FinalReport

I. INTRODUCTION

1.1 The European Social Survey According to the official website of the European Social Survey (ESS), the ESS is a survey conducted academically across Europe every two-year on a cross-national basis since 2001 through face-to-face interviews. Variables measured in this survey include the attitudes, beliefs and behaviour patterns of diverse populations. Its three main aims are, to monitor and interpret changing public attitudes and values within Europe and to investigate how they interact with Europe's changing institutions, to advance and consolidate improved methods of cross-national survey measurement in Europe and beyond, and to develop a series of European social indicators, including attitudinal indicators.

Until now, the ESS has had ten rounds, the first round conducted in 2002 and the tenth round in 2020. For analysis in this report, we will use data from the ninth round. In the ninth round, the survey covers 30 countries and employs the most rigorous methodologies funded by the Members, Observers and Guests of the ESS European Research Infrastructure Consortium (ESS ERIC) who represent national governments.

The survey involves strict random probability sampling with a minimum target response rate of 70% and rigorous translation protocols. The hour-long face-to-face interview includes questions on a variety of core topics repeated from previous rounds of the survey and also two modules developed for Round 9 covering Justice and Fairness in Europe, and the Timing of Life (the latter is a partial repeat of a module from Round 3).

The scope of this survey is all persons aged 15 and over resident within private households, regardless of their nationality, citizenship, language or legal status, in the listed countries conducted from August 30, 2018, to January 27, 2020. The listed countries in the ninth round are Albania, Austria, Belgium, Bulgaria, Croatia, Cyprus, Czechia, Denmark, Estonia, Finland, France, Germany, Germany, Hungary, Iceland, Ireland, Italy, Latvia, Lithuania, Montenegro, Netherlands, Norway, Poland, Portugal, Romania, Serbia, Slovakia, Slovenia, Spain, Sweden, Sweden, Switzerland, United Kingdom.

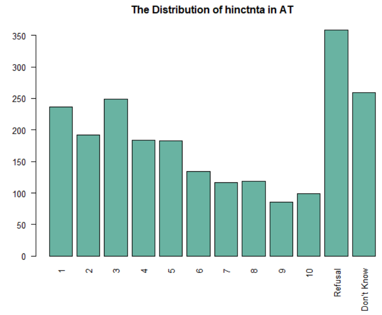
By exploring the documentation file, we found that in the ninth round, there are four weights in this survey. Design Weights The purpose of the design weights (DWEIGHT) is to correct for unequal probabilities for selection due to the sampling design used. In general design weights were computed for each country as follows. w = 1/(PROB1…PROBk) is a nx1 vector of weights; k depends on the number of stages of the sampling design. All weights were rescaled in a way that the sum of the final weights equals n, i.e. rescaled weights = n*w/sum(w). It is not recommended to use this weight without non-response correction. Post-stratification Weights The purpose of the post-stratified design weights (PSPWGHT) is to reduce sampling error, non-coverage, and non-response bias, using auxiliary information specified by the sampling design. The post-stratification targets use information about age, gender, education and region. Raking (iterative proportional fitting) has been used in the production of the post-stratified weights. It also takes into account differences in population size across countries. Analysis Weights The analysis weight (ANWEIGHT) corrects for population size when combining two or more countries' data, and is calculated as ANWEIGHT=PSPWGHT*PWEIGHT. This is a weight in all analyses, it is constructed by first deriving the design weight, then applying a post-stratification adjustment, and then a population size adjustment. Population Weights The Population size weight (PWEIGHT) corrects for population size when combining two or more country's data, and is calculated as PWEIGHT=[Population aged 15 years and over]/[(Net sample in data file)*10 000]

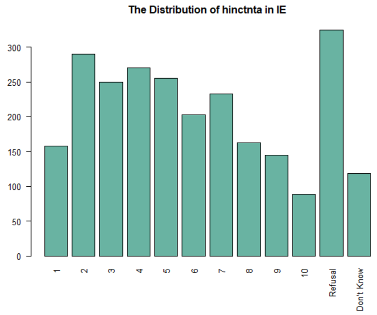
1.2 Objective The purpose of this report is to impute missing values in the variable HINCTNTA. This variable is household income in deciles. The categories in variable HINCTNTA are national and based on deciles of the actual household income range in the given country. These deciles are derived from different sources. The median income is the reference point and the 10 deciles are calculated with the median itself at the top of the fifth decile (category F).

