Racial Bias in Gifted and Talented Programs

Data Science I (STAT 301-1)

Alexandra Chang

Contents

Introduction

Overview

For my final project, I conducted an exploratory data analysis (EDA) on data from the Civil Rights Data Collection (CRDC) — a biennial study conducted by the Office for Civil Rights within the U.S. Department of Education — for the 2017-2018 school year. In this EDA, I sought to visualize the racial disparities that exist in Gifted and Talented (GAT) programs across the United States, specifically the underrepresentation of BIPOC students in GAT programs. In addition, I investigate how these racial disparities differ by state.

Datasets

The CRDC collects "a variety of information including student enrollment and educational programs and services, most of which is disaggregated by race/ethnicity, sex, limited English proficiency, and disability." For this project, I used data sets from the compressed public-use data file that is published on the Department of Education's website. This file contained the entire 2017-2018 CRDC in CSV format, but I specifically used the "Enrollment" and "Gifted and Talented" data sets. This EDA utilizes two data sets: `Enrollment.csv` and `Gifted and Talented.csv`. The data sets were respectively renamed `enroll_raw` and `gat_raw` after import.

Data Import

Load packages

First, I loaded the necessary packages for conducting this EDA: `tidyverse` and `ggplot2`.

```
# Loading package(s)
library(tidyverse)
library(ggplot2)
```

Data import

I started this project with two unprocessed data sets to tidy: `Enrollment.csv` and `Gifted and Talented.csv`. After importing both, I stored `Enrollment.csv` as `enroll_raw` and `Gifted and Talented.csv` as `gat_raw`.

```
enroll_raw <- read.csv(file = "data/unprocessed/Enrollment.csv")
gat_raw <- read.csv(file = "data/unprocessed/Gifted and Talented.csv")
```

Relational data: preparation

In Chapter 13 of R for Data Science, we learned about relational data, which entails combining and exploring multiple data sets. In order to establish a relationship between `enroll_raw` and `gat_raw`, I first identified the key that can join the two tables. Within this data set, a primary key (`COMBOKEY`) already existed to identify each observation, which allowed me to easily use the `inner_join()` function from the `dplyr` package; each unique `COMBOKEY` represents a different school in the United States.

```
intersect(names(enroll_raw),
          names(gat_raw)) # identify key
```

```
## [1] "LEA_STATE"      "LEA_STATE_NAME" "LEAID"          "LEA_NAME"
## [5] "SCHID"          "SCH_NAME"       "COMBOKEY"       "JJ"
```

```
enroll_raw %>%
  count(COMBOKEY) %>%
  filter(n > 1) # confirm key
```

```
## [1] COMBOKEY n
## <0 rows> (or 0-length row.names)
```

```
gat_raw %>%
  count(COMBOKEY) %>%
  filter(n > 1) # confirm key
```

```
## [1] COMBOKEY n
## <0 rows> (or 0-length row.names)
```

Data transformation

Next, I transformed `gat_raw` to filter out schools that do not have GAT programs, leaving only schools that *do* have GAT programs.

```
gat_raw <- gat_raw %>%
  filter(SCH_GT_IND == "Yes")
```

In order to make later analyses easier, I renamed the variables more intuitively. This required cross-checking the variable names with the provided code book, which can be located by the path `misc/crdc notes/2017-18 CRDC File Structure.xlsx`. Additionally, I collapsed the race and ethnicity categories "American Indian/Alaska Native," "Native Hawaiian/Pacific Islander," and "Two or More Races" into a single

category: "other race." I also constructed a BIPOC category by adding the Asian, Black, Hispanic, and other race populations. This left me with six race and ethnicity categories: `Asian`, `Black`, `Hispanic`, `white`, `other`, and `BIPOC`. Finally, I filtered out all-male and all-female schools from `enroll`.

```
GAT <- gat_raw %>%
  as_tibble() %>%
  mutate(gat_raw,
         state = LEA_STATE,
         gat_hisp_m = SCH_GTENR_HI_M, gat_hisp_f = SCH_GTENR_HI_F,
         gat_asian_m = SCH_GTENR_AS_M, gat_asian_f = SCH_GTENR_AS_F,
         gat_black_m = SCH_GTENR_BL_M, gat_black_f = SCH_GTENR_BL_F,
         gat_white_m = SCH_GTENR_WH_M, gat_white_f = SCH_GTENR_WH_F,
         gat_other_m = SCH_GTENR_AM_M + SCH_GTENR_HP_M + SCH_GTENR_TR_M,
         gat_other_f = SCH_GTENR_AM_F + SCH_GTENR_HP_F + SCH_GTENR_TR_F,
         gat_bipoc_m = gat_hisp_m + gat_asian_m + gat_black_m + gat_other_m,
         gat_bipoc_f = gat_hisp_f + gat_asian_f + gat_black_f + gat_other_f,
         gat_male = gat_hisp_m + gat_asian_m + gat_black_m + gat_white_m + gat_other_m,
         gat_female = gat_hisp_f + gat_asian_f + gat_black_f + gat_white_f + gat_other_f,
         gat_total = gat_male + gat_female) %>%
  select(COMBOKEY, state, gat_asian_m, gat_asian_f,
         gat_black_m, gat_black_f, gat_hisp_m, gat_hisp_f,
         gat_white_m, gat_white_f, gat_other_m, gat_other_f,
         gat_bipoc_m, gat_bipoc_f,
         gat_male, gat_female, gat_total)

enroll <- enroll_raw %>%
  as_tibble() %>%
  mutate(enroll_raw,
         state = LEA_STATE,
         hisp_m = SCH_ENR_HI_M, hisp_f = SCH_ENR_HI_F,
         asian_m = SCH_ENR_AS_M, asian_f = SCH_ENR_AS_F,
         black_m = SCH_ENR_BL_M, black_f = SCH_ENR_BL_F,
         white_m = SCH_ENR_WH_M, white_f = SCH_ENR_WH_F,
         other_m = SCH_ENR_AM_M + SCH_ENR_HP_M + SCH_ENR_TR_M,
         other_f = SCH_ENR_AM_F + SCH_ENR_HP_F + SCH_ENR_TR_F,
         bipoc_m = hisp_m + asian_m + black_m + other_m,
         bipoc_f = hisp_f + asian_f + black_f + other_f,
         male = hisp_m + asian_m + black_m + white_m + other_m,
         female = hisp_f + asian_f + black_f + white_f + other_f,
         total = male + female) %>%
  filter(male > 0,
         female > 0) %>%
  select(COMBOKEY, state, asian_m, asian_f,
         black_m, black_f, hisp_m, hisp_f,
         white_m, white_f, other_m, other_f,
         bipoc_m, bipoc_f, male, female, total)
```

Analysis I: Representation of students by race and ethnicity in GAT programs

Data transformation

First, I collapsed the population variables — which were disaggregated by sex — by race for both `GAT` and `enroll`. These data frames were renamed `GAT_RE` — which contains the variables `COMBOKEY`, `state`, `gat_asian`, `gat_black`, `gat_hisp`, `gat_white`, `gat_other`, `gat_bipoc`, and `gat_total` — and `enroll_RE` — which contains the variables `COMBOKEY`, `asian`, `black`, `hisp`, `white`, `other`, `bipoc`, and `total`.

```
GAT_RE <- GAT %>%
  mutate(GAT,
         gat_asian = gat_asian_m + gat_asian_f,
         gat_black = gat_black_m + gat_black_f,
         gat_hisp = gat_hisp_m + gat_hisp_f,
         gat_white = gat_white_m + gat_white_f,
         gat_other = gat_other_m + gat_other_f,
         gat_bipoc = gat_asian + gat_black + gat_hisp + gat_other) %>%
  filter(gat_total > 0) %>%
  select(COMBOKEY, state,
         gat_asian, gat_black, gat_hisp,
         gat_white, gat_other, gat_bipoc,
         gat_total)

enroll_RE <- enroll %>%
  mutate(enroll,
         asian = asian_m + asian_f,
         black = black_m + black_f,
         hisp = hisp_m + hisp_f,
         white = white_m + white_f,
         other = other_m + other_f,
         bipoc = asian + black + hisp + other) %>%
  select(COMBOKEY,
         asian, black, hisp, white, other, bipoc,
         total)
```

Relational data: `inner_join()`

Then, I joined the two data sets using a mutating join, `inner_join()`. For this project, `inner_join()` matched pairs of observations in `GAT_RE` and `enroll_RE` by `COMBOKEY`, creating one pooled data set `RE` — which contains the variables `COMBOKEY`, `state`, `gat_asian`, `asian`, `gat_black`, `black`, `gat_hisp`, `hisp`, `gat_white`, `white`, `gat_other`, `other`, `gat_bipoc`, `bipoc`, `gat_total`, and `total`.

```
RE <- enroll_RE %>%
  inner_join(GAT_RE,
             by = "COMBOKEY") %>%
  select(COMBOKEY, state, gat_asian, asian,
         gat_black, black,
         gat_hisp, hisp,
         gat_white, white,
         gat_other, other,
         gat_bipoc, bipoc,
         gat_total, total)
```

Tidy data

The most difficult part in tidying this data set was creating a functional race column. The first step was to use the `unite()` function in order collapse the two populations that were separated by race and ethnicity: `gat_RACE` and `RACE`. By uniting the two variables before pivoting the table, I could create a single race variable — which I later convert into a factor — as opposed to two variables, which would not constitute a tidy data set. Next, using `pivot_longer()`, I pivoted the table, creating two new variables: `race` and `count`. `count` hosted the united population variables from the untidy data frame `RE`, but then separated `count` into `gat_count` — the number of students in the GAT program — and `race_count` — the total number of students at the school disaggregated by `race`. Although it might technically be redundant to name this new variable `race_count` as opposed to just `count`, I found that this naming convention was much clearer in distinguishing `race_count` from `gat_count` and `total`.

