interest
census <- censusdata %>% dplyr::select(State, County, TotalPop, Men,
Women, Hispanic, White, Black, Native, Asian, Pacific, Income,
Poverty, Unemployment)

# Filter dataset to 2000 or later where public assistance funding did
occur; limit dataset to county data
data <- rawdata %>% filter(declarationType != "FM") %>%
filter(fyDeclared > 1999) %>% filter(paProgramDeclared == 1) %>%
filter(designatedArea != 'Statewide') %>%
dplyr::select(designatedArea, state, fipsCountyCode, fipsStateCode)

# Clean county variable for FIPS code integration
data$designatedArea <- str_remove_all(data$designatedArea, "(County)")
data$designatedArea <- str_remove_all(data$designatedArea, "(Parish)")
data$designatedArea <- str_remove_all(data$designatedArea,
"(Township)")
data$designatedArea <- gsub("\\(", " ", data$designatedArea)
data$designatedArea <- gsub("\\)", " ", data$designatedArea)

# Compute count of disasters per county
disaster_data <- data %>% group_by(designatedArea, state) %>%
mutate(count = n()) %>% unique()
```

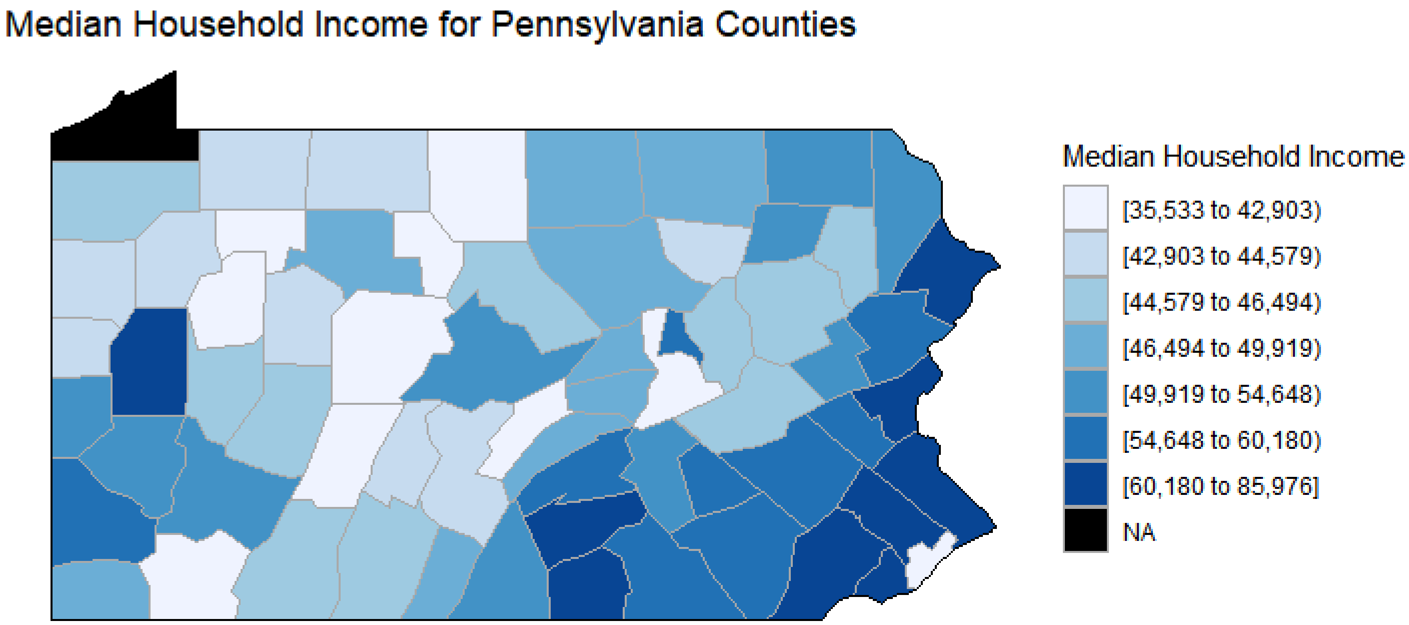
```r
# Prepare for FIPS code integration
disaster_data$full_county <-
paste0(str_trim(disaster_data$designatedArea),sep = " ",
disaster_data$state)
census$county <- census$County
census$state <- census$State

# Limit public assistance dataset to 2000 or later
PublicAssistanceData$declarationDate <-
substr(PublicAssistanceData$declarationDate, 0, 4)
PublicAssistanceData <- PublicAssistanceData %>%
filter(declarationDate > 1999)
padata <- PublicAssistanceData %>% dplyr::select(state, stateCode,
disasterNumber, county, totalObligated, countyCode) %>%
group_by(disasterNumber, county, state) %>% unique()

