European Values Study (EVS) 2017: Weighting Data

gesis Leibniz-Institut für Sozialwissenschaften

GESIS Papers

2020|15

European *Values* Study

European Values Study (EVS) 2017

Weighting Data

European Values Study (EVS)

GESIS Papers 2020|**15**

European Values Study (EVS) 2017

Weighting Data

European Values Study (EVS)

GESIS – Leibniz-Institut für Sozialwissenschaften 2020

GESIS Papers

GESIS – Leibniz-Institut für Sozialwissenschaften
Datenarchiv für Sozialwissenschaften
International Surveys
Unter Sachsenhausen 5-8
53667 Köln

E-Mail: evsservice@gesis.org

ISSN: 2364-3781 (Online)
Herausgeber,
Druck und Vertrieb: GESIS – Leibniz-Institut für Sozialwissenschaften
Unter Sachsenhausen 6-8, 50667 Köln

Table of Contents

Acknowledgments 4
1 Introduction 5
2 Calibration weights 6
2.1 Calibration variables 6
2.2 Missing values imputation 7
3 Population size weights 8
4 Design weights 9
5 Bibliography 10
Appendix A. Template raking procedure (software used: Rstudio) 11
Appendix B - Country specific deviations in calibration variables 16

Acknowledgments

This work was undertaken under supervision of the EVS Methodology Group (Ruud Luijkx, Alice Ramos, Bart Meuleman, Ivan Rimac, Frédéric Gonthier, Markus Quandt, Michael Braun, Dominque Joye).

The fieldwork of the 2017 European Values Study (EVS) was financially supported by universities and research institutes, national science foundations, charitable trusts and foundations, companies and church organizations in the EVS member countries.

The project would not have been possible without the National Program Directors in the EVS member countries and their local teams.

We are grateful to Matthias Sand (GESIS, Mannheim) and Christian Bruch (GESIS, Mannheim) for their help with implementing the weighting procedures

Special thanks also go to the following members of the EVS teams at Tilburg University and GESIS-DAS for their contribution in preparing and distributing this document:

> Giovanni Borghesan (EVS, Tilburg), Evelyn Brislinger (GESIS-DAS), Santiago Gomez (EVS, Tilburg), Angelica Maineri (EVS, Tilburg), Esma Morgül (EVS, Tilburg), and Ivet Solanes (GESIS-DAS). In addition, we would like to thank Kerstin Beck for helping to finalise and publish the documentation on the EVS 2017 data.

1 Introduction

The report outlines the procedure used to compute the EVS17 weights and information on the source of the population statistics and adaptations required in each country. The weighting of survey data generally denotes every operation that alters the relative importance of sampling units (of groups or units) for the purpose of estimating relevant statistics (e.g. means or totals) of a target population.

The weights included in the full release of EVS data are two versions of calibration weights, a population size weight and – for a limited number of countries – a design weight. For the countries where it is provided the design weight has not been factored in the computation of calibration weights.

The type of analysis that one is planning to conduct guides the choice of the weights to use. The combinations of these are described in more detail in the table below.

Please note: weights are only provided for the EVS 2017 Integrated dataset:

> EVS (2020): European Values Study 2017: Integrated Dataset (EVS 2017).
> GESIS Data Archive, Cologne. ZA7500 Data file Version 4.0.0, doi:10.4232/1.13560.

Table 1: Recommended weights by research type

Research type	Recommended weight to be used
Single country	Calibration weights or design weights
Country comparison with combined statistics	Calibration weights or design weights & Population size weights
Country comparison without combined statistics	Calibration weights or design weights
Country comparison with combined statistics and reference to groups of countries (i.e. region)	Calibration weights or design weights & Population size weights

2 Calibration weights

Calibration weights serve various purposes. Two of the main reasons for that sort of weighting would be the potential reduction in an estimate's variance and the potential reduction in bias due to nonresponse/missing data. Hence, these weights adjust (some) socio-demographic characteristics in the sample population to the distribution of the target population. The calibration weights have been computed for each country based on separate population margins of age, sex, educational levels, and region provided by the countries themselves. The sources of the population statistics are listed in the Appendix. Two versions of calibration weights have been computed with different combinations of calibration variables:

- **'gweight'** has been computed using the marginal distribution of age, sex, educational attainment and region. This weight is provided as a standard version for consistency with previous releases.
- **'gweight_no_edu'** has been computed using the marginal distribution of age, sex and region. This weight is provided for researchers that reject the assumption of consistent measuring of educational level across countries (cf. Ortmanns V. & Schneider S.L. 2016) and/or assume that non-responses have the same distribution across educational levels.

