

R Code

R Markdown

This is a report of the work that has been done by Group K for the Janta ka Mood Intern. The members were Deepak, Abhijeet and Kartikeya.

```
library(tidyverse)

data <- readxl::read_xlsx("C:\\Users\\hp\\Downloads\\mp_ac_152_split.xlsx") # the electoral roll for th
data <- data[!duplicated(data$VoterID),]
form20 <- as.data.frame(readxl::read_xlsx("C:\\Users\\hp\\Downloads\\152_Form20.xlsx")) # the form20 da
```

We first calculate the vote share of both the candidates.

```
form20["Vote Share INC"] <- form20$Digvijay_Singh.INC.*100/form20$Total_Votes
form20["Vote Share BJP"] <- form20$Pragya_Thakur.BJP.*100/form20$Total_Votes
head(form20)
```

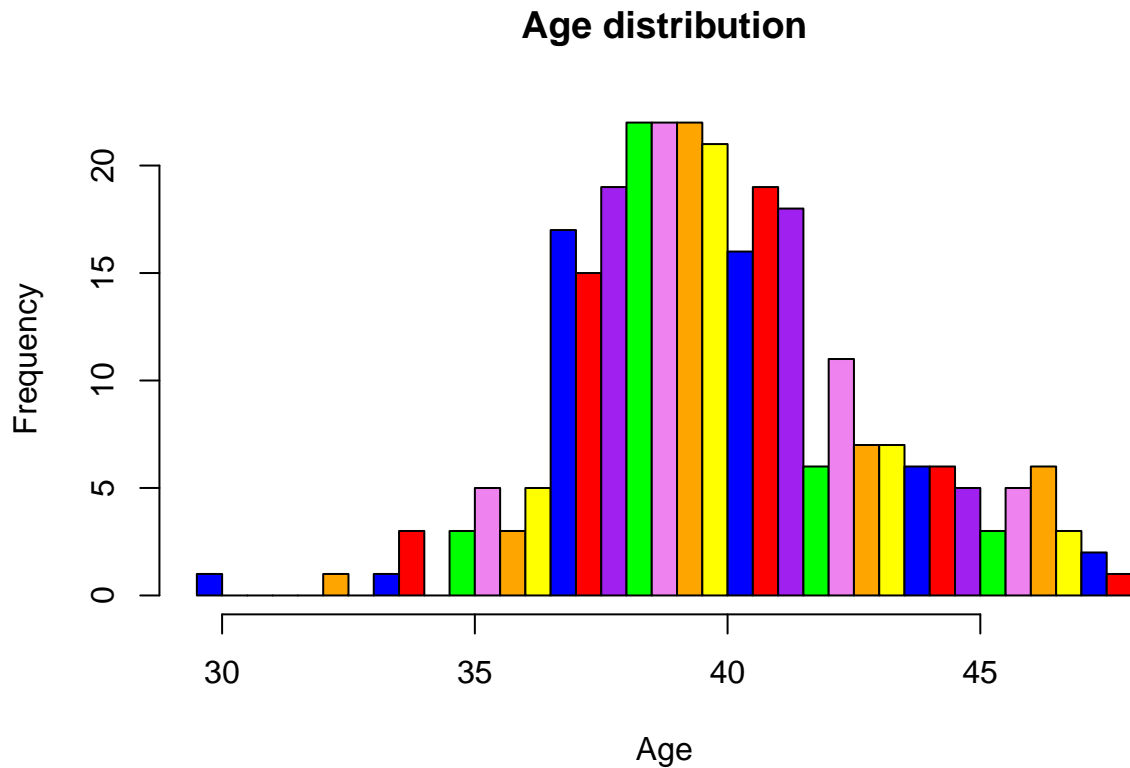
```
##   PartNo Digvijay_Singh.INC. Pragya_Thakur.BJP. Total_votes_for_candidates NOTA
## 1      0          41963          87905          133408 798
## 2      1           91          254          357 0
## 3      2           83          590          686 1
## 4      3           62          433          508 0
## 5      4          222          615          860 4
## 6      5          215          526          765 7
##   Total_Votes Vote Share INC Vote Share BJP
## 1    134206    31.26760    65.50005
## 2     357    25.49020    71.14846
## 3     687    12.08151    85.88064
## 4     508    12.20472    85.23622
## 5     864    25.69444    71.18056
## 6     772    27.84974    68.13472
```

We then calculate the average age of each polling booth.

```
avg <- aggregate(data$Age, list(data$PartNo), FUN=mean)
head(avg)
```

```
##   Group.1      x
## 1      1 37.31470
## 2      2 38.84005
## 3      3 38.33800
## 4      4 38.57179
## 5      5 37.50515
## 6      6 41.30075
```

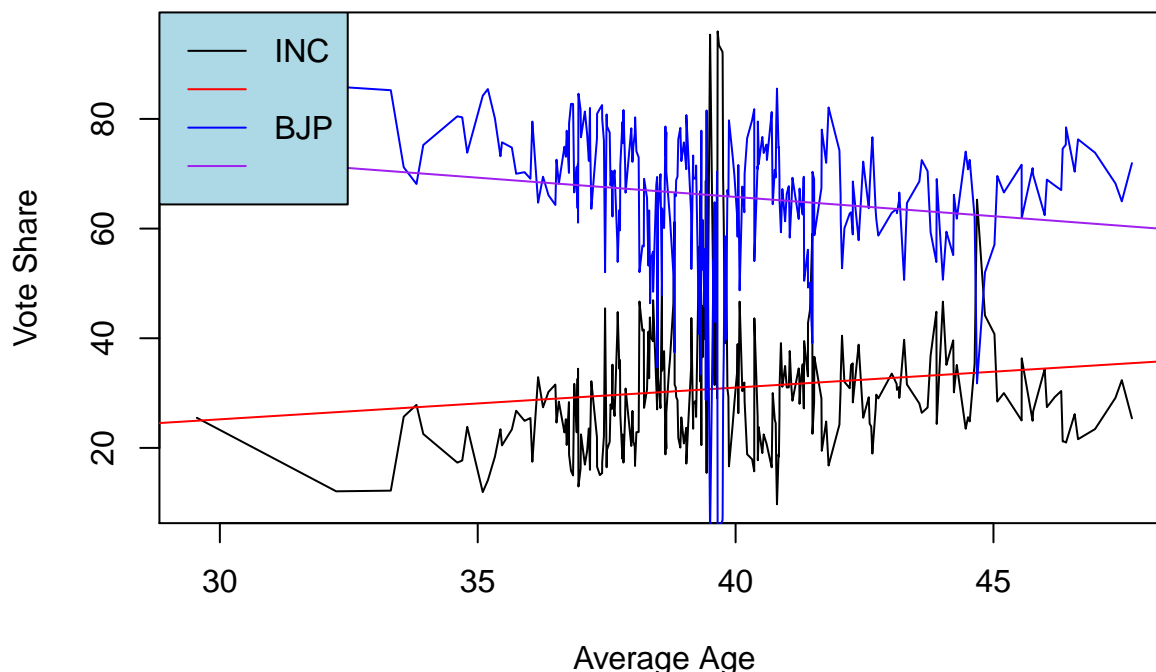
This is a plot of the frequency of the average age distribution in the overall constituency no. 152. As we can see the average age groups more or less lie within