# Compute county total public assistance obligated
padata <- padata %>% group_by(county, state) %>%
mutate(county_disaster_count = n()) %>% mutate(countyObligated =
sum(totalObligated))
padata <- padata %>% group_by(county, state) %>% dplyr::select(county,
state, county_disaster_count, countyObligated, countyCode, stateCode)
%>% unique()
padata$full_county <- paste0(padata$county, sep=" ", padata$stateCode)

# Merge census and FEMA public assistance data
newdata <- merge(padata, census, by=c("county", "state"))
newdata$County <- NULL
newdata$State <- NULL
newdata <- newdata %>% mutate(county_disasters_perCap =
county_disaster_count/TotalPop) %>% mutate(county_obligated_perCap =
countyObligated/TotalPop)
newdata <- newdata %>% unique() %>% arrange(-county_obligated_perCap)

# Create dataset for mapping
mappingdata <- merge(newdata, disaster_data, by=c("full_county"))
# Use R package to bring in FIPS codes for every county
for (row in 1:nrow(mappingdata)){
  mappingdata$region[[row]] <- 0
  newfips <- 0
  try(newfips <- fips(mappingdata$state.x[[row]],
mappingdata$county[[row]]))
  mappingdata$region[[row]] <- newfips
}
write.csv(mappingdata, "DS320Census-FEMA-Final.csv")

# Exploratory Data Analysis
# Fix "value" and "region" variables to create choropleth maps
mappingdata$value <- mappingdata$county_obligated_perCap
mappingdata <- mappingdata %>% group_by(county, state.x) %>% unique()

# Compute a diversity metric as a sum of all racial minority
population percentages
mappingdata <- mappingdata %>% rowwise() %>% mutate(diversity =
sum(Black, Hispanic, Native, Asian, Pacific))
mappingdata$region <- mappingdata$region %>% as.double()
penn <- mappingdata %>% filter(state.x == "Pennsylvania")

# Create map of Pennsylvania for public assistance funding
county_choropleth(penn, title = "Public Assistance Per Capita in
Pennsylvania", legend = "Dollars Per Capita", state_zoom =
"pennsylvania")
The following regions were missing and are being set to NA: 42049
```



```
penn$value <- penn$diversity
county_choropleth(penn, title = "Racial Minority Population Density
for Pennsylvania Counties", legend = "% of Population Identifying as a
Racial Minority", state_zoom = "pennsylvania")
The following regions were missing and are being set to NA: 42049
```

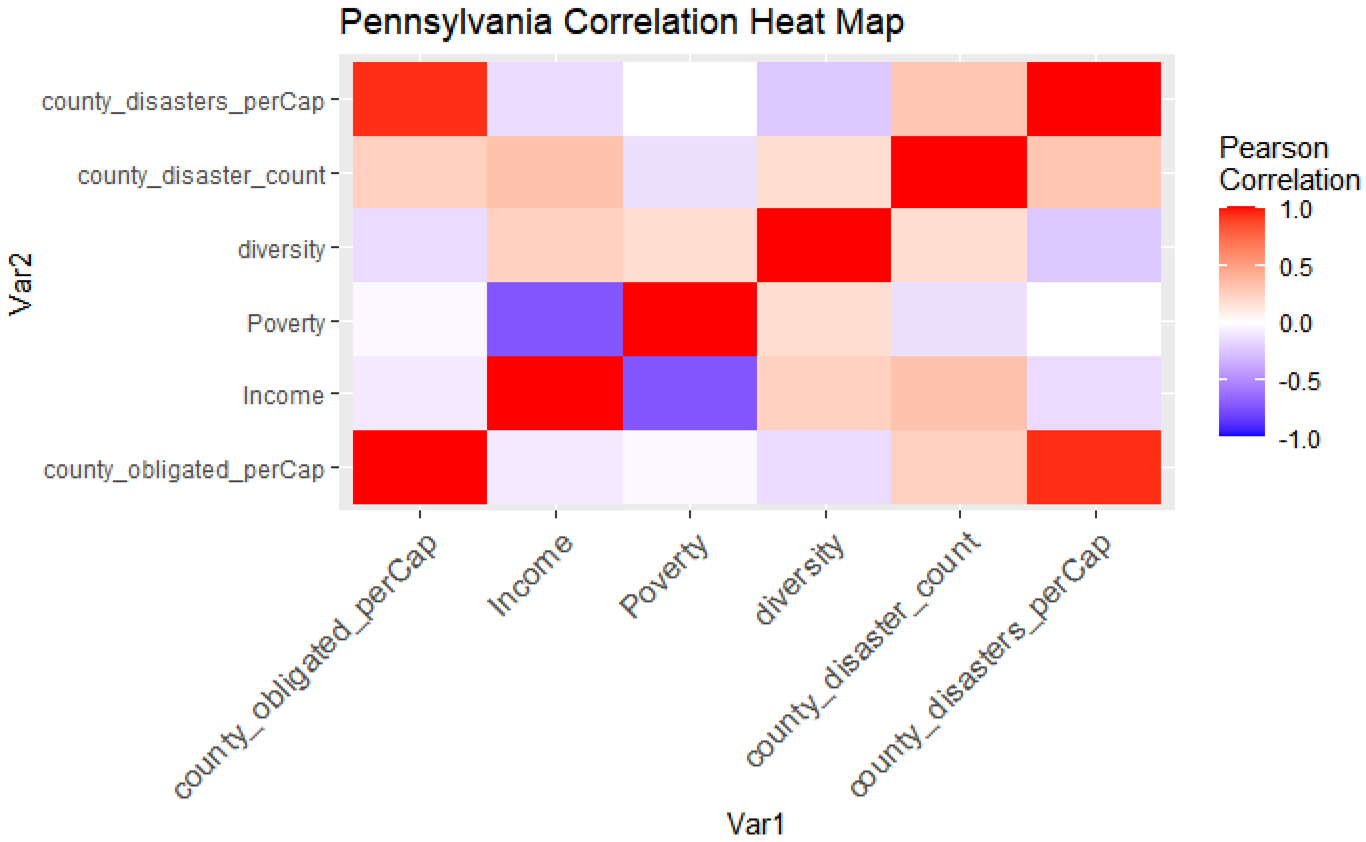

Racial Minority Population Density for Pennsylvania Counties
% of Population Identifying as a Racial Minority
[1.6 to 2.6)
[2.6 to 3.9)
[3.9 to 5.5)
[5.5 to 7.4)
[7.4 to 12.8)
[12.8 to 19.0)
[19.0 to 61.9]
NA

