Replication script ‘Experience of Discrimination in Egalitarian Societies: The Sámi and Majority Populations in Sweden and Norway’

for anonymous review

2023-03-16

This document is prepared in order to enable the replication of results presented in the paper “Experience of Discrimination in Egalitarian Societies: The Sámi and Majority Populations in Sweden and Norway”. It can be viewed as an html document for easy reading, and the code snippets below can be run in R.

Please note that the code relies on tidyverse framework. More specifically, the following packages are used:

```
library(knitr)
library(tidyverse)
library(skimr)
library(broom)
library(kableExtra)
library(simputation)
```

Data preparation

We begin with loading the dataset containing variables that we will need in the analyses. The data is from “Nordic Peoples Survey” (NPS).

```
load("NPS_df_select.RData")
df <- df_nps_sub
rm(df_nps_sub)
```

Ethnic background

We re-define ethnic background broadly to include any of own, mother’s, or father’s ethnic background. In the case of the Sámi, we include self-identification as well.

```
df <- df %>%
  mutate(
    maj_bg = ifelse(ethnbg_maj=="yes" | ethnbg_mo_maj=="yes" |
                      ethnbg_fa_maj=="yes", "yes", "no") %>% as_factor() %>%
      fct_relevel("no", "yes"),
    sami_bg = ifelse(ethnbg_sam=="yes" | ethnbg_mo_sam=="yes" | ethnbg_fa_sam=="yes" |
                       !is.na(selfident_sam) & selfident_sam=="yes",
                     "yes", "no") %>% as_factor() %>% fct_relevel("no", "yes"),
    fin_bg = ifelse(ethnbg_fin=="yes" | ethnbg_mo_fin=="yes" |
                      ethnbg_fa_fin=="yes", "yes", "no") %>% as_factor() %>%
      fct_relevel("no", "yes"),
    tor_bg = ifelse(ethnbg_tor=="yes" | ethnbg_mo_tor=="yes" |
                      ethnbg_fa_tor=="yes", "yes", "no") %>% as_factor() %>%
      fct_relevel("no", "yes"),
    imm_bg = ifelse(ethnbg_imm=="yes" | ethnbg_mo_imm=="yes" |
                      ethnbg_fa_imm=="yes", "yes", "no") %>% as_factor() %>%
      fct_relevel("no", "yes"),
    oth_min_bg = ifelse(ethnbg_oth=="yes" | ethnbg_mo_oth=="yes" |
                          ethnbg_fa_oth=="yes", "yes", "no") %>%
      as_factor() %>% fct_relevel("no", "yes")
    )
```

We also create a category of national minorities for those who indicate a minority background that is neither Sámi nor immigrant.

```
df <- df %>%
  mutate(natmin_bg = ifelse(fin_bg=="yes" | tor_bg=="yes" | oth_min_bg=="yes",
                            "yes", "no") %>% as_factor() %>% fct_relevel("no", "yes"))
```

For clearer contrasts, we combine different ethnic backgrounds into a single categorical variable with four distinct categories. To deal with overlaps (multiple backgrounds), first we distinguish between people with majority-only background and any minority background. Among those with a minority background, we prioritise the Sámi (as our main interest), then assign immigrants to their own category, and finally we use the category of national minorities from above. If none of these was selected, we define it as a missing value.

```
df <- df %>%
  mutate(
    eth_bg_cats = case_when(
      maj_bg=="yes" & sami_bg=="no" & imm_bg=="no" & natmin_bg=="no" ~ "Majority only",
      sami_bg=="yes" ~ "Sami",
      imm_bg=="yes" ~ "Immigrant",
      natmin_bg=="yes" ~ "National minorities",
      TRUE ~ NA_character_
      ) %>% as_factor() %>% fct_relevel("Majority only")
  )
```

Income

In order to calculate equivalence income, first we need to calculate the coefficients based on the household size. We will use OECD modified scale, so the first adults get 1, each additional adult gets 0.5, and each child gets 0.3.

```
df <- df %>%
  mutate(equiv_coef = 1 + (hhsize_adu - 1)*0.5 + hhsize_chi*0.3)
```

Income data is stored in two variables, first as the value in currency, and second (for those who have not provided the information in currency value) as one of 10 income brackets. To combine these, we transform the first into the second by using the same cut-off values. To calculate equivalence income, we need currency values; therefore, we assign the mid-values of the income brackets, then divide by the household size coefficient. We do this separately for each country due to differences in currencies, and we calculate standardised values within country sub-samples so that these are comparable between countries.

```
df_swe <- df %>%
  filter(country == "sweden") %>%
  mutate(
    hhinc_yg_comb_num = case_when(
      hhinc_yg < 185000 ~ 1,
      hhinc_yg >= 185000 & hhinc_yg < 279000 ~ 2,
      hhinc_yg >= 279000 & hhinc_yg < 350000 ~ 3,
      hhinc_yg >= 350000 & hhinc_yg < 431000 ~ 4,
      hhinc_yg >= 431000 & hhinc_yg < 513000 ~ 5,
      hhinc_yg >= 513000 & hhinc_yg < 589000 ~ 6,
      hhinc_yg >= 589000 & hhinc_yg < 671000 ~ 7,
      hhinc_yg >= 671000 & hhinc_yg < 780000 ~ 8,
      hhinc_yg >= 780000 & hhinc_yg < 971000 ~ 9,
      hhinc_yg >= 971000 ~ 10,
      TRUE ~ hhinc_yg_grp_num
    ),
    hhinc_yg_comb_rev = case_when(
      hhinc_yg_comb_num == 1 ~ 123000,
      hhinc_yg_comb_num == 2 ~ 232000,
      hhinc_yg_comb_num == 3 ~ 314500,
      hhinc_yg_comb_num == 4 ~ 390500,
      hhinc_yg_comb_num == 5 ~ 472000,
      hhinc_yg_comb_num == 6 ~ 551000,
      hhinc_yg_comb_num == 7 ~ 630000,
      hhinc_yg_comb_num == 8 ~ 725500,
      hhinc_yg_comb_num == 9 ~ 875500,
      hhinc_yg_comb_num == 10 ~ 1138500
    ),
    hhinc_yg_equiv = hhinc_yg_comb_rev / equiv_coef,
    hhinc_yg_equiv_std = scale(hhinc_yg_equiv)[,1]
  )

df_nor <- df %>%
  filter(country == "norway") %>%
  mutate(
    hhinc_yg_comb_num = case_when(
      hhinc_yg < 256000 ~ 1,
      hhinc_yg >= 256000 & hhinc_yg < 351000 ~ 2,
      hhinc_yg >= 351000 & hhinc_yg < 452000 ~ 3,
      hhinc_yg >= 452000 & hhinc_yg < 551000 ~ 4,
      hhinc_yg >= 551000 & hhinc_yg < 666000 ~ 5,
      hhinc_yg >= 666000 & hhinc_yg < 813000 ~ 6,
      hhinc_yg >= 813000 & hhinc_yg < 986000 ~ 7,
      hhinc_yg >= 986000 & hhinc_yg < 1194000 ~ 8,
      hhinc_yg >= 1194000 & hhinc_yg < 1526000 ~ 9,
      hhinc_yg >= 1526000 ~ 10,
      TRUE ~ hhinc_yg_grp_num
    ),
    hhinc_yg_comb_rev = case_when(
      hhinc_yg_comb_num == 1 ~ 211500,
      hhinc_yg_comb_num == 2 ~ 303500,
      hhinc_yg_comb_num == 3 ~ 401500,
      hhinc_yg_comb_num == 4 ~ 501500,
      hhinc_yg_comb_num == 5 ~ 608500,
      hhinc_yg_comb_num == 6 ~ 739500,
```

```
      hhinc_yg_comb_num == 7 ~ 899500,
      hhinc_yg_comb_num == 8 ~ 1090000,
      hhinc_yg_comb_num == 9 ~ 1360000,
      hhinc_yg_comb_num == 10 ~ 1791000
    ),
    hhinc_yg_equiv = hhinc_yg_comb_rev / equiv_coef,
    hhinc_yg_equiv_std = scale(hhinc_yg_equiv)[,1]
  )

df <- bind_rows(df_swe, df_nor)
rm(df_swe, df_nor)
```

Age

We calculate age based on the given year of birth, and we add a variable for age-squared to account for a possible quadratic effect of age.

```
df <- df %>%
  mutate(
    age = 2021 - byear,
    age_sq = age^2
  )
```

Discrimination

The variables that we process from this point onward are concerned with the second stage questionnaire data. So we create this subset.

```
df_ssq <- df %>%
  filter(participation == "CATI & stage II")
```

In the second-stage questionnaire, answer categories for the question on discrimination included frequency. To ensure compatibility and comparability between samples, we transform the second version into a yes-no format similar to the one asked in the telephone interview. This will also help us to run logit regressions consistently.

We begin with a description and comparison of the two versions.

```
df_ssq %>%
  summarise(cati_na = sum(is.na(disc_exp_cati)),
            onli_na = sum(is.na(disc_exp)),
            caon_na = sum(is.na(disc_exp) & is.na(disc_exp_cati)),
            cati_na_prop = sum(is.na(disc_exp_cati))/n(),
            onli_na_prop = sum(is.na(disc_exp))/n(),
            caon_na_prop = sum(is.na(disc_exp) & is.na(disc_exp_cati))/n(),
            cati_yes_prop = sum(disc_exp_cati=="yes", na.rm=TRUE)/n(),
            onli_yes_prop = sum(disc_exp!="never" & !is.na(disc_exp), na.rm=TRUE)/n(),
            caon_yes_prop = sum(disc_exp_cati=="yes" |
                                  (disc_exp!="never" & !is.na(disc_exp)), na.rm=TRUE)/n()) %
>%
  pivot_longer(everything()) %>% kable(format = "html") %>%
  kable_classic(full_width = F)
```

name	value
cati_na	9.0000000
onli_na	155.0000000
caon_na	2.0000000
cati_na_prop	0.0063380
onli_na_prop	0.1091549
caon_na_prop	0.0014085
cati_yes_prop	0.1922535
onli_yes_prop	0.3239437
caon_yes_prop	0.3542254

```
df_ssq %>%
  group_by(disc_exp_cati) %>%
  summarise(p_no = sum(disc_exp=="never", na.rm=TRUE)/n(),
            p_yes = sum(disc_exp!="never" & !is.na(disc_exp), na.rm=TRUE)/n(),
            na = sum(is.na(disc_exp))/n()) %>% kable(format = "html") %>%
  kable_classic(full_width = F)
```

disc_exp_cati	p_no	p_yes	na
no	0.6933216	0.1994728	0.1072056
yes	0.0439560	0.8424908	0.1135531
NA	0.4444444	0.3333333	0.2222222

```
df_ssq %>%
  group_by(disc_exp) %>%
  summarise(p_no = sum(disc_exp_cati=="no", na.rm=TRUE)/n(),
            p_yes = sum(disc_exp_cati!="no" & !is.na(disc_exp_cati), na.rm=TRUE)/n(),
            na = sum(is.na(disc_exp_cati))/n()) %>% kable(format = "html") %>%
  kable_classic(full_width = F)
```

disc_exp	p_no	p_yes	na
never	0.9801242	0.0149068	0.0049689
on rare occasions	0.6475410	0.3442623	0.0081967
every now and then	0.3437500	0.6500000	0.0062500
often	0.2500000	0.7500000	0.0000000
NA	0.7870968	0.2000000	0.0129032

In the second-stage questionnaire, there are 155 missing values for the discrimination question, corresponding to 11% of the total. Only 9 respondents had not answered the discrimination question at the at the telephone interview, and 2 of them did not answer it at the second stage. Apparently a considerable number did not want to answer the question for a second time.

People are more likely to change their mind from no to yes (20%), than from yes to no (4%). ‘On rare occasions’ seems to be interpreted closer to ‘no’: 65% had said ‘no’ in the telephone interview. Meanwhile, not answering looks like a substitute for saying no: 79% of second-stage missing values were ’no’s at the telephone interview (respondents unwilling to answer for a second time). So it makes sense to minimise missing values with answers from the telephone interview.