We then ensure that the same polling booths are in each dataset and then sort the sheets with their counts and plot the polling booth wise distribution and run a regression line to give an idea about the overall effect of age on the vote share.

```
final <- avg %>% arrange(x)
form20_a <- form20[2:282,]
plot(final$x, form20_a$`Vote Share INC`, type = "l", xlab = "Average Age",
      ylab = "Vote Share", main = "Age - Combined")
abline(lm(form20_a$`Vote Share INC`~final$x), col = "red")
lines(final$x, form20_a$`Vote Share BJP`, col="blue")
abline(lm(form20_a$`Vote Share BJP`~final$x), col = "purple")
legend("topleft", legend = c("INC", " ", "BJP", " "), col = c("black", "red", "blue", "purple"),
      lty = 1, bg = "lightblue")
```

Age – Combined



We can infer from this plot that as the average age increases, the vote share for the BJP decreases and that for the INC increases. This means that voters with a higher number of elder voters prefer to vote for the Congress more than they prefer to vote for the BJP.

We move on to the gender analysis from the age analysis. We group the data by polling booths and genders to get a count of the genders in each polling booth. The third gender is being ignored for the time being.

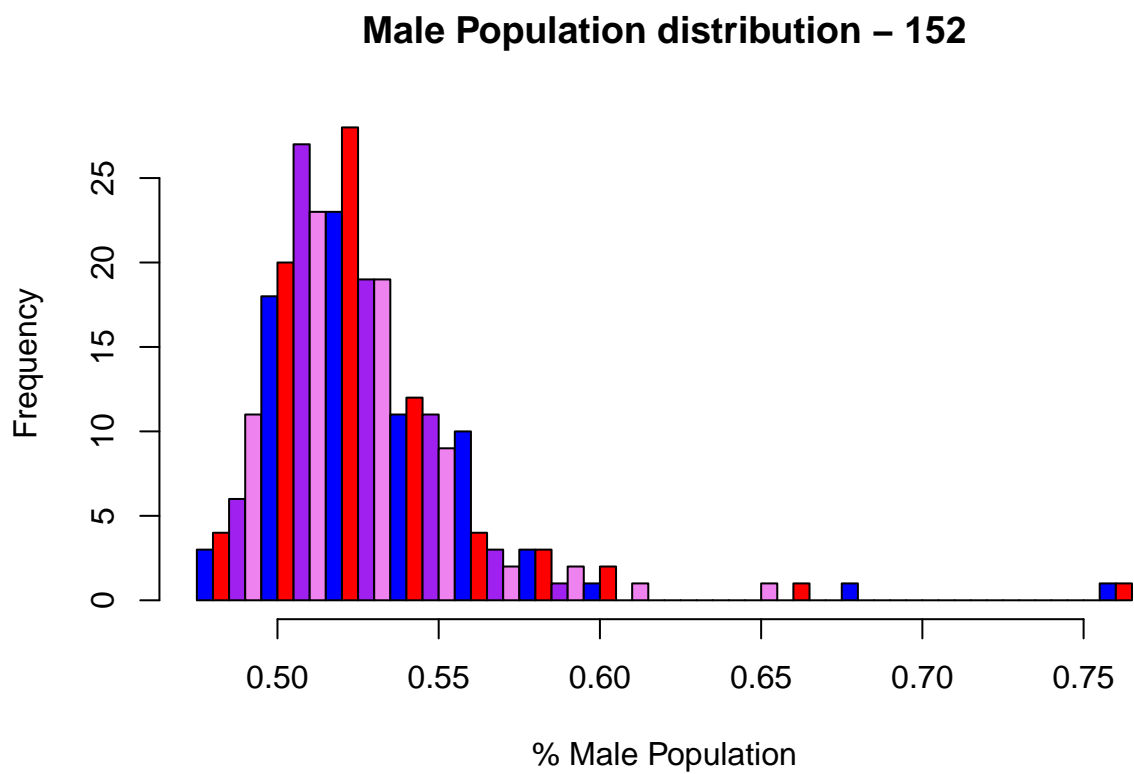
```
sex <- as.data.frame(data %>% group_by(PartNo, Sex) %>% count())
head(sex)
```

```
##   PartNo Sex    n
## 1      1  F  226
## 2      1  M  257
## 3      2  F  412
## 4      2  M  457
## 5      3  F  254
## 6      3  M  317
```

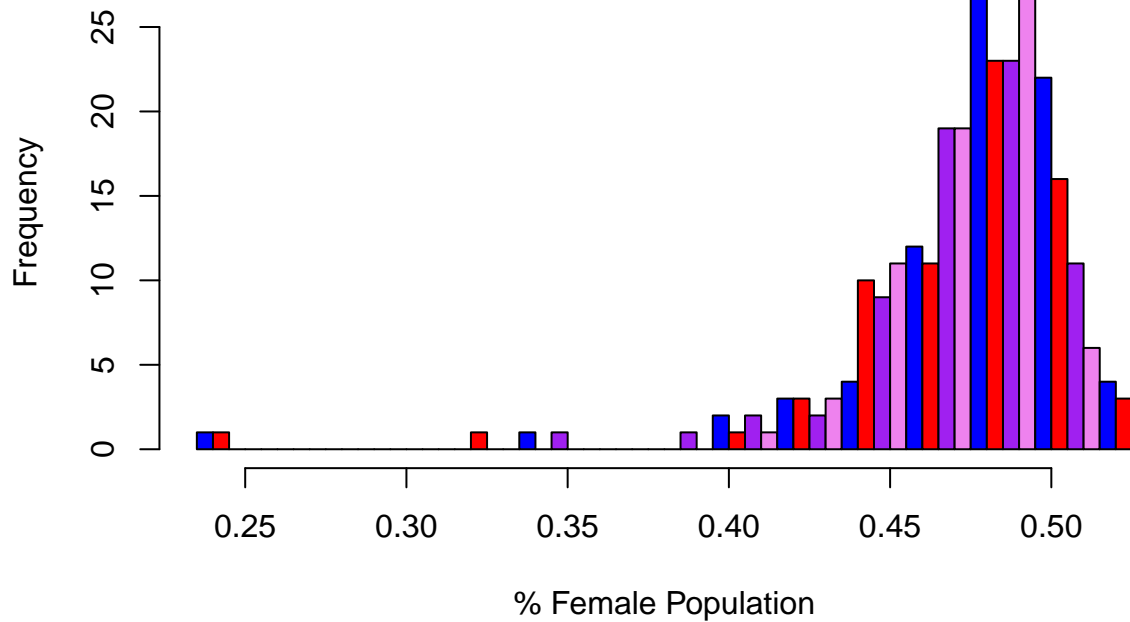
We separate the male and female numbers from the counts calculated above and thereafter calculate the percentage of male and female populations respectively.

```
female <- sex[which(sex$Sex=="F"),]
male <- sex[which(sex$Sex=="M"),]
male["Percentage"] <- male$n/(male$n+female$n)
female["Percentage"] <- female$n/(male$n+female$n)
```

