Editting Data

This is an R Markdown Notebook. When you execute code within the notebook, the results appear beneath the code.

Try executing this chunk by clicking the *Run* button within the chunk or by placing your cursor inside it and pressing *Cmd+Shift+Enter*.

```
loading_zipdata <- "~/repos/Diversity-Richness/Zip.Code.Datasets/zip_data_unedited_nolabels.csv" #makin
Unedited_zipdata <- read.csv(loading_zipdata) #loading the census data
```

```
reduced_collumns_zip <- c(1:2, 33:34, 42, 47:146) #removing unnecessary columns


Zipcode_Census_data<- Unedited_zipdata[ , reduced_collumns_zip] #make a new table without unnecessary c

colnames(Zipcode_Census_data) <- c("FIPS", "Name", "3_Digit_Tabulation", "5_Digit_Tabulation" , "Area_Na

#writing usable column names
```

```
write.csv(Zipcode_Census_data, file = "~/repos/Diversity-Richness/Zip.Code.Datasets/Zipcode_Census_data
#saving the table as a csv file
```

```
library(tidyverse)
```

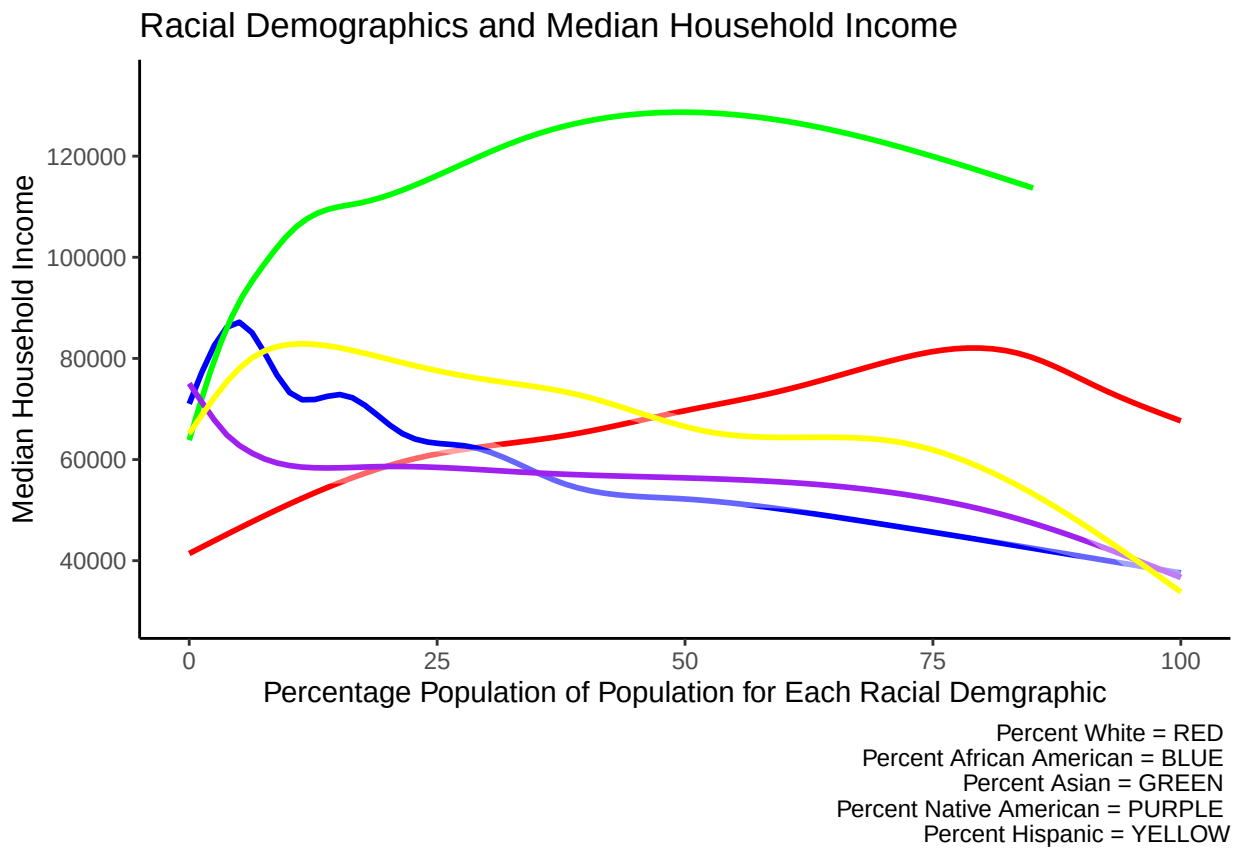
```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.4.3     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.0
## v purrr     1.0.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(ggplot2)

values <- c("Percent White" = "red","Percent African American" = "blue",  "Percent Asian" = "green", "Pe
text_needed <-
"Percent White = RED
Percent African American = BLUE
Percent Asian = GREEN
Percent Native American = PURPLE
Percent Hispanic = YELLOW"

Zipcode_Census_data %>%
  group_by()
```

```
## # A tibble: 33,774 x 105
##     FIPS Name        `3_Digit_Tabulation` `5_Digit_Tabulation` Area_Name
##    <int> <chr>                      <int>                <int> <chr>
##  1   601 ZCTA5 00601                    6                  601 ZCTA5 00601
##  2   602 ZCTA5 00602                    6                  602 ZCTA5 00602
##  3   603 ZCTA5 00603                    6                  603 ZCTA5 00603
##  4   606 ZCTA5 00606                    6                  606 ZCTA5 00606
##  5   610 ZCTA5 00610                    6                  610 ZCTA5 00610
##  6   611 ZCTA5 00611                    6                  611 ZCTA5 00611
##  7   612 ZCTA5 00612                    6                  612 ZCTA5 00612
##  8   616 ZCTA5 00616                    6                  616 ZCTA5 00616
##  9   617 ZCTA5 00617                    6                  617 ZCTA5 00617
## 10   622 ZCTA5 00622                    6                  622 ZCTA5 00622
## # i 33,764 more rows
## # i 100 more variables: Total_Population <int>, Population_Density <dbl>,
## #   Area <dbl>, Total_Population1 <int>, Total_Population_Male <int>,
## #   Total_Population_Female <int>, X_Total_Population_Male <dbl>,
## #   X_Total_Population_Female <dbl>, Total_Population2 <int>,
## #   Total_Population_Under_5_Years <int>, Total_Population_5_to_9_Years <int>,
## #   Total_Population_10_to_14_Years <int>, ...
```

