```
---
title: "2019 CES Election:fully voted result analysis"
author: "Feixue Han"
date: "12/16/2020"
output: pdf_document
---
#abstract
In main idea of the Canadian election survey is to show the main reasons for they way people vote for the
candidates. According to this survey, it also invests the similarity and the difference of the voting in Canada
and in other country. Furthermore, the result of the survey contributes to the science knowledge of the
motivation of the voters and the meaning of the election. Moreover, the information of altitude and opinions of
the Canadian citizens on social, economic, and political issues can be provided to public.
In the survey, we analysis the subset of 2016 statistic Canada of Education Census. The table provides education
data from Census of population according tot eh geographic and year.

#introduction
In 2019 October 13th, the 43th Canadian federal election was held and Justin Trudeau became the new Prime Minster
Canada. As a member of the Liberal Party, Justin Trudeau won 157 seats from the government. In Canada, in
37,802,043 people who have right to vote, 66% of the population voted for the election. However, there are still
nearly 33% of the people who did not vote.
In this report, we will build a MRP model base on the CES data and post-stratification data. If 100% of the
people who have the right to vote voted for the Election in 2019. The Prime Minster Justin Trudeau will have more
supporting rate during the election.
The 2019 CES data and ---- was chosen to investigate the result.  The result of the survey data shows that the
substantial influence on the Canadian Election on the Canadian president candidates.  Voting and elections are
the most basic elements of democracy. The CES has recorded Canadians’ political behaviour and altitudes and the
preference on the key political issues. The study aims to explore the important results of the Canadian election
if all the people voted for the election.

#data
```{r}
install.packages("tidyverse")
install.packages("devtools")
install.packages("haven")
devtools::install_github("hodgettsp/cesR")
install.packages("visdat")
library(tidyverse)
library(visdat)
library(cesR)
library(skimr)
library(knitr)
library(labelled)
library(haven)



get_ces("ces2019_web")
ces2019_web <- to_factor(ces2019_web)
head(ces2019_web)


library(tidyverse)
data<-read_csv('8-402-X2016010-T1-CANPR-eng.csv')
educ_cols_count<-c("Total - Highest certificate, diploma or degree (2016 counts)"
             ,"No certificate, diploma or degree (2016 counts)"
             ,"Secondary (high) school diploma or equivalency certificate (2016 counts)"
             ,"Apprenticeship or trades certificate or diploma (2016 counts)"
             ,"College, CEGEP or other non-university certificate or diploma (2016 counts)"
             ,"University certificate or diploma below bachelor level (2016 counts)")
educ_cols_percent<-c("Total - Highest certificate, diploma or degree (% distribution 2016)"
                     ,"No certificate, diploma or degree (% distribution 2016)"
                     ,"Secondary (high) school diploma or equivalency certificate (% distribution 2016)"
                     ,"Apprenticeship or trades certificate or diploma (% distribution 2016)"
                     ,"College, CEGEP or other non-university certificate or diploma (% distribution 2016)"
                     ,"University certificate or diploma below bachelor level (% distribution 2016)")
data_pivot<-data %>% select(c("Age","Sex",educ_cols))%>%
                       pivot_longer(cols=educ_cols_count, names_to='education',values_to="total_count")

raw_data_survey <- read_dta("ns20200625.dta")