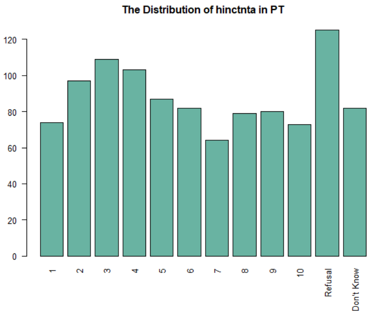
II. MISSING DATA CHALLENGE

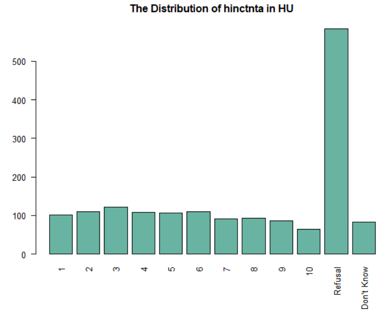
2.1 Select five countries Here, we limit to five countries only to impute missing values. The five countries are selected randomly. The selected countries are Austria, Ireland, Portugal, Hungary, and Poland. These countries have different sources. In Austria and Portugal, it refers to annual household income with a lower limit is €15,300 €5,636 and an upper limit is €77,500 and €35,092, respectively. In Ireland, it refers to weekly household income with an upper limit of €1,680 and a lower limit of €270. Differently, in Hungary and Poland, household income refers to monthly income with a lower limit and an upper limit in Hungary Ft130,000 and Ft410,000 and zł1,700 and zł8,801 in Poland, respectively.

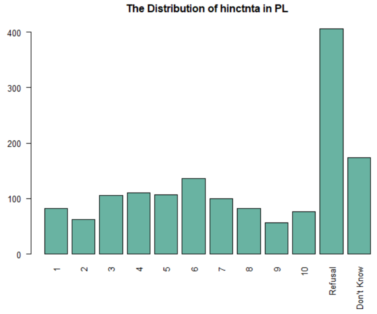
The missing values themselves have three different types, which are refusal, don't know, and no answer and coded differently. Here are the bar charts of five countries to show the frequency of missing values and each decile group.











2.2 Missing values imputation In order to fix missing values across countries, we decided to impute every country separately for some reasons. First, the variable HINCTNTA itself is differently distributed for every country, so simultaneous imputation would not make sense. This is also in line with other studies (Plumpton et al., 2016; Sintonen et al., 2016; Dorsch & Maarek, 2019; Weber & Denk, 2011; Landrum & Becker, 2001). Another thing is, not only is HINCTNTA differently distributed but the variable is divided into deciles, as can be seen in the plots, meaning that the range also differs for every country.

As we know, the income variable is a continuous variable. However, what we have here is in an ordinal scale because it is grouped as deciles. Since the incomes are reported as deciles rather than the raw values, where each decile contains 10% of incomes in a country, this may yield some difficulties when imputing the variable. One option would be to treat the variables as nominal and then use polynomial regression to impute the income classes, but this would ignore the ordering of the variable and thus come with more difficulties.

Ryder et al. (2011) recommend using the midpoint for each income class as a surrogate to be used for imputation so that the variable can be treated as a continuous variable. For instance, in an income class indicating €10000 - €16000, €13000 will be used as a surrogate.

If we then use predictive mean matching as an imputation method, only observed values are used as possible imputed values, and thus these imputed values can again be transformed to the established income classes after imputation. Predictive mean matching is a hot deck method that calculates the predicted value of the target variable based on the specified imputation model. The method establishes a small set of potential donors for each missing data from all complete cases that has the closest predicted value to the predicted value for the missing data then a random donor is taken from the candidate to replace the missing value assuming the missing data and observed data have the same distribution (van Buuren, 2018).

In the previous section, we also mentioned that there are weights included as variables in this survey. Based on previous research, weights are included to fix missing values. Quartagno et al. (2020) used an imputation model where the weights are included as additional variables. Kim et al. (2006) and Seaman et al. (2012) suggested a better imputation model should include not only the weights but also all interactions between weights and covariates. This can be done easily when missing data are confined to the outcome variable—but not when data are missing in all variables. Andridge and Little (2009) used the sampling weight as a stratifying variable alongside additional adjustment variables when forming adjustment cells (hot deck imputations).

