Problem Set 1

Sebastian Urbina

26 September, 2023

From the files section on CourseWorks, download the file fec22.txt, which contains data for candidate political action committees for the 2022 elections in the U.S. Use the file fec.codebook.txt to see the values for the fields. Write R code to do the following.

```
#Loading packages and erase memory
rm(list=ls())
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.3     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.4.3     v tibble    3.2.1
## v lubridate 1.9.2     v tidyr     1.3.0
## v purrr     1.0.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become error
```

```
library(haven)
library(psych)
```

```
##
## Attaching package: 'psych'
##
## The following objects are masked from 'package:ggplot2':
##
##     %+%, alpha
```

1.Read the data into a data object called fec22.df using the appropriate command. Report the number of records/observations in the data.

```
#Importing data fec22.txt from local disc
setwd("/Volumes/TOSHIBA EXT/1.1_Columbia University/Fall 2023/POLSGU4716_001_2023_3 - Data Science for
getwd()
```

```
## [1] "/Volumes/TOSHIBA EXT/1.1_Columbia University/Fall 2023/POLSGU4716_001_2023_3 - Data Science for
```

```
#Loading data
fec22.df <- read.delim("fec22.txt", header=FALSE, sep = "|")
```

```
# Number of records/observations in the data
print(dim(fec22.df))
```

```
## [1] 4027   30
```

```
#Names of variables
names(fec22.df)
```

```
##  [1] "V1"  "V2"  "V3"  "V4"  "V5"  "V6"  "V7"  "V8"  "V9"  "V10" "V11" "V12"
## [13] "V13" "V14" "V15" "V16" "V17" "V18" "V19" "V20" "V21" "V22" "V23" "V24"
## [25] "V25" "V26" "V27" "V28" "V29" "V30"
```

```
# Characteristics of variables
str(fec22.df) #V21-v25 variables with constant NA values
```

```
## ’data.frame’:    4027 obs. of  30 variables:
##  $ V1 : chr  "H2AK00200" "H2AK01158" "H2AK01240" "H2AK00218" ...
##  $ V2 : chr  "CONSTANT,CHRISTOPHER" "PELTOLA,MARY" "WOOL, ADAM L" "REVAK, JOSHUA CARL" ...
##  $ V3 : chr  "C" "I" "O" "O" ...
##  $ V4 : int  1 1 1 2 2 2 2 2 2 2 ...
##  $ V5 : chr  "DEM" "DEM" "DEM" "REP" ...
##  $ V6 : num  164638 7751293 16217 121841 1971161 ...
##  $ V7 : num  0 186868 0 0 112963 ...
##  $ V8 : num  164638 7060033 16217 121841 1924781 ...
##  $ V9 : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ V10: num  0 0 0 0 0 ...
##  $ V11: num  0 691260 0 0 46380 ...
##  $ V12: num  615 25 1100 0 0 ...
##  $ V13: num  0 0 0 0 0 0 650000 0 0 0 ...
##  $ V14: num  0 0 0 0 0 0 0 0 0 0 ...
##  $ V15: num  0e+00 0e+00 0e+00 0e+00 0e+00 0e+00 2e+05 0e+00 0e+00 0e+00 ...
##  $ V16: num  0 0 0 0 0 0 0 0 0 0 ...
##  $ V17: num  143180 0 0 0 2525 ...
##  $ V18: num  158023 7149826 15117 116666 1770698 ...
##  $ V19: chr  "AK" "AK" "AK" "AK" ...
##  $ V20: int  1 1 1 1 1 1 1 1 1 1 ...
##  $ V21: logi  NA NA NA NA NA NA ...
##  $ V22: logi  NA NA NA NA NA NA ...
##  $ V23: logi  NA NA NA NA NA NA ...
##  $ V24: logi  NA NA NA NA NA NA ...
##  $ V25: logi  NA NA NA NA NA NA ...
##  $ V26: num  1000 384021 0 5000 81305 ...
##  $ V27: num  5000 10000 0 0 0 0 0 0 0 0 ...
##  $ V28: chr  "12/31/2022" "12/31/2022" "07/15/2022" "09/16/2022" ...
##  $ V29: num  8300 136658 0 14600 43128 ...
##  $ V30: num  0 3913 0 0 1000 ...
```

```r
head(fec22.df) # first observations
```

```