Because using `unite()` and `separate()` converts variables into character vectors, I needed to transform `gat_count` and `race_count` back into numeric vectors This was necessary in order to create visualizations and calculate proportions using my tidied data. Additionally, I filtered out schools with fewer than 10 students, and I arranged the columns into an intuitive order using `select()`.

```
RE <- RE %>%
  unite(asian, gat_asian, asian, sep = "_") %>%
  unite(black, gat_black, black, sep = "_") %>%
  unite(hisp, gat_hisp, hisp, sep = "_") %>%
  unite(white, gat_white, white, sep = "_") %>%
  unite(other, gat_other, other, sep = "_") %>%
  unite(bipoc, gat_bipoc, bipoc, sep = "_") %>%
  pivot_longer(c("asian", "black", "hisp", "white", "other", "bipoc"),
               names_to = "race", values_to = "count") %>%
  separate(count, into = c("gat_count", "race_count"), sep = "_") %>%
  mutate(gat_count = as.numeric(gat_count),
         race_count = as.numeric(race_count)) %>% # typeof() = numeric
  filter(total >= 10) %>% # school must have more than 10 students
  select(COMBOKEY, state, race, gat_count, race_count,
         gat_total, total) # select columns
```

While `RE` is a tidy data set, I decided to also create a grouped data frame as a second way of viewing this data set. `RE_nest` is a nested data frame that allows you to view population data by school. Below, I printed an example using the first element of `GAT and race populations` for the school whose `COMBOKEY` is `010000500879`.

```
RE_nest <- RE %>%
  group_by(state, COMBOKEY) %>%
  nest() %>%
  mutate("GAT and race populations" = data) %>%
  select(COMBOKEY, state, "GAT and race populations") # view data by school

RE_nest
```

```
## # A tibble: 55,732 x 3
## # Groups:   COMBOKEY, state [55,732]
##    COMBOKEY     state `GAT and race populations`
##    <chr>        <chr> <list>
##  1 010000500879 AL    <tibble [6 x 5]>
##  2 010000500889 AL    <tibble [6 x 5]>
##  3 010000600193 AL    <tibble [6 x 5]>
##  4 010000600877 AL    <tibble [6 x 5]>
##  5 010000600880 AL    <tibble [6 x 5]>
```

```
##  6 010000601685 AL    <tibble [6 x 5]>
##  7 010000601812 AL    <tibble [6 x 5]>
##  8 010000602209 AL    <tibble [6 x 5]>
##  9 010000800303 AL    <tibble [6 x 5]>
## 10 010000800839 AL    <tibble [6 x 5]>
## # ... with 55,722 more rows
```

```
RE_nest$"GAT and race populations"[[1]]
```

```
## # A tibble: 6 x 5
##   race  gat_count race_count gat_total total
##   <chr>     <dbl>      <dbl>     <int> <int>
## 1 asian         1          5        86   854
## 2 black         2         24        86   854
## 3 hisp         19        437        86   854
## 4 white        63        368        86   854
## 5 other         1         20        86   854
## 6 bipoc        23        486        86   854
```

Factors

In order to make analysis easier, I converted made race — a categorical variable — into a factor with six levels: `asian`, `black`, `hisp`, `white`, `other`, and `bipoc`.

```
race_levels <- c(
  "asian", "black", "hisp", "white", "other", "bipoc"
)

race <- factor(RE$race, levels = race_levels)
```

Data export

I exported `RE` and `RE_nest`, which can be located through the following paths: `data/processed/Race and ethnicity_tidy.rds`, `data/processed/Race and ethnicity_tidy.csv`, and `data/processed/Race and ethnicity by school_tidy.rds`.

```
saveRDS(RE, file = "data/processed/Race and ethnicity_tidy.rds")
write_csv(RE, file = "data/processed/Race and ethnicity_tidy.csv")
saveRDS(RE_nest, file = "data/processed/Race and ethnicity by school_tidy.rds")
```

Data transformation

The last steps before visualizing this data set were to calculate (1) the proportion of students in a GAT program for each level in the factor `race` and (2) the proportion of the school enrolled in a GAT program.

(1) For each school, I calculated the proportion of X race students in a GAT program by dividing `gat_count` by `race_count`, which are both disaggregated by `race`. I call this new variable `prop_gat_race`.

(2) Second, I calculated the proportion of the entire school enrolled in a GAT program by dividing `gat_total` by `total`. This variable is stored as `prop_gat_total`.

I call this new data frame — which includes both `prop_gat_race` and `prop_gat_total` — `RE_prop`.

```
RE_prop <- RE %>%
  mutate(prop_gat_race = gat_count / race_count,
         prop_gat_total = gat_total / total)
RE_prop
```

```
## # A tibble: 334,392 x 9
##    COMBOKEY state race  gat_count race_count gat_total total prop_gat_race
##    <chr>    <chr> <chr>     <dbl>      <dbl>     <int> <int>         <dbl>
##  1 0100005~ AL    asian         1          5        86   854        0.2
##  2 0100005~ AL    black         2         24        86   854        0.0833
##  3 0100005~ AL    hisp         19        437        86   854        0.0435
##  4 0100005~ AL    white        63        368        86   854        0.171
##  5 0100005~ AL    other         1         20        86   854        0.05
##  6 0100005~ AL    bipoc        23        486        86   854        0.0473
##  7 0100005~ AL    asian         0          2        80   906        0
##  8 0100005~ AL    black         0         35        80   906        0
##  9 0100005~ AL    hisp         14        443        80   906        0.0316
## 10 0100005~ AL    white        63        393        80   906        0.160
## # ... with 334,382 more rows, and 1 more variable: prop_gat_total <dbl>
```

At some schools, there were no students of a level in `race`, so dividing by `race_count` — or dividing by `0` — produces `NaN` ("not a number"). For these cases, I replaced `NaN` with `0`. Additionally, note that when `prop_gat_race` equals `0`, there will be missing values in the visualizations below.

```
RE_prop$prop_gat_race[is.nan(RE_prop$prop_gat_race)] <- 0
```

After looking through my data and testing various visualizations, I decided to narrow my definition of a "Gifted and Talented" program to only include programs where less than half of the school's population was in a GAT program. Thus, I filtered out schools where `prop_gat_total` was greater than `0.5`. In addition, I found a few instances where `prop_gat_race` was greater than `1`, meaning there were more students of X race enrolled in a GAT program than there were students of X race in the entire school; I removed these schools from `RE_prop`, and I believe that they were errors from coding. Second, I only included schools where `prop_gat_total` was greater than `0`, meaning that there was at least one student enrolled in the school's GAT program.

```
RE_prop <- RE_prop %>%
  filter(prop_gat_race < 1) %>%
  filter(prop_gat_total < 0.5) %>%
  filter(prop_gat_total > 0)
```

Data visualization

In the following series of visualizations, I demonstrate that BIPOC students are underrepresented in GAT programs, while white students are overrepresented in GAT programs.

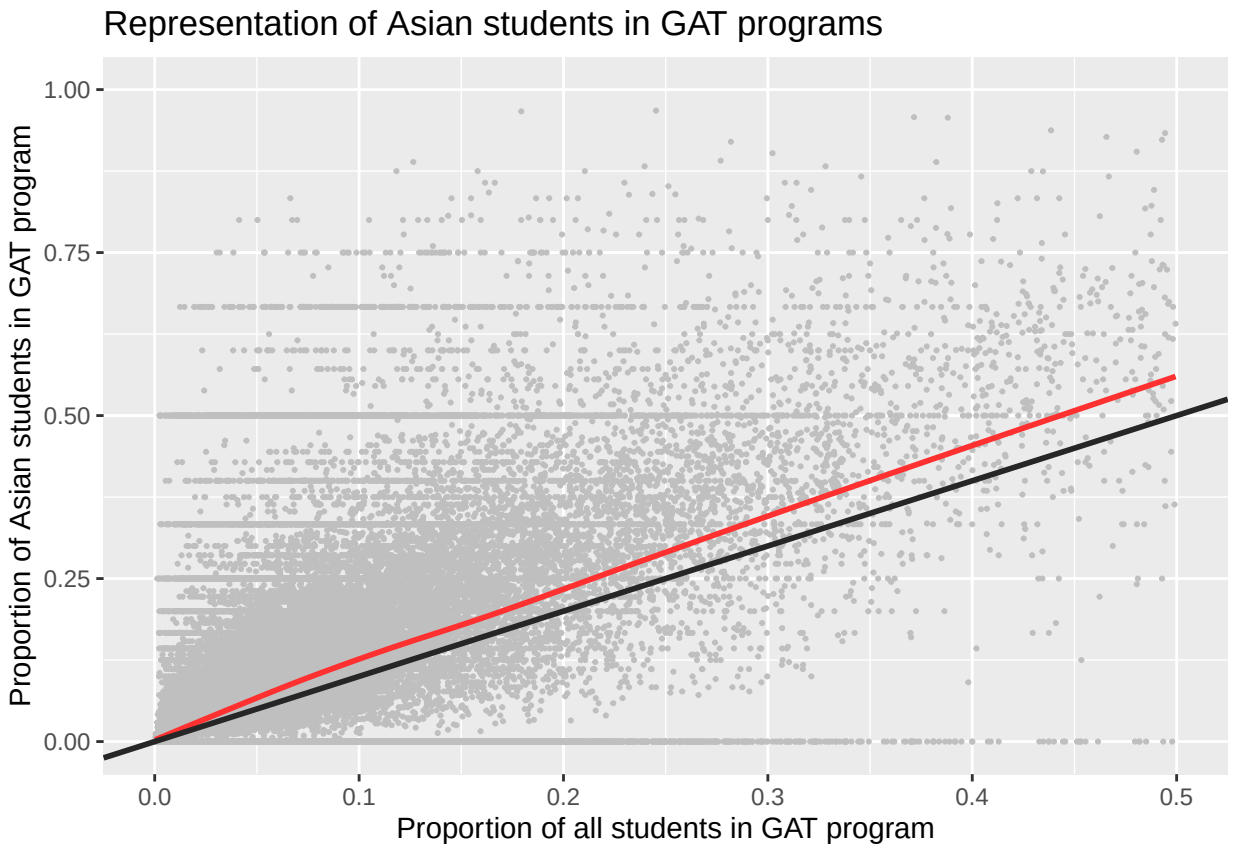
(1) Representation of Asian students in GAT programs: This graphic visualizes the representation of Asian students in GAT programs by graphing the proportion of Asian students in GAT programs by the proportion of a school's entire population that is in its GAT program. In other words, I graphed `prop_gat_total` on the x-axis and `prop_gat_race` on the y-axis after filtering observations where `race == "asian"`. Each point on this graph represents an individual school.