From the four kinds of weights, we only use analysis weight (ANWEIGHT) as a predictor variable because this is a weight in all analyses and can correct population size when combining more than one country. In addition to the analysis weight, we will use ten variables and interactions between those variables with the weight. The variables used are chosen based on the previous studies and rational reasoning.

1. PDWRK: partner doing paid work last 7 days
   - Household income meaning income from for all workers in a household. We assume that if respondent's partner is an active worker within the last 7 days then it will influence the total household income.
2. BTHCLD: ever given birth to/ fathered a child
   - According to Kolk (2021), fertility for both men and women groups have positive relationship. It mentioned that men and women with two or more children have higher income than people with one or no child.
3. GNDR: gender of respondents
   - Based on data from International Monetary Fund (2015), more men work than women in most countries and they get paid more for similar work. Therefore, respondent's gender obviously has relationship with the household income.
4. MARITALB: legal marital status
   - Ideally, legal marital status has relationship with the household income. It is inline with study from Balcazar (2019) that mentioned married individuals have the highest incomes level out of all groups (single, married, divorce, separated, never married). Moreover, unmarried couple in some countries counted as separate households.
5. LRSCALE: placement on left right scale
   - We take self-placement of Left-Right into consideration because household income is a significant predictor of respondent's Left-Right self-placement, controlling all other variables (Esposito & Theuerkauf, 2021). It also mentioned a positive sign of income indicates that one's perception of family prosperity is related to one's placement on the right side of the ceteris paribus scale.
6. DSCRGRP: member of a group discriminated against in this country
7. HHMMB: number of people living regularly as member of household
8. AGEA: age of respondents
9. WKHTOT: total hours normally worked per week in main job overtime included
10. EISCED: highest level of education

(PDWRK), (BTHCLD), (GNDR), (MARITALB), (LRSCALE), (DSCRGRP), (HHMMB), (AGEA), (WKHTOT), (EISCED).

III. Methodology and Results

3.1 Input data and packages

```
#devtools::install_github("amices/ggmice")
library(tidyverse)
library(mice)
library(ggmice)
library(psych)
library(visdat)

#Input Data
ess <- readRDS("Ess round 9.RDS")
```

3.2 Data processing

In order to obtain the data of the 5 chosen countries, we have to divide the original data.

```
#Find the column full of NAs
findNACol <- function(data){
  ind_vec <- c()
  j <- 1
  for (i in 1 : length(data[1, ])) {
    if(sum(is.na(data[, i])) == length(data[, i])){
      ind_vec[j] <- i
      j <- j + 1
    }
  }
  return(ind_vec)
}
```

```
#Cutting the whole dataset by countries and get rid of NA columns
cutd <- function(data = ess){
  cntrynames <- names(table(data$cntry))
  num_cntry <- length(cntrynames)
  cntrydata_list <- list()
  for (k in 1 : num_cntry) {
    cntry <- filter(data, cntry == cntrynames[k])
    index <- findNACol(cntry)
    processed <- cntry[, -index]

    cntrydata_list[[k]] <- processed
  }

  names(cntrydata_list) <- cntrynames
  return(cntrydata_list)
}

cntrydatalist <- cutd(ess)
```

Next, rename the data of each countries that we chose

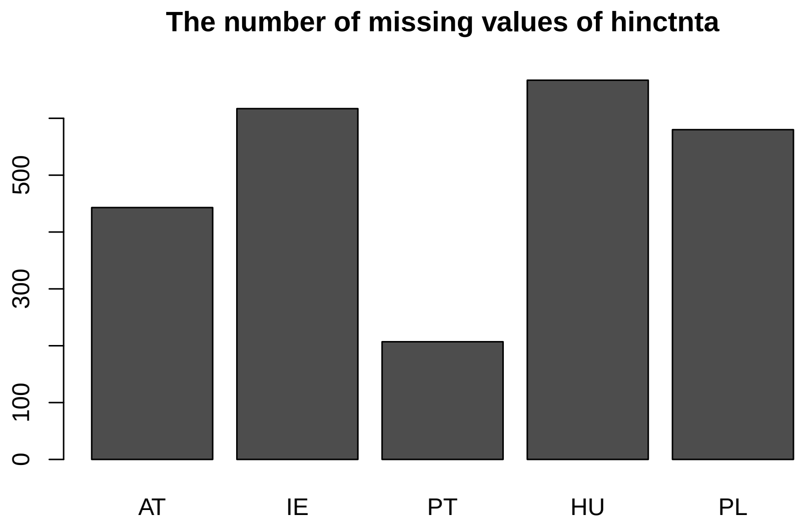
```
AT <- cntrydatalist$AT
IE <- cntrydatalist$IE
PT <- cntrydatalist$PT
```