```
ggplot(Zipcode_Census_data) +
  geom_smooth(aes(x = X_Total_Population_White_Alone, y = Median_Household_Income), fill ="white", colo
  geom_smooth(aes(x = X_Total_Population_Black_or_African_American_Alone, y = Median_Household_Income),
  geom_smooth(aes(x = X_Total_Population_Asian_Alone, y = Median_Household_Income), fill = "white", col
  geom_smooth(aes(x = X_Total_Population_American_Indian_And_Native_Alaskan_Alone, y = Median_Household
  geom_smooth(aes(x = X_Hispanic, y = Median_Household_Income), fill = "white", color = "yellow", show.
  labs(x = "Percentage Population of Population for Each Racial Demgraphic", y = "Median Household Inco
         title = "Racial Demographics and Median Household Income", caption = text_needed) +
  theme_classic()
```

```
## `geom_smooth()` using method = 'gam' and formula = 'y ~ s(x, bs = "cs")'

## Warning: Removed 3153 rows containing non-finite values (`stat_smooth()`).

## `geom_smooth()` using method = 'gam' and formula = 'y ~ s(x, bs = "cs")'

## Warning: Removed 3153 rows containing non-finite values (`stat_smooth()`).

## `geom_smooth()` using method = 'gam' and formula = 'y ~ s(x, bs = "cs")'

## Warning: Removed 3153 rows containing non-finite values (`stat_smooth()`).

## `geom_smooth()` using method = 'gam' and formula = 'y ~ s(x, bs = "cs")'

## Warning: Removed 3153 rows containing non-finite values (`stat_smooth()`).

## `geom_smooth()` using method = 'gam' and formula = 'y ~ s(x, bs = "cs")'

## Warning: Removed 3153 rows containing non-finite values (`stat_smooth()`).
```




```
#May want to add further descriptors or learn how to add multiple lines
```

(ref:Income_By_Racial_Demographic_Figure) Income By Racial Demographic Figure

```
library(tidyverse)
library(ggplot2)

ggplot(Zipcode_Census_data) +
  geom_smooth(aes(x = X_Total_Population_White_Alone, y = Median_Household_Income, color = "#A1C9F4"), 
  geom_smooth(aes(x = X_Total_Population_Black_or_African_American_Alone, y = Median_Household_Income, 
  geom_smooth(aes(x = X_Total_Population_Asian_Alone, y = Median_Household_Income, color = "#8DE5A1"), 
  geom_smooth(aes(x = X_Total_Population_American_Indian_And_Native_Alaskan_Alone, y = Median_Household
  geom_smooth(aes(x = X_Hispanic, y = Median_Household_Income, color = "#FF9F9B"), fill = "#FF9F9B", alp
  scale_color_manual(values = c( "#8DE5A1" = "#8DE5A1","#A1C9F4" = "#A1C9F4","#D0BBFF" = "#D0BBFF", "#F
                     name = "Race",
                     labels = c("Asian", "White", "Native American/Alaskan", "Hispanic", "Black")) +
  labs(x = "Percentage Population of Population for Each Racial Demgraphic", y = "Median Household Incor
```

```
## `geom_smooth()` using method = 'gam' and formula = 'y ~ s(x, bs = "cs")'

