"— title:"Assignment 2w" author: "Mounica Subramani" date: "January 25, 2018" output: pdf_document —

```
# Import required library files

library(ggplot2)
library(tidyverse)
```

```
## -- Attaching packages ----------------------------------------------------------- tidyverse 1.2.1 --
```

```
## v tibble  1.4.1     v purrr   0.2.4
## v tidyr   0.7.2     v dplyr   0.7.4
## v readr   1.1.1     v stringr 1.2.0
## v tibble  1.4.1     v forcats 0.2.0
```

```
## -- Conflicts -------------------------------------------------------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(dplyr)
library(rmarkdown)
library(nycflights13)
library(maps)
```

```
##
## Attaching package: 'maps'
```

```
## The following object is masked from 'package:purrr':
##
##     map
```

```
library(measurements)
library(RSQLite)
library(tidyr)
library(DBI)
library(RMySQL)
```

```
##
## Attaching package: 'RMySQL'
```

```
## The following object is masked from 'package:RSQLite':
##
##     isIdCurrent
```

```
options(width = 90)
```

Part A

Problem 1

Find a dataset that is personally interesting to you. It may be a publicly-available dataset, or a dataset for which you have permission to use and share results.

Import the dataset into R, put it into a tidy format, and print the first ten observations of the dataset.

```
# The data set is taken from Kaggle "https://www.kaggle.com/borapajo/food-choices"

# Read the data set into R.
```

```r
A_Data <- read_csv("C:/Users/mouni/Downloads/Sem 1/R Lang/Datasets/food-choices/food_coded.csv")
```

```
## Warning: Duplicated column names deduplicated: 'comfort_food_reasons_coded' =>
## 'comfort_food_reasons_coded_1' [12]
```

```
## Parsed with column specification:
## cols(
##   .default = col_integer(),
##   GPA = col_character(),
##   calories_day = col_double(),
##   calories_scone = col_double(),
##   comfort_food = col_character(),
##   comfort_food_reasons = col_character(),
##   cook = col_double(),
##   cuisine = col_double(),
##   diet_current = col_character(),
##   drink = col_double(),
##   eating_changes = col_character(),
##   employment = col_double(),
##   exercise = col_double(),
##   father_education = col_double(),
##   father_profession = col_character(),
##   fav_cuisine = col_character(),
##   fav_food = col_double(),
##   food_childhood = col_character(),
##   healthy_meal = col_character(),
##   ideal_diet = col_character(),
##   income = col_double()
##   # ... with 13 more columns
## )
```

```
## See spec(...) for full column specifications.
```

```r
# Put the data in a tidy format.

# Replace the numeric values for gender and other intended columns with
# appropriate values from the codebook of the data set.

A_Data$Gender[A_Data$Gender == 1] <- "Female"
A_Data$Gender[A_Data$Gender == 2] <- "Male"

A_Data$breakfast[A_Data$breakfast == 1] <- "Cereal"
A_Data$breakfast[A_Data$breakfast == 2] <- "Donut"

A_Data$calories_day[A_Data$calories_day == 1] <- "No Idea"
A_Data$calories_day[A_Data$calories_day == 2] <- "Not Imprtnt"
A_Data$calories_day[A_Data$calories_day == 3] <- "Moderately Imprtnt"
A_Data$calories_day[A_Data$calories_day == 4] <- "Very Imprtnt"

A_Data$coffee[A_Data$coffee == 1] <- "Creamy Frapuccino"
A_Data$coffee[A_Data$coffee == 2] <- "Expresso"

A_Data$comfort_food_reasons_coded[A_Data$comfort_food_reasons_coded == 1] <- "Stress"
A_Data$comfort_food_reasons_coded[A_Data$comfort_food_reasons_coded == 2] <- "Boredom"
A_Data$comfort_food_reasons_coded[A_Data$comfort_food_reasons_coded == 3] <- "depressed"
```

```r
A_Data$comfort_food_reasons_coded[A_Data$comfort_food_reasons_coded == 5] <- "Lazy"
A_Data$comfort_food_reasons_coded[A_Data$comfort_food_reasons_coded == 7] <- "Happy"
A_Data$comfort_food_reasons_coded[A_Data$comfort_food_reasons_coded == 4] <- "Hunger"
A_Data$comfort_food_reasons_coded[A_Data$comfort_food_reasons_coded == 6] <- "ColdWeather"
A_Data$comfort_food_reasons_coded[A_Data$comfort_food_reasons_coded == 8] <- "WatchTV"
A_Data$comfort_food_reasons_coded[A_Data$comfort_food_reasons_coded == 9] <- "None"


A_Data$cuisine[A_Data$cuisine == 1] <- "American"
A_Data$cuisine[A_Data$cuisine == 2] <- "Mexican Spanish"
A_Data$cuisine[A_Data$cuisine == 3] <- "Korean/Asian"
A_Data$cuisine[A_Data$cuisine == 4] <- "Indian"
A_Data$cuisine[A_Data$cuisine == 5] <- "American inspired international dishes"
A_Data$cuisine[A_Data$cuisine == 6] <- "Other"

A_Data$eating_changes_coded[A_Data$eating_changes_coded == 1] <- "Worse"
A_Data$eating_changes_coded[A_Data$eating_changes_coded == 2] <- "Better"
A_Data$eating_changes_coded[A_Data$eating_changes_coded == 3] <- "Same"
A_Data$eating_changes_coded[A_Data$eating_changes_coded == 4] <- "Unclear"

## Print the first ten observations of the dataset.

head(A_Data, n=10)
```

```
## # A tibble: 10 x 61
##    GPA   Gender breakfast calories_chicken calories_day      calories_scone coffee
##    <chr> <chr>  <chr>                <int> <chr>                     <dbl> <chr>
##  1 2.4   Male   Cereal                 430 NaN                         315 Creamy Frapu~
##  2 3.654 Female Cereal                 610 Moderately Imprt~           420 Expresso
##  3 3.3   Female Cereal                 720 Very Imprtnt                420 Expresso
##  4 3.2   Female Cereal                 430 Moderately Imprt~           420 Expresso
##  5 3.5   Female Cereal                 720 Not Imprtnt                 420 Expresso
##  6 2.25  Female Cereal                 610 Moderately Imprt~           980 Expresso
##  7 3.8   Male   Cereal                 610 Moderately Imprt~           420 Expresso
##  8 3.3   Female Cereal                 720 Moderately Imprt~           420 Creamy Frapu~
##  9 3.3   Female Cereal                 430 NaN                         420 Creamy Frapu~
## 10 3.3   Female Cereal                 430 Moderately Imprt~           315 Expresso
## # ... with 54 more variables: comfort_food <chr>, comfort_food_reasons <chr>,
## #   comfort_food_reasons_coded <chr>, cook <dbl>, comfort_food_reasons_coded_1 <int>,
## #   cuisine <chr>, diet_current <chr>, diet_current_coded <int>, drink <dbl>,
## #   eating_changes <chr>, eating_changes_coded <chr>, eating_changes_coded1 <int>,
## #   eating_out <int>, employment <dbl>, ethnic_food <int>, exercise <dbl>,
## #   father_education <dbl>, father_profession <chr>, fav_cuisine <chr>, fav_cuisine_coded
## #   <int>, fav_food <dbl>, food_childhood <chr>, fries <int>, fruit_day <int>,
## #   grade_level <int>, greek_food <int>, healthy_feeling <int>, healthy_meal <chr>,
## #   ideal_diet <chr>, ideal_diet_coded <int>, income <dbl>, indian_food <int>,
## #   italian_food <int>, life_rewarding <dbl>, marital_status <dbl>, meals_dinner_friend
## #   <chr>, mother_education <dbl>, mother_profession <chr>, nutritional_check <int>,
## #   on_off_campus <dbl>, parents_cook <int>, pay_meal_out <int>, persian_food <dbl>,
## #   self_perception_weight <dbl>, soup <dbl>, sports <dbl>, thai_food <int>,
## #   tortilla_calories <dbl>, turkey_calories <int>, type_sports <chr>, veggies_day <int>,
## #   vitamins <int>, waffle_calories <int>, weight <chr>
```