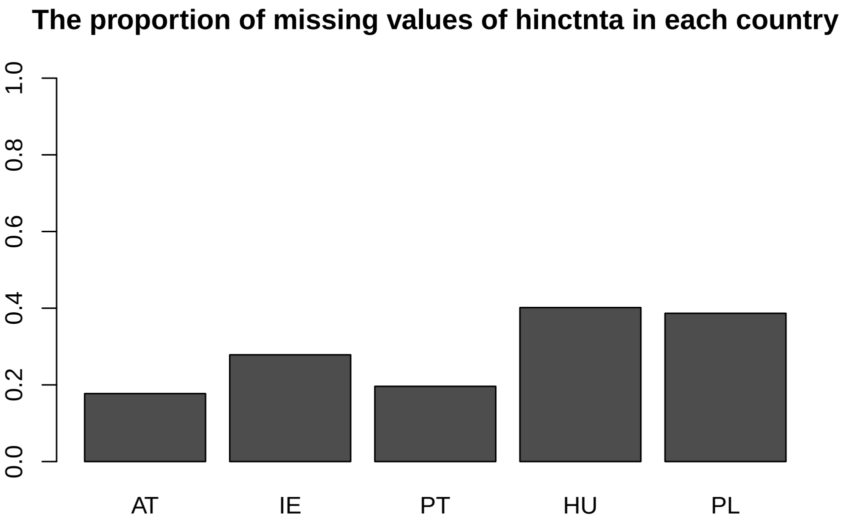
```
HU <- cntrydatalist$HU
PL <- cntrydatalist$PL
```

Replace the specific answers as NA

```
AT$hinctnta[AT$hinctnta == 88] <- NA
AT$hinctnta[AT$hinctnta == 77] <- NA

IE$hinctnta[IE$hinctnta == 88] <- NA
IE$hinctnta[IE$hinctnta == 77] <- NA

PT$hinctnta[PT$hinctnta == 88] <- NA
PT$hinctnta[PT$hinctnta == 77] <- NA

HU$hinctnta[HU$hinctnta == 88] <- NA
HU$hinctnta[HU$hinctnta == 77] <- NA

PL$hinctnta[PL$hinctnta == 88] <- NA
PL$hinctnta[PL$hinctnta == 77] <- NA
```

And then we can take a brief look at the patterns of missing data





Missing data patterns are not shown as they are uninformative.

To solve missingness in the income variable (hinctnta), we will use multiple imputation. However, since this variable is reported in deciles, imputation will not be straightforward. Ryder et al. (2011) recommend using the midpoint for each class as a surrogate to use for imputation. Furthermore, Donnelly and Pop-Eleches (2018) recommend to use the lower bound of the 10th category plus the width of category 9 as a surrogate for the highest decile. Since deciles differ across countries, this will be done seperately for each country.