This is a plot of the frequency of the percentage of male and female population in the overall constituency no. 152.



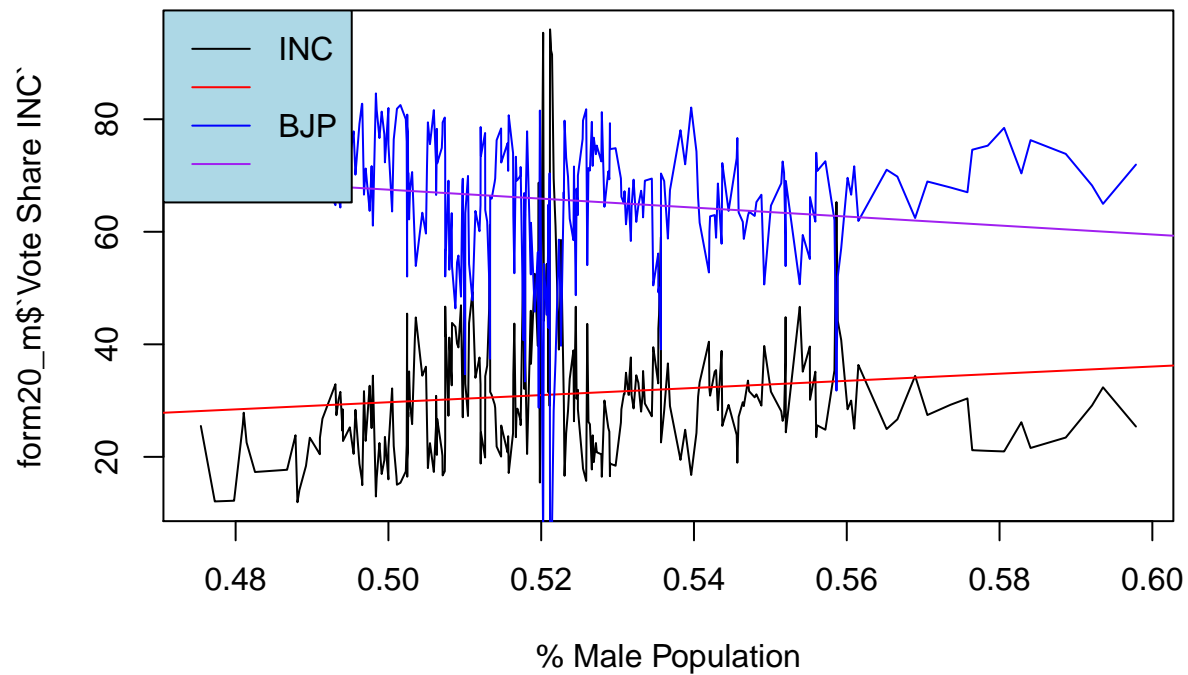
Female Population distribution – 152



We filter the outliers and We plot the vote shares of the parties against the % male population and the % female population respectively.

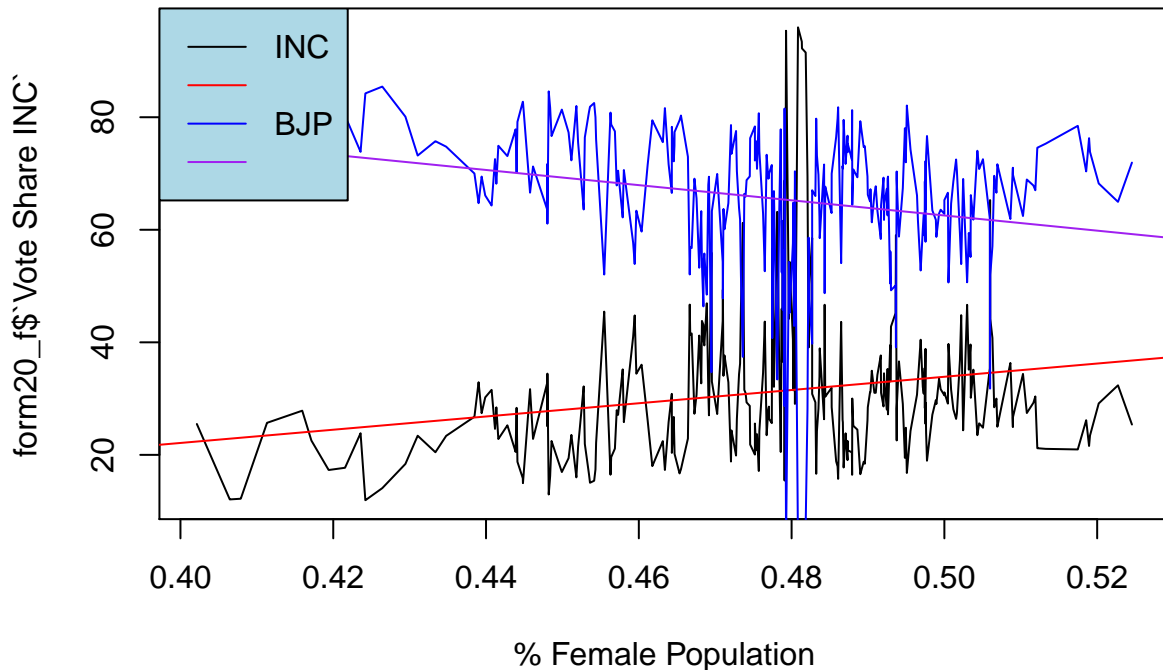
```
male <- male %>% arrange(Percentage) %>% filter(Percentage < 0.60)
form20_m <- form20[form20$PartNo %in% male$PartNo, ]
plot(male$Percentage, form20_m$`Vote Share INC`, type = 'l', main = "Male Votes Combined", xlab = "% Male Population", ylab = "Vote Share INC", col = "red")
abline(lm(form20_m$`Vote Share INC`~male$Percentage), col = "red")
lines(male$Percentage, form20_m$`Vote Share BJP`, col = "blue")
abline(lm(form20_m$`Vote Share BJP`~male$Percentage), col = "purple")
legend("topleft", legend = c("INC", " ", "BJP", " "), col = c("black", "red", "blue", "purple"),
      lty = 1, bg = "lightblue")
```