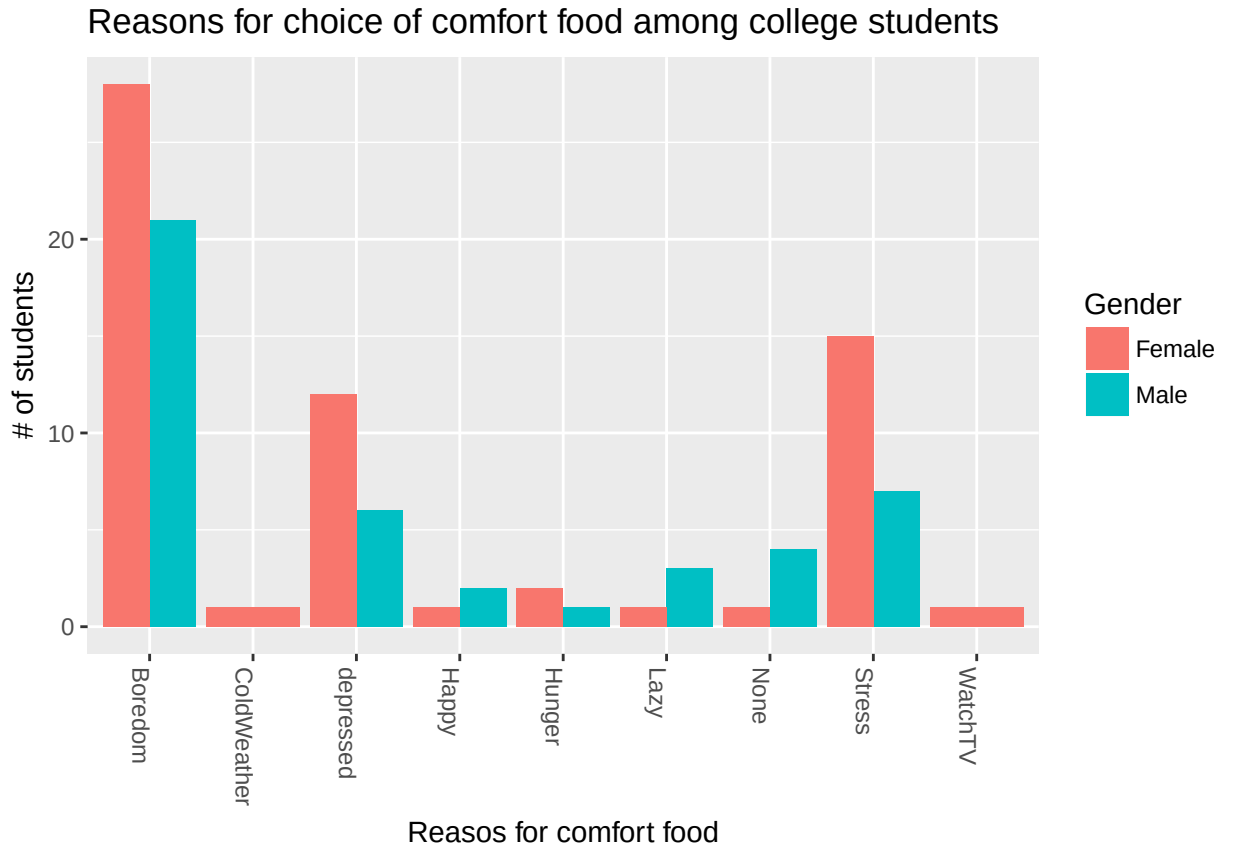
Problem 2

Step 1: Perform exploratory data analysis on the dataset, using the techniques learned in class. Calculate summary statistics that are of interest to you and create plots using ggplot2 that show your findings.

Step 2: Create an attractive PowerPoint or Keynote slide including your name, a description of your dataset, and your key findings, incorporating any plots and/or tables that are most relevant and interesting. Make sure you cite the source of the data!

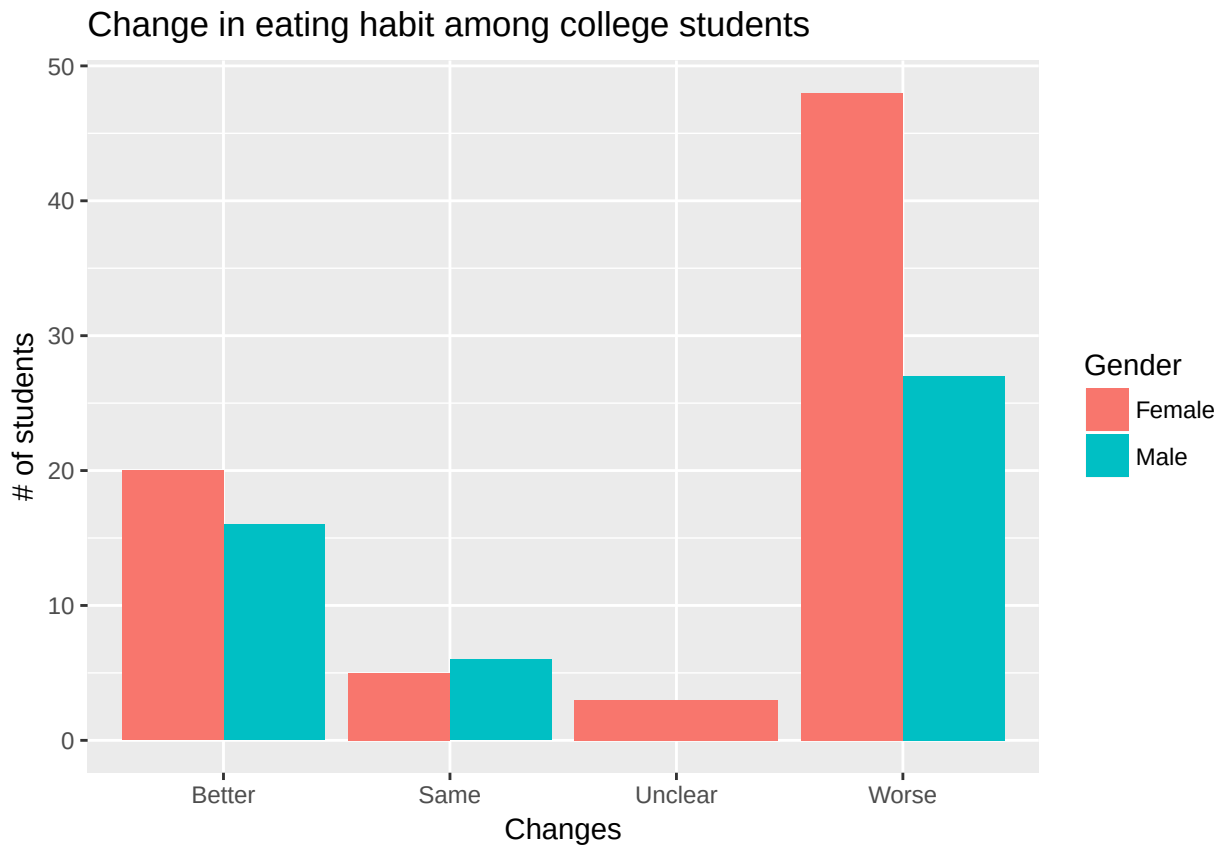
Step 3: Export this slide to PDF, and upload it to Piazza as a public Note titled "[mini-poster] your name" in the "miniposter" folder, along with a brief description of the dataset by the homework due date.

```
# show the reasons for comfort food and analyses which gender is suffering or
# experiencing the extreme mood swings and also, who is more depressed or sad.

A_Data_plot <- A_Data %>%
  group_by(comfort_food_reasons_coded, Gender) %>%
  summarise(count=n())

A_Data_plot<- A_Data_plot[complete.cases(A_Data_plot), ]

A_Data_plot %>%
  ggplot(aes(x=comfort_food_reasons_coded, y=count, fill=Gender)) +
  geom_col(position="dodge") +
  labs(title="Reasons for choice of comfort food among college students",
       x= "Reasos for comfort food",
       y= "# of students") +
  theme(axis.text.x = element_text(angle = -90, hjust =0))
```



Female students are finding depressed and stressed feelings for their food reasons rather than male students.

```
# Eating habit changes were observed as students entered colleges.
# Analyze the changes in eating as per the gender.

A_Data_plot2 <- A_Data %>%
  group_by(eating_changes_coded, Gender) %>%
  summarise(count=n())

A_Data_plot2 %>%
  ggplot(aes(x=eating_changes_coded, y=count, fill=Gender)) +
  geom_col(position="dodge") +
  labs(title="Change in eating habit among college students",
       x= "Changes",
       y= "# of students")
```



It is clear that, Female students have their worst changes in their eating habit than male students.

```
# summarize total number of male and female students

A_Data_Summ <- A_Data %>%
group_by(Gender) %>%
summarise(count=n())

A_Data_Summ
```

```
## # A tibble: 2 x 2
##   Gender count
```

```
##   <chr>  <int>
## 1 Female    76
## 2 Male      49
```

```
# Categorize the type of coffee the male and female students prefer

A_Data_Summ <- A_Data %>%
group_by(Gender,coffee) %>%
summarise(count=n())

A_Data_Summ
```

```
## # A tibble: 4 x 3
## # Groups: Gender [?]
##   Gender coffee            count
##   <chr>  <chr>             <int>
## 1 Female Creamy Frapuccino    18
## 2 Female Expresso             58
## 3 Male   Creamy Frapuccino    13
## 4 Male   Expresso             36
```