##          V1                   V2 V3 V4  V5         V6       V7         V8 V9
## 1 H2AK00200 CONSTANT,CHRISTOPHER  C  1 DEM  164637.90      0.0  164637.90  0
## 2 H2AK01158         PELTOLA,MARY  I  1 DEM 7751293.39 186868.2 7060033.09  0
## 3 H2AK01240         WOOL, ADAM L  O  1 DEM   16217.07      0.0   16217.07  0
## 4 H2AK00218   REVAK, JOSHUA CARL  O  2 REP  121841.00      0.0  121841.00  0
## 5 H2AK00226         PALIN, SARAH  O  2 REP 1971160.93 112963.4 1924781.35  0
## 6 H2AK01059        PURHAM, RANDY  C  2 REP    1548.51      0.0    5621.60  0
##   V10       V11     V12 V13 V14 V15 V16       V17        V18 V19 V20 V21 V22
## 1   0      0.00  614.85   0   0   0   0 143180.09  158023.05  AK   1  NA  NA
## 2   0 691260.30   25.00   0   0   0   0      0.00 7149826.02  AK   1  NA  NA
## 3   0      0.00 1100.00   0   0   0   0      0.00   15117.00  AK   1  NA  NA
## 4   0      0.00    0.00   0   0   0   0      0.00  116666.00  AK   1  NA  NA
## 5   0  46379.58    0.00   0   0   0   0   2525.05 1770697.90  AK   1  NA  NA
## 6 140      0.00    0.00   0   0   0   0      0.00    1548.51  AK   1  NA  NA
##   V23 V24 V25      V26   V27        V28       V29     V30
## 1  NA  NA  NA   1000.0  5000 12/31/2022   8300.00    0.00
## 2  NA  NA  NA 384020.6 10000 12/31/2022 136657.70 3912.66
## 3  NA  NA  NA      0.0     0 07/15/2022      0.00    0.00
## 4  NA  NA  NA   5000.0     0 09/16/2022  14600.00    0.00
## 5  NA  NA  NA  81305.0     0 12/31/2022  43128.37 1000.00
## 6  NA  NA  NA      0.0     0 07/27/2022      0.00    0.00
```

```r
tail(fec22.df)# last observations
```

```
##             V1                         V2 V3 V4  V5        V6       V7
## 4022 S0WY00129               LUDWIG, YANA  O  1 DEM       0.0      0.0
## 4023 S0WY00152           BEN DAVID, MERAV  O  1 DEM       0.0      0.0
## 4024 S0WY00137 LUMMIS, CYNTHIA MARIE MRS.  I  2 REP  419107.3 103175.0
## 4025 S4WY00147              MILLER, BRYAN  O  2 REP       0.0      0.0
## 4026 S6WY00068           BARRASSO, JOHN A  I  2 REP 1881044.4 151890.8
## 4027 S6WY00126            ENZI, MICHAEL B  I  2 REP     410.0      0.0
##              V8 V9        V10       V11 V12 V13 V14    V15 V16      V17
## 4022     596.96  0   11973.28       0.0   0   0   0      0   0     0.00
## 4023   14278.00  0   14278.00       0.0   0   0   0   3830   0     0.00
## 4024  417700.81  0  114450.23  115856.8   0   0   0 140500   0     0.00
## 4025       0.00  0       0.00       0.0   0   0   0      0   0 30901.51
## 4026 1515357.44  0 4121889.51 4487576.5   0   0   0      0   0     0.00
## 4027  253421.57  0  253011.57       0.0   0   0   0      0   0     0.00
##            V18 V19 V20 V21 V22 V23 V24 V25      V26 V27        V28     V29
## 4022       0.0  WY   1  NA  NA  NA  NA  NA      0.0   0 05/24/2022    0.00
## 4023       0.0  WY   1  NA  NA  NA  NA  NA      0.0   0 04/13/2021    0.00
## 4024  190264.8  WY   1  NA  NA  NA  NA  NA 124667.6   0 12/31/2022 1460.25
## 4025       0.0  WY   1  NA  NA  NA  NA  NA      0.0   0 06/30/2022    0.00
## 4026 1113785.7  WY   1  NA  NA  NA  NA  NA 614175.0   0 12/31/2022 1171.00
## 4027       0.0  WY   1  NA  NA  NA  NA  NA      0.0   0 09/30/2021    0.00
##        V30
## 4022     0
## 4023     0
## 4024 10000
## 4025     0
## 4026  2500
## 4027     0
```

```
view(fec22.df)
```

