GSS Analysis Report

Qiyu Huang& Yuhan Zhu

10/19/2020

Abstract

We are going to take a analysis on th Canadian General Social Survey. We have used logistic regression and linear regression to see relationships. Deetails are shown later in the discussion part.

Introduction

We want to find if people are older, will they have more kids, more relationships, and have early kids when they are young? We also want to find if the income level is related with gender, with total number of children, and with satisfaction of their lives. Will the family be richer if they have more kids or poorer is what we want to find out as well.

Data

install.packages(“janitor”) install.packages(“tidyverse”) install.packages(“readr”) install.packages(“dplyr”) library(janitor) library(tidyverse) library(readr) library(dplyr)

Data cleaning Part

importing raw data

raw_data <- read_csv(“C:/Users/12069/Desktop/heixiannv/AAGe4G0U.csv”) dict <- read_lines(“gss_dict.txt”, skip = 18) labels_raw <- read_file(“gss_labels.txt”)

set up dictionary

variable_descriptions <- as_tibble(dict) %>% filter(value!=“}”) %>% mutate(value = str_replace(value, ".+%[0-9].*f[ ]{2,}“,”“)) %>% mutate(value = str_remove_all(value,”"")) %>% rename(variable_description = value) %>% bind_cols(tibble(variable_name = colnames(raw_data)[-1]))

variable names and values

labels_raw_tibble <- as_tibble(str_split(labels_raw, “;”)[[1]]) %>% filter(row_number()!=1) %>% mutate(value = str_remove(value, “define”)) %>% mutate(value = str_replace(value, “[ ]{2,}”, “XXX”)) %>% mutate(splits =

str_split(value, “XXX”)) %>% rowwise() %>% mutate(variable_name = splits[1], cases = splits[2]) %>% mutate(cases = str_replace_all(cases, “”, "")) %>% select(variable_name, cases) %>% drop_na()

variable name

labels_raw_tibble <- labels_raw_tibble %>% mutate(splits = str_split(cases, “[ ]{0,}"[ ]{0,}”))

creating a function

add_cw_text <- function(x, y){ if(!is.na(as.numeric(x))){ x_new <- paste0(y, “==”, x,“~”) } else{ x_new <- paste0(“"”,x,“",”) } return(x_new) } ### Another function cw_statements <- labels_raw_tibble %>% rowwise() %>% mutate(splits_with_cw_text = list(modify(splits, add_cw_text, y = variable_name))) %>% mutate(cw_statement = paste(splits_with_cw_text, collapse = "“)) %>% mutate(cw_statement = paste0(”case_when(“, cw_statement,”TRUE~"NA")“)) %>% mutate(cw_statement = str_replace(cw_statement,”,"",“,”,")) %>% select(variable_name, cw_statement)

Do some final cleans of this function

cw_statements <- cw_statements %>% mutate(variable_name = str_remove_all(variable_name, “\r”)) %>% mutate(cw_statement = str_remove_all(cw_statement, “\r”))

Apply that dictionary to the raw data

gss <- raw_data %>% select(CASEID, agedc, achd_1c, achdmpl, totchdc, acu0c, agema1c, achb1c, rsh_131a, arretwk, slm_01, sex, brthcan, brthfcan, brthmcan, brthmacr, brthprvc, yrarri, prv, region, luc_rst, marstat, amb_01, vismin, alndimmg, bpr_16, bpr_19, ehg3_01b, odr_10, livarr12, dwelc, hsdsizec, brthpcan, brtpprvc, visminpr, rsh_125a, eop_200, uhw_16gr, lmam_01, acmpryr, srh_110, srh_115, religflg, rlr_110, lanhome, lan_01, famincg2, ttlincg2, noc1610, cc_20_1, cc_30_1, ccmoc1c, cor_031, cor_041, cu0rnkc, pr_cl, chh0014c, nochricc, grndpa, gparliv, evermar, ma0_220, nmarevrc, ree_02, rsh_131b, rto_101, rto_110, rto_120, rtw_300, sts_410, csp_105, csp_110a, csp_110b, csp_110c, csp_110d, csp_160, fi_110) %>% mutate_at(vars(agedc:fi_110), .funs = funs(ifelse(.>=96, NA, .))) %>% mutate_at(.vars = vars(sex:fi_110), .funs = funs(eval(parse(text = cw_statements %>% filter(variable_name==deparse(substitute(.))) %>% select(cw_statement) %>% pull()))))