PART B

Problem 3

Create a bar plot showing the total number of enrolled students of each race

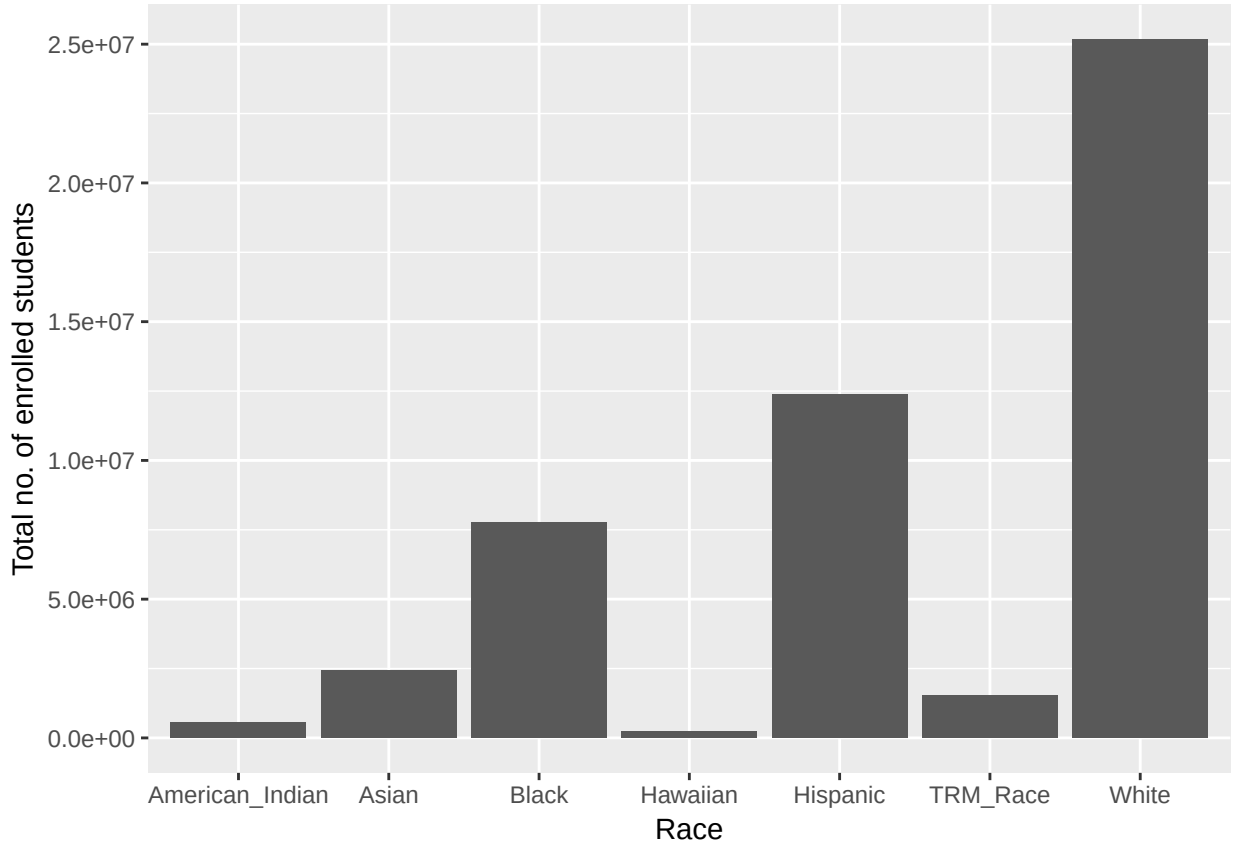
```
## Read the CSV file "CRDC2013_14_SCH" into "C_Data" data frame.

C_Data <- read_csv("C:/users/mouni/Downloads/crdc201314/CRDC2013_14_SCH.csv",
                   na=c("-9", "-5", "-2"))
```

```
## Parsed with column specification:
## cols(
##   .default = col_integer(),
##   LEA_STATE = col_character(),
##   LEA_NAME = col_character(),
##   SCH_NAME = col_character(),
##   COMBOKEY = col_character(),
##   LEAID = col_character(),
##   SCHID = col_character(),
##   JJ = col_character(),
##   CCD_LATCOD = col_double(),
##   CCD_LONCOD = col_double(),
##   NCES_SCHOOL_ID = col_character(),
##   MATCH_FLAG = col_character(),
##   SCH_GRADE_PS = col_character(),
##   SCH_GRADE_KG = col_character(),
##   SCH_GRADE_G01 = col_character(),
##   SCH_GRADE_G02 = col_character(),
##   SCH_GRADE_G03 = col_character(),
##   SCH_GRADE_G04 = col_character(),
##   SCH_GRADE_G05 = col_character(),
```

```
##   SCH_GRADE_G06 = col_character(),
##   SCH_GRADE_G07 = col_character()
##   # ... with 75 more columns
## )
```

```
## See spec(...) for full column specifications.
```

```
# calculating total number of enrolled students of each race
# -and store it in a seperate data frame.

CRDC_ENR_Race <- transmute(C_Data,
                        ENR_HI = SCH_ENR_HI_M + SCH_ENR_HI_F,
                        ENR_AM = SCH_ENR_AM_M + SCH_ENR_AM_F,
                        ENR_AS = SCH_ENR_AS_M + SCH_ENR_AS_F,
                        ENR_HP = SCH_ENR_HP_M + SCH_ENR_HP_F,
                        ENR_BL = SCH_ENR_BL_M + SCH_ENR_BL_F,
                        ENR_WH = SCH_ENR_WH_M + SCH_ENR_WH_F,
                        ENR_TR = SCH_ENR_TR_M + SCH_ENR_TR_F,
                        )

# Calculating sum of enrolled students in each race

df1 <- CRDC_ENR_Race %>%
  summarise(Hispanic=sum(ENR_HI, na.rm=TRUE),
            American_Indian=sum(ENR_AM, na.rm=TRUE),
            Asian=sum(ENR_AS, na.rm=TRUE),
            Hawaiian=sum(ENR_HP, na.rm=TRUE),
            Black=sum(ENR_BL, na.rm=TRUE),
            White=sum(ENR_WH, na.rm=TRUE),
            TRM_Race=sum(ENR_TR, na.rm=TRUE))

# gathering the data under a single column called race and values under total

df <- gather(df1,
             `Hispanic`, `American_Indian`, `Asian`,
             `Hawaiian`, `Black`, `White`, `TRM_Race`,
             key = "Race",
             value = "Tot")

# plot showing the total number of enrolled students of each race

ggplot(df,
       mapping = aes(x=Race,
       y=Tot)) + geom_col() + labs(y="Total no. of enrolled students")
```



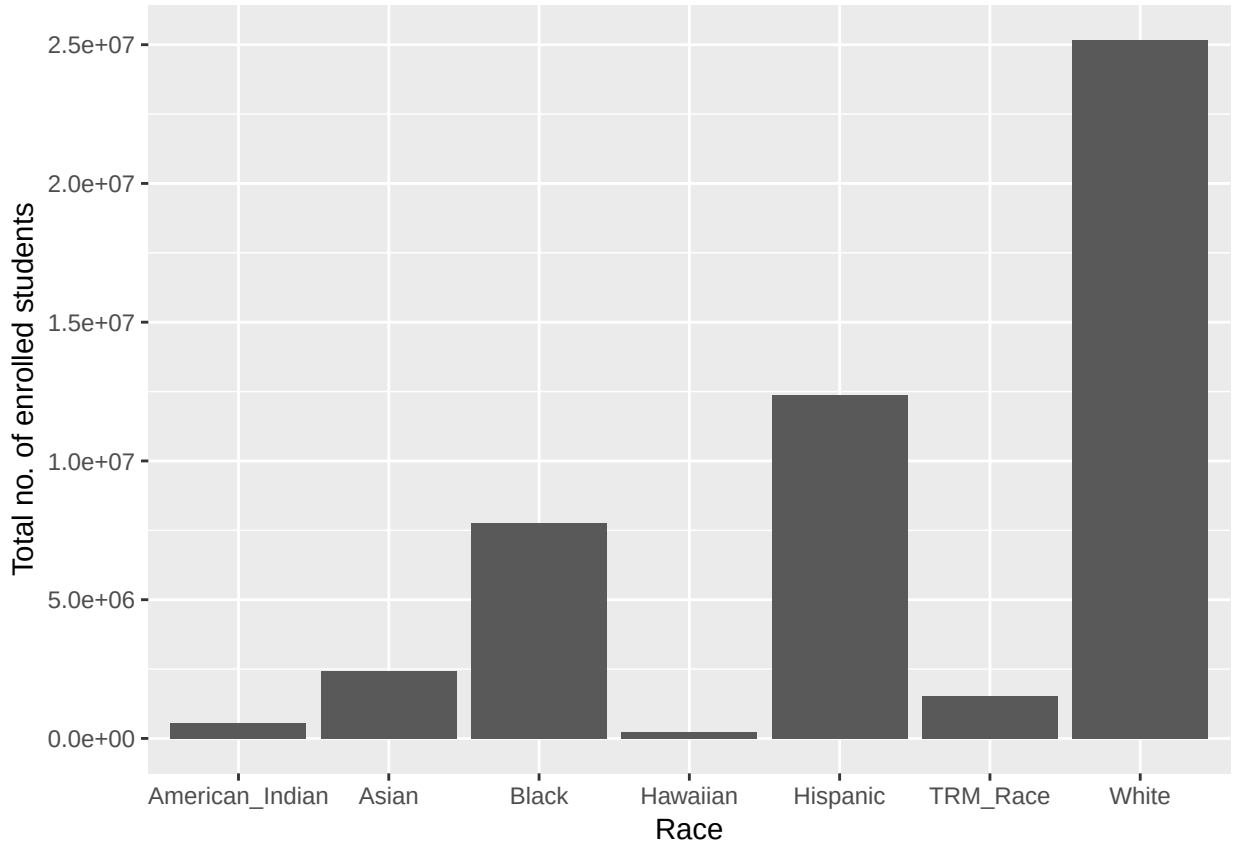
Problem 4

Create a bar plot showing the number of students of each race enrolled in an Advanced Mathematics class.