I included a line of identity $y = x$ to be used as a reference to compare the (un)equality of representation in GAT programs by race. If Asian students in GAT programs were represented in proportion to the total number of students in GAT programs for any given school, our fitted model from `geom_smooth()` would lie exactly along the line $y = x$. Our model suggests that Asian students are slightly overrepresented in GAT programs.

```
RE_prop %>%
  filter(race == "asian") %>%
  ggplot(aes(x = prop_gat_total, y = prop_gat_race)) +
  geom_jitter(color = "grey75", size = 0.4) +
  geom_smooth(color = "firebrick1", se = F) +
  geom_abline(intercept = 0, slope = 1, size = 1, color = "grey15") +
  scale_x_continuous(name = "Proportion of all students in GAT program", limits = c(0, 0.5)) +
  scale_y_continuous(name = "Proportion of Asian students in GAT program", limits = c(0,1)) +
  labs(title = "Representation of Asian students in GAT programs")
```

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

```
## Warning: Removed 14587 rows containing missing values (geom_point).
```



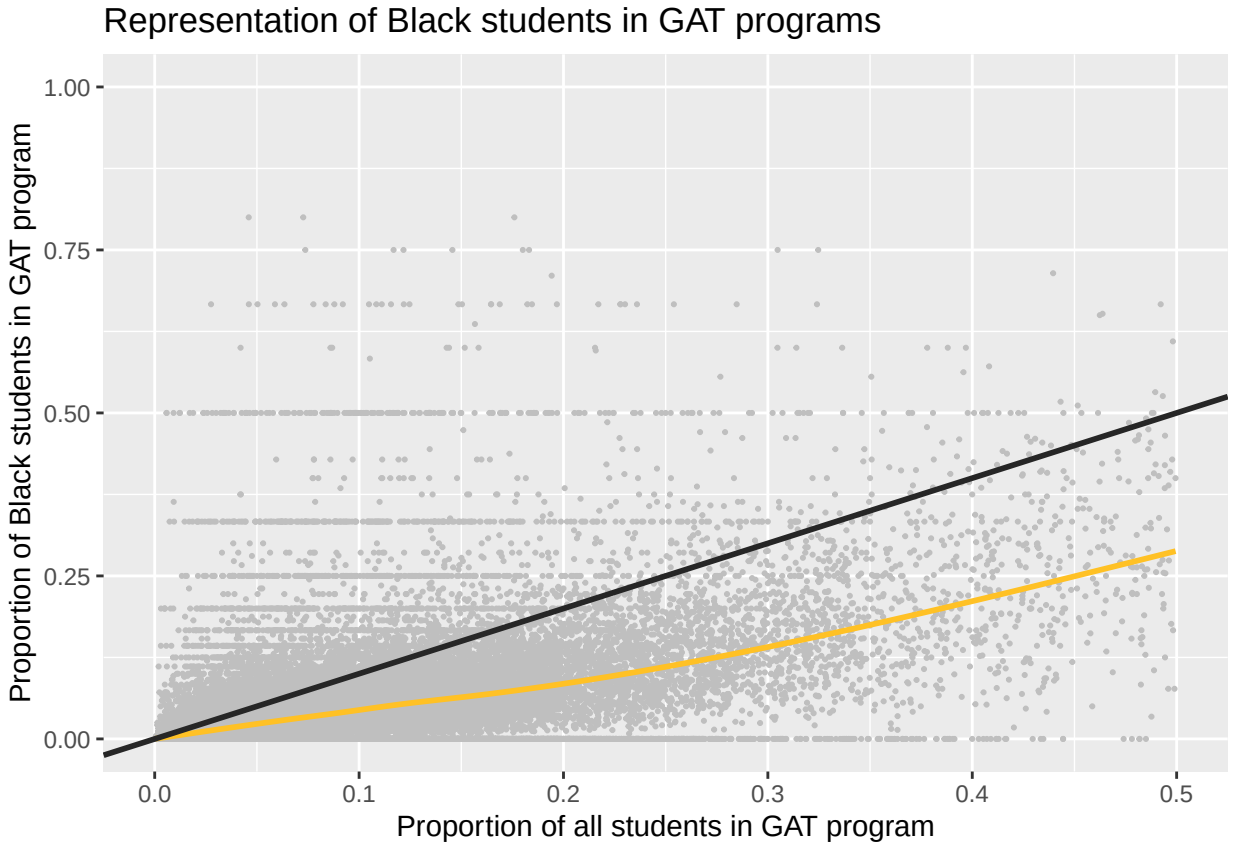
(2) Representation of Black students in GAT programs: This graphic visualizes the representation of Black students in GAT programs by graphing the proportion of Black students in GAT programs by the proportion of a school's entire population that is in its GAT program.

As explained above, if Black students in GAT programs were represented in proportion to the total number of students in GAT programs for any given school, our fitted model from `geom_smooth()` would lie exactly along the line $y = x$. Our model suggests that Black students are significantly underrepresented in GAT programs.

```
RE_prop %>%
  filter(race == "black") %>%
  ggplot(aes(x = prop_gat_total, y = prop_gat_race)) +
  geom_jitter(color = "grey75", size = 0.4) +
  geom_smooth(color = "goldenrod1", se = F) +
  geom_abline(intercept = 0, slope = 1, size = 1, color = "grey15") +
  scale_x_continuous(name = "Proportion of all students in GAT program", limits = c(0, 0.5)) +
  scale_y_continuous(name = "Proportion of Black students in GAT program", limits = c(0,1)) +
  labs(title = "Representation of Black students in GAT programs")
```

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

```
## Warning: Removed 13909 rows containing missing values (geom_point).
```



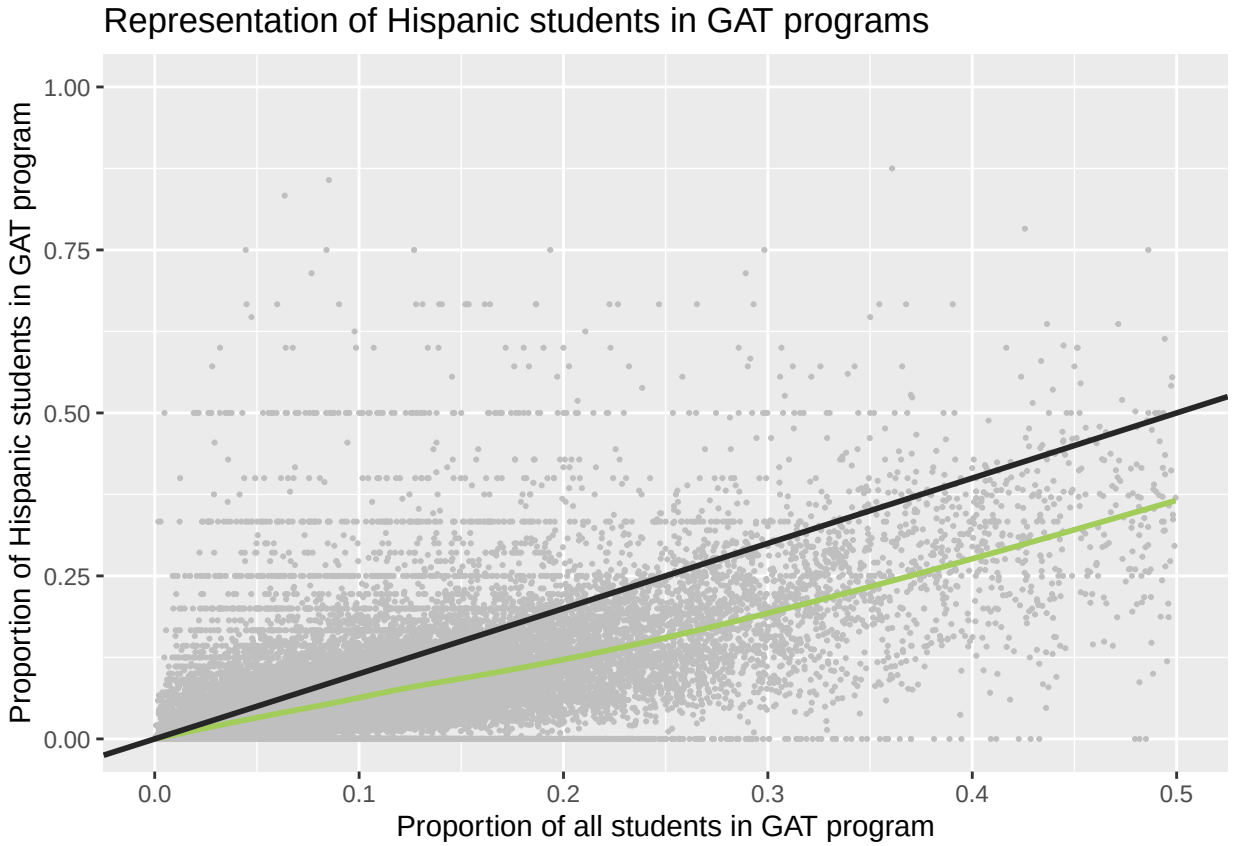
(3) Representation of Hispanic students in GAT programs: This graphic visualizes the representation of Hispanic students in GAT programs by graphing the proportion of Hispanic students in GAT programs by the proportion of a school's entire population that is in its GAT program.