To combine variables, we follow this strategy: (1) if the respondents did not answer the question either in telephone interview or in the questionnaire, it is missing; (2) if the respondent did not answer the question in the questionnaire but had answered the question with a “no” in the telephone interview, we assign “no”; (3) if the answer in the questionnaire is “never”, we assign “no”; (4) if the answer in the questionnaire is “on rare occasions” and the answer in the telephone interview was “no”, we assign “no”; (5) for all remaining cases, we assign “yes”.

```
df_ssq <- df_ssq %>%
  mutate(disc_exp_comb = case_when(
    is.na(disc_exp) & is.na(disc_exp_cati) ~ NA_character_,
    is.na(disc_exp) & disc_exp_cati == "no" ~ "no",
    disc_exp == "never" ~ "no",
    disc_exp == "on rare occasions" & disc_exp_cati == "no" ~"no",
    TRUE ~ "yes") %>% as_factor() %>% fct_relevel("no", "yes"))
```

Language variables

Imputation

Before combining language variables, we should deal with the missing values, otherwise missing values in a single dimension will result in missing values for the whole variable. We follow this strategy for each battery of questions (self-assessed proficiency in Sámi, home use of Sámi, general use of Sámi):

- We remove the all-missing cases
- We use k-nearest-neighbour method to impute data
- We add back the all-missing cases
- Finally, we add them back to the main dataset as new variables with suffix _imp.

```
df_prof_sami <- df_ssq %>%
  select(id, country, sami_bg, prof_sami_passive_num:prof_sami_write_num)
df_prof_sami <- df_prof_sami %>%
  filter(if_any(prof_sami_passive_num:prof_sami_write_num, ~!is.na(.))) %>%
  impute_knn( . -country -sami_bg -id ~ . -country -sami_bg -id | country + sami_bg) %>%
  bind_rows(df_prof_sami %>%
            filter(!if_any(prof_sami_passive_num:prof_sami_write_num, ~!is.na(.)))) %>%
  arrange(as.numeric(id))

df_prof_maj <- df_ssq %>%
  select(id, country, sami_bg, prof_maj_passive_num:prof_maj_write_num)
df_prof_maj <- df_prof_maj %>%
  filter(if_any(prof_maj_passive_num:prof_maj_write_num, ~!is.na(.))) %>%
  impute_knn( . -country -sami_bg -id ~ . -country -sami_bg -id | country + sami_bg) %>%
  bind_rows(df_prof_maj %>%
            filter(!if_any(prof_maj_passive_num:prof_maj_write_num, ~!is.na(.)))) %>%
  arrange(as.numeric(id))

df_hlang <- df_ssq %>%
  select(id, country, hlang_mot_bef_num:hlang_och_cur_num)
df_hlang <- df_hlang %>%
  filter(if_any(hlang_mot_bef_num:hlang_och_cur_num, ~!is.na(.))) %>%
  impute_knn( . -country -id ~ . -country -id | country) %>%
  bind_rows(df_hlang %>%
            filter(!if_any(hlang_mot_bef_num:hlang_och_cur_num, ~!is.na(.)))) %>%
  arrange(as.numeric(id))

df_lit_sami <- df_ssq %>%
  select(id, country, freq_radio_sami_num:freq_writelong_sami_num)
df_lit_sami <- df_lit_sami %>%
  filter(if_any(freq_radio_sami_num:freq_writelong_sami_num, ~!is.na(.))) %>%
  impute_knn( . -country -id ~ . -country -id | country) %>%
  bind_rows(df_lit_sami %>%
            filter(!if_any(freq_radio_sami_num:freq_writelong_sami_num, ~!is.na(.)))) %>%
  arrange(as.numeric(id))

df_lit_maj <- df_ssq %>%
  select(id, country, freq_radio_maj_num:freq_writelong_maj_num)
df_lit_maj <- df_lit_maj %>%
  filter(if_any(freq_radio_maj_num:freq_writelong_maj_num, ~!is.na(.))) %>%
  impute_knn( . -country -id ~ . -country -id | country) %>%
  bind_rows(df_lit_maj %>%
            filter(!if_any(freq_radio_maj_num:freq_writelong_maj_num, ~!is.na(.)))) %>%
  arrange(as.numeric(id))

df_lang_imp <- bind_cols(df_prof_sami, df_prof_maj[-c(1:3)], df_hlang[-c(1:2)],
                         df_lit_sami[-c(1:2)], df_lit_maj[-c(1:2)])
rm(df_prof_sami, df_prof_maj, df_hlang, df_lit_sami, df_lit_maj)

df_ssq <- df_ssq %>%
  left_join(df_lang_imp, by = "id", suffix = c("", "_imp")) %>%
  select(-country_imp, -sami_bg_imp)
```

Dimension reduction

We use Principal Component Analysis (PCA) to see how the observed data can be reduced to fewer variables, and how much variation would be retained/lost with this reduction.

```
pca_prof_sami <- prcomp(na.omit(df_lang_imp[4:7]), scale. = TRUE)
pca_prof_maj <- prcomp(na.omit(df_lang_imp[8:11]), scale. = TRUE)
pca_hlang <- prcomp(na.omit(df_lang_imp[12:26]), scale. = TRUE)
pca_lit_sami <- prcomp(na.omit(df_lang_imp[27:32]), scale. = TRUE)
pca_lit_maj <- prcomp(na.omit(df_lang_imp[33:38]), scale. = TRUE)

summary(pca_prof_sami) ; print(pca_prof_sami)
```

```
## Importance of components:
##                           PC1    PC2    PC3     PC4
## Standard deviation     1.7868 0.7541 0.4050 0.27334
## Proportion of Variance 0.7982 0.1422 0.0410 0.01868
## Cumulative Proportion  0.7982 0.9403 0.9813 1.00000
```

```
## Standard deviations (1, .., p=4):
## [1] 1.7867846 0.7541155 0.4049611 0.2733441
##
## Rotation (n x k) = (4 x 4):
##                             PC1         PC2        PC3         PC4
## prof_sami_passive_num 0.4272919  0.83752128 -0.3267498 -0.09599110
## prof_sami_active_num  0.5250689  0.05650205  0.8474384 -0.05439034
## prof_sami_read_num    0.5308548 -0.28647654 -0.2614533  0.75350286
## prof_sami_write_num   0.5098210 -0.46184068 -0.3266877 -0.64812101
```

```
summary(pca_prof_maj) ; print(pca_prof_maj)
```

```
## Importance of components:
##                           PC1     PC2     PC3     PC4
## Standard deviation     1.8801 0.43924 0.40426 0.33022
## Proportion of Variance 0.8837 0.04823 0.04086 0.02726
## Cumulative Proportion  0.8837 0.93188 0.97274 1.00000
```

```
## Standard deviations (1, .., p=4):
## [1] 1.8800529 0.4392427 0.4042565 0.3302176
##
## Rotation (n x k) = (4 x 4):
##                            PC1         PC2         PC3         PC4
## prof_maj_passive_num 0.4993080 -0.07729038  0.81854639 -0.27331222
## prof_maj_active_num  0.4926774 -0.79265660 -0.35254890  0.06836456
## prof_maj_read_num    0.5063625  0.38787881 -0.01514635  0.77001145
## prof_maj_write_num   0.5015551  0.46397385 -0.45327880 -0.57245880
```

```
summary(pca_hlang) ; print(pca_hlang)
```

```
## Importance of components:
##                           PC1     PC2     PC3     PC4     PC5    PC6     PC7
## Standard deviation     3.5764 0.78021 0.68716 0.59360 0.49810 0.4382 0.28223
## Proportion of Variance 0.8527 0.04058 0.03148 0.02349 0.01654 0.0128 0.00531
## Cumulative Proportion  0.8527 0.89331 0.92479 0.94828 0.96482 0.9776 0.98293
##                           PC8     PC9    PC10    PC11   PC12    PC13    PC14
## Standard deviation     0.2654 0.21361 0.19292 0.17679 0.1595 0.13900 0.12799
## Proportion of Variance 0.0047 0.00304 0.00248 0.00208 0.0017 0.00129 0.00109
## Cumulative Proportion  0.9876 0.99066 0.99314 0.99523 0.9969 0.99821 0.99931
##                           PC15
## Standard deviation     0.10204
## Proportion of Variance 0.00069
## Cumulative Proportion  1.00000
```

```
## Standard deviations (1, .., p=15):
##  [1] 3.5764346 0.7802053 0.6871642 0.5935993 0.4981003 0.4381597 0.2822298
##  [8] 0.2653813 0.2136109 0.1929182 0.1767872 0.1595479 0.1390049 0.1279864
## [15] 0.1020434
##
## Rotation (n x k) = (15 x 15):
##                          PC1          PC2         PC3          PC4          PC5
## hlang_mot_bef_num 0.2541569  0.298697922 -0.33148551  0.063887547 -0.062372152
## hlang_mot_aft_num 0.2544142  0.183083868 -0.43588521  0.015677168  0.364270041
## hlang_mot_cur_num 0.2545924  0.007015149 -0.41807692 -0.009865989  0.453833667
## hlang_fat_bef_num 0.2534484  0.230206229  0.40087633 -0.207721889 -0.017331894
## hlang_fat_aft_num 0.2563472  0.153265468  0.39957908 -0.291934936  0.319830616
## hlang_fat_cur_num 0.2616020  0.020408840  0.28603690 -0.270004296  0.317224788
## hlang_sib_bef_num 0.2646817  0.058604101 -0.13028478 -0.299096019 -0.423070270
## hlang_sib_aft_num 0.2676860 -0.015199466 -0.13812963 -0.276598696 -0.347290528
## hlang_sib_cur_num 0.2662545 -0.063328399 -0.15699061 -0.242410340 -0.322431771
## hlang_grp_bef_num 0.2510222  0.309531058  0.13645110  0.492819592 -0.205054016
## hlang_grp_aft_num 0.2558470  0.198915571  0.17228008  0.491649988 -0.033545261
## hlang_grp_cur_num 0.2644153  0.032675325  0.09070114  0.185120951 -0.072427041
## hlang_och_bef_num 0.2563341 -0.467498388  0.04390533  0.121487219  0.044648042
## hlang_och_aft_num 0.2562184 -0.466300015  0.05270161  0.133479811  0.008241392
## hlang_och_cur_num 0.2552074 -0.469252544  0.03492292  0.134662129  0.013696024
##                            PC6         PC7          PC8         PC9
## hlang_mot_bef_num  0.481415178 -0.35007416  0.008607733 -0.09126165
## hlang_mot_aft_num  0.035630154  0.01077385  0.553520815 -0.09574779
## hlang_mot_cur_num -0.132210865  0.17647603 -0.625495878  0.21538804
## hlang_fat_bef_num  0.462590077 -0.13261515 -0.247794884  0.21281011
## hlang_fat_aft_num  0.002262628  0.15664601  0.185537216  0.15714951
## hlang_fat_cur_num -0.257930180 -0.01761261  0.008559923 -0.45377249
## hlang_sib_bef_num  0.026375947 -0.02044949  0.081260037  0.07285863
## hlang_sib_aft_num -0.172046881  0.16780550  0.138652749  0.12679972
## hlang_sib_cur_num -0.261059331  0.21564838 -0.154883505 -0.09410148
## hlang_grp_bef_num  0.095509546  0.40358720 -0.215485963 -0.43522058
## hlang_grp_aft_num -0.277467579  0.13675287  0.262684460  0.48858833
## hlang_grp_cur_num -0.450332574 -0.73835012 -0.139291902 -0.07478466
## hlang_och_bef_num  0.166348000  0.01382207  0.074068495 -0.18276731
## hlang_och_aft_num  0.157939669  0.02485456  0.116436081 -0.19575200
## hlang_och_cur_num  0.170274426 -0.03344396 -0.051924405  0.35288165
##                           PC10        PC11        PC12        PC13        PC14
## hlang_mot_bef_num  0.4663694251 -0.01082375  0.04361835  0.14822760  0.33215236
## hlang_mot_aft_num -0.3074387035 -0.09078838  0.18200609 -0.13807099 -0.32049092
## hlang_mot_cur_num -0.0502243697  0.05487848 -0.24012274  0.03671438  0.02766137
## hlang_fat_bef_num -0.0715972387 -0.20667073 -0.02154440 -0.20358464 -0.43354631
## hlang_fat_aft_num -0.2260196787 -0.18272246 -0.01236486  0.23869927  0.46332431
## hlang_fat_cur_num  0.4262010446  0.41376132  0.10726148 -0.05632006 -0.11139017
## hlang_sib_bef_num -0.2507606866  0.56912626 -0.35782933 -0.02826817 -0.12834820
## hlang_sib_aft_num -0.0079911856 -0.10641073  0.02770149  0.06770334  0.33744649
## hlang_sib_cur_num  0.2381416869 -0.46239482  0.29087961 -0.07603013 -0.23485127
## hlang_grp_bef_num -0.2929884620  0.09468931  0.15251773  0.04255871  0.09318537
## hlang_grp_aft_num  0.3890480446  0.04077943 -0.20614685 -0.08226503 -0.12405615
## hlang_grp_cur_num -0.2921558267 -0.13404999  0.02352556  0.03291472  0.05757350
## hlang_och_bef_num  0.0002490619 -0.14028149 -0.29042831 -0.66495176  0.28454469
## hlang_och_aft_num  0.0371566523 -0.17520199 -0.34140404  0.62414735 -0.27646881
## hlang_och_cur_num -0.0633052572  0.33663033  0.64385799  0.05894656  0.02946665
##                           PC15
```

```
## hlang_mot_bef_num  0.117461540
## hlang_mot_aft_num -0.064926997
## hlang_mot_cur_num -0.052115948
## hlang_fat_bef_num -0.250278586
## hlang_fat_aft_num  0.351681040
## hlang_fat_cur_num -0.151807332
## hlang_sib_bef_num  0.309778790
## hlang_sib_aft_num -0.698745953
## hlang_sib_cur_num  0.417324464
## hlang_grp_bef_num -0.034937178
## hlang_grp_aft_num  0.036917205
## hlang_grp_cur_num -0.006591097
## hlang_och_bef_num  0.056133627
## hlang_och_aft_num -0.064709497
## hlang_och_cur_num  0.039661862
```

```
summary(pca_lit_sami) ; print(pca_lit_sami)
```

```
## Importance of components:
##                           PC1    PC2     PC3     PC4    PC5     PC6
## Standard deviation     2.0189 0.8711 0.62867 0.61625 0.4673 0.41454
## Proportion of Variance 0.6793 0.1265 0.06587 0.06329 0.0364 0.02864
## Cumulative Proportion  0.6793 0.8058 0.87167 0.93496 0.9714 1.00000
```

```
## Standard deviations (1, .., p=6):
## [1] 2.0189169 0.8710658 0.6286658 0.6162503 0.4673209 0.4145422
## 
## Rotation (n x k) = (6 x 6):
##                                 PC1         PC2         PC3        PC4
## freq_radio_sami_num      0.3626749  0.59088749 -0.35971510  0.6233922
## freq_tv_sami_num         0.3742256  0.51145562  0.69780156 -0.2950239
## freq_readshort_sami_num  0.4429394 -0.03987833 -0.30153025 -0.4357208
## freq_readlong_sami_num   0.4494246 -0.15675699 -0.09952563 -0.1448022
## freq_writeshort_sami_num 0.4355950 -0.26223332 -0.27986064 -0.1396990
## freq_writelong_sami_num  0.3748350 -0.54252721  0.45225016  0.5422241
##                                   PC5         PC6
## freq_radio_sami_num       0.007596225 -0.03533293
## freq_tv_sami_num         -0.146715172  0.05363647
## freq_readshort_sami_num   0.176884496 -0.70010943
## freq_readlong_sami_num    0.631895264  0.58590108
## freq_writeshort_sami_num -0.739476395  0.31117122
## freq_writelong_sami_num   0.031809858 -0.25615208
```

```
summary(pca_lit_maj) ; print(pca_lit_maj)
```

```
## Importance of components:
##                           PC1    PC2    PC3    PC4     PC5     PC6
## Standard deviation     1.6279 1.0059 0.9623 0.7972 0.68688 0.55193
## Proportion of Variance 0.4417 0.1686 0.1543 0.1059 0.07863 0.05077
## Cumulative Proportion  0.4417 0.6103 0.7647 0.8706 0.94923 1.00000
```

```
## Standard deviations (1, .., p=6):
## [1] 1.6279379 1.0058824 0.9623366 0.7971811 0.6868825 0.5519256
##
## Rotation (n x k) = (6 x 6):
##                                PC1        PC2         PC3         PC4
## freq_radio_maj_num      -0.4442815  0.2639731 -0.39658216 -0.02913599
## freq_tv_maj_num         -0.4085194  0.2273775 -0.54553873  0.34227023
## freq_readshort_maj_num  -0.4476850 -0.5413688 -0.01158945 -0.06198612
## freq_readlong_maj_num   -0.3851137  0.3936731  0.26458774 -0.74933878
## freq_writeshort_maj_num -0.4342972 -0.5420337  0.22596546  0.03042195
## freq_writelong_maj_num  -0.3136517  0.3698085  0.65108272  0.56189483
##                                  PC5        PC6
## freq_radio_maj_num       0.713435318 -0.2565460
## freq_tv_maj_num         -0.510976536  0.3248891
## freq_readshort_maj_num  -0.315692804 -0.6347120
## freq_readlong_maj_num   -0.226911781  0.1170670
## freq_writeshort_maj_num  0.280611744  0.6219779
## freq_writelong_maj_num  -0.004378795 -0.1587786
```