Comment on any similarities or differences between this distribution and the one you plotted in Problem 3.

```
CRDC_ENR_ADVM <- transmute(C_Data,
                        ADVM_ENR_HI = SCH_MATHENR_ADVM_HI_M + SCH_MATHENR_ADVM_HI_F,
                        ADVM_ENR_AM = SCH_MATHENR_ADVM_AM_M + SCH_MATHENR_ADVM_AM_F,
                        ADVM_ENR_AS = SCH_MATHENR_ADVM_AS_M + SCH_MATHENR_ADVM_AS_F,
                        ADVM_ENR_HP = SCH_MATHENR_ADVM_HP_M + SCH_MATHENR_ADVM_HP_F,
                        ADVM_ENR_BL = SCH_MATHENR_ADVM_BL_M + SCH_MATHENR_ADVM_BL_F,
                        ADVM_ENR_WH = SCH_MATHENR_ADVM_WH_M + SCH_MATHENR_ADVM_WH_F,
                        ADVM_ENR_TR = SCH_MATHENR_ADVM_TR_M + SCH_MATHENR_ADVM_TR_F,
                        )

# Calculating sum of enrolled students of each race in Advanced mathematics class.

ADVM <- CRDC_ENR_ADVM %>%
  summarise(Hispanic=sum(ADVM_ENR_HI, na.rm=TRUE),
            American_Indian=sum(ADVM_ENR_AM, na.rm=TRUE),
            Asian=sum(ADVM_ENR_AS, na.rm=TRUE),
            Hawaiian=sum(ADVM_ENR_HP, na.rm=TRUE),
            Black=sum(ADVM_ENR_BL, na.rm=TRUE),
            White=sum(ADVM_ENR_WH, na.rm=TRUE),
            TRM_Race=sum(ADVM_ENR_TR, na.rm=TRUE))
```

```r
# gathering the data under a single column called race and values under total

ADVM_ENR <- gather(ADVM,
                   `Hispanic`, `American_Indian`, `Asian`,
                   `Hawaiian`, `Black`, `White`, `TRM_Race`,
                   key = "Race",
                   value = "Tot")

# plot showing the total number of enrolled students of each race

ggplot(df,
       mapping = aes(x=Race,
       y=Tot)) + geom_col() + labs(y="Total no. of enrolled students")
```




```r
# There is one similarity between both the plots, where number of students enrolled
#for each race is proportional to # of students enrolled for each race in advanced
# mathematics. The graph is pretty similar with difference in enrolled numbers.
```

PART C

Problem 5

Filter the data to include only the authors for whom a gender was predicted with a probability of 0.99 or

greater, and then create a bar plot showing the number of distinct male and female authors in the dataset.

```
rm(list = "authors")
```

```
## Warning in rm(list = "authors"): object 'authors' not found
```

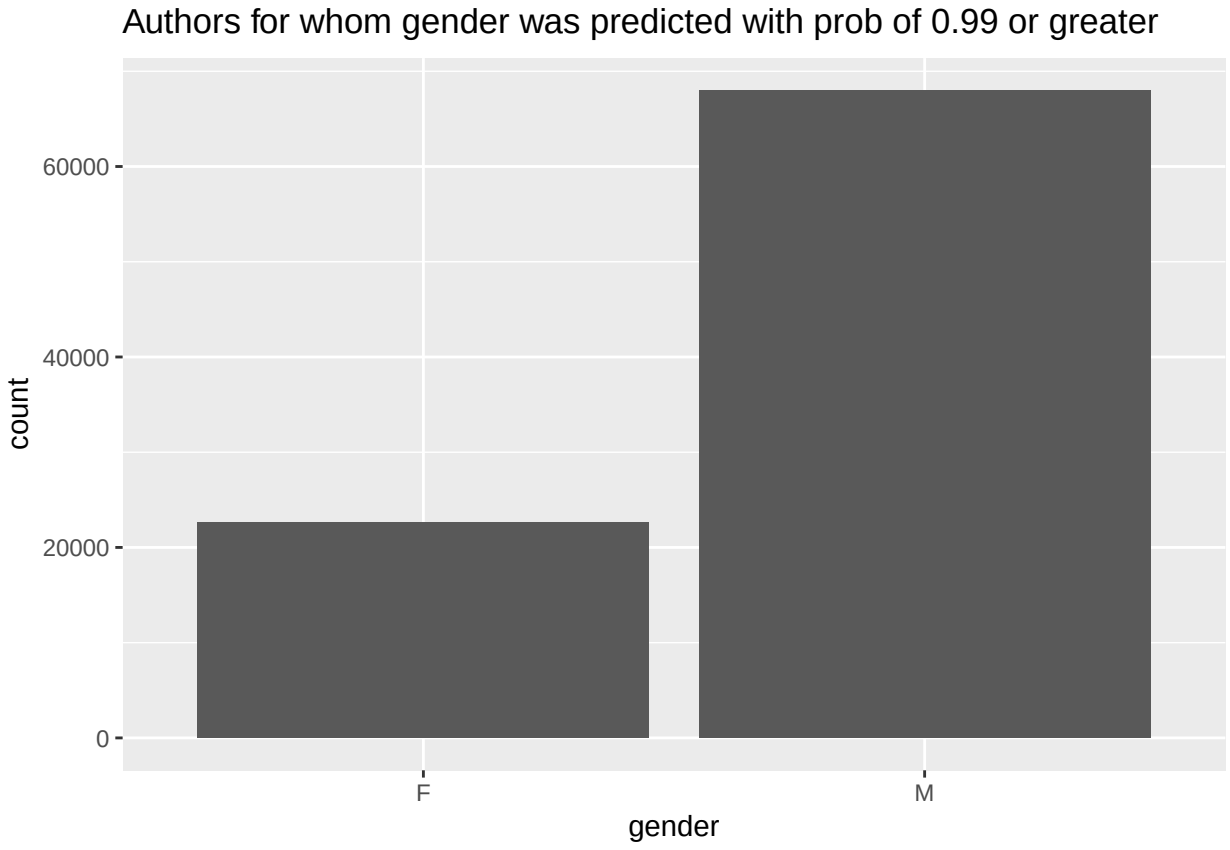
```
driver <- dbDriver("MySQL")
con <- dbConnect(MySQL(), dbname="dbpl", user="root",password="Chinchilla@1")

Gen <- tbl(con, "general")
Gen
```

```
## # Source: table<general> [?? x 10]
## # Database: mysql 5.7.21-log [root@localhost:/dbpl]
##    k                      year conf  crossref     cs    de    se    th publisher link
##    <chr>                 <int> <chr> <chr>     <int> <int> <int> <int> <chr>     <chr>
##  1 conf/aaai/0001M13      2013 AAAI  conf/aaa~     1     0     0     0 AAAI      http://~
##  2 conf/aaai/0001T15      2015 AAAI  conf/aaa~     1     0     0     0 AAAI      http://~
##  3 conf/aaai/0001TZLL14   2014 AAAI  conf/aaa~     1     0     0     0 AAAI      http://~
##  4 conf/aaai/0001VD15     2015 AAAI  conf/aaa~     1     0     0     0 AAAI      http://~
##  5 conf/aaai/0001YT15     2015 AAAI  conf/aaa~     1     0     0     0 AAAI      http://~
##  6 conf/aaai/0002GYSZL14  2014 AAAI  conf/aaa~     1     0     0     0 AAAI      http://~
##  7 conf/aaai/0002Z15      2015 AAAI  conf/aaa~     1     0     0     0 AAAI      http://~
##  8 conf/aaai/0002ZL15     2015 AAAI  conf/aaa~     1     0     0     0 AAAI      http://~
##  9 conf/aaai/0003MGF14    2014 AAAI  conf/aaa~     1     0     0     0 AAAI      http://~
## 10 conf/aaai/0005YJZ15    2015 AAAI  conf/aaa~     1     0     0     0 AAAI      http://~
## # ... with more rows
```

```
Author <- tbl(con, "authors")
Author
```

```
## # Source: table<authors> [?? x 6]
## # Database: mysql 5.7.21-log [root@localhost:/dbpl]
##       id k                      pos name                gender  prob
##    <dbl> <chr>                <int> <chr>               <chr>  <dbl>
##  1  1.00 conf/aaai/0001M13        0 Chang Wang 0001     M      0.630
##  2  2.00 conf/aaai/0001M13        1 Sridhar Mahadevan   M      1.00
##  3  3.00 conf/aaai/0001T15        0 Claudia Schulz 0001 F      1.00
##  4  4.00 conf/aaai/0001T15        1 Francesca Toni      F      1.00
##  5  5.00 conf/aaai/0001TZLL14     0 Jing Zhang 0001     F      0.720
##  6  6.00 conf/aaai/0001TZLL14     1 Jie Tang            F      0.690
##  7  7.00 conf/aaai/0001TZLL14     2 Honglei Zhuang      -      2.00
##  8  8.00 conf/aaai/0001TZLL14     3 Cane Wing-ki Leung  F      0.750
##  9  9.00 conf/aaai/0001TZLL14     4 Juan-Zi Li          -      2.00
## 10 10.0  conf/aaai/0001VD15       0 Bart Bogaerts 0001  M      0.990
## # ... with more rows
```