The two weight variables have been computed employing a raking algorithm specifically prepared for this procedure. A template of the algorithm is provided in Appendix A. Ultimately, all calibration weights have been trimmed at the 97.5th percentile for avoiding extreme values. Hence, the mean of calibration weights in each country is slightly different from 1. A description of calibration variables is provided in the appendix together with the country-specific deviations that occurred in the computation of weight. The dataset is provided including the variables with the imputed values for transparency and reproducibility purposes, a suffix "_weight" has been added to these variables.

2.1 Calibration variables

The variables used for calibration are age, gender, region, and level of education. The standard coding for age categories, sex and educational level described in Table 2.). In some cases, the data provided at the country level did not match such coding. Consequently, the coding has been adjusted to match population data, the cases, where these changes occurred are listed in Appendix B. As concerns region, the NUTS2 regions were used (with some exceptions, noted in Appendix B, where NUTS1 levels were used – or no regional variable was used due to data availability).

Table 2: Coding of calibration variables

Age	
Code	Label
1	18-24
2	25-34
3	35-44
4	45-54
5	55-64
6	65-74
7	75+

Sex	
Code	Label
1	Male
2	Female

Education	
Code	Label
1	Low
2	Medium
3	High

2.2 Missing values imputation

The application of weighting procedures requires complete observations without missing values in the weighting variables. For this reason, missing values resulting from non-response on the respondent's gender, age, and educational level, were imputed through the R package *mice*. The imputed values for calibration variables can be found in the dataset with the suffix _Calibration. In the case of region, an administrative variable, no imputation was needed. Our approach consisted of producing a single imputation based on 50 iterations, employing the *Classifications and regression trees* method. The employment of multiple auxiliary variables is meant to get a more accurate prediction of the imputed values based on relevant respondent's characteristics. The variables used as predictors for the imputation have been selected by the EVS central team because they represent core aspects of one's value orientation (e.g. belonging to a religious denomination and interest in politics), or because they correlate with the weighting variables (e.g. Age at which full-time education was completed and educational level).

These variables are

- *v51* (Belonging to a religious denomination),
- *v242* (Age at which full-time education was completed),
- *v97* (Interest in politics),
- *v234* (Current marital status),
- *v227* (Nationality),
- *v225*, (Respondent's sex),
- *v243_r* (Respondent's level of education in three categories) and *age_r3* (Respondent's age in three categories).

More details on these variables can be found in the ZA7500 Variable Report.

3 Population size weights

Population size weights ('pweight') are provided for rescaling the weights to a shared denominator across all countries. These weights must be applied whenever one ought to analyse together different countries and avoids the overrepresentation of small countries when compared to bigger ones. The population size weight is computed as follows:

$$Population\ size\ weight = \frac{\text{Population size aged 18 years and older}}{\text{Net sample size} * 10\,000}$$

Table 3 reports the target population size by country used for computing population size weights.

Table 3: Target population size by country

Country	Target Population
AL - Albania	2007877
AM - Armenia	2190686
AT - Austria	7509125
AZ - Azerbaijan	7354320
BA - Bosnia Herzegovina	2838458
BG - Bulgaria	6181241
BY - Belarus	7304173
CH - Switzerland	6909664
CZ - Czechia	8585396
DE - Germany	68084270
DK - Denmark	4024325
EE - Estonia	1065731
ES - Spain	35096178
FI - Finland	4407913
FR - France	49764122
GB - Great Britain	50644094
GE - Georgia	5775216
HR - Croatia	3437119
HU - Hungary	8381900
IS - Iceland	241300
IT - Italy	48238236
LT - Lithuania	2337516
ME - Montenegro	474655
MK - North Macedonia	1814644
NL - Netherlands	13794988
NO - Norway	4183147
PL - Poland	31798656
PT - Portugal	8107271
RO - Romania	18267514
RS - Serbia	6137160
RU - Russia	121153927
SE - Sweden	7998644
SI - Slovenia	1628479
SK - Slovakia	4432721

