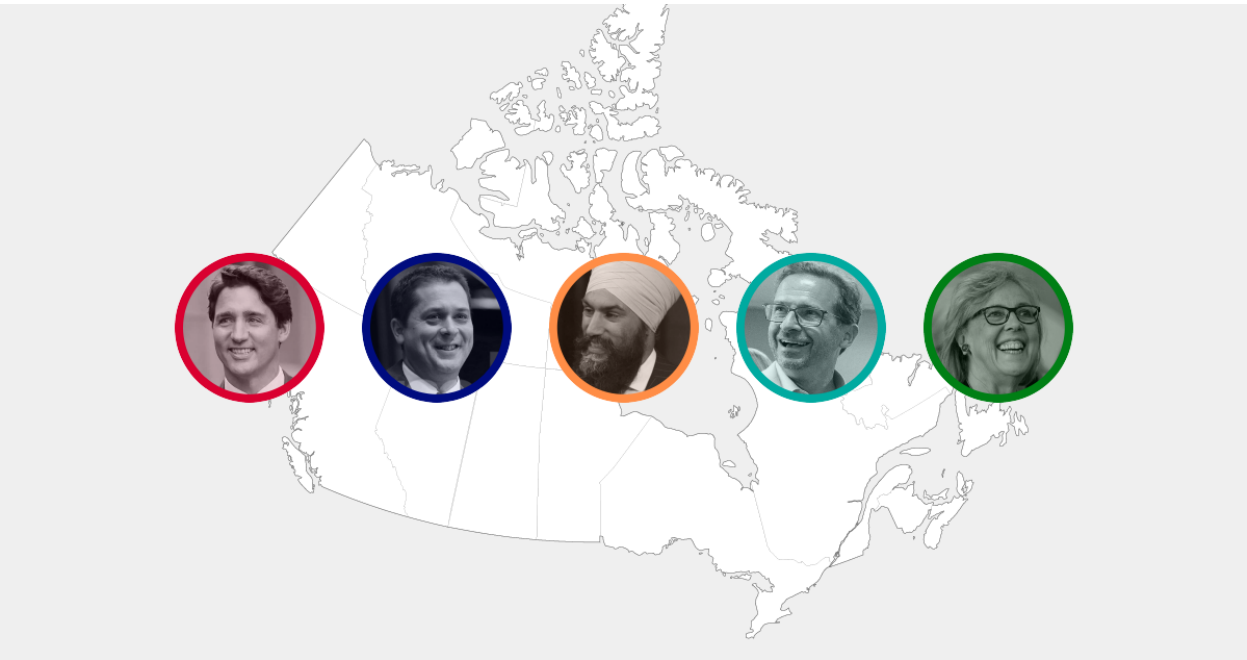
Predicting the overall popular vote for the next Canadian Federal Election using Socio-economic factors

STA304 - Assignment 2

Uzair Mirza Shruti Sood Pranav Sethi Krisha Selva

May 28, 2021



Introduction

Country List			
Country Name or Area Name	ISO ALPHA 2 Code	ISO ALPHA 3 Code	ISO numeric Code
Afghanistan	AF	AFG	004
Aland Islands	AX	ALA	248
Albania	AL	ALB	008
Algeria	DZ	DZA	012
American Samoa	AS	ASM	016
Andorra	AD	AND	020
Angola	AO	AGO	024

Problems

-Election results changing, identifying the right shift. The effects it has, [CAMBRIDGE ANALYTICA] russia, US ellection

##Basic Goal -To Identify and predict the next winner of the General Election - How this Goal helps solve the problem ##Significance <Here you should have a few paragraphs of text introducing the problem, getting the reader interested/ready for the rest of the report.>

<Introduce terminology.>

<Highlight hypotheses.>

<Optional: You can also include a description of each section of this report as a last paragraph.>

<Type here a paragraph introducing the data, its context and as much info about the data collection process that you know.>

<Type here a summary of the cleaning process (**only add in stuff beyond my original gss_cleaning.R code**). You only need to describe additional cleaning that you and your group did.>]

#Data cleaning

```
#filtering out the required columns
census_data.1 <- select(census_data, 2,12,16,17,19,28,47)
#head(census_data.1)

# adding the place of birth to our data set
census_data.2 <- census_data.1 %>% mutate(place_birth = coalesce(place_birth_macro_region, place_birth_
#[https://stackoverflow.com/questions/14563531/combine-column-to-remove-nas]

# removing the columns for place of birth
census_data.3 <- select(census_data.2, -3,-4)
#view(census_data.3)

# remove the NA
census_data.4 <- na.omit(census_data.3)
```

#Syncing data variables with Census data so that we only have the variables avalible to us

```
## EDUCATION

# combining the "Trade certificate or diploma" +"College, CEGEP or other non-university certificate or
```

```
# into "College, CEGEP, trade certificate or other non-university certificate or di..."
  # reason: to sync with the survey data categoery
census_data.4$education <- replace(census_data.4$education, census_data.4$education ==
                                     "College, CEGEP or other non-university certificate or di...",
                                   "College, CEGEP, trade certificate or other non-university certifica
census_data.4$education <- replace(census_data.4$education, census_data.4$education ==
                                     "Trade certificate or diploma", "College, CEGEP, trade certificate

## Age

# we drop the decimal point to stay consistent with the survey

census_data.4$age <- floor(census_data.4$age)
#https://stackoverflow.com/questions/40399255/remove-decimals-from-a-column-in-a-data-frame
```

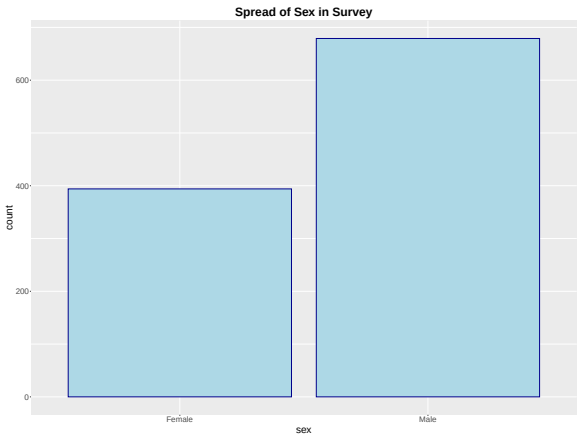
#Dropping the columns in the census data which are not present in the survey data

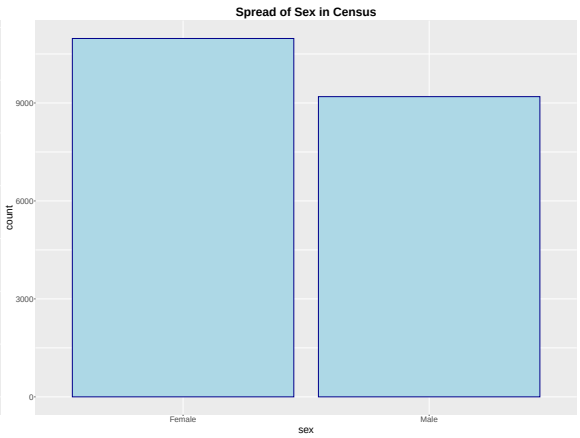
```
#census_data.8 <- census_data.7 %>%
 # select(sex, province, feelings_life, education, religion_participation, age, income_family)
```

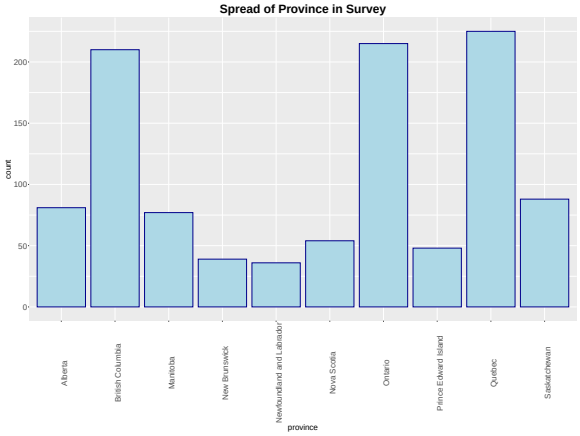
Data Analysis

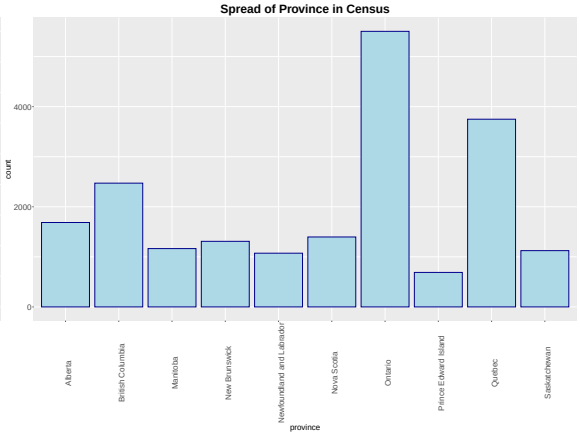
```
mytable.1 <- table(survey_data$sex)
lbls.1 <- paste(names(mytable.1), "\n", mytable.1, sep="")

mytable.2 <- table(census_data$sex)
lbls.2 <- paste(names(mytable.2), "\n", mytable.2, sep="")

# Pie Chart from data frame with Appended Sample Sizes                    [https://www.statme
# spreads of Sex, pie chart
#ggarrange(

#pie(mytable.1, labels = lbls.1,
  # main="Sex in Survey"),
#pie(mytable.2, labels = lbls.2,
  # main="Sex in Census"), ncol=2, nrow=1
#)

ggarrange(
ggplot(survey_data, aes(x=sex ))+
  geom_bar(color="darkblue", fill="lightblue", bins = 40) +
  ggtitle("Spread of Sex in Survey") + theme(axis.text=element_text(size=20),
  axis.title=element_text(size=25),plot.title= element_text(size = 30, hjust = 0.5,
  face = "bold")) ,

ggplot(census_data.4, aes(x=sex))+
  geom_bar(color="darkblue", fill="lightblue", bins = 40) +
  ggtitle("Spread of Sex in Census")+ theme(axis.text=element_text(size=20),
  axis.title=element_text(size=25),plot.title= element_text(size = 30, hjust = 0.5,
  face = "bold")),
```

