Management and Analysis of Large Survey Data Sets Using the `memisc` Package

Martin Elff

Universität Mannheim
Lehrstuhl für Politische Wissenschaft
und International Vergleichende Sozialforschung

August 7, 2008

Importing foreign data files

Declaring the external file

```
library(memisc)
allbus_file <- "ZA4243_GCUM.SAV"
allbus <- spss.system.file(allbus_file)
allbus
```

```
SPSS system file 'ZA4243_GCUM.SAV'
    with 1250 variables and 47947 observations
```

```
object.size(allbus)
```

```
[1] 8697408
```

That is 8.3 MB although the cumulated ALLBUS (German General Social Survey) data file has size 76.8 MB and the completely uncompressed numerical data would need at least 228.6 MB!

Getting a description of variables

```
description(allbus)
```

```
v1    'ZA STUDY NUMBER'
v2    'YEAR'
v3    'SPLIT QUESTIONNAIRE'
v4    'RESPONDENT ID NUMBER'
v5    'REGION OF INTERVIEW: WEST - EAST'
v6    'GERMAN CITIZENSHIP?'
v7    'INTERVIEW: CAPI OR PAPI?'
v8    'SAMPLING DESIGN'
v9    'CURRENT ECONOMIC SITUATION IN GERMANY'
```

(...)

```
v1249 'WEIGHT: E-W+TRANSF. TO HOUSEHOLD-LEVEL'
v1250 'RELEASE'
```

Reading in a subset of variables

```
classd.churchat.data <- subset(allbus,
  select=c(
    year                      = v2,
    east.west                 = v5,
    left.right                = v19,
    vote.intention            = v24,
    birthyear                 = v482,
    age                       = v484,
    sex                       = v486,
    rdenom                    = v487,
    churchat                  = v489,
    sc.leav.cert              = v493,
    still.training            = v497,
    resp.curr.empl.status     = v513,
    nonemployment.status      = v514,
    resp.goldthorpe           = v531,
    spouse.goldthorpe         = v765,
    father.goldthorpe         = v923
    ))
```

The imported subset

```
classd.churchat.data
```

```
Data set with 47947 observations and 24 variables

   year east.west left.right vote.intention birthyear age    sex ...
1  1980      West                   CDU-CSU      1924  56   MALE ...
2  1980      West                       SPD      1912  68   MALE ...
3  1980      West                       SPD      1929  51   MALE ...
4  1980      West                       SPD      1936  44 FEMALE ...
5  1980      West                   CDU-CSU      1912  68 FEMALE ...
6  1980      West                       SPD      1960  20   MALE ...
7  1980      West      RIGHT        CDU-CSU      1917  63 FEMALE ...
8  1980      West                       SPD      1930  50 FEMALE ...
9  1980      West                       SPD      1906  74 FEMALE ...
10 1980      West                   CDU-CSU      1954  26   MALE ...
11 1980      West                   CDU-CSU      1933  47   MALE ...
12 1980      West                       SPD      1931  49 FEMALE ...
13 1980      West                       SPD      1934  46   MALE ...
14 1980      West                       SPD      1944  36   MALE ...
15 1980      West                       SPD      1952  28 FEMALE ...
16 1980      West                THE GREENS      1936  44   MALE ...
17 1980      West      RIGHT        CDU-CSU      1932  48 FEMALE ...
18 1980      West                       SPD      1934  46 FEMALE ...
19 1980      West                       SPD      1910  70 FEMALE ...
20 1980      West            WOULD NOT VOTE      1917  63   MALE ...
21 1980      West                   CDU-CSU      1920  60 FEMALE ...
22 1980      West                       SPD      1930  50   MALE ...
23 1980      West        *97       *REFUSED      1917  63   MALE ...
24 1980      West                       SPD      1928  52   MALE ...
25 1980      West                       SPD      1925  55 FEMALE ...
.. .... ......... .......... .............. ......... ... ...... ...
(25 of 47947 observations shown)
```

The imported subset

```
class(classd.churchat.data)
```

```
[1] "data.set"
attr(,"package")
[1] "memisc"
```

```
object.size(classd.churchat.data)
```

```
[1] 4883688
```

This is only 4.6 MB, the complete data were at least 228.6 MB. The complete data make even my 1GB office computer choke...