Change the attributes name

gss <- gss %>% clean_names() %>% rename(age = agedc, age_first_child = achd_1c, age_youngest_child_under_6 = achdmpl, total_children = totchdc, age_start_relationship = acu0c, age_at_first_marriage = agema1c, age_at_first_birth = achb1c, distance_between_houses = rsh_131a, age_youngest_child_returned_work =

arretwk, feelings_life = slm_01, sex = sex, place_birth_canada = brthcan, place_birth_father = brthfcan, place_birth_mother = brthmcan, place_birth_macro_region = brthmacr, place_birth_province = brthprvc, year_arrived_canada = yrarri, province = prv, region = region, pop_center = luc_rst, marital_status = marstat, aboriginal = amb_01, vis_minority = vismin, age_immigration = alndimmg, landed_immigrant = bpr_16, citizenship_status = bpr_19, education = ehg3_01b, own_rent = odr_10, living_arrangement = livarr12, hh_type = dwelc, hh_size = hsdsizec, partner_birth_country = brthpcan, partner_birth_province = brtpprvc, partner_vis_minority = visminpr, partner_sex = rsh_125a, partner_education = eop_200, average_hours_worked = uhw_16gr, worked_last_week = lmam_01, partner_main_activity = acmpryr, self_rated_health = srh_110, self_rated_mental_health = srh_115, religion_has_affiliation = religflg, regilion_importance = rlr_110, language_home = lanhome, language_knowledge = lan_01, income_family = famincg2, income_respondent = ttlincg2, occupation = noc1610, childcare_regular = cc_20_1, childcare_type = cc_30_1, childcare_monthly_cost = ccmoc1c, ever_fathered_child = cor_031, ever_given_birth = cor_041, number_of_current_union = cu0rnkc, lives_with_partner = pr_cl, children_in_household = chh0014c, number_total_children_intention = nochricc, has_grandchildren = grndpa, grandparents_still_living = gparliv, ever_married = evermar, current_marriage_is_first = ma0_220, number_marriages = nmarevrc, religion_participation = ree_02, partner_location_residence = rsh_131b, full_part_time_work = rto_101, time_off_work_birth = rto_110, reason_no_time_off_birth = rto_120, returned_same_job = rtw_300, satisfied_time_children = sts_410, provide_or_receive_fin_supp = csp_105, fin_supp_child_supp = csp_110a, fin_supp_child_exp = csp_110b, fin_supp_lump = csp_110c, fin_supp_other = csp_110d, fin_supp_agreement = csp_160, future_children_intention = fi_110)

Clean up

gss <- gss %>% mutate_at(vars(age:future_children_intention), .funs = funs(ifelse(.==“Valid skip”|.==“Refusal”|.==“Not stated”, “NA”, .))) gss <- gss %>% mutate(is_male = ifelse(sex==“Male”, 1, 0)) gss <- gss %>% mutate_at(vars(fin_supp_child_supp:fin_supp_other), .funs = funs(case_when(.==“Yes”1, .==“No”0, .==“NA”~as.numeric(NA)))) main_act <- raw_data %>% mutate(main_activity = case_when(mpl_105a==“Yes”~ “Working at a paid job/business”, mpl_105b==“Yes” ~ “Looking for paid work”, mpl_105c==“Yes” ~ “Going to school”, mpl_105d==“Yes” ~ “Caring for children”, mpl_105e==“Yes” ~ “Household work”, mpl_105i==“Yes” ~ “Other”, TRUE~ “NA”)) %>% select(main_activity) %>% pull() age_diff <- raw_data %>% select(marstat, aprcu0c, adfgrma0) %>% mutate_at(.vars = vars(aprcu0c:adfgrma0), .funs = funs(eval(parse(text = cw_statements %>% filter(variable_name==deparse(substitute(.))) %>% select(cw_statement) %>% pull()))))) %>% mutate(age_diff = ifelse(marstat==“Living common-law”, aprcu0c, adfgrma0)) %>% mutate_at(vars(age_diff), .funs = funs(ifelse(.==“Valid skip”|.==“Refusal”|.==“Not stated”, “NA”, .))) %>% select(age_diff) %>% pull() gss <-