#survey
raw_data_survey <- labelled::to_factor(raw_data_survey)
reduced_data_survey <-
  raw_data_survey %>%
```
```

```
  select(vote_2020,
         vote_intention,
         registration,
         age,
         gender,
         education,
         state,
         household_income,
         race_ethnicity)

reduced_data_survey$age<-as.numeric(reduced_data_survey$age)

filtered_data_survey<-reduced_data_survey %>%
                          filter(registration=="Registered"&
                                   vote_intention!="No, I am not eligible to vote"&
                                   vote_intention!="No, I will not vote but I am eligible"&
                                   (vote_2020=="yes"|vote_2020=="No")
                          )
filtered_data_survey<-na.omit(filtered_data_survey)

rm(raw_data_survey,reduced_data_survey)

#census data
raw_data_census <- read_dta("usa_00004.dta")
raw_data_census <- labelled::to_factor(raw_data_census)

reduced_data_census <-
  raw_data_census %>%
  select(perwt,
         citizen,
         age,
         sex,
         educd,
         stateicp,
         hhincome,
         race
         )

reduced_data_census$age<-as.numeric(reduced_data_census$age)

#age
filtered_data_survey<-filtered_data_survey %>%
                      mutate(agegroup = case_when(age <=20 ~ '20 or less',
                                                  age >20  & age <= 35 ~ '21 to 35',
                                                  age >35  & age <= 50 ~ '35 to 50',
                                                  age >50  & age <= 65 ~ '50 to 65',
                                                  age >65  & age <= 80 ~ '65 to 80',
                                                  age >80 ~ 'above 80'
                                                  ))
filtered_data_census<-filtered_data_census %>%
  mutate(agegroup = case_when(age <=20 ~ '20 or less',
                              age >20  & age <=  35 ~ '21 to 35',
                              age >35  & age <= 50 ~ '35 to 50',
                              age >50  & age <= 65 ~ '50 to 65',
                              age >65  & age <= 80 ~ '65 to 80',
                              age >80 ~ 'above 80'
  ))

unique(filtered_data_census$agegroup)
unique(filtered_data_survey$agegroup)


#sex and gender
unique(filtered_data_census$sex)
unique(filtered_data_survey$gender)
filtered_data_census$sex<-ifelse(filtered_data_census$sex=="female","Female","Male")

filtered_data_census<-rename(filtered_data_census,gender=sex)

unique(filtered_data_census$gender)
unique(filtered_data_survey$gender)

#education
unique(filtered_data_census$educd)
unique(filtered_data_survey$education)
```

```
  mutate(educd2 = case_when(educd =="associate's degree, type not specified" ~ 'Associate Degree',
                            educd=="doctoral degree"~'Doctorate degree',
                            educd =="master's degree" ~ 'Masters degree',
                            educd=="professional degree beyond a bachelor's degree" ~ "College Degree (such as
B.A., B.S.)",
                            educd =="bachelor's degree" ~ "College Degree (such as B.A., B.S.)",
                            educd %in% edu.somecoll~"Completed some college, but no degree",
                            educd %in% edu.highsch~"High school graduate",
                            educd %in% grade9to11~"Completed some high school",
                            educd %in% grade4to8~"Middle School - Grades 4 - 8",
                            educd %in% grade3.less ~"3rd Grade or less"
  ))

filtered_data_census<-rename(filtered_data_census,education=educd2)
filtered_data_census$educd<-NULL
unique(filtered_data_census$education)
unique(filtered_data_survey$education)

#income
x<-unique(filtered_data_survey$household_income)
min(filtered_data_census$hhincome)
max(filtered_data_census$hhincome)
filtered_data_census<-filtered_data_census %>%
  mutate(household_income = case_when(hhincome<=14999 ~ "Less than $14,999",
                                      hhincome>=15000 & hhincome<=19999~"$15,000 to $19,999",
                                      hhincome>=20000 & hhincome<=24999~"$20,000 to $24,999",
                                      hhincome>=25000 & hhincome<=29999~"$25,000 to $29,999",
                                      hhincome>=30000 & hhincome<=34999~"$30,000 to $34,999",
                                      hhincome>=35000 & hhincome<=39999~"$35,000 to $39,999",
                                      hhincome>=40000 & hhincome<=44999~"$40,000 to $44,999",
                                      hhincome>=45000 & hhincome<=49999~"$45,000 to $49,999",
                                      hhincome>=50000 & hhincome<=54999~"$50,000 to $54,999",
                                      hhincome>=55000 & hhincome<=59999~"$55,000 to $59,999",
                                      hhincome>=60000 & hhincome<=64999~"$60,000 to $64,999",
                                      hhincome>=65000 & hhincome<=69999~"$65,000 to $69,999",
                                      hhincome>=70000 & hhincome<=74999~"$70,000 to $74,999",
                                      hhincome>=75000 & hhincome<=79999~"$75,000 to $79,999",
                                      hhincome>=80000 & hhincome<=84999~"$80,000 to $84,999",
                                      hhincome>=85000 & hhincome<=89999~"$85,000 to $89,999",
                                      hhincome>=90000 & hhincome<=94999~"$90,000 to $94,999",
                                      hhincome>=95000 & hhincome<=99999~"$95,000 to $99,999",
                                      hhincome>=100000 & hhincome<=124999~"$100,000 to $124,999",
                                      hhincome>=125000 & hhincome<=149999~"$125,000 to $149,999",
                                      hhincome>=150000 & hhincome<=174999~"$150,000 to $174,999",
                                      hhincome>=175000 & hhincome<=199999~"$175,000 to $199,999",
                                      hhincome>=200000 & hhincome<=249999~"$200,000 to $249,999",
                                      hhincome>=250000~"$250,000 and above"
                                      ))

filtered_data_census$hhincome<-NULL

unique(filtered_data_census$household_income)
unique(filtered_data_survey$household_income)

#Multi-level regression
library(lme4)
library(brms)
library(tidybayes)
library(caret)
library(ROCR)

model_logit1 <- glmer(vote_2020~(1+race+agegroup|cell)+gender+education+state+household_income,
                     data = survey.data,
                     family=binomial)

summary(model_logit1)

prob.1<-predict(model_logit1,type=c('response'))
result_model1<-ifelse(prob.1>=0.5,"Joe Biden","Donald Trump")
survey.data.result<-cbind(survey.data,result_model1)

#post stratification
vote_2020_prob<-predict(model_logit1,census.data[,c("agegroup","gender","education","state",
                                          "household_income","race","cell")],type="response")
```

```
vote_2020_pred<-ifelse(vote_2020_prob>0.5,"Joe Biden","Donald Trump")
census.data.result<-cbind(census.data,vote_2020_pred)

####calculate total votes based on person weight####
census.data.result$trump_votes<-ifelse(census.data.result$vote_2020_pred=="",census.data.result$perwt,0)
census.data.result$biden_votes<-ifelse(census.data.result$vote_2020_pred=="",census.data.result$perwt,0)

```

```
#model
The purpose of our study is to predict the vote outcome of the 2910 Canadian Election Survey in CES
dataset(include citation). We used multilevel regression and post-stratification technique for this analysis. In
the following sub-sections I will describe the model specifics and the calculation for the post-stratification
process.

#result
Using a binary logistic model, we got two summary statistics tables and the probability of voting for
the Canadian Election. The data for fully voted person was summarized in the first table. The education level,
age and gender are also mentioned.

#discussion
#weekness
#reference
#appendix
#next step
```