From this exploratory analysis, we can see that the data has 4027 observations and 30 variables. The variables V21 to V25 have constant NA. Also, we can see that the names of the variables are changed from V1 to V30, so we will use the codebook to guide us in our analysis.

2.Report any variables that are missing values systematically. Is this what you expect? Why or why not?

```
str(fec22.df) #V21-v25 variables with constant NA values
```

```
## 'data.frame':    4027 obs. of  30 variables:
##  $ V1 : chr  "H2AK00200" "H2AK01158" "H2AK01240" "H2AK00218" ...
##  $ V2 : chr  "CONSTANT,CHRISTOPHER" "PELTOLA,MARY" "WOOL, ADAM L" "REVAK, JOSHUA CARL" ...
##  $ V3 : chr  "C" "I" "O" "O" ...
##  $ V4 : int  1 1 1 2 2 2 2 2 2 2 ...
##  $ V5 : chr  "DEM" "DEM" "DEM" "REP" ...
##  $ V6 : num  164638 7751293 16217 121841 1971161 ...
##  $ V7 : num  0 186868 0 0 112963 ...
##  $ V8 : num  164638 7060033 16217 121841 1924781 ...
##  $ V9 : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ V10: num  0 0 0 0 0 ...
##  $ V11: num  0 691260 0 0 46380 ...
##  $ V12: num  615 25 1100 0 0 ...
##  $ V13: num  0 0 0 0 0 0 650000 0 0 0 ...
##  $ V14: num  0 0 0 0 0 0 0 0 0 0 ...
##  $ V15: num  0e+00 0e+00 0e+00 0e+00 0e+00 0e+00 2e+05 0e+00 0e+00 0e+00 ...
##  $ V16: num  0 0 0 0 0 0 0 0 0 0 ...
##  $ V17: num  143180 0 0 0 2525 ...
##  $ V18: num  158023 7149826 15117 116666 1770698 ...
##  $ V19: chr  "AK" "AK" "AK" "AK" ...
##  $ V20: int  1 1 1 1 1 1 1 1 1 1 ...
##  $ V21: logi  NA NA NA NA NA NA ...
##  $ V22: logi  NA NA NA NA NA NA ...
##  $ V23: logi  NA NA NA NA NA NA ...
##  $ V24: logi  NA NA NA NA NA NA ...
##  $ V25: logi  NA NA NA NA NA NA ...
##  $ V26: num  1000 384021 0 5000 81305 ...
##  $ V27: num  5000 10000 0 0 0 0 0 0 0 0 ...
##  $ V28: chr  "12/31/2022" "12/31/2022" "07/15/2022" "09/16/2022" ...
##  $ V29: num  8300 136658 0 14600 43128 ...
##  $ V30: num  0 3913 0 0 1000 ...
```

```
sum(is.na(fec22.df$V21))
```

```
## [1] 4027
```

```r
sum(is.na(fec22.df$V22))
```

```
## [1] 4027
```

```r
sum(is.na(fec22.df$V23))
```

```
## [1] 4027
```

```r
sum(is.na(fec22.df$V24))
```

```
## [1] 4027
```

```r
sum(is.na(fec22.df$V25))
```

```
## [1] 4027
```

```r
print(describe(fec22.df[,c(21:25)]))
```

```
## Converted non-numeric matrix input to numeric.  Are you sure you wanted to do this. Please check you

## Warning in FUN(newX[, i], ...): no non-missing arguments to min; returning Inf

## Warning in FUN(newX[, i], ...): no non-missing arguments to min; returning Inf

## Warning in FUN(newX[, i], ...): no non-missing arguments to min; returning Inf

## Warning in FUN(newX[, i], ...): no non-missing arguments to min; returning Inf

## Warning in FUN(newX[, i], ...): no non-missing arguments to min; returning Inf

## Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning -Inf

## Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning -Inf

## Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning -Inf

## Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning -Inf

## Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning -Inf

##      vars n mean sd median trimmed mad min  max range skew kurtosis se
## V21*    1 0  NaN NA     NA     NaN  NA Inf -Inf  -Inf   NA       NA NA
## V22*    2 0  NaN NA     NA     NaN  NA Inf -Inf  -Inf   NA       NA NA
## V23*    3 0  NaN NA     NA     NaN  NA Inf -Inf  -Inf   NA       NA NA
## V24*    4 0  NaN NA     NA     NaN  NA Inf -Inf  -Inf   NA       NA NA
## V25*    5 0  NaN NA     NA     NaN  NA Inf -Inf  -Inf   NA       NA NA
```