Male Votes Combined



```
female <- female %>% arrange(Percentage) %>% filter(Percentage > 0.40)
form20_f <- form20[form20$PartNo %in% female$PartNo, ]
plot(female$Percentage, form20_f$`Vote Share INC`, type = 'l', main = "Female votes Combined", xlab = "% Male Population", ylab = "form20_f$`Vote Share INC`")
abline(lm(form20_f$`Vote Share INC`~female$Percentage), col = "red")
lines(female$Percentage, form20_f$`Vote Share BJP`, col = "blue")
abline(lm(form20_f$`Vote Share BJP`~female$Percentage), col = "purple")
legend("topleft", legend = c("INC", " ", "BJP", " "), col = c("black", "red", "blue", "purple"),
      lty = 1, bg = "lightblue")
```

Female votes Combined



We scraped the internet to collect muslim first and last names in order to classify the population in the electoral on the basis of either Muslim or Hindu. The assumptions made is that the percentage of minorities that are not Hindu or Muslim is low and that of the population is not Muslim then it is bound to be Hindu.

We had pre-divided the names in the electoral roll into first, middle and last names. We first compare the last names to the dataset for muslim last names and isolate that data from the dataset. We then follow the same process with the remaining data set for the muslim first names to ensure the remaining names are also classified.

```
data2 <- readxl::read_xlsx("C:\\Users\\hp\\Downloads\\Religion Last Names.xlsx", sheet = 1)
data3 <- readxl::read_xlsx("C:\\Users\\hp\\Downloads\\Religion Last Names.xlsx", sheet = 2)

lol1 <- data[tolower(data$LastNameEng) %in% tolower(data2$Name),]
lol1["Religion"] <- "muslim"
dataf <- subset(data, !(tolower(data$LastNameEng) %in% tolower(data2$Name)))
lol2 <- dataf[tolower(dataf$FirstNameEng) %in% tolower(data3$Name),]
lol2["Religion"] <- "muslim"
dataf <- subset(dataf, !(tolower(dataf$FirstNameEng) %in% tolower(data3$Name)))
dataf["Religion"] <- "hindu"
dat <- rbind(lol1, lol2, dataf)
```

We group the combined dataset on the basis of part no. and the religion and then get the count of the population in each. We further separate the percentages of each population.

```
lmao <- as.data.frame(dat %>% group_by(PartNo, Religion) %>% count())
hindu <- lmao[which(lmao$Religion=="hindu"),]
```

```

muslim <- lmao[which(lmao$Religion=="muslim"),]

#giving values to booths in which the count is equal to 0

booth_38 <- c(38, "muslim", 0)
booth_87 <- c(87, "muslim", 0)
muslim <- rbind(muslim[1:37, ], booth_38, muslim[38:279, ])
muslim <- rbind(muslim[1:86, ], booth_87, muslim[87:280, ])
hindu$n <- as.numeric(hindu$n)
muslim$n <- as.numeric(muslim$n)

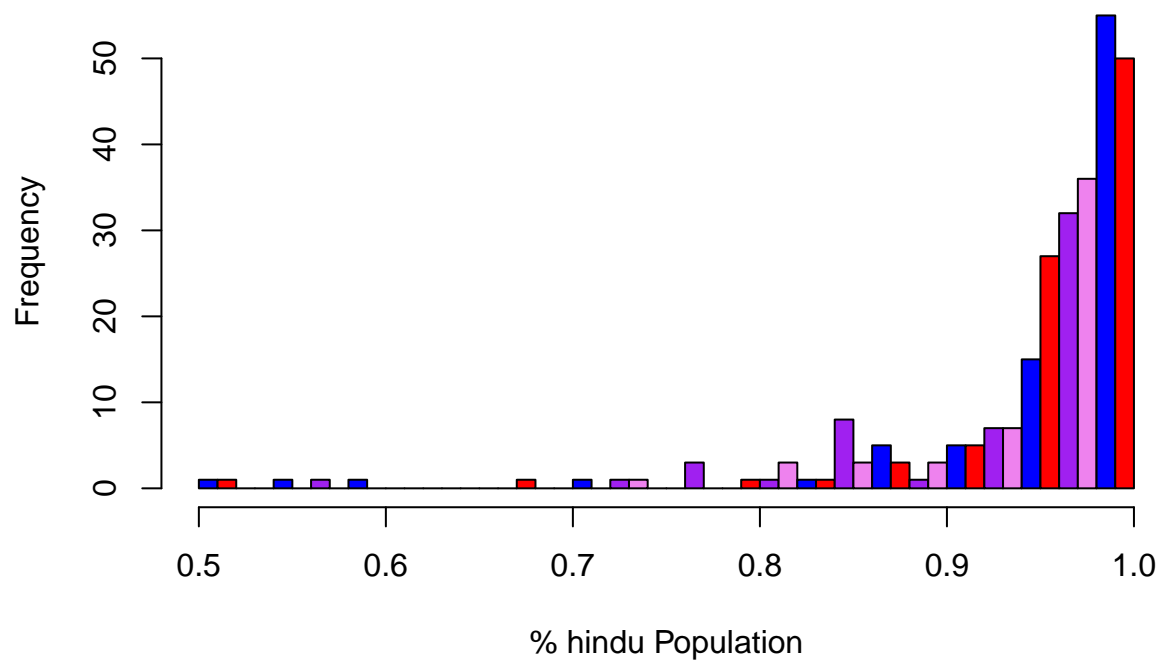
#Calculating the percentage of hindu and muslim population

hindu["Percentage"] <- hindu$n/(hindu$n + muslim$n)
muslim["Percentage"] <- muslim$n/(hindu$n + muslim$n)

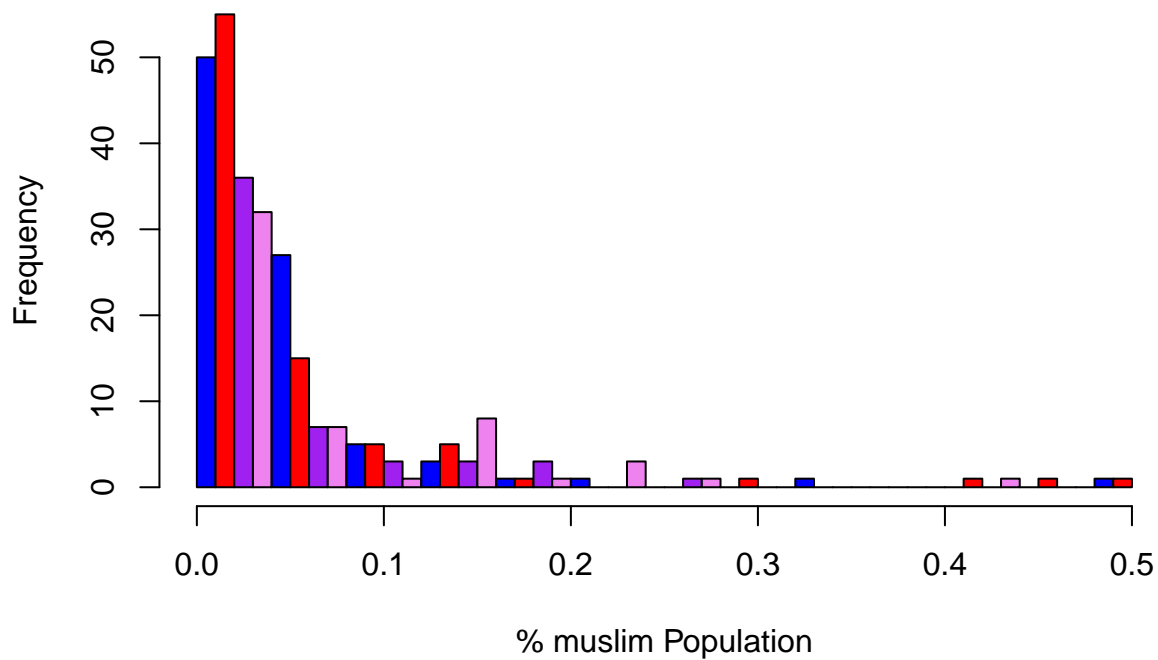
```