```
penn$value <- penn$Income
county_choropleth(penn, title = "Median Household Income for
Pennsylvania Counties", legend = "Median Household Income", state_zoom
= "pennsylvania")
The following regions were missing and are being set to NA: 42049
```



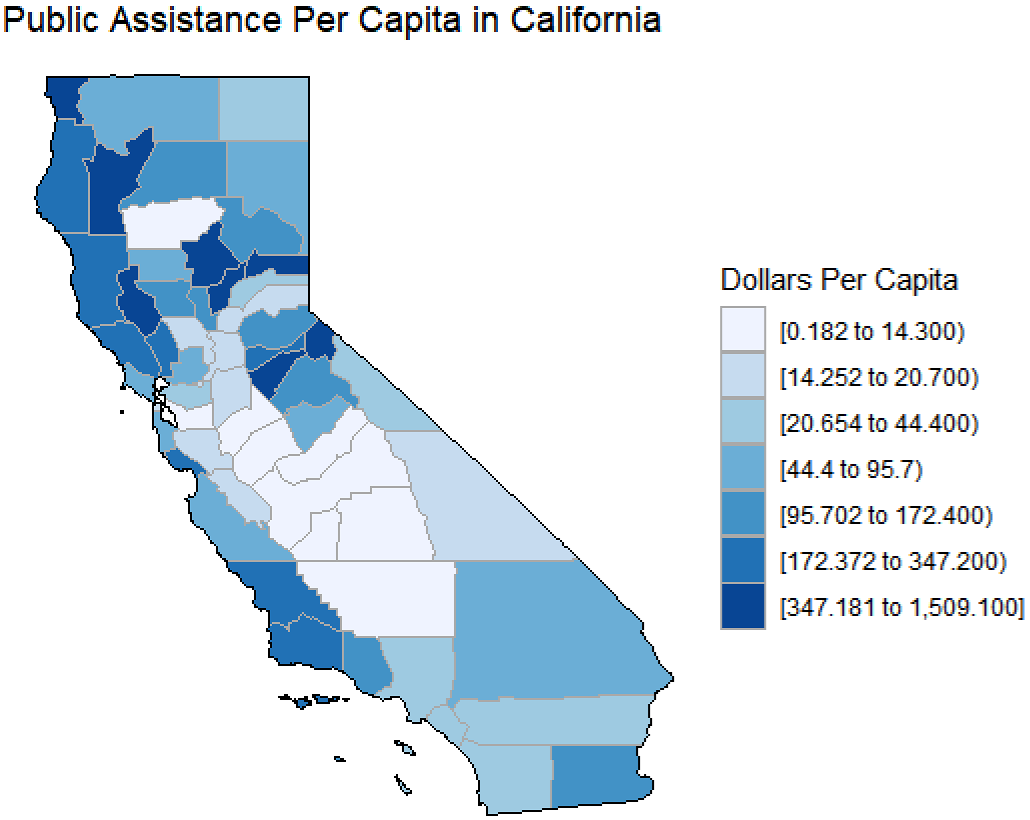
```
cor(penn[, c('county_obligated_perCap', 'Income', 'diversity',
'county_disaster_count', 'county_disasters_perCap', 'Poverty')])