As explained above, if Hispanic students in GAT programs were represented in proportion to the total number of students in GAT programs for any given school, our fitted model from `geom_smooth()` would lie exactly along the line $y = x$. Our model suggests that Hispanic students are slightly underrepresented in GAT programs.

```
RE_prop %>%
  filter(race == "hisp") %>%
  ggplot(aes(x = prop_gat_total, y = prop_gat_race)) +
  geom_jitter(color = "grey75", size = 0.4) +
  geom_smooth(color = "darkolivegreen3", se = F) +
  geom_abline(intercept = 0, slope = 1, size = 1, color = "grey15") +
  scale_x_continuous(name = "Proportion of all students in GAT program", limits = c(0, 0.5)) +
  scale_y_continuous(name = "Proportion of Hispanic students in GAT program", limits = c(0,1)) +
  labs(title = "Representation of Hispanic students in GAT programs")
```

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

```
## Warning: Removed 8576 rows containing missing values (geom_point).
```



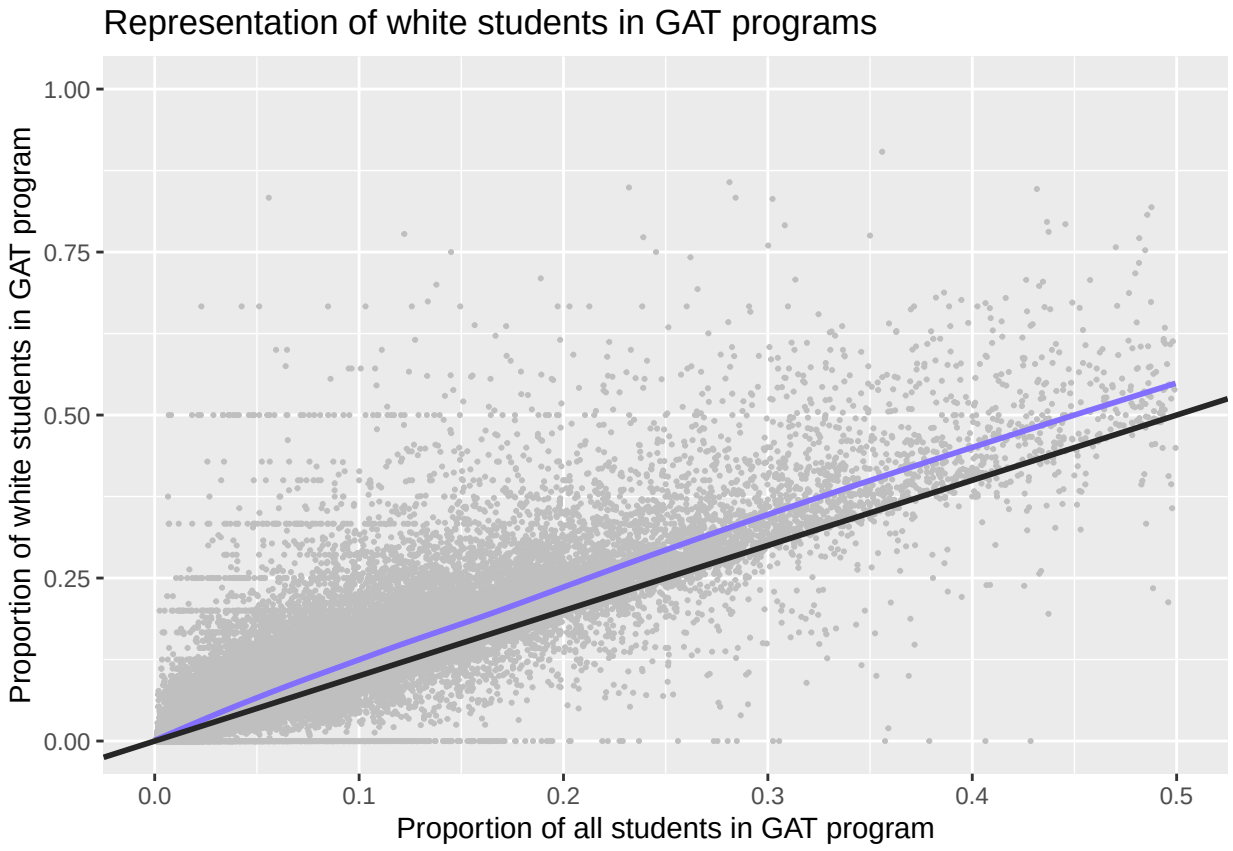
(4) Representation of white students in GAT programs: This graphic visualizes the representation of white students in GAT programs by graphing the proportion of white students in GAT programs by the proportion of a school's entire population that is in its GAT program.

As explained above, if white students in GAT programs were represented in proportion to the total number of students in GAT programs for any given school, our fitted model from `geom_smooth()` would lie exactly along the line $y = x$. Our model suggests that white students are significantly overrepresented in GAT programs.

```
RE_prop %>%
  filter(race == "white") %>%
  ggplot(aes(x = prop_gat_total, y = prop_gat_race)) +
  geom_jitter(color = "grey75", size = 0.4) +
  geom_smooth(color = "slateblue1", se = F) +
  geom_abline(intercept = 0, slope = 1, size = 1, color = "grey15") +
  scale_x_continuous(name = "Proportion of all students in GAT program", limits = c(0, 0.5)) +
  scale_y_continuous(name = "Proportion of white students in GAT program", limits = c(0,1)) +
  labs(title = "Representation of white students in GAT programs")
```

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

```
## Warning: Removed 2635 rows containing missing values (geom_point).
```



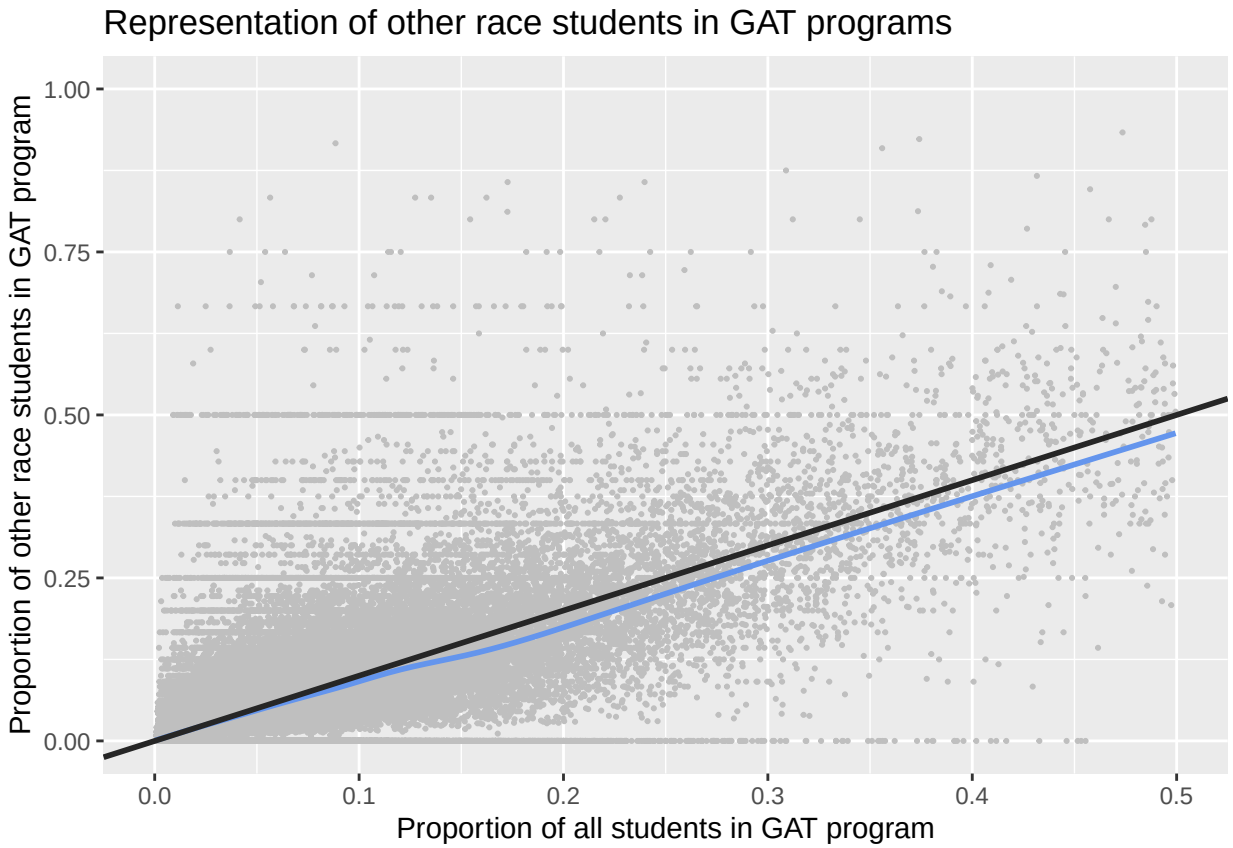
(5) Representation of other race students in GAT programs: This graphic visualizes the representation of other race students in GAT programs by graphing the proportion of other race students in GAT programs by the proportion of a school's entire population that is in its GAT program.