Looking at the first components, a good part of the variation can be explained by a compound variable, especially for self-assessed proficiency and home use (over 80%), while for general use the proportion of variance would be 68%.

Next, we want to use individual loadings for calculating weighted average scores. We turn these values into weights such that they add up to 1. The summaries above show that they are quite close to each other, so weighted average would not be too different from simple average.

```
prof_sami_pca_wt <- pca_prof_sami$rotation[,1]
prof_maj_pca_wt <- pca_prof_maj$rotation[,1]
hlang_pca_wt <- pca_hlang$rotation[,1]
lit_sami_pca_wt <- pca_lit_sami$rotation[,1]
lit_maj_pca_wt <- pca_lit_maj$rotation[,1]

weights_list <- list(prof_sami_pca_wt, prof_maj_pca_wt, hlang_pca_wt,
                     lit_sami_pca_wt, lit_maj_pca_wt)
weights_list <- map(weights_list, ~ . / sum(.))

df_ssq <- df_ssq %>%
  mutate(
    prof_sami_com = prof_sami_passive_num_imp * weights_list[[1]][1] +
           prof_sami_active_num_imp * weights_list[[1]][2] +
           prof_sami_read_num_imp * weights_list[[1]][3] +
           prof_sami_write_num_imp * weights_list[[1]][4],
    prof_maj_com = prof_maj_passive_num_imp * weights_list[[2]][1] +
           prof_maj_active_num_imp * weights_list[[2]][2] +
           prof_maj_read_num_imp * weights_list[[2]][3] +
           prof_maj_write_num_imp * weights_list[[2]][4],
    hlang_com = hlang_mot_bef_num_imp * weights_list[[3]][1] +
           hlang_mot_aft_num_imp * weights_list[[3]][2] +
           hlang_mot_cur_num_imp * weights_list[[3]][3] +
           hlang_fat_bef_num_imp * weights_list[[3]][4] +
           hlang_fat_aft_num_imp * weights_list[[3]][5] +
           hlang_fat_cur_num_imp * weights_list[[3]][6] +
           hlang_sib_bef_num_imp * weights_list[[3]][7] +
           hlang_sib_aft_num_imp * weights_list[[3]][8] +
           hlang_sib_cur_num_imp * weights_list[[3]][9] +
           hlang_grp_bef_num_imp * weights_list[[3]][10] +
           hlang_grp_aft_num_imp * weights_list[[3]][11] +
           hlang_grp_cur_num_imp * weights_list[[3]][12] +
           hlang_och_bef_num_imp * weights_list[[3]][13] +
           hlang_och_aft_num_imp * weights_list[[3]][14] +
           hlang_och_cur_num_imp * weights_list[[3]][15],
    lit_sami_com = freq_radio_sami_num_imp * weights_list[[4]][1] +
           freq_tv_sami_num_imp * weights_list[[4]][2] +
           freq_readshort_sami_num_imp * weights_list[[4]][3] +
           freq_readlong_sami_num_imp * weights_list[[4]][4] +
           freq_writeshort_sami_num_imp * weights_list[[4]][5] +
           freq_writelong_sami_num_imp * weights_list[[4]][6],
    lit_maj_com = freq_radio_maj_num_imp * weights_list[[5]][1] +
           freq_tv_maj_num_imp * weights_list[[5]][2] +
           freq_readshort_maj_num_imp * weights_list[[5]][3] +
           freq_readlong_maj_num_imp * weights_list[[5]][4] +
           freq_writeshort_maj_num_imp * weights_list[[5]][5] +
           freq_writelong_maj_num_imp * weights_list[[5]][6]
  )
rm(df_lang_imp, pca_hlang, pca_lit_sami, pca_prof_sami, pca_prof_maj,
   weights_list, hlang_pca_wt, lit_sami_pca_wt, prof_sami_pca_wt, prof_maj_pca_wt,
   lit_maj_pca_wt, pca_lit_maj)
```

Education

First, we check how meaningful different categories of education level are.

```
df_ssq %>%
  group_by(country) %>%
  summarise(
    lvl_0 = sum(edulevel=="compulsory education not completed", na.rm=T)/n(),
    lvl_1 = sum(edulevel=="compulsory education", na.rm=T)/n(),
    lvl_2 = sum(edulevel=="upper secondary", na.rm=T)/n(),
    lvl_3 = sum(edulevel=="advanced vocational", na.rm=T)/n(),
    lvl_4 = sum(edulevel=="BA-level", na.rm=T)/n(),
    lvl_5 = sum(edulevel=="MA-level or higher", na.rm=T)/n(),
    no_ans = sum(is.na(edulevel))/n()) %>%
  pivot_longer(!country) %>%
  pivot_wider(names_from = "country") %>%
  kable(format = "html", digits = 3) %>%
  kable_classic(full_width = F)
```

name	norway	sweden
lvl_0	0.004	0.002
lvl_1	0.036	0.058
lvl_2	0.217	0.377
lvl_3	0.120	0.186
lvl_4	0.307	0.158
lvl_5	0.199	0.122
no_ans	0.118	0.097

There are very few people who selected levels 0 and 1: we merge them with level 2 (pre-tertiary education). We can see country differences for level 3 (higher in Sweden), and levels 4 and 5 (higher in Norway): we keep level 3 (advanced vocational education or university education without degree) as it is, and we combine levels 4 and 5 (education with academic degree).

```
df_ssq <- df_ssq %>%
  mutate(
    edulevel_red = case_when(
      edulevel=="compulsory education not completed" |
        edulevel=="compulsory education" |
        edulevel=="upper secondary" ~ "Pre-tertiary" ,
      edulevel=="advanced vocational" ~ "Advanced vocational",
      edulevel=="BA-level" |
        edulevel=="MA-level or higher" ~ "University",
      TRUE ~ NA_character_
    ) %>% as_factor()
  )
```

Descriptive statistics

Categorical variables

Telephone interview data - full

Here we calculate the statistics for the first version of discrimination experience, ethnic background categories, and gender.

```
df %>%
  group_by(country) %>%
  summarise(
    N_disc = sum(disc_exp_cati=="yes", na.rm = TRUE),
    R_disc = sum(disc_exp_cati=="yes", na.rm = TRUE)/n(),
    N_sami = sum(eth_bg_cats=="Sami", na.rm = TRUE),
    R_sami = sum(eth_bg_cats=="Sami", na.rm = TRUE)/n(),
    N_natmin = sum(eth_bg_cats=="National minorities", na.rm = TRUE),
    R_natmin = sum(eth_bg_cats=="National minorities", na.rm = TRUE)/n(),
    N_imm = sum(eth_bg_cats=="Immigrant", na.rm = TRUE),
    R_imm = sum(eth_bg_cats=="Immigrant", na.rm = TRUE)/n(),
    N_maj = sum(eth_bg_cats=="Majority", na.rm = TRUE),
    R_maj = sum(eth_bg_cats=="Majority", na.rm = TRUE)/n(),
    N_male = sum(gender=="male", na.rm = TRUE),
    R_male = sum(gender=="male", na.rm = TRUE)/n(),
    N_female = sum(gender=="female", na.rm = TRUE),
    R_female = sum(gender=="female", na.rm = TRUE)/n(),
    N_div = sum(gender=="diverse", na.rm = TRUE),
    R_div = sum(gender=="diverse", na.rm = TRUE)/n()
    ) %>%
  pivot_longer(!country) %>%
  pivot_wider(names_from = "country") %>%
  kable(format = "html", digits = 3) %>%
  kable_classic(full_width = F)
```

name	norway	sweden
N_disc	268.000	534.000
R_disc	0.112	0.177
N_sami	1072.000	847.000
R_sami	0.447	0.280
N_natmin	242.000	804.000
R_natmin	0.101	0.266
N_imm	192.000	211.000
R_imm	0.080	0.070
N_maj	0.000	0.000
R_maj	0.000	0.000
N_male	1119.000	1361.000
R_male	0.467	0.451
N_female	1277.000	1636.000
R_female	0.533	0.542
N_div	0.000	6.000
R_div	0.000	0.002