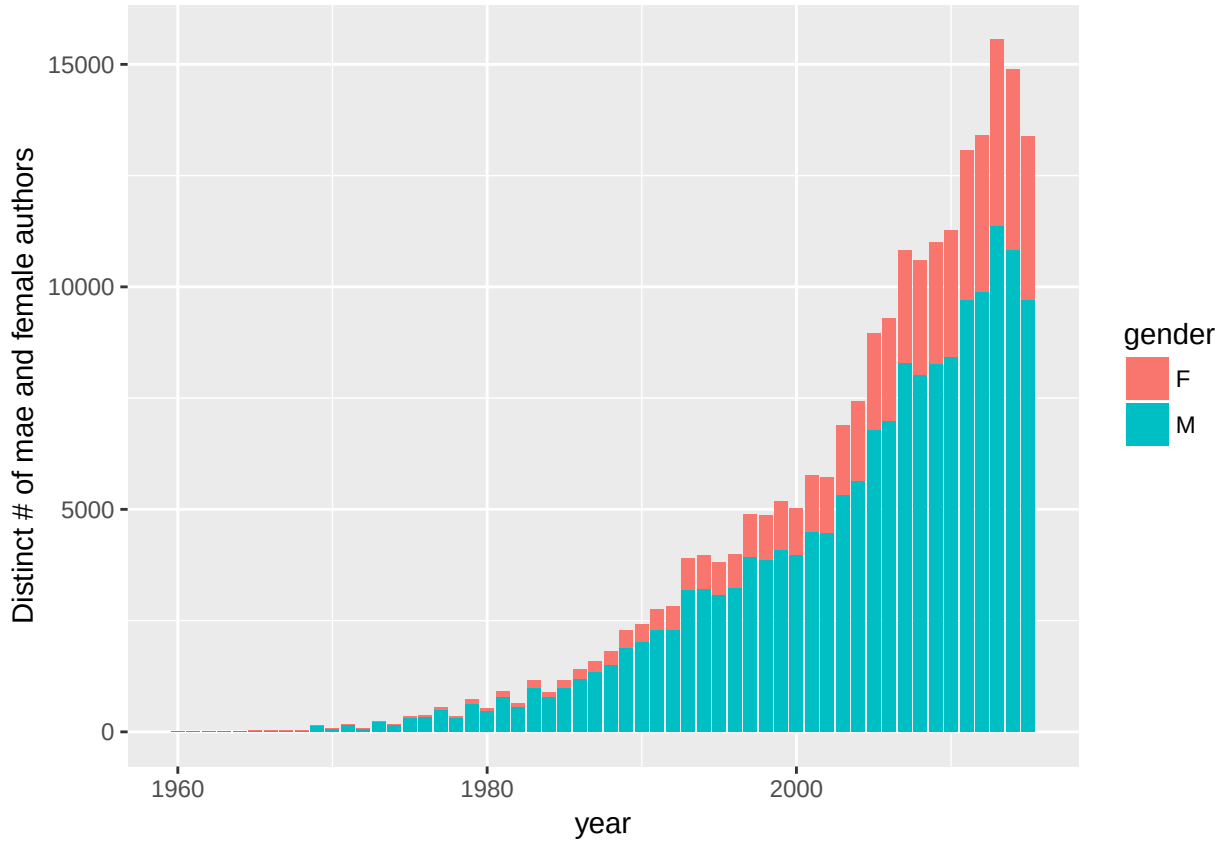
```
Author %>%
  filter(prob>=0.99 && gender=="M" | gender=="F") %>%
  group_by(gender) %>%
  summarise(count=n_distinct(name)) %>%
  collect() %>%
  ggplot(aes(x=gender, y=count)) + geom_col() +
  labs(title = "Authors for whom gender was predicted with prob of 0.99 or greater")
```



Problem 6

Again including only the authors for whom a gender was predicted with a probability of 0.99 or greater, create a stacked bar plot showing the number of distinct male and female authors published each year.

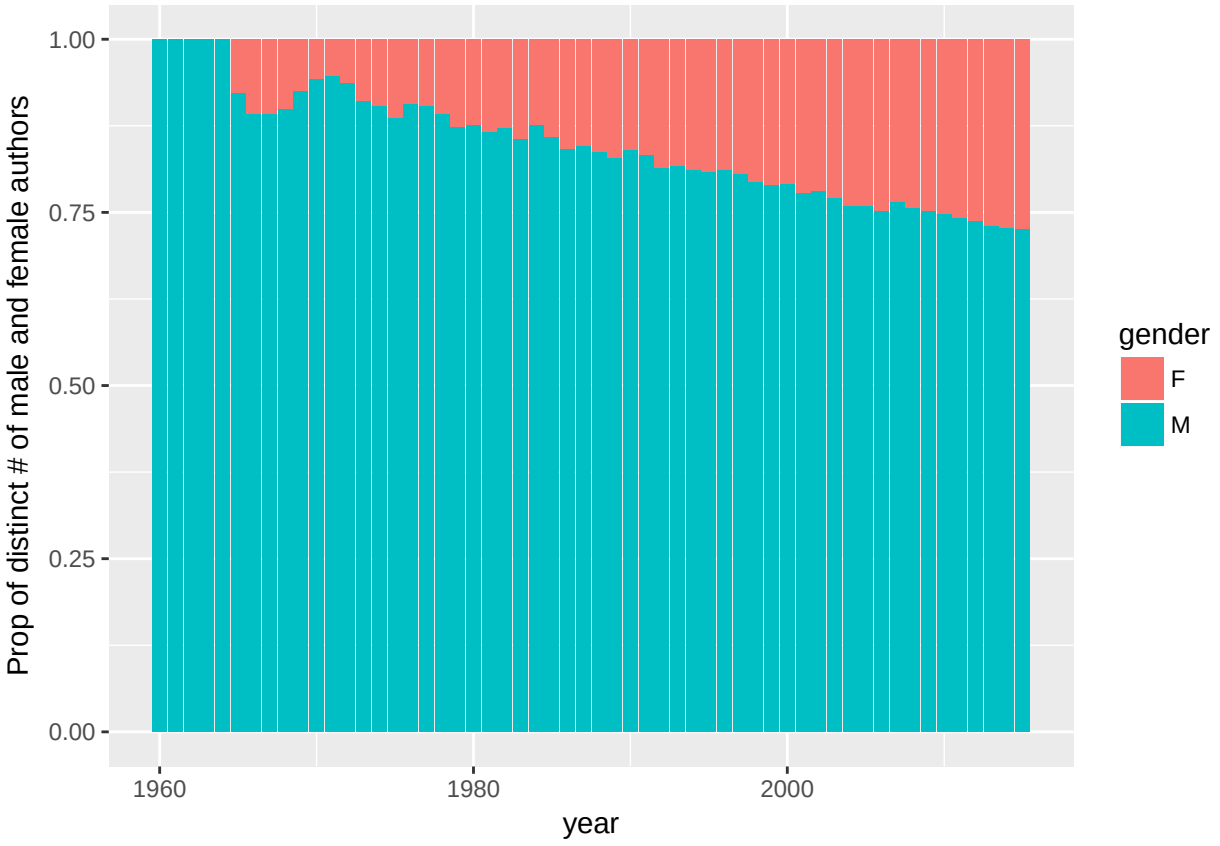
```
inner_join(Author, Gen, by="k") %>%
  filter(prob>=0.99 && gender=="M" | gender=="F") %>%
  group_by(year, gender) %>%
  summarise(total=n_distinct(name)) %>%
  collect() %>%
  ggplot(aes(x=year, y=total, fill=gender)) +
  geom_bar(stat = "identity") +
  labs(y="Distinct # of mae and female authors")
```



Problem 7

Still including only the authors for whom a gender was predicted with a probability of 0.99 or greater, create a stacked bar plot showing the proportions of distinct male and female authors published each year. (The stacked bars for each year will sum to one.)

```
inner_join(Author, Gen, by="k") %>%
  filter(prob>=0.99 && gender=="M" | gender=="F") %>%
  group_by(year, gender) %>%
  summarise(Total=n_distinct(name)) %>%
  collect() %>%
  ggplot(aes(x=year, y=Total, fill=gender)) +
  geom_bar(stat = "identity", position = "fill") +
  labs(y="Prop of distinct # of male and female authors")
```



PART D

Problem 8

We would like to investigate how certain questions break down among trans women, trans men, and non-binary participants. However, the survey is sometimes outdated in its terminology and also includes many questioning participants who have not transitioned yet.

