CS01 - Youth Disconnection

Yohan Kim, Ming Qiu, Kevin Lam, Khiem Pham

Introduction

Disconnected youth refers to people between the ages of 16 and 24 who aren't working or in school.[1] Measure of America, a project dedicated to understanding opportunity and well-being in America, claims that people who go through a period of disconnection as young adults are less likely to be employed, own a home, and report good health by the time they're in their thirties.[1] Some risk factors that may contribute to this include poverty, poor mental health, and racial/ethnic disparities.[1] For this case study, we will investigate the relationship between disconnection and different ethnic subgroups of youths overtime.

Questions

How have youth disconnection rates in American youth changed since 2008?

How has this changed for different genders and ethnic groups? Are any groups particularly disconnected?

Additional Questions

How will the disconnection percentages of each race change in upcoming years? Which ethnic group is more likely to show improvement in reduction of disconnection rate and which one is less likely to show improvement??

Load packages

```
library('pdftools')
library('magick')
library('tesseract')
library('OCSdata')
library('tidyverse')
library('knitr')
library('broom')
library('Kendall')
library('tidymodels')
library('ggrepel')
```

The Data

Data Import

We load the data and the pdf.

```
load_raw_data("data/ocs-bp-youth-disconnection", outpath='.')
pdf_tools_example <- pdf_text('data/raw/Making_the_Connection.pdf')
```

We read in the major ethnic groups image and extract the numbers and text.

```r
major_racial_ethnic_groups <-
  image_read('data/raw/Major_ethnic_groups_screenshot.png')
major_groups <- image_ocr(major_racial_ethnic_groups)
```

We read in the asian subgroups 2017 image and extract the numbers and text.

```r
asian_subgroups <- image_read('data/raw/asian_subgroups_2017.png')

asian_sub_2017 <- image_ocr(asian_subgroups)
asian_sub_2017
```

```
## [1] "United States 11.5\nMale 11.8\nFemale 11.1\nASIAN 6.8\nAsian Male 65\nAsian Female 67\nCHINESE 4
```

We read in the asian sub 2017 A, asian sub 2017 B, and asian sub 2017 C images and extract the numbers and text from all of them.

```r
asian_subgroups_A <- image_read('data/raw/asian_sub_2017_A.png')
asian_sub_2017_A <- image_ocr(asian_subgroups_A)

asian_subgroups_B <- image_read('data/raw/asian_sub_2017_B.png')
asian_sub_2017_B <- image_ocr(asian_subgroups_B)

asian_subgroups_C <- image_read('data/raw/asian_sub_2017_C.png')
asian_sub_2017_C <- image_ocr(asian_subgroups_C)
```

We read in the asian sub 2018 A and asian sub 2018 B images and extract the numbers and text from both of them.

```r
asian_sub_2018_A <- image_read('data/raw/asian_sub_2018_A.png')
asian_sub_2018_A <- image_ocr(asian_sub_2018_A)
asian_sub_2018_B <- image_read('data/raw/asian_sub_2018_B.png')
asian_sub_2018_B <- image_ocr(asian_sub_2018_B)
```

We read in the latinx sub 2017 A, latinx sub 2017 B, latinx sub 2017 C, and latinx sub 2018 images and extract the numbers and text from all of them.

```r
latinx_imageA <- image_read("data/raw/latinx_sub_2017_A.png")
latinx_imageB <- image_read("data/raw/latinx_sub_2017_B.png")
latinx_imageC <- image_read("data/raw/latinx_sub_2017_C.png")
latinx_sub_2018 <- image_read("data/raw/latinx_subgroups_2018.png")

latinx_sub_2017_A <- image_ocr(latinx_imageA)
latinx_sub_2017_B <- image_ocr(latinx_imageB)
latinx_sub_2017_C <- image_ocr(latinx_imageC)
latinx_sub_2018 <- image_ocr(latinx_sub_2018)
```

Create a make_rows() function that will take in text and split it into new lines to get them into different rows, we unlisted it to take it out of a bigger list, and turns it into a tibble.

```r
make_rows <- function(text){
  text |>
  str_split("\n") |>
  unlist() |>
  as_tibble()
}
```

We combine the asian sub 2018 A and asian sub 2018 B into a single vector and pass it into make_rows().

```r
asian_sub_2018 <- str_c(asian_sub_2018_A, asian_sub_2018_B)
asian_sub_2018 <- make_rows(asian_sub_2018)
asian_sub_2018
```

```
## # A tibble: 23 x 1
##    value
##    <chr>
##  1 "CHINESE : 41"
##  2 "Men 4.5"
##  3 "Women : 3.7"
##  4 ""
##  5 "INDIAN 5.4"
##  6 "Men 4.7"
##  7 "Women : 6.1"
##  8 ""
##  9 "KOREAN : 5.5"
## 10 "Men 5.6"
## # ... with 13 more rows
```

Data Wrangling

We split the major_groups data everytime there's a newline character, unlist to take it out of a bigger list, and turn it into a tibble.

```r
major_groups <- major_groups |>
  str_split(pattern='\n') |>
  unlist() |>
  as_tibble()

major_groups
```

```
## # A tibble: 19 x 1
##    value
##    <chr>
##  1 "United States 12.6 14.7 14.1 13.2 11.7 11.5"
##  2 "Male 12.3 15.2 14.5 13.3 12.1 11.8"
##  3 "Female 12.9 14.1 13.7 13.0 11.2 11.1"
##  4 "ASIAN 7.1 8.5 78 79 6.6 6.6"
##  5 "Asian Male 6.3 8.3 74 7.2 6.7 6.5"
##  6 "Asian Female 7.9 8.6 8.1 8.6 6.6 6.7"
##  7 "WHITE 9.7 11.7 11.2 10.8 9.7 9.4"
##  8 "White Male 9.5 12.3 11.5 10.8 10.0 9.6"
##  9 "White Female 10.0 11.1 10.8 10.7 9.4 9.1"
## 10 "LATINO 16.7 18.5 17.3 15.2 13.7 13.2"
## 11 "Latino Male 13.6 16.8 16.0 14.0 12.6 12.4"
## 12 "Latina Female 20.2 20.3 18.8 16.5 14.8 13.9"
## 13 "BLACK 20.4 22.5 22.4 20.6 17.2 17.9"
## 14 "Black Male 23.7 26.0 25.6 23.5 20.1 20.8"
## 15 "Black Female 17.0 19.0 19.3 17.6 14.2 14.8"
## 16 "NATIVE AMERICAN 24.4 28.8 27.0 26.3 25.8 23.9"
## 17 "Native American Male 25.0 30.9 28.0 26.9 28.1 23.3"
## 18 "Native American Female 23.9 26.7 25.9 25.6 23.4 24.5"
## 19 ""
```

We pass the major_groups into separate to separate information into columns. We use separate again to separate each year into columns. We then drop all rows that contain NA values.

```r
# Creating Columns
major_groups <- major_groups |>
  separate(col=value,
           into=c("Group", "Years"),
           sep="(?<=[[:alpha:]])\\s(?=[0-9])")
```

```
## Warning: Expected 2 pieces. Missing pieces filled with `NA` in 1 rows [19].
```

```r
# Separate Columns
major_groups <- major_groups |>
  separate(col=Years,
           into=c("2008","2010","2012","2014","2016","2017"),
           sep=(" "))

major_groups <- major_groups |>
  drop_na()

major_groups
```

```
## # A tibble: 18 x 7
##    Group                  `2008` `2010` `2012` `2014` `2016` `2017`
##    <chr>                  <chr>  <chr>  <chr>  <chr>  <chr>  <chr>
##  1 United States          12.6   14.7   14.1   13.2   11.7   11.5
##  2 Male                   12.3   15.2   14.5   13.3   12.1   11.8
##  3 Female                 12.9   14.1   13.7   13.0   11.2   11.1
##  4 ASIAN                  7.1    8.5    78     79     6.6    6.6
##  5 Asian Male             6.3    8.3    74     7.2    6.7    6.5
##  6 Asian Female           7.9    8.6    8.1    8.6    6.6    6.7
##  7 WHITE                  9.7    11.7   11.2   10.8   9.7    9.4
##  8 White Male             9.5    12.3   11.5   10.8   10.0   9.6
##  9 White Female           10.0   11.1   10.8   10.7   9.4    9.1
## 10 LATINO                 16.7   18.5   17.3   15.2   13.7   13.2
## 11 Latino Male            13.6   16.8   16.0   14.0   12.6   12.4
## 12 Latina Female          20.2   20.3   18.8   16.5   14.8   13.9
## 13 BLACK                  20.4   22.5   22.4   20.6   17.2   17.9
## 14 Black Male             23.7   26.0   25.6   23.5   20.1   20.8
## 15 Black Female           17.0   19.0   19.3   17.6   14.2   14.8
## 16 NATIVE AMERICAN        24.4   28.8   27.0   26.3   25.8   23.9
## 17 Native American Male   25.0   30.9   28.0   26.9   28.1   23.3
## 18 Native American Female 23.9   26.7   25.9   25.6   23.4   24.5
```

We remove the decimal points, convert to numeric, and get out decimal point back in the Group column.

```r
major_groups <- major_groups |>
  mutate(
    across(.cols = -Group,
           ~ str_remove(string = ., pattern = "\\.")),  # remove decimal points
    across(.cols = -Group, as.numeric),  # convert to numeric
    across(.cols = -Group, ~ . * 0.1)   # get our decimal point back
  )
major_groups
```

```
## # A tibble: 18 x 7
##    Group                  `2008` `2010` `2012` `2014` `2016` `2017`
##    <chr>                   <dbl>  <dbl>  <dbl>  <dbl>  <dbl>  <dbl>
```

```
##  1 United States            12.6    14.7    14.1    13.2    11.7    11.5
##  2 Male                     12.3    15.2    14.5    13.3    12.1    11.8
##  3 Female                   12.9    14.1    13.7    13      11.2    11.1
##  4 ASIAN                     7.1     8.5     7.8     7.9     6.6     6.6
##  5 Asian Male                6.3     8.3     7.4     7.2     6.7     6.5
##  6 Asian Female              7.9     8.6     8.1     8.6     6.6     6.7
##  7 WHITE                     9.7    11.7    11.2    10.8     9.7     9.4
##  8 White Male                9.5    12.3    11.5    10.8    10       9.6
##  9 White Female             10      11.1    10.8    10.7     9.4     9.1
## 10 LATINO                   16.7    18.5    17.3    15.2    13.7    13.2
## 11 Latino Male              13.6    16.8    16      14      12.6    12.4
## 12 Latina Female            20.2    20.3    18.8    16.5    14.8    13.9
## 13 BLACK                    20.4    22.5    22.4    20.6    17.2    17.9
## 14 Black Male               23.7    26      25.6    23.5    20.1    20.8
## 15 Black Female             17      19      19.3    17.6    14.2    14.8
## 16 NATIVE AMERICAN          24.4    28.8    27      26.3    25.8    23.9
## 17 Native American Male     25      30.9    28      26.9    28.1    23.3
## 18 Native American Female   23.9    26.7    25.9    25.6    23.4    24.5
```