Telephone interview data - Sámi-only

Here we calculate statistics for home use of a Sámi language.

```
df %>%
  filter(sami_bg=="yes") %>%
  group_by(country) %>%
  summarise(
    N_hlang = sum(hlang_sam=="yes", na.rm = TRUE),
    R_hlang = sum(hlang_sam=="yes", na.rm = TRUE)/n(),
    N_hlang_par = sum(hlang_par_sam=="yes", na.rm = TRUE),
    R_hlang_par = sum(hlang_par_sam=="yes", na.rm = TRUE)/n(),
    N_hlang_gra = sum(hlang_gra_sam=="yes", na.rm = TRUE),
    R_hlang_gra = sum(hlang_gra_sam=="yes", na.rm = TRUE)/n()
  ) %>%
  pivot_longer(!country) %>%
  pivot_wider(names_from = "country") %>%
  kable(format = "html", digits = 3) %>%
  kable_classic(full_width = F)
```

name	norway	sweden
N_hlang	233.000	173.000
R_hlang	0.217	0.204
N_hlang_par	473.000	260.000
R_hlang_par	0.441	0.307
N_hlang_gra	787.000	458.000
R_hlang_gra	0.734	0.541

Second-stage questionnaire data

Here we calculate statistics for the combined version of discrimination experience, and education levels as reduced to three groups.

```
df_ssq %>%
  group_by(country) %>%
  summarise(
    N_disc = sum(disc_exp_comb=="yes", na.rm = TRUE),
    R_disc = sum(disc_exp_comb=="yes", na.rm = TRUE)/n(),
    N_edu1 = sum(edulevel_red=="Pre-tertiary", na.rm = TRUE),
    R_edu1 = sum(edulevel_red=="Pre-tertiary", na.rm = TRUE)/n(),
    N_edu2 = sum(edulevel_red=="Advanced vocational", na.rm = TRUE),
    R_edu2 = sum(edulevel_red=="Advanced vocational", na.rm = TRUE)/n(),
    N_edu3 = sum(edulevel_red=="University", na.rm = TRUE),
    R_edu3 = sum(edulevel_red=="University", na.rm = TRUE)/n()
    ) %>%
  pivot_longer(!country) %>%
  pivot_wider(names_from = "country") %>%
  kable(format = "html", digits = 3) %>%
  kable_classic(full_width = F)
```

name	norway	sweden
N_disc	90.000	243.000
R_disc	0.179	0.265
N_edu1	129.000	401.000
R_edu1	0.257	0.437
N_edu2	60.000	171.000
R_edu2	0.120	0.186

name	norway	sweden
N_edu3	254.000	257.000
R_edu3	0.506	0.280

Numeric variables

Telephone interview data

Here we calculate statistics for adjusted household income and age.

```
df %>%
  group_by(country) %>%
  summarise_at(c("hhinc_yg_equiv", "age"),
               list(min = min,
                    median = median,
                    mean = mean,
                    max = max,
                    sd = sd),
                na.rm=TRUE) %>%
  pivot_longer(!country) %>%
  pivot_wider(names_from = "country") %>%
  kable(format = "html", digits = 0) %>%
  kable_classic(full_width = F)
```

name	norway	sweden
hhinc_yg_equiv_min	84600	14471
age_min	18	18
hhinc_yg_equiv_median	503704	314500
age_median	55	55
hhinc_yg_equiv_mean	567654	335918
age_mean	54	53
hhinc_yg_equiv_max	1791000	1138500
age_max	97	101
hhinc_yg_equiv_sd	258067	209191
age_sd	16	17

Second-stage questionnaire data - full

Here we calculate statistics for self-assessed proficiency in a Sámi language and in the majority language, self-placement in social ladder, satisfaction with democracy, and perceived fairness of wealth distribution.

```
df_ssq %>%
  group_by(country) %>%
  summarise_at(c("prof_sami_com", "prof_maj_com"),
               list(min = min,
                    median = median,
                    mean = mean,
                    max = max,
                    sd = sd),
                na.rm=TRUE) %>%
    pivot_longer(!country) %>%
  pivot_wider(names_from = "country") %>%
  kable(format = "html", digits = 2) %>%
  kable_classic(full_width = F)
```

name	norway	sweden
prof_sami_com_min	0.00	0.00
prof_maj_com_min	1.75	0.25
prof_sami_com_median	0.21	0.00
prof_maj_com_median	4.00	4.00
prof_sami_com_mean	0.66	0.35
prof_maj_com_mean	3.66	3.66
prof_sami_com_max	4.00	4.00
prof_maj_com_max	4.00	4.00
prof_sami_com_sd	1.02	0.73
prof_maj_com_sd	0.51	0.53

```
df_ssq %>%
  group_by(country) %>%
  summarise_at(c("soclad_ind_num", "satdem_num", "fairwealth_num"),
               list(min = min,
                    median = median,
                    mean = mean,
                    max = max,
                    sd = sd),
                na.rm=TRUE) %>%
    pivot_longer(!country) %>%
  pivot_wider(names_from = "country") %>%
  kable(format = "html", digits = 2) %>%
  kable_classic(full_width = F)
```

name	norway	sweden
soclad_ind_num_min	1.00	1.00
satdem_num_min	1.00	1.00
fairwealth_num_min	1.00	1.00
soclad_ind_num_median	6.00	6.00
satdem_num_median	7.00	6.00
fairwealth_num_median	6.00	4.00
soclad_ind_num_mean	6.07	5.87
satdem_num_mean	6.45	5.69
fairwealth_num_mean	5.74	3.92
soclad_ind_num_max	10.00	10.00
satdem_num_max	10.00	10.00
fairwealth_num_max	10.00	10.00

name	norway	sweden
soclad_ind_num_sd	1.58	1.52
satdem_num_sd	2.10	2.34
fairwealth_num_sd	2.05	1.95

Second-stage questionnaire data - Sámi-only

Here we calculate statistics for home use and general use of a Sámi language and the majority language.

```
df_ssq %>%
  filter(sami_bg=="yes") %>%
  group_by(country) %>%
  summarise_at(c("hlang_com", "lit_sami_com", "lit_maj_com"),
               list(min = min,
                    median = median,
                    mean = mean,
                    max = max,
                    sd = sd),
                na.rm=TRUE) %>%
    pivot_longer(!country) %>%
  pivot_wider(names_from = "country") %>%
  kable(format = "html", digits = 2) %>%
  kable_classic(full_width = F)
```

name	norway	sweden
hlang_com_min	1.00	1.00
lit_sami_com_min	0.00	0.00
lit_maj_com_min	0.00	0.84
hlang_com_median	1.00	1.00
lit_sami_com_median	0.45	0.15
lit_maj_com_median	3.74	3.61
hlang_com_mean	1.54	1.34
lit_sami_com_mean	0.91	0.52
lit_maj_com_mean	3.62	3.54
hlang_com_max	5.00	5.00
lit_sami_com_max	3.85	3.69
lit_maj_com_max	4.00	4.00
hlang_com_sd	1.17	0.77
lit_sami_com_sd	1.05	0.81
lit_maj_com_sd	0.47	0.48

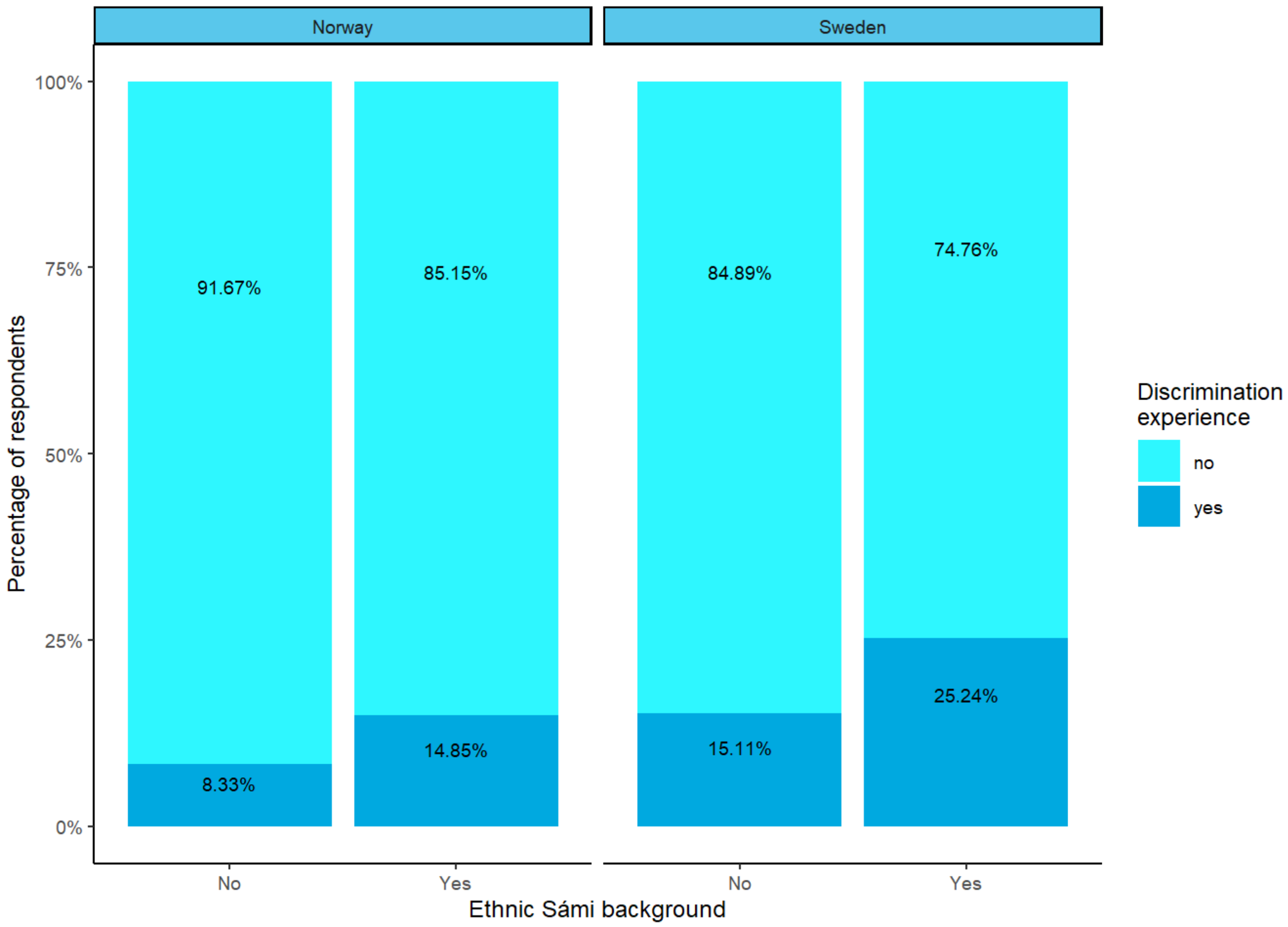
Exploratory analyses

Plots

The lines that export plots to files are included as comments to avoid unnecessary overwriting when the code is run multiple times. To see the files as used in the paper, the comment character (#) should be removed.