These are the frequency distributions of the Hindu and Muslim population in the 152 constituency

Hindu population distribution – 152



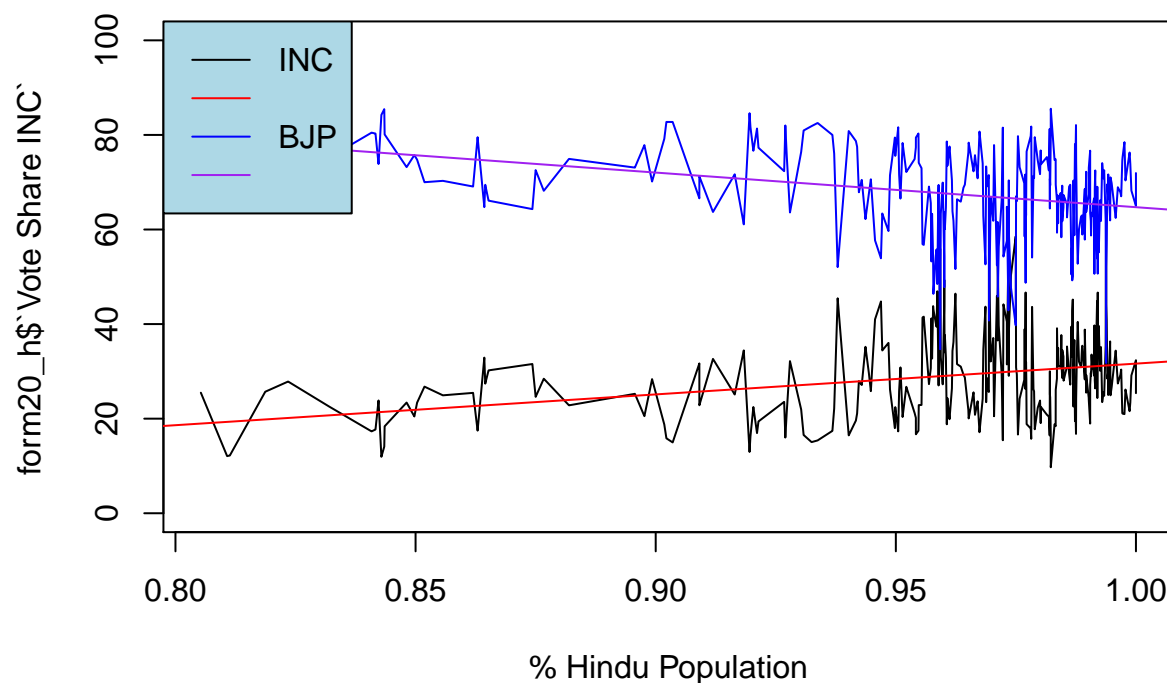
Muslim population distribution – 152



We then plot the vote share of each polling booth against the percentage of Hindu and Muslim Populations and then draw a regression line to get an overall idea of a voting relation between the parameters.

```
hindu <- hindu %>% arrange(Percentage) %>% filter(Percentage > 0.8) #arranging in order
form20_h <- form20[form20$PartNo %in% hindu$PartNo, ]
plot(hindu$Percentage, form20_h$`Vote Share INC`, ylim = c(0,100), type = 'l', main = "Hindu Votes - Cor")
abline(lm(form20_h$`Vote Share INC`~hindu$Percentage), col = "red")
lines(hindu$Percentage, form20_h$`Vote Share BJP`, col = "blue")
abline(lm(form20_h$`Vote Share BJP`~hindu$Percentage), col = "purple")
legend("topleft", legend = c("INC", " ", "BJP", " "), col = c("black", "red", "blue", "purple"),
      lty = 1, bg = "lightblue")
```