4 Design weights

Design weights ('dweight') are meant to adjust individual's probabilities of being included in the sample. The use of these weights, accounts for the variation in individual selection probabilities, which are likely to be different, especially in complex sample designs with multiple stages. More detailed information on the sampling procedures used by the country teams can be found in the EVS 2017 Method Report.

More specifically, in a complex sample design, an individual 's selection probability is the product of each individual selection probability at every stage which, for sampling choices or chance and may differ for each respondent. On the opposite, a simple random sample, would lead the selection probabilities to be the same for every respondent (n/N) and hence the same design weight, namely, 1. Hence, the final weights are computed as the inverse of the product of the selection probabilities at each sampling stage.

For example, in a sample design with three stages, PROB = ¬PROB_PSU*PROB_SSU*PROB_USU. These weights are then rescaled to the sample size in a way by which their mean is 1 and their sum equals the sample size. Consequently, the scaled weight takes the form of:

$$dweight = \frac{1}{PROB} \Big/ \sum \frac{1}{PROB} * n$$

The accuracy of design weights is assessed by comparing the target population and the sum of the unscaled design weights divided by the response rate. Design weights will be provided for Azerbaijan, Croatia, Poland, Russia and Germany.

5 Bibliography

Ortmanns, V., & Schneider, S. L. (2016). Can we assess representativeness of cross-national surveys using the education variable?. In Survey Research Methods Vol. 10(3): 189-210. doi: http://dx.doi.org/10.18148/srm/2016.v10i3.6608.