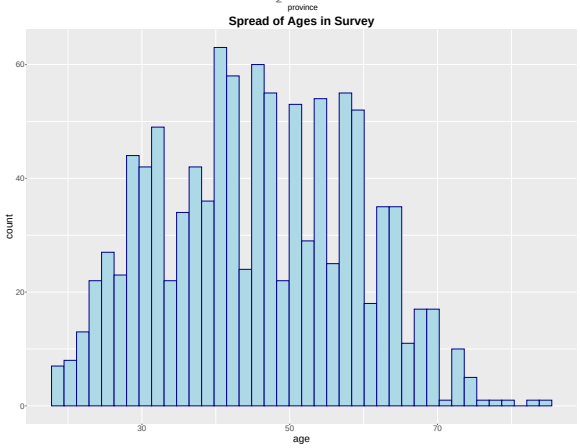
```
ggplot(survey_data, aes(x=province ))+
  geom_bar(color="darkblue", fill="lightblue", bins = 40) +
  ggtitle("Spread of Province in Survey")+ theme(axis.text=element_text(size=20),
  axis.title=element_text(size=20),plot.title= element_text(size = 30, hjust = 0.5,
  face = "bold"),axis.text.x = element_text(angle = 90)),

ggplot(census_data.4, aes(x=province))+
  geom_bar(color="darkblue", fill="lightblue", bins = 50) +
  ggtitle("Spread of Province in Census")+ theme(axis.text=element_text(size=20),
  axis.title=element_text(size=20),plot.title= element_text(size = 30, hjust = 0.5,
  face = "bold"),axis.text.x = element_text(angle = 90)),

ggplot(survey_data, aes(x=age ))+
  geom_histogram(color="darkblue", fill="lightblue", bins = 40) +
  ggtitle("Spread of Ages in Survey")+ theme(axis.text=element_text(size=20),
  axis.title=element_text(size=25),plot.title= element_text(size = 30, hjust = 0.5,
  face = "bold")),

ggplot(census_data.4, aes(x=age))+
  geom_histogram(color="darkblue", fill="lightblue", bins = 50) +
  ggtitle("Spread of Ages in Census")+ theme(axis.text=element_text(size=20),
  axis.title=element_text(size=25),plot.title= element_text(size = 30, hjust = 0.5,
  face = "bold")),

ggplot(survey_data, aes(x=income_family ))+
  geom_bar(color="darkblue", fill="lightblue", bins = 40) +
  ggtitle("Spread of Family's incomes in Survey")+ theme(axis.text=element_text(size=20),
  axis.title=element_text(size=25),plot.title= element_text(size = 30, hjust = 0.5,
  face = "bold")),

ggplot(census_data.4, aes(x=income_family))+
  geom_bar(color="darkblue", fill="lightblue", bins = 50) +
  ggtitle("Spread of Family's incomes in Census")+ theme(axis.text=element_text(size=20),
  axis.title=element_text(size=25),plot.title= element_text(size = 30, hjust = 0.5,
  face = "bold")),
 ncol=2, nrow=4)
```

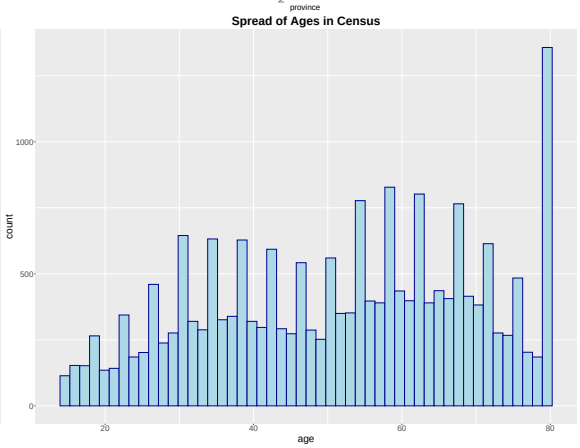

Spread of Sex in Survey
count
sex
Female
Male

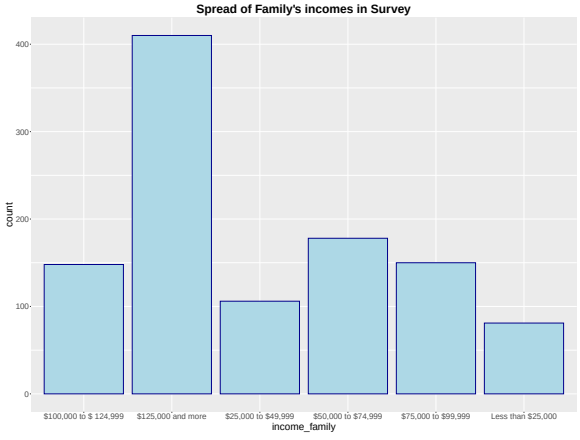

Spread of Sex in Census
count
sex
Female
Male

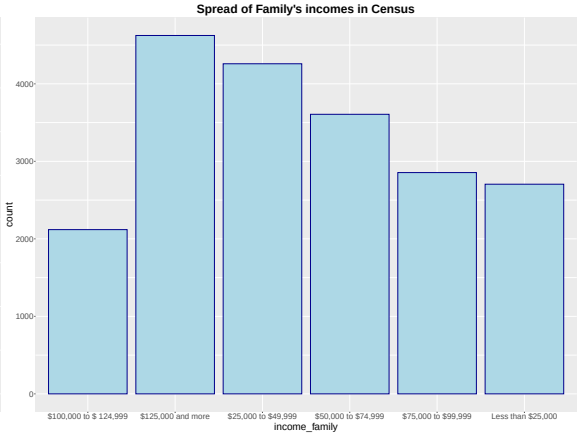

Spread of Province in Survey
count
province
Alberta
British Columbia
Manitoba
New Brunswick
Newfoundland and Labrador
Nova Scotia
Ontario
Prince Edward Island
Quebec
Saskatchewan


Spread of Province in Census
count
province
Alberta
British Columbia
Manitoba
New Brunswick
Newfoundland and Labrador
Nova Scotia
Ontario
Prince Edward Island
Quebec
Saskatchewan


Spread of Ages in Survey
count
age


Spread of Ages in Census
count
age


Spread of Family's incomes in Survey
count
income_family
$100,000 to $ 124,999
$125,000 and more
$25,000 to $49,999
$50,000 to $74,999
$75,000 to $99,999
Less than $25,000


Spread of Family's incomes in Census
count
income_family
$100,000 to $ 124,999
$125,000 and more
$25,000 to $49,999
$50,000 to $74,999
$75,000 to $99,999
Less than $25,000

```r
# get the frequency for each variable using
table(census_data.4$province)
```

```
##
##                  Alberta          British Columbia                  Manitoba
##                     1686                      2472                      1165
##           New Brunswick Newfoundland and Labrador              Nova Scotia
##                     1310                      1073                      1397
##                  Ontario      Prince Edward Island                    Quebec
##                     5504                       689                      3749
##             Saskatchewan
##                     1124
```

Methods

Regression Model Choice and Setup

The goal is to make the prediction for the upcoming election. There are alot of factors which effect someone's voting preference such as their area of residence, their place of birth, income levels, age, sex and many more. Now in order to account for and see how all these factors effect someone's decision in choosing a party to vote for we will be using **Regression**. Regression is a statistical technique which measures the relationship as to how the changes in different quantities(independent variable) effect the quantity of interest(dependent variable).
In Canada the party which forms the government is the party which wins majority of the 338 available seats. These sectors are divided and belong to different provinces, the distribution of these seats across the provinces are as follows[**https://www.elections.ca/content.aspx?section=res&dir=cir/red/allo&document=index&lang=e**]:

Table 1: Seat distribution across proviences

Provience	Seats
British Columbia	42
Alberta	34
Saskatchewan	14
Manitoba	14
Ontario	121
Quebec	78
New Brunswick	10
Nova Scotia	11
Prince Edward Island	4
Newfoundland and Labrador	7
Ontario	121
Yukon	1
Northwest Territories	1
Nunavut	1
Total	338

Based on this table we can see how the distribution of the seats is not even across the provinces, hence we need to account for this. Furthermore another thing to note is how political preference has a really high correlation with which province a person belongs to.[**https://www.jstor.org/stable/2146812?seq=1**]
Having established how the distribution of seats are not even and how the voting preference differs across

the provinces, to account for this we will be running our regression models for each province to get better results.
Another thing to note is that within each province there how voting preference shifts across different income levels[**https://pdf.sciencedirectassets.com/277811/1-s2.0-S1877042814X00030/1-s2.0-S1877042813052051/main.pdf?X-Amz-Security-Token=IQoJb3JpZ2luX2VjENP%2F%2F%2F%2F%2F%2F%2F%2F%2F%2F%2FwEaCXVzLWVhc3QtMSJGMEQCIFw2TxwYvifcSORdKy1tY9jMLmNeI2F70AiAjxJZy9H%2F8mu7WLmtusZ%2FXaRQ6xFe7Fac9nvRp0vJs%2FSr6Awh8EAQaDDA1OTAwMzU02BQ1kTcvjGoEvFHERhB9C5YKrB8%2BEba4%2FE63KiZjmclrZcskNo92tUSc8IbkM169P24Dpuw3k2ZAH2B9TBiyBnRCzHIsj7p82YexO8vI315YyQMTrGTcgfriOqKwxXBMiAb858IDCNPKG%2FD7f8Bf3wJSBGDsUlou2eyjLwN3bBso1Lv5gMXMautKRjuAn6cX2XUefGmjySP2LLSaEVBwGyLL3cv8n2FYXDWjWJsdz7TQ8E%2BLO6hefSrCUEztKUKtYj42Psj2%2Fx2bScowFnBBt4beFowD2zLnLPvdEBwqn2FeQh14a9SmxjW9kYFxwRS6adCrhctb9JAwrJlFiuv%2Fwxb40LaCcdGKayloT1QElFct8buxL%2F0Xj462Q1dYwlGkCrz9nY%2BwJEHqJdXwy%2BFzDl%2FFMWid%2Bfxqs34SkAHjyeajwtsLqTrBTb%2Fhz6cQC0uoQXJtVmgwTv%2FbEMi8RKj0khUNIcM8Wr5qCGpfnyciR7I7TwAsTXvIlafOc26F%2FYbNFjCqGjql%2BQYbMJaSyoUGOqYBWFxwr1c68ArKcOHYp70uWO2KLaI%2BIsODpeqLYoq2vQU92BIeQwesN7QUzagoSkyKZqf7vVnv8ZI64w3glYvGJ%2BWIKmAk6QtGQxefGS6mil5VIaoAxN7QXxLjpVzF2BhSj%2Fu5LHI6RemwXF4IIO1X2Uqqq0mHV4KrblzuDQfE45m4ACvjRvN38jdlVNp74oPcH4R%2Btfu%2Fe9DHiNmFECl0n78kw%3D%3D&X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Date=20210529T184210Z&X-Amz-SignedHeaders=host&X-Amz-Expires=300&X-Amz-Credential=ASIAQ3PHCVTYRNLVCUDN%2F20210529%2Fus-east-1%2Fs3%2Faws4_request&X-Amz-Signature=cfcf9f5534d9712b8f6bf183ffb373b9c483ebb2b8ed9f89da7f8c4e349d002a&hash=1d974ebb4e50b2b09c568f2da77736c59e4508311cc941cae68af32d42eea57a&host=68042c943591013ac2b2430a89b270f6af2c76d8dfd086a07176afe7c76c2c61&pii=S1877042813052051&tid=spdf-fa89cf0e-a1e8-40fb-a1f4-73f8badd3c0f&sid=6f95b77b8646a9423a3b81965227e4gxrqa&type=client**]. To account for this we will be using **Multilevel Regression**. Multilevel regression differs from normal regression on the way that in normal regression not accounting for the different levels which effect our data would have an offset in our regression line. The intercept of the line needs to be adjusted to account for the different levels. **NOTE** here in our case "level" refers to the groups within each province, which is ***family income level*** which we expect to have similar effects on the variable of interest.
Lastly another important aspect about the past elections has been that they have been dominated by particular parties. These political parties also happen to dominate and have majority number of seats in each province, hence estimating and predicting whether they win their particular provinces will acts as an estimate as to whether they will win the general election and form the government[**https://en.wikipedia.org/wiki/Politics_of_Canada#:~:text=The%20two%20dominant%20political%20parties,well%20as%20its%20numero**
To make this estimate we are going to hypothesise and setup our model using the *survey data* and train it to estimate whether someone **will be voting for a particular party or not**. To do this we will be using **Multilevel Logistic Regression**, logistic regression is used when we are dealing with a **binary response variable**.