                        county_obligated_perCap      Income  diversity
county_disaster_count county_disasters_perCap      Poverty
county_obligated_perCap              1.00000000 -0.08559873 -0.1519786
0.2497059           9.386658e-01 -2.845711e-02
Income                              -0.08559873  1.00000000  0.2425679
0.3197209          -1.471124e-01 -7.282018e-01
diversity                           -0.15197857  0.24256786  1.0000000
0.1770620          -2.370823e-01  1.794566e-01
county_disaster_count                0.24970590  0.31972086  0.1770620
1.0000000           2.988764e-01 -1.337672e-01
county_disasters_perCap              0.93866583 -0.14711242 -0.2370823
0.2988764           1.000000e+00 -2.683122e-07
Poverty                             -0.02845711 -0.72820183  0.1794566
-0.1337672          -2.683122e-07  1.000000e+00

# Create heatmap to visualize Pennsylvania correlation matrix
pennheat <- penn %>% select(county_obligated_perCap, Income, Poverty,
diversity, county_disaster_count, county_disasters_perCap)
Adding missing grouping variables: `county`, `state.x`
pennheat$state.x <- NULL
```

```
pennheat$county <- NULL

cormatrix <- melt(round(cor(pennheat), 2))
ggplot(data = cormatrix, aes(x=Var1, y=Var2, fill=value)) +
geom_tile() + scale_fill_gradient2(low = "blue", high = "red", mid =
"white", midpoint = 0, limit = c(-1,1), space = "Lab",
name="Pearson\nCorrelation") + theme(axis.text.x = element_text(angle
= 45, vjust = 1, size = 12, hjust = 1)) + labs(title = "Pennsylvania
Correlation Heat Map")
```

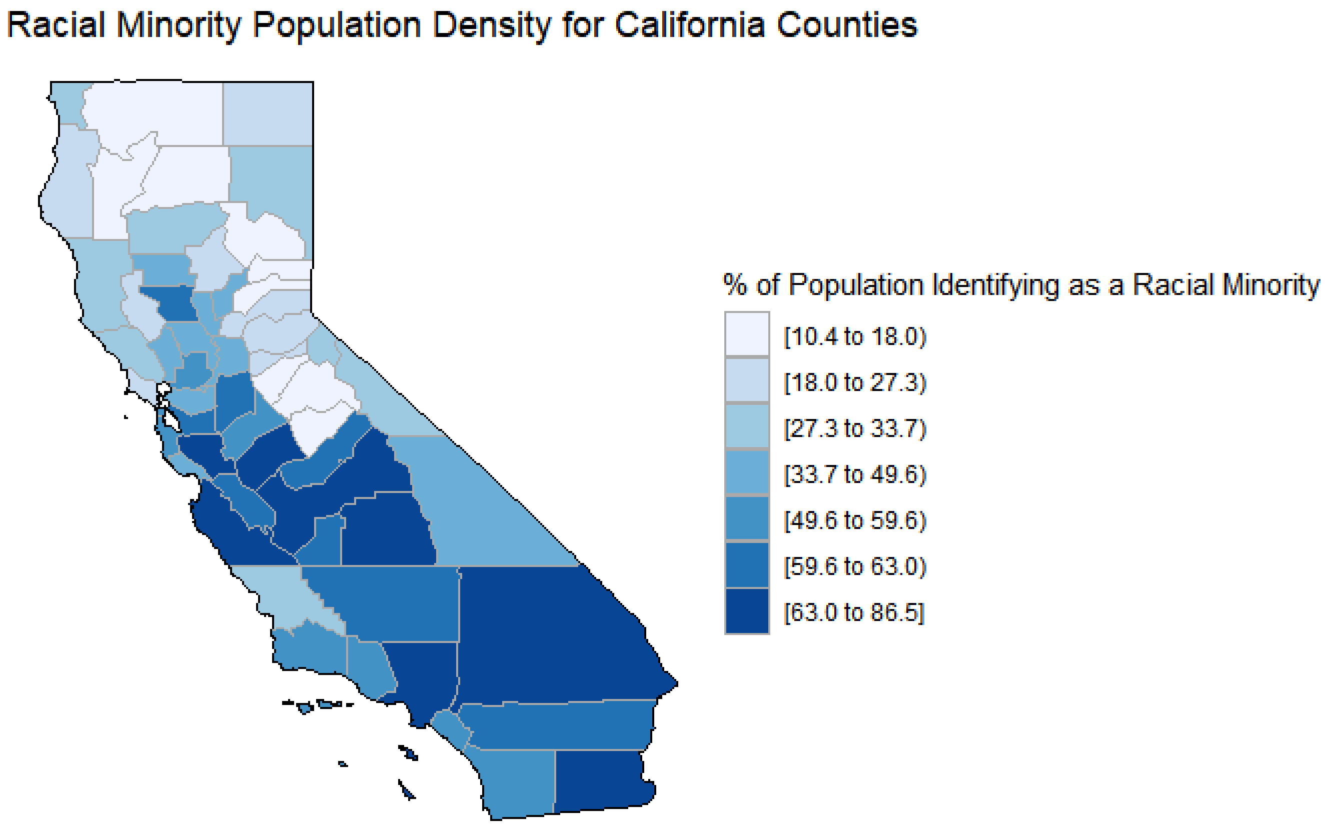