We recode all United States, Female, and Male values to All_races in the Race_Ethnicity column, and remove all strings with the pattern "Female|Male".

```
major_groups  <- major_groups |>
mutate(Race_Ethnicity = dplyr::recode(Group, "United States" = "All_races",
                                  "Female" = "All_races",
                                  "Male" = "All_races"),
   Race_Ethnicity = str_remove(string = Race_Ethnicity,
                               pattern = "Female|Male"))
```

We extract all the values containing the pattern "Female|Male" and replace all NA values with "All" in the Gender column.

```
major_groups  <- major_groups |>
  mutate(Gender = str_extract(string = Group,
                              pattern = "Female|Male")) |>
  mutate(Gender = replace_na(Gender, replace = "All"))
```

We apply pivot_longer to convert the year columns to one year column.

```
major_groups <- major_groups |>
  pivot_longer(cols=contains("20"),
               names_to="Year",
               values_to="Percent",
               names_prefix="Perc_") |>
  mutate(Year=as.numeric(Year))
major_groups
```

```
## # A tibble: 108 x 5
##    Group         Race_Ethnicity Gender  Year Percent
##    <chr>         <chr>          <chr>  <dbl>   <dbl>
##  1 United States All_races      All     2008    12.6
##  2 United States All_races      All     2010    14.7
##  3 United States All_races      All     2012    14.1
##  4 United States All_races      All     2014    13.2
##  5 United States All_races      All     2016    11.7
##  6 United States All_races      All     2017    11.5
##  7 Male          All_races      Male    2008    12.3
```

```
## 8 Male          All_races      Male    2010    15.2
## 9 Male          All_races      Male    2012    14.5
## 10 Male         All_races      Male    2014    13.3
## # ... with 98 more rows
```

We pass the asian sub 2017, asian sub 2017 A, asian sub 2017 B, and asian sub 2017 C data into make rows.

```
asian_sub_2017 <- make_rows(asian_sub_2017)
asian_sub_2017
```

```
## # A tibble: 33 x 1
##    value
##    <chr>
##  1 United States 11.5
##  2 Male 11.8
##  3 Female 11.1
##  4 ASIAN 6.8
##  5 Asian Male 65
##  6 Asian Female 67
##  7 CHINESE 43
##  8 Chinese Male AT
##  9 Chinese Female 3.9
## 10 VIETNAMESE 5.5
## # ... with 23 more rows
```

```
asian_sub_2017_A <- make_rows(asian_sub_2017_A)
asian_sub_2017_B <- make_rows(asian_sub_2017_B)
asian_sub_2017_C <- make_rows(asian_sub_2017_C)
asian_sub_2017_C
```

```
## # A tibble: 6 x 1
##   value
##   <chr>
## 1 "FILIPINO 7.3"
## 2 "Filipino Male 6.5"
## 3 "Filipino Female 8.1"
## 4 "HMONG 14.0"
## 5 "Hmong Male 18.6"
## 6 ""
```

We combine the asian sub 2017 tibbles into one.

```
asian_sub_2017 <- bind_rows(asian_sub_2017_A,
                            asian_sub_2017_B,
                            asian_sub_2017_C)
asian_sub_2017
```

```
## # A tibble: 28 x 1
##    value
##    <chr>
##  1 United States 11.5
##  2 Male 11.8
##  3 Female 11.1
##  4 ASIAN 6.6
##  5 Asian Male 6.5
##  6 Asian Female 6.7
##  7 CHINESE 4.3
```

```
##  8 Chinese Male 4.7
##  9 Chinese Female 3.9
## 10 VIETNAMESE 5.5
## # ... with 18 more rows
```

We combine all the code previously to create a function that makes it easier to clean a table.

```
# All work above as function for 2017
clean_table <- function(table){
  table |>
    separate(col = value,
             into = c("Group", "Percentage"),
             sep =  "(?<=[[:alpha:]])\\s(?=[0-9])") |>
    drop_na() |>
    mutate(Group = str_to_title(Group)) |>
    mutate(Percentage = str_remove(string = Percentage,
                                   pattern = "\\.")) |>
    separate(Percentage, c("Percent"), sep = " ") |>
    mutate(Percent = as.numeric(Percent)) |>
    mutate(Percent = Percent * 0.1) |>
    mutate(Race_Ethnicity = recode(Group,
                                   "United States" = "All_races",
                                   "Female" = "All_races",
                                   "Male" = "All_races")) |>
    mutate(Race_Ethnicity = str_remove(string = Race_Ethnicity,
                                       pattern = " Female| Male")) |>
    mutate(Gender = str_extract(string = Group,
                                pattern ="Female|Male")) |>
    mutate(Gender = replace_na(Gender, replace = "All"))
}
```

We clean the asian_sub_2017 data using clean_table().

```
asian_sub_2017 <- clean_table(table = asian_sub_2017)
```

```
## Warning: Expected 2 pieces. Missing pieces filled with `NA` in 3 rows [17, 22,
## 28].
```

```
asian_sub_2017
```

```
## # A tibble: 25 x 4
##    Group           Percent Race_Ethnicity Gender
##    <chr>             <dbl> <chr>          <chr>
##  1 United States      11.5 All_races      All
##  2 Male               11.8 All_races      Male
##  3 Female             11.1 All_races      Female
##  4 Asian               6.6 Asian          All
##  5 Asian Male          6.5 Asian          Male
##  6 Asian Female        6.7 Asian          Female
##  7 Chinese             4.3 Chinese        All
##  8 Chinese Male        4.7 Chinese        Male
##  9 Chinese Female      3.9 Chinese        Female
## 10 Vietnamese          5.5 Vietnamese     All
## # ... with 15 more rows
```

We combine the latinx sub 2017 A B C into a single vector.

```r
latinx_sub_2017 <- stringr::str_c(latinx_sub_2017_A,
                                  latinx_sub_2017_B,
                                  latinx_sub_2017_C)
latinx_sub_2017
```

```
## [1] "LATINO 13.2\nLatino Male 12.4\nLatina Female 13.9\nSOUTH AMERICAN 8.4\nSouth American Male 9.1\n
```

We change string pattern to Male instead of Female to fix the type in the latinx data.

```r
latinx_sub_2017 <- latinx_sub_2017 |>
  str_replace(pattern = "DR, Cuban Female 15.7\nPR",
              replacement = "DR, Cuban Male 15.7\nPR")
```

We apply make_rows() to the latinx sub 2017 data and clean it using clean_table().

```r
latinx_sub_2017 <- make_rows(latinx_sub_2017)
latinx_sub_2017 <- clean_table(table = latinx_sub_2017)
```

```
## Warning: Expected 2 pieces. Missing pieces filled with `NA` in 1 rows [19].
```

We create a new function to clean the table for 2018 data that will be similar to clean_table() with a few modifications.

```r
# revised clean table function for 2018
clean_table_2018 <- function(table){
  table |>
    separate(col = value,
             into = c("Group", "Percent"),
             sep =  "(?<=[[:alpha:]])\\s:\\s|\\s(?=[0-9])") |>
    mutate(Group = str_remove(string = Group,
                              pattern = ":")) |>
    drop_na() |>
    mutate(Group = str_to_title(string = Group)) |>
    mutate(Percent = str_remove(string = Percent,
                                pattern = "\\.")) |>
    mutate(Percent = as.numeric(Percent)) |>
    mutate(Percent = Percent * 0.1) |>
    mutate(Race_Ethnicity = str_replace(string = Group,
                                        pattern = "Men|Women",
                                        replacement = "missing")) |>
    mutate(Race_Ethnicity = na_if(Race_Ethnicity, "missing")) |>
    fill(Race_Ethnicity, .direction = "down") |>
    mutate(Gender = str_extract(string = Group,
                                pattern = "Men|Women")) |>
    mutate(Gender = replace_na(Gender, replace = "All"))
}
```

We apply clean_table_2018 to the asian sub 2018 data.

```r
asian_sub_2018 <- clean_table_2018(asian_sub_2018)
```

```
## Warning: Expected 2 pieces. Missing pieces filled with `NA` in 6 rows [4, 8, 15,
## 19, 21, 23].
```

We add 3 rows with the following values.

```r
asian_sub_2018 <- asian_sub_2018 |>
  add_row(Group = "Asian", Percent = 6.2,
          Race_Ethnicity = "Asian", Gender = "All") |>
```

```
  add_row(Group = "Asian", Percent = 6.4,
          Race_Ethnicity = "Asian", Gender = "Men") |>
  add_row(Group = "Asian", Percent = 6.1,
          Race_Ethnicity = "Asian", Gender = "Women")
```

```
asian_sub_2018
```

```
## # A tibble: 20 x 4
##    Group      Percent Race_Ethnicity Gender
##    <chr>        <dbl> <chr>          <chr>
##  1 Chinese        4.1 Chinese        All
##  2 Men            4.5 Chinese        Men
##  3 Women          3.7 Chinese        Women
##  4 Indian         5.4 Indian         All
##  5 Men            4.7 Indian         Men
##  6 Women          6.1 Indian         Women
##  7 Korean         5.5 Korean         All
##  8 Men            5.6 Korean         Men
##  9 Women          5.4 Korean         Women
## 10 Vietnamese     6.3 Vietnamese     All
## 11 Men            7.6 Vietnamese     Men
## 12 Women          5   Vietnamese     Women
## 13 Filipino       6.8 Filipino       All
## 14 Men            6.3 Filipino       Men
## 15 Women          7.4 Filipino       Women
## 16 Hmong         10.2 Hmong          All
## 17 Cambodian     13.8 Cambodian      All
## 18 Asian          6.2 Asian          All
## 19 Asian          6.4 Asian          Men
## 20 Asian          6.1 Asian          Women
```