Post stratatification and Estimation

Now in our case we will be first dicinde our **CENSUS DATA** for each provience then then into different **bins**. The concept of bins here is based on grouping the data into different groups and these are the variables which are present in our regression model. Now running this data set on the generated model for each popular party in each province will generate **log odds** of the particular group voting in favour of our hypothesised party.

Now in general the log odds generated from the regression model will look like:

$$\log(\frac{\hat{p}_{h_0}}{1-\hat{p}_{h_0}}) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_k x_k \qquad i \subset [1, k]$$

Here β_0 is the measure for the level is random intercept which accounts for the different income levels and

$ _{i}, i [1,k] $ are the other factors which effect our model.

Although as previously mentioned that there are many variables which effect someone's voting preference but due to the limitation of data we will be dividing our data set in different proviences and then for each provience we will be making our model on **family income level**, **sex**, **education level** and **age group**. So the resulting β_i will be the following:

Table 2: β_i and the corresponding measure

β_i	Variable
β_0	Family income-level
β_1	Sex
β_2	Education level
β_3	Age group

From log Odds to Estimating Probability

Now after generating the model and running the model on our test data to get the log odds we would need to extract the probability of that particular person voting for in favour of the party. To do this we would need to follow these 4 steps:

STEP 1:

$$\log(\frac{\hat{p}_{h_0}}{1-\hat{p}_{h_0}}) = \beta_0 + \Sigma_{i=1}^{k}\beta_i x_i \quad \implies \frac{\hat{p}_{h_0}}{1-\hat{p}_{h_0}} = \exp(\beta_0 + \Sigma_{i=1}^{k}\beta_i x_i)$$

STEP 2:

$$\hat{p}_{h_0} = (1-\hat{p}_{h_0})(\exp{(\beta_0 + \Sigma_{i=1}^{k}\beta_i x_i)}) \quad \implies \hat{p}_{h_0} = (\exp{(\beta_0 + \Sigma_{i=1}^{k}\beta_i x_i)}) - \hat{p}_{h_0}(\exp{(\beta_0 + \Sigma_{i=1}^{k}\beta_i x_i)})$$

STEP 3:

$$\hat{p}_{h_0} + \hat{p}_{h_0}(\exp{(\beta_0 + \Sigma_{i=1}^{k}\beta_i x_i)}) = \exp(\beta_0 + \Sigma_{i=1}^{k}\beta_i x_i) \quad \implies \hat{p}_{h_0}(1 + (\exp{(\beta_0 + \Sigma_{i=1}^{k}\beta_i x_i)}) = \exp{(\beta_0 + \Sigma_{i=1}^{k}\beta_i x_i)}$$

STEP 4:

$$\hat{p}_{h_0}(1 + (\exp{(\beta_0 + \Sigma_{i=1}^{k}\beta_i x_i)}) = \exp{(\beta_0 + \Sigma_{i=1}^{k}\beta_i x_i)} \quad \implies \hat{p}_{h_0} = \frac{\exp{(\beta_0 + \Sigma_{i=1}^{k}\beta_i x_i)}}{(1 + (\exp{(\beta_0 + \Sigma_{i=1}^{k}\beta_i x_i)})}$$

So we have:

$$\hat{p}_{h_0} = \frac{\exp{(\beta_0 + \Sigma_{i=1}^{k}\beta_i x_i)}}{(1 + (\exp{(\beta_0 + \Sigma_{i=1}^{k}\beta_i x_i)})}$$

Getting the final results **Post Stratification** Now note by grouping the data, some bins might have a higher weight that it more number of entries compared to the other bins. Hence we need to account for this, we do this using this general formula:

$$\hat{y}^{PS} = \frac{\Sigma_{i=1}^{n} N_i \hat{y}_i}{\Sigma_{i=1}^{n} N_i} \quad n \subset [2,k] \quad \hat{y}^{PS}, \hat{y}_i \subset [0,1]$$

Here $\hat{y}^{PS}$ in our case would be the **expected probability** of the total province sample voting in favor of the political party being tested now. N_i is the size of the bin and $\hat{y}_i$ is the probability assigned to the bin and finally k is the number of bins generated in our sample data.

Table 3: Popular political parties

Provience	Seats
Alberta	NDP (New Democratic Party, New Democrats, NDPers) Conservatives (Tory, PCs, Conservative Party of Canada) Liberal (Grits)
British Columbia	NDP (New Democratic Party, New Democrats, NDPers) Conservatives (Tory, PCs, Conservative Party of Canada) Liberal (Grits)
Ontario	NDP (New Democratic Party, New Democrats, NDPers) Conservatives (Tory, PCs, Conservative Party of Canada) Liberal (Grits) NDP (New Democratic Party, New Democrats, NDPers)
Manitoba	Conservatives (Tory, PCs, Conservative Party of Canada) Liberal (Grits)
Quebec	(Coalition Avenir Québec or the CAQ)* (Quebec Liberal Party)* Bloc Québécois (BQ, PQ, Bloc, Parti Québéc
New Brunswick	Conservatives (Tory, PCs, Conservative Party of Canada) Liberal (Grits) NDP (New Democratic Party, New Democrats, NDPers)
Saskatchewan	Conservatives (Tory, PCs, Conservative Party of Canada) Liberal (Grits) NDP (New Democratic Party, New Democrats, NDPers)
Nova Scotia	Conservatives (Tory, PCs, Conservative Party of Canada) Liberal (Grits) NDP (New Democratic Party, New Democrats, NDPers)
New Brunswick	NDP (New Democratic Party, New Democrats, NDPers) Liberal (Grits) Conservatives (Tory, PCs, Conservative Party of Canada)

Results

As mentioned we are generating results for every province, so here we need to first outline the popular political parties in each province. Here is a table of the popular parties in each province[]:

*Note the given data was not available so we choose the overall popular parties in the region which are Liberals and Conservatives.

Note that to make the bins and have a healthy number of people in each bin we ended up grouping people in different age groups instead of leaving them as it is. Hence we ended up with grouping the different ages, the groups that we had are

Alberta

NDP (New Democratic Party, New Democrats, NDPers)

Based on this we

grouping based on age for bin size

```
#removing ppl less than 18 to match with                                    ---Move to data cleaning--- **L
 census_data.44 <- subset(census_data.4, age >18)
  #census_data.4[census_data.4$age > 18, ]

# age group for census data                                                    --- Model limitation for bin
census_data.5 <- census_data.44 %>%
 mutate(age_group = case_when(census_data.44$age <18 ~ "Under 18",
                              census_data.44$age <25 ~ "18 to 24 years",
                              census_data.44$age <45 ~ "25 to 44 years",
                              census_data.44$age <65 ~ "45 to 64 years",
                              census_data.44$age >64 ~ "Over 65",
                              ))
# dropping place of birth and their age
census_data.6 <- select(census_data.5, -6, -1)


# survey data grouping based on age                                          ---Model Limitation----
survey_data.1 <- survey_data %>%
 mutate(age_group = case_when(survey_data$age <18 ~ "Under 18",
                              survey_data$age <25 ~ "18 to 24 years",
                              survey_data$age <45 ~ "25 to 44 years",
                              survey_data$age <65 ~ "45 to 64 years",
                              survey_data$age >64 ~ "Over 65",
                              ))

# removing feeling life, religion participation, age                         --- CHECK maybe a better model
survey_data.2 <- select(survey_data.1, -3, -6)
survey_data.2 <- select(survey_data.2, -5)


# prob function                            --add this in the model---
est_p <- function(sum){
  return(exp(sum)/(1+(exp(sum))))
}
```

#Alberta ##NDPxAlberta

```r
# getting the NDP survey data to make the models
NDP_predic.1 <- survey_data.2 %>%
  mutate(NDP_pref = case_when(survey_data.2$political_pref == "NDP (New Democratic Party, New Democrats
                              survey_data.2$political_pref != "NDP (New Democratic Party, New Democrats

# group the survey data by the provience filter out Alberta ppl       --Filtering out the NDP survey da
AlbertaXsurveyXNDP <- NDP_predic.1[NDP_predic.1$province == "Alberta",]
# make the model for predicting the results on Alberta,
glm_NDP_model.2<-glmer(NDP_pref ~ sex + education + (1|income_family) + age_group, data=AlbertaXsurveyX
#results
#summary(glm_NDP_model.2)

## assign bins and get the counts for the census data

# filter out for Alberta in the census data
albertaXcensus <- census_data.5[census_data.5$province =="Alberta",]
# get the bin count corresdonidng to the model we make
albertaXcensus.1 <- albertaXcensus %>%
  count(sex, education,  income_family, age_group)

# make the logg odds for the regression model
X <- predict(glm_NDP_model.2, albertaXcensus.1)
#view(X)
# appending the log odds to the  NDPxAlberta
albertaXcensus.1$NDP_odds <- X
# getting the probability for each group
albertaXcensus.1 <- albertaXcensus.1 %>% mutate(prob_NDP = est_p(NDP_odds))

## post-strart for the albertaX_NDP

## sum of N                              --- add this in the model---
N <- sum(albertaXcensus.1$n)
# prediction for NDPxAlberta
NDP_Vote_percen_Alberta <- albertaXcensus.1 %>% summarise(NDP_est_Alberta = (sum(n*prob_NDP)/N))
NDP_Vote_percen_Alberta
```

```
## # A tibble: 1 x 1
##   NDP_est_Alberta
##             <dbl>
## 1           0.250
```