As explained above, if other race students in GAT programs were represented in proportion to the total number of students in GAT programs for any given school, our fitted model from `geom_smooth()` would lie exactly along the line $y = x$. Our model suggests that other race students are slightly underrepresented in GAT programs, but the trend line is very close to the line of identity.

```
RE_prop %>%
  filter(race == "other") %>%
  ggplot(aes(x = prop_gat_total, y = prop_gat_race)) +
  geom_jitter(color = "grey75", size = 0.4) +
  geom_smooth(color = "cornflowerblue", se = F) +
  geom_abline(intercept = 0, slope = 1, size = 1, color = "grey15") +
  scale_x_continuous(name = "Proportion of all students in GAT program", limits = c(0, 0.5)) +
  scale_y_continuous(name = "Proportion of other race students in GAT program", limits = c(0,1)) +
  labs(title = "Representation of other race students in GAT programs")
```

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

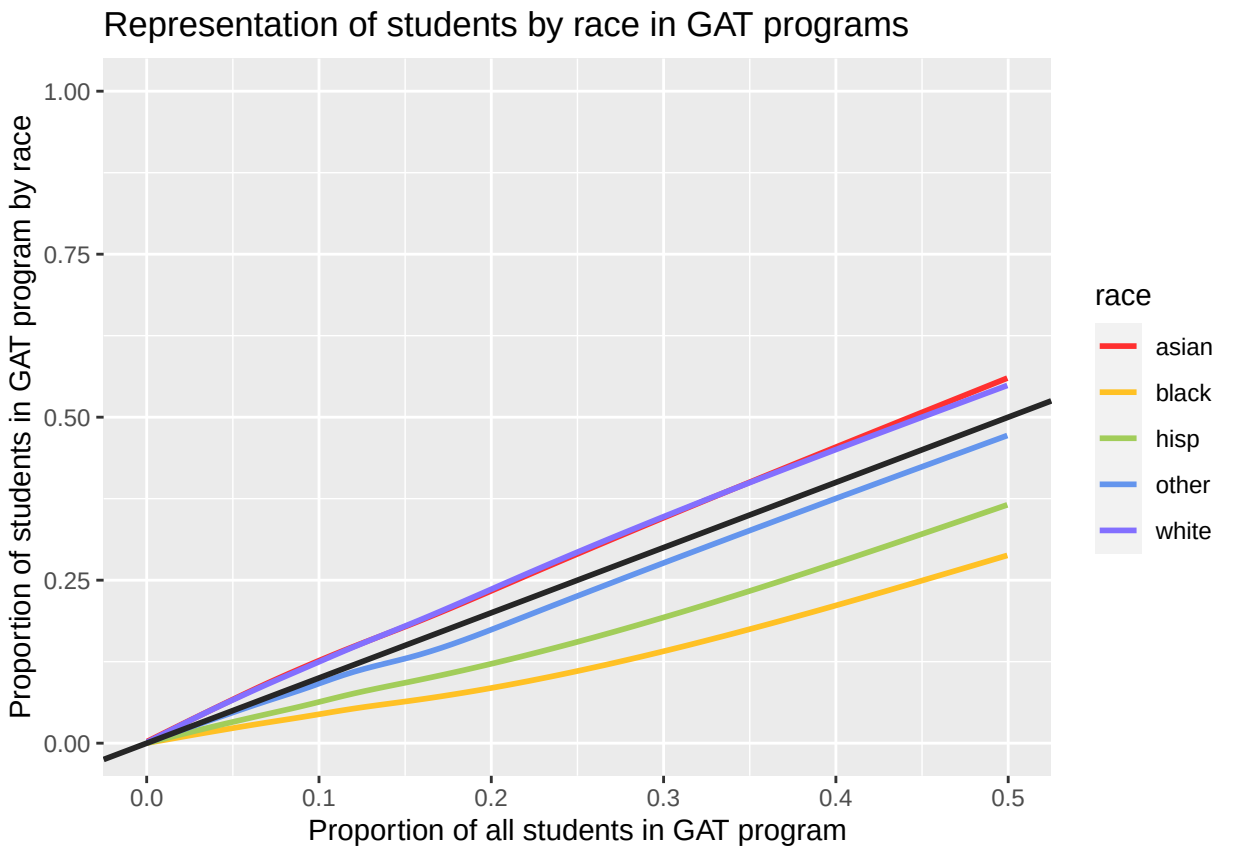
```
## Warning: Removed 12324 rows containing missing values (geom_point).
```



(6) Representation of students by race in GAT programs: This visualization includes the graphics produced by `ggplot2::geom_smooth()` above. By laying these five model lines together on a single graph, we can view the discrepancies that exist in GAT programs by race. Although Asian students are overrepresented in GAT programs — likely because Asian students are racialized by the "model minority" myth — all other non-white racialized groups were underrepresented in GAT programs. Additionally, white students are overrepresented in GAT programs.

```
RE_prop %>%
  filter(race != "bipoc") %>%
  ggplot(aes(prop_gat_total, prop_gat_race, color = race)) +
  geom_smooth(se = F) +
  scale_color_manual(values = c("firebrick1", "goldenrod1",
                                "darkolivegreen3", "cornflowerblue", "slateblue1")) +
  geom_abline(intercept = 0, slope = 1, size = 1, color = "grey15") +
  scale_x_continuous(name = "Proportion of all students in GAT program", limits = c(0, 0.5)) +
  scale_y_continuous(name = "Proportion of students in GAT program by race", limits = c(0,1)) +
  labs(title = "Representation of students by race in GAT programs")
```

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



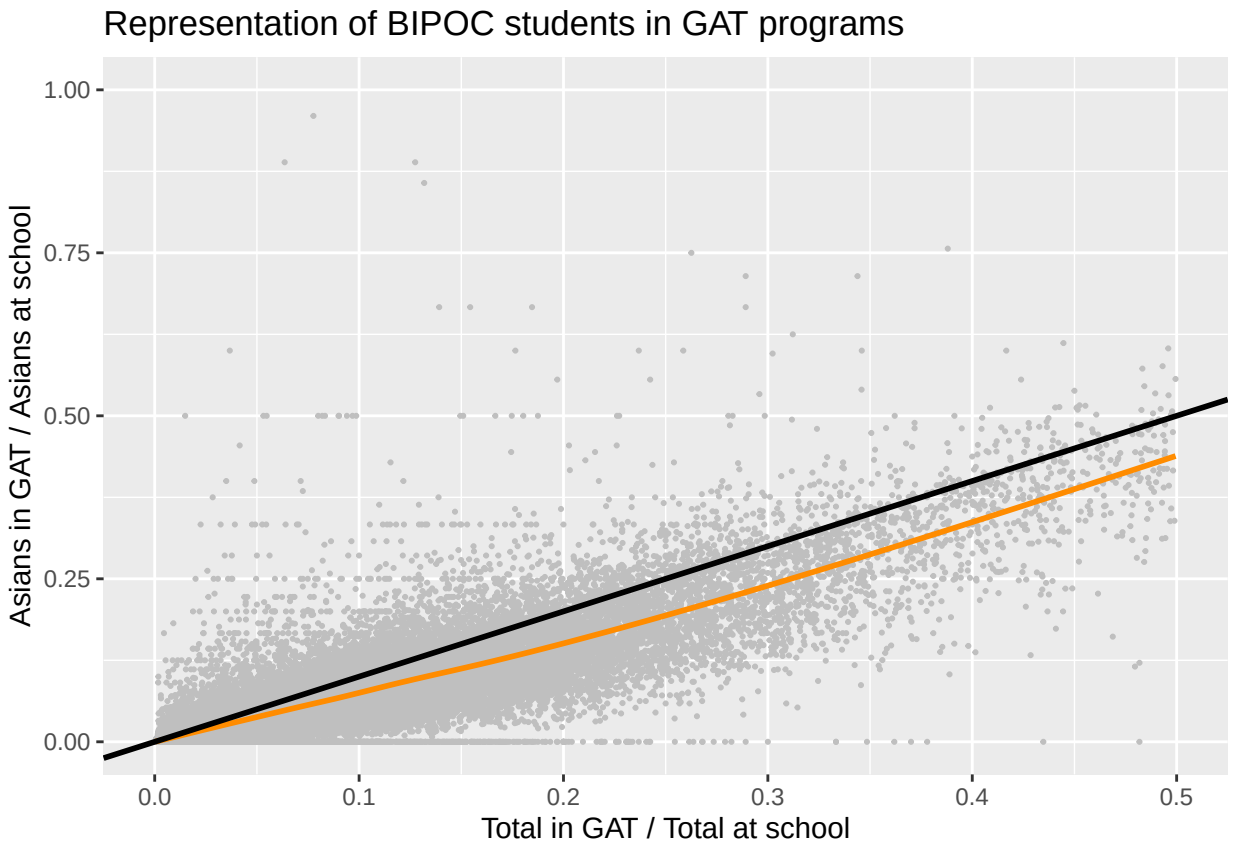
(7) Representation of BIPOC students in GAT programs: This graphic visualizes the representation of BIPOC students in GAT programs by graphing the proportion of BIPOC students in GAT programs by the proportion of a school's entire population that is in its GAT program. To reiterate, the category "BIPOC" is a summation from the five non-`white` `race` categories.