Add year column with corresponding year values for 2017 and 2018 data.

```
asian_sub_2017 <- asian_sub_2017 |>
  mutate(Year = 2017)
asian_sub_2018 <- asian_sub_2018 |>
  mutate(Year = 2018)
```

We convert Men to Male and Women to Female to keep it consistent throughout the datasets.

```
asian_sub_2018 <- asian_sub_2018 |>
  mutate(across(.cols = c(Gender, Group),
                ~ str_replace(string = .,
                              pattern = "Men",
                              replacement = "Male")),
         across(.cols = c(Gender, Group),
                ~ str_replace(string = .,
                              pattern = "Women",
                              replacement = "Female")))
```

We combine the asian 2017 and 2018 datasets.

```
asian_subgroups <- bind_rows(asian_sub_2017, asian_sub_2018)
```

We add in NA values to account for the cases that only have one value for a group.

```
asian_subgroups <- asian_subgroups |>
  select(-Group) |>
  pivot_wider(names_from = Year,
              values_from = Percent) |>
  pivot_longer(cols = -c(Race_Ethnicity, Gender),
               names_to = "Year",
               values_to= "Percent")
```

We clean the latinx sub 2018 data using the steps we used previously for the asian data.

```
# cleaning - wash, rinse, repeat
latinx_sub_2018 <- str_replace_all(string = latinx_sub_2018,
                                 pattern = "\\s:\n{2}|\n{2}", #remove two newline characters
                                 replacement = " ")
latinx_sub_2018 <- make_rows(latinx_sub_2018 )
latinx_sub_2018 <- clean_table_2018(latinx_sub_2018)
```

```
## Warning: Expected 2 pieces. Missing pieces filled with `NA` in 1 rows [12].
```

We create a new function, fix_latinx_naming(), to specify the naming of the Race_Ethnicity values.

```
fix_latinx_naming <- function(table){
  table |>
  mutate(Group = str_replace(string = Group,
                             pattern = "Pr, Dr, Cuban",
                             replacement = "Puerto Rican, Dominican, Cuban"),
          Race_Ethnicity = str_replace(string = Race_Ethnicity,
                                       pattern = "Pr, Dr, Cuban",
                                       replacement = "Puerto Rican, Dominican, Cuban"))
}
latinx_sub_2017 <- fix_latinx_naming(latinx_sub_2017)
latinx_sub_2018 <- fix_latinx_naming(latinx_sub_2018)
```

We add 3 new rows with the following values.

```
latinx_sub_2018 <- latinx_sub_2018 |>
  add_row(Group = "Latinx", Percent = 12.8,
          Race_Ethnicity = "Latinx", Gender = "All") |>
  add_row(Group = "Latinx", Percent = 12.3,
          Race_Ethnicity = "Latinx", Gender = "Men") |>
  add_row(Group = "Latinx", Percent = 13.3,
          Race_Ethnicity = "Latinx", Gender = "Women")
```

We recode the gender values so it's Male and Female instead of Men and Women.

```
latinx_sub_2018 <- latinx_sub_2018 |>
  mutate(across(.cols = c(Gender, Group),
                ~ str_replace(string = ., pattern = "Men", replacement = "Male")),
         across(.cols = c(Gender, Group),
                ~ str_replace(string = ., pattern = "Women", replacement = "Female")))
```

We add year values to indicate which data is from 2017 and 2018. We combine the latinx 2017 and 2018 datasets.

```
latinx_sub_2017 <- latinx_sub_2017 |>
  mutate(Year = 2017)
latinx_sub_2018 <- latinx_sub_2018 |>
```

```
  mutate(Year = 2018)
latinx_subgroups <- bind_rows(latinx_sub_2017, latinx_sub_2018)
```

We add the missing categories.

```
latinx_subgroups <- latinx_subgroups |>
  select(-Group) |>
  pivot_wider(names_from = Year, values_from = Percent) |>
  pivot_longer(cols = -c(Race_Ethnicity, Gender),
               names_to ="Year" ,
               values_to="Percent")
```

```
major_groups
```

```
## # A tibble: 108 x 5
##    Group         Race_Ethnicity Gender  Year Percent
##    <chr>         <chr>          <chr>  <dbl>   <dbl>
##  1 United States All_races      All     2008    12.6
##  2 United States All_races      All     2010    14.7
##  3 United States All_races      All     2012    14.1
##  4 United States All_races      All     2014    13.2
##  5 United States All_races      All     2016    11.7
##  6 United States All_races      All     2017    11.5
##  7 Male          All_races      Male    2008    12.3
##  8 Male          All_races      Male    2010    15.2
##  9 Male          All_races      Male    2012    14.5
## 10 Male          All_races      Male    2014    13.3
## # ... with 98 more rows
```

Saving Data

We save the 3 wrangled data so we don't have to rerun the cleaning the code when we want to use them.

```
save(major_groups, asian_subgroups, latinx_subgroups, file = "data/wrangled_data.rda")
readr::write_csv(major_groups, file = "data/wrangled_major_groups_data.csv")
readr::write_csv(asian_subgroups, file = "data/wrangled_asian_subgroups_data.csv")
readr::write_csv(latinx_subgroups, file = "data/wrangled_latinx_subgroups_data.csv")
```

Loading Data

We load the 3 wrangled data.

```
major_groups <- read_csv("data/wrangled_major_groups_data.csv")
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   Group = col_character(),
##   Race_Ethnicity = col_character(),
##   Gender = col_character(),
##   Year = col_double(),
##   Percent = col_double()
## )
```

```
asian_subgroups <- read_csv("data/wrangled_asian_subgroups_data.csv")
```

```
##
```

```
## -- Column specification --------------------------------------------------------
## cols(
##   Race_Ethnicity = col_character(),
##   Gender = col_character(),
##   Year = col_double(),
##   Percent = col_double()
## )
```

```
latinx_subgroups <- read_csv("data/wrangled_latinx_subgroups_data.csv")
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   Race_Ethnicity = col_character(),
##   Gender = col_character(),
##   Year = col_double(),
##   Percent = col_double()
## )
```

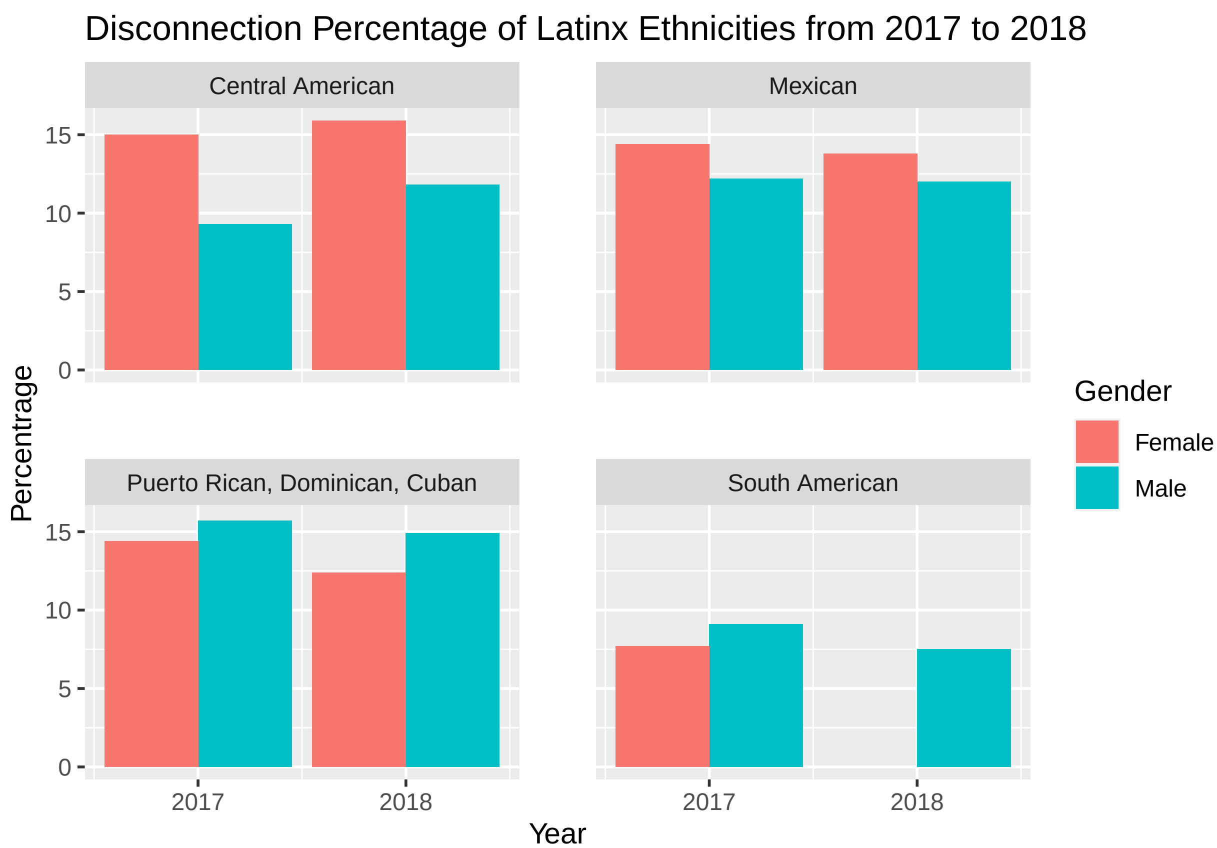
EDA

Latinx visualization

We filter out Latinx, Latina, Latino, Other Latina, and Other Latino from Race_Ethnicity because there are missing data for the percents. We create 4 bar plots for each ethnicity with Year on the x axis and Percentage on the y axis, and create bars by gender.

```
latinx_subgroups |>
  #Filtering out the race and ethnicity where there is missing data for percents.
  filter(Race_Ethnicity != "Latinx" &
           Race_Ethnicity != "Latina" &
           Race_Ethnicity != "Latino" &
           Race_Ethnicity != "Other Latina" &
           Race_Ethnicity != "Other Latino",
         Gender != "All") |>
  ggplot(
      aes(x = Year, y = Percent, fill = Gender)) +
  geom_bar(stat = "identity",
           position = position_dodge()) +
  facet_wrap(~ Race_Ethnicity) +
  scale_x_continuous(breaks = seq(2017, 2018)) +
  theme(panel.spacing = unit(2, "lines")) + #https://stackoverflow.com/questions/3681647/ggplot-how-to-
  labs(
    x = "Year",
    y = "Percentrage",
    title = "Disconnection Percentage of Latinx Ethnicities from 2017 to 2018",
    color = "Ethnicity"
  )
```

```
## Warning: Removed 1 rows containing missing values (geom_bar).
```