## Warning: Removed 3153 rows containing non-finite values (`stat_smooth()`).

## `geom_smooth()` using method = 'gam' and formula = 'y ~ s(x, bs = "cs")'

## Warning: Removed 3153 rows containing non-finite values (`stat_smooth()`).
```

```
## `geom_smooth()` using method = 'gam' and formula = 'y ~ s(x, bs = "cs")'

## Warning: Removed 3153 rows containing non-finite values (`stat_smooth()`).

## `geom_smooth()` using method = 'gam' and formula = 'y ~ s(x, bs = "cs")'

## Warning: Removed 3153 rows containing non-finite values (`stat_smooth()`).

## `geom_smooth()` using method = 'gam' and formula = 'y ~ s(x, bs = "cs")'

## Warning: Removed 3153 rows containing non-finite values (`stat_smooth()`).
```

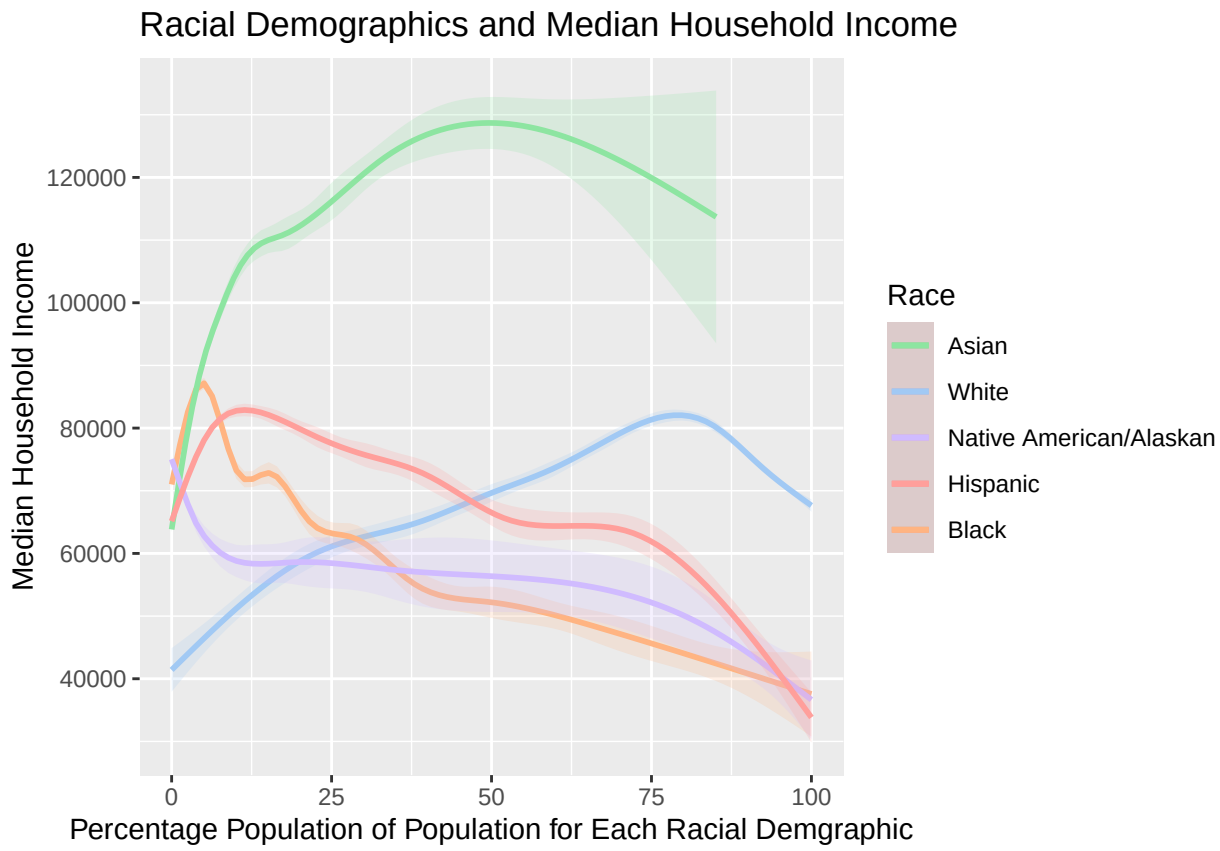


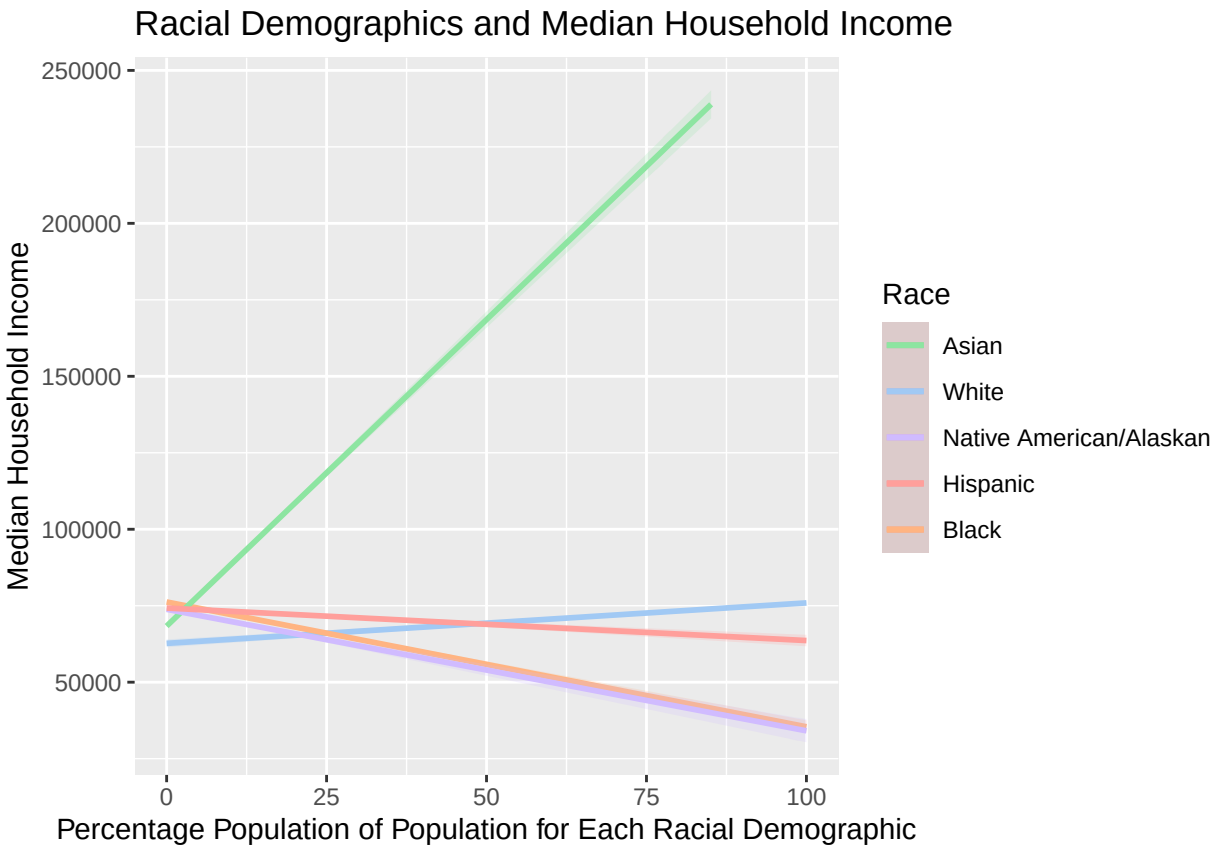
Figure 1: (ref:Income_By_Racial_Demographic_Figure)

```
#Don't know how to correct red shading on key
```

```
library(tidyverse)
library(ggplot2)


ggplot(Zipcode_Census_data) +
  geom_smooth(aes(x = X_Total_Population_White_Alone, y = Median_Household_Income, color = "#A1C9F4"), 
  geom_smooth(aes(x = X_Total_Population_Black_or_African_American_Alone, y = Median_Household_Income,
```

```
  geom_smooth(aes(x = X_Total_Population_Asian_Alone, y = Median_Household_Income, color = "#8DE5A1"), 
  geom_smooth(aes(x = X_Total_Population_American_Indian_And_Native_Alaskan_Alone, y = Median_Household_
  geom_smooth(aes(x = X_Hispanic, y = Median_Household_Income, color = "#FF9F9B"), fill = "#FF9F9B", alp
  scale_color_manual(values = c( "#8DE5A1" = "#8DE5A1","#A1C9F4" = "#A1C9F4","#D0BBFF" = "#D0BBFF", "#FF
                     name = "Race",
                     labels = c("Asian", "White", "Native American/Alaskan", "Hispanic", "Black")) +
  labs(x = "Percentage Population of Population for Each Racial Demographic", y = "Median Household Inco
```

```
## `geom_smooth()` using formula = 'y ~ x'