As mentioned, variables **V21 to V25** have systematically missing values. This can occur because these variables are from another data set, and we can hypothesize that the data was not merged correctly or that the "key" variables for the merging at the moment to realize a left_joint don't have a match in the other data set, resulting in an import of new variables with data NA. We can hypothesize this because variables v21 to V25 came from the Election result data included in the 1996-2006 files.

3. Subset the data to produce two different data objects—one for Senate candidates and one for House candidates (the variable *CAND OFFICE DISTRICT* equals 0 for Senate candidates, is greater than 0 for House candidates). Do a check that will give you a sense that the subsetting worked correctly.

```r
#Creating new variable for Sen and Hou candidates
fec22.df <- fec22.df %>%
mutate(cand = case_when(V20 == 0 ~ "Senate",
                        V20 > 0 ~ "House",
                        TRUE ~ "NA"))

table(fec22.df$cand)
```

```
##
##  House     NA Senate
##   3406      4    617
```

```r
# Creating subset for Senate candidates
sen_df <- fec22.df %>%
filter(V20 == 0) %>%
mutate(n = 1)

# Creating subset for House candidates
hou_df <-fec22.df %>%
filter(V20 > 0) %>%
mutate(n = 1)

#Cheking the subsets for correct outcome
describe(sen_df$V20)
```

```
##    vars   n mean sd median trimmed mad min max range skew kurtosis se
## X1    1 617    0  0      0       0   0   0   0     0  NaN      NaN  0
```

```r
summary(sen_df$V20)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       0       0       0       0       0       0
```

```r
sum(sen_df$n)
```

```
## [1] 617
```

```r
describe(hou_df$V20)
```

```
##    vars    n  mean    sd median trimmed  mad min max range skew kurtosis   se
## X1    1 3406 10.28 10.58      6    8.26 5.93   1  53    52 1.73     2.79 0.18
```

```r
summary(hou_df$V20)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1.00    3.00    6.00   10.28   14.00   53.00
```

```r
sum(hou_df$n)
```

```
## [1] 3406
```

4. Calculate and report the mean, median, and standard deviation for total receipts (variable name TTL RECEIPTS) for races for each chamber. Do this for the subsets produced in the previous step without using dplyr. Also do this on the original data that you read in (i.e., fec22.df) using dplyr and compare the results from the two approaches.

```r
# Creating a matrix for putting the results
results <- matrix(NA, nrow = 2, ncol = 3 )
# Calculating the mean, median and sd for Senate
results[1,1] <- round(mean(sen_df$V6),0)
results[1,2] <- round(median(sen_df$V6),0)
results[1,3] <- round(sd(sen_df$V6),0)
# Calculating the mean, median and sd for House
results[2,1] <- round(mean(hou_df$V6),0)
results[2,2] <- round(median(hou_df$V6),0)
results[2,3] <- round(sd(hou_df$V6),0)
# Naming the columns and rows
colnames(results) <- c("Mean","Median","SD")
#Naming the rows
rownames(results) <- c("Senate candidates"," House candidates")
# Printing the results into the matrix created
results <- tibble(results)
print(results)
```

```
## # A tibble: 2 x 1
##   results[,"Mean"] [,"Median"]  [,"SD"]
##              <dbl>       <dbl>    <dbl>
## 1          3018274       24984 12378427
## 2           671722       33824  2212341
```

