Making Choices about the EVS data

Edoardo Costantini

9/22/2020

Countries and Observations: Matrix Desing VS Integrated data

Remeber from the pdf file ZA7500_mr.pdf * The Integrated Dataset (ZA7500) contains data from 55,256 respondents and 33 countries. * The Matrix Design Dataset (ZA7502) includes data from 10,598 respondents and the four countries (DE IS CH NL) that used the matrix design.

Observations are devided in

```
lapply(list(int.dt = int.dt$mm_select_sample,
            mad.dt = mad.dt$mm_select_sample), table)
```

```
## $int.dt
##
##     1     2     3     4
## 47195  3793   437  3851
##
## $mad.dt
##
##    2    3    4    5    6    7
## 1316  437 3851 1635 3237  122
```

And you can see here what those values mean:

```
val_labels(int.dt$mm_select_sample)
```

```
##                                        interviewer-administered (CAPI PAPI CATI)
##                                                                                1
## self-administered full-length questionnaire: original question order (CAWI Mail)
##                                                                                2
## self-administered full-length questionnaire: modified question order (CAWI Mail)
##                                                                                3
##                             self-administered matrix: with follow-up (CAWI Mail)
##                                                                                4
##                     self-administered matrix: follow-up non response (CAWI Mail)
##                                                                                5
##                     self-administered matrix: first survey only (CAWI Mail - DE)
##                                                                                6
##                                          break-off (less than 50% valid answers)
##                                                                                7
```

If you consider the 6 EU founding countries (Belgium, France, Germany, Italy, Luxembourg, Netherlands), this is how observations are distributed among the two datasets:

```
countries  <- c("Belgium", "France", "Germany", "Italy", "Luxembourg", "Netherlands")
```

```
tab.1 <- table(int.df$country, int.df$mm_select_sample)
print(tab.1[rownames(tab.1) %in% countries, ])
```

```
##
##                  1    2    3    4
##   France      1870    0    0    0
##   Germany     1494  676    0    0
##   Italy       2277    0    0    0
##   Netherlands  686    0    0 1718
```

```
tab.2 <- table(mad.df$country, mad.df$mm_select_sample)
print(tab.2[rownames(tab.2) %in% countries, ])
```

```
##
##                  2    3    4    5    6    7
##   France         0    0    0    0    0    0
##   Germany      676    0    0    0 3237   49
##   Italy          0    0    0    0    0    0
##   Netherlands    0    0 1718  324    0   11
```

- Belgium and Luxembourg are not surveyed by EVS 2017.
- Netherlands has almost 2000 observations in group 4 (self-administered matrix)

In **conclusion**:

- Countries to keep: France, Germany, Italy, Netherlands
- Subsamples to keep: 1 and 4 from the integrated dataset

Variables to keep

Generic variables by type:

```
id  <- "id_cocas"
ord <- paste0("v", c(1:8, 32:39, 46:50, 63:70, 72:84,
                     97:107, 115:168, 170:172,
                     176:203, 205:224,
                     240, 242, 267:274, 280,
                     c("174_LR", "239_r", "239a", "239b")))
dic <- paste0("v", c(9:31,40:45,51,57:61,71,85:95,
                     112,169,225,227,230,232,248,259,260))
nom <- paste0("v", c(52, 62,   # religiosity
                     108:111, 113:114, 234, 238))
```

Political tendencies

Use a vairable measuring self reported tendency to vote for parties (recoded by EVS into a continuous variable)

```
pol <- "v175_LR"
val_labels(int.df[, pol])
```

```
##  multiple answers Mail           no follow-up follow-up non response
##                     -10                     -9                     -8
##           other missing      item not included         not applicable
##                      -5                     -4                     -3
##               no answer              dont know                   left
##                      -2                     -1                      1
##                  right       not classifiable
##                     10                     44
```

```
table(int.df[, pol])
```

```
## 
##    -5    -3    -2    -1     1     2     3     4     5     6     7     8     9 
## 10821  4859  9927  7902   863  1157  2961  3731  2293  4058  3451  2015   679 
##    10    44 
##   443   116
```

The 116 'not classificable' cases are assigned missing values

```
int.df[which(int.df[, pol] == 44), pol] <- NA
```

Age

For age, I use a constructed age vairables in "number of years old" format.

```
age <- "age"
val_labels(int.df[, age])
```

```
##  multiple answers Mail           no follow-up follow-up non response 
##                    -10                     -9                     -8 
##          other missing      item not included         not applicable 
##                     -5                     -4                     -3 
##              no answer              dont know           82 and older 
##                     -2                     -1                     82
```

```
table(int.df[, age])
```

```
## 
##   -2   -1   18   19   20   21   22   23   24   25   26   27   28   29   30   31 
##  311   11  521  818  762  782  779  759  749  745  825  842  843  849  859  856 
##   32   33   34   35   36   37   38   39   40   41   42   43   44   45   46   47 
##  809  819  820  896  884  942  901  905  976  912  902  887  873  912  892  944 
##   48   49   50   51   52   53   54   55   56   57   58   59   60   61   62   63 
##  946  983  978  942  957 1017  933  940  934  994 1050  954 1020  941  957  999 
##   64   65   66   67   68   69   70   71   72   73   74   75   76   77   78   79 
##  989 1008  890  972  945  892  892  834  637  640  610  532  539  504  528  417 
##   80   81   82 
##  434  364 1519
```

Education

For education, I use the ISCED version and I will be treating it as continuous. I do this for the education of the respondent and their father and mother.

```
# Education
  edu <- c("v243_ISCED_1", # continuous is fine
           "v262_ISCED_1",
           "v263_ISCED_1")

  lapply(int.df[, edu], function(x) {
    list(label = var_label(x),
         table = table(x))
  }
  )
```

```
## $v243_ISCED_1
## $v243_ISCED_1$label
## [1] "educational level respondent: ISCED-code one digit (Q81)"
##
## $v243_ISCED_1$table
## x
##    -2    -1     0     1     2     3     4     5     6     7     8    66
##   291    77   458  2423  8140 22778  2556  4187  6156  7593   533    84
##
##
## $v262_ISCED_1
## $v262_ISCED_1$label
## [1] "educational level father: ISCED-code one digit (Q99)"
##
## $v262_ISCED_1$table
## x
##    -3    -2    -1     0     1     2     3     4     5     6     7     8    66
##    58   920  3079  3697  7664 10774 18030  1584  2756  2199  3923   472   120
##
##
## $v263_ISCED_1
## $v263_ISCED_1$label
## [1] "educational level mother: ISCED-code one digit (Q100)"
##
## $v263_ISCED_1$table
## x
##    -3    -2    -1     0     1     2     3     4     5     6     7     8    66
##    16   834  2248  4385  8891 13521 16111  1301  2429  2337  2965   170    68
```

Values 66 need to be recoded as missings as it does not belong in any order of education.

```
val_labels(int.df[, edu[1]])
```

```
##                no follow-up      follow-up non response
##                          -9                          -8
##               other missing           item not included
##                          -5                          -4
##              not applicable                   no answer
##                          -3                          -2
##                   dont know           Less than primary
##                          -1                           0
##                     Primary             Lower secondary
##                           1                           2
##             Upper secondary Post-secondary non tertiary
##                           3                           4
##        Short-cycle tertiary      Bachelor or equivalent
##                           5                           6
##        Master or equivalent      Doctoral or equivalent
##                           7                           8
##                       other
##                          66
```

```
# Check presence of 66 cases
for (j in edu) {
  int.df[which(int.df[, j] == 66), j] <- NA
}
```