Assignment3_IDMP

Mansi Pravin Thanki

2023-02-26

```r
install.packages("tidyverse", repos = "http://cran.us.r-project.org")
```

```
##
## The downloaded binary packages are in
##  /var/folders/r9/2cgj8871421bvklfhk05xfzc0000gn/T//RtmpHr3eyS/downloaded_packages
```

```r
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.0     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.4.1     v tibble    3.1.8
## v lubridate 1.9.2     v tidyr     1.3.0
## v purrr     1.0.1
## -- 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
```

```r
library(readr)
library(dplyr)
```

Part A

Problems 1–3 use data from the US Department of Education's Civil Rights Data Collection. It was downloaded from the zipped 2017-2018 data available at https://www2.ed.gov/about/offices/list/ocr/docs/crdc-2017-18.html. The Public Use Data File User's Manual and a spreadsheet describing the file structure are included in the zipped files, or can be downloaded at the same location. Use these as a reference to help you understand the dataset. The CRDC data is supplemented by statistical data from EDFacts (not included). We will use only the CRDC data. Import all CRDC reserve codes as missing values.

Loading dataset. Imported all CRDC reserve codes as missing values.

```r
enrollment_dataset <- read_csv("/Users/mansipravinthanki/Downloads/2017-18-crdc-data-corrected-publicati
## Rows: 97632 Columns: 123
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (11): LEA_STATE, LEA_STATE_NAME, LEAID, LEA_NAME, SCHID, SCH_NAME, COMB...
## dbl (112): SCH_PSENR_HI_M, SCH_PSENR_HI_F, SCH_PSENR_AM_M, SCH_PSENR_AM_F, S...
##
## i Use ‘spec()‘ to retrieve the full column specification for this data.
## i Specify the column types or set ‘show_col_types = FALSE‘ to quiet this message.
```

```
as.tibble(enrollment_dataset)
```

```
## Warning: ‘as.tibble()‘ was deprecated in tibble 2.0.0.
## i Please use ‘as_tibble()‘ instead.
## i The signature and semantics have changed, see ‘?as_tibble‘.
```

```
## # A tibble: 97,632 x 123
##    LEA_STATE LEA_STA~1 LEAID LEA_N~2 SCHID SCH_N~3 COMBO~4 JJ    SCH_P~5 SCH_P~6
##    <chr>     <chr>     <chr> <chr>   <chr> <chr>   <chr>   <chr> <chr>   <chr>
##  1 AL        ALABAMA   0100~ Alabam~ 01705 Wallac~ 010000~ Yes   <NA>    <NA>
##  2 AL        ALABAMA   0100~ Alabam~ 01706 McNeel~ 010000~ Yes   <NA>    <NA>
##  3 AL        ALABAMA   0100~ Alabam~ 01876 Alabam~ 010000~ No    <NA>    <NA>
##  4 AL        ALABAMA   0100~ Alabam~ 99995 AUTAUG~ 010000~ Yes   <NA>    <NA>
##  5 AL        ALABAMA   0100~ Albert~ 00870 Albert~ 010000~ No    <NA>    <NA>
##  6 AL        ALABAMA   0100~ Albert~ 00871 Albert~ 010000~ No    <NA>    <NA>
##  7 AL        ALABAMA   0100~ Albert~ 00879 Evans ~ 010000~ No    <NA>    <NA>
##  8 AL        ALABAMA   0100~ Albert~ 00889 Albert~ 010000~ No    <NA>    <NA>
##  9 AL        ALABAMA   0100~ Albert~ 01616 Big Sp~ 010000~ No    <NA>    <NA>
## 10 AL        ALABAMA   0100~ Albert~ 02150 Albert~ 010000~ No    Yes     Yes
## # ... with 97,622 more rows, 113 more variables: SCH_PSENR_NONIDEA_A5 <chr>,
## #   SCH_PSENR_HI_M <dbl>, SCH_PSENR_HI_F <dbl>, SCH_PSENR_AM_M <dbl>,
## #   SCH_PSENR_AM_F <dbl>, SCH_PSENR_AS_M <dbl>, SCH_PSENR_AS_F <dbl>,
## #   SCH_PSENR_HP_M <dbl>, SCH_PSENR_HP_F <dbl>, SCH_PSENR_BL_M <dbl>,
## #   SCH_PSENR_BL_F <dbl>, SCH_PSENR_WH_M <dbl>, SCH_PSENR_WH_F <dbl>,
## #   SCH_PSENR_TR_M <dbl>, SCH_PSENR_TR_F <dbl>, TOT_PSENR_M <dbl>,
## #   TOT_PSENR_F <dbl>, SCH_PSENR_LEP_M <dbl>, SCH_PSENR_LEP_F <dbl>, ...
```

Problem 1

We would like to know the distribution of students by race and gender across all schools. Calculate and visualize the overall proportions of enrolled students of every race and gender combination out of the total number of students across all schools. Describe the distribution.

Calculate the total number of students across all schools

```
total_males <- enrollment_dataset$TOT_ENR_M[!is.na(enrollment_dataset$TOT_ENR_M)]
total_females <- enrollment_dataset$TOT_ENR_F[!is.na(enrollment_dataset$TOT_ENR_F)]
total_students <- sum(total_males, total_females)
paste("The total number of enrolled students across all schools are", total_students)
```

```
## [1] "The total number of enrolled students across all schools are 50922401"
```

Tidying the data to help generate the visualization

get all the SCH_ENR_RACE_GENDER columns

```
# first  get all the columns that start with "SCH_ENR_"
untidy_columns_df <- enrollment_dataset %>% select(all_of(starts_with("SCH_ENR_")))

# get the column names from the untidy_columns_df dataframe
racegender_columnNames <- sort(colnames(untidy_columns_df))
racegender_columnNames
```

```
##  [1] "SCH_ENR_504_F"  "SCH_ENR_504_M"  "SCH_ENR_AM_F"   "SCH_ENR_AM_M"
##  [5] "SCH_ENR_AS_F"   "SCH_ENR_AS_M"   "SCH_ENR_BL_F"   "SCH_ENR_BL_M"
##  [9] "SCH_ENR_HI_F"   "SCH_ENR_HI_M"   "SCH_ENR_HP_F"   "SCH_ENR_HP_M"
## [13] "SCH_ENR_IDEA_F" "SCH_ENR_IDEA_M" "SCH_ENR_LEP_F"  "SCH_ENR_LEP_M"
## [17] "SCH_ENR_TR_F"   "SCH_ENR_TR_M"   "SCH_ENR_WH_F"   "SCH_ENR_WH_M"
```