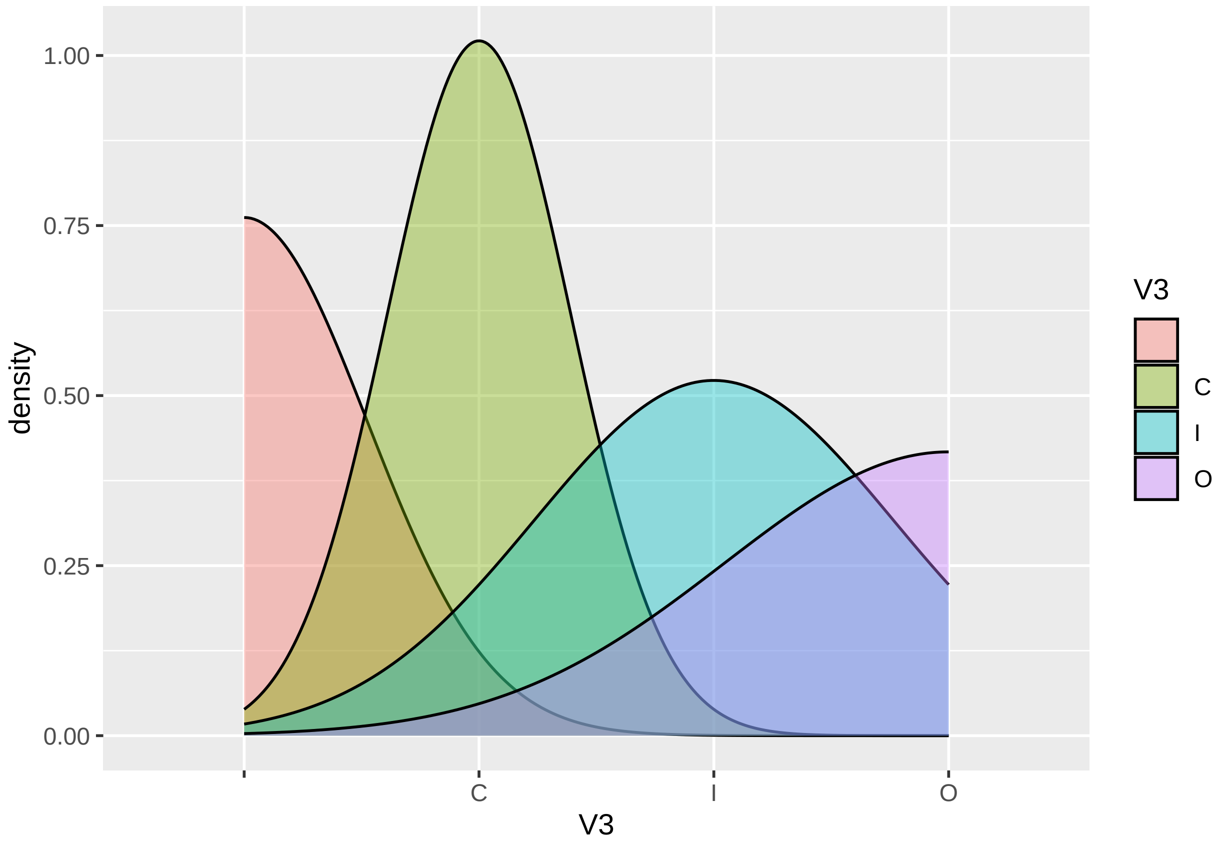
```r
# Calculating the mean, median and sd for Senate and House using dplyr and the original data, for this
print(fec22.df %>%
group_by(V20==0) %>%
summarise(mean = mean(V6, na.rm = TRUE),
          median = median(V6, na.rm = TRUE),
          sd = sd(V6, na.rm = TRUE)))
```

```
## # A tibble: 3 x 4
##   ‘V20 == 0‘      mean median        sd
##   <lgl>          <dbl>  <dbl>     <dbl>
## 1 FALSE       671722. 33824.  2212341.
## 2 TRUE       3018274. 24984  12378427.
## 3 NA            5997.  4782.     6224.
```