As explained above, if BIPOC students in GAT programs were represented in proportion to the total number of students in GAT programs for any given school, our fitted model from `geom_smooth()` would lie exactly along the line $y = x$. Our model suggests that BIPOC students are significantly underrepresented in GAT programs.

```
RE_prop %>%
  filter(race == "bipoc") %>%
  ggplot(aes(x = prop_gat_total, y = prop_gat_race)) +
  geom_jitter(color = "grey75", size = 0.4) +
  geom_smooth(color = "darkorange", se = F) +
  geom_abline(intercept = 0, slope = 1, size = 1) +
  scale_x_continuous(name = "Total in GAT / Total at school", limits = c(0, 0.5)) +
  scale_y_continuous(name = "Asians in GAT / Asians at school", limits = c(0,1)) +
  labs(title = "Representation of BIPOC students in GAT programs")
```

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

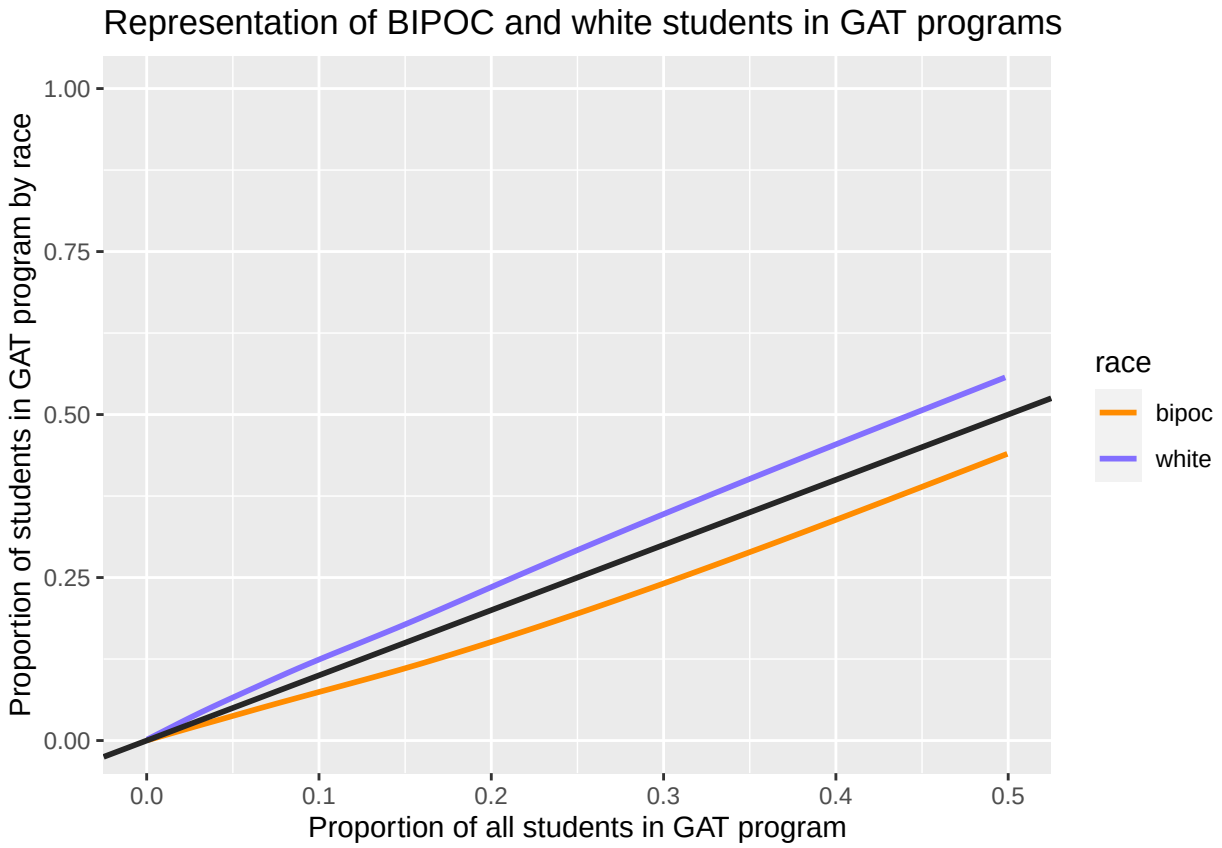
```
## Warning: Removed 3540 rows containing missing values (geom_point).
```



(8) Representation of BIPOC and white students in GAT programs: To compare the representation of white and non-white students in GAT programs, I only included the visualizations produced by `ggplot2::geom_smooth()` for the `race` categories `bipoc` and `white`. As seen in this plot, white students are overrepresented in GAT programs while BIPOC students are underrepresented in GAT programs.

```
RE_prop %>%
  filter(race == c("bipoc", "white")) %>%
  ggplot(aes(prop_gat_total, prop_gat_race, color = race)) +
  geom_smooth(se = F) +
  scale_color_manual(values = c("dark orange", "slateblue1")) +
  geom_abline(intercept = 0, slope = 1, size = 1, color = "grey15") +
  scale_x_continuous(name = "Proportion of all students in GAT program", limits = c(0, 0.5)) +
  scale_y_continuous(name = "Proportion of students in GAT program by race", limits = c(0,1)) +
  labs(title = "Representation of BIPOC and white students in GAT programs")
```

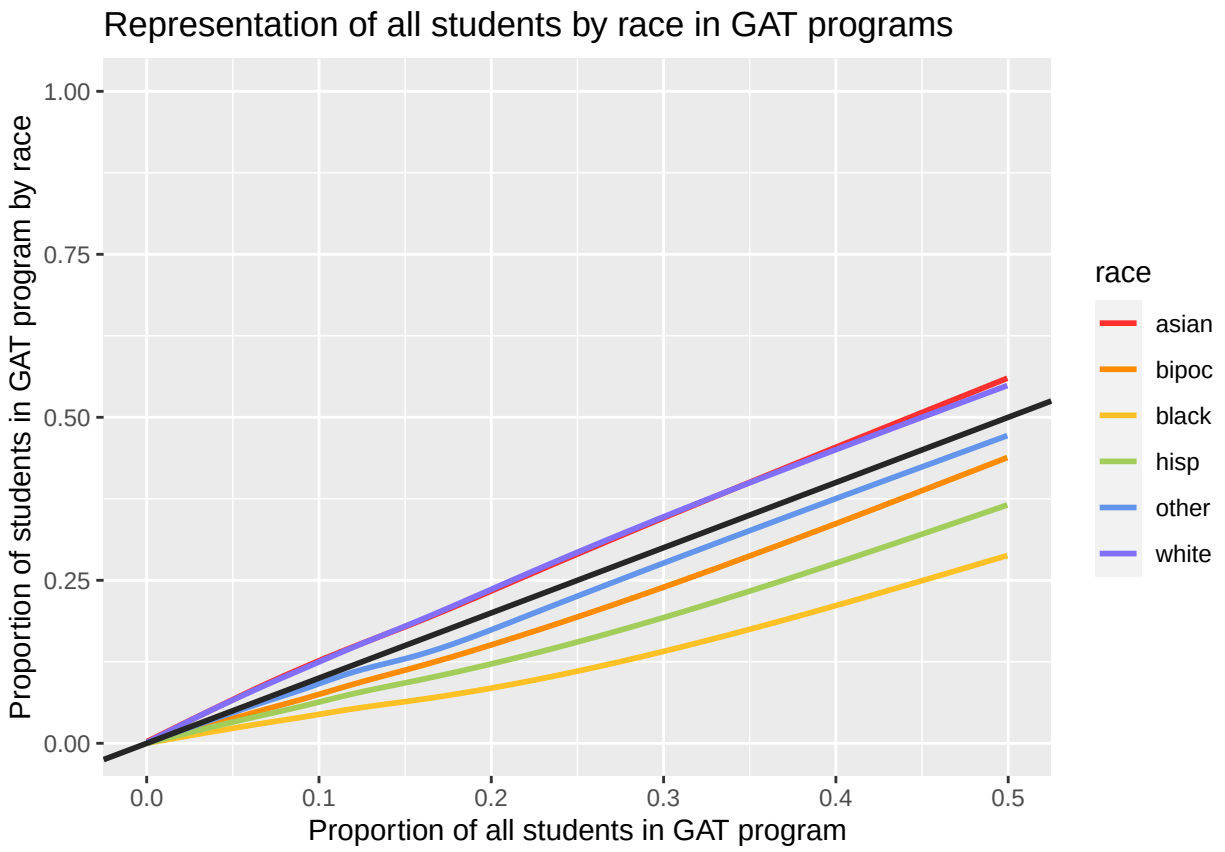
```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```



(9) Representation of all students in GAT programs: In this graph, I included all six `race` categories in one visualization: `asian`, `black`, `hisp`, `white`, `other`, and `bipoc`. As seen below, Asian and white students are the most overrepresented populations in GAT programs from our data set, while all other racialized groups are underrepresented in GAT programs likely due to their minority status. Black students are the most underrepresented group in Gifted and Talented programs, indicating the ways in which anti-Blackness and anti-Black racism permeate the U.S. school system.