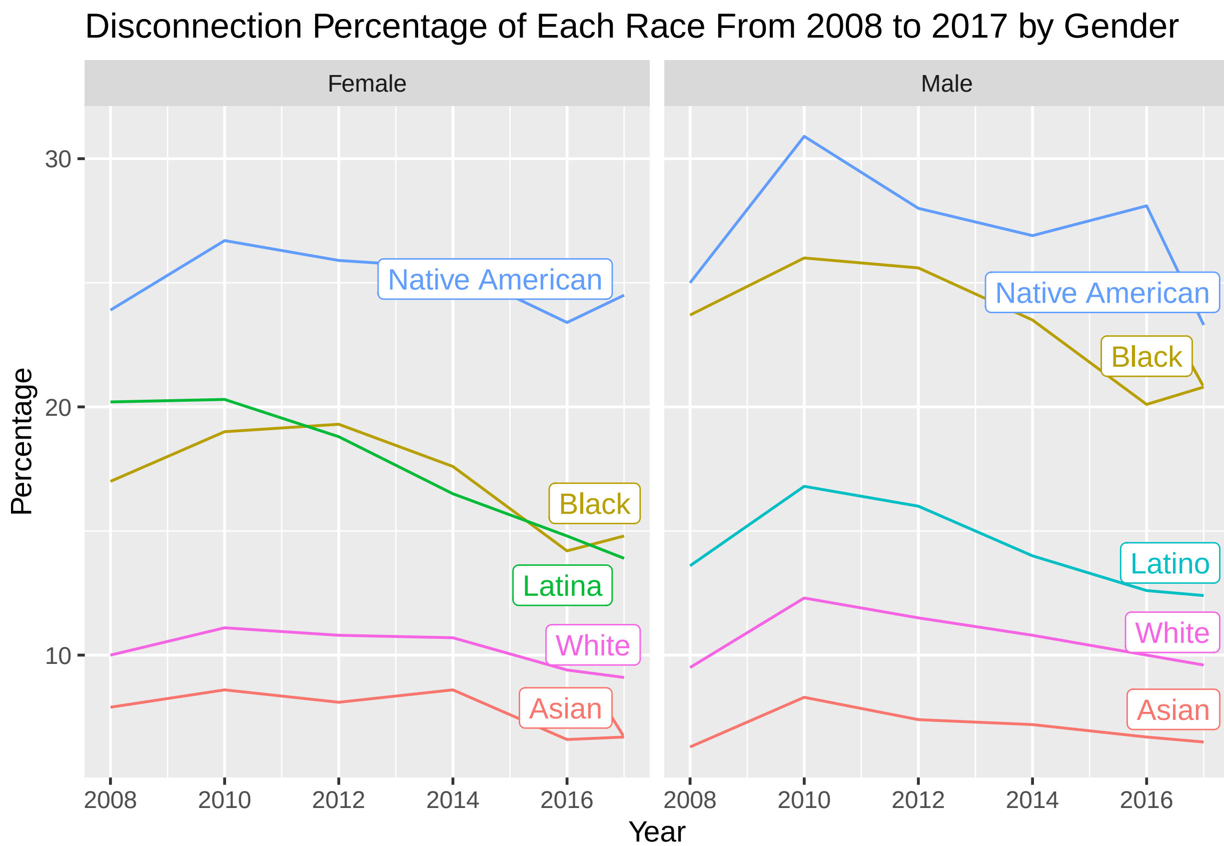
The above graph shows the disconnection Percentage of each ethnicity in the Latinx Group From 2017 to 2018 faceted by ethnicity. We can see that Central American ethnicity has a larger difference between gender compared to other ethnicities. In addition, for the Central American ethnicity, we can see that their disconnection percentage increased from 2017 to 2018, but other ethnicities decreased in disconnection percentage. Due to some missing data, this graph only shows the percentage for South American Males in 2018.

Major visualization

We filter out All_races for Race_Ethnicity, All for Gender, and United States for Group. We create 2 line plots for each gender with Year on the x axis and Percent on the y axis and create lines based on Race_Ethnicity.

```
major_groups |> filter(Race_Ethnicity != 'All_races', Gender != 'All', Group != 'United States') |>
  mutate(label = if_else(Year == max(Year), as.character(Race_Ethnicity), NA_character_)) |>
  ggplot(aes(x = Year, y = Percent, color = Race_Ethnicity)) +
  geom_line(size = 0.5) +
  facet_wrap(Gender~.) +
  labs(title = 'Disconnection Percentage of Each Race From 2008 to 2017 by Gender',
       y = 'Percentage',
       x = 'Year',
       color = 'Race Ethnicity') +
  scale_x_continuous(breaks=seq(2008,2018,by=2)) +
  #https://stackoverflow.com/questions/29357612/plot-labels-at-ends-of-lines
  # Directly labels plot lines at the end of the line
  geom_label_repel(aes(label = label),
                   nudge_x = 1,
```

```r
                    nudge_y = 0.2,
                    na.rm = T) +
  # Remove the legend
  scale_color_discrete(guide = "none")
```



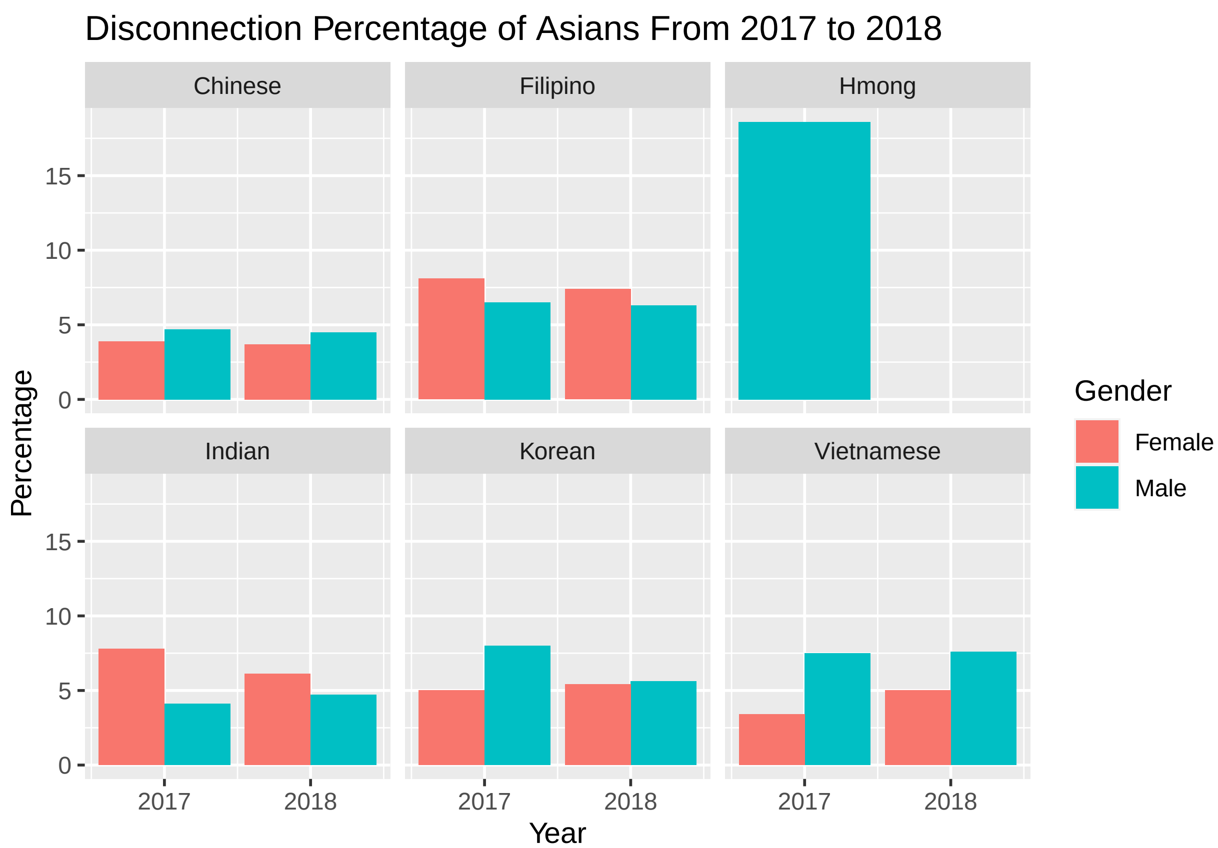
The above graph shows the disconnection of each race from 2008 to 2017 faceted by gender. We see that the Black and Native American have larger differences of disconnection percentage between Female and male. We can also learn that Males of all race except Latino have higher percentage of disconnection than females. We also noticed that all races of both genders except Black Female reached the highest percentage of disconnection in 2010 and had a slow decrease after this year.

Asian visualization

We filter out All_races and Asian for Race_Ethnicity and All for Gender. We create 6 bar plots for each ethnicity with Year on the x axis and Percent on the y axix, and create bars based on gender.

```r
asian_subgroups |> filter(Race_Ethnicity != 'All_races', Race_Ethnicity != 'Asian', Gender != 'All') |>
  ggplot(aes(x = Year, y = Percent, fill = Gender)) +
  geom_bar(position="dodge", stat="identity") +
  facet_wrap(Race_Ethnicity~.) +
  labs(title = 'Disconnection Percentage of Asians From 2017 to 2018',
       y = 'Percentage',
       x = 'Year',
       color = 'Gender') +
  scale_x_continuous(breaks=seq(2017,2019,by=1))
```

```
## Warning: Removed 1 rows containing missing values (geom_bar).
```



The above graph shows the disconnection Percentage of each ethnicity in the Asian Group From 2017 to 2018 faceted by ethnicity. We can see that Indian, Korean and Vietnamese have larger differences between gender.

The disconnection percentage trends between 2017 and 2018 for each ethnicity are the following:

- Chinese: decreased for both genders
- Filipino: decreased for both genders
- Indian: decreased for females, increased for males
- Korean: increased for females, decreased for males
- Vietnamese: increased for females, remained relatively equal for males.