Experience of discrimination per country and ehtnic background

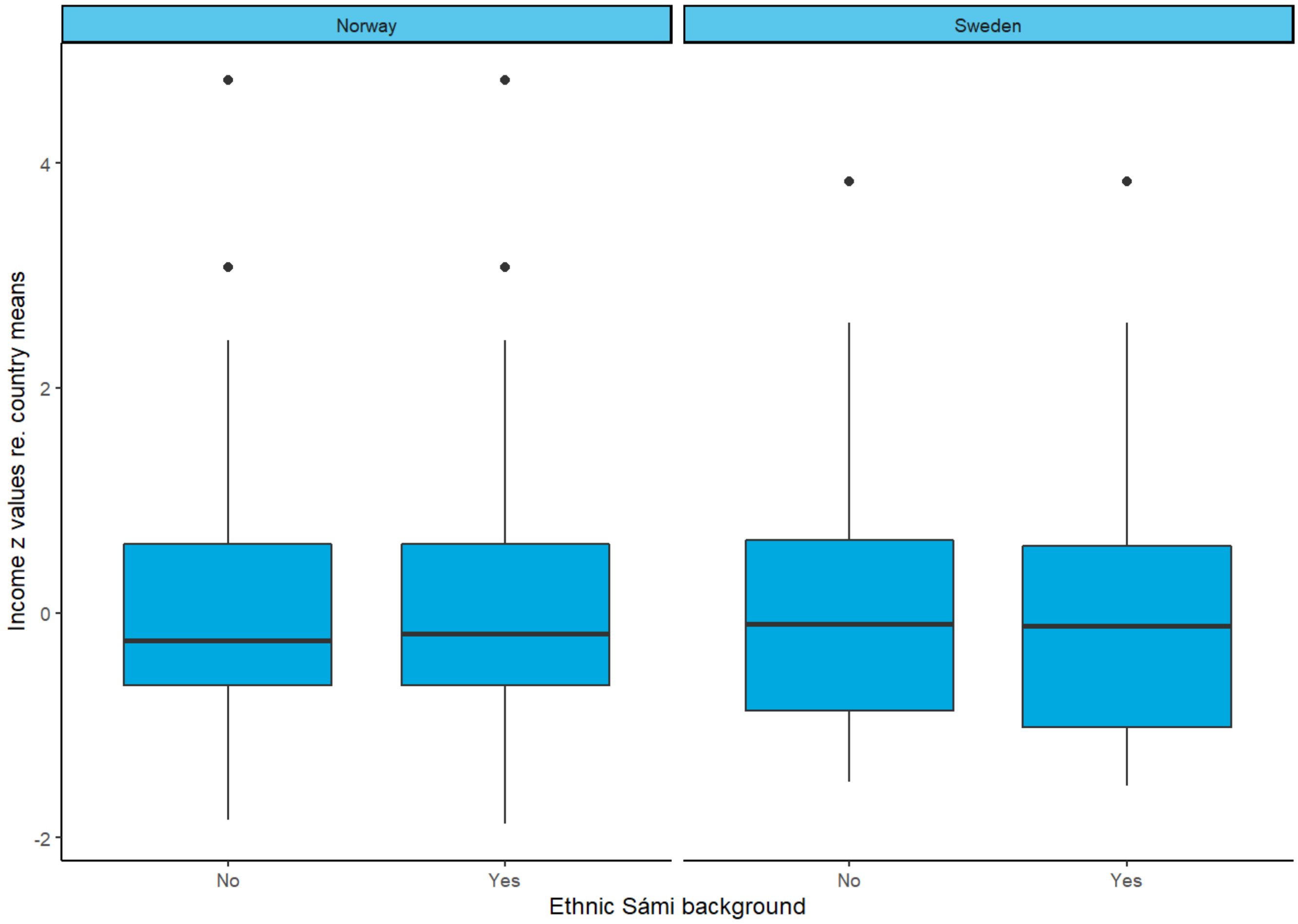
```
plot_disc <- df %>%
  drop_na(disc_exp_cati) %>%
  group_by(country, sami_bg, disc_exp_cati) %>%
  summarise(count = n()) %>%
  mutate(ratio = round(count / sum(count), 4),
         percent = paste0(ratio*100, "%")) %>%
  ggplot(aes(x = sami_bg, y = count, fill = disc_exp_cati)) +
    geom_bar(position = "fill", stat = "identity") +
    geom_text(aes(label = percent), position = position_fill(vjust = 0.7),
              size=2.5) +
    scale_y_continuous(labels = scales::percent) +
    scale_x_discrete(labels=c("No", "Yes")) +
    scale_fill_manual(values = c("#2EF7FF", "#00A9E0")) +
    guides(fill = guide_legend(title = "Discrimination\nexperience")) +
    facet_wrap(~country, labeller = as_labeller(c(`norway` = "Norway",
                                                   `sweden` = "Sweden"))) +
    xlab("Ethnic Sámi background") +
    ylab("Percentage of respondents") +
    theme_classic(base_size = 9) +
    theme(strip.background = element_rect(fill="#59C7EB"))
plot_disc
```




```
#ggsave(filename = "plot_disc.png", plot = plot_disc, device = "png",
#       width = 11, height = 6.67, units = "cm", dpi = 300)
```

Adjusted household income per country and ethnic background

```
plot_income <- df %>%
  ggplot(aes(x = sami_bg, y = hhinc_yg_equiv_std)) +
  geom_boxplot(fill = "#00A9E0") +
  scale_x_discrete(labels=c("No", "Yes")) +
  facet_wrap(~country, labeller = as_labeller(c(`norway` = "Norway",
                                                `sweden` = "Sweden"))) +
  xlab("Ethnic Sámi background") +
  ylab("Income z values re. country means") +
  theme_classic(base_size = 9) +
  theme(strip.background = element_rect(fill="#59C7EB"))
plot_income
```




```
#ggsave(filename = "plot_income.png", plot = plot_income, device = "png",
#       width = 8, height = 6.67, units = "cm", dpi = 300)
```

Use of a Sámi language and the majority language per country, among Sámi respondents.

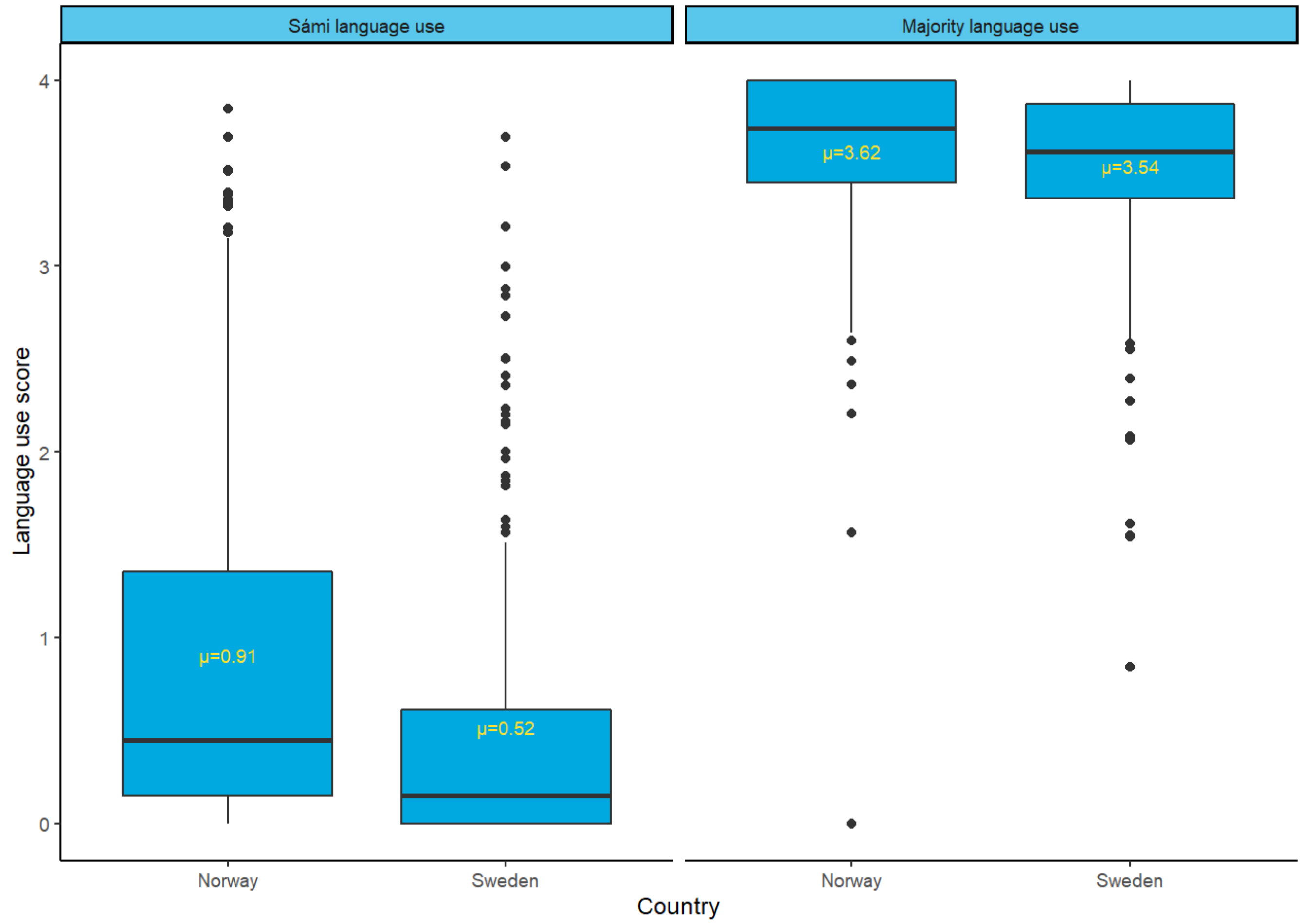
First we bring together the two variables into long data format:

```
sami_use <- df_ssq %>%
  filter(sami_bg=="yes") %>%
  select(country, lit_sami_com) %>%
  mutate(var_type = rep("Sami use", length(country))) %>%
  rename(lang_use = lit_sami_com)
maj_use <- df_ssq %>%
  filter(sami_bg=="yes") %>%
  select(country, lit_maj_com) %>%
  mutate(var_type = rep("Maj. lang. use", length(country))) %>%
  rename(lang_use = lit_maj_com)
lang_use <- bind_rows(sami_use, maj_use) %>%
  mutate(var_type = as_factor(var_type))

means <- lang_use %>%
  group_by(var_type, country) %>%
  summarise(mean = round(mean(lang_use, na.rm = TRUE), 2)) %>%
  mutate(mean_text = paste0("\u03bc", "=", mean))
```

Next, we generate the plot with this dataset:

```
plot_lang <- lang_use %>%
  ggplot(aes(x = country, y = lang_use)) +
  geom_boxplot(fill = "#00A9E0") +
  scale_x_discrete(labels=c("Norway", "Sweden")) +
  geom_text(data=means, aes(label = mean_text, y=mean), colour = "#FFE22E",
            size=2.5) +
  facet_wrap(~var_type,
             labeller = as_labeller(
               c(`Sami use` = "Sámi language use",
                 `Maj. lang. use` = "Majority language use"))) +
  xlab("Country") +
  ylab("Language use score") +
  theme_classic(base_size = 9) +
  theme(strip.background = element_rect(fill="#59C7EB"))
plot_lang
```




```
#ggsave(filename = "plot_lang.png", plot = plot_lang, device = "png",
#       width = 8, height = 6.67, units = "cm", dpi = 300)

rm(maj_use, sami_use, lang_use, means)
```