```
RE_prop %>%
  ggplot(aes(prop_gat_total, prop_gat_race, color = race)) +
  geom_smooth(se = F) +
  scale_color_manual(values = c("firebrick1", "darkorange", "goldenrod1",
                                "darkolivegreen3", "cornflowerblue", "slateblue1")) +
  geom_abline(intercept = 0, slope = 1, size = 1, color = "grey15") +
  scale_x_continuous(name = "Proportion of all students in GAT program", limits = c(0, 0.5)) +
  scale_y_continuous(name = "Proportion of students in GAT program by race", limits = c(0,1)) +
  labs(title = "Representation of all students by race in GAT programs")
```

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

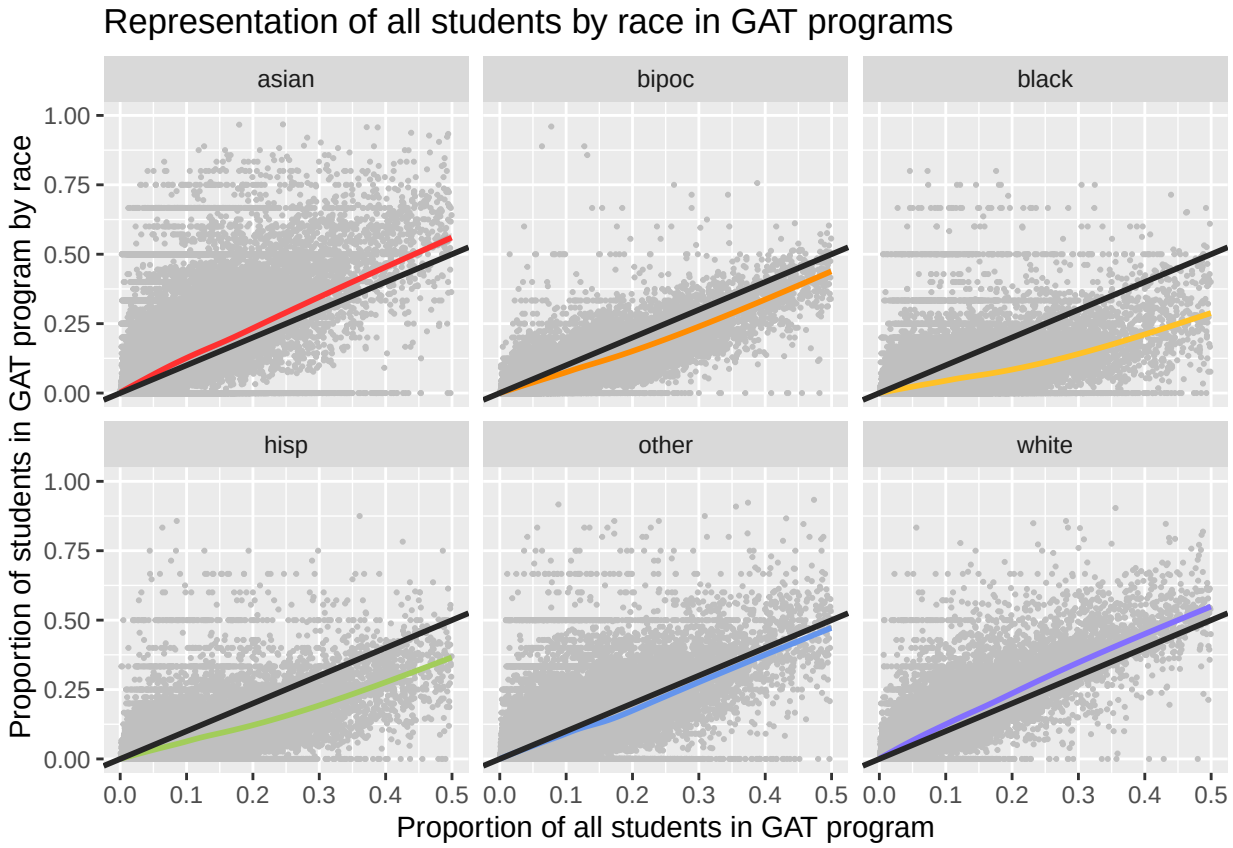


(10) Representation of all students in GAT programs, facetted by race: This last graph is a final way that we can visualize the racial disparities in GAT programs. This visualization is a facetted graph that divides our jitter plot into six subplots by `race`, allowing us to view the trends detailed above side by side.

```
RE_prop %>%
  ggplot(aes(x = prop_gat_total, y = prop_gat_race, color = race)) +
  geom_jitter(color = "grey75", size = 0.4) +
  geom_smooth() +
  scale_color_manual(values = c("firebrick1", "darkorange", "goldenrod1",
                                "darkolivegreen3", "cornflowerblue", "slateblue1")) +
  geom_abline(intercept = 0, slope = 1, size = 1, color = "grey15") +
  facet_wrap(vars(race)) + theme(legend.position = "none") +
  scale_x_continuous(name = "Proportion of all students in GAT program", limits = c(0, 0.5)) +
  scale_y_continuous(name = "Proportion of students in GAT program by race", limits = c(0,1)) +
  labs(title = "Representation of all students by race in GAT programs")
```

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

```
## Warning: Removed 55133 rows containing missing values (geom_point).
```



Analysis II: Ratio of BIPOC to white students in GAT programs by state

In this section, I explore how the representation of BIPOC and white students in GAT programs vary by state. This second analysis is far more brief than Analysis I, but I found the findings interesting, and in turn, included two visualizations in this EDA.

Relational data: `inner_join()`

To start, I joined the data sets `enroll` and `GAT` using a mutating join, `inner_join()`. Again, I matched pairs of observations in `enroll` and `GAT` by `COMBOKEY`, creating one pooled data set `by_state`.

```
by_state <- enroll %>%
  inner_join(GAT,
             by = "COMBOKEY")
```

Data transformation

The next step in this process was to transform this data table. First, I renamed and created the variables `gat_white`, `white`, `gat_bipoc`, and `bipoc` by collapsing sex by race I decided to dichotomize `race` for this analysis because I wanted to investigate how the proportion of students of color in GAT programs compares to the proportion of white students in GAT programs. Thus, I only needed two levels in the factor race: `bipoc` and `white`. Additionally, I filtered this data set to only include schools with a total population that is greater than 10.

Next, I summed the populations of `gat_white`, `white`, `gat_bipoc`, and `bipoc` by state, leaving me with 50 observations (one observation for each state). Using these four variables, I created `bipoc_white_ratio` which is the ratio of BIPOC to white students in GAT programs disaggregated by state. To calculate this ratio, I found the proportion of BIPOC students represented in GAT programs by dividing `gat_bipoc` by `bipoc`. In the same way, I calculated the proportion of white students represented in GAT programs by dividing `gat_white` by `white`. Finally, I used these two proportions to calculate the ratio of BIPOC to white students in GAT programs by dividing `gat_bipoc/bipoc` by `gat_white/white`; I also multiplied this ratio — `gat_bipoc/bipoc:gat_white/white` — by `100` for scaling purposes.

```
by_state <- by_state %>%
  mutate(state = state.x,
         gat_white = gat_white_m + gat_white_f,
         white = white_m + white_f,
         gat_bipoc = gat_bipoc_m + gat_bipoc_f,
         bipoc = bipoc_m + bipoc_f) %>%
  filter(total >= 10) %>%
  select(state, gat_white, white,
         gat_bipoc, bipoc,
         gat_total, total) %>%
  group_by(state) %>%
  summarize_all(sum) %>%
  mutate(bipoc_white_ratio =
           ((gat_bipoc/bipoc)/(gat_white/white)) * 100)
```

Tidy data

Using the same process as Analysis I, I tidied the data frame `by_state`. I used the functions `unite()`, `pivot_longer()`, `separate()`, `mutate()`, `select()`, and `factor()` just as I did above. The tidied data frame `by_state` is a table with 100 observations of five variables: `state`, `race`, `gat_count`, `race_count`, and `bipoc_white_ratio`.