3.3 Create decile objects and join for imputation and change the levels of hinctnta to be the median (the middle of the category) of the deciles

```
# Austria
AT_deciles <- cbind(1:10, c(7650, 18200, 23400, 28350, 34050
                            , 40150, 47300, 56000, 69050, 94400)) %>%
  as.data.frame()
colnames(AT_deciles) <- c("hinctnta", "income")
AT_deciles$income <- as.numeric(AT_deciles$income) # make numeric

AT <- AT %>% left_join(AT_deciles, by = "hinctnta") # add income surrogate

# Ireland
IE_deciles <- cbind(1:10, c(135, 327.50, 447.5, 572.5, 710
                            , 857.5, 1022.5, 1227.5, 1510, 2020)) %>%
  as.data.frame()
colnames(IE_deciles) <- c("hinctnta", "income")
IE_deciles$income <- as.numeric(IE_deciles$income) # make numeric

IE <- IE %>% left_join(IE_deciles, by = "hinctnta") # add income surrogate

# Hungary
HU_deciles <- cbind(1:10, c(6500, 149500, 184500, 214500, 244500
                            , 274500, 304500, 339500, 384500, 450000)) %>%
  as.data.frame()
```

```
colnames(HU_deciles) <- c("hinctnta", "income")
HU_deciles$income <- as.numeric(HU_deciles$income) # make numeric

HU <- HU %>% left_join(HU_deciles, by = "hinctnta") # add income surrogate

# Portugal
PT_deciles <- cbind(1:10, c(2818, 6709, 8847.5, 11265, 13885, 16556.5
                          , 19728, 23948, 30566.5, 44143)) %>%
  as.data.frame()
colnames(PT_deciles) <- c("hinctnta", "income")
PT_deciles$income <- as.numeric(PT_deciles$income) # make numeric

PT <- PT %>% left_join(PT_deciles, by = "hinctnta") # add income surrogate

# Poland
PL_deciles <- cbind(1:10, c(850, 2000.5, 3650.5, 3300.5, 3950.5, 4650.5
                          , 5450.5, 6450.5, 7900.5, 10600)) %>%
  as.data.frame()
colnames(PL_deciles) <- c("hinctnta", "income")
PL_deciles$income <- as.numeric(PL_deciles$income) # make numeric

PL <- PL %>% left_join(PL_deciles, by = "hinctnta") # add income surrogate
```