Appendix A. Template raking procedure (software used: Rstudio)

```
1.  ####################################################
2.  ####### Master Script New Raking Procedure ########
3.  ####################################################
4.
5.  # ------------------------------------------------- #
6.  #  European Values Survey                           #
7.  #  Script - Calibration weights/Raking process      #
8.  #  Country: #######                                 #
9.  #  Last modified: ##/##/## by ############          #
10. # ------------------------------------------------- #
11. rm(list = ls())
12.
13. # - Loading the packages and setting the working directory - #
14. if (!require(haven)){ install.packages('haven') }
15. if (!require(dplyr)){ install.packages('dplyr') }
16. if (!require(readxl)){ install.packages('readxl') }
17. if (!require(mice)){ install.packages('mice') }
18.
19. # Set directory
20. User <- Sys.info()[['user']]
21. setwd(paste0('C:\\Users\\',User,'\\surfdrive\\Shared\\EVS
    Weights\\NEW WEIGHTS'))
22.
23. # Load sample data
24. sample_evs <- read_sav("######.sav")
25. sample_evs <- plyr::rename(sample_evs, c("id_cocas" = "id_cocas",
    "v225"="sex", "age_r3"="agecat", "v243_r"="edu", "v275b_N2" = "re-
    gion"))
26.
27.
28. ##############################
29. # Load Population statistics #
30. ##############################
31.
32. # AGE
33. read_excel("####.xlsx",
34.            sheet = "A_by_R") -> pop_AbyR
35. pop_AbyR <- plyr::rename(pop_AbyR, c("N_group"= "N_groupa",
    "N_pop_region" = "N_pop_regiona" ))
36. pop_AbyR <- pop_AbyR[,c("country", "region", "agecat", "N_groupa",
    "N_pop_regiona")]
37.
38. # SEX
39. read_excel("####.xlsx",
40.            sheet = "S_by_R") -> pop_SbyR
41. pop_SbyR <- plyr::rename(pop_SbyR, c("N_group"= "N_groups",
    "N_pop_region" = "N_pop_regions" ))
42. pop_SbyR <- pop_SbyR[,c("country", "region", "sex", "N_groups",
    "N_pop_regions")]
43.
44. # EDUCATION
45. read_excel("####.xlsx",
46.            sheet = "E_by_R")  ->  pop_EbyR
47.
```

```
48. pop_EbyR <- plyr::rename(pop_EbyR, c("N_group"= "N_groupedu",
    "N_pop_region" = "N_pop_regionedu" ))
49. pop_EbyR <- pop_EbyR[,c("country", "region", "edu", "N_groupedu",
    "N_pop_regionedu")]
50.
51. # REGION
52. read_excel("####.xlsx",
53.            sheet = "##_by_R") -> pop_R
54. pop_R <- pop_R[,c("country","region", "N_group", "N_pop_region")]
55.
56. # AGGREGATION FOR EMPTY CELLS
57. sample_evs$region <- as.character(sample_evs$region)
58. sample_evs$region <- substr(sample_evs$region,1,3)
59. pop_R$region <- substr(pop_R$region,1,3)
60.
61. sample_evs$agecat[sample_evs$agecat==#] <- #
62. pop_AbyR$agecat[pop_AbyR$agecat==#] <- #
63.
64. # IMPUTATION
65. sample_evs %>%
66. select(id_cocas, v51, v242, v97, v234, v227, sex, agecat, region,
    edu)  %>%
67. mutate(sex    = factor(sex),
68.        edu  = factor(edu),
69.       agecat  = factor(agecat),
70.       region = factor(region),
71.       v51 = factor(v51),
72.       v242 = factor(v242),
73.       v97 = factor(v97),
74.       v234 = factor(v234),
75.       v227 = factor(v227))   -> imp_data
76.
77. # Imputation routine for missing values
78. sam_imp <- mice(imp_data,m=1,maxit=1,meth='cart',seed=500)
79. summary(sam_imp)
80.
81. # set id_cocas to 0 (so it doesn't impute)
82. pred_matrix <-sam_imp$predictorMatrix
83. pred_matrix[, 1] <- 0
84.
85. # leave only imputation of agecat, edu, sex and region
86. pred_matrix[1:6,] <- 0
87. pred_matrix
88. # run mice again using new predictor matrix and the full number of
    iterations
89. sam_imp_reduced <- mice(sam_imp$data, m=1, maxit=50, pred_matrix,
    meth='cart',seed=500)
90.
91. # Check imputations
92. sam_imp_reduced$imp$agecat
93. sam_imp_reduced$imp$edu
94. sam_imp_reduced$imp$sex
95. sam_imp_reduced$imp$region
96.
97.
98. # Adding imputed values to dataframe
99. sample_evs <- complete(sam_imp_reduced, 1)
100. sample_evs <- sample_evs[, c("id_cocas", "region", "agecat", "sex",
    "edu")]
101.
102. # POPULATION MARGINS
103. masage <- pop_AbyR %>%
104.   group_by(agecat) %>%
105.   dplyr:::select(agecat, N_groupa) %>%
```

```
106.   summarise(x = sum(N_groupa)) %>%
107.   ungroup() %>%
108.   mutate(n = sum(x),
109.          freq = x/n)
110.
111. massex <- pop_SbyR %>%
112.   group_by(sex) %>%
113.   dplyr:::select(sex, N_groups) %>%
114.   summarise(x = sum(N_groups)) %>%
115.   ungroup() %>%
116.   mutate(n = sum(x),
117.          freq = x/n)
118.
119. masreg <- pop_R %>%
120.   group_by(region) %>%
121.   dplyr:::select(region, N_group) %>%
122.   summarise(x = sum(N_group)) %>%
123.   ungroup() %>%
124.   mutate(n = sum(x),