gss %>% mutate(main_activity = main_act, age_diff = age_diff) gss <- gss %>% rowwise() %>% mutate(hh_size = str_remove(string = hh_size, pattern = "\ ."*)) %>% mutate(hh_size = case_when(hh_size=="One" ~ 1, hh_size=="Two" ~ 2, hh_size=="Three" ~ 3, hh_size=="Four" ~ 4, hh_size=="Five" ~ 5, hh_size=="Six" ~ 6)) gss <- gss %>% rowwise() %>% mutate(number_marriages = str_remove(string = number_marriages, pattern = "\ .*")) %>% mutate(number_marriages = case_when(number_marriages=="No" ~ 0, number_marriages=="One" ~ 1, number_marriages=="Two" ~ 2, number_marriages=="Three" ~ 3, number_marriages=="Four" ~ 4)) gss <- gss %>% rowwise() %>% mutate(number_total_children_known = ifelse(number_total_children_intention=="Don't know"|number_total_children_intention=="NA", 0, 1)) %>% mutate(number_total_children_intention = str_remove(string = number_total_children_intention, pattern = "\ .*")) %>% mutate(number_total_children_intention = case_when(number_total_children_intention=="None" ~ 0, number_total_children_intention=="One" ~ 1, number_total_children_intention=="Two" ~ 2, number_total_children_intention=="Three" ~ 3, number_total_children_intention=="Four" ~ 4, number_total_children_intention=="Don't" ~ as.numeric(NA)))

save to a new csv file and finish cleaning

write_csv(gss, "gss.csv")

Model

I have used ggplot, histogram, logistic regression, linear regression to analaysis these datasets.

This is the mathmetical notation for linear regression :y = β0 + β1x + e, where β0 is the intercept, there could be β2,β3 and so on. Y must be numerical. Predictors can be both numerical and categorical. In this case, I have picked some numerical variables as predictors and age as Y.

This is the mathmetical notation for logistic regression : log(p/(1-p)) = β0 + β1x,where β0 is the intercept, there could be β2,β3 and so on. P is the probability of an event that is going to occur. β1 is a coefficient represents changes in log adds for every one unit increase in x. Y can either be 1 or 0.

Results

summary(gss) gss %>% ggplot(aes(x=age_at_first_birth, y=age))+ geom_point() gss %>% ggplot(aes(x=age_first_child, y=age))+ geom_point() gss %>% ggplot(aes(x=total_children, y=age))+ geom_point() gss %>% ggplot(aes(x=

total_children + age_first_child + age_at_first_birth + age_at_first_marriage+age_start_relationship, y= age))+ geom_point()

linear regression

df <- lm(age ~ total_children + age_first_child + age_at_first_birth + age_at_first_marriage+age_start_relationship, data=gss) summary(df) dff <-lm(age ~ total_children + income_family, data=gss) summary(dff)