using tidying techniques pivot_longer, str_sub and filter to get Race, Gender

and Count columns

```
# using pivot_longer
enr_race_dataset <-pivot_longer(enrollment_dataset, cols=racegender_columnNames, names_to = "Race",
                           values_to = "Count")
```

```
## Warning: Using an external vector in selections was deprecated in tidyselect 1.1.0.
## i Please use ‘all_of()‘ or ‘any_of()‘ instead.
##   # Was:
##   data %>% select(racegender_columnNames)
##
##   # Now:
##   data %>% select(all_of(racegender_columnNames))
##
## See <https://tidyselect.r-lib.org/reference/faq-external-vector.html>.
```

```
# using str_sub to extract Race from "SCH_ENR_Race_Gender"
enr_race_dataset$Gender <- str_sub(enr_race_dataset$Race, start = -1, end = -1)

# filtering out the columns that do not include race
enr_race_dataset <- filter(enr_race_dataset, !Race %in% c("SCH_ENR_504_F",
                                                          "SCH_ENR_504_M",
                                                          "SCH_ENR_IDEA_F",
                                                          "SCH_ENR_IDEA_M",
                                                          "SCH_ENR_LEP_F",
                                                          "SCH_ENR_LEP_M"))
```

```
# using str_sub to extract Gender from "SCH_ENR_Race_Gender" and
# create a Gender column from it
enr_race_dataset$Race <- str_sub(enr_race_dataset$Race, start = -4, end = -3)

# selecting the columns out of tidied dataset
enr_race_dataset <- select(enr_race_dataset, SCHID, SCH_NAME, COMBOKEY, Race, Gender, Count, TOT_ENR_M,

as.tibble(enr_race_dataset)
```

```
## # A tibble: 1,366,848 x 8
##    SCHID SCH_NAME                     COMBO~1 Race  Gender Count TOT_E~2 TOT_E~3
##    <chr> <chr>                        <chr>   <chr> <chr>  <dbl>   <dbl>   <dbl>
##  1 01705 Wallace Sch - Mt Meigs Camp~ 010000~ AM    F          0     133       0
##  2 01705 Wallace Sch - Mt Meigs Camp~ 010000~ AM    M          2     133       0
##  3 01705 Wallace Sch - Mt Meigs Camp~ 010000~ AS    F          0     133       0
##  4 01705 Wallace Sch - Mt Meigs Camp~ 010000~ AS    M          0     133       0
##  5 01705 Wallace Sch - Mt Meigs Camp~ 010000~ BL    F          0     133       0
##  6 01705 Wallace Sch - Mt Meigs Camp~ 010000~ BL    M         72     133       0
##  7 01705 Wallace Sch - Mt Meigs Camp~ 010000~ HI    F          0     133       0
##  8 01705 Wallace Sch - Mt Meigs Camp~ 010000~ HI    M          5     133       0
##  9 01705 Wallace Sch - Mt Meigs Camp~ 010000~ HP    F          0     133       0
## 10 01705 Wallace Sch - Mt Meigs Camp~ 010000~ HP    M          0     133       0
## # ... with 1,366,838 more rows, and abbreviated variable names 1: COMBOKEY,
## #   2: TOT_ENR_M, 3: TOT_ENR_F
```

Grouping by each race and gender and calculating the proportion out of all enrolled students

```
race_gender_prop_dataset <- enr_race_dataset %>%
    group_by(Race, Gender) %>%
    summarise(Race_count = sum(Count[!is.na(Count)]))
```

```
## ‘summarise()‘ has grouped output by ’Race’. You can override using the
## ‘.groups‘ argument.
```

```
race_gender_prop_dataset$Proportion <- race_gender_prop_dataset$Race_count/total_students
as.tibble(race_gender_prop_dataset)
```

```
## # A tibble: 14 x 4
##    Race  Gender Race_count Proportion
##    <chr> <chr>       <dbl>      <dbl>
##  1 AM    F          245129    0.00481
##  2 AM    M          257342    0.00505
##  3 AS    F         1281702    0.0252
##  4 AS    M         1344407    0.0264
##  5 BL    F         3763447    0.0739
##  6 BL    M         3933267    0.0772
##  7 HI    F         6763088    0.133
##  8 HI    M         7099395    0.139
##  9 HP    F           93838    0.00184
## 10 HP    M           99586    0.00196
## 11 TR    F          957267    0.0188
## 12 TR    M          987608    0.0194
## 13 WH    F        11646416    0.229
## 14 WH    M        12449909    0.244
```

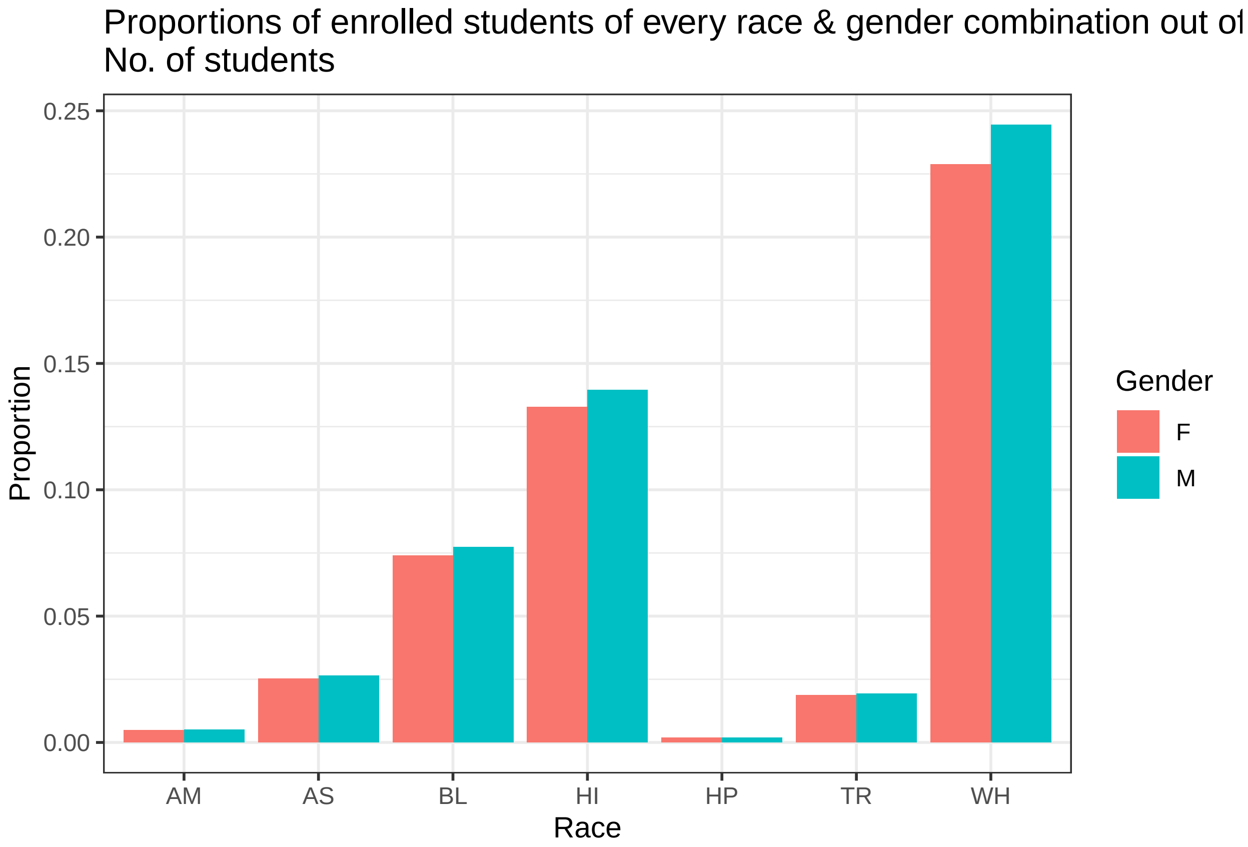
if we sum up all the proportions, it sums up to 1

```
sum(race_gender_prop_dataset$Proportion)
```

```
## [1] 1
```