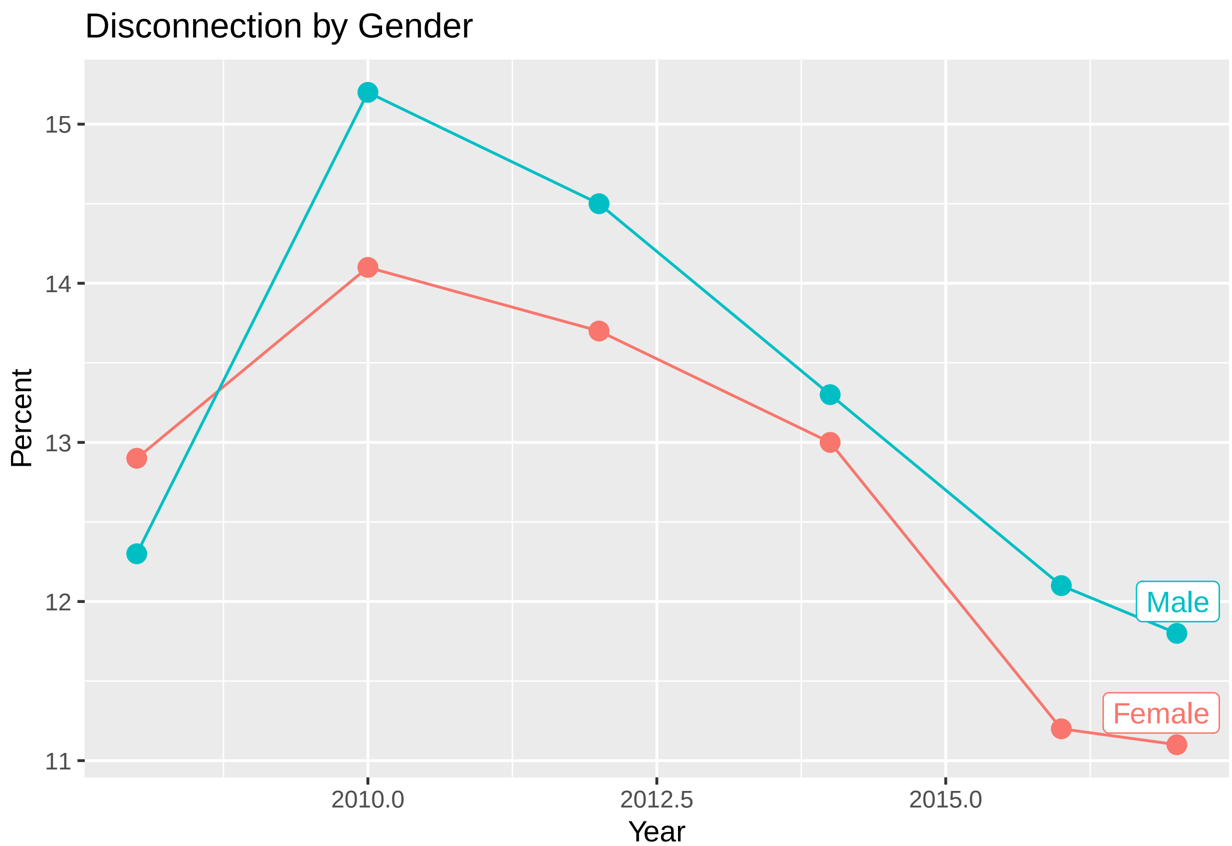
Due to some missing data, this graph only shows the percentage for Hmomg Male in 2017.

Disconnection by Gender

```
major_groups |>
  filter(Gender != "All", Race_Ethnicity == "All_races") |>
  mutate(label = if_else(Year == max(Year), as.character(Gender), NA_character_)) |>
  ggplot(aes(x = Year, y = Percent, color = Gender)) +
    geom_line(size = 0.5) +
    geom_point(size = 3) + labs(title="Disconnection by Gender") + scale_color_manual(values=c("#ef8a62
  #https://stackoverflow.com/questions/29357612/plot-labels-at-ends-of-lines
  # Directly labels plot lines at the end of the line
  geom_label_repel(aes(label = label),
                   nudge_x = 1,
                   nudge_y = 0.2,
                   na.rm = T) +
```

```
  # Remove the legend
  scale_color_discrete(guide = "none")
```

```
## Scale for 'colour' is already present. Adding another scale for 'colour',
## which will replace the existing scale.
```

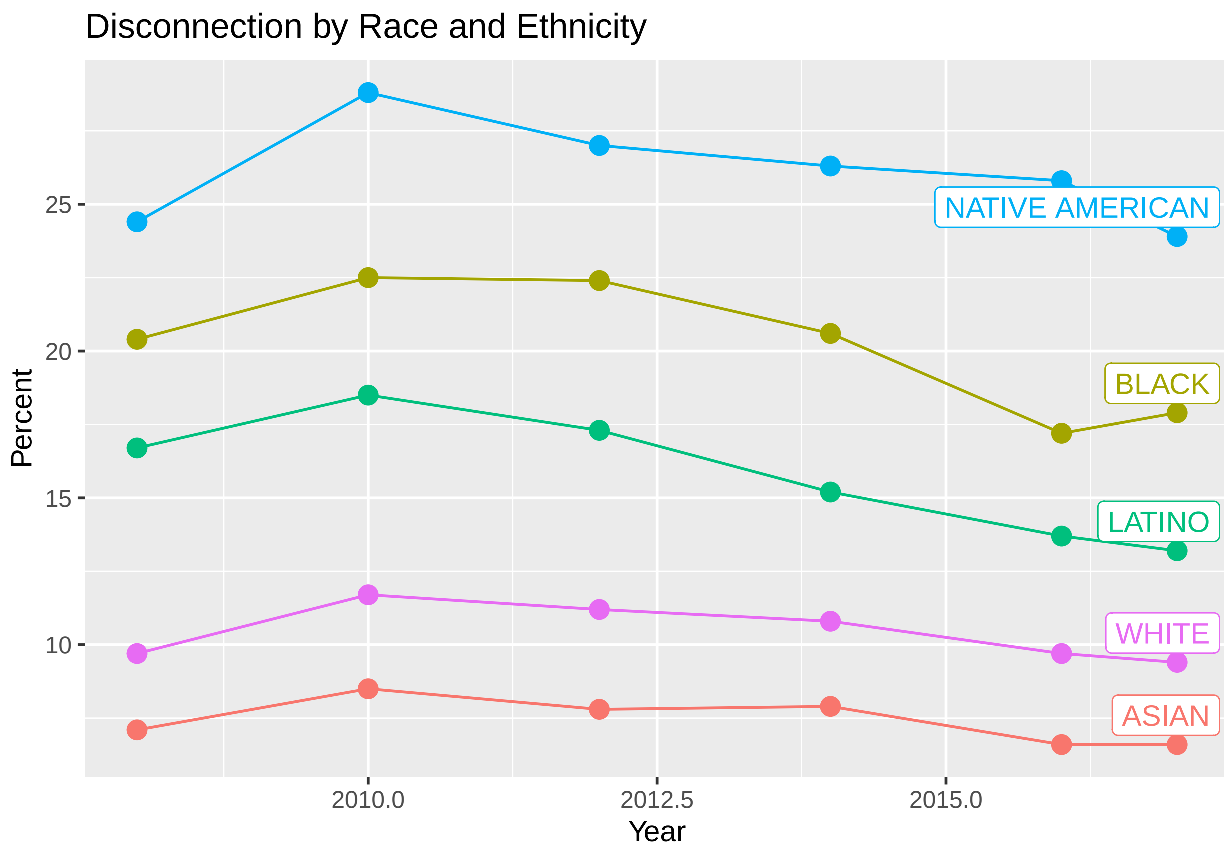


The above visualization shows the percentages of disconnection by gender overtime. For both genders, the percentages increase from 2008 and 2010, but decreases from 2010-2018. In 2008, the female percentage of disconnection was greater than male, but between 2008-2010, the male percentage of disconnection was greater than female. From 2010 onwards, the male percentage of disconnection was greater than female.

Disconnection by Race

```
major_groups |>
  filter(Gender == "All", Group != "United States") |>
  mutate(label = if_else(Year == max(Year), as.character(Race_Ethnicity), NA_character_)) |>
  ggplot(aes(x = Year, y = Percent, color = Race_Ethnicity)) +
    geom_line(size = 0.5) +
    geom_point(size = 3) + labs(title="Disconnection by Race and Ethnicity") +
  #https://stackoverflow.com/questions/29357612/plot-labels-at-ends-of-lines
  # Directly label plot lines to match data with visualization easier since theres multiple lines
  geom_label_repel(aes(label = label),
                   nudge_x = 2,
                   nudge_y = 1,
                   na.rm = T) +
  # Remove the legend
```

```
  scale_color_discrete(guide = "none")
```



The above visualization shows the disconnection percentages of different races between 2008-2018. From 2008 to 2010, there was an increase in percentage for all ethnicities. From 2010 to 2018, the percentages of disconnection for most races slowly decreased, with the exception of Black youth, who's percentage increased from 2016 to 2018.

Extend the Analysis

We will perform linear regression for each race to find the relationship between the Year and Percentages in order to observe the trends of disconnection percentages. This will help us predict the disconnection percentages of each race in upcoming years and investigate the relationship between disconnection percentages and the disconnection percentage rates for each race.