125.          freq = x/n)
126.
127. masedu <- pop_EbyR %>%
128.   group_by(edu) %>%
129.   dplyr:::select(edu, N_groupedu) %>%
130.   summarise(x = sum(N_groupedu)) %>%
131.   ungroup() %>%
132.   mutate(n = sum(x),
133.          freq = x/n)
134.
135. # SAMPLE MARGINS
136. sampdat <- sample_evs %>%
137.   dplyr:::select(sex,agecat,region) %>%
138.   group_by(sex, agecat, region) %>%
139.   count(sex, agecat, region) %>%
140.   ungroup() %>%
141.   mutate(freq=n/sum(n))
142.
143. masage <- as.data.frame(masage)
144. massex <- as.data.frame(massex)
145. masreg <- as.data.frame(masreg)
146. masedu <- as.data.frame(masedu)
147. sampdat <- as.data.frame(sampdat)
148.
149.
150. # CHECK
151. masage$freq
152. count(sampdat, agecat)
153.
154. massex$freq
155. count(sampdat, sex)
156.
157. masreg$region
158. count(sampdat, region)
159.
160. ##########
161. # RAKING #
162. ##########
163.
164. # Raking without education
165. w_0 <- rep(1,times=nrow(sampdat))
166. times <- 0
167. while(times <= 1000){
168.   # Sex
169.   saxe <- aggregate(sampdat$freq*w_0,list(sex=sampdat$sex),sum)
```

```
170.    w_1 <- massex[,4]/saxe[,2]
171.    w_1 <- w_1[as.numeric(sampdat$sex)]
172.    # Age
173.    saxs <- aggre-
   gate(sampdat$freq*w_0*w_1,list(agecat=sampdat$agecat),sum)
174.    w_2 <- masage[,4]/saxs[,2]
175.    w_2 <- w_2[as.numeric(sampdat$agecat)]
176.    #Region
177.    reg <- aggre-
   gate(sampdat$freq*w_0*w_1*w_2,list(region=sampdat$region),sum)
178.    w_3 <- masreg[,4]/reg[,2]
179.    w_3 <- w_3[as.integer(factor(sampdat$region))]
180.    #w4
181.    w_4 <- w_0*w_1*w_2*w_3
182.    if(max(abs(w_0-w_4))>0.0000005)
183.    {w_0<-w_4
184.    times<-times+1}
185.    else {break}
186.    cat("iteration",times,"\n")
187.  }
188.
189.  # Margins must be equal
190.  aggregate(sampdat$freq*w_4,list(agecat=sampdat$agecat),sum)
191.  masage$freq
192.
193.  aggregate(sampdat$freq*w_4,list(sex=sampdat$sex),sum)
194.  massex$freq
195.
196.  aggregate(sampdat$freq*w_4,list(region=sampdat$region),sum)
197.  masreg$freq
198.
199.  # Untrimmed weights
200.  sampdat$weight_n <- w_0
201.
202.  # Trimmed weights
203.  bound <- quantile(w_0, c(.975))
204.  sampdat$weight_nt <- trunc.bounds(w_0, c(0, bound))
205.
206.  sample_evs <- left_join(sample_evs, sampdat, by = c("agecat"=
   "agecat", "region"= "region", "sex"= "sex"))
207.
208.  summary(sample_evs$weight_n) # Mean must be exactly 1
209.  sd(sample_evs$weight_n)
210.  summary(sample_evs$weight_nt)
211.
212.
213.
214.  rm(sampdat)
215.  sampdat <- as.data.frame(sample_evs %>%
216.
   dplyr:::select(sex,region,agecat,edu) %>%
217.                              group_by(sex,region,agecat,edu) %>%
218.                              count(sex,region,agecat,edu) %>%
219.                              ungroup() %>%
220.                              mutate(freq=n/sum(n)))
221.
222.  masedu$freq
223.  count(sampdat, edu)
224.
225.  # Raking with education
226.  w_0 <- rep(1,times=nrow(sampdat))
227.  times <- 0
228.  while(times <= 1000){
229.    # Sex
```

```
saxe <- aggregate(sampdat$freq*w_0,list(sex=sampdat$sex),sum)
w_1 <- massex[,4]/saxe[,2]
w_1 <- w_1[as.numeric(sampdat$sex)]
# Age
saxs <- aggregate(sampdat$freq*w_0*w_1,list(agecat=sampdat$agecat),sum)
w_2 <- masage[,4]/saxs[,2]
w_2 <- w_2[as.numeric(sampdat$agecat)]
#Education
sexs <- aggregate(sampdat$freq*w_0*w_1*w_2,list(edu=sampdat$edu),sum)
w_3 <- masedu[,4]/sexs[,2]
w_3 <- w_3[as.numeric(sampdat$edu)]
#Region
reg <- aggregate(sampdat$freq*w_0*w_1*w_2*w_3,list(region=sampdat$region),sum)
w_4 <- masreg[,4]/reg[,2]
w_4 <- w_4[as.integer(factor(sampdat$region))]
#w4
w_5 <- w_0*w_1*w_2*w_3*w_4
if(max(abs(w_0-w_5))>0.0000005)
{w_0<-w_5
times<-times+1}
else {break}
cat("iteration",times,"\n")
}

# Margins must be equal
aggregate(sampdat$freq*w_5,list(agecat=sampdat$agecat),sum)
masage$freq

aggregate(sampdat$freq*w_5,list(sex=sampdat$sex),sum)
massex$freq

aggregate(sampdat$freq*w_5,list(region=sampdat$region),sum)
masreg$freq

aggregate(sampdat$freq*w_5,list(edu=sampdat$edu),sum)
masedu$freq


# Untrimmed weights
sampdat$weight_e <- w_0

# Trimmed weights
bound <- quantile(w_0, c(.975))
sampdat$weight_et <- trunc.bounds(w_0, c(0, bound))

sample_evs <- left_join(sample_evs, sampdat, by = c("agecat"="agecat", "region"= "region", "sex"= "sex","edu" = "edu" ))

summary(sample_evs$weight_e) # Mean must be exactly 1
sd(sample_evs$weight_e)
summary(sample_evs$weight_et)


sample_evs <- sample_evs[, c("id_cocas", "region", "agecat", "sex", "edu","weight_e","weight_et","weight_n","weight_nt")]
write_dta(sample_evs, "#####.dta") # put country name
write_sav(sample_evs, "#####.sav") # put country name
```