```
# Explore California state data as a correlation matrix
cal <- mappingdata %>% filter(state.x == "California")
cor(cal[, c('county_obligated_perCap', 'Income', 'diversity',
'county_disaster_count', 'county_disasters_perCap', 'Poverty')])

                        county_obligated_perCap     Income   diversity
county_disaster_count county_disasters_perCap     Poverty
county_obligated_perCap              1.00000000 -0.2790185 -0.4370569
-0.08837728              0.70281882  0.07953493
Income                              -0.27901850  1.0000000  0.2288367
0.24780965             -0.15433557 -0.74895642
diversity                           -0.43705690  0.2288367  1.0000000
0.24096936             -0.28460435  0.24115115
county_disaster_count               -0.08837728  0.2478097  0.2409694
```

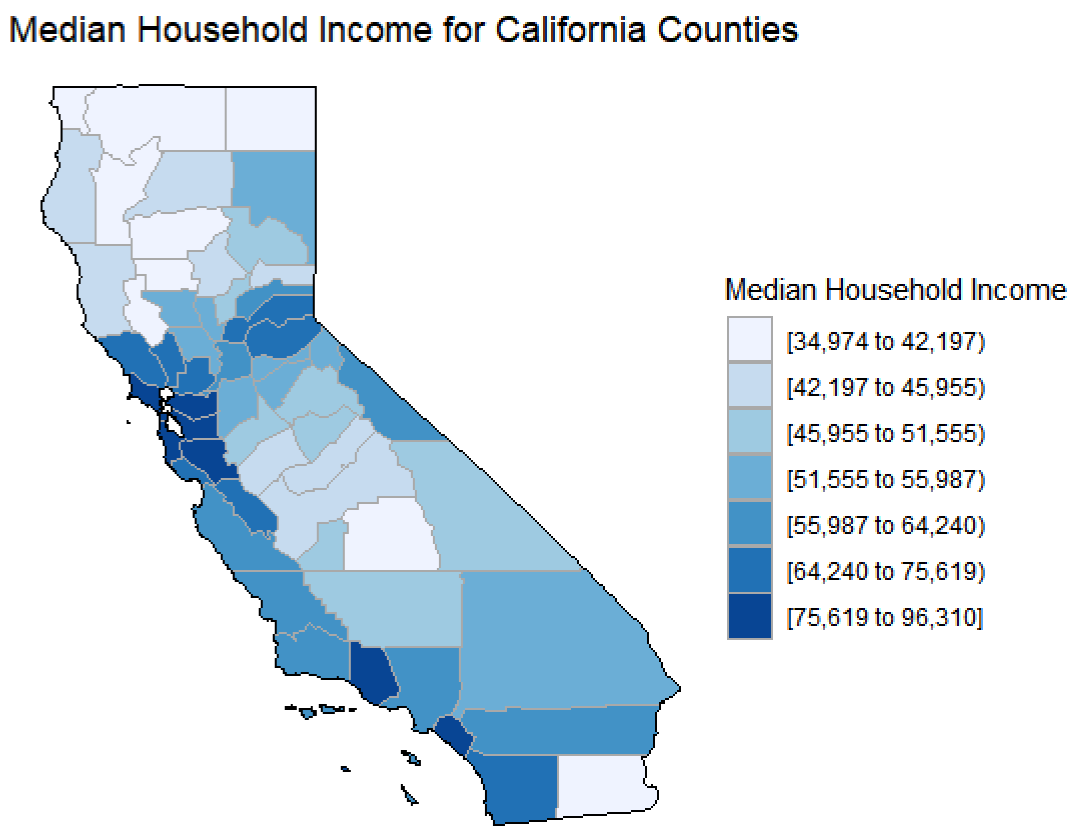
```
1.00000000             -0.12482412 -0.11218381
county_disasters_perCap              0.70281882 -0.1543356 -0.2846043
-0.12482412              1.00000000 -0.02030527
Poverty                              0.07953493 -0.7489564  0.2411512
-0.11218381             -0.02030527  1.00000000

# Create choropleth map of California public assistance funding
county_choropleth(cal, title = "Public Assistance Per Capita in
California", legend = "Dollars Per Capita", state_zoom = "california")
```