Recode all relevant variables used for imputation model (missingness and variable levels)

```
# Clean important variables chosen for the imputation model
# Define missing values and recode variables for the model

# Austria
AT$eisced[AT$eisced == 55] <- NA
AT$eisced <- factor(AT$eisced, levels = c("1", "2", "3", "4", "5", "6", "7"),
                    ordered = T)
AT$bthcld[AT$bthcld == 1] <- 0
AT$bthcld[AT$bthcld == 2] <- 1
AT$dscrgrp[AT$dscrgrp == 1] <- 0
AT$dscrgrp[AT$dscrgrp == 2] <- 1

AT$bthcld[AT$bthcld != 0 & AT$bthcld != 1] <- NA
AT$maritalb[!(AT$maritalb %in% c(1:6))] <- NA
AT$lrscale[!(AT$lrscale %in% c(0:10))] <- NA
AT$dscrgrp[AT$dscrgrp != 0 & AT$dscrgrp != 1] <- NA
AT$hhmmb[AT$hhmmb %in% c(77, 88)] <- NA
AT$agea[AT$agea == 999] <- NA

AT$bthcld <- as.factor(AT$bthcld)
AT$maritalb <- as.factor(AT$maritalb)
AT$dscrgrp <- as.factor(AT$dscrgrp)

# Hungary
HU$eisced[HU$eisced == 55] <- NA
HU$eisced <- factor(HU$eisced, levels = c("1", "2", "3", "4", "5", "6", "7"),
                    ordered = T)
HU$bthcld[HU$bthcld == 1] <- 0
HU$bthcld[HU$bthcld == 2] <- 1
```

```
HU$dscrgrp[HU$dscrgrp == 1] <- 0
HU$dscrgrp[HU$dscrgrp == 2] <- 1

HU$bthcld[HU$bthcld != 0 & HU$bthcld != 1] <- NA
HU$maritalb[!(HU$maritalb %in% c(1:6))] <- NA
HU$lrscale[!(HU$lrscale %in% c(0:10))] <- NA
HU$dscrgrp[HU$dscrgrp != 0 & HU$dscrgrp != 1] <- NA
HU$hhmmb[HU$hhmmb %in% c(77, 88)] <- NA
HU$agea[HU$agea == 999] <- NA

HU$bthcld <- as.factor(HU$bthcld)
HU$maritalb <- as.factor(HU$maritalb)
HU$dscrgrp <- as.factor(HU$dscrgrp)

# Ireland
IE$eisced[IE$eisced == 55] <- NA
IE$eisced <- factor(IE$eisced, levels = c("1", "2", "3", "4", "5", "6", "7"),
                    ordered = T)
IE$bthcld[IE$bthcld == 1] <- 0
IE$bthcld[IE$bthcld == 2] <- 1
IE$dscrgrp[IE$dscrgrp == 1] <- 0
IE$dscrgrp[IE$dscrgrp == 2] <- 1

IE$bthcld[IE$bthcld != 0 & IE$bthcld != 1] <- NA
IE$maritalb[!(IE$maritalb %in% c(1:6))] <- NA
IE$lrscale[!(IE$lrscale %in% c(0:10))] <- NA
IE$dscrgrp[IE$dscrgrp != 0 & IE$dscrgrp != 1] <- NA
IE$hhmmb[IE$hhmmb %in% c(77, 88)] <- NA
IE$agea[IE$agea == 999] <- NA

IE$bthcld <- as.factor(IE$bthcld)
IE$maritalb <- as.factor(IE$maritalb)
IE$dscrgrp <- as.factor(IE$dscrgrp)

# Portugal
PT$eisced[PT$eisced == 55] <- NA
PT$eisced <- factor(PT$eisced, levels = c("1", "2", "3", "4", "5", "6", "7"),
                    ordered = T)
PT$bthcld[PT$bthcld == 1] <- 0
PT$bthcld[PT$bthcld == 2] <- 1
PT$dscrgrp[PT$dscrgrp == 1] <- 0
PT$dscrgrp[PT$dscrgrp == 2] <- 1

PT$bthcld[PT$bthcld != 0 & PT$bthcld != 1] <- NA
PT$maritalb[!(PT$maritalb %in% c(1:6))] <- NA
PT$lrscale[!(PT$lrscale %in% c(0:10))] <- NA
PT$dscrgrp[PT$dscrgrp != 0 & PT$dscrgrp != 1] <- NA
PT$hhmmb[PT$hhmmb %in% c(77, 88)] <- NA
PT$agea[PT$agea == 999] <- NA

PT$bthcld <- as.factor(PT$bthcld)
PT$maritalb <- as.factor(PT$maritalb)
PT$dscrgrp <- as.factor(PT$dscrgrp)
```

```
# Poland
PL$eisced[PL$eisced == 55] <- NA
PL$eisced <- factor(PL$eisced, levels = c("1", "2", "3", "4", "5", "6", "7"),
                    ordered = T)
PL$bthcld[PL$bthcld == 1] <- 0
PL$bthcld[PL$bthcld == 2] <- 1
PL$dscrgrp[PL$dscrgrp == 1] <- 0
PL$dscrgrp[PL$dscrgrp == 2] <- 1

PL$bthcld[PL$bthcld != 0 & PL$bthcld != 1] <- NA
PL$maritalb[!(PL$maritalb %in% c(1:6))] <- NA
PL$lrscale[!(PL$lrscale %in% c(0:10))] <- NA
PL$dscrgrp[PL$dscrgrp != 0 & PL$dscrgrp != 1] <- NA
PL$hhmmb[PL$hhmmb %in% c(77, 88)] <- NA
PL$agea[PL$agea == 999] <- NA

PL$bthcld <- as.factor(PL$bthcld)
PL$maritalb <- as.factor(PL$maritalb)
PL$dscrgrp <- as.factor(PL$dscrgrp)
```

3.4 Make subset of data with variables to be used. We will use 9 variables in addition to the analysis weight so we can weight the data during imputation

```
# Create variable vector containing the names of relevant variables
variables <- c("pdwrk", "bthcld", "gndr", "maritalb","lrscale", "dscrgrp",
               "hhmmb", "agea", "wkhtot", "anweight", "income", "eisced")

# Select subsets of data with relevant variables
AT_sub <- AT %>% select(variables)
```

```
## Warning: Using an external vector in selections was deprecated in tidyselect 1.1.0.
## i Please use `all_of()` or `any_of()` instead.
##   # Was:
##   data %>% select(variables)
##
##   # Now:
##   data %>% select(all_of(variables))
##
## See <https://tidyselect.r-lib.org/reference/faq-external-vector.html>.
```

```
HU_sub <- HU %>% select(variables)

IE_sub <- IE %>% select(variables)

PT_sub <- PT %>% select(variables)

PL_sub <- PL %>% select(variables)
```