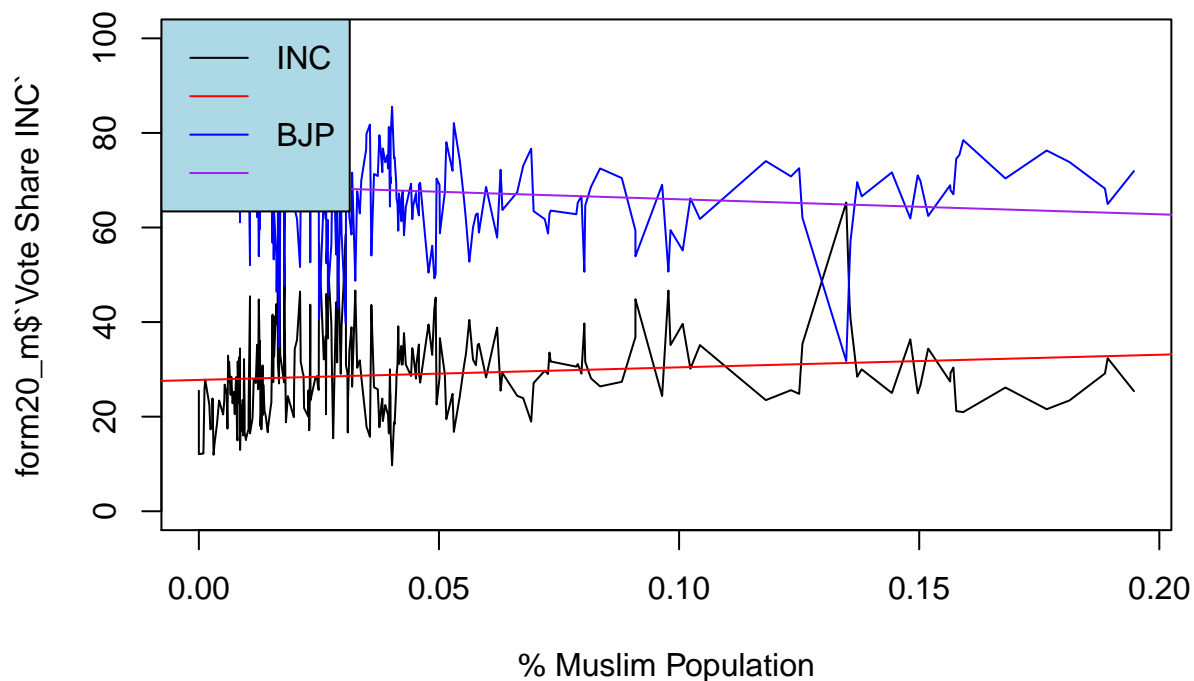
Hindu Votes – Combined



It is astounding to see that even though the average vote shares are much higher, the regression line for the INC increases as the % hindu population increases contrary to popular opinion.

```
muslim <- muslim %>% arrange(Percentage) %>% filter(Percentage < 0.2) #arranging in order
form20_m <- form20[form20$PartNo %in% muslim$PartNo, ]
plot(muslim$Percentage, form20_m$`Vote Share INC`, ylim = c(0,100), type = 'l', main = "Muslim Votes - ")
abline(lm(form20_m$`Vote Share INC`~muslim$Percentage), col = "red")
lines(muslim$Percentage, form20_m$`Vote Share BJP`, col = "blue")
abline(lm(form20_m$`Vote Share BJP`~muslim$Percentage), col = "purple")
legend("topleft", legend = c("INC", " ", "BJP", " "), col = c("black", "red", "blue", "purple"),
      lty = 1, bg = "lightblue")
```

Muslim Votes – Combined



Owing to the fact that the percentage of muslim population in 152 constituency is very less in most of the polling booths, there presence doesn't really make much of a difference as can be seen by the regression lines which have ~0 slope. However a slight decrease can be seen in the vote share of the BJP and a slight increase in that of the INC.

We then proceeded to create our R shiny dashboard. Depending on the candidate you can make a choice between Digvijay Singh and Sadhvi Pragya Thakur, the analysis changes accordingly.

Clicking the bars of the age groups leads you to the voters of that age group specifically.

We can see the percentage of voters of each demographic. as well as the vote distribution. Depending on the candidate when we select the to include the other votes, how the voting pattern would have changed can be seen in the pie chart presented.

The colour of the winning margin boxes change to specify the winning party as well

Some preliminary data manipulation has been done for ease of working

```
data <- as_data_frame(dat)
```