Logistic Regression

gss <- gss %>% mutate(sex = case_when(sex == “Female” ~ ‘1’, sex == “Male” ~‘0’))
gss <- gss %>% mutate(income_status = case_when(income_family == “Less than $25,000” ~ ‘1’, income_family == “$25,000 to $49,999” ~‘2’, income_family == “$50,000 to $74,999” ~‘3’, income_family == “$75,000 to $99,999” ~‘4’, income_family == “$100,000 to $ 124,999” ~‘5’, income_family == "$125,000 and more" ~'6')) table(gss$income_status)
gss_numeric <- as.data.frame(apply(gss,2,as.numeric))
hist(gss_numeric
$income_status, xlab = 'income_status', ylab = 'Numbersofincome_status', main = 'CountsofIncomel$
sex,breaks = 2,xlab = ‘sex’,ylab = ‘Numbers of sex’,main = ‘Counts of sex’, col = ‘green’)

Standard Logistic Regression

df2 <- glm(sex ~ age +age_first_child+total_children+feelings_life + as.factor(income_status), data= gss_numeric, family=“binomial”) summary(df2)

Survey Estimation for Logistic Regression

n=length(gss_numeric$sex) N=60000 install.packages(“survey”) library(survey)

Using the Survey Library

b = rep(N, n) a <- svydesign(id=~1, data=gss_numeric, fpc=b) c <- svyglm(sex ~ age +age_first_child+total_children+feelings_life + as.factor(income_status),a, family=“binomial”) summary(c)

Discussion

By using summary, we can see the details of each variable.

Then we created several ggplots to determine relationships between factors.

There is a positive linear relationship between age_first_child and age.

There is a postive linear relationship between age and total_children, age_first_children, age_at_first_birth, age_at_first_marriage and age_start_relationship.

Then we run linear regression since we foud some relationship.

The function will be Y = (-0.54 + -0.01* total_children + 0.99* age_first_child +1.02* age_at_first_birth-0.01* age_at_first_marriage+0.02*age_start_relationship)

Since there are many missing values that has been removed, about 20272 rows. So we recreated a linear regression, this time, I added categorical variables income_family to test if age and income are related.

The function will be Y = 39.75 + 5.31* total_children - 1.54* '125,000 and more' + 8.07* '25,000 to $49,999' + 5.47*'50,000 to $74,999' + 1.90* '75,000 to $99,999' + 6.86*'Less than $25,000'

In order to generate logistic regression. We created a binary variable as our Y value which is sex. We set female = 1 and male = 0.

Then we sepertaed the income_family by 1,2,3,4,5,6 levels. '1' is the poorest family and '6' is richest family as the rank.

Since logistic regression Y is a binary.Current Y value is a character. So we changed GSS file to numerical and named it gss_numeric.

We created a histogram to show the income status from 1 through 6. According to the histogram, there are many income_family that have income over $125,000 and between $25,000 to $49,000.

We also created a histogram of female and male. According to the histogram, there are more females than males in the dataset.

The logistic regression function we got is Y = 3.79-0.097* age+0.091* age_first_child-0.19* total_children+0.026*feelings_life-0.294x2-0.58x3-0.63x4-0.73x5-0.63x6
There is about 1000 difference between null deviance and residual deviance, the larger the difference is, the better the model is.

Weaknesses

There are many missing values even after data cleaning. If by removing all the NAs,there will be 0 rows left. So in order to run some analysis, we have to obtain some missing values. But due to this is a questionaire, it's hard to obtain perfect answers. There are too many columns, factors in the dataset, which can create mislead when doing analysis.

Next Steps

Since we have produced predictive model, which is the logistic regression. We can try to predict some values. We also would like to create other algorithms such as random forest, or decision tree. Because there are too many factors in the dataset random forest and decision tree can deal with large dataset with multidimensions. we think GSS dataset is multidimensional.And of course, we should also set up a following survey to fill up these missing values by doing our best. Obtaining good dataset can lead to a better result.

References

Technology, A. (n.d.). Data Centre. Retrieved October 20, 2020, from http://dc.chass.utoronto.ca/myaccess.html Tyagi, P. (2018, December 25). Decision Tree. Retrieved from https://medium.com/@pytyagi/decision-tree-ac0c9e3b8258