Appendix B - Country specific deviations in calibration variables

The section below lists the population data sources and all the country-specific deviations both for individual and country-level data. The item “missing values imputed” indicates how many observations had missing data on the variables of interest and have hence been imputed with the procedure described above. ‘None’ means there were no missing values on the variables of interest, hence there was no need for imputation.

Albania - AL

Source of data: N. A.
Population statistics: Standard classification, margins equalized; not weighted on regions due to lack of statistics at regional level.
Missing values imputed: 5 Education

Armenia - AM

Source of data: 2011 Armenian Population Census
Population statistics: Standard Classification, no region in population statistics
Missing values imputed: None

Austria - AT

Source of data: 2015 Bildungsstandregister
Population statistics: Standard classification.
Missing values imputed: 8 Education

Azerbaijan - AZ

Source of data: Azərbaycan Respublikasinin Dövlət Statistika Komitəsi (State Statistical Committee of the Republic of Azerbaijan)
Population statistics: Educational distribution starts at 15 years old, NUTS1 regions.
Missing values imputed: None

Belarus - BY

Source of data: National Statistical Committee of the Republic of Belarus as of January 01, 2017 for age and gender. Survey data "Generations and Gender" for Education.
Population statistics: Standard classification (no region).
Missing values imputed: 1 education

Bulgaria - BG

Source of data: N. A.
Population statistics: Age categories stop at 60 years old.
Missing values imputed: 7 Education, 18 Age

Croatia - HR

Source of data: N.A.
Population statistics: ISCED code 2 coded as 'Medium'.
Missing values imputed: 6 Education, 1 Age.

Czech Republic - CZ

Source of data: Czech 2011 Population Census
Population statistics: Standard classification.
Missing values imputed: 17 Education, 66 Age

Denmark - DK

Source of data: Statistics Denmark
Population statistics: Standard classification.
Missing values imputed: 16 Education, 4 Age, 4 Sex

Estonia - EE

Source of data: Estonian Statistics
Population statistics: Standard classification.
Missing values imputed: None

N.B. Please note that education data from Statistics Department is a mix of the 2011 census, population register (people may mention education voluntarily) and different education registers.