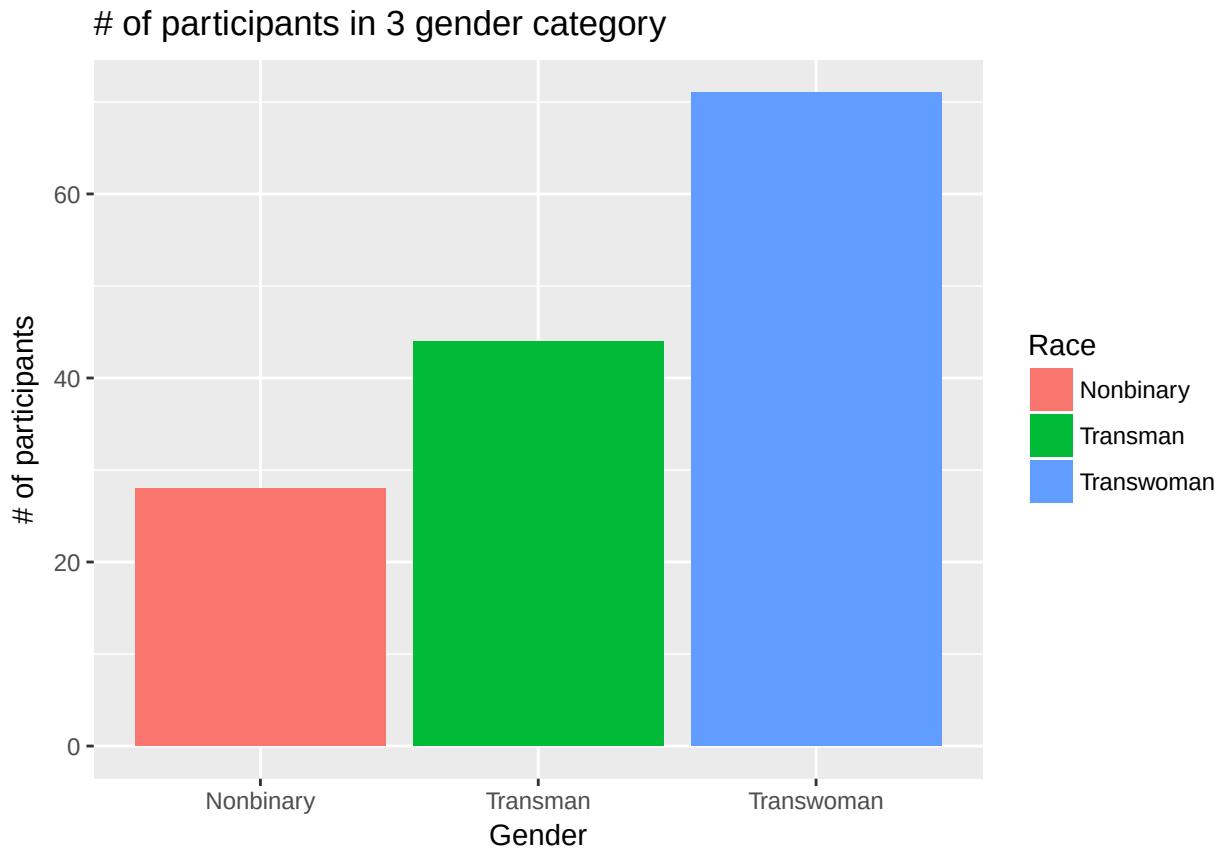
Transform the data to include 3 gender categories for men, women, and non-binary participants. Use the following definitions when transformaing the dataset: (1) trans women are women who were assigned male-at-birth; (2) trans men are men who were assigned-female-at-birth; (3) combine the "Genderqueer" and "Androgynous" categories to create a single "Non-binary" category. Filter the dataset to include only participants in these categories.

Create a bar plot showing the number of participants of each of the above genders.

Then create bar plots showing the proportion of participants who have been fired or denied a job due to their transgender status and/or gender expression. The plots should be faceted by gender and show separate proportions for trans women, trans men, and non-binary participants. (Do not include missing data in the plot.)

```
# Load the data into the data set.

load("C:/Users/mouni/Downloads/Sem 1/R Lang/ICPSR_31721/DS0001/31721-0001-Data.rda")

D_Data <- da31721.0001
```

```r
# Creating two more colums to acoomodate simplied values of Q5 and Q6.

samp <- mutate(D_Data, present=ifelse(Q6 == "Man", "Man", Q6),
               by_birth= ifelse(Q5 == "Man", "Man", Q5))

# samp1 dataset holding necessary columns

samp1 <- transmute(samp,
                   RKey = RESPKEY,
                   Birth = by_birth,
                   Present = present)

# classifying the data into transman and transwoman

samp1 <- mutate(samp1,
                Trans_woman = ifelse(Birth == "1" & Present == "2", "Trans_woman", NA),
                Trans_man = ifelse(Birth == "2" & Present == "1", "Trans_man", NA),
                Non_binary = ifelse(Present == "4" | Present == "6", "Non_binary", NA))

# Combineing 3 column values into one

samp1 <- unite(samp1, Race, c(Trans_woman, Trans_man, Non_binary), remove=FALSE)

samp1$Race = gsub("NA", "", samp1$Race)
samp1$Race = gsub("_", "", samp1$Race)

samp2 <- samp1 %>%
  group_by(Race) %>%
  count()

samp2 <- samp2[-c(1),]

ggplot(samp2, aes(x=Race, y=n, fill = Race)) +
  geom_col() + labs(title ="# of participants in 3 gender category",
                    x="Gender", y="# of participants")
```

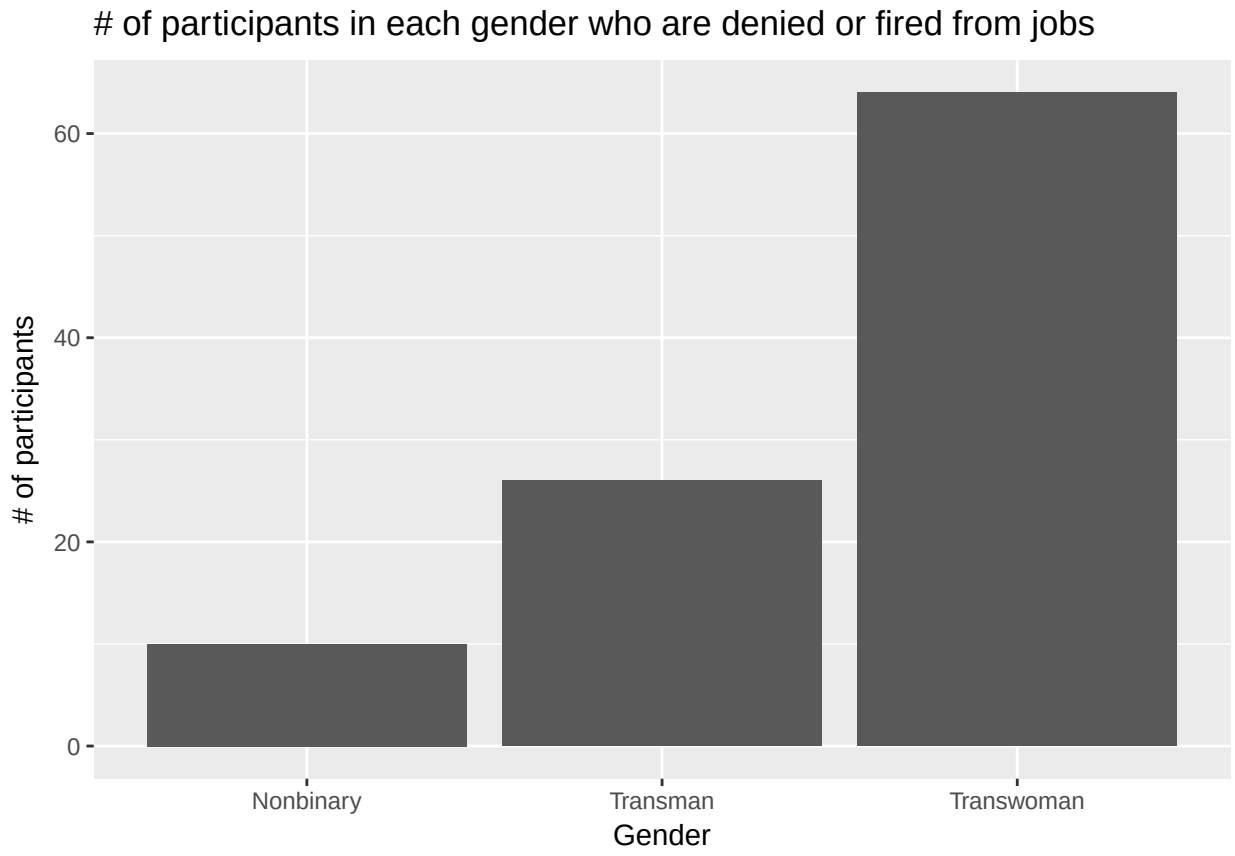



```
# Data set includes participant response who have been fired or denied a job.

JobStatus <- transmute(samp,
                       RKey = RESPKEY,
                       Ques84 = Q84,
                       Ques86 = Q86)

# Joining samp1 data which already has transgender information in it

samp1 <- inner_join(samp1, JobStatus, by="RKey")

Jobs <- mutate(samp1,
                       Deny=ifelse(Ques84 == "Yes", "Yes", Ques84),
                       Fire=ifelse(Ques86 == "Yes", "Yes", Ques86))

Jobs <- transmute(Jobs,
                  Rkey = RKey,
                  Race = Race,
                  Denied = ifelse(Deny == "1", "Denied", NA),
                  Fired = ifelse(Fire == "1", "Fired", NA))

# Job1 has data of participants who have been denied a job

Job1 <- Jobs %>%
  group_by(Race, Denied) %>%
  count()
```

```r
# Job1 has data of participants who have been fired from a job

Job2 <- Jobs %>%
  group_by(Race, Fired) %>%
  count()

Job1_2 <- inner_join(Job1, Job2, by="Race")

Job1_2 <- Job1_2[-c(1:4),]

# Calculating the total of those who have been both fired and denied a job

Job1_2 <- mutate(Job1_2,
                 Both_Tot = n.x + n.y)

Job1_2 <- mutate(Job1_2,
                 coun = ifelse(Denied == "Denied" & Fired == "Fired", Both_Tot, NA),
                 coun1 = ifelse(Denied == "Denied" & is.na(Fired), n.x, NA),
                 coun2 = ifelse(is.na(Denied) & Fired == "Fired", n.y, NA))

Job1_2 <- unite(Job1_2, val, c(coun, coun1, coun2), remove=FALSE)

Job1_2$val = gsub("NA", "", Job1_2$val)
Job1_2$val = gsub("_", "", Job1_2$val)

Both <- transmute(Job1_2,
                  Race1 = Race,
                  Total = val)
```

```
## Adding missing grouping variables: `Race`, `Denied`
```

```r
Both$Race1 <- NULL
Both$Denied <- NULL

Both <- Both[-c(4,8,12),]

Both <- Both %>%
  group_by(Race) %>%
  summarise(tot = sum(as.numeric(Total)))

# Creating bar plots showing the proportion of participants who have been
# fired or denied a job due to their transgender status and/or gender expression.

# This plots is faceted by gender and shows separate proportions for trans women,
# trans men, and non-binary participants.

Both %>%
  group_by(Race) %>%
  ggplot(aes(x=Race, y=tot)) +
  geom_col() +
labs(title='# of participants in each gender who are denied or fired from jobs',
     x='Gender', y='# of participants')
```