```
# Create choropleth map of California diversity
cal$value <- cal$diversity
county_choropleth(cal, title = "Racial Minority Population Density for
California Counties", legend = "% of Population Identifying as a
Racial Minority", state_zoom = "california")
```

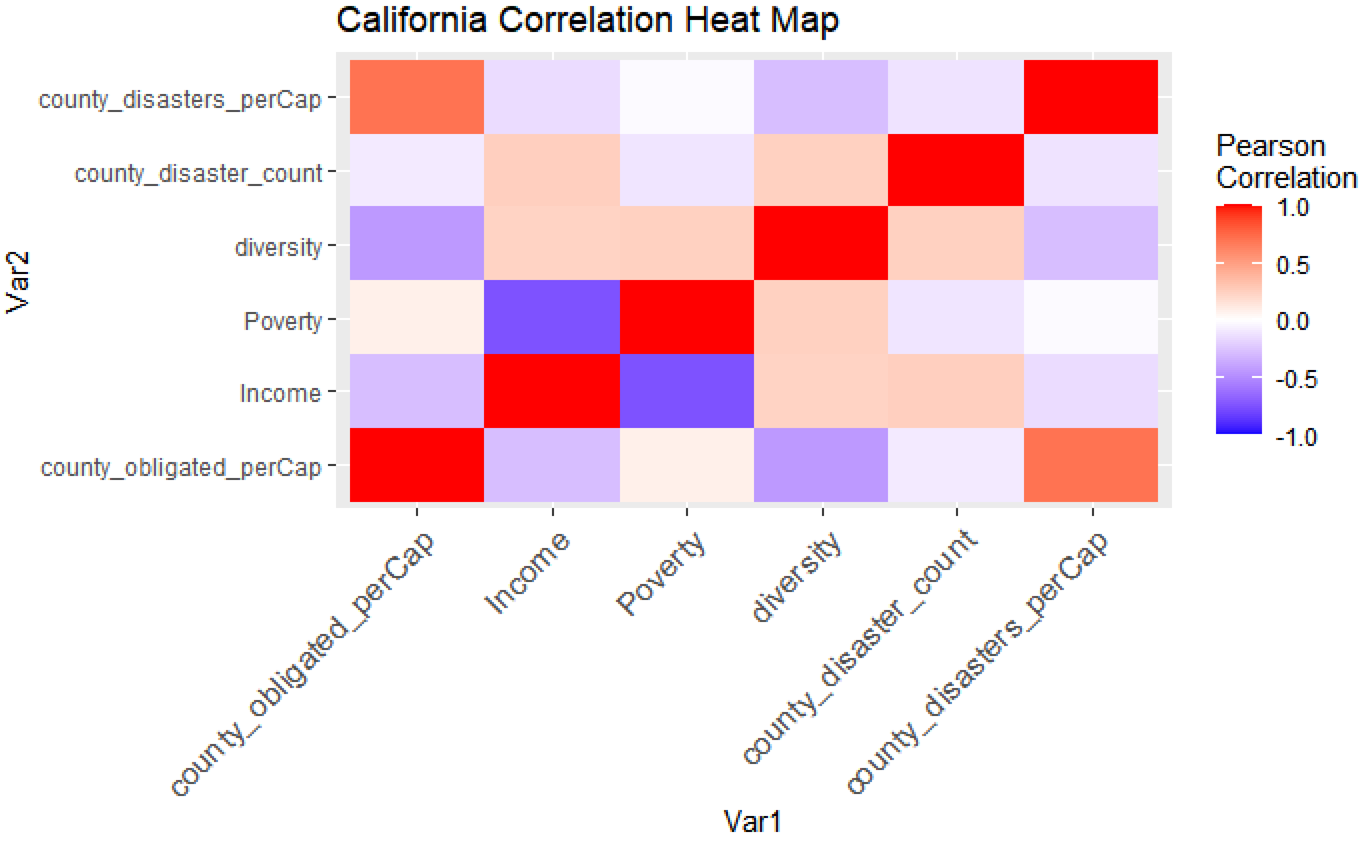

Racial Minority Population Density for California Counties
% of Population Identifying as a Racial Minority
[10.4 to 18.0)
[18.0 to 27.3)
[27.3 to 33.7)
[33.7 to 49.6)
[49.6 to 59.6)
[59.6 to 63.0)
[63.0 to 86.5]