5. For the data that includes only House candidates, produce density plots that shows two distributions—one for candidates who are incumbents and one for candidates who are challengers. The variable CAND ICI equals “I” for incumbents, equals “C” for challengers, and equals “O” for candidates in open seat races. Write a sentence that summarizes what you see.

```
# Density Plot for House candidates, incumbents,challengers and candidates in open seats.
hou_df2 <- hou_df %>%
filter(V3 == "I" | V3 == "O" | V3 == "C")
p <- ggplot(hou_df, aes(x=V3, fill= V3)) +
  geom_density(alpha=0.4)

print(p)
```



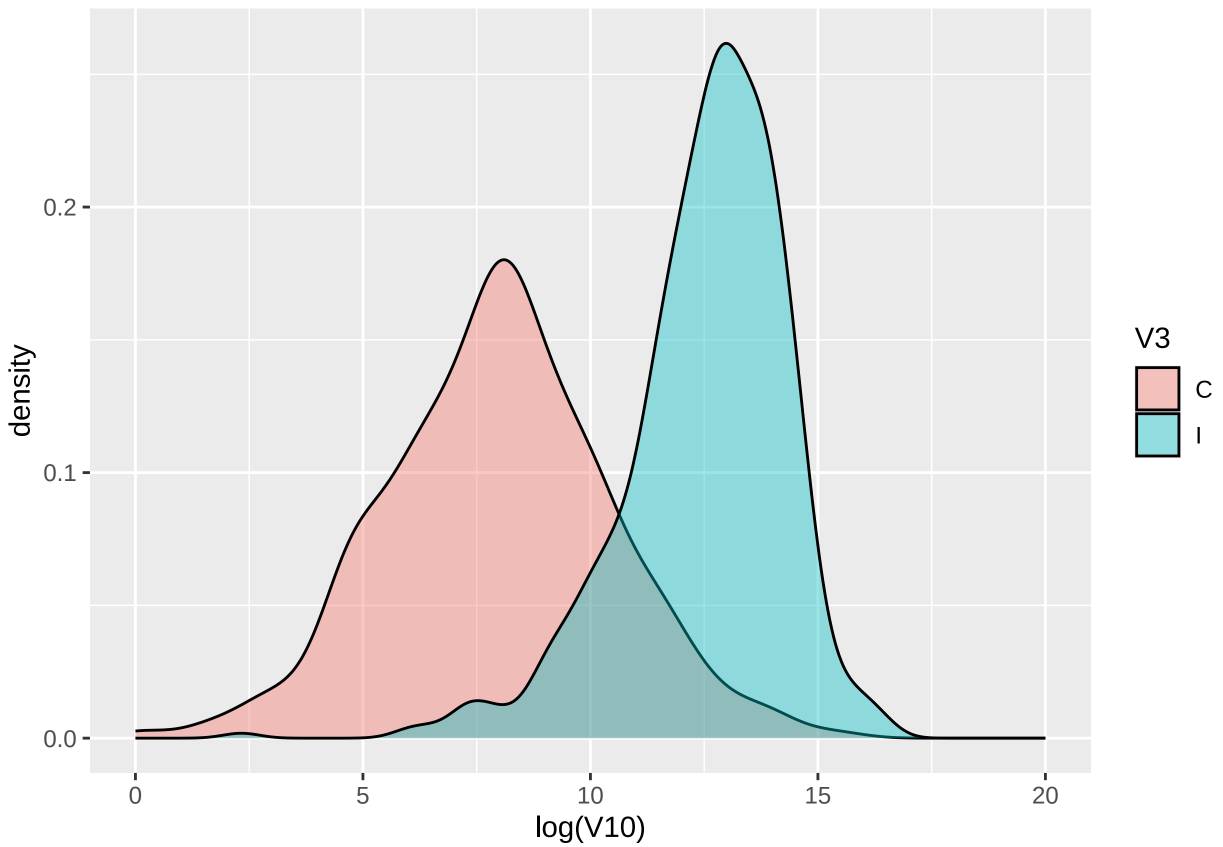
```
#Density plot for House candidates, incumbents observing the variable V10, which is the Beginning cash
hou_df2 <- hou_df %>%
filter(V3 == "I" |  V3 == "C")
p <- ggplot(hou_df2, aes(x=log(V10), y = ..density.., fill= V3)) +
  geom_density(alpha=0.4)

p  + scale_x_continuous(limits = c(0, 20))
```

```
## Warning: The dot-dot notation (‘..density..‘) was deprecated in ggplot2 3.4.0.
## i Please use ‘after_stat(density)‘ instead.
## This warning is displayed once every 8 hours.
## Call ‘lifecycle::last_lifecycle_warnings()‘ to see where this warning was
## generated.

## Warning in log(V10): NaNs produced

## Warning: Removed 1344 rows containing non-finite values (‘stat_density()‘).
```