Visualizing the graph

```
library(ggplot2)
ggplot(race_gender_prop_dataset, aes(x = Race, y = Proportion, fill = Gender)) +
  geom_bar(position="dodge",stat = "identity") +
  labs(x = "Race", y = "Proportion", fill = "Gender") +
  ggtitle("Proportions of enrolled students of every race & gender combination out of total
No. of students ") +
theme_bw()
```



Observations:

1. Out of all enrolled students across all schools, the students from **White (WH) race** constitute the **maximum proportion for both male and female** genders
2. The **'Native Hawaiian or Other Pacific Islander'(HP)** race students constitute the **lowest proportion for both male and female** genders.
3. Male Vs Female comparison for the races:

- **Male** student population is significantly **higher** for the races: **White, Hispanic and Black**.
- **Male** student population is very **slightly higher** for the **Asian (AS) and Two or More raced (TR)**.
- There is equal distribution for male and female students for the races **Native American (AM) and 'Native Hawaiian or Other Pacific Islander'(HP)**

#Q2

We would like to know the distribution of Advanced Placement (AP) students (i.e., students enrolled in at least one AP course) by race and gender across all schools. Filter the data to include only schools with AP programs. Calculate and visualize the overall proportions of AP students of every race and gender combination out of the total number of AP students across all schools. Describe the distribution. How does it compare to the distribution from Problem 1?

Loading dataset. Imported all CRDC reserve codes as missing values.

```
ap_dataset <- read_csv("/Users/mansipravinthanki/Downloads/2017-18-crdc-data-corrected-publication 2/20
```

```
## Rows: 97632 Columns: 134
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (13): LEA_STATE, LEA_STATE_NAME, LEAID, LEA_NAME, SCHID, SCH_NAME, COMB...
## dbl (121): SCH_APCOURSES, SCH_APENR_HI_M, SCH_APENR_HI_F, SCH_APENR_AM_M, SC...
##
## i Use ‘spec()‘ to retrieve the full column specification for this data.
## i Specify the column types or set ‘show_col_types = FALSE‘ to quiet this message.
```

```
as.tibble(ap_dataset)
```

```
## # A tibble: 97,632 x 134
##    LEA_STATE LEA_STA~1 LEAID LEA_N~2 SCHID SCH_N~3 COMBO~4 JJ    SCH_A~5 SCH_A~6
##    <chr>     <chr>     <chr> <chr>   <chr> <chr>   <chr>   <chr> <chr>     <dbl>
##  1 AL        ALABAMA   0100~ Alabam~ 01705 Wallac~ 010000~ Yes   <NA>         NA
##  2 AL        ALABAMA   0100~ Alabam~ 01706 McNeel~ 010000~ Yes   <NA>         NA
##  3 AL        ALABAMA   0100~ Alabam~ 01876 Alabam~ 010000~ No    No           NA
##  4 AL        ALABAMA   0100~ Alabam~ 99995 AUTAUG~ 010000~ Yes   <NA>         NA
##  5 AL        ALABAMA   0100~ Albert~ 00870 Albert~ 010000~ No    <NA>         NA
##  6 AL        ALABAMA   0100~ Albert~ 00871 Albert~ 010000~ No    Yes           8
##  7 AL        ALABAMA   0100~ Albert~ 00879 Evans ~ 010000~ No    <NA>         NA
##  8 AL        ALABAMA   0100~ Albert~ 00889 Albert~ 010000~ No    <NA>         NA
##  9 AL        ALABAMA   0100~ Albert~ 01616 Big Sp~ 010000~ No    <NA>         NA
## 10 AL        ALABAMA   0100~ Albert~ 02150 Albert~ 010000~ No    <NA>         NA
## # ... with 97,622 more rows, 124 more variables: SCH_APSEL <chr>,
## #   SCH_APENR_HI_M <dbl>, SCH_APENR_HI_F <dbl>, SCH_APENR_AM_M <dbl>,
## #   SCH_APENR_AM_F <dbl>, SCH_APENR_AS_M <dbl>, SCH_APENR_AS_F <dbl>,
## #   SCH_APENR_HP_M <dbl>, SCH_APENR_HP_F <dbl>, SCH_APENR_BL_M <dbl>,
## #   SCH_APENR_BL_F <dbl>, SCH_APENR_WH_M <dbl>, SCH_APENR_WH_F <dbl>,
## #   SCH_APENR_TR_M <dbl>, SCH_APENR_TR_F <dbl>, TOT_APENR_M <dbl>,
## #   TOT_APENR_F <dbl>, SCH_APENR_LEP_M <dbl>, SCH_APENR_LEP_F <dbl>, ...
```