```
# Create choropleth map of California income
cal$value <- cal$Income
county_choropleth(cal, title = "Median Household Income for California
Counties", legend = "Median Household Income", state_zoom =
"california")
```



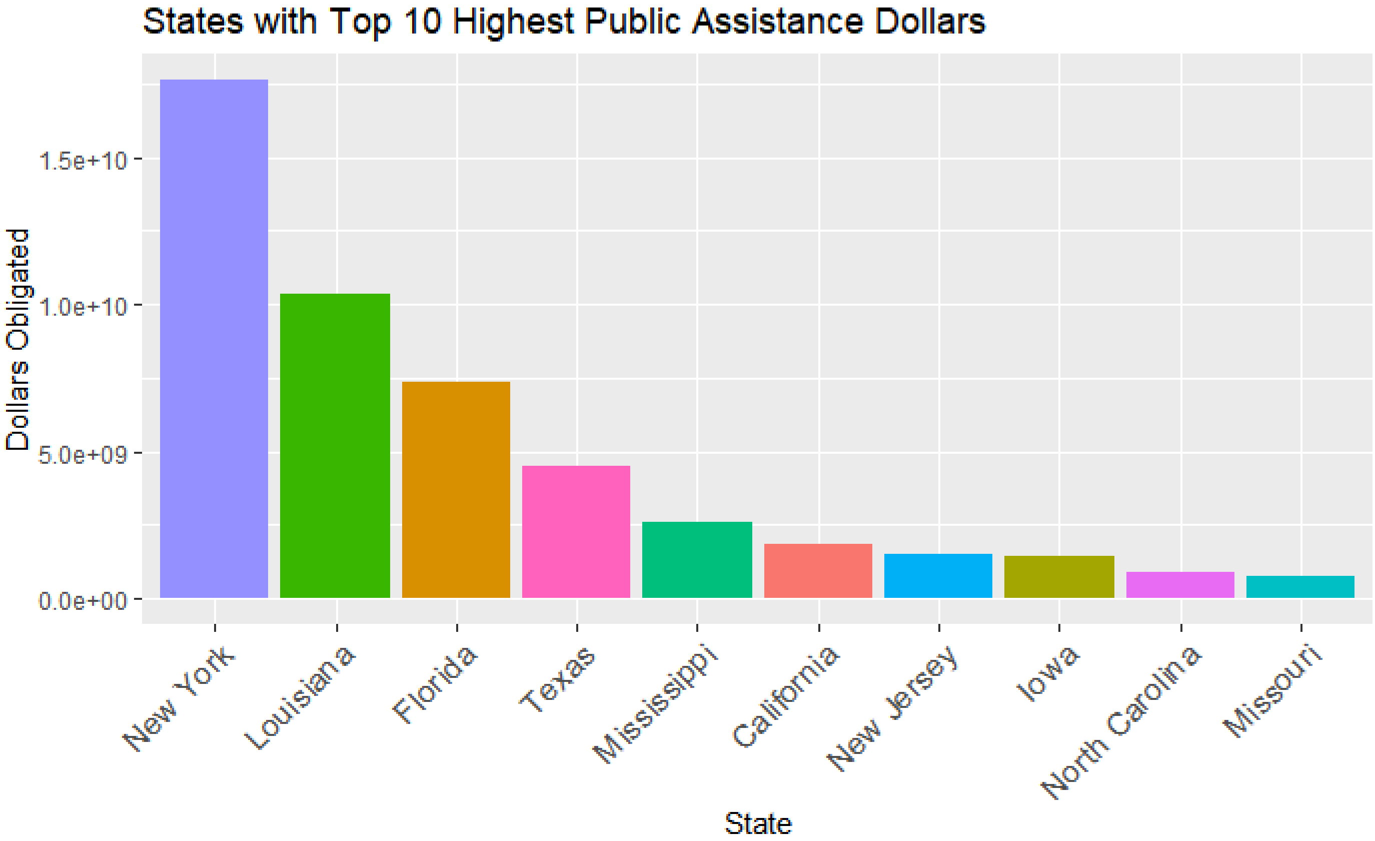
```
# Conduct correlation testing to evaluate impact of diversity and
income on California public assistance funding
cor.test(cal$county_obligated_perCap, cal$diversity)
    Pearson's product-moment correlation

data:  x and y
t = -3.6363, df = 56, p-value = 0.0006028
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 -0.6248171 -0.2015095
sample estimates:
       cor
-0.4370569

cor.test(cal$county_obligated_perCap, cal$Income)

    Pearson's product-moment correlation

data:  x and y
t = -2.1743, df = 56, p-value = 0.03392
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 -0.50119360 -0.02233219
```

```
sample estimates:
       cor
-0.2790185

# Create a heatmap to visualize California correlation matrix
calheat <- cal %>% select(county_obligated_perCap, Income, Poverty,
diversity, county_disaster_count, county_disasters_perCap)
Adding missing grouping variables: `county`, `state.x`
calheat$state.x <- NULL
calheat$county <- NULL