## Warning: Removed 3153 rows containing non-finite values (`stat_smooth()`).

## `geom_smooth()` using formula = 'y ~ x'

## Warning: Removed 3153 rows containing non-finite values (`stat_smooth()`).

## `geom_smooth()` using formula = 'y ~ x'

## Warning: Removed 3153 rows containing non-finite values (`stat_smooth()`).

## `geom_smooth()` using formula = 'y ~ x'

## Warning: Removed 3153 rows containing non-finite values (`stat_smooth()`).

## `geom_smooth()` using formula = 'y ~ x'

## Warning: Removed 3153 rows containing non-finite values (`stat_smooth()`).
```

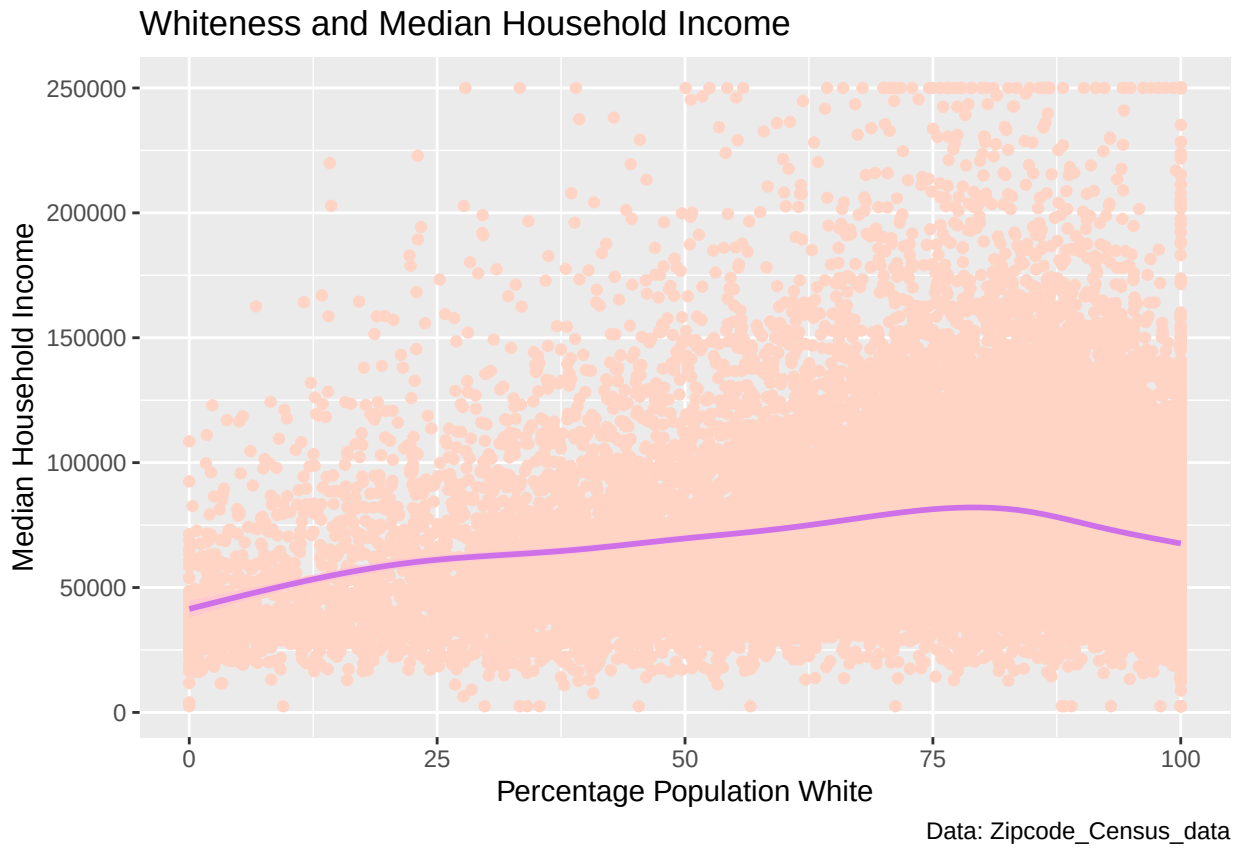


```
library(tidyverse)
library(ggplot2)

ggplot(Zipcode_Census_data) +
  geom_jitter(aes(x = X_Total_Population_White_Alone, y = Median_Household_Income), fill ="black", colo
   geom_smooth(aes(x = X_Total_Population_White_Alone, y = Median_Household_Income), fill ="#ffb5e2", c
  labs(x = "Percentage Population White", y = "Median Household Income",
         title = "Whiteness and Median Household Income", caption = "Data: Zipcode_Census_data")
```

```
## `geom_smooth()` using method = 'gam' and formula = 'y ~ s(x, bs = "cs")'
```