##ConservativesXAlberta

```r
## Making the model and the predictions

# getting the NDP survey data to make the models
Conservatives_predic.1 <- survey_data.2 %>%
  mutate(Consv_pref = case_when(survey_data.2$political_pref == "Conservatives (Tory, PCs, Conservative
                              survey_data.2$political_pref != "Conservatives (Tory, PCs, Conservative Pa

# filter out based on Alberta
# group the survey data by the provience filter out Alberta ppl       --Filtering out the NDP survey da
AlbertaXsurveyXConsv <- Conservatives_predic.1[Conservatives_predic.1$province == "Alberta",]
```

```r
# make the model for predicting the results on Alberta,
glm_Consv_model<-glmer(Consv_pref ~ sex + education + (1|income_family) + age_group, data=AlbertaXsurvey
#results
#summary(glm_Consv_model)

## now assign bins and make the prdicitons
# filter out for Alberta in the census data
albertaXcensus <- census_data.5[census_data.5$province == "Alberta",]    #### Can remove this line##
# get the bin count corresdonidng to the model we make
albertaXcensus.1 <- albertaXcensus %>%
  count(sex, education,  income_family, age_group)        #### Can remove this line##

# make the logg odds for the regression model
X <- predict(glm_Consv_model, albertaXcensus.1)
#view(X)
# appending the log odds to the  NDPxAlberta
albertaXcensus.1$Consv_odds <- X
# getting the probability for each group
albertaXcensus.1 <- albertaXcensus.1 %>% mutate(prob_Consv = est_p(Consv_odds))

## post-strart for the albertaX_Consv

## sum of N                              --- add this in the model---
N <- sum(albertaXcensus.1$n)
# prediction for NDPxAlberta
Consv_Vote_percen_Alberta <- albertaXcensus.1 %>% summarise(Consv_est_Alberta = (sum(n*prob_Consv)/N))
Consv_Vote_percen_Alberta
```

```
## # A tibble: 1 x 1
##   Consv_est_Alberta
##               <dbl>
## 1             0.678
```

##LiberalXAlberta

```r
## Making the model and the predictions

# getting the NDP survey data to make the models
Liberal_predic.1 <- survey_data.2 %>%
  mutate(Lib_pref = case_when(survey_data.2$political_pref == "Liberal (Grits)" ~ 1,
                              survey_data.2$political_pref != "Liberal (Grits))" ~ 0))

# filter out based on Alberta
# group the survey data by the province filter out Alberta ppl       --Filtering out the NDP survey dat
AlbertaXsurveyXLib <- Liberal_predic.1[Liberal_predic.1$province == "Alberta",]
# make the model for predicting the results on Alberta,
glm_Lib_model<-glmer(Lib_pref ~ sex + education + (1|income_family) + age_group, data=AlbertaXsurveyXLi
#results
#summary(glm_Lib_model)

## now assign bins and make the prdicitons
# filter out for Alberta in the census data
albertaXcensus <- census_data.5[census_data.5$province == "Alberta",]    #### Can remove this line##
```

```r
# get the bin count corresdonidng to the model we make
albertaXcensus.1 <- albertaXcensus %>%
  count(sex, education,  income_family, age_group)        #### Can remove this line##

# make the logg odds for the regression model
X <- predict(glm_Lib_model, albertaXcensus.1)
#view(X)
# appending the log odds to the  NDPxAlberta
albertaXcensus.1$Lib_odds <- X
# getting the probability for each group
albertaXcensus.1 <- albertaXcensus.1 %>% mutate(prob_Lib = est_p(Lib_odds))

## post-strart for the albertaX_Consv

## sum of N                                --- add this in the model---
N <- sum(albertaXcensus.1$n)
# prediction for NDPxAlberta
Lib_Vote_percen_Alberta <- albertaXcensus.1 %>% summarise(Lib_est_Alberta = (sum(n*prob_Lib)/N))
Lib_Vote_percen_Alberta
```

```
## # A tibble: 1 x 1
##   Lib_est_Alberta
##             <dbl>
## 1           0.125
```

BASED on the result we can see that Majority of the people in Alberta ….

#British Columbia ##NDPxBritish Columbia

```r
# getting the NDP survey data to make the models
NDP_predic.2 <- survey_data.2 %>%
  mutate(NDP_pref = case_when(survey_data.2$political_pref == "NDP (New Democratic Party, New Democrats
                              survey_data.2$political_pref != "NDP (New Democratic Party, New Democrats

# group the survey data by the province filter out British Columbia ppl       --Filtering out the NDP s
BcXsurveyXNDP <- NDP_predic.2[NDP_predic.2$province == "British Columbia",]
# make the model for predicting the results on Alberta,
glm_NDP_model.1<-glmer(NDP_pref ~ sex + education + (1|income_family) + age_group, data=BcXsurveyXNDP, f
#results
#summary(glm_NDP_model.1)

## assign bins and get the counts for the census data

# filter out for British Columbia in the census data
BcXcensus <- census_data.5[census_data.5$province == "British Columbia",]
# get the bin count corresdonidng to the model we make
BcXcensus.1 <- albertaXcensus %>%
  count(sex, education,  income_family, age_group)

# make the logg odds for the regression model
X <- predict(glm_NDP_model.1, BcXcensus.1)
#view(X)
# appending the log odds to the  NDPxBritish Columbia
```

```
albertaXcensus.1$NDP_odds <- X
# getting the probability for each group
albertaXcensus.1 <- albertaXcensus.1 %>% mutate(prob_NDP = est_p(NDP_odds))

## post-strart for the albertaX_NDP

## sum of N                              --- add this in the model---
N <- sum(BcXcensus.1$n)
# prediction for NDPxAlberta
NDP_Vote_percen_BC <- albertaXcensus.1 %>% summarise(NDP_est_BC = (sum(n*prob_NDP)/N))
NDP_Vote_percen_BC
```

```
## # A tibble: 1 x 1
##   NDP_est_BC
##        <dbl>
## 1      0.231
```

##Conservatives x British Columbia

```
# getting the NDP survey data to make the models
Conservatives_predic.1 <- survey_data.2 %>%
  mutate(Consv_pref = case_when(survey_data.2$political_pref == "Conservatives (Tory, PCs, Conservative
                                survey_data.2$political_pref != "Conservatives (Tory, PCs, Conservative Pa

# filter out based on Alberta
# group the survey data by the provience filter out British Columbia ppl       --Filtering out the NDP
BcXsurveyXConsv <- Conservatives_predic.1[Conservatives_predic.1$province == "British Columbia",]
# make the model for predicting the results on British Columbia,
glm_Consv_model_Bc<-glmer(Consv_pref ~ sex + education + (1|income_family) + age_group, data=BcXsurveyX
#results
#summary(glm_Consv_model_Bc)

## now assign bins and make the predicitons
# filter out for Alberta in the census data
BcXcensus <- census_data.5[census_data.5$province == "British Columbia",]    #### Can remove this line#
# get the bin count corresdonidng to the model we make
BcXcensus.1 <- BcXcensus %>%
  count(sex, education,  income_family, age_group)        #### Can remove this line##

# make the logg odds for the regression model
X <- predict(glm_Consv_model_Bc, BcXcensus.1)
#view(X)
# appending the log odds to the  NDPxBritish Columbia
BcXcensus.1$Consv_odds <- X
# getting the probability for each group
BcXcensus.1 <- BcXcensus.1 %>% mutate(prob_Consv_Bc = est_p(Consv_odds))

## post-strart for the albertaX_Consv

## sum of N                              --- add this in the model---
N <- sum(BcXcensus.1$n)
# prediction for NDPxBritish Columbia
```

```
Consv_Vote_percen_Bc <- BcXcensus.1 %>% summarise(Consv_est_Bc = (sum(n*prob_Consv_Bc)/N))
Consv_Vote_percen_Bc
```

```
## # A tibble: 1 x 1
##   Consv_est_Bc
##          <dbl>
## 1        0.277
```

##LiberalXBc

```
## Making the model and the predictions

# getting the NDP survey data to make the models
Liberal_predic.1 <- survey_data.2 %>%
  mutate(Lib_pref = case_when(survey_data.2$political_pref == "Liberal (Grits)" ~ 1,
                              survey_data.2$political_pref != "Liberal (Grits))" ~ 0))

# filter out based on BC
# group the survey data by the province filter out Bc ppl       --Filtering out the Liberal survey data
BcXsurveyXLib <- Liberal_predic.1[Liberal_predic.1$province == "British Columbia",]
# make the model for predicting the results on Alberta,
glm_Lib_model_Bc <-glmer(Lib_pref ~ sex + education + (1|income_family) + age_group, data=BcXsurveyXLib
#summary(glm_Lib_model_Bc)

## now assign bins and make the prdicitons
# filter out for Bc in the census data
BcXcensus <- census_data.5[census_data.5$province == "British Columbia",]    #### Can remove this line#
# get the bin count corresdonidng to the model we make
BcXcensus.1 <- BcXcensus %>%
  count(sex, education,  income_family, age_group)        #### Can remove this line##

# make the logg odds for the regression model
X <- predict(glm_Lib_model_Bc, BcXcensus.1)
#view(X)
# appending the log odds to the  NDPxBc
BcXcensus.1$Lib_odds <- X
# getting the probability for each group
BcXcensus.1 <- BcXcensus.1 %>% mutate(prob_Lib = est_p(Lib_odds))

## post-strart for the BcX_Consv

## sum of N                              --- add this in the model---
N <- sum(BcXcensus.1$n)
# prediction for NDPxBc
Lib_Vote_percen_Bc <- BcXcensus.1 %>% summarise(Lib_est_Bc = (sum(n*prob_Lib)/N))
Lib_Vote_percen_Bc
```

```
## # A tibble: 1 x 1
##   Lib_est_Bc
##        <dbl>
## 1      0.267
```

#Ontario ##NDPxOntario

```r
# getting the NDP survey data to make the models
NDP_predic.1 <- survey_data.2 %>%
  mutate(NDP_pref = case_when(survey_data.2$political_pref == "NDP (New Democratic Party, New Democrats
                              survey_data.2$political_pref != "NDP (New Democratic Party, New Democrats

# group the survey data by the provience filter out Ontario ppl       --Filtering out the NDP survey da
OntarioXsurveyXNDP <- NDP_predic.1[NDP_predic.1$province == "Ontario",]
# make the model for predicting the results on Ontario,
glm_NDP_model.2<-glmer(NDP_pref ~ sex + education + (1|income_family) + age_group, data=OntarioXsurveyX
#results
#summary(glm_NDP_model.2)

## assign bins and get the counts for the census data

# filter out for Ontario in the census data
ontarioXcensus <- census_data.5[census_data.5$province =="Ontario",]
# get the bin count corresdonidng to the model we make
ontarioXcensus.1 <- ontarioXcensus %>%
  count(sex, education,  income_family, age_group)

# make the logg odds for the regression model
X <- predict(glm_NDP_model.2, ontarioXcensus.1)
#view(X)
# appending the log odds to the  NDPxOntario
ontarioXcensus.1$NDP_odds <- X
# getting the probability for each group
ontarioXcensus.1 <- ontarioXcensus.1 %>% mutate(prob_NDP_Ontario = est_p(NDP_odds))

## post-strart for the albertaX_NDP

## sum of N                              --- add this in the model---
N <- sum(ontarioXcensus.1$n)
# prediction for NDPxOntario
NDP_Vote_percen_Ontario <- ontarioXcensus.1 %>% summarise(NDP_est_Ontario = (sum(n*prob_NDP_Ontario)/N)
NDP_Vote_percen_Ontario
```

```
## # A tibble: 1 x 1
##   NDP_est_Ontario
##             <dbl>
## 1           0.148
```