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

```
data <- data %>% drop_na(PollingStationAddressEn)
data$Age_Group <- ifelse(data$Age < 28, "18-27", ifelse(data$Age < 38, "28-37", ifelse(data$Age < 48, "48-57", "58+")))
colnames(data)
```

```
## [1] "...1" "ACNo"
```

```
## [3] "PartNo" "SectionNo"
## [5] "SNo" "HouseNoEn"
## [7] "HouseNo" "VoterNameEn"
## [9] "VoterName" "Sex"
## [11] "RelationNameEn" "RelationName"
## [13] "RelationType" "Age"
## [15] "VoterID" "PollingStationAddressEn"
## [17] "PollingStationAddress" "Section"
## [19] "SectionEn" "FirstNameEng"
## [21] "MiddleNameEng" "LastNameEng"
## [23] "RFirstNameEng" "RMiddleNameEng"
## [25] "RLastNameEng" "Religion"
## [27] "Age_Group"
```

```
d <- dim(data)

const <- as.list(c("Digvijay Singh", "Sadhvi Pragya Thakur"))

#Formatting the data

data$PartNo <- sprintf("%03d", as.numeric(data$PartNo))
form20$PartNo <- sprintf("%03d", as.numeric((form20$PartNo)))
head(form20$PartNo)
```

```
## [1] "000" "001" "002" "003" "004" "005"
```

```
y <- as_tibble(cbind("000 Entire Constituency", d[1]))
```

```
## Warning: The 'x' argument of 'as_tibble.matrix()' must have unique column names if '.name_repair' is
## Using compatibility '.name_repair'.
```

```
colnames(y) <- c("Booth", "n")
y
```

```
## # A tibble: 1 x 2
##   Booth          n
##   <chr>        <chr>
## 1 000 Entire Constituency 222917
```

```
data["Booth"] <- paste(data$PartNo, data$PollingStationAddressEn) #Naming the data
lol <- data %>% group_by(Booth) %>% count()
lol <- rbind(y, lol)
lol %>% drop_na(Booth)
```

```
## # A tibble: 282 x 2
##   Booth          n
##   <chr>        <chr>
## 1 000 Entire Constituency 222917
## 2 001 GOVT.PRIMARY SCHOOL BUILDING BISAN KHEDI 483
## 3 002 GOVT.PRIMARY SCHOOL BUILDING GRAM GAURA 869
## 4 003 Govt. Primary School Building Beelkheda 571
```

```
## 5 004 KOPAL H. S. SCHOOL BARKHEDI KALAN SOUTH PART ROOM NO-11      1184
## 6 005 KOPAL H. S. SCHOOL BARKHEDI KALAN EAST PART ROOM NO- 12      1166
## 7 006 KOPAL H. S. SCHOOL BARKHEDI KALAN NORTH PARTH ROOM NO - 16    399
## 8 007 GOVT. PRIMARY SCHOOL BUILDING BARKHEDI KALAN HALLNORTH PART    858
## 9 008 GOVT. PRIMARY SCHOOL BUILDING SEWANIYAN GOND EAST PART ROOM NO-2 740
## 10 009 GOVT. PRIMARY SCHOOL BUILDING SEWANIYAN GOND EAST PART ROOM NO-3 680
## # ... with 272 more rows
```

```
lol <- cbind(lol, form20)
lol["Voting Percentage"] <- (as.numeric(lol$Total_Votes)/as.numeric(lol$n))
lol["Voting Difference"] <- abs(as.numeric(lol$Digvijay_Singh.INC.) - as.numeric(lol$Pragya_Thakur.BJP.))
lol["Non Votes"] <- as.numeric(lol$Total_Votes) - (as.numeric(lol$Digvijay_Singh.INC.) + as.numeric(lol$Pragya_Thakur.BJP.))
lol$Winner <- ifelse(lol$Digvijay_Singh.INC. < lol$Pragya_Thakur.BJP., "BJP", "INC")
head(lol)
```

```
##                               Booth      n PartNo
## 1                               000 Entire Constituency 222917    000
## 2                001 GOVT.PRIMARY SCHOOL BUILDING BISAN KHEDI    483    001
## 3                002 GOVT.PRIMARY SCHOOL BUILDING GRAM GAURA    869    002
## 4                003 Govt. Primary School Building Beelkheda    571    003
## 5 004 KOPAL H. S. SCHOOL BARKHEDI KALAN SOUTH PART ROOM NO-11    1184    004
## 6 005 KOPAL H. S. SCHOOL BARKHEDI KALAN EAST PART ROOM NO- 12    1166    005
## Digvijay_Singh.INC. Pragya_Thakur.BJP. Total_votes_for_candidates NOTA
## 1                41963                87905                133408    798
## 2                 91                 254                 357      0
## 3                 83                 590                 686      1
## 4                 62                 433                 508      0
## 5                 222                 615                 860      4
## 6                 215                 526                 765      7
## Total_Votes Vote Share INC Vote Share BJP Voting Percentage Voting Difference
## 1        134206        31.26760        65.50005        0.6020447        45942
## 2          357         25.49020        71.14846        0.7391304         163
## 3          687         12.08151        85.88064        0.7905639         507
## 4          508         12.20472        85.23622        0.8896673         371
## 5          864         25.69444        71.18056        0.7297297         393
## 6          772         27.84974        68.13472        0.6620926         311
## Non Votes Winner
## 1          4338      BJP
## 2           12      BJP
## 3           14      BJP
## 4           13      BJP
## 5           27      BJP
## 6           31      BJP
```

#number of swing votes

```
hehe <- data %>%
  group_by(PartNo)%>%
  filter(Age <= 23) %>%
  tally() %>%
  pull() %>%
  as.data.frame()
```

```

huhu <- data %>%
  filter(Age <= 23) %>%
  tally() %>%
  pull() %>%
  as.data.frame()

f <- rbind(huhu, hehe)

lol["Swing"] <- f

lol["Other"] <- lol$Total_Votes - lol$Digvijay_Singh.INC. - lol$Pragya_Thakur.BJP.

foo1 <- cbind(as.data.frame(lol$Booth), lol$Digvijay_Singh.INC., lol$Pragya_Thakur.BJP. + lol$Other)
colnames(foo1) <- c("Booth", "INC - Digvijay Singh", "BJP - Sadhvi Pragya Thakur")
foo2 <- cbind(as.data.frame(lol$Booth), lol$Digvijay_Singh.INC. + lol$Other, lol$Pragya_Thakur.BJP.)
colnames(foo2) <- c("Booth", "INC - Digvijay Singh", "BJP - Sadhvi Pragya Thakur")
foo3 <- cbind(as.data.frame(lol$Booth), lol$Digvijay_Singh.INC., lol$Pragya_Thakur.BJP., lol$Other)
colnames(foo3) <- c("Booth", "INC - Digvijay Singh", "BJP - Sadhvi Pragya Thakur", "Others")

library(shiny)
library(shinydashboard)
library(ECharts2Shiny)
library(scales)
library(r2d3)

ui <- dashboardPage(
  dashboardHeader(
    title = "Polling Booth Dashboard",
    titleWidth = 200
  ),
  dashboardSidebar(
    width = 300,
    selectInput(
      inputId = "party",
      label = "Select Candidate:",
      choices = const,
      selectize = FALSE
    ),
    sidebarMenu(
      selectInput(
        inputId = "booth",
        label = "Polling Booth:",
        choices = lol[1],
        selected = 1,
        selectize = TRUE
      ),
      actionLink("remove", "Remove detail tabs")
    )
  ),
  dashboardBody(
    tags$head(tags$script(loadEChartsLibrary())),
    tabsetPanel(
      id = "tabs",

```

```

tabPanel(
  loadEChartsLibrary(),
  title = "Main Dashboard",
  value = "page1",
  fluidRow(
    valueBoxOutput("total_voters", width = 2),
    valueBoxOutput("total_votes", width = 2),
    valueBoxOutput("Vot_per", width = 2),
    valueBoxOutput("Vot_diff", width = 2),
    valueBoxOutput("Vot_non", width = 2),
    valueBoxOutput("total_swing", width = 2),
  ),
  fluidRow(
    valueBoxOutput("male_vote", width = 3),
    valueBoxOutput("female_vote", width = 3),
    valueBoxOutput("hindu_vote", width = 3),
    valueBoxOutput("muslim_vote", width = 3),
  ),
  fluidRow(),
  fluidRow(
    column(
      width = 4,
      d3Output("age_groups")
    ),
    column(
      width = 3,
      selectInput(inputId = "select",
        label = h3("Select Category"),
        choices = list("Current Vote Share" = 3, "Vote Share with others included" = 2),
        selected = 3),
      valueBoxOutput("text", width = 100),
    ),
    column(
      width = 5,
      tags$div(id="test_2", style="width:100%; height:300px;"), # Specify the div for the chart.
      deliverChart(div_id = "test_2"),
    )# Deliver the plotting
  )
)
)
)
)

server <- function(input, output, session) {
  tab_list <- NULL

  base_voters <- reactive({
    res <- data
    if(input$booth != "000 Entire Constituency") res <- filter(res, Booth ==input$booth)
    res
  })
  base_votes <- reactive({
    res1 <- lol