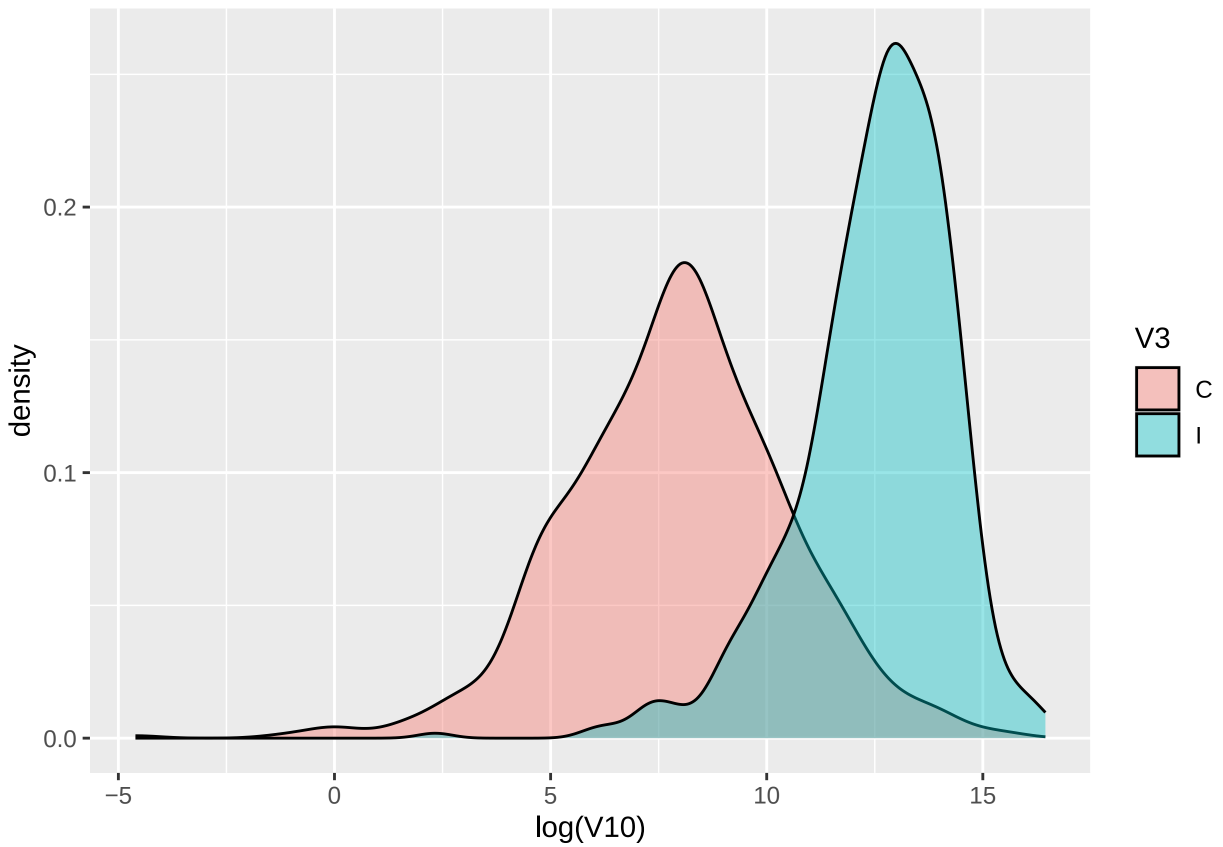


```
print(p)
```

```
## Warning in log(V10): NaNs produced

## Warning in log(V10): NaNs produced

## Warning: Removed 1340 rows containing non-finite values (‘stat_density()‘).
```