Problem 9

Using the full dataset again, transform the dataset to have a column for race indicating the race of the participant. Include only the racial demographics with publicly available data (i.e., African American, Caucasian, Hispanic/Latinx, and Native American).

(Participants with two or more races may appear on multiple rows. Do not use the pre-calculated 'RACE' column in the dataset, which does not properly disambiguate multiracial participants.)

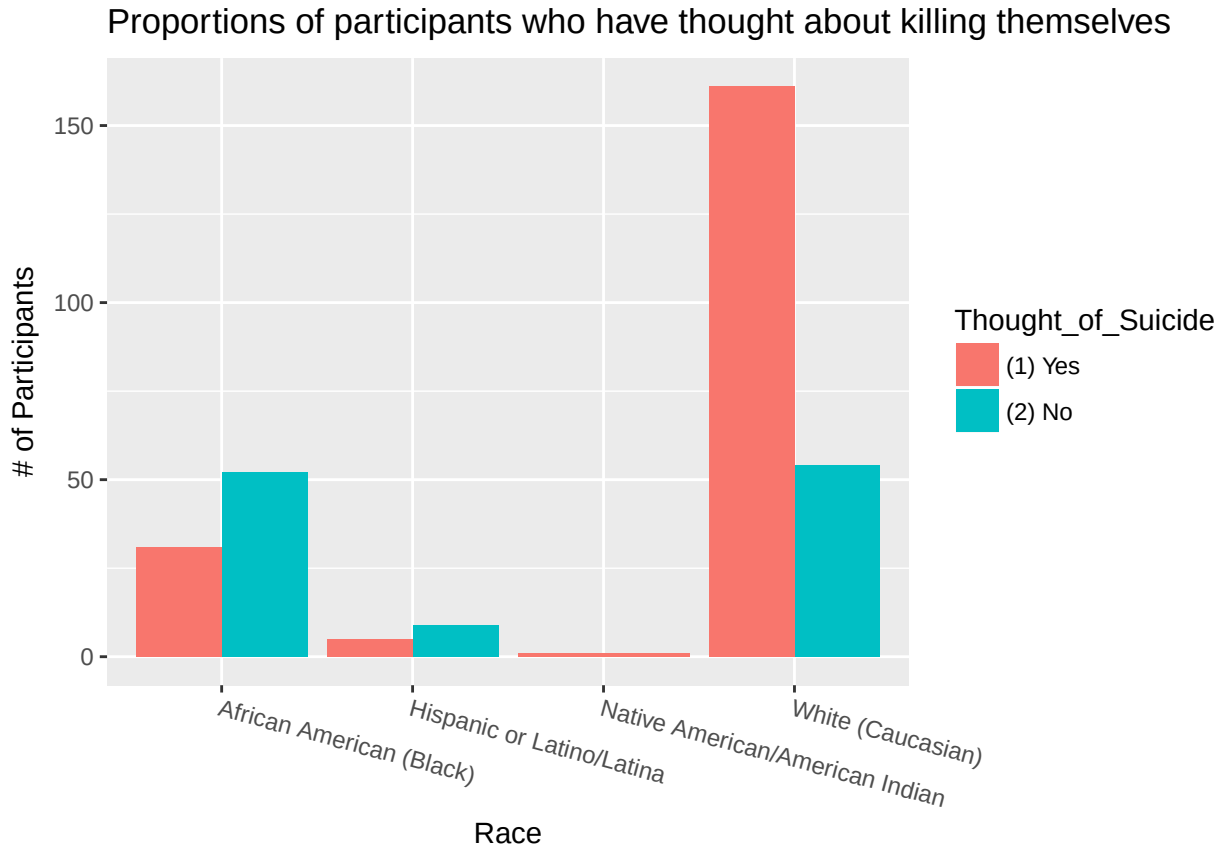
Then create bar plots showing the proportions of participants who have thought about killing themselves for African American, Caucasian, Hispanic/Latinx, and Native American demographics. (Do not include missing data in the plot.)

One of the findings reported in the National Transgender Discrimination Survey (http://www.thetaskforce.org/injustice-every-turn-report-national-transgender-discrimination-survey/) was that a staggering 41% of the respondents reported attempting suicide, compared to 1.6% in the general population.

Calculate the total proportion of participants who have attempted suicide in the Virginia THIS survey. (Include all participants.) Is it higher or lower than the national average for trans people?