##Conservatives x Ontario

```r
# getting the NDP survey data to make the models
Conservatives_predic.1 <- survey_data.2 %>%
  mutate(Consv_pref = case_when(survey_data.2$political_pref == "Conservatives (Tory, PCs, Conservative
                              survey_data.2$political_pref != "Conservatives (Tory, PCs, Conservative Pa

# filter out based on Ontario
# group the survey data by the provience filter out Ontario ppl       --Filtering out the NDP survey da
OnXsurveyXConsv <- Conservatives_predic.1[Conservatives_predic.1$province == "Ontario",]
# make the model for predicting the results on Ontario,
glm_Consv_model_On<-glmer(Consv_pref ~ sex + education + (1|income_family) + age_group, data=OnXsurveyX
```

```r
#results
#summary(glm_Consv_model_On)

## now assign bins and make the predicitons
# filter out for Ontario in the census data
OnXcensus <- census_data.5[census_data.5$province == "Ontario",]    #### Can remove this line##
# get the bin count corresdonidng to the model we make
OnXcensus.1 <- OnXcensus %>%
  count(sex, education,  income_family, age_group)        #### Can remove this line##

# make the logg odds for the regression model
X <- predict(glm_Consv_model_Bc, OnXcensus.1)
#view(X)
# appending the log odds to the  ConsxOntario
OnXcensus.1$Consv_odds <- X
# getting the probability for each group
OnXcensus.1 <- OnXcensus.1 %>% mutate(prob_Consv_On = est_p(Consv_odds))

## post-strart for the OntarioX_Consv

## sum of N                              --- add this in the model---
N <- sum(OnXcensus.1$n)
# prediction for ConsxOntario
Cons_Vote_percen_Bc <- OnXcensus.1 %>% summarise(Consv_est_On = (sum(n*prob_Consv_On)/N))
Cons_Vote_percen_Bc
```

```
## # A tibble: 1 x 1
##   Consv_est_On
##          <dbl>
## 1        0.276
```

##Liberal X Ontario

```r
## Making the model and the predictions

# getting the NDP survey data to make the models
Liberal_predic.1 <- survey_data.2 %>%
  mutate(Lib_pref = case_when(survey_data.2$political_pref == "Liberal (Grits)" ~ 1,
                              survey_data.2$political_pref != "Liberal (Grits))" ~ 0))

# filter out based on Ontario
# group the survey data by the province filter out Bc ppl       --Filtering out the Liberal survey data
OnXsurveyXLib <- Liberal_predic.1[Liberal_predic.1$province == "Ontario",]
# make the model for predicting the results on Ontario,
glm_Lib_model_On <-glmer(Lib_pref ~ sex + education + (1|income_family) + age_group, data=OnXsurveyXLib
#summary(glm_Lib_model_On)

## now assign bins and make the prdicitons
# filter out for Ontario in the census data
OnXcensus <- census_data.5[census_data.5$province == "Ontario",]    #### Can remove this line##
# get the bin count corresdonidng to the model we make
OnXcensus.1 <- OnXcensus %>%
  count(sex, education,  income_family, age_group)        #### Can remove this line##
```

```r
# make the logg odds for the regression model
X <- predict(glm_Lib_model_Bc, OnXcensus.1)
#view(X)
# appending the log odds to the  NDPxOntario
OnXcensus.1$Lib_odds <- X
# getting the probability for each group
OnXcensus.1 <- OnXcensus.1 %>% mutate(prob_Lib = est_p(Lib_odds))

## post-strart for the OntarioX_Consv

## sum of N                              --- add this in the model---
N <- sum(OnXcensus.1$n)
# prediction for NDPxOntario
Lib_Vote_percen_On <- OnXcensus.1 %>% summarise(Lib_est_On = (sum(n*prob_Lib)/N))
Lib_Vote_percen_On
```

```
## # A tibble: 1 x 1
##   Lib_est_On
##        <dbl>
## 1      0.268
```

#Manitoba ##NDPxManitoba

```r
# getting the NDP survey data to make the models
NDP_predic.1 <- survey_data.2 %>%
  mutate(NDP_pref = case_when(survey_data.2$political_pref == "NDP (New Democratic Party, New Democrats
                              survey_data.2$political_pref != "NDP (New Democratic Party, New Democrats

# group the survey data by the provience filter out Manitoba ppl       --Filtering out the NDP survey d
MnXsurveyXNDP <- NDP_predic.1[NDP_predic.1$province == "Manitoba",]
# make the model for predicting the results on Manitoba,
glm_NDP_model.Mn<-glmer(NDP_pref ~ sex + education + (1|income_family) + age_group, data=MnXsurveyXNDP,
#results
#summary(glm_NDP_model.Mn)

## assign bins and get the counts for the census data

# filter out for Manitoba in the census data
MnXcensus <- census_data.5[census_data.5$province =="Manitoba",]
# get the bin count corresponding to the model we make
MnXcensus.1 <- MnXcensus %>%
  count(sex, education,  income_family, age_group)

# make the logg odds for the regression model
X <- predict(glm_NDP_model.Mn, MnXcensus.1)
#view(X)
# appending the log odds to the  NDPxManitoba
MnXcensus.1$NDP_odds <- X
# getting the probability for each group
MnXcensus.1 <- MnXcensus.1 %>% mutate(prob_NDP_Manitoba = est_p(NDP_odds))

## post-strart for the ManitobaX_NDP
```

```r
## sum of N                              --- add this in the model---
N <- sum(MnXcensus.1$n)
# prediction for NDPxManitoba
NDP_Vote_percen_Manitoba <- MnXcensus.1 %>% summarise(NDP_est_Manitoba = (sum(n*prob_NDP_Manitoba)/N))
NDP_Vote_percen_Manitoba
```

```
## # A tibble: 1 x 1
##   NDP_est_Manitoba
##              <dbl>
## 1            0.120
```

##Conservatives x Manitoba

```r
# getting the NDP survey data to make the models
Conservatives_predic.1 <- survey_data.2 %>%
  mutate(Consv_pref = case_when(survey_data.2$political_pref == "Conservatives (Tory, PCs, Conservative
                                survey_data.2$political_pref != "Conservatives (Tory, PCs, Conservative Pa

# filter out based on Manitoba
# group the survey data by the provience filter out Manitoba ppl       --Filtering out the NDP survey da
MnXsurveyXConsv <- Conservatives_predic.1[Conservatives_predic.1$province == "Manitoba",]
# make the model for predicting the results on Manitoba,
glm_Consv_model_Mn<-glmer(Consv_pref ~ sex + education + (1|income_family) + age_group, data=MnXsurveyX
#results
#summary(glm_Consv_model_Mn)

## now assign bins and make the predicitons
# filter out for Ontario in the census data
MnXcensus <- census_data.5[census_data.5$province == "Manitoba",]    #### Can remove this line##
# get the bin count corresdonidng to the model we make
MnXcensus.1 <- MnXcensus %>%
  count(sex, education,  income_family, age_group)        #### Can remove this line##

# make the logg odds for the regression model
X <- predict(glm_Consv_model_Bc, MnXcensus.1)
#view(X)
# appending the log odds to the  ConsxOntario
MnXcensus.1$Consv_odds <- X
# getting the probability for each group
MnXcensus.1 <- MnXcensus.1 %>% mutate(prob_Consv_Mn = est_p(Consv_odds))

## post-strart for the OntarioX_Consv

## sum of N                              --- add this in the model---
N <- sum(MnXcensus.1$n)
# prediction for ConsxOntario
Cons_Vote_percen_Mn <- MnXcensus.1 %>% summarise(Consv_est_Mn = (sum(n*prob_Consv_Mn)/N))
Cons_Vote_percen_Mn
```

```
## # A tibble: 1 x 1
##   Consv_est_Mn
##          <dbl>
## 1        0.311
```

##Liberal X Manitoba

```
## Making the model and the predictions

# getting the NDP survey data to make the models
Liberal_predic.1 <- survey_data.2 %>%
  mutate(Lib_pref = case_when(survey_data.2$political_pref == "Liberal (Grits)" ~ 1,
                              survey_data.2$political_pref != "Liberal (Grits))" ~ 0))

# filter out based on Manitoba
# group the survey data by the province filter out Manitoba ppl       --Filtering out the Liberal survey
MnXsurveyXLib <- Liberal_predic.1[Liberal_predic.1$province == "Manitoba",]
# make the model for predicting the results on Manitoba,
glm_Lib_model_Mn <-glmer(Lib_pref ~ sex + education + (1|income_family) + age_group, data=MnXsurveyXLib
#summary(glm_Lib_model_Mn)

## now assign bins and make the prdicitons
# filter out for Manitoba in the census data
MnXcensus <- census_data.5[census_data.5$province == "Manitoba",]    #### Can remove this line##
# get the bin count corresdonidng to the model we make
MnXcensus.1 <- MnXcensus %>%
  count(sex, education,  income_family, age_group)        #### Can remove this line##

# make the logg odds for the regression model
X <- predict(glm_Lib_model_Mn, MnXcensus.1)
#view(X)
# appending the log odds to the  NDPxManitoba
MnXcensus.1$Lib_odds <- X
# getting the probability for each group
MnXcensus.1 <- MnXcensus.1 %>% mutate(prob_Lib = est_p(Lib_odds))

## post-strart for the ManitobaX_Consv

## sum of N                              --- add this in the model---
N <- sum(MnXcensus.1$n)
# prediction for NDPxManitoba
Lib_Vote_percen_Mn <- MnXcensus.1 %>% summarise(Lib_est_Mn = (sum(n*prob_Lib)/N))
Lib_Vote_percen_Mn
```

```
## # A tibble: 1 x 1
##   Lib_est_Mn
##        <dbl>
## 1      0.183
```

#Quebec Limitation—No data on (Coalition Avenir Québec or the CAQ) and (Quebec Liberal Party) <- biggest parties in Quebec ##Bloc Quebecois X Quebec

```
## Making the model and the predictions

# getting the NDP survey data to make the models
Quebecois_predic <- survey_data.2 %>%
  mutate(Qb_pref = case_when(survey_data.2$political_pref == "Bloc Québécois (BQ, PQ, Bloc, Parti Québé
                              survey_data.2$political_pref != "Bloc Québécois (BQ, PQ, Bloc, Parti Québe
```

```r
# filter out based on Quebec
# group the survey data by the province filter out Quebec ppl       --Filtering out the Liberal survey
QuXsurveyXQb <- Quebecois_predic[Quebecois_predic$province == "Quebec",]
# make the model for predicting the results on Quebec,
glm_Qb_model_Qb <-glmer(Qb_pref ~ sex + education + (1|income_family) + age_group, data=QuXsurveyXQb, fa
#summary(glm_Qb_model_Qb)

## now assign bins and make the prdicitons
# filter out for Quebec in the census data
QbXcensus <- census_data.5[census_data.5$province == "Quebec",]    #### Can remove this line##
# get the bin count corresdonidng to the model we make
QbXcensus.1 <- QbXcensus %>%
  count(sex, education,  income_family, age_group)        #### Can remove this line##

# make the logg odds for the regression model
X <- predict(glm_Qb_model_Qb, QbXcensus.1)
#view(X)
# appending the log odds to the  NDPxQuebec
QbXcensus.1$Qb_odds <- X
# getting the probability for each group
QbXcensus.1 <- QbXcensus.1 %>% mutate(prob_Qb = est_p(Qb_odds))

## post-strart for the QuebecX_Qb

## sum of N                              --- add this in the model---
N <- sum(QbXcensus.1$n)
# prediction for NDPxManitoba
Qb_Vote_percen_Qb <- QbXcensus.1 %>% summarise(Qb_est_Qb = (sum(n*prob_Qb)/N))
Qb_Vote_percen_Qb
```

```
## # A tibble: 1 x 1
##   Qb_est_Qb
##       <dbl>
## 1     0.245
```