```

```

    res1 <- filter(res1, Booth ==input$booth)
    res1
  })
  share_votes <- reactive({
    res2 <- foo1
    res3 <- foo2
    res4 <- foo3
    if(input$select==2 && input$party== "Sadhvi Pragya Thakur"){
      rest <- filter(res2, Booth==input$booth)
      res <- data.frame(
        name = c("INC - Digvijay Singh", "BJP - Sadhvi Pragya Thakur"),
        value = c(rest$`INC - Digvijay Singh`, rest$`BJP - Sadhvi Pragya Thakur`)
      )
    }
    if(input$select==2 && input$party == "Digvijay Singh"){
      rest <- filter(res3, Booth==input$booth)
      res <- data.frame(
        name = c("INC - Digvijay Singh", "BJP - Sadhvi Pragya Thakur"),
        value = c(rest$`INC - Digvijay Singh`, rest$`BJP - Sadhvi Pragya Thakur`)
      )
    }
    if(input$select==3){
      rest <- filter(res4, Booth==input$booth)
      res <- data.frame(
        name = c("INC - Digvijay Singh", "BJP - Sadhvi Pragya Thakur", "Others"),
        value = c(rest$`INC - Digvijay Singh`, rest$`BJP - Sadhvi Pragya Thakur`, rest$Others )
      )
    }
    res
  })
  output$total_voters <- renderValueBox({
    base_voters() %>%
      tally() %>%
      pull() %>%
      as.integer() %>%
      prettyNum(big.mark = ",") %>%
      valueBox(subtitle = "Number of Voters")
  })
  output$total_swing <- renderValueBox({
    base_voters() %>%
      filter(Age <= 23) %>%
      tally() %>%
      pull() %>%
      as.integer() %>%
      prettyNum(big.mark = ",") %>%
      valueBox(subtitle = "Number of Swing Voters")
  })
  output$total_votes <- renderValueBox({
    base_voters() %>%
      select("Total_Votes") %>%
      pull() %>%
      as.integer() %>%
      prettyNum(big.mark = ",") %>%

```



```

    valueBox(subtitle = "Number of Votes Given", col = "blue")
  })
output$Vot_per <- renderValueBox({
  base_votes () %>%
    select(("Voting Percentage")) %>%
    as.numeric () %>%
    percent(accuracy = 0.01, decimal.mark = ".") %>%
    valueBox(subtitle = "Voting Percentage", col = "blue")
})
output$Vot_diff <- renderValueBox({
  haha <- base_votes()%>%select("Winner")
  if(haha == "BJP"){
    base_votes () %>%
      select(("Voting Difference")) %>%
      as.integer () %>%
      prettyNum(big.mark = ",",) %>%
      valueBox(subtitle = "Winning Margin - Sadhvi Pragya", col = "orange")
  }else if(haha == "INC"){
    base_votes () %>%
      select(("Voting Difference")) %>%
      as.integer () %>%
      prettyNum(big.mark = ",",) %>%
      valueBox(subtitle = "Winning Margin - Digvijay Singh", col = "green")
  }
})
output$Vot_non <- renderValueBox({
  base_votes () %>%
    select("Non Votes") %>%
    as.integer () %>%
    prettyNum(big.mark = ",",) %>%
    valueBox(subtitle = "Non BJP-INC Votes", col = "blue")
})
output$age_groups <- renderD3({
  res <- base_voters() %>%
    group_by(Age_Group) %>%
    tally() %>%
    collect() %>%
    mutate(
      y = n,
      x = Age_Group
    ) %>%
    select(x, y)

  res <- res %>% mutate(label = x)
  r2d3(res, "col_plot.js")
})

get_details <- function(group = NULL, religion = NULL) {
  # Create a generic details function that can be called
  # by different dashboard events
  res <- base_voters()
  if (!is.null(group)) res <- filter(res, Age_Group == group)
  if (!is.null(religion)) res <- filter(res, Religion == religion)
}

```

```

    res %>%
      select(
        Booth, HouseNoEn, VoterNameEn, Sex, Age, VoterID, Religion
      ) %>%
      group_by(HouseNoEn)%>%
      collect() %>%
      arrange(VoterNameEn, .by_group = TRUE)
  }
  observeEvent(input$column_clicked != "", {

    age_g <- input$column_clicked
    tab_title <- paste(
      input$PartNo, "- Age Group", age_g
    )
    if (!(tab_title %in% tab_list)) {
      appendTab(
        inputId = "tabs",
        tabPanel(
          tab_title,
          DT::renderDataTable(
            get_details(group = age_g)
          )
        )
      )
      tab_list <- c(tab_list, tab_title)
    }
    updateTabsetPanel(session, "tabs", selected = tab_title)
  },
  ignoreInit = TRUE
)
output$male_vote <- renderValueBox({
  base_voters() %>%
  filter(Sex == "M") %>%
  tally() %>%
  pull() %>%
  as.integer() %>%
  prettyNum(big.mark = ",") %>%
  valueBox(subtitle = "Number of Male Voters")
})
output$female_vote <- renderValueBox({
  base_voters() %>%
  filter(Sex == "F") %>%
  tally() %>%
  pull() %>%
  as.integer() %>%
  prettyNum(big.mark = ",") %>%
  valueBox(subtitle = "Number of Female Voters")
})
output$hindu_vote <- renderValueBox({
  base_voters() %>%
  filter(Religion == "hindu") %>%

```

```

    tally() %>%
    pull() %>%
    as.integer() %>%
    prettyNum(big.mark = ",") %>%
    valueBox(subtitle = "Number of Hindu Voters")

  })
  output$muslim_vote <- renderValueBox({
    base_voters() %>%
    filter(Religion == "muslim") %>%
    tally() %>%
    pull() %>%
    as.integer() %>%
    prettyNum(big.mark = ",") %>%
    valueBox(subtitle = "Number of Muslim Voters")

  })
  output$text <- renderValueBox({
    res <- share_votes()
    out <- ifelse(res[1,2]>res[2,2], "INC Wins", "BJP Wins")
    out <- out %>%
    as.character() %>%
    valueBox(subtitle = "Who wins?")

  })
  observeEvent(share_votes(),
    {
      renderPieChart(div_id = "test_2",
                     data = share_votes(),
                     radius = "70%", center_x = "50%", center_y = "50%")
    })
  observeEvent(input$remove, {
    # Use purrr's walk command to cycle through each
    # panel tabs and remove them
    tab_list %>%
    walk(~ removeTab("tabs", .x))
    tab_list <-<- NULL
  })
}

shinyApp(ui, server)

```