```r
by_state <- by_state %>%
  unite(white, gat_white, white, sep = "_") %>%
  unite(bipoc, gat_bipoc, bipoc, sep = "_") %>%
  pivot_longer(c("white", "bipoc"),
               names_to = "race", values_to = "count") %>%
  separate(count, into = c("gat_count", "race_count"), sep = "_") %>%
  mutate(gat_count = as.numeric(gat_count),
         race_count = as.numeric(race_count)) %>%
  select(state, race, gat_count, race_count, bipoc_white_ratio)

race_levels_by_state <- c(
  "bipoc", "white"
)

race <- factor(by_state$race, levels = race_levels_by_state)

by_state
```

```
## # A tibble: 100 x 5
##    state race  gat_count race_count bipoc_white_ratio
##    <chr> <chr>     <dbl>      <dbl>              <dbl>
##  1 AK    white      4261      50054               54.1
##  2 AK    bipoc      2054      44600               54.1
##  3 AL    white     33603     280469               45.5
##  4 AL    bipoc     11581     212236               45.5
##  5 AR    white     32369     278296               68.4
##  6 AR    bipoc     13915     174785               68.4
##  7 AZ    white     29659     272092               45.2
##  8 AZ    bipoc     22751     462165               45.2
##  9 CA    white    104000     862829               71.6
## 10 CA    bipoc    275255    3191278               71.6
## # ... with 90 more rows
```

Data export

To view this data frame, you can follow the path `data/processed/Summation by state_tidy.rds` or `data/processed/Summation by state_tidy.csv`.

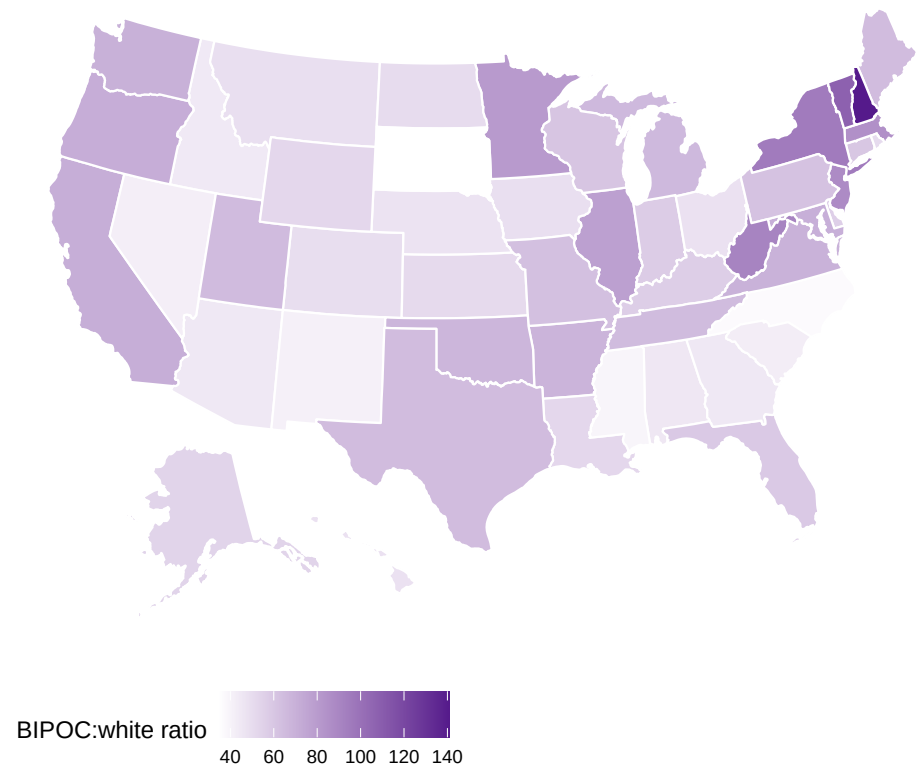
```r
saveRDS(by_state, file = "data/processed/Summation by state_tidy.rds")
write_csv(by_state, file = "data/processed/Summation by state_tidy.csv")
```

Data visualization

Using the package `usmap`, I plotted `bipoc_white_ratio` for each state. The shading of each state indicates the ratio of BIPOC:white students that are represented in GAT programs. The darker shade of purple that a state is, the higher the BIPOC:white ratio. In order to quantitatively compare these ratios by state, I also included a bar chart visualization.

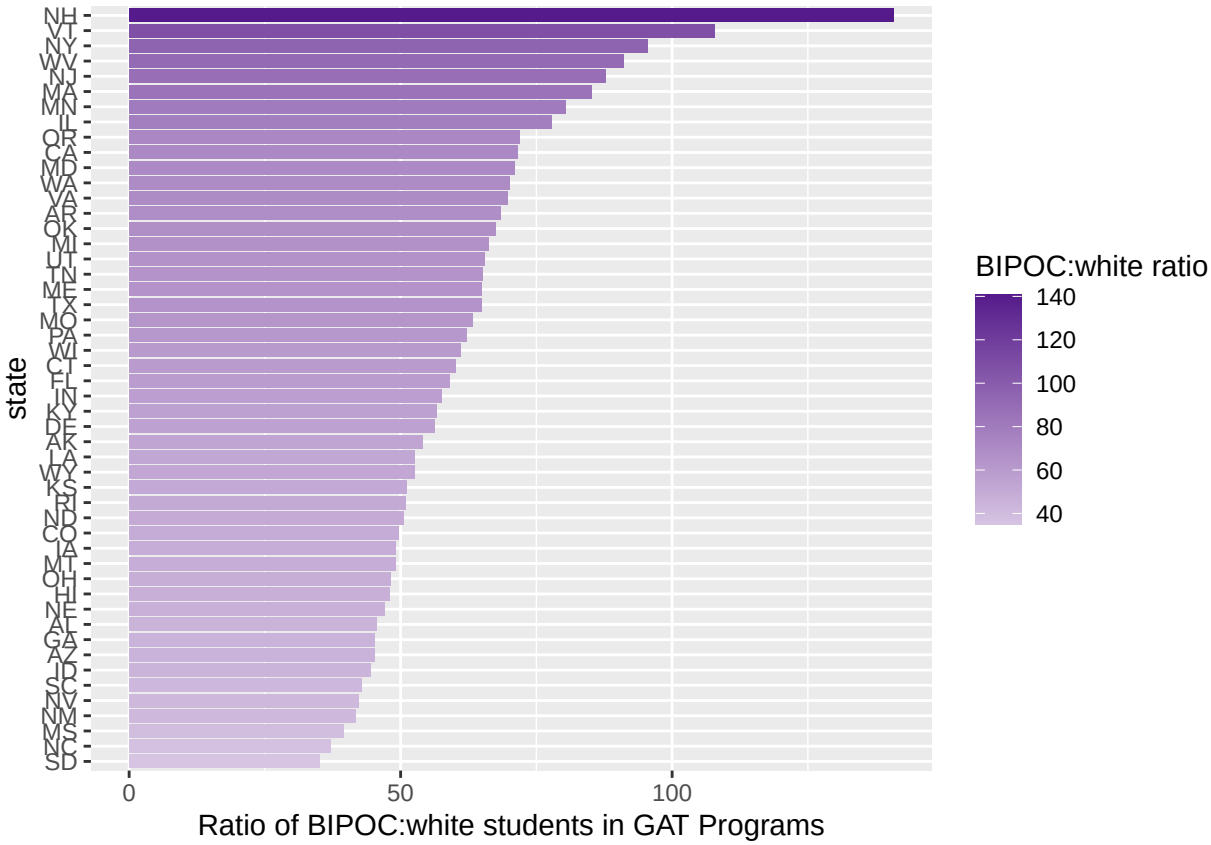
```
library(usmap)

plot_usmap(data = by_state, values = "bipoc_white_ratio", color = "white") +
  scale_fill_continuous(low = "white", high = "purple4",
    name = "BIPOC:white ratio", label = scales::comma) +
  labs(title = "Ratio of BIPOC:white students in GAT programs by state") +
  theme(legend.position = "bottom")
```




```
by_state_barchart <- by_state %>%
  select(state, bipoc_white_ratio) %>%
  arrange(-bipoc_white_ratio) %>%
  unique()

ggplot(by_state_barchart, aes(state, bipoc_white_ratio)) +
  geom_col(aes(reorder(state, bipoc_white_ratio), fill = bipoc_white_ratio)) +
  scale_fill_gradient2(low = "white", high = "purple4") +
  scale_y_continuous(name = "Ratio of BIPOC:white students in GAT Programs") +
  labs(fill = "BIPOC:white ratio") +
  coord_flip()
```



From these two graphics, we can observe that the states with the highest ratio of BIPOC:white students in GAT programs are New Hampshire, Vermont, and New York, and the states with the lowest ratio of BIPOC:white students in GAT programs are South Dakota, North Carolina, and Mississippi.

Conclusion

This project provided me with an opportunity for me to pursue my passions for social justice and education in a new setting. As an Ethnic Studies major, educational inequities, such as the opportunity gap and the racialization of academic success, are issues of particular interest to me. Before beginning these analyses, I hypothesized that students of color would be underrepresented in these "gifted" programs, while white students would be disproportionately selected for these programs due to racial biases. From Analysis I, we observe that there is a correlation between race and students that are representation in Gifted and Talented programs in the United States. Racially minoritized students — with the exception of Asian students — tend to be underrepresented in GAT programs. Black students are especially underrepresented in GAT programs, which I hypothesize is rooted in anti-Blackness.

While Analysis I demonstrates my knowledge in `tidyverse`, Analysis II showcases the statistical manipulation techniques that I have developed throughout this quarter. I hope to continue this project in the future, likely in my own time, and use an intersectional approach to investigate how race *and* sex influence the representation of students in GAT programs; based on outside knowledge, I believe that Black women and women of color will be substantially underrepresented in GAT programs. For me, a significant shortcoming of this data was the perpetuation of the gender binary. By collecting data by sex, this survey overlooked the ways in which gender minorities — or non-cisgender students — experience minority stress, which contributes to academic underperformance in school. Accordingly, it is impossible to conduct gender-affirming analyses using data from the CRDC even though the survey's primary purpose is to "collect data on key education and civil rights issues in our nation's public schools".