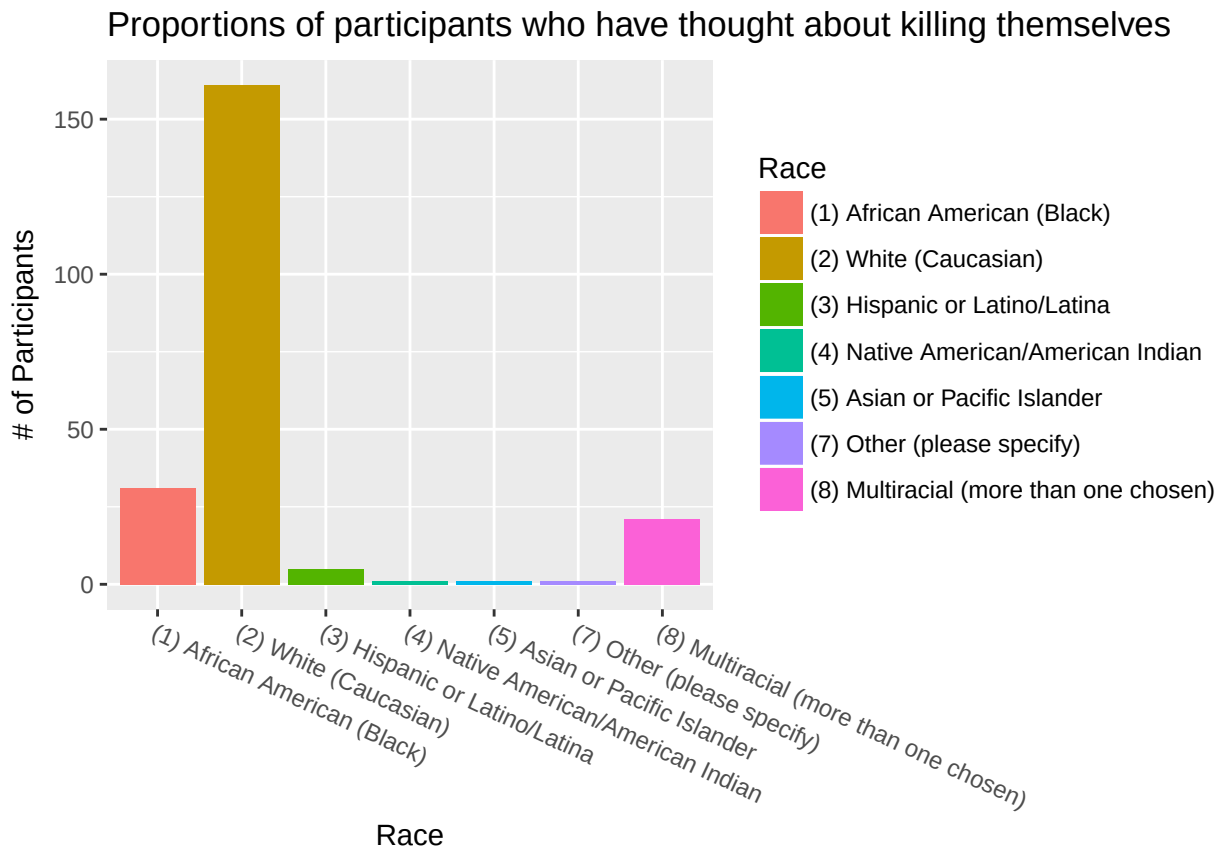
```
kill <- transmute(samp,
                  RKey = RESPKEY,
                  Thought_of_Suicide= Q131,
                  Race = RACE)


kill <- mutate(kill,
               Race1 = ifelse(Race == "(1) African American (Black)" |
                                Race == "(2) White (Caucasian)" |
```

```
                                Race == "(3) Hispanic or Latino/Latina" |
                                Race == "(4) Native American/American Indian",
                              kill$Race, NA))

kill <- mutate(kill,
               African_American = ifelse(Race1 == "1",
                                         "African American (Black)", NA),
               Caucasian = ifelse(Race1 == "2",
                                  "White (Caucasian)", NA),
               Hispanic = ifelse(Race1 == "3",
                                 "Hispanic or Latino/Latina", NA),
               Native_American = ifelse(Race1 == "4",
                                        "Native American/American Indian", NA))

kill$Race <- NULL

kill <- unite(kill, Race2, c(African_American,
                             Caucasian, Hispanic, Native_American),
              remove=FALSE)

kill$Race = gsub("NA", "", kill$Race2)
kill$Race = gsub("_", "", kill$Race)

kill$Race2 <- NULL
kill$Race1 <- NULL
kill$African_American <- NULL
kill$Caucasian <- NULL
kill$Hispanic <- NULL
kill$Native_American <- NULL

kill1 <- kill %>%
  group_by(Race,Thought_of_Suicide) %>%
  summarise(count = n())

Kill1 <- mutate(kill1,
               Thought_of_Suicide1 = ifelse(Thought_of_Suicide == "1", "Yes", "No"))

kill1 <- kill1[-c(1:3),]

kill1 <- kill1[complete.cases(kill1), ]

Kill2 <- kill1 %>%
group_by(Race, Thought_of_Suicide) %>%
summarise(prop_Suicide=mean(count, na.rm=TRUE))

Kill2 %>%
  group_by(Race) %>%
  ggplot(aes(x=Race, y=prop_Suicide, fill = Thought_of_Suicide)) +
  geom_col(position = "dodge") +
  labs(title="Proportions of participants who have thought about killing themselves",
       x= "Race", y= "# of Participants") +
  theme(axis.text.x = element_text(angle = -15, hjust =0))
```




```
# Calculate the total proportion of participants who have attempted suicide in
# the Virginia THIS survey. (Include all participants.)

Suicide <- samp1

Suicide <- transmute(samp,
                  RKey = RESPKEY,
                  Ques131= Q131,
                  Race = RACE)

kill <- mutate(kill,
               Race1 = ifelse(Race == "(1) African American (Black)" |
                                Race == "(2) White (Caucasian)" |
                                Race == "(3) Hispanic or Latino/Latina" |
                                Race == "(4) Native American/American Indian",
                              kill$Race, NA))

kill <- mutate(kill,
                African_American = ifelse(Race1 == "1",
                                          "African American (Black)", NA),
                Caucasian = ifelse(Race1 == "2",
                                   "White (Caucasian)", NA),
                Hispanic = ifelse(Race1 == "3",
                                  "Hispanic or Latino/Latina", NA),
                Native_American = ifelse(Race1 == "4",
                                         "Native American/American Indian", NA))
```

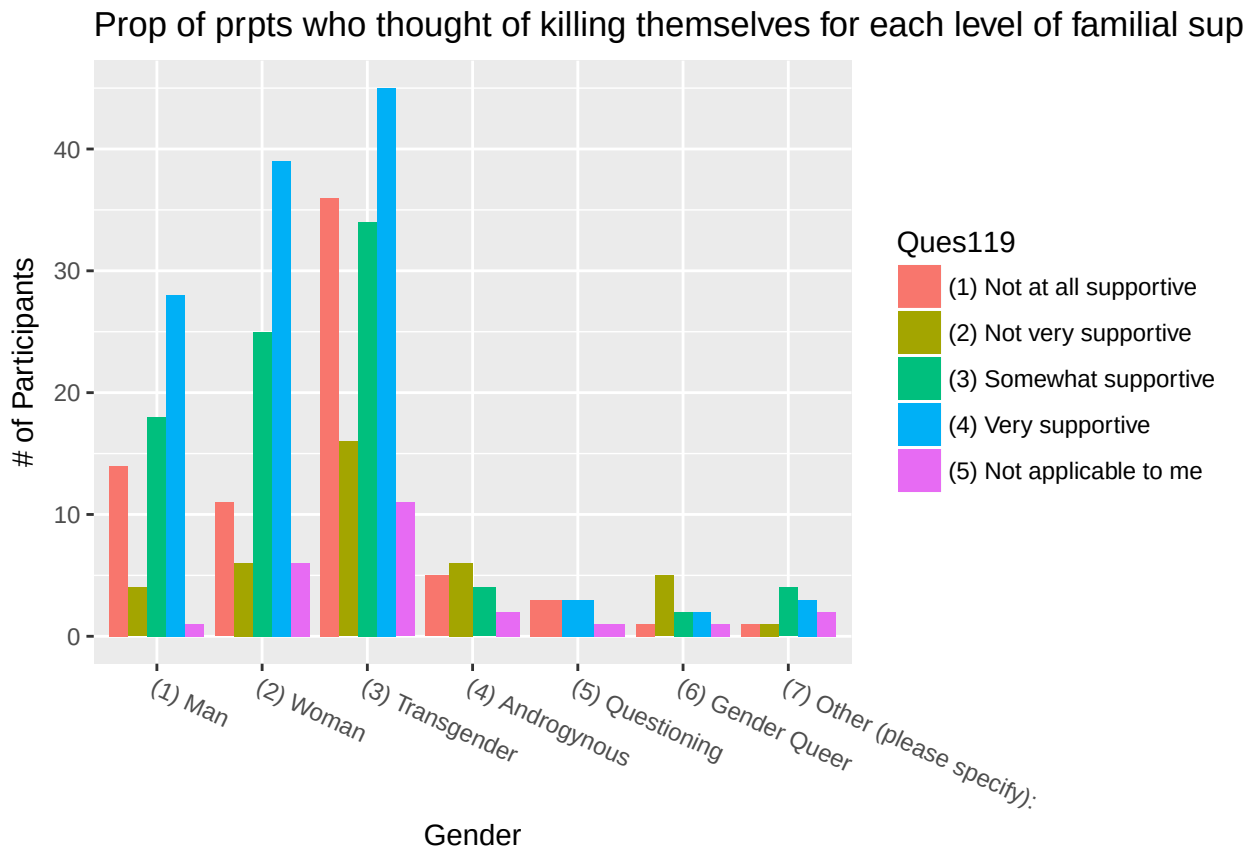
```r
Suicide <- mutate(Suicide, Thought_of_Suicide=ifelse(Ques131 == "Yes", "Yes", Ques131),
                  Thought_of_Suicide1 = ifelse(Thought_of_Suicide == "1", "Yes", "No"))

Suicide <- Suicide[complete.cases(Suicide), ]

Suicide <- Suicide %>%
group_by(Race, Thought_of_Suicide1) %>%
summarise(count = n())

Suicide %>%
  filter(Thought_of_Suicide1 == "Yes") %>%
  ggplot(aes(x=Race, y=count, fill=Race)) +
  geom_col() +
  labs(title="Proportions of participants who have thought about killing themselves",
       x= "Race", y= "# of Participants") +
  theme(axis.text.x = element_text(angle = -25, hjust =0))
```



Problem 10

We would like to know if having a birth family supportive of one's gender identity and expression reduces the risk of suicide. Create bar plots showing the proportions of participants who have thought about killing themselves for each level of familial support. (Do not include participants who declined to answer.) What do you notice?

```r
support <- transmute(samp,
                     RKey = RESPKEY,
                     Ques6 = Q6,
                     Ques119 = Q119)

support <- support %>%
  group_by(Ques6, Ques119) %>%
  summarise(count = n())

support <- support[complete.cases(support), ]

support %>%
  ggplot(aes(x=Ques6, y=count, fill=Ques119)) +
  geom_col(position = "dodge") +
  labs(title="Prop of prpts who thought of killing themselves for each level of familial support",
       x= "Gender", y= "# of Participants") +
  theme(axis.text.x = element_text(angle = -25, hjust =0))
```




```r
# Participants - prpts

# Transgenders end up killing themselves a lot more than others when their
# birth families are not at all supportive.
```