Asian

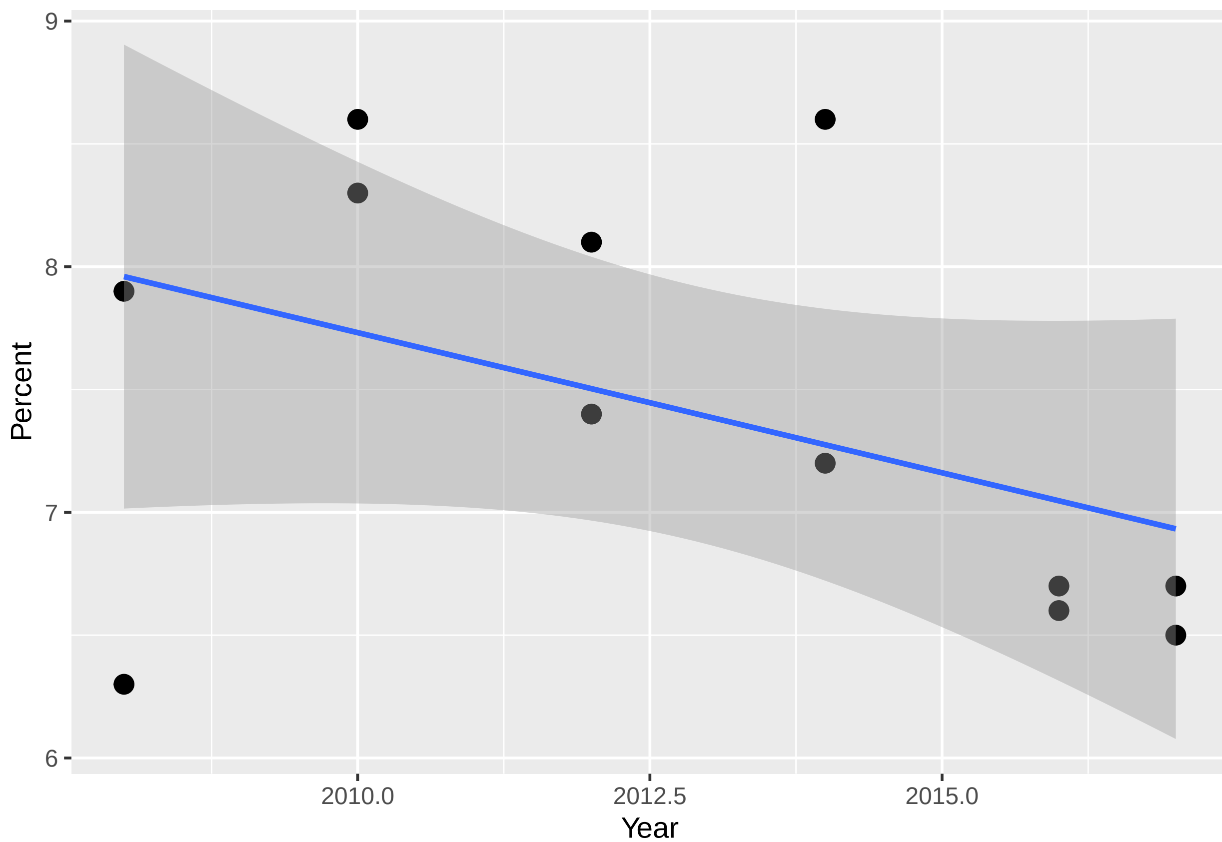
```
asian_data <-
major_groups |>
filter(Race_Ethnicity=="Asian")

linear_reg() |>
set_engine("lm") |>
fit(Percent ~ Year, data=(asian_data)) |>
tidy()
```

```
## # A tibble: 2 x 5
```

```
##   term        estimate std.error statistic p.value
##   <chr>          <dbl>     <dbl>     <dbl>   <dbl>
## 1 (Intercept)  237.      147.          1.61   0.139
## 2 Year          -0.114    0.0732     -1.56   0.150
```

```
major_groups |>
  filter(Race_Ethnicity == "Asian") |>
  ggplot(aes(x = Year, y = Percent)) +
    geom_point(size = 3) +
    geom_smooth(method = "lm")
```

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



In each year by Asian race, they're expected on average to be 0.114% decrease in youth disconnection percentage rate.

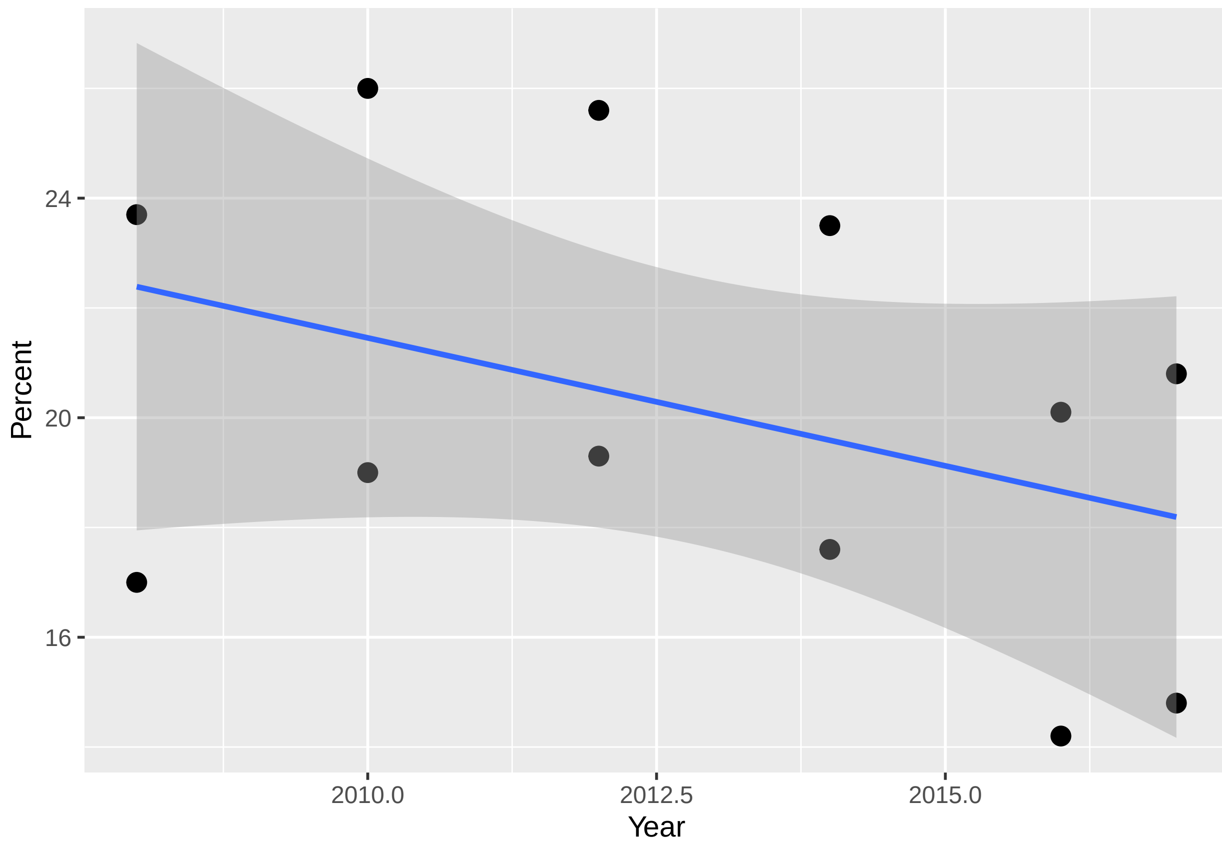
Black

```
black_data <-
major_groups |>
filter(Race_Ethnicity=="Black")

linear_reg() |>
set_engine("lm") |>
fit(Percent ~ Year, data=(black_data)) |>
tidy()
```

```
## # A tibble: 2 x 5
##   term        estimate std.error statistic p.value
##   <chr>          <dbl>     <dbl>     <dbl>   <dbl>
## 1 (Intercept)  959.      693.         1.38   0.197
## 2 Year          -0.466     0.344     -1.35   0.205
```

```
major_groups |>
  filter(Race_Ethnicity == "Black") |>
  ggplot(aes(x = Year, y = Percent)) +
    geom_point(size = 3) +
    geom_smooth(method = "lm")
```

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



In each year by Black race, they're expected on average to be 0.466% decrease in youth disconnection percentage rate.

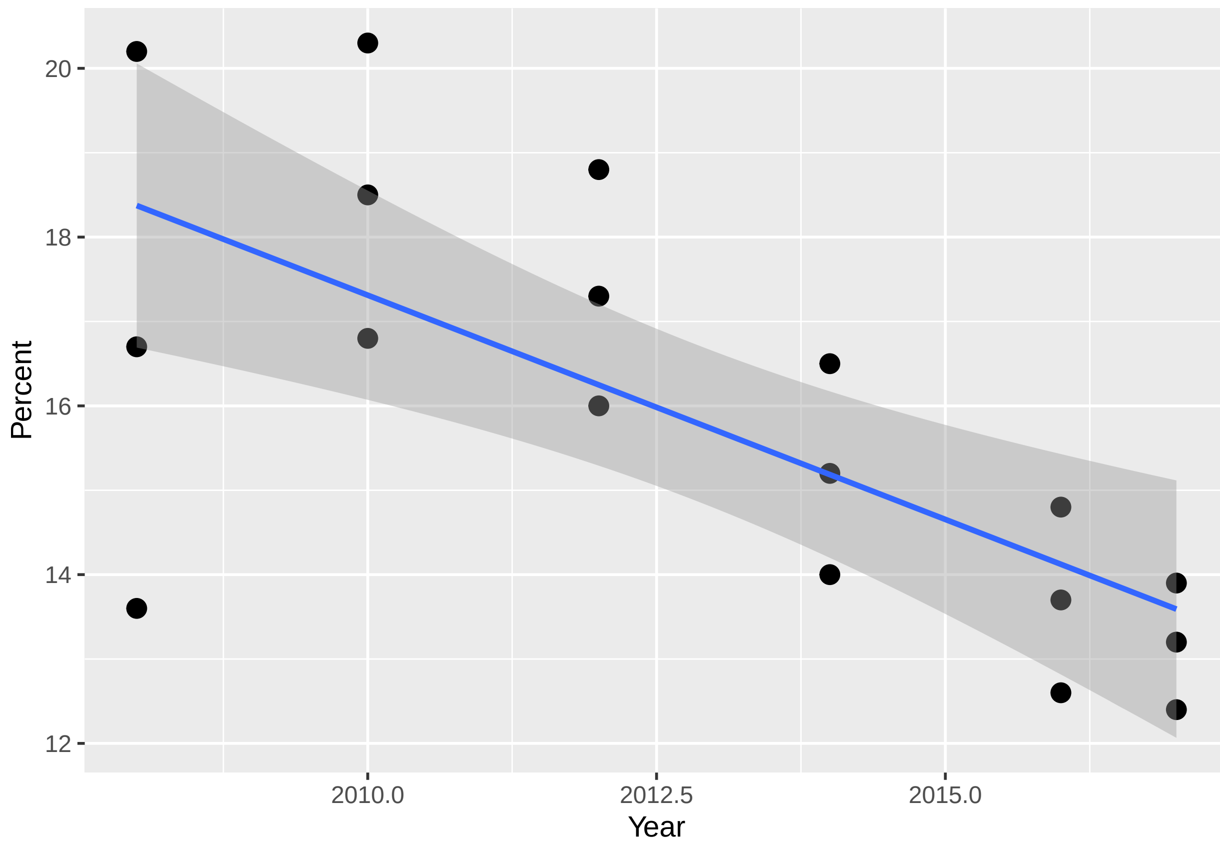
Latinx

```
latinx_data <-
major_groups |>
filter(Race_Ethnicity=="Latino" | Race_Ethnicity=="Latina" | Race_Ethnicity=="LATINO")

linear_reg() |>
set_engine("lm") |>
fit(Percent ~ Year, data=(latinx_data)) |>
tidy()
```

```
## # A tibble: 2 x 5
##   term        estimate std.error statistic p.value
##   <chr>          <dbl>     <dbl>     <dbl>   <dbl>
## 1 (Intercept) 1085.      276.         3.93 0.00119
## 2 Year          -0.531     0.137     -3.87 0.00135
```

```
major_groups |>
  filter(Race_Ethnicity=="Latino" | Race_Ethnicity=="Latina" | Race_Ethnicity=="LATINO") |>
  ggplot(aes(x = Year, y = Percent)) +
    geom_point(size = 3) +
    geom_smooth(method = "lm")
```

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



In each year by Latino race, they're expected on average to be 0.531% decrease in youth disconnection percentage rate.