filtering out the schools having AP Programs by using SCH_APENR_IND as an indicator whether school has AP program or not

```
ap_dataset <- filter(ap_dataset, SCH_APENR_IND == "Yes")
as.tibble(ap_dataset)
```

```
## # A tibble: 13,809 x 134
##    LEA_STATE LEA_STA~1 LEAID LEA_N~2 SCHID SCH_N~3 COMBO~4 JJ    SCH_A~5 SCH_A~6
##    <chr>     <chr>     <chr> <chr>   <chr> <chr>   <chr>   <chr> <chr>     <dbl>
##  1 AL        ALABAMA   0100~ Albert~ 00871 Albert~ 010000~ No    Yes           8
##  2 AL        ALABAMA   0100~ Marsha~ 00872 Asbury~ 010000~ No    Yes            3
##  3 AL        ALABAMA   0100~ Marsha~ 00878 Dougla~ 010000~ No    Yes            6
##  4 AL        ALABAMA   0100~ Marsha~ 00883 Kate D~ 010000~ No    Yes            6
##  5 AL        ALABAMA   0100~ Marsha~ 01585 Brindl~ 010000~ No    Yes            5
##  6 AL        ALABAMA   0100~ Hoover~ 00251 Hoover~ 010000~ No    Yes           15
##  7 AL        ALABAMA   0100~ Hoover~ 01456 Spain ~ 010000~ No    Yes           16
##  8 AL        ALABAMA   0100~ Madiso~ 00831 Bob Jo~ 010000~ No    Yes           16
##  9 AL        ALABAMA   0100~ Madiso~ 02198 James ~ 010000~ No    Yes           19
## 10 AL        ALABAMA   0100~ Leeds ~ 02096 Leeds ~ 010001~ No    Yes            7
## # ... with 13,799 more rows, 124 more variables: SCH_APSEL <chr>,
## #   SCH_APENR_HI_M <dbl>, SCH_APENR_HI_F <dbl>, SCH_APENR_AM_M <dbl>,
## #   SCH_APENR_AM_F <dbl>, SCH_APENR_AS_M <dbl>, SCH_APENR_AS_F <dbl>,
## #   SCH_APENR_HP_M <dbl>, SCH_APENR_HP_F <dbl>, SCH_APENR_BL_M <dbl>,
## #   SCH_APENR_BL_F <dbl>, SCH_APENR_WH_M <dbl>, SCH_APENR_WH_F <dbl>,
## #   SCH_APENR_TR_M <dbl>, SCH_APENR_TR_F <dbl>, TOT_APENR_M <dbl>,
## #   TOT_APENR_F <dbl>, SCH_APENR_LEP_M <dbl>, SCH_APENR_LEP_F <dbl>, ...
```

Calculating the total number of AP students across all schools

```r
total_apmales <- ap_dataset$TOT_APENR_M[!is.na(ap_dataset$TOT_APENR_M)]
total_apfemales <- ap_dataset$TOT_APENR_F[!is.na(ap_dataset$TOT_APENR_F)]
total_apstudents <- sum(total_apmales, total_apfemales)
paste("The total number of enrolled AP students across all schools are", total_apstudents)
```

```
## [1] "The total number of enrolled AP students across all schools are 3030991"
```

Tidying the data to help generate the visualization

get all the SCH_ENR_RACE_GENDER columns

```r
# first  get all the columns that start with "SCH_APENR_"
untidy_ap_columns_df <- ap_dataset %>% select(all_of(starts_with("SCH_APENR_")))

# get the column names from the untidy_ap_columns_df dataframe
ap_racegender_columnNames <- sort(colnames(untidy_ap_columns_df))
ap_racegender_columnNames <- ap_racegender_columnNames[!ap_racegender_columnNames %in% c("SCH_APENR_IND
ap_racegender_columnNames
```

```
##  [1] "SCH_APENR_AM_F"   "SCH_APENR_AM_M"   "SCH_APENR_AS_F"   "SCH_APENR_AS_M"  
##  [5] "SCH_APENR_BL_F"   "SCH_APENR_BL_M"   "SCH_APENR_HI_F"   "SCH_APENR_HI_M"  
##  [9] "SCH_APENR_HP_F"   "SCH_APENR_HP_M"   "SCH_APENR_IDEA_F" "SCH_APENR_IDEA_M"
## [13] "SCH_APENR_LEP_F"  "SCH_APENR_LEP_M"  "SCH_APENR_TR_F"   "SCH_APENR_TR_M"  
## [17] "SCH_APENR_WH_F"   "SCH_APENR_WH_M"
```

using tidying techniques pivot_longer, str_sub and filter to get Race, Gender

and Count columns

```
ap_enr_race_dataset <-pivot_longer(ap_dataset, cols=ap_racegender_columnNames, names_to = "Race",
                           values_to = "APCount")
```

```
## Warning: Using an external vector in selections was deprecated in tidyselect 1.1.0.
## i Please use `all_of()` or `any_of()` instead.
##   # Was:
##   data %>% select(ap_racegender_columnNames)
##
##   # Now:
##   data %>% select(all_of(ap_racegender_columnNames))
##
## See <https://tidyselect.r-lib.org/reference/faq-external-vector.html>.
```

```
# using str_sub to extract year from "SCH_APENR_Race_Gender"
ap_enr_race_dataset$Gender <- str_sub(ap_enr_race_dataset$Race, start = -1, end = -1)

ap_enr_race_dataset <- filter(ap_enr_race_dataset, !Race %in% c("SCH_APENR_504_F", "SCH_APENR_504_M", "S

ap_enr_race_dataset$Race <- str_sub(ap_enr_race_dataset$Race, start = -4, end = -3)

# selecting the columns out of tidied dataset
ap_enr_race_dataset <- select(ap_enr_race_dataset, SCHID, SCH_NAME,COMBOKEY, Race, Gender, APCount, TOT_
ap_enr_race_dataset
```

```
## # A tibble: 193,326 x 8
##    SCHID SCH_NAME                COMBOKEY   Race  Gender APCount TOT_A~1 TOT_A~2
##    <chr> <chr>                   <chr>      <chr> <chr>    <dbl>   <dbl>   <dbl>
##  1 00871 Albertville High School 010000500~ AM    F            1     121     170
##  2 00871 Albertville High School 010000500~ AM    M            0     121     170
##  3 00871 Albertville High School 010000500~ AS    F            2     121     170
##  4 00871 Albertville High School 010000500~ AS    M            3     121     170
##  5 00871 Albertville High School 010000500~ BL    F            5     121     170
##  6 00871 Albertville High School 010000500~ BL    M            1     121     170
##  7 00871 Albertville High School 010000500~ HI    F           47     121     170
##  8 00871 Albertville High School 010000500~ HI    M           36     121     170
##  9 00871 Albertville High School 010000500~ HP    F            0     121     170
## 10 00871 Albertville High School 010000500~ HP    M            0     121     170
## # ... with 193,316 more rows, and abbreviated variable names 1: TOT_APENR_M,
## #   2: TOT_APENR_F
```

```
as.tibble(ap_enr_race_dataset)
```

```
## # A tibble: 193,326 x 8
##    SCHID SCH_NAME                COMBOKEY   Race  Gender APCount TOT_A~1 TOT_A~2
##    <chr> <chr>                   <chr>      <chr> <chr>    <dbl>   <dbl>   <dbl>
##  1 00871 Albertville High School 010000500~ AM    F            1     121     170
##  2 00871 Albertville High School 010000500~ AM    M            0     121     170
##  3 00871 Albertville High School 010000500~ AS    F            2     121     170
##  4 00871 Albertville High School 010000500~ AS    M            3     121     170
##  5 00871 Albertville High School 010000500~ BL    F            5     121     170
##  6 00871 Albertville High School 010000500~ BL    M            1     121     170
##  7 00871 Albertville High School 010000500~ HI    F           47     121     170
##  8 00871 Albertville High School 010000500~ HI    M           36     121     170
##  9 00871 Albertville High School 010000500~ HP    F            0     121     170
## 10 00871 Albertville High School 010000500~ HP    M            0     121     170
## # ... with 193,316 more rows, and abbreviated variable names 1: TOT_APENR_M,
## #   2: TOT_APENR_F
```

```
apr_enr_dataset <- ap_enr_race_dataset %>%
    group_by(Race,Gender) %>%
    summarise(Race_count = sum(APCount[!is.na(APCount)]))
```