cormatrix <- melt(round(cor(calheat), 2))
ggplot(data = cormatrix, aes(x=Var1, y=Var2, fill=value)) +
geom_tile() + scale_fill_gradient2(low = "blue", high = "red", mid =
"white", midpoint = 0, limit = c(-1,1), space = "Lab",
name="Pearson\nCorrelation") + theme(axis.text.x = element_text(angle
= 45, vjust = 1, size = 12, hjust = 1)) + labs(title = "California
Correlation Heat Map")
```

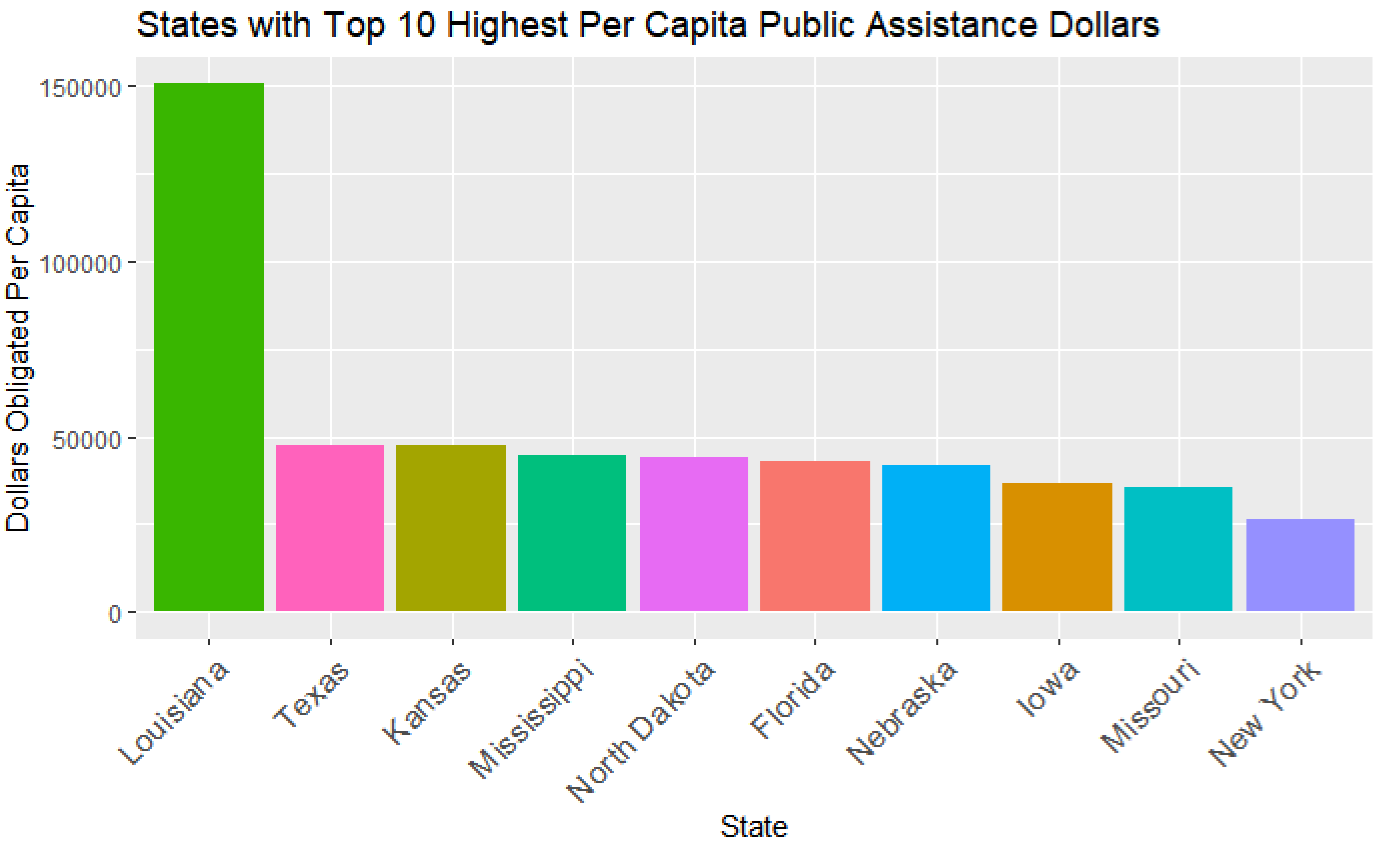


```
# Compute states with highest overall public assistance dollars
obligated since 2000
total_ob <- mappingdata %>% group_by(state.x) %>%
summarise(total_obligated = sum(countyObligated)) %>%
arrange(desc(total_obligated)) %>% head(10)
```

```
ggplot(total_ob, aes(x=reorder(state.x, -total_obligated),
y=total_obligated, fill=state.x)) + geom_bar(stat='identity') +
theme(axis.text.x = element_text(angle = 45, vjust = 1, size = 12,
hjust = 1)) + labs(title = "States with Top 10 Highest Public
Assistance Dollars", x = "State", y="Dollars Obligated") +
theme(legend.position = "none")
```



```
# Compute states with highest per capita public assistance funding
percap_ob <- mappingdata %>% group_by(state.x) %>%
summarise(percap_obligated = sum(county_obligated_perCap)) %>%
arrange(desc(percap_obligated)) %>% head(10)
ggplot(percap_ob, aes(x=reorder(state.x, -percap_obligated),
y=percap_obligated, fill=state.x)) + geom_bar(stat='identity') +
theme(axis.text.x = element_text(angle = 45, vjust = 1, size = 12,
hjust = 1)) + labs(title = "States with Top 10 Highest Per Capita
Public Assistance Dollars", x = "State", y="Dollars Obligated Per
Capita") + theme(legend.position = "none")
```


States with Top 10 Highest Per Capita Public Assistance Dollars
Dollars Obligated Per Capita
150000
100000
50000
0
Louisiana
Texas
Kansas
Mississippi
North Dakota
Florida
Nebraska
Iowa
Missouri
New York
State