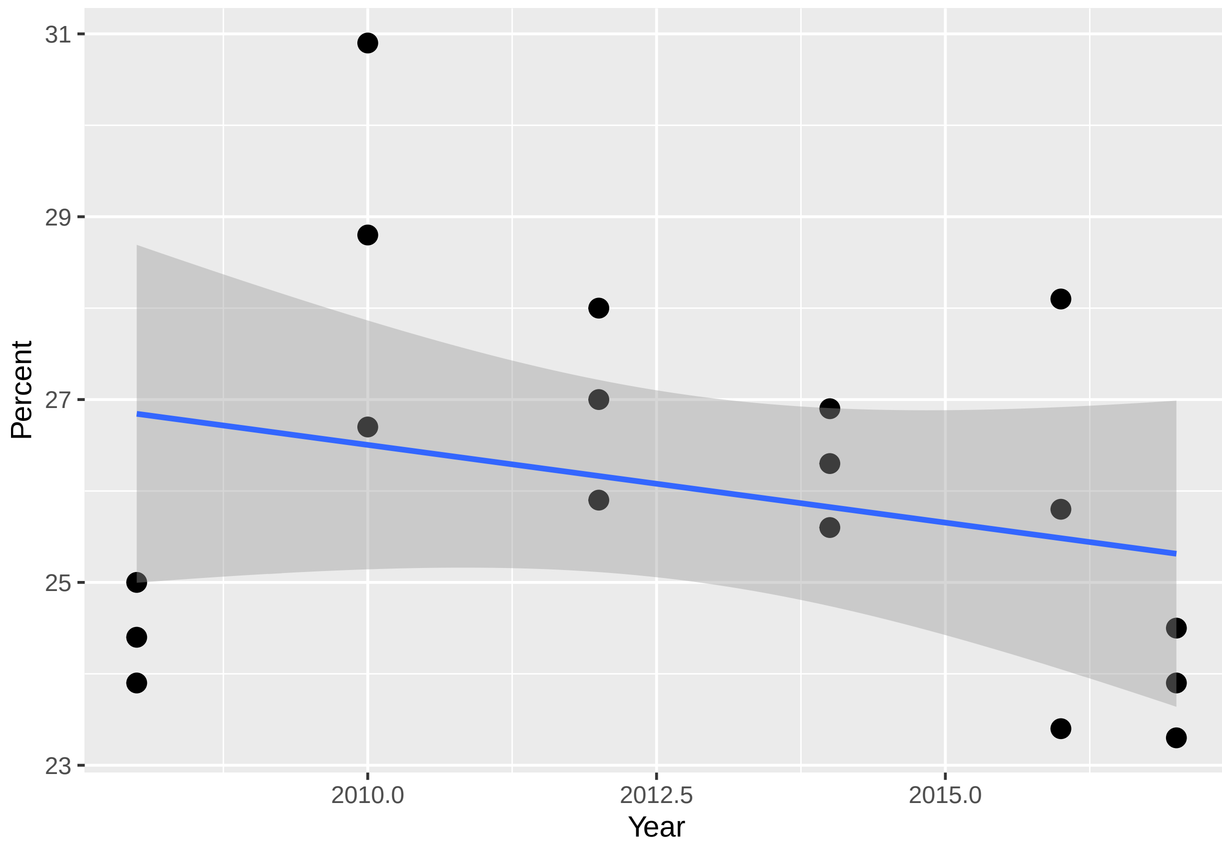
Native American

```
na_data <-
major_groups |>
filter(Race_Ethnicity=="Native American" | Race_Ethnicity=="NATIVE AMERICAN")

linear_reg() |>
set_engine("lm") |>
fit(Percent ~ Year, data=(na_data)) |>
tidy()
```

```
## # A tibble: 2 x 5
##   term        estimate std.error statistic p.value
##   <chr>          <dbl>     <dbl>     <dbl>   <dbl>
## 1 (Intercept)  368.      303.         1.21   0.242
## 2 Year          -0.170     0.151     -1.13   0.276
```

```
major_groups |>
  filter(Race_Ethnicity=="Native American" | Race_Ethnicity=="NATIVE AMERICAN") |>
  ggplot(aes(x = Year, y = Percent)) +
    geom_point(size = 3) +
    geom_smooth(method = "lm")
```

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



In each year by Native American race, they're expected on average to be 0.170% decrease in youth disconnection percentage rate.

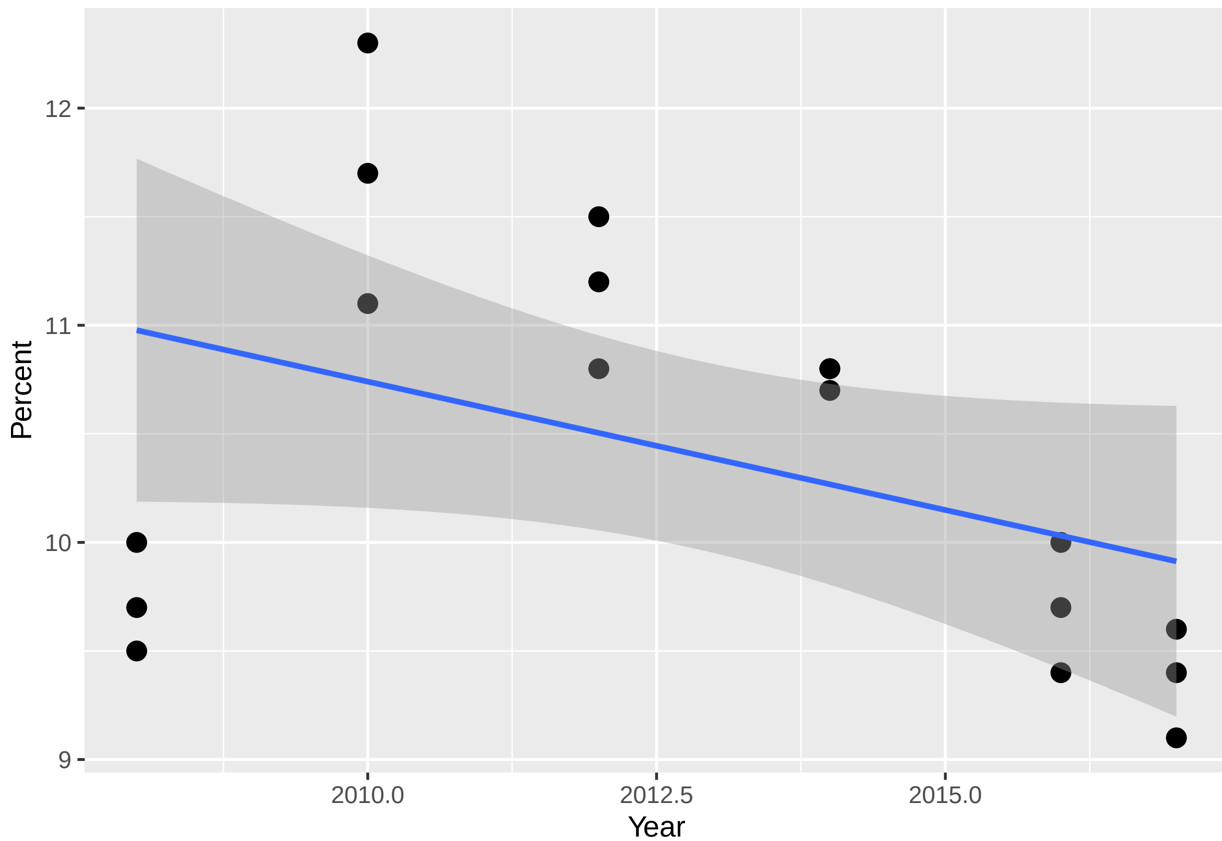
White

```
white_data <-
major_groups |>
filter(Race_Ethnicity=="White" | Race_Ethnicity=="WHITE")

linear_reg() |>
set_engine("lm") |>
fit(Percent ~ Year, data=(white_data)) |>
tidy()
```

```
## # A tibble: 2 x 5
##   term        estimate std.error statistic p.value
##   <chr>          <dbl>     <dbl>     <dbl>   <dbl>
## 1 (Intercept)  248.     130.          1.92  0.0732
## 2 Year          -0.118    0.0644     -1.84  0.0848
```

```
major_groups |>
  filter(Race_Ethnicity=="White" | Race_Ethnicity=="WHITE") |>
  ggplot(aes(x = Year, y = Percent)) +
    geom_point(size = 3) +
    geom_smooth(method = "lm")
```

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



In each year by White race, they're expected on average to be 0.118% decrease in youth disconnection percentage rate.

Results of Extended Analysis

Native American has the highest percentage disconnection, who has a decrease percentage rate in the middle (.17%).

Black has the second highest percentage disconnection, who has the second highest decrease percentage rate (.466%).

Latinx is in the middle in terms of percentage disconnection, who has the highest decrease percentage rate (.531%).

White has the second lowest percentage disconnection, who has the second lowest decrease percentage rate

(.118%).