Finland - FI

Source of data: Tilastokeskus.
Population statistics: Standard classification; region FI20 missing in survey data hence dropped.
Missing values imputed: 6 Education, 2 Sex, 35 Age, 3 Region

France - FR

Source of data: INSEE enquête Emploi 2014 – updated in 2017
Population statistics: Population distribution for education stops at 70 years old; not weighted on regions due to lack of statistics at regional level.
Missing values imputed: 7 Education, 5 Age

Georgia - GE

Source of data: N. A.
Population statistics: Standard classification.
Missing values imputed: 2 Education.

Germany – DE

Source of data: German Microcensus 2016
Population statistics: Standard classification. Regions are NUTS1.
Missing values imputed: 100 Education, 89 Age, 53 Sex.

Great Britain - GB

Source of data: N. A.
Population statistics: Standard classification, Region are NUTS1.
Missing values imputed: 11 Education, 12 Age

Hungary - HU

Source of data: 2016 Microcensus.
Population statistics: Standard classification, NUTS2.
Missing values imputed: 8 Education

Iceland - IS

Source of data: N. A.
Population statistics: Education distribution ends at 74 years old.
Missing values imputed: 27 Education, 20 Sex, 22 Age

Italy - IT

Source of data: Eurostat (January 1, 2018) for Age and Gender. ISTAT Census data 2011 for Education.
Population statistics: Standard classification. Regions are NUTS1.
Missing values imputed: 13 Education.

Lithuania - LT

Source of data: N. A.
Population statistics: not weighted on regions due to lack of statistics at regional level.
Missing values imputed: 7 Education, 1 Age

Macedonia - MKs

Source of data: N. A.
Population statistics: Standard classification; not weighted on regions due to lack of statistics at regional level.
Missing values imputed: 12 Education, 4 Sex, 8 Age

Netherlands – NL

Source of data: *Centraal Bureau voor de Statistiek* for Age and Gender distribution, CBS statline for Education.
Population statistics: Standard classification.
Missing values imputed: 35 Edu.

Norway - NO

Source of data: Statistics Norway (Education: 2018; age and gender: NA)
Population statistics: different age categories, not weighted on regions due to lack of statistics at regional level.
Missing values imputed: 20 Education, 2 Age.

Poland - PL

Source of data: PESEL (Universal Electronic System for Registration of the Population) registry for Age and Gender, Labour force survey in Poland I quarter 2018 for Education.
Population statistics: Standard classification.
Missing values imputed: 8 Education

Portugal- PT

Source of data: http://www.ine.pt. Data updated in February 2014.
Population statistics: Standard classification. Gweight not weighted on region.
Missing values imputed: 3 Age.

Romania - RO

Source of data: Institutul naţional de statistică; Education from 2011 Census
Population statistics: Standard classification.
Missing values imputed: 40 Education, 54 Age.

Russia - RU

Source of data: Russian National Census 2010.
Population statistics: Standard classification.
Missing values imputed: 7 Education.

Serbia - RS

Source of data: N. A.
Population statistics: Standard classification.
Missing values imputed: 8 Education, 19 Age

Slovakia - SK

Source of data: N. A.
Population statistics: Age classification ends at 65 years old.
Missing values imputed: 9 Education.

Slovenia - SI

Source of data: Statistical Office of the Republic of Slovenia
Population statistics: Age starts at 20 and ends at 65.
Missing values imputed: 6 Education.

Spain - ES

Source of data: Encuesta del Padrón, INE; Encuesta de Población Activa, INE
Population statistics: Population distribution for education starts at 25 years old.
Missing values imputed: 7 Education

Sweden - SE

Source of data: N. A.
Population statistics: Standard classification.
Missing values imputed: 14 Education, 2 Sex, 8 Age.

Switzerland - CH

Source of data: Statistik der Bevölkerung und der Haushalte (STATPOP).
Population statistics: Standard classification.
Missing values imputed: 130 Education, 2 Age.