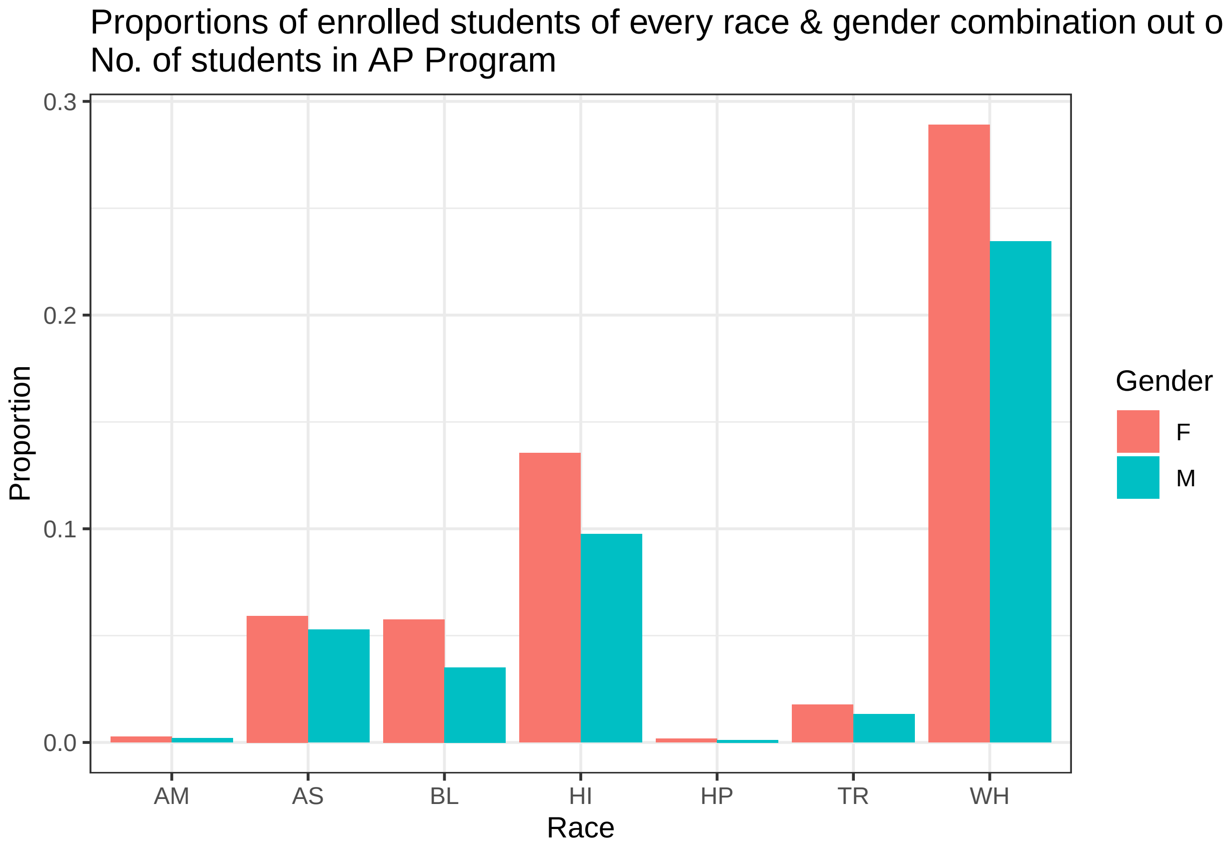
```
## `summarise()` has grouped output by 'Race'. You can override using the
## `.groups` argument.
```

```
apr_enr_dataset$Proportion <- ((apr_enr_dataset$Race_count) * 1.0)/total_apstudents
as.tibble(apr_enr_dataset)
```

```
## # A tibble: 14 x 4
##    Race  Gender Race_count Proportion
##    <chr> <chr>       <dbl>      <dbl>
##  1 AM    F            8475    0.00280
##  2 AM    M            5811    0.00192
##  3 AS    F          179533    0.0592
##  4 AS    M          160350    0.0529
##  5 BL    F          174634    0.0576
##  6 BL    M          106512    0.0351
##  7 HI    F          410834    0.136
##  8 HI    M          295258    0.0974
##  9 HP    F            5118    0.00169
## 10 HP    M            3599    0.00119
## 11 TR    F           53437    0.0176
## 12 TR    M           40209    0.0133
## 13 WH    F          876243    0.289
## 14 WH    M          710978    0.235
```

```
library(ggplot2)

ggplot(apr_enr_dataset, aes(x = Race, y = Proportion, fill = Gender)) +
  geom_bar(position="dodge",stat = "identity") +
  labs(x = "Race", y = "Proportion", fill = "Gender") +
  ggtitle("Proportions of enrolled students of every race & gender combination out of total
No. of students in AP Program ") +
  theme_bw()
```



Observations:

1. The number of female students in AP program is significantly more across all the Races.
2. There are predominantly more male and female students that belong to **White (WH)** race.
3. The least number of male and female students belong to **'Native Hawaiian or Other Pacific Islander'(HP)**
4. When compared to the distribution in Q1, you can see that in **Q1 the male dominance was higher** across all races, whereas there are **higher percentage of females across all races in Q2**.
5. The distribution across the races have differences: **Asian and Black population** is more in AP programs