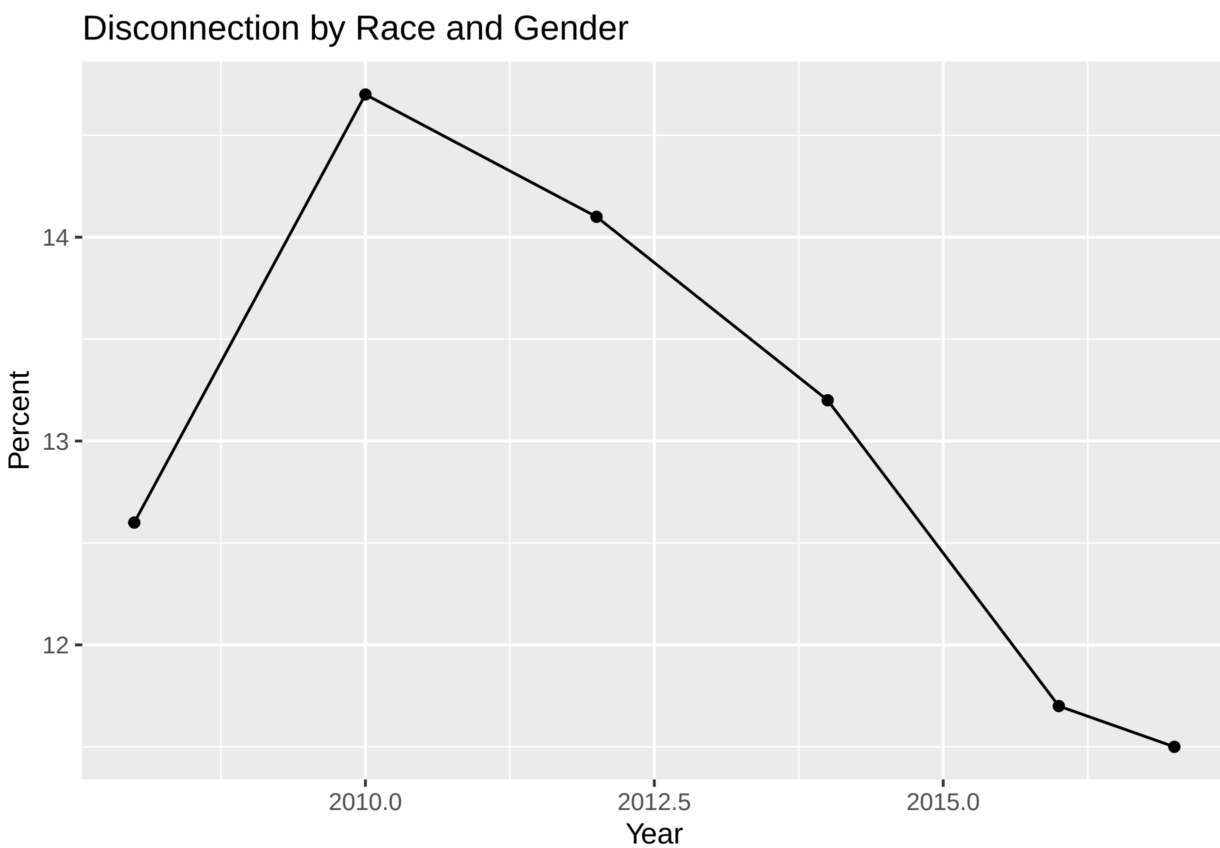
Asian has the lowest percentage disconnection, who has the lowest decrease percentage rate (.114%).

Discussion of Extended Analysis

Some of the trends we observed between percentage disconnection and decrease percentage rate made sense. Asians and Whites, for example, had an already relatively low percentage disconnection, so the low decrease percentage rate was reasonable. However, this trend starts to break, with Latinx (middle percentage disconnection) having the highest decrease percentage rate and Native Americans (highest percentage disconnection) having a middle decrease percentage rate. This is most likely due to the unique circumstances that each race has, such as their environment and income.

Disconnection by Race and Gender

```
major_groups |>
  filter(Gender == "All", Race_Ethnicity == "All_races") |>
  ggplot(., mapping=aes(
    x=Year,
    y=Percent
  )) +
  geom_point() +
  geom_line() + labs(title="Disconnection by Race and Gender")
```



The above visualization shows the percentages for races and genders between 2008 and 2018. Similar to what we saw in the previous graphs, there's an increase between 2008 and 2010, and a decrease from 2010 - 2018.

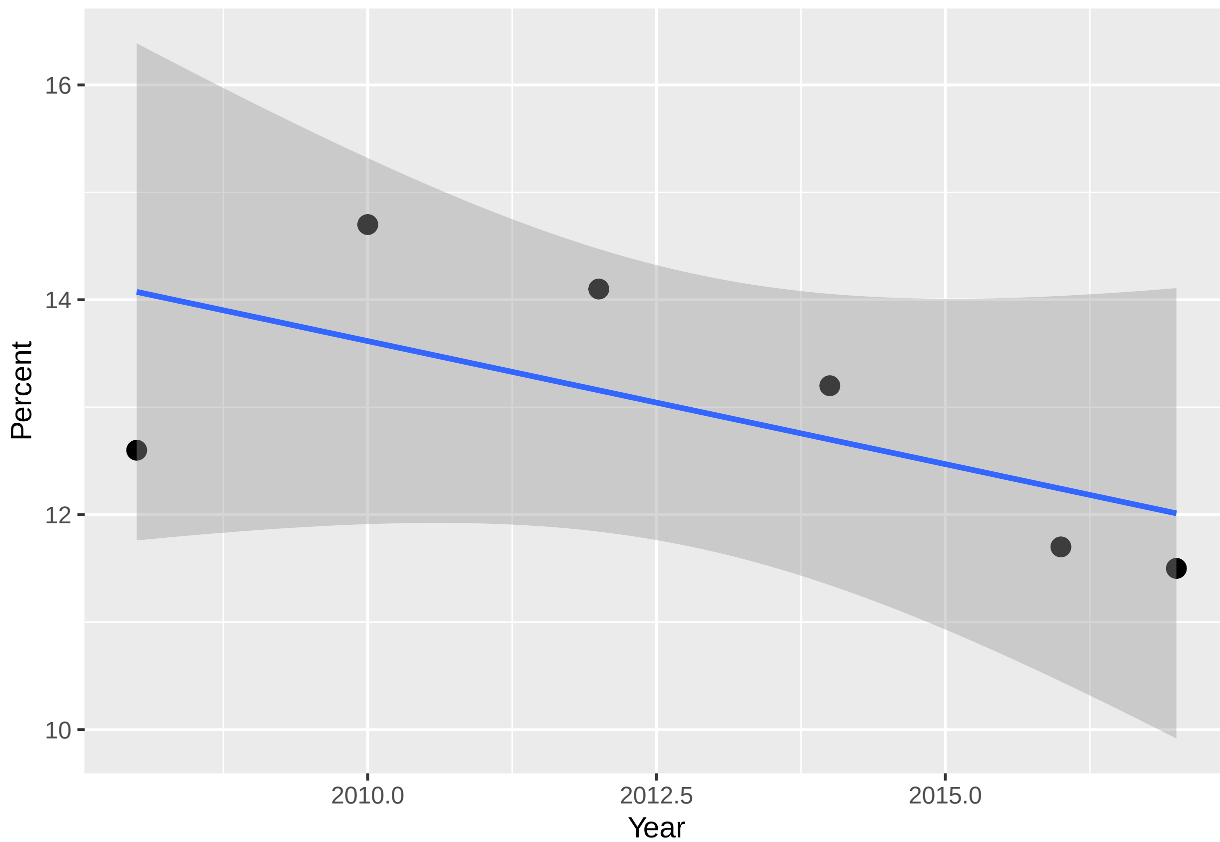
Data Analysis

Linear Regression

We apply the parametric linear regression model to determine the strength of the relationship between Year and Percent

```
# linear regression
major_groups |>
  filter(Gender == "All", Race_Ethnicity == "All_races") |>
  ggplot(aes(x = Year, y = Percent)) +
    geom_point(size = 3) +
    geom_smooth(method = "lm")
```

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



The grey area represents the expected range at each point across the plot of where the true value would fall.

Mann-Kendall

We apply the nonparametric Mann-Kendall model to test if there's a monotonic association over time.

H0 (null hypothesis): Data are not consistently increasing/decreasing (no monotonic trend)

Ha (alternative hypothesis): Data are consistently increasing/decreasing (there is a monotonic trend)

```
# mann kendall test
major_groups |> # instead of whole major group, sub-categorize by race and present different Mann-Kenda
  filter(Gender == "All", Race_Ethnicity == "All_races") |>
  pull(Percent) |>
```

```
  MannKendall() |>
  tidy()
```

```
## # A tibble: 1 x 5
##   statistic p.value kendall_score denominator var_kendall_score
##       <dbl>   <dbl>         <dbl>       <dbl>             <dbl>
## 1    -0.600   0.133            -9        15.0              28.3
```

Results

For the linear regression model, the standard error range (grey area) is relatively large compared to the actual range for the points.

For the Mann-Kendall test, we got an S score of -9 and a p-value of 0.133.

Discussion of Results

For the linear regression model, since the standard error range is relatively larger compared to the actual range for the points, it's not accurate enough for us to be able to answer our question.

For the Mann-Kendall test, we got an S score of -9, which indicates a possible downward trend but it isn't large enough to conclude there is monotonicity. For p-value, we got a value of 0.133. This is greater than 0.05, indicating it is not statistically significant and we fail to reject the null hypothesis, so there is strong evidence that the data has no monotonic trend.

Conclusion

Based on our results, we conclude that there is no underlying trend of disconnection percentages for different ethnic groups over the years. Additionally, for different genders and ethnic groups, we found that the disconnection percentage increases from 2008-2010, but decreases from 2010-2018. The most disconnected race is Native Americans, followed by Black, Latino, White, and Asian. This order of disconnection remains the same throughout the years.

Furthermore, based on our extended analysis, we discovered that the percentage of disconnection for Latinx group has the highest decrease percentage rate (.531%); Asian has the lowest decrease percentage rate (.114%). This suggests that Asian group is less likely to see major improvement on decreasing the disconnection in upcoming years while the Latinx group is more likely to have a major improvement on reduction of disconnection rate in upcoming years.

References

(1) https://measureofamerica.org/disconnected-youth/