```
## Warning: Removed 3153 rows containing non-finite values (`stat_smooth()`).
```

```
## Warning: Removed 3153 rows containing missing values (`geom_point()`).
```




```
#May want to add further descriptors or learn how to add multiple lines
```

```
Zipcode_with_Zones <- Zipcode_Census_data %>%
  mutate(Zipcode_Zone= case_when(
   substr(Name, 7, 8) == "00" ~ "Puerto Rico",
   substr(Name, 7, 7) == "0" ~ "North East",
    substr(Name, 7, 7) == "1" ~ "New York Region (Lower North East)",
    substr(Name, 7, 7) == "2" ~ "Central East Coast",
```

```
    substr(Name, 7, 7) == "3" ~ "South East",
    substr(Name, 7, 7) == "4" ~ "Great Lakes",
    substr(Name, 7, 7) == "5" ~ "Northern Midwest",
    substr(Name, 7, 7) == "6" ~ "Central Interior",
    substr(Name, 7, 7) == "7" ~ "Texas Region",
    substr(Name, 7, 7) == "8" ~ "Western Interior",
    substr(Name, 7, 7) == "9" ~ "West Coast (and Hawaii/Alaska)",
    substr(Name, 7, 8) == "00" ~ "Puerto Rico",
    TRUE ~ "Other"  # Default case
  ), .after = Area_Name)

Zipcode_with_States <- Zipcode_with_Zones %>%
  mutate(State_Territory= case_when(
    between(as.integer(substr(Name, 7, 9)), 039, 049) ~ "ME",
    between(as.integer(substr(Name, 7, 9)), 030, 038) ~ "NH",
    between(as.integer(substr(Name, 7, 9)), 010, 027) ~ "MA",
    between(as.integer(substr(Name, 7, 9)), 028, 029) ~ "RI",
    between(as.integer(substr(Name, 7, 9)), 150, 196) ~ "PA",
    between(as.integer(substr(Name, 7, 9)), 197, 199) ~ "DE",
    between(as.integer(substr(Name, 7, 9)), 206, 219) ~ "MD",
    between(as.integer(substr(Name, 7, 9)), 200, 205) ~ "DC",
    between(as.integer(substr(Name, 7, 9)), 220, 246) ~ "VA",
    between(as.integer(substr(Name, 7, 9)), 247, 269) ~ "WV",
    between(as.integer(substr(Name, 7, 9)), 386, 399) ~ "MS",
    between(as.integer(substr(Name, 7, 9)), 370, 385) ~ "TN",
    between(as.integer(substr(Name, 7, 9)), 700, 715) ~ "LA",
    between(as.integer(substr(Name, 7, 9)), 716, 729) ~ "AR",
    between(as.integer(substr(Name, 7, 9)), 550, 567) ~ "MN",
    between(as.integer(substr(Name, 7, 9)), 820, 831) ~ "WY",
    between(as.integer(substr(Name, 7, 9)), 832, 839) ~ "ID",
    between(as.integer(substr(Name, 7, 9)), 870, 884) ~ "NM",
    between(as.integer(substr(Name, 7, 9)), 889, 899) ~ "NV",
    between(as.integer(substr(Name, 7, 9)), 900, 961) ~ "CA",
    between(as.integer(substr(Name, 7, 9)), 980, 994) ~ "WA",
    between(as.integer(substr(Name, 7, 9)), 967, 968) ~ "HI",