##Conservatives X Quebec

```r
# getting the Conservative survey data to make the models
Conservatives_predic.1 <- survey_data.2 %>%
  mutate(Consv_pref = case_when(survey_data.2$political_pref == "Conservatives (Tory, PCs, Conservative
                                survey_data.2$political_pref != "Conservatives (Tory, PCs, Conservative Pa

# filter out based on Quebec
# group the survey data by the provience filter out Quebec ppl       --Filtering out the Conservative s
QbXsurveyXConsv <- Conservatives_predic.1[Conservatives_predic.1$province == "Quebec",]
# make the model for predicting the results on Quebec,
glm_Consv_model_Qb<-glmer(Consv_pref ~ sex + education + (1|income_family) + age_group, data=QbXsurveyX
#results
#summary(glm_Consv_model_Qb)

## now assign bins and make the predicitons
# filter out for Ontario in the census data
```

```r
QbXcensus <- census_data.5[census_data.5$province == "Quebec",]    #### Can remove this line##
# get the bin count corresdonidng to the model we make
QbXcensus.1 <- QbXcensus %>%
  count(sex, education,  income_family, age_group)        #### Can remove this line##

# make the logg odds for the regression model
X <- predict(glm_Consv_model_Qb, QbXcensus.1)
#view(X)
# appending the log odds
QbXcensus.1$Consv_odds <- X
# getting the probability for each group
QbXcensus.1 <- QbXcensus.1 %>% mutate(prob_Consv_Qb = est_p(Consv_odds))

## post-strart for the QuebecX_Consv

## sum of N                              --- add this in the model---
N <- sum(QbXcensus.1$n)
# prediction for ConsxQuebec
Cons_Vote_percen_Qb <- QbXcensus.1 %>% summarise(Consv_est_Qb = (sum(n*prob_Consv_Qb)/N))
Cons_Vote_percen_Qb
```

```
## # A tibble: 1 x 1
##   Consv_est_Qb
##          <dbl>
## 1        0.170
```

##Liberal X Quebec

```r
## Making the model and the predictions

# getting the NDP survey data to make the models
Liberal_predic.1 <- survey_data.2 %>%
  mutate(Lib_pref = case_when(survey_data.2$political_pref == "Liberal (Grits)" ~ 1,
                              survey_data.2$political_pref != "Liberal (Grits))" ~ 0))

# filter out based on Quebec
# group the survey data by the province filter out Quebec ppl       --Filtering out the Liberal survey
QbXsurveyXLib <- Liberal_predic.1[Liberal_predic.1$province == "Quebec",]
# make the model for predicting the results on Quebec,
glm_Lib_model_Qb <-glmer(Lib_pref ~ sex + education + (1|income_family) + age_group, data=QbXsurveyXLib
#summary(glm_Lib_model_Qb)

## now assign bins and make the predictions
# filter out for Quebec in the census data
QbXcensus <- census_data.5[census_data.5$province == "Quebec",]    #### Can remove this line##
# get the bin count corresdonidng to the model we make
QbXcensus.1 <- QbXcensus %>%
  count(sex, education,  income_family, age_group)        #### Can remove this line##

# make the logg odds for the regression model
X <- predict(glm_Lib_model_Qb, QbXcensus.1)
#view(X)
# appending the log odds to the  NDPxManitoba
```

```
QbXcensus.1$Lib_odds <- X
# getting the probability for each group
QbXcensus.1 <- QbXcensus.1 %>% mutate(prob_Lib = est_p(Lib_odds))

## post-strart for the Quebec_X_Liberal

## sum of N                              --- add this in the model---
N <- sum(QbXcensus.1$n)
# prediction for LiberalxQuebec
Lib_Vote_percen_Qb <- QbXcensus.1 %>% summarise(Lib_est_Qb = (sum(n*prob_Lib)/N))
Lib_Vote_percen_Qb
```

```
## # A tibble: 1 x 1
##   Lib_est_Qb
##        <dbl>
## 1      0.396
```

#New Brunswick LIMITATION no one BW age 18-24 in survey ##New BrunswickXLiberal

```
## Making the model and the predictions

# getting the NDP survey data to make the models
Liberal_predic.1 <- survey_data.2 %>%
  mutate(Lib_pref = case_when(survey_data.2$political_pref == "Liberal (Grits)" ~ 1,
                              survey_data.2$political_pref != "Liberal (Grits))" ~ 0))

# filter out based on New Brunswick
# group the survey data by the province filter out New Brunswick ppl       --Filtering out the Liberal
NbXsurveyXLib <- Liberal_predic.1[Liberal_predic.1$province == "New Brunswick",]

# make the model for predicting the results on New Brunswick,
glm_Lib_model_Nb <-glmer(Lib_pref ~ sex + education + (1|income_family) + age_group, data=NbXsurveyXLib
#summary(glm_Lib_model_Nb)

## now assign bins and make the predictions
# filter out for New Brunswick in the census data
NbXcensus <- census_data.5[census_data.5$province == "New Brunswick",]    #### Can remove this line##
## removing the bin of ppl less than 18 because not in the survey bin
NbXcensus <- NbXcensus[NbXcensus$age_group != "18 to 24 years",]
# get the bin count corresdonidng to the model we make
NbXcensus.1 <- NbXcensus %>%
  count(sex, education,  income_family, age_group)        #### Can remove this line##

# make the logg odds for the regression model
X <- predict(glm_Lib_model_Nb, NbXcensus.1)
#view(X)
# appending the log odds to the  NDPxManitoba
NbXcensus.1$Lib_odds <- X
# getting the probability for each group
NbXcensus.1 <- NbXcensus.1 %>% mutate(prob_Lib = est_p(Lib_odds))

## post-strart for the New Brunswick_X_Liberal
```

```r
## sum of N                                --- add this in the model---
N <- sum(NbXcensus.1$n)
# prediction for LiberalxNb
Lib_Vote_percen_Nb <- NbXcensus.1 %>% summarise(Lib_est_Nb = (sum(n*prob_Lib)/N))
Lib_Vote_percen_Nb
```

```
## # A tibble: 1 x 1
##   Lib_est_Nb
##        <dbl>
## 1      0.230
```

##New BrunswickXConservative

```r
# getting the Conservative survey data to make the models
Conservatives_predic.1 <- survey_data.2 %>%
  mutate(Consv_pref = case_when(survey_data.2$political_pref == "Conservatives (Tory, PCs, Conservative
                                survey_data.2$political_pref != "Conservatives (Tory, PCs, Conservative Pa

# filter out based on New Brunswick
# group the survey data by the provience filter out New Brunswick ppl       --Filtering out the Conserv
NbXsurveyXConsv <- Conservatives_predic.1[Conservatives_predic.1$province == "New Brunswick",]
# make the model for predicting the results on New Brunswick,
glm_Consv_model_Nb<-glmer(Consv_pref ~ sex + education + (1|income_family) + age_group, data=NbXsurveyX
#results
#summary(glm_Consv_model_Nb)

## now assign bins and make the predicitons
# filter out for Ontario in the census data
NbXcensus <- census_data.5[census_data.5$province == "New Brunswick",]    #### Can remove this line##
# get the bin count corresdonidng to the model we make
## removing the bin of ppl less than 18 because not in the survey bin
NbXcensus <- NbXcensus[NbXcensus$age_group != "18 to 24 years",]
NbXcensus.1 <- NbXcensus %>%
  count(sex, education,  income_family, age_group)        #### Can remove this line##

# make the logg odds for the regression model
X <- predict(glm_Consv_model_Nb, NbXcensus.1)
#view(X)
# appending the log odds
NbXcensus.1$Consv_odds <- X
# getting the probability for each group
NbXcensus.1 <- NbXcensus.1 %>% mutate(prob_Consv_Nb = est_p(Consv_odds))

## post-strart for the New BrunswickX_Consv

## sum of N                                --- add this in the model---
N <- sum(NbXcensus.1$n)
# prediction for ConsxNew Brunswick
Cons_Vote_percen_Nb <- NbXcensus.1 %>% summarise(Consv_est_Nb = (sum(n*prob_Consv_Nb)/N))
Cons_Vote_percen_Nb
```

```
## # A tibble: 1 x 1
```

```
##   Consv_est_Nb
##          <dbl>
## 1        0.397
```

##New BrunswickXGreen

```
# getting the Conservative survey data to make the models
Green_predic.1 <- survey_data.2 %>%
  mutate(Green_pref = case_when(survey_data.2$political_pref == "Green Party (Greens)" ~ 1,
                                survey_data.2$political_pref != "Green Party (Greens)" ~ 0))

# filter out based on New Brunswick
# group the survey data by the provience filter out New Brunswick ppl       --Filtering out the Green s
NbXsurveyXGreen <- Green_predic.1[Green_predic.1$province == "New Brunswick",]
# make the model for predicting the results on New Brunswick,
glm_Green_model_Nb<-glmer(Green_pref ~ sex + education + (1|income_family) + age_group, data=NbXsurveyX
#results
#summary(glm_Green_model_Nb)

## now assign bins and make the predicitons
# filter out for New Brunsw in the census data
NbXcensus <- census_data.5[census_data.5$province == "New Brunswick",]    #### Can remove this line##
# get the bin count corresdonidng to the model we make
## removing the bin of ppl less than 18 because not in the survey bin
NbXcensus <- NbXcensus[NbXcensus$age_group != "18 to 24 years",]
NbXcensus.1 <- NbXcensus %>%
  count(sex, education,  income_family, age_group)        #### Can remove this line##

# make the logg odds for the regression model
X <- predict(glm_Green_model_Nb, NbXcensus.1)
#view(X)
# appending the log odds
NbXcensus.1$Green_odds <- X
# getting the probability for each group
NbXcensus.1 <- NbXcensus.1 %>% mutate(prob_Green_Nb = est_p(Green_odds))

## post-strart for the New BrunswickX_Consv

## sum of N                              --- add this in the model---
N <- sum(NbXcensus.1$n)
# prediction for ConsxNew Brunswick
Green_Vote_percen_Nb <- NbXcensus.1 %>% summarise(Green_est_Nb = (sum(n*prob_Green_Nb)/N))
Green_Vote_percen_Nb
```

```
## # A tibble: 1 x 1
##   Green_est_Nb
##          <dbl>
## 1        0.255
```