#Q3

```
enr_race_dataset$Total_ENR_School <- enr_race_dataset$TOT_ENR_M + enr_race_dataset$TOT_ENR_F
as.tibble(enr_race_dataset)
```

```
## # A tibble: 1,366,848 x 9
##    SCHID SCH_NAME             COMBO~1 Race  Gender Count TOT_E~2 TOT_E~3 Total~4
##    <chr> <chr>                <chr>   <chr> <chr>  <dbl>   <dbl>   <dbl>   <dbl>
##  1 01705 Wallace Sch - Mt Me~ 010000~ AM    F          0     133       0     133
##  2 01705 Wallace Sch - Mt Me~ 010000~ AM    M          2     133       0     133
##  3 01705 Wallace Sch - Mt Me~ 010000~ AS    F          0     133       0     133
##  4 01705 Wallace Sch - Mt Me~ 010000~ AS    M          0     133       0     133
##  5 01705 Wallace Sch - Mt Me~ 010000~ BL    F          0     133       0     133
##  6 01705 Wallace Sch - Mt Me~ 010000~ BL    M         72     133       0     133
##  7 01705 Wallace Sch - Mt Me~ 010000~ HI    F          0     133       0     133
##  8 01705 Wallace Sch - Mt Me~ 010000~ HI    M          5     133       0     133
##  9 01705 Wallace Sch - Mt Me~ 010000~ HP    F          0     133       0     133
## 10 01705 Wallace Sch - Mt Me~ 010000~ HP    M          0     133       0     133
## # ... with 1,366,838 more rows, and abbreviated variable names 1: COMBOKEY,
## #   2: TOT_ENR_M, 3: TOT_ENR_F, 4: Total_ENR_School
```

```
## Sum
enr_race_dataset <- filter(enr_race_dataset, Race !="WH")
enrolled_df <- enr_race_dataset %>%
    group_by(COMBOKEY) %>%
    summarise(NonWhite_Race_count = sum(Count[!is.na(Count)]),
            TotalEnrollment = sum(Total_ENR_School[1]))
```

```
enrolled_df$ENRProportion <- enrolled_df$NonWhite_Race_count/enrolled_df$TotalEnrollment
as.tibble(enrolled_df)
```

```
## # A tibble: 97,632 x 4
##    COMBOKEY     NonWhite_Race_count TotalEnrollment ENRProportion
##    <chr>                      <dbl>           <dbl>         <dbl>
##  1 010000201705                  79             133         0.594
##  2 010000201706                  40              58         0.690
##  3 010000201876                  18              58         0.310
##  4 010000299995                  21              31         0.677
##  5 010000500870                 418             807         0.518
##  6 010000500871                 731            1449         0.504
##  7 010000500879                 486             854         0.569
##  8 010000500889                 513             906         0.566
##  9 010000501616                 243             414         0.587
## 10 010000502150                 644            1014         0.635
## # ... with 97,622 more rows
```

```
ap_enr_race_dataset$Total_AP_ENR_School <- ap_enr_race_dataset$TOT_APENR_M + ap_enr_race_dataset$TOT_AP
as.tibble(ap_enr_race_dataset)
```

```
## # A tibble: 193,326 x 9
##    SCHID SCH_NAME            COMBO~1 Race  Gender APCount TOT_A~2 TOT_A~3 Total~4
##    <chr> <chr>               <chr>   <chr> <chr>    <dbl>   <dbl>   <dbl>   <dbl>
##  1 00871 Albertville High ~ 010000~ AM    F            1     121     170     291
##  2 00871 Albertville High ~ 010000~ AM    M            0     121     170     291
##  3 00871 Albertville High ~ 010000~ AS    F            2     121     170     291
##  4 00871 Albertville High ~ 010000~ AS    M            3     121     170     291
##  5 00871 Albertville High ~ 010000~ BL    F            5     121     170     291
##  6 00871 Albertville High ~ 010000~ BL    M            1     121     170     291
##  7 00871 Albertville High ~ 010000~ HI    F           47     121     170     291
##  8 00871 Albertville High ~ 010000~ HI    M           36     121     170     291
##  9 00871 Albertville High ~ 010000~ HP    F            0     121     170     291
## 10 00871 Albertville High ~ 010000~ HP    M            0     121     170     291
## # ... with 193,316 more rows, and abbreviated variable names 1: COMBOKEY,
## #   2: TOT_APENR_M, 3: TOT_APENR_F, 4: Total_AP_ENR_School
## Sum
ap_enr_race_dataset <- filter(ap_enr_race_dataset, Race !="WH")
ap_enrolled_df <- ap_enr_race_dataset %>%
    group_by(COMBOKEY) %>%
    summarise(NonWhite_APRace_count = sum(APCount[!is.na(APCount)]),
            APTotalEnrollment = sum(Total_AP_ENR_School[1]))
```

```
ap_enrolled_df$APProportion <- ap_enrolled_df$NonWhite_APRace_count/ap_enrolled_df$APTotalEnrollment
as.tibble(ap_enrolled_df)
```

```
## # A tibble: 13,809 x 4
##    COMBOKEY     NonWhite_APRace_count APTotalEnrollment APProportion
##    <chr>                        <dbl>             <dbl>        <dbl>
##  1 010000500871                   101               291       0.347
##  2 010000600872                    14                46       0.304
##  3 010000600878                    27               246       0.110
##  4 010000600883                     6               206       0.0291
##  5 010000601585                     5               102       0.0490
##  6 010000700251                   219               534       0.410
##  7 010000701456                   159               512       0.311
##  8 010000800831                   115               284       0.405
##  9 010000802198                   137               434       0.316
## 10 010001102096                    33               118       0.280
## # ... with 13,799 more rows
```

```
enr_apenr_joined_dataset <- enrolled_df %>%
  inner_join(ap_enrolled_df, by=c("COMBOKEY"="COMBOKEY"))
as.tibble(enr_apenr_joined_dataset)
```