Multivariate models

Imputations

Missing values are likely to create biased results in multivariate models. This is particularly concerning in the second-stage questionnaire data since the sample size is smaller and a larger number of variables are included in the models. However, large number of variables also mean that missing values can be accurately rectified through imputation. Here we apply imputation using variables relevant for the social status of respondents (income, education, employment, satisfaction with various aspects of life), and complete missing values based on similarity of cases (k nearest neighbour method). We apply imputation only if at least one crucial objective measure (income or education) is non-missing.

```
df_ssq_status <- df_ssq %>%
  select(id, country, hhinc_yg_equiv_std, soclad_ind_num, edulevel,
         hhincome_source, hhincome_sector, employment, satlife_num,
         freechoice_num, satdem_num, fairwealth_num)

df_ssq_status <- df_ssq_status %>%
  filter(!(is.na(hhinc_yg_equiv_std) & is.na(edulevel))) %>%
  impute_knn(hhinc_yg_equiv_std + edulevel ~ . -id -country
             -satdem_num -fairwealth_num  | country) %>%
  impute_knn(satdem_num + fairwealth_num + soclad_ind_num ~ . -id -country | country) %>%
  mutate(hhinc_yg_equiv_std = as.numeric(hhinc_yg_equiv_std),
         satdem_num = as.numeric(satdem_num),
         fairwealth_num = as.numeric(fairwealth_num),
         soclad_ind_num = as.numeric(soclad_ind_num)) %>%
  bind_rows(df_ssq_status %>%
              filter(is.na(hhinc_yg_equiv_std) & is.na(edulevel)))

df_imp_vars <- df_ssq_status %>%
  select(id, hhinc_yg_equiv_std, edulevel, satdem_num, fairwealth_num, soclad_ind_num)

df_ssq <- df_ssq %>%
  left_join(df_imp_vars, by = "id", suffix = c("", "_imp"))

rm(df_ssq_status, df_imp_vars)

df_ssq <- df_ssq %>%
  mutate(
    edulevel_red_imp = case_when(
      edulevel_imp=="compulsory education not completed" |
        edulevel_imp=="compulsory education" |
        edulevel_imp=="upper secondary" ~ "Pre-tertiary" ,
      edulevel_imp=="advanced vocational" ~ "Advanced vocational",
      edulevel_imp=="BA-level" |
        edulevel_imp=="MA-level or higher" ~ "University",
      TRUE ~ NA_character_
    ) %>% as_factor()
  )
```

Full sample

Telephone interview data

Base model:

```
m_ti_1 <- glm(disc_exp_cati ~
                eth_bg_cats + country + hhinc_yg_equiv_std +
                age + age_sq + gender,
              family = binomial(), data = df)
summary(m_ti_1)
```

```
##
## Call:
## glm(formula = disc_exp_cati ~ eth_bg_cats + country + hhinc_yg_equiv_std +
##     age + age_sq + gender, family = binomial(), data = df)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.1062  -0.6172  -0.4525  -0.2983   2.8708
##
## Coefficients:
##                               Estimate Std. Error z value Pr(>|z|)
## (Intercept)                 -1.971e+00  4.295e-01  -4.589 4.45e-06 ***
## eth_bg_catsSami              8.372e-01  1.083e-01   7.730 1.07e-14 ***
## eth_bg_catsImmigrant         6.528e-01  1.728e-01   3.777 0.000158 ***
## eth_bg_catsNational minorities  3.324e-01  1.307e-01   2.543 0.010977 *
## countrysweden                5.995e-01  9.491e-02   6.316 2.68e-10 ***
## hhinc_yg_equiv_std          -1.664e-01  4.655e-02  -3.575 0.000351 ***
## age                          2.068e-02  1.790e-02   1.155 0.248009
## age_sq                      -5.949e-04  1.829e-04  -3.252 0.001146 **
## gendermale                   2.780e-02  8.827e-02   0.315 0.752842
## genderdiverse               -1.123e+01  1.970e+02  -0.057 0.954543
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 3708.9  on 4419  degrees of freedom
## Residual deviance: 3393.3  on 4410  degrees of freedom
##   (996 observations deleted due to missingness)
## AIC: 3413.3
##
## Number of Fisher Scoring iterations: 10
```

Model with interaction term (Income-ethnic background):

```
m_ti_2 <- glm(disc_exp_cati ~
                eth_bg_cats * hhinc_yg_equiv_std + country +
                age + age_sq + gender,
              family = binomial(), data = df)
summary(m_ti_2)
```

```
##
## Call:
## glm(formula = disc_exp_cati ~ eth_bg_cats * hhinc_yg_equiv_std +
##     country + age + age_sq + gender, family = binomial(), data = df)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.1317  -0.6201  -0.4523  -0.3002   2.9206
##
## Coefficients:
##                                               Estimate Std. Error z value
## (Intercept)                                 -1.975e+00  4.301e-01  -4.592
## eth_bg_catsSami                              8.322e-01  1.084e-01   7.674
## eth_bg_catsImmigrant                         6.547e-01  1.737e-01   3.768
## eth_bg_catsNational minorities               3.315e-01  1.307e-01   2.537
## hhinc_yg_equiv_std                          -2.781e-02  8.906e-02  -0.312
## countrysweden                                6.031e-01  9.505e-02   6.345
## age                                          2.088e-02  1.792e-02   1.165
## age_sq                                      -5.974e-04  1.831e-04  -3.263
## gendermale                                   2.875e-02  8.843e-02   0.325
## genderdiverse                               -1.125e+01  1.970e+02  -0.057
## eth_bg_catsSami:hhinc_yg_equiv_std          -2.002e-01  1.131e-01  -1.770
## eth_bg_catsImmigrant:hhinc_yg_equiv_std     -1.097e-01  1.749e-01  -0.627
## eth_bg_catsNational minorities:hhinc_yg_equiv_std -1.913e-01  1.331e-01  -1.437
##                                             Pr(>|z|)
## (Intercept)                                 4.40e-06 ***
## eth_bg_catsSami                             1.66e-14 ***
## eth_bg_catsImmigrant                        0.000164 ***
## eth_bg_catsNational minorities              0.011183 *
## hhinc_yg_equiv_std                          0.754837
## countrysweden                               2.23e-10 ***
## age                                         0.243968
## age_sq                                      0.001103 **
## gendermale                                  0.745102
## genderdiverse                               0.954461
## eth_bg_catsSami:hhinc_yg_equiv_std          0.076692 .
## eth_bg_catsImmigrant:hhinc_yg_equiv_std     0.530490
## eth_bg_catsNational minorities:hhinc_yg_equiv_std 0.150635
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 3708.9  on 4419  degrees of freedom
## Residual deviance: 3389.8  on 4407  degrees of freedom
##   (996 observations deleted due to missingness)
## AIC: 3415.8
##
## Number of Fisher Scoring iterations: 10
```

Model with interaction term (Country-ethnic background):

```
m_ti_3 <- glm(disc_exp_cati ~
                eth_bg_cats * country + hhinc_yg_equiv_std +
                age + age_sq + gender,
              family = binomial(), data = df)
summary(m_ti_3)
```

```
##
## Call:
## glm(formula = disc_exp_cati ~ eth_bg_cats * country + hhinc_yg_equiv_std +
##     age + age_sq + gender, family = binomial(), data = df)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.1118  -0.6240  -0.4527  -0.2930   2.9293
##
## Coefficients:
##                                               Estimate Std. Error z value
## (Intercept)                                 -2.112e+00  4.439e-01  -4.758
## eth_bg_catsSami                              9.871e-01  1.825e-01   5.408
## eth_bg_catsImmigrant                         1.116e+00  2.742e-01   4.070
## eth_bg_catsNational minorities               6.531e-01  2.821e-01   2.315
## countrysweden                                8.334e-01  1.857e-01   4.487
## hhinc_yg_equiv_std                          -1.618e-01  4.661e-02  -3.471
## age                                          2.015e-02  1.791e-02   1.125
## age_sq                                      -5.922e-04  1.830e-04  -3.237
## gendermale                                   2.924e-02  8.832e-02   0.331
## genderdiverse                               -1.124e+01  1.970e+02  -0.057
## eth_bg_catsSami:countrysweden               -2.154e-01  2.280e-01  -0.944
## eth_bg_catsImmigrant:countrysweden          -7.419e-01  3.536e-01  -2.098
## eth_bg_catsNational minorities:countrysweden -4.274e-01  3.187e-01  -1.341
##                                             Pr(>|z|)
## (Intercept)                                 1.96e-06 ***
## eth_bg_catsSami                             6.37e-08 ***
## eth_bg_catsImmigrant                        4.69e-05 ***
## eth_bg_catsNational minorities              0.020607 *
## countrysweden                               7.22e-06 ***
## hhinc_yg_equiv_std                          0.000519 ***
## age                                         0.260750
## age_sq                                      0.001209 **
## gendermale                                  0.740589
## genderdiverse                               0.954489
## eth_bg_catsSami:countrysweden               0.344939
## eth_bg_catsImmigrant:countrysweden          0.035885 *
## eth_bg_catsNational minorities:countrysweden 0.179917
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 3708.9  on 4419  degrees of freedom
## Residual deviance: 3388.3  on 4407  degrees of freedom
##   (996 observations deleted due to missingness)
## AIC: 3414.3
##
## Number of Fisher Scoring iterations: 10
```

Questionnaire data

Model adding language proficiency:

```
m_ssq_1 <- glm(disc_exp_comb ~
                 eth_bg_cats + country +
                 hhinc_yg_equiv_std_imp + prof_sami_com +
                 age + age_sq + gender + edulevel_red_imp,
               family = binomial(), data = df_ssq)
summary(m_ssq_1)
```

```
##
## Call:
## glm(formula = disc_exp_comb ~ eth_bg_cats + country + hhinc_yg_equiv_std_imp +
##     prof_sami_com + age + age_sq + gender + edulevel_red_imp,
##     family = binomial(), data = df_ssq)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.5652  -0.7571  -0.5860  -0.3673   2.2434
##
## Coefficients:
##                                 Estimate Std. Error z value Pr(>|z|)
## (Intercept)                   -4.490e-01  7.243e-01  -0.620  0.53532
## eth_bg_catsSami                4.702e-01  1.723e-01   2.729  0.00635 **
## eth_bg_catsImmigrant          -1.391e-01  3.356e-01  -0.415  0.67849
## eth_bg_catsNational minorities  1.750e-01  1.980e-01   0.884  0.37688
## countrysweden                  6.686e-01  1.668e-01   4.009 6.09e-05 ***
## hhinc_yg_equiv_std_imp        -1.858e-01  7.632e-02  -2.435  0.01491 *
## prof_sami_com                  3.407e-01  7.885e-02   4.321 1.55e-05 ***
## age                           -2.119e-02  2.856e-02  -0.742  0.45816
## age_sq                        -7.893e-05  2.873e-04  -0.275  0.78351
## gendermale                    -6.303e-02  1.405e-01  -0.449  0.65372
## genderdiverse                  1.252e+01  3.247e+02   0.039  0.96924
## edulevel_red_impPre-tertiary   -2.657e-01  1.912e-01  -1.390  0.16464
## edulevel_red_impUniversity     -3.199e-01  1.972e-01  -1.622  0.10470
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1362.6  on 1247  degrees of freedom
## Residual deviance: 1261.1  on 1235  degrees of freedom
##   (172 observations deleted due to missingness)
## AIC: 1287.1
##
## Number of Fisher Scoring iterations: 11
```

Model with subjective variables:

```
m_ssq_2 <- glm(disc_exp_comb ~
                 eth_bg_cats + country +
                 hhinc_yg_equiv_std_imp + prof_sami_com +
                 soclad_ind_num_imp + satdem_num_imp + fairwealth_num_imp +
                 age + age_sq + gender + edulevel_red_imp,
               family = binomial(), data = df_ssq)
summary(m_ssq_2)
```

```
##
## Call:
## glm(formula = disc_exp_comb ~ eth_bg_cats + country + hhinc_yg_equiv_std_imp +
##     prof_sami_com + soclad_ind_num_imp + satdem_num_imp + fairwealth_num_imp +
##     age + age_sq + gender + edulevel_red_imp, family = binomial(),
##     data = df_ssq)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.7233  -0.7377  -0.5291  -0.2808   2.3941
##
## Coefficients:
##                                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)                     1.956e+00  8.166e-01   2.395 0.016604 *
## eth_bg_catsSami                 4.498e-01  1.764e-01   2.549 0.010790 *
## eth_bg_catsImmigrant           -8.873e-02  3.431e-01  -0.259 0.795944
## eth_bg_catsNational minorities  2.418e-01  2.043e-01   1.184 0.236537
## countrysweden                   4.456e-01  1.830e-01   2.435 0.014904 *
## hhinc_yg_equiv_std_imp         -8.344e-02  8.035e-02  -1.038 0.299058
## prof_sami_com                   3.121e-01  8.118e-02   3.845 0.000121 ***
## soclad_ind_num_imp             -1.851e-01  5.250e-02  -3.525 0.000423 ***
## satdem_num_imp                 -1.367e-01  3.612e-02  -3.786 0.000153 ***
## fairwealth_num_imp             -7.745e-02  4.410e-02  -1.756 0.079036 .
## age                            -3.153e-02  2.953e-02  -1.068 0.285695
## age_sq                          6.198e-05  2.975e-04   0.208 0.834984
## gendermale                     -7.337e-02  1.446e-01  -0.507 0.611816
## genderdiverse                   1.204e+01  3.247e+02   0.037 0.970419
## edulevel_red_impPre-tertiary   -3.437e-01  1.979e-01  -1.737 0.082347 .
## edulevel_red_impUniversity     -1.016e-01  2.043e-01  -0.497 0.619005
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1362.6  on 1247  degrees of freedom
## Residual deviance: 1205.3  on 1232  degrees of freedom
##   (172 observations deleted due to missingness)
## AIC: 1237.3
##
## Number of Fisher Scoring iterations: 11
```