    between(as.integer(substr(Name, 7, 9)), 995, 999) ~ "AK",
    between(as.integer(substr(Name, 7, 9)), 962, 966) ~ "AP",
    between(as.integer(substr(Name, 7, 9)), 006, 009) ~ "PR/VI",
    between(as.integer(substr(Name, 7, 8)), 10, 14) ~ "NY",
    between(as.integer(substr(Name, 7, 8)), 07, 08) ~ "NJ",
    between(as.integer(substr(Name, 7, 8)), 27, 28) ~ "NC",
    between(as.integer(substr(Name, 7, 8)), 30, 31) ~ "GA",
    between(as.integer(substr(Name, 7, 8)), 32, 34) ~ "FL",
    between(as.integer(substr(Name, 7, 8)), 35, 36) ~ "AL",
    between(as.integer(substr(Name, 7, 8)), 40, 42) ~ "KY",
    between(as.integer(substr(Name, 7, 8)), 43, 45) ~ "OH",
    between(as.integer(substr(Name, 7, 8)), 46, 47) ~ "IN",
    between(as.integer(substr(Name, 7, 8)), 48, 49) ~ "MI",
    between(as.integer(substr(Name, 7, 8)), 50, 12) ~ "IA",
    between(as.integer(substr(Name, 7, 8)), 53, 54) ~ "WI",
    between(as.integer(substr(Name, 7, 8)), 60, 62) ~ "IL",
    between(as.integer(substr(Name, 7, 8)), 63, 65) ~ "MO",
    between(as.integer(substr(Name, 7, 8)), 66, 67) ~ "KS",
```

```
    between(as.integer(substr(Name, 7, 8)), 68, 69) ~ "NE",
    between(as.integer(substr(Name, 7, 8)), 73, 74) ~ "OK",
    between(as.integer(substr(Name, 7, 8)), 75, 79) ~ "TX",
    between(as.integer(substr(Name, 7, 8)), 80, 81) ~ "CO",
    between(as.integer(substr(Name, 7, 8)), 85, 86) ~ "AZ",
    str_detect(Name, "05") ~ "VT",
    str_detect(Name, "06") ~ "CT",
    str_detect(Name, "29") ~ "SC",
    str_detect(Name, "57") ~ "SD",
    str_detect(Name, "58") ~ "ND",
    str_detect(Name, "59") ~ "MT",
    str_detect(Name, "84") ~ "UT",
    str_detect(Name, "97") ~ "OR",
    str_detect(Name, "09") ~ "AE",
    str_detect(Name, "340") ~ "AA",
    str_detect(Name, "969") ~ "PW/FM/MH/MP/GU",
    str_detect(Name, "96799") ~ "AS",
    TRUE ~ "Other"  # Default case
  ), .after = Zipcode_Zone)

write.csv(Zipcode_with_States, file = "~/repos/Diversity-Richness/Zip.Code.Datasets/Zipcode_data_with_A
```

(ref:Whiteness_By_Region_Boxplot) Whiteness By Region Boxplot

```
# Create the boxplot
ggplot(Zipcode_with_Zones, aes(x = Zipcode_Zone, y = X_Total_Population_White_Alone, fill = Zipcode_Zon
  geom_boxplot(outlier.shape = NA) +
  labs(x = "ZIP Code Zones", y = "Percentage White",
       title = "Percentage of Population White by Zip Code Region", fill = "Zone Names") +
  theme(axis.text.x = element_blank())
```

```
## Warning: Removed 587 rows containing non-finite values (`stat_boxplot()`).
```