#Saskatchewan ##Cons

```r
# getting the NDP survey data to make the models
Conservatives_predic.1 <- survey_data.2 %>%
  mutate(Consv_pref = case_when(survey_data.2$political_pref == "Conservatives (Tory, PCs, Conservative
                                survey_data.2$political_pref != "Conservatives (Tory, PCs, Conservative Pa

# filter out based on Saskatchewan
# group the survey data by the provience filter out Saskatchewan ppl       --Filtering out the NDP surv
SaXsurveyXConsv <- Conservatives_predic.1[Conservatives_predic.1$province == "Saskatchewan",]
# make the model for predicting the results on Saskatchewan,
glm_Consv_model_Sa<-glmer(Consv_pref ~ sex + education + (1|income_family) + age_group, data=SaXsurveyX
#results
#summary(glm_Consv_model_Sa)

## now assign bins and make the predicitons
# filter out for Saskatchwan in the census data
SaXcensus <- census_data.5[census_data.5$province == "Saskatchewan",]    #### Can remove this line##
# get the bin count corresdonidng to the model we make
SaXcensus.1 <- SaXcensus %>%
  count(sex, education,  income_family, age_group)        #### Can remove this line##

# make the logg odds for the regression model
X <- predict(glm_Consv_model_Sa, SaXcensus.1)
#view(X)
# appending the log odds to the  NDPxSaskatchewan
SaXcensus.1$Consv_odds <- X
# getting the probability for each group
SaXcensus.1 <- SaXcensus.1 %>% mutate(prob_Consv_Sa = est_p(Consv_odds))

## post-strart for the SaskatchewanX_Consv

## sum of N                              --- add this in the model---
N <- sum(SaXcensus.1$n)
# prediction for NDPxSaskatchewan
Consv_Vote_percen_Sa <- SaXcensus.1 %>% summarise(Consv_est_Sa = (sum(n*prob_Consv_Sa)/N))
Consv_Vote_percen_Sa
```

```
## # A tibble: 1 x 1
##   Consv_est_Sa
##          <dbl>
## 1        0.618
```

##Liberal

```r
## Making the model and the predictions

# getting the NDP survey data to make the models
Liberal_predic.1 <- survey_data.2 %>%
  mutate(Lib_pref = case_when(survey_data.2$political_pref == "Liberal (Grits)" ~ 1,
                              survey_data.2$political_pref != "Liberal (Grits))" ~ 0))

# filter out based on Saskatchewan
# group the survey data by the province filter out Saskatchewan ppl       --Filtering out the Liberal s
SaXsurveyXLib <- Liberal_predic.1[Liberal_predic.1$province == "Saskatchewan",]
```

```r
# make the model for predicting the results on Saskatchewan,
glm_Lib_model_Sa <-glmer(Lib_pref ~ sex + education + (1|income_family) + age_group, data=SaXsurveyXLib
#summary(glm_Lib_model_Sa)

## now assign bins and make the prdicitons
# filter out for Saskatchewan in the census data
SaXcensus <- census_data.5[census_data.5$province == "Saskatchewan",]    #### Can remove this line##
# get the bin count corresdonidng to the model we make
SaXcensus.1 <- SaXcensus %>%
  count(sex, education,  income_family, age_group)        #### Can remove this line##

# make the logg odds for the regression model
X <- predict(glm_Lib_model_Sa, SaXcensus.1)
#view(X)
# appending the log odds to the  NDPxSaskatchewan
SaXcensus.1$Lib_odds <- X
# getting the probability for each group
SaXcensus.1 <- SaXcensus.1 %>% mutate(prob_Lib = est_p(Lib_odds))

## post-strart for the SaskatchewanX_Consv

## sum of N                              --- add this in the model---
N <- sum(SaXcensus.1$n)
# prediction for NDPxSaskatchewan
Lib_Vote_percen_Sa <- SaXcensus.1 %>% summarise(Lib_est_Sa = (sum(n*prob_Lib)/N))
Lib_Vote_percen_Sa
```

```
## # A tibble: 1 x 1
##   Lib_est_Sa
##        <dbl>
## 1      0.135
```

##NDP

```r
# getting the NDP survey data to make the models
NDP_predic.1 <- survey_data.2 %>%
  mutate(NDP_pref = case_when(survey_data.2$political_pref == "NDP (New Democratic Party, New Democrats
                              survey_data.2$political_pref != "NDP (New Democratic Party, New Democrats

# group the survey data by the provience filter out Saskatchewan ppl       --Filtering out the NDP surv
SaXsurveyXNDP <- NDP_predic.1[NDP_predic.1$province == "Saskatchewan",]
# make the model for predicting the results on Saskatchewan,
glm_NDP_model.Sa<-glmer(NDP_pref ~ sex + education + (1|income_family) + age_group, data=SaXsurveyXNDP,
#results
#summary(glm_NDP_model.Sa)

## assign bins and get the counts for the census data

# filter out for Saskatchewan in the census data
SaXcensus <- census_data.5[census_data.5$province =="Saskatchewan",]
# get the bin count corresponding to the model we make
SaXcensus.1 <- SaXcensus %>%
  count(sex, education,  income_family, age_group)
```

```r
# make the logg odds for the regression model
X <- predict(glm_NDP_model.Sa, SaXcensus.1)
#view(X)
# appending the log odds to the  NDPxSaskatchewan
SaXcensus.1$NDP_odds <- X
# getting the probability for each group
SaXcensus.1 <- SaXcensus.1 %>% mutate(prob_NDP_Manitoba = est_p(NDP_odds))

## post-strart for the SaskatchewanX_NDP

## sum of N                              --- add this in the model---
N <- sum(SaXcensus.1$n)
# prediction for NDPxSaskatchewan
NDP_Vote_percen_Saskatchewan <- SaXcensus.1 %>% summarise(NDP_est_Saskatchewan = (sum(n*prob_NDP_Manitol
NDP_Vote_percen_Saskatchewan
```

```
## # A tibble: 1 x 1
##   NDP_est_Saskatchewan
##                  <dbl>
## 1                0.102
```

#Nova Scotia ##Cons

```r
# getting the NDP survey data to make the models
Conservatives_predic.1 <- survey_data.2 %>%
  mutate(Consv_pref = case_when(survey_data.2$political_pref == "Conservatives (Tory, PCs, Conservative
                                survey_data.2$political_pref != "Conservatives (Tory, PCs, Conservative Pa

# filter out based on Nova Scotia
# group the survey data by the provience filter out Nova Scotia ppl       --Filtering out the NDP survey
NsXsurveyXConsv <- Conservatives_predic.1[Conservatives_predic.1$province == "Nova Scotia",]
# make the model for predicting the results on Nova Scotia,
glm_Consv_model_Ns<-glmer(Consv_pref ~ sex + education + (1|income_family) + age_group, data=NsXsurveyX
#results
#summary(glm_Consv_model_Ns)

## now assign bins and make the predicitons
# filter out for Nova Scotia in the census data
NsXcensus <- census_data.5[census_data.5$province == "Nova Scotia",]    #### Can remove this line##
# get the bin count corresdonidng to the model we make
NsXcensus.1 <- NsXcensus %>%
  count(sex, education,  income_family, age_group)        #### Can remove this line##

# make the logg odds for the regression model
X <- predict(glm_Consv_model_Ns, NsXcensus.1)
#view(X)
# appending the log odds to the  NDPxNova Scotia
NsXcensus.1$Consv_odds <- X
# getting the probability for each group
NsXcensus.1 <- NsXcensus.1 %>% mutate(prob_Consv_Sa = est_p(Consv_odds))

## post-strart for the Nova ScotiaX_Consv
```

```
## sum of N                              --- add this in the model---
N <- sum(SaXcensus.1$n)
# prediction for NDPxNova Scotia
Consv_Vote_percen_Ns <- NsXcensus.1 %>% summarise(Consv_est_Ns = (sum(n*prob_Consv_Sa)/N))
Consv_Vote_percen_Ns
```

```
## # A tibble: 1 x 1
##   Consv_est_Ns
##          <dbl>
## 1        0.388
```

##Liberal

```
## Making the model and the predictions

# getting the NDP survey data to make the models
Liberal_predic.1 <- survey_data.2 %>%
  mutate(Lib_pref = case_when(survey_data.2$political_pref == "Liberal (Grits)" ~ 1,
                              survey_data.2$political_pref != "Liberal (Grits))" ~ 0))

# filter out based on Nova Scotia
# group the survey data by the province filter out Nova Scotia ppl       --Filtering out the Liberal su
NsXsurveyXLib <- Liberal_predic.1[Liberal_predic.1$province == "Nova Scotia",]
# make the model for predicting the results on Nova Scotia,
glm_Lib_model_Ns <-glmer(Lib_pref ~ sex + education + (1|income_family) + age_group, data=NsXsurveyXLib
#summary(glm_Lib_model_Ns)

## now assign bins and make the prdicitons
# filter out for Nova Scotia in the census data
NsXcensus <- census_data.5[census_data.5$province == "Nova Scotia",]    #### Can remove this line##
# get the bin count corresdonidng to the model we make
NsXcensus.1 <- NsXcensus %>%
  count(sex, education,  income_family, age_group)        #### Can remove this line##

# make the logg odds for the regression model
X <- predict(glm_Lib_model_Ns, NsXcensus.1)
#view(X)
# appending the log odds to the  LiberalxNova Scotia
NsXcensus.1$Lib_odds <- X
# getting the probability for each group
NsXcensus.1 <- NsXcensus.1 %>% mutate(prob_Lib = est_p(Lib_odds))

## post-strart for the Nova ScotiaX_Consv

## sum of N                              --- add this in the model---
N <- sum(NsXcensus.1$n)
# prediction for NDPxSaskatchewan
Lib_Vote_percen_Ns <- NsXcensus.1 %>% summarise(Lib_est_Ns = (sum(n*prob_Lib)/N))
Lib_Vote_percen_Ns
```

```
## # A tibble: 1 x 1
##   Lib_est_Ns
```

```
##        <dbl>
## 1      0.348
```