Sámi-only sample

First, we create Sámi-only data frames.

```
df_sa <- df %>% filter(sami_bg=="yes")
df_ssq_sa <- df_ssq %>% filter(sami_bg=="yes")
```

Telephone interview data

Base model:

```
m_ti_sa_1 <- glm(disc_exp_cati ~
                   maj_bg + country + hhinc_yg_equiv_std +
                   hlang_sam + hlang_par_sam + age + age_sq + gender,
                 family = binomial(), data = df_sa)
summary(m_ti_sa_1)
```

```
##
## Call:
## glm(formula = disc_exp_cati ~ maj_bg + country + hhinc_yg_equiv_std +
##     hlang_sam + hlang_par_sam + age + age_sq + gender, family = binomial(),
##     data = df_sa)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.6949  -0.6568  -0.4613  -0.2954   2.6473
##
## Coefficients:
##                      Estimate Std. Error z value Pr(>|z|)
## (Intercept)        -6.929e-01  6.958e-01  -0.996  0.31936
## maj_bgyes          -6.012e-01  1.894e-01  -3.174  0.00150 **
## countrysweden       6.363e-01  1.416e-01   4.493 7.02e-06 ***
## hhinc_yg_equiv_std -1.973e-01  7.373e-02  -2.676  0.00746 **
## hlang_samyes        1.090e+00  2.104e-01   5.180 2.22e-07 ***
## hlang_par_samyes    1.124e-01  1.904e-01   0.590  0.55499
## age                 8.830e-03  2.824e-02   0.313  0.75448
## age_sq             -5.041e-04  2.852e-04  -1.768  0.07710 .
## gendermale          1.008e-01  1.390e-01   0.725  0.46823
## genderdiverse      -1.187e+01  3.247e+02  -0.037  0.97085
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1525.9  on 1569  degrees of freedom
## Residual deviance: 1322.6  on 1560  degrees of freedom
##   (349 observations deleted due to missingness)
## AIC: 1342.6
##
## Number of Fisher Scoring iterations: 11
```

Model with interaction term (Country-income):

```
m_ti_sa_2 <- glm(disc_exp_cati ~
                   maj_bg + country * hhinc_yg_equiv_std +
                   hlang_sam +hlang_par_sam +
                   age + age_sq + gender,
                 family = binomial(), data = df_sa)
summary(m_ti_sa_2)
```

```
##
## Call:
## glm(formula = disc_exp_cati ~ maj_bg + country * hhinc_yg_equiv_std +
##     hlang_sam + hlang_par_sam + age + age_sq + gender, family = binomial(),
##     data = df_sa)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.7416  -0.6503  -0.4634  -0.2935   2.6786
##
## Coefficients:
##                                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)                     -6.943e-01  6.969e-01  -0.996  0.31912
## maj_bgyes                       -6.092e-01  1.901e-01  -3.205  0.00135 **
## countrysweden                    6.245e-01  1.420e-01   4.397 1.10e-05 ***
## hhinc_yg_equiv_std              -9.426e-02  1.036e-01  -0.910  0.36272
## hlang_samyes                     1.095e+00  2.108e-01   5.192 2.09e-07 ***
## hlang_par_samyes                 1.066e-01  1.907e-01   0.559  0.57611
## age                              9.842e-03  2.828e-02   0.348  0.72781
## age_sq                          -5.179e-04  2.857e-04  -1.813  0.06983 .
## gendermale                       1.041e-01  1.392e-01   0.748  0.45459
## genderdiverse                   -1.190e+01  3.247e+02  -0.037  0.97077
## countrysweden:hhinc_yg_equiv_std -1.979e-01  1.448e-01  -1.367  0.17157
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1525.9  on 1569  degrees of freedom
## Residual deviance: 1320.8  on 1559  degrees of freedom
##   (349 observations deleted due to missingness)
## AIC: 1342.8
##
## Number of Fisher Scoring iterations: 11
```

Model with interaction term (Country-language):

```
m_ti_sa_3 <- glm(disc_exp_cati ~
                   maj_bg + country * hlang_sam +
                   hlang_par_sam + hhinc_yg_equiv_std +
                   age + age_sq + gender,
                 family = binomial(), data = df_sa)
summary(m_ti_sa_3)
```

```
##
## Call:
## glm(formula = disc_exp_cati ~ maj_bg + country * hlang_sam +
##     hlang_par_sam + hhinc_yg_equiv_std + age + age_sq + gender,
##     family = binomial(), data = df_sa)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.7731  -0.6521  -0.4645  -0.2954   2.6258
##
## Coefficients:
##                             Estimate Std. Error z value Pr(>|z|)
## (Intercept)               -6.359e-01  6.968e-01  -0.913 0.361460
## maj_bgyes                 -6.047e-01  1.907e-01  -3.172 0.001515 **
## countrysweden              5.004e-01  1.690e-01   2.962 0.003061 **
## hlang_samyes               8.893e-01  2.524e-01   3.524 0.000426 ***
## hlang_par_samyes           9.574e-02  1.901e-01   0.504 0.614584
## hhinc_yg_equiv_std        -1.973e-01  7.398e-02  -2.667 0.007649 **
## age                        9.616e-03  2.825e-02   0.340 0.733577
## age_sq                    -5.118e-04  2.854e-04  -1.793 0.072947 .
## gendermale                 1.093e-01  1.394e-01   0.784 0.433149
## genderdiverse             -1.180e+01  3.247e+02  -0.036 0.971011
## countrysweden:hlang_samyes  4.442e-01  3.052e-01   1.455 0.145563
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1525.9  on 1569  degrees of freedom
## Residual deviance: 1320.5  on 1559  degrees of freedom
##   (349 observations deleted due to missingness)
## AIC: 1342.5
##
## Number of Fisher Scoring iterations: 11
```

Questionnaire data

Model with home and general language use:

```
m_ssq_sa_1 <- glm(disc_exp_comb ~
                    maj_bg + country + hhinc_yg_equiv_std_imp +
                    hlang_com + lit_sami_com +
                    age + age_sq + gender + edulevel_red_imp,
                  family = binomial(), data = df_ssq_sa)
summary(m_ssq_sa_1)
```

```
##
## Call:
## glm(formula = disc_exp_comb ~ maj_bg + country + hhinc_yg_equiv_std_imp +
##     hlang_com + lit_sami_com + age + age_sq + gender + edulevel_red_imp,
##     family = binomial(), data = df_ssq_sa)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.0526  -0.7703  -0.5910   0.8986   2.1653
##
## Coefficients:
##                               Estimate Std. Error z value Pr(>|z|)
## (Intercept)                  1.583e+00  1.257e+00   1.259  0.20802
## maj_bgyes                   -1.963e-01  3.664e-01  -0.536  0.59214
## countrysweden                5.716e-01  2.544e-01   2.247  0.02465 *
## hhinc_yg_equiv_std_imp      -2.426e-01  1.302e-01  -1.864  0.06235 .
## hlang_com                    9.664e-02  1.528e-01   0.633  0.52697
## lit_sami_com                 4.601e-01  1.537e-01   2.994  0.00276 **
## age                         -9.252e-02  4.601e-02  -2.011  0.04434 *
## age_sq                       5.502e-04  4.629e-04   1.189  0.23458
## gendermale                   5.881e-02  2.321e-01   0.253  0.79998
## genderdiverse                1.298e+01  5.354e+02   0.024  0.98066
## edulevel_red_impPre-tertiary  2.393e-02  3.386e-01   0.071  0.94364
## edulevel_red_impUniversity   -7.334e-02  3.409e-01  -0.215  0.82968
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 529.92  on 438  degrees of freedom
## Residual deviance: 463.73  on 427  degrees of freedom
##   (99 observations deleted due to missingness)
## AIC: 487.73
##
## Number of Fisher Scoring iterations: 12
```

Model with interaction term (Country-language use):

```
m_ssq_sa_2 <- glm(disc_exp_comb ~
                    maj_bg + country * lit_sami_com +
                    hhinc_yg_equiv_std_imp +
                    age + age_sq + gender + edulevel_red_imp,
                  family = binomial(), data = df_ssq_sa)
summary(m_ssq_sa_2)
```

```
##
## Call:
## glm(formula = disc_exp_comb ~ maj_bg + country * lit_sami_com +
##     hhinc_yg_equiv_std_imp + age + age_sq + gender + edulevel_red_imp,
##     family = binomial(), data = df_ssq_sa)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.6762  -0.7629  -0.5738   0.8424   2.0687
##
## Coefficients:
##                               Estimate Std. Error z value Pr(>|z|)
## (Intercept)                  1.508e+00  1.247e+00   1.210  0.22632
## maj_bgyes                   -2.106e-01  3.412e-01  -0.617  0.53700
## countrysweden                7.930e-02  3.109e-01   0.255  0.79868
## lit_sami_com                 2.679e-01  1.606e-01   1.668  0.09532 .
## hhinc_yg_equiv_std_imp      -2.159e-01  1.300e-01  -1.661  0.09667 .
## age                         -7.190e-02  4.646e-02  -1.547  0.12175
## age_sq                       3.246e-04  4.705e-04   0.690  0.49028
## gendermale                   1.691e-01  2.325e-01   0.727  0.46705
## genderdiverse                1.297e+01  5.354e+02   0.024  0.98067
## edulevel_red_impPre-tertiary  4.230e-02  3.368e-01   0.126  0.90006
## edulevel_red_impUniversity   -8.612e-02  3.380e-01  -0.255  0.79889
## countrysweden:lit_sami_com    7.054e-01  2.578e-01   2.736  0.00622 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 541.81  on 445  degrees of freedom
## Residual deviance: 467.50  on 434  degrees of freedom
##   (92 observations deleted due to missingness)
## AIC: 491.5
##
## Number of Fisher Scoring iterations: 12
```

Exporting regression tables

Here we write a function that turns the model outputs into html tables with essential information, including odds ratios.

```
beautify <- function(m_obj, tbl_cap) {
  m_tbl <- tidy(m_obj) %>%
  mutate(Coefficient = round(estimate, 2),
         OR = round(exp(estimate), 2),
         SE = round(std.error, 2),
         "p-value" = ifelse(p.value<0.01, "<0.01",
                            as.character(round(p.value, 2)))) %>%
  select(-c("estimate", "std.error", "statistic", "p.value")) %>%
  kable(format = "html",
        caption = tbl_cap) %>%
  kable_classic(full_width = F) %>%
  add_footnote(paste0("N = ", length(m_obj$y),
                      "; Null dev. = ", round(m_obj$null.deviance),
                      " (on ", m_obj$df.null, " df)",
                      "; Res. dev. = ", round(m_obj$deviance),
                      " (on ", m_obj$df.residual, " df)",
                      "; AIC = ", round(m_obj$aic),
                      "; Log-likelihood = ", round(logLik(m_obj))),
               notation = "none")
  return(m_tbl)
}
```

We generate tables with this function

```
beautify(m_ti_1, "Logit regression model from CATI data.
                  Outcome variable: Discrimination experienced")
```