(ref:Demographics_By_Region_Table) Demographics By Region Table

```
Zipcode_with_States %>%
  group_by(Zipcode_Zone) %>%
  summarize(
    mean.percent.white = mean(X_Total_Population_White_Alone, na.rm = T),
    sd.percent.white = sd(X_Total_Population_White_Alone, na.rm = T),
    mean.percent.black = mean(X_Total_Population_Black_or_African_American_Alone, na.rm = T),
    sd.percent.black = sd(X_Total_Population_Black_or_African_American_Alone, na.rm = T),
    mean.percent.asian = mean(X_Total_Population_Asian_Alone, na.rm = T),
    sd.percent.asian = sd(X_Total_Population_Asian_Alone, na.rm = T),
    mean.percent.native.american = mean(X_Total_Population_American_Indian_And_Native_Alaskan_Alone, na
    sd.percent.native.american = sd(X_Total_Population_American_Indian_And_Native_Alaskan_Alone, na.rm =
    mean.percent.hispanic = mean(X_Hispanic, na.rm = T),
    sd.percent.hispanic = sd(X_Hispanic, na.rm = T),
    n.per.region = n()
  ) %>%
  knitr::kable(col.names = c("ZIP Code Region", "Percent White",    "SD White", "Percent Black",    "SD
               digits = 2,
```

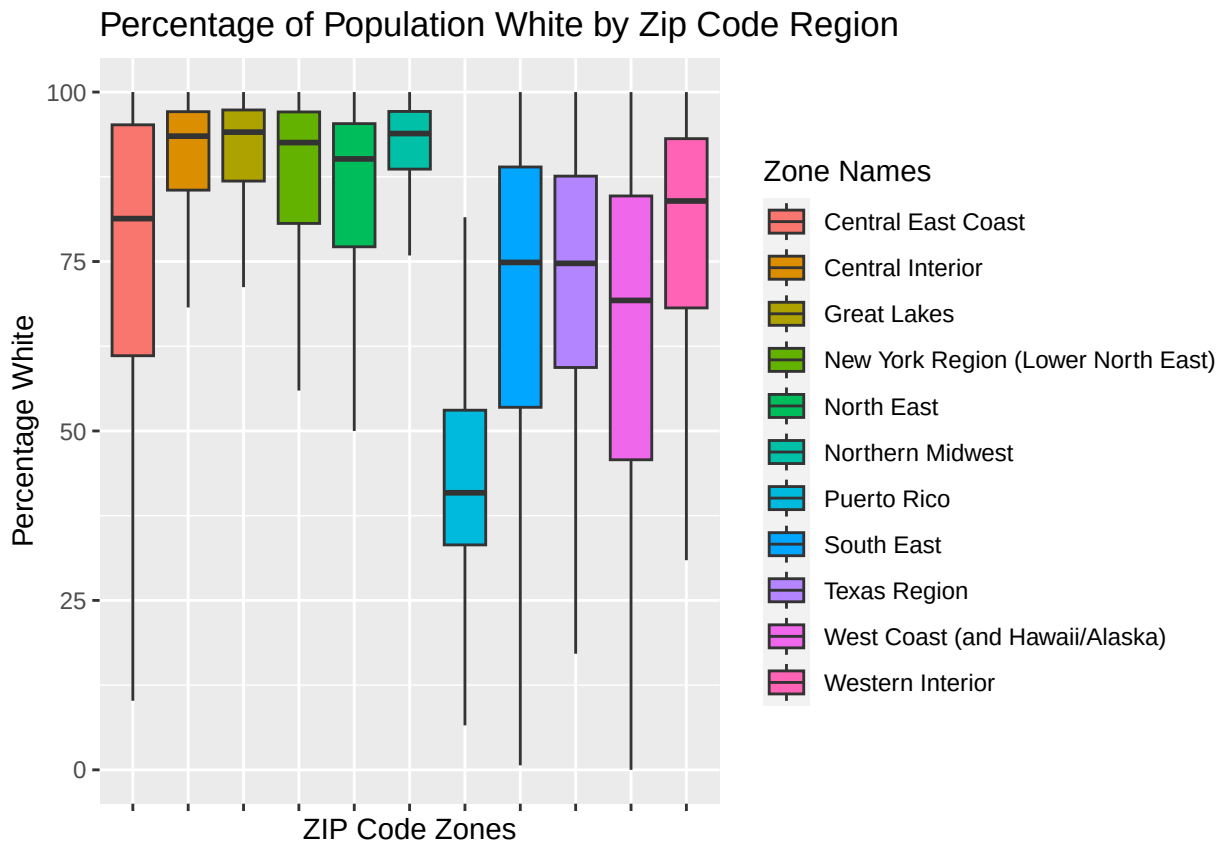


Figure 2: (ref:Whiteness_By_Region_Boxplot)

```
              align = "lrlrlrlrlrlr",
              caption = "Average ZIP Code Demographics by Region") %>%
kableExtra::kable_styling(position = "left", latex_options = c("scale_down", "hold_position")) %>%
 kableExtra::column_spec(seq(2, 12, 2), bold = TRUE)
```

```
## Warning in styling_latex_scale(out, table_info, "down"): Longtable cannot be
## resized.
```

Table 1: Average ZIP Code Dem

ZIP Code Region	Percent White	SD White	Percent Black	SD Black	Percent Asian	SD As
Central East Coast	**75.02**	23.79	**15.97**	20.58	**1.85**	4.47
Central Interior	**87.65**	16.50	**4.12**	12.27	**1.34**	3.58
Great Lakes	**88.36**	16.24	**5.32**	13.45	**1.05**	2.67
New York Region (Lower North East)	**84.84**	19.50	**5.30**	12.05	**2.59**	6.03
North East	**83.01**	18.66	**4.39**	9.62	**3.95**	7.19
Northern Midwest	**89.37**	15.50	**1.48**	5.30	**1.05**	2.77
Puerto Rico	**43.15**	18.15	**9.12**	9.05	**0.23**	0.45
South East	**68.73**	24.74	**21.41**	24.39	**1.55**	3.32
Texas Region	**71.48**	20.73	**11.12**	17.71	**1.62**	3.89
West Coast (and Hawaii/Alaska)	**63.64**	26.01	**3.04**	5.78	**7.91**	12.41
Western Interior	**76.22**	24.31	**1.77**	4.17	**1.43**	2.87

I will now refer to Figure @ref(fig:Income_By_Racial_Demographic_Figure), Figure @ref(figWhiteness_By_Region_Boxplo and Table @ref(tabDemographics_By_Region_Table).