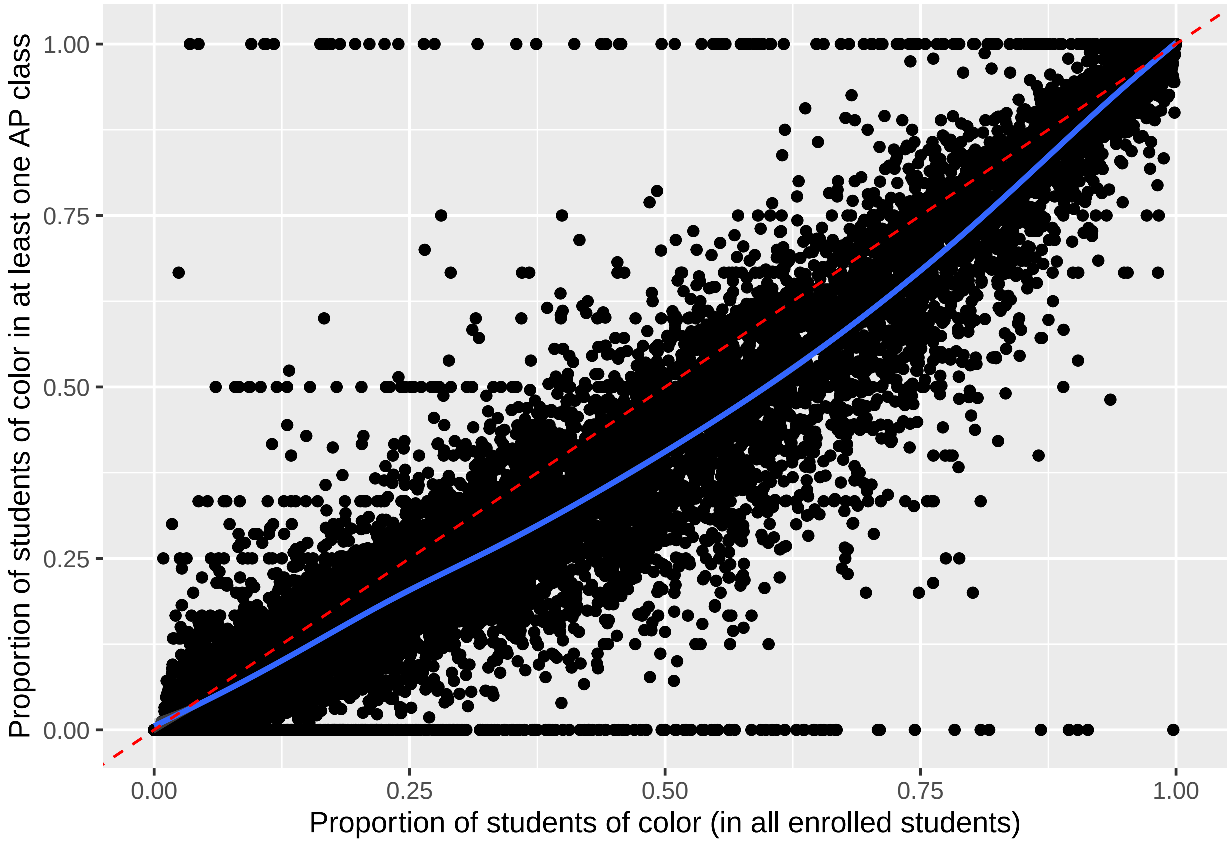
```
## # A tibble: 13,809 x 7
##    COMBOKEY     NonWhite_Race_count TotalEnrol~1 ENRPr~2 NonWh~3 APTot~4 APPro~5
##    <chr>                      <dbl>        <dbl>   <dbl>   <dbl>   <dbl>   <dbl>
##  1 010000500871                 731         1449  0.504      101     291  0.347
##  2 010000600872                 225          547  0.411       14      46  0.304
##  3 010000600878                 231          591  0.391       27     246  0.110
##  4 010000600883                  19          452  0.0420       6     206  0.0291
##  5 010000601585                  50          632  0.0791       5     102  0.0490
##  6 010000700251                1302         2886  0.451      219     534  0.410
##  7 010000701456                 657         1669  0.394      159     512  0.311
##  8 010000800831                 717         1779  0.403      115     284  0.405
##  9 010000802198                 709         1920  0.369      137     434  0.316
## 10 010001102096                 203          488  0.416       33     118  0.280
## # ... with 13,799 more rows, and abbreviated variable names 1: TotalEnrollment,
## #   2: ENRProportion, 3: NonWhite_APRace_count, 4: APTotalEnrollment,
## #   5: APProportion
```

```
# Create a scatter plot with a smooth line and a reference line with slope 1
ggplot(enr_apenr_joined_dataset, aes(x = ENRProportion, y = APProportion)) +
  geom_point() +
  geom_smooth() +
  geom_abline(slope = 1, linetype = "dashed", color="red") +
  labs(x = "Proportion of students of color (in all enrolled students)",
       y = "Proportion of students of color in at least one AP class")
```

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

```
## Warning: Removed 5 rows containing non-finite values (‘stat_smooth()‘).
```

```
## Warning: Removed 5 rows containing missing values (‘geom_point()‘).
```



Observations:

1. To answer the question: Are students of color typically underrepresented in AP classes?

- **Yes**, since a large number of points lie below the reference line (the intercept), it means that the students of color are typically underrepresented in AP classes.
- There is a positive correlation relationship between the proportion of student of colors (in all schools) to the proportion of non-white students in atleast one AP class

```
library(RSQLite)

# connect to the SQLite database
dbConnection <- dbConnect(RSQLite::SQLite(), dbname = "/Users/mansipravinthanki/Downloads/DBLP-CSR-sqli
```

#Q4 Filter the data to include only the authors for whom a gender was predicted as ‘male’ or ‘female’ with a probability of 0.90 or greater, and then visualize the total number of distinct male and female authors published each year. Comment on the visualization.

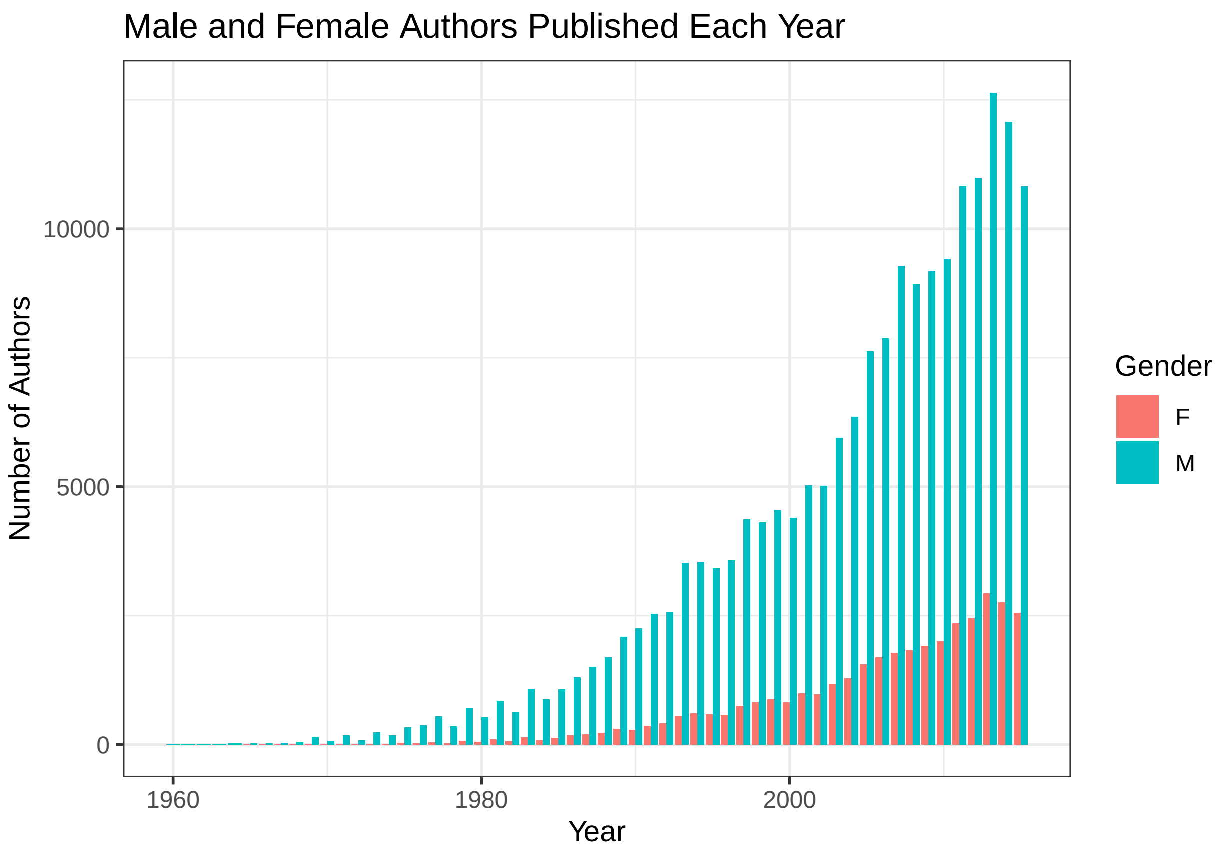
```
query <- "
SELECT year, gender, COUNT(DISTINCT name) AS count
FROM authors
JOIN general ON authors.k = general.k
WHERE gender IN ('M', 'F')
AND prob >= 0.9
GROUP BY year, gender
"

# execute the query and store the results in a data frame
results <- dbGetQuery(dbConnection, query)
as.tibble(results)
```

```
## # A tibble: 107 x 3
##     year gender count
##    <int> <chr>  <int>
##  1  1960 M          6
##  2  1961 M         18
##  3  1962 M         15
##  4  1963 M         13
##  5  1964 M         23
##  6  1965 F          2
##  7  1965 M         29
##  8  1966 F          2
##  9  1966 M         29
## 10  1967 F          4
## # ... with 97 more rows
```

```
# load the ggplot2 library for visualization
library(ggplot2)

ggplot(results, aes(x =year, y = count, fill = gender)) +
  geom_bar(position="dodge",stat = "identity") +
  labs(x = "Year", y = "Number of Authors", fill = "Gender") +
  ggtitle("Male and Female Authors Published Each Year") +
  theme_bw()
```