3.5 Create interactions with weight variabels with covariates in imputation model

```r
# Interactions with anweight for Portugal
PT_sub$anweight_pdwrk <- PT_sub$anweight * PT_sub$pdwrk
PT_sub$anweight_bthcld <- PT_sub$anweight * as.numeric(PT_sub$bthcld)
PT_sub$anweight_lrscale <- PT_sub$anweight * PT_sub$lrscale
PT_sub$anweight_dscrgrp <- PT_sub$anweight * as.numeric(PT_sub$dscrgrp)
PT_sub$anweight_hhmmb <-PT_sub$anweight * PT_sub$hhmmb
PT_sub$anweight_agea <- PT_sub$anweight * PT_sub$agea
PT_sub$anweight_wkhtot <- PT_sub$anweight * PT_sub$wkhtot
PT_sub$anweight_income <- PT_sub$anweight *PT_sub$income
PT_sub$anweight_eisced <- PT_sub$anweight * as.numeric(PT_sub$eisced)

# Interactions with anweight for Austria
AT_sub$anweight_pdwrk <- AT_sub$anweight * AT_sub$pdwrk
AT_sub$anweight_bthcld <- AT_sub$anweight * as.numeric(AT_sub$bthcld)
AT_sub$anweight_lrscale <- AT_sub$anweight * AT_sub$lrscale
AT_sub$anweight_dscrgrp <- AT_sub$anweight * as.numeric(AT_sub$dscrgrp)
AT_sub$anweight_hhmmb <- AT_sub$anweight * AT_sub$hhmmb
AT_sub$anweight_agea <- AT_sub$anweight * AT_sub$agea
AT_sub$anweight_wkhtot <- AT_sub$anweight * AT_sub$wkhtot
AT_sub$anweight_income <- AT_sub$anweight * AT_sub$income
AT_sub$anweight_eisced <- AT_sub$anweight * as.numeric(AT_sub$eisced)

# Interactions with anweight for Hungary
HU_sub$anweight_pdwrk <- HU_sub$anweight * HU_sub$pdwrk
HU_sub$anweight_bthcld <- HU_sub$anweight * as.numeric(HU_sub$bthcld)
HU_sub$anweight_lrscale <- HU_sub$anweight * HU_sub$lrscale
HU_sub$anweight_dscrgrp <-HU_sub$anweight * as.numeric(HU_sub$dscrgrp)
HU_sub$anweight_hhmmb <- HU_sub$anweight * HU_sub$hhmmb
HU_sub$anweight_agea <- HU_sub$anweight * HU_sub$agea
HU_sub$anweight_wkhtot <- HU_sub$anweight * HU_sub$wkhtot
HU_sub$anweight_income <- HU_sub$anweight * HU_sub$income
HU_sub$anweight_eisced <- HU_sub$anweight * as.numeric(HU_sub$eisced)

# Interactions with anweight for Ireland
IE_sub$anweight_pdwrk <- IE_sub$anweight * IE_sub$pdwrk
IE_sub$anweight_bthcld <- IE_sub$anweight * as.numeric(IE_sub$bthcld)
IE_sub$anweight_lrscale <- IE_sub$anweight * IE_sub$lrscale
IE_sub$anweight_dscrgrp <- IE_sub$anweight * as.numeric(IE_sub$dscrgrp)
IE_sub$anweight_hhmmb <- IE_sub$anweight * IE_sub$hhmmb
IE_sub$anweight_agea <- IE_sub$anweight * IE_sub$agea
IE_sub$anweight_wkhtot <- IE_sub$anweight * IE_sub$wkhtot
IE_sub$anweight_income <- IE_sub$anweight * IE_sub$income
IE_sub$anweight_eisced <- IE_sub$anweight * as.numeric(IE_sub$eisced)

# Interactions with anweight for Poland
PL_sub$anweight_pdwrk <- PL_sub$anweight * PL_sub$pdwrk
PL_sub$anweight_bthcld <- PL_sub$anweight * as.numeric(PL_sub$bthcld)
PL_sub$anweight_lrscale <- PL_sub$anweight * PL_sub$lrscale
PL_sub$anweight_dscrgrp <- PL_sub$anweight * as.numeric(PL_sub$dscrgrp)
PL_sub$anweight_hhmmb <- PL_sub$anweight * PL_sub$hhmmb
PL_sub$anweight_agea <- PL_sub$anweight * PL_sub$agea
PL_sub$anweight_wkhtot <-  PL_sub$anweight * PL_sub$wkhtot
PL_sub$anweight_income <- PL_sub$anweight * PL_sub$income
```