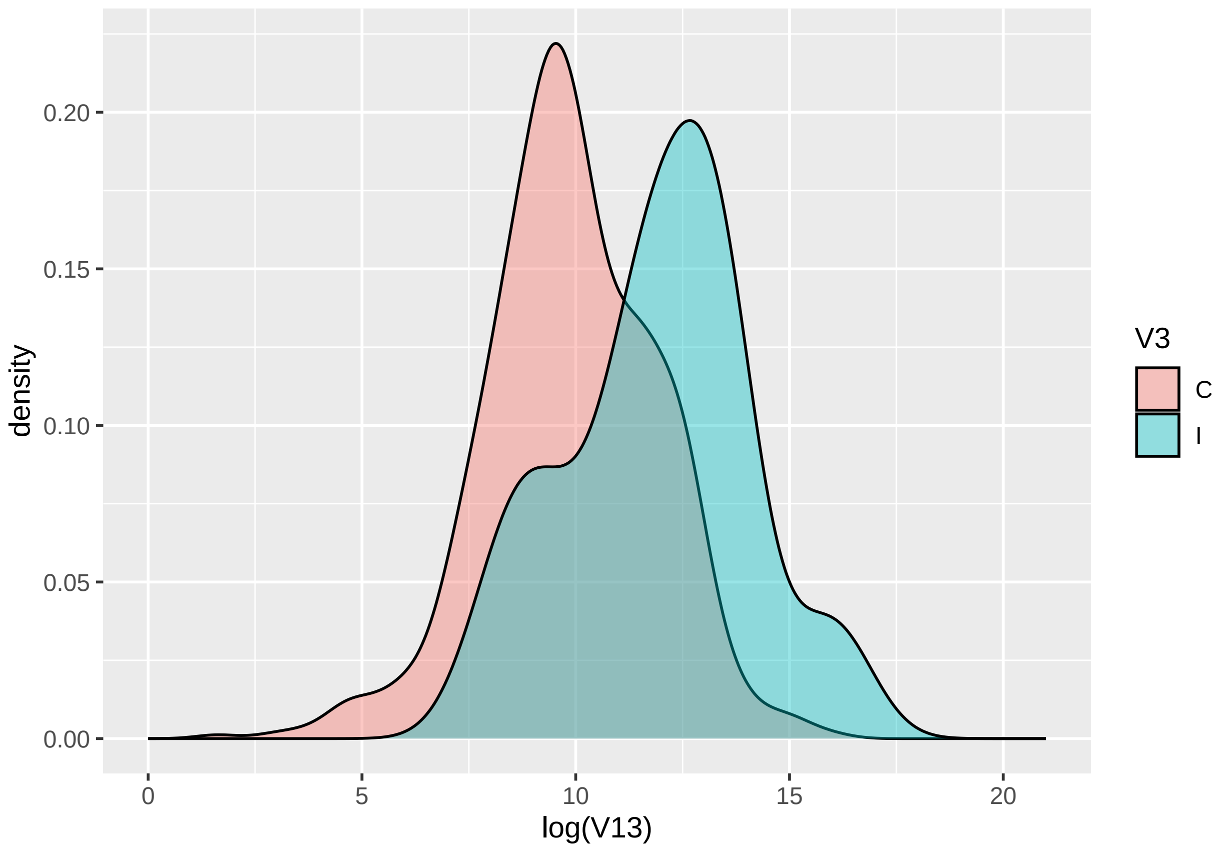
```
#Density plot for House candidates, incumbents observing the variable V10, which is the Loans from cand
hou_df2 <- hou_df %>%
filter(V3 == "I" |  V3 == "C")
p1 <- ggplot(hou_df2, aes(x=log(V13), y = ..density.., fill= V3)) +
  geom_density(alpha=0.4)

p1 <- p1 + scale_x_continuous(limits = c(0, 21))

print(p1)
```

```
## Warning in log(V13): NaNs produced
```

```
## Warning: Removed 1910 rows containing non-finite values (‘stat_density()‘).
```


density
0.20
0.15
0.10
0.05
0.00
0
5
10
15
20
log(V13)
V3
C
I