Observations:

1. The number of distinct male authors have always been significantly higher than the number of distinct female authors across the years.
2. The trend can be seen that the number of authors have significantly risen over the years
3. The number of female authors barely cross the 2500 count mark

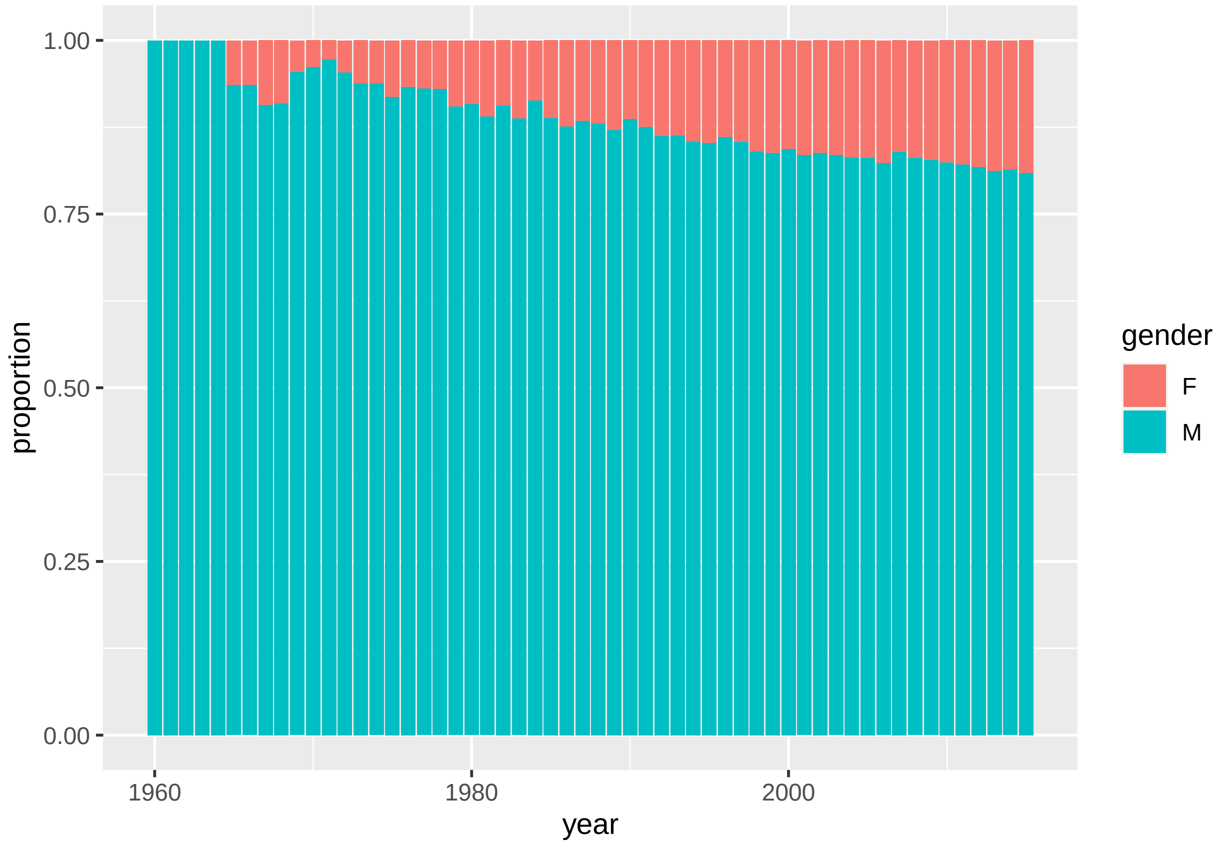
#Q5

Still including only the authors for whom a gender was predicted with a probability of 0.90 or greater, create a stacked bar plot showing the proportions of distinct male authors vs. distinct female authors published each year. (The stacked bars for each year will sum to one.) Comment on the visualization.

```
query <-
"SELECT year, gender,
COUNT(DISTINCT name) * 1.0 / SUM(COUNT(DISTINCT name))
OVER (PARTITION BY year) AS proportion
FROM authors
JOIN general ON authors.k = general.k
WHERE prob >= 0.9 AND gender IN ('M', 'F')
GROUP BY year, gender"

# Run the query and store the results in a data frame
results <- dbGetQuery(dbConnection, query)
```

```r
# Create the stacked bar plot
ggplot(results, aes(x=year, y=proportion, fill=gender)) +
  geom_bar(position = "stack", stat="identity")
```



Observations:

1. The proportion of male authors have always been greater than the number of female authors over the years
2. If you sum up the proprtions for each year, the addition mounts to 1.
3. For the initial few years, there are no female authors at all. Due to this, the proportion of male authors for initial years is 1.