##NDP

```
# getting the NDP survey data to make the models
NDP_predic.1 <- survey_data.2 %>%
  mutate(NDP_pref = case_when(survey_data.2$political_pref == "NDP (New Democratic Party, New Democrats
                              survey_data.2$political_pref != "NDP (New Democratic Party, New Democrats

# group the survey data by the provience filter out Nova Scotia ppl       --Filtering out the NDP surve
NsXsurveyXNDP <- NDP_predic.1[NDP_predic.1$province == "Nova Scotia",]
# make the model for predicting the results on Nova Scotia,
glm_NDP_model.Ns<-glmer(NDP_pref ~ sex + education + (1|income_family) + age_group, data=NsXsurveyXNDP,
#results
#summary(glm_NDP_model.Ns)

## assign bins and get the counts for the census data

# filter out for Nova Scotia in the census data
NsXcensus <- census_data.5[census_data.5$province =="Nova Scotia",]
# get the bin count corresponding to the model we make
NsXcensus.1 <- NsXcensus %>%
  count(sex, education,  income_family, age_group)

# make the logg odds for the regression model
X <- predict(glm_NDP_model.Ns, NsXcensus.1)
#view(X)
# appending the log odds to the  NDPxNova Scotia
NsXcensus.1$NDP_odds <- X
# getting the probability for each group
NsXcensus.1 <- NsXcensus.1 %>% mutate(prob_NDP_Ns = est_p(NDP_odds))

## post-strart for the Nova ScotiaX_NDP

## sum of N                              --- add this in the model---
N <- sum(NsXcensus.1$n)
# prediction for NDPxNova Scotia
NDP_Vote_percen_Ns <- NsXcensus.1 %>% summarise(NDP_est_Ns = (sum(n*prob_NDP_Ns)/N))
NDP_Vote_percen_Ns
```

```
## # A tibble: 1 x 1
##   NDP_est_Ns
##        <dbl>
## 1      0.156
```

#Newfoundland and Labrador —-AGE GROUP 18-24 limitation ##NDP

```
# getting the NDP survey data to make the models
NDP_predic.1 <- survey_data.2 %>%
  mutate(NDP_pref = case_when(survey_data.2$political_pref == "NDP (New Democratic Party, New Democrats
                              survey_data.2$political_pref != "NDP (New Democratic Party, New Democrats
```

```r
# group the survey data by the provience filter out Newfoundland and Labrador ppl       --Filtering out
NLXsurveyXNDP <- NDP_predic.1[NDP_predic.1$province == "Newfoundland and Labrador",]
# make the model for predicting the results on Newfoundland and Labrador,
glm_NDP_model.NL<-glmer(NDP_pref ~ sex + education + (1|income_family) + age_group, data=NLXsurveyXNDP,
#results
#summary(glm_NDP_model.NL)

## assign bins and get the counts for the census data

# filter out for Newfoundland and Labrador in the census data
NLXcensus <- census_data.5[census_data.5$province =="Newfoundland and Labrador",]
## removing the bin of ppl less than 18 because not in the survey bin
NLXcensus <- NLXcensus[NLXcensus$age_group != "18 to 24 years",]
# get the bin count corresponding to the model we make
NLXcensus.1 <- NLXcensus %>%
  count(sex, education,  income_family, age_group)

# make the logg odds for the regression model
X <- predict(glm_NDP_model.NL, NLXcensus.1)
#view(X)
# appending the log odds to the  NDPxNewfoundland and Labrador
NLXcensus.1$NDP_odds <- X
# getting the probability for each group
NLXcensus.1 <- NLXcensus.1 %>% mutate(prob_NDP_NL = est_p(NDP_odds))

## post-strart for the Newfoundland and LabradorX_NDP

## sum of N                              --- add this in the model---
N <- sum(NLXcensus.1$n)
# prediction for NDPxNewfoundland and Labrador
NDP_Vote_percen_NL <- NLXcensus.1 %>% summarise(NDP_est_NL = (sum(n*prob_NDP_NL)/N))
NDP_Vote_percen_NL
```

```
## # A tibble: 1 x 1
##   NDP_est_NL
##        <dbl>
## 1      0.381
```

##Liberal

```r
## Making the model and the predictions

# getting the Liberal survey data to make the models
Liberal_predic.1 <- survey_data.2 %>%
  mutate(Lib_pref = case_when(survey_data.2$political_pref == "Liberal (Grits)" ~ 1,
                              survey_data.2$political_pref != "Liberal (Grits))" ~ 0))

# filter out based on Newfoundland and Labrador
# group the survey data by the province filter out Newfoundland and Labrador ppl       --Filtering out
NLXsurveyXLib <- Liberal_predic.1[Liberal_predic.1$province == "Newfoundland and Labrador",]
# make the model for predicting the results on Newfoundland and Labrador,
glm_Lib_model_NL <-glmer(Lib_pref ~ sex + education + (1|income_family) + age_group, data=NLXsurveyXLib
#summary(glm_Lib_model_NL)
```

```r
## now assign bins and make the prdicitons
# filter out for Newfoundland and Labrador in the census data
NLXcensus <- census_data.5[census_data.5$province == "Newfoundland and Labrador",]    #### Can remove t
## removing the bin of ppl less than 18 because not in the survey bin
NLXcensus <- NLXcensus[NLXcensus$age_group != "18 to 24 years",]
# get the bin count corresdonidng to the model we make
NLXcensus.1 <- NLXcensus %>%
  count(sex, education,  income_family, age_group)        #### Can remove this line##

# make the logg odds for the regression model
X <- predict(glm_Lib_model_NL, NLXcensus.1)
#view(X)
# appending the log odds to the  LiberalxNewfoundland and Labrador
NLXcensus.1$Lib_odds <- X
# getting the probability for each group
NLXcensus.1 <- NLXcensus.1 %>% mutate(prob_Lib = est_p(Lib_odds))

## post-strart for the Newfoundland and LabradorX_Consv

## sum of N                              --- add this in the model---
N <- sum(NsXcensus.1$n)
# prediction for NDPxNewfoundland and Labrador
Lib_Vote_percen_NL <- NLXcensus.1 %>% summarise(Lib_est_NL = (sum(n*prob_Lib)/N))
Lib_Vote_percen_NL
```

```
## # A tibble: 1 x 1
##   Lib_est_NL
##        <dbl>
## 1      0.337
```

##Conservatives

```r
# getting the Conservative survey data to make the models
Conservatives_predic.1 <- survey_data.2 %>%
  mutate(Consv_pref = case_when(survey_data.2$political_pref == "Conservatives (Tory, PCs, Conservative
                                survey_data.2$political_pref != "Conservatives (Tory, PCs, Conservative Pa

# filter out based on Newfoundland and Labrador
# group the survey data by the provience filter out Newfoundland and Labrador ppl       --Filtering out
NLXsurveyXConsv <- Conservatives_predic.1[Conservatives_predic.1$province == "Newfoundland and Labrador
# make the model for predicting the results on Newfoundland and Labrador,
glm_Consv_model_NL<-glmer(Consv_pref ~ sex + education + (1|income_family) + age_group, data=NLXsurveyX
#results
#summary(glm_Consv_model_NL)

## now assign bins and make the predicitons
# filter out for Newfoundland and Labrador in the census data
NLXcensus <- census_data.5[census_data.5$province == "Newfoundland and Labrador",]    #### Can remove t
# get the bin count corresdonidng to the model we make
## removing the bin of ppl less than 18 because not in the survey bin
NLXcensus <- NLXcensus[NLXcensus$age_group != "18 to 24 years",]
NLXcensus.1 <- NLXcensus %>%
```

```
  count(sex, education,  income_family, age_group)        #### Can remove this line##

# make the logg odds for the regression model
X <- predict(glm_Consv_model_NL, NLXcensus.1)
#view(X)
# appending the log odds
NLXcensus.1$Consv_odds <- X
# getting the probability for each group
NLXcensus.1 <- NLXcensus.1 %>% mutate(prob_Consv_NL = est_p(Consv_odds))

## post-strart for the Newfoundland and LabradorX_Consv

## sum of N                              --- add this in the model---
N <- sum(NLXcensus.1$n)
# prediction for ConsxNewfoundland and Labrador
Cons_Vote_percen_NL <- NLXcensus.1 %>% summarise(Consv_est_NL = (sum(n*prob_Consv_NL)/N))
# Cons_Vote_percen_NL
```

^Due to such a small percentage * n NaN = 0/N

#Grouping based on proviences

```
# make them in groups by province
alberta <- census_data.4 %>% group_by(province = "Alberta")
british_columbia <- census_data.4 %>% group_by(province = "British Columbia")
manitoba<- census_data.5 %>% group_by(province = "Manitoba")
new_brunswick<- census_data.5 %>% group_by(province = "New Brunswick")
newfoundland_labrador<- census_data.5 %>% group_by(province = "Newfoundland and Labrador")
nova_scotia<- census_data.5 %>% group_by(province = "Nova Scotia")
ontario <- census_data.5 %>% group_by(province = "Ontario")
prince_edward_island<- census_data.5 %>% group_by(province = "Prince Edward Island")
quebec<- census_data.5 %>% group_by(province = "Quebec")
saskatchewan<- census_data.5 %>% group_by(province = "Saskatchewan")
```

<Remember, you may want to use multiple datasets here, if you do end up using multiple data sets, or merging the data, be sure to describe this in the cleaning process and be sure to discuss important aspects of all the data that you used.>

<Include a description of the important variables.>

<Include a description of the numerical summaries. Remember you can use `r` to use inline R code.>

```
# Use this to create some plots. Should probably describe both the sample and population.
```

<Include a clear description of the plot(s). I would recommend one paragraph for each plot.>

Methods

<Include some text introducing the methodology, maybe restating the problem/goal of this analysis.>

Model Specifics

<I will (incorrectly) be using a linear regression model to model the proportion of voters who will vote for Donald Trump. This is a naive model. I will only be using age, which is recorded as a numeric variable, to model the probability of voting for Donald Trump. The simple linear regression model I am using is:>

$$y = \beta_0 + \beta_1 x_{age} + \epsilon$$

<Where y represents the …. β_0 represents….>

Post-Stratification

<In order to estimate the proportion of voters…..>

<To put math/LaTeX inline just use one set of dollar signs. Example: $\hat{y}^{PS}$ >

$$include.your.mathematical.model.here.if.you.have.some.math.to.show$$

All analysis for this report was programmed using `R version 4.0.2`.

Results

<Here you present your results. You may want to put them into a well formatted table. Be sure that there is some text describing the results.>

<Note: Alternatively you can use the `knitr::kable` function to create a well formatted table from your code. See here: https://rmarkdown.rstudio.com/lesson-7.html.>

<Remember you can use `r` to use inline R code.>

<Include an explanation/interpretation of the visualizations. Make sure to comment on the appropriateness of the assumptions/results.>

Conclusions

Drawbacks

```
**relevant fields not found in the su
```

<Here you should give a summary of the Hypotheses, Methods and Results>

<Highlight Key Results.>

<Talk about big picture.>

<Comment on any Weaknesses.>

<End with a concluding paragraph to wrap up the report.>

Bibliography

1. Grolemund, G. (2014, July 16) *Introduction to R Markdown.* RStudio. https://rmarkdown.rstudio.com/articles_intro.html. (Last Accessed: January 15, 2021)

2. Dekking, F. M., et al. (2005) *A Modern Introduction to Probability and Statistics: Understanding why and how.* Springer Science & Business Media.

3. Allaire, J.J., et. el. *References: Introduction to R Markdown.* RStudio. https://rmarkdown.rstudio.com/docs/. (Last Accessed: January 15, 2021)