Logit regression model from CATI data. Outcome variable: Discrimination experienced

term	Coefficient	OR	SE	p-value
(Intercept)	-1.97	0.14	0.43	<0.01
eth_bg_catsSami	0.84	2.31	0.11	<0.01
eth_bg_catsImmigrant	0.65	1.92	0.17	<0.01
eth_bg_catsNational minorities	0.33	1.39	0.13	0.01
countrysweden	0.60	1.82	0.09	<0.01
hhinc_yg_equiv_std	-0.17	0.85	0.05	<0.01
age	0.02	1.02	0.02	0.25
age_sq	0.00	1.00	0.00	<0.01
gendermale	0.03	1.03	0.09	0.75
genderdiverse	-11.23	0.00	196.97	0.95

N = 4420; Null dev. = 3709 (on 4419 df); Res. dev. = 3393 (on 4410 df); AIC = 3413; Log-likelihood = -1697

```
beautify(m_ti_2, "Logit regression model from CATI data.
                   Outcome variable: Discrimination experienced")
```

Logit regression model from CATI data. Outcome variable: Discrimination experienced

term	Coefficient	OR	SE	p-value
(Intercept)	-1.97	0.14	0.43	<0.01
eth_bg_catsSami	0.83	2.30	0.11	<0.01
eth_bg_catsImmigrant	0.65	1.92	0.17	<0.01
eth_bg_catsNational minorities	0.33	1.39	0.13	0.01

N = 4420; Null dev. = 3709 (on 4419 df); Res. dev. = 3390 (on 4407 df); AIC = 3416; Log-likelihood = -1695

term	Coefficient	OR	SE	p-value
hhinc_yg_equiv_std	-0.03	0.97	0.09	0.75
countrysweden	0.60	1.83	0.10	<0.01
age	0.02	1.02	0.02	0.24
age_sq	0.00	1.00	0.00	<0.01
gendermale	0.03	1.03	0.09	0.75
genderdiverse	-11.25	0.00	196.97	0.95
eth_bg_catsSami:hhinc_yg_equiv_std	-0.20	0.82	0.11	0.08
eth_bg_catsImmigrant:hhinc_yg_equiv_std	-0.11	0.90	0.17	0.53
eth_bg_catsNational minorities:hhinc_yg_equiv_std	-0.19	0.83	0.13	0.15

N = 4420; Null dev. = 3709 (on 4419 df); Res. dev. = 3390 (on 4407 df); AIC = 3416; Log-likelihood = -1695

```
beautify(m_ti_3, "Logit regression model from CATI data.
                   Outcome variable: Discrimination experienced")
```

Logit regression model from CATI data. Outcome variable: Discrimination experienced

term	Coefficient	OR	SE	p-value
(Intercept)	-2.11	0.12	0.44	<0.01
eth_bg_catsSami	0.99	2.68	0.18	<0.01
eth_bg_catsImmigrant	1.12	3.05	0.27	<0.01
eth_bg_catsNational minorities	0.65	1.92	0.28	0.02
countrysweden	0.83	2.30	0.19	<0.01
hhinc_yg_equiv_std	-0.16	0.85	0.05	<0.01
age	0.02	1.02	0.02	0.26
age_sq	0.00	1.00	0.00	<0.01
gendermale	0.03	1.03	0.09	0.74
genderdiverse	-11.24	0.00	196.97	0.95
eth_bg_catsSami:countrysweden	-0.22	0.81	0.23	0.34
eth_bg_catsImmigrant:countrysweden	-0.74	0.48	0.35	0.04
eth_bg_catsNational minorities:countrysweden	-0.43	0.65	0.32	0.18

N = 4420; Null dev. = 3709 (on 4419 df); Res. dev. = 3388 (on 4407 df); AIC = 3414; Log-likelihood = -1694

```
beautify(m_ssq_1, "Logit regression model from questionnaire data.
                   Outcome variable: Discrimination experienced")
```

Logit regression model from questionnaire data. Outcome variable: Discrimination experienced

term	Coefficient	OR	SE	p-value
(Intercept)	-0.45	0.64	0.72	0.54
eth_bg_catsSami	0.47	1.60	0.17	<0.01
eth_bg_catsImmigrant	-0.14	0.87	0.34	0.68
eth_bg_catsNational minorities	0.18	1.19	0.20	0.38
countrysweden	0.67	1.95	0.17	<0.01
hhinc_yg_equiv_std_imp	-0.19	0.83	0.08	0.01
prof_sami_com	0.34	1.41	0.08	<0.01
age	-0.02	0.98	0.03	0.46
age_sq	0.00	1.00	0.00	0.78
gendermale	-0.06	0.94	0.14	0.65
genderdiverse	12.52	273959.04	324.74	0.97
edulevel_red_impPre-tertiary	-0.27	0.77	0.19	0.16

N = 1248; Null dev. = 1363 (on 1247 df); Res. dev. = 1261 (on 1235 df); AIC = 1287; Log-likelihood = -631

term	Coefficient	OR	SE	p-value
edulevel_red_impUniversity	-0.32	0.73	0.20	0.1

N = 1248; Null dev. = 1363 (on 1247 df); Res. dev. = 1261 (on 1235 df); AIC = 1287; Log-likelihood = -631

```
beautify(m_ssq_2, "Logit regression model from questionnaire data.
                Outcome variable: Discrimination experienced")
```

Logit regression model from questionnaire data. Outcome variable: Discrimination experienced

term	Coefficient	OR	SE	p-value
(Intercept)	1.96	7.07	0.82	0.02
eth_bg_catsSami	0.45	1.57	0.18	0.01
eth_bg_catsImmigrant	-0.09	0.92	0.34	0.8
eth_bg_catsNational minorities	0.24	1.27	0.20	0.24
countrysweden	0.45	1.56	0.18	0.01
hhinc_yg_equiv_std_imp	-0.08	0.92	0.08	0.3
prof_sami_com	0.31	1.37	0.08	<0.01
soclad_ind_num_imp	-0.19	0.83	0.05	<0.01
satdem_num_imp	-0.14	0.87	0.04	<0.01
fairwealth_num_imp	-0.08	0.93	0.04	0.08
age	-0.03	0.97	0.03	0.29
age_sq	0.00	1.00	0.00	0.83
gendermale	-0.07	0.93	0.14	0.61
genderdiverse	12.04	169809.34	324.74	0.97
edulevel_red_impPre-tertiary	-0.34	0.71	0.20	0.08
edulevel_red_impUniversity	-0.10	0.90	0.20	0.62

N = 1248; Null dev. = 1363 (on 1247 df); Res. dev. = 1205 (on 1232 df); AIC = 1237; Log-likelihood = -603

```
beautify(m_ti_sa_1, "Logit regression model from Sámi-only CATI data.
                  Outcome variable: Discrimination experienced")
```

Logit regression model from Sámi-only CATI data. Outcome variable: Discrimination experienced

term	Coefficient	OR	SE	p-value
(Intercept)	-0.69	0.50	0.70	0.32
maj_bgyes	-0.60	0.55	0.19	<0.01
countrysweden	0.64	1.89	0.14	<0.01
hhinc_yg_equiv_std	-0.20	0.82	0.07	<0.01
hlang_samyes	1.09	2.97	0.21	<0.01
hlang_par_samyes	0.11	1.12	0.19	0.55
age	0.01	1.01	0.03	0.75
age_sq	0.00	1.00	0.00	0.08
gendermale	0.10	1.11	0.14	0.47
genderdiverse	-11.87	0.00	324.74	0.97

N = 1570; Null dev. = 1526 (on 1569 df); Res. dev. = 1323 (on 1560 df); AIC = 1343; Log-likelihood = -661

```
beautify(m_ti_sa_2, "Logit regression model from Sámi-only CATI data.
                  Outcome variable: Discrimination experienced")
```

Logit regression model from Sámi-only CATI data. Outcome variable: Discrimination experienced

term	Coefficient	OR	SE	p-value

term	Coefficient	OR	SE	p-value
(Intercept)	-0.69	0.50	0.70	0.32
maj_bgyes	-0.61	0.54	0.19	<0.01
countrysweden	0.62	1.87	0.14	<0.01
hhinc_yg_equiv_std	-0.09	0.91	0.10	0.36
hlang_samyes	1.09	2.99	0.21	<0.01
hlang_par_samyes	0.11	1.11	0.19	0.58
age	0.01	1.01	0.03	0.73
age_sq	0.00	1.00	0.00	0.07
gendermale	0.10	1.11	0.14	0.45
genderdiverse	-11.90	0.00	324.74	0.97
countrysweden:hhinc_yg_equiv_std	-0.20	0.82	0.14	0.17

N = 1570; Null dev. = 1526 (on 1569 df); Res. dev. = 1321 (on 1559 df); AIC = 1343; Log-likelihood = -660

```
beautify(m_ti_sa_3, "Logit regression model from Sámi-only CATI data.
                    Outcome variable: Discrimination experienced")
```

Logit regression model from Sámi-only CATI data. Outcome variable: Discrimination experienced

term	Coefficient	OR	SE	p-value
(Intercept)	-0.64	0.53	0.70	0.36
maj_bgyes	-0.60	0.55	0.19	<0.01
countrysweden	0.50	1.65	0.17	<0.01
hlang_samyes	0.89	2.43	0.25	<0.01
hlang_par_samyes	0.10	1.10	0.19	0.61
hhinc_yg_equiv_std	-0.20	0.82	0.07	<0.01
age	0.01	1.01	0.03	0.73
age_sq	0.00	1.00	0.00	0.07
gendermale	0.11	1.12	0.14	0.43
genderdiverse	-11.80	0.00	324.74	0.97
countrysweden:hlang_samyes	0.44	1.56	0.31	0.15

N = 1570; Null dev. = 1526 (on 1569 df); Res. dev. = 1321 (on 1559 df); AIC = 1343; Log-likelihood = -660

```
beautify(m_ssq_sa_1, "Logit regression model from Sámi-only questionnaire data.
                     Outcome variable: Discrimination experienced")
```

Logit regression model from Sámi-only questionnaire data. Outcome variable: Discrimination experienced

term	Coefficient	OR	SE	p-value
(Intercept)	1.58	4.87	1.26	0.21
maj_bgyes	-0.20	0.82	0.37	0.59
countrysweden	0.57	1.77	0.25	0.02
hhinc_yg_equiv_std_imp	-0.24	0.78	0.13	0.06
hlang_com	0.10	1.10	0.15	0.53
lit_sami_com	0.46	1.58	0.15	<0.01
age	-0.09	0.91	0.05	0.04
age_sq	0.00	1.00	0.00	0.23
gendermale	0.06	1.06	0.23	0.8
genderdiverse	12.98	432950.22	535.41	0.98
edulevel_red_impPre-tertiary	0.02	1.02	0.34	0.94

N = 439; Null dev. = 530 (on 438 df); Res. dev. = 464 (on 427 df); AIC = 488; Log-likelihood = -232

term	Coefficient	OR	SE	p-value
edulevel_red_impUniversity	-0.07	0.93	0.34	0.83
N = 439; Null dev. = 530 (on 438 df); Res. dev. = 464 (on 427 df); AIC = 488; Log-likelihood = -232				

```
beautify(m_ssq_sa_2, "Logit regression model from Sámi-only questionnaire data.
                     Outcome variable: Discrimination experienced")
```

Logit regression model from Sámi-only questionnaire data. Outcome variable: Discrimination experienced

term	Coefficient	OR	SE	p-value
(Intercept)	1.51	4.52	1.25	0.23
maj_bgyes	-0.21	0.81	0.34	0.54
countrysweden	0.08	1.08	0.31	0.8
lit_sami_com	0.27	1.31	0.16	0.1
hhinc_yg_equiv_std_imp	-0.22	0.81	0.13	0.1
age	-0.07	0.93	0.05	0.12
age_sq	0.00	1.00	0.00	0.49
gendermale	0.17	1.18	0.23	0.47
genderdiverse	12.97	429862.43	535.41	0.98
edulevel_red_impPre-tertiary	0.04	1.04	0.34	0.9
edulevel_red_impUniversity	-0.09	0.92	0.34	0.8
countrysweden:lit_sami_com	0.71	2.02	0.26	<0.01
N = 446; Null dev. = 542 (on 445 df); Res. dev. = 468 (on 434 df); AIC = 492; Log-likelihood = -234				