Data manipulation

Some more complex data setup

```
classd.churchat.data <- within(classd.churchat.data,{
  east.west <- relabel(east.west,
       "OLD FEDERAL STATES"="West",
       "NEW FEDERAL STATES"="East"
       )

  InEduc <- (year < 1986 & resp.curr.empl.status %in% c(6,10)) |
            (year > 1986 & nonemployment.status %in% c(1,5)) |
            (year == 1986 & sc.leav.cert == 7 | still.training %in% 1:3)
  respClass <- recode(resp.goldthorpe,
              "Agricultural"                = 1 <- c(6,10,12),
              "Petty Bourgeoisie"           = 2 <- 4:5,
              "Higher/Middle Service Class" = 3 <- 1,
              "Lower Service Class"         = 4 <- 2,
              "Routine Non-Manual"          = 5 <- c(3,11),
              "Technicians, Supervisors"    = 6 <- 7,
              "Skilled Workers"             = 7 <- 8,
              "Semi-/Unskilled Workers"     = 8 <- 9
          )
  spouseClass <- recode(spouse.goldthorpe,
              "Agricultural"                = 1 <- c(6,10,12),
              "Petty Bourgeoisie"           = 2 <- 4:5,
              "Higher/Middle Service Class" = 3 <- 1,
              "Lower Service Class"         = 4 <- 2,
              "Routine Non-Manual"          = 5 <- c(3,11),
              "Technicians, Supervisors"    = 6 <- 7,
              "Skilled Workers"             = 7 <- 8,
              "Semi-/Unskilled Workers"     = 8 <- 9
          )
  fatherClass <- recode(father.goldthorpe,
              "Agricultural"                = 1 <- c(6,10,12),
              "Petty Bourgeoisie"           = 2 <- 4:5,
              "Higher/Middle Service Class" = 3 <- 1,
              "Lower Service Class"         = 4 <- 2,
              "Routine Non-Manual"          = 5 <- c(3,11),
              "Technicians, Supervisors"    = 6 <- 7,
              "Skilled Workers"             = 7 <- 8,
              "Semi-/Unskilled Workers"     = 8 <- 9
          )
  dominance.matrix <- rbind(
      c(0,0,0,0,1,1,1,1), # what is dominated by Agricultural?
      c(0,0,0,0,1,1,1,1), # what is dominated by Petty Bourgeoisie ?
      c(1,1,0,1,1,1,1,1), # what is dominated by Higher/middle Service Class ?
      c(0,0,0,0,1,1,1,1), # what is dominated by Lower Service Class ?
      c(0,0,0,0,0,0,0,1), # what is dominated by Routine Non-Manual ?
      c(0,0,0,0,0,0,1,1), # what is dominated by Technicians and Supervisors?
      c(0,0,0,0,0,0,0,1), # what is dominated by Skilled Workers?
      c(0,0,0,0,0,0,0,0)  # what is dominated by Semi-/Unskilled Workers?
      )
  dominating.of <- function(x,y){
      x <- as.integer(x)
      y <- as.integer(y)
      ifelse(is.na(x) & y %in% 1:12,y,
          ifelse(x %in% 1:12 & is.na(y), x,
              ifelse(dominance.matrix[cbind(x,y)],x,y)))
      }
  classd <- ifelse(InEduc,fatherClass,dominating.of(spouseClass,respClass))
  labels(classd) <- labels(respClass)
  rm(InEduc,respClass,spouseClass,fatherClass,dominance.matrix,dominating.of)
  churchat4 <- recode(churchat,
                "At least once a week"  = 1 <- 1:2,
                "At least once a month" = 2 <- 3,
                "Less often"            = 3 <- 4:5,
                "Never"                 = 4 <- 6
                )
  vote.int <- recode(vote.intention,
                    "Other" = 90 <- c(5,20,30,90),
                    otherwise="copy"
                    )
  vote.int <- relabel(vote.int,
                "CDU-CSU"        = "CDU.CSU",
                "SPD"            = "SPD",
                "FDP"            = "FDP",
                "THE GREENS"     = "Greens",
                "PDS"            = "PDS",
                "WOULD NOT VOTE" = "No Voteint."
              )
  byear.categ <- cases(
                  "    -1919" = birthyear < 1920,
                  "1920-1929" = birthyear < 1930,
                  "1930-1939" = birthyear < 1940,
                  "1940-1949" = birthyear < 1950,
                  "1950-1959" = birthyear < 1960,
                  "1960-1969" = birthyear < 1970,
                  "1970-1979" = birthyear < 1980,
                  "1980+    " = birthyear >=1980
              )
  age.categ <- cases(
           "18-29" = age >= 18 & age < 30,
           "30-39" = age >= 30 & age < 40,
           "40-49" = age >= 40 & age < 50,
           "50-59" = age >= 50 & age < 60,
           "60+  " = age >= 60
          )
  measurement(birthyear) <- "interval"
  measurement(age) <- "ratio"

  SPD <- recode(vote.int,
              SPD = 1 <- 2,
            Other = 0 <- c(1,3:6,90)
            )
  description(SPD) <- "SPD vs. other"
  valid.values(SPD) <- 0:1
  measurement(SPD) <- "interval"

  SPDn <- recode(vote.int,
              SPD = 1 <- 2,
            Other = 0 <- c(1,3:6,90,91)
            )
  description(SPDn) <- "SPD vs. other or no vote"
  valid.values(SPDn) <- 0:1
  measurement(SPDn) <- "interval"

  labels(year) <- NULL
  decade <- ifelse(east.west=="West",
                  (year - min(year))/10 ,
                  (year - min(year[east.west=="East"]))/10
                  )
})
```

The `within` method for data sets

```
classd.churchat.data <- within(classd.churchat.data,{

})
```

- `within()` is a new S3 generic function present in *R* since version 2.6 — **much** more useful than `transform()`
- *memisc* (since verson 0.9) provides a `within()` method for `"data.set"` objects.
- Added functionality: all results of computations are automatically dropped if they not fit into the data set (in terms of mode or length), otherwise they are coerced into class `"item"`

Recoding of survey items

```
    respClass <- recode(resp.goldthorpe,
            "Agricultural"                = 1 <- c(6,10,12),
            "Petty Bourgeoisie"           = 2 <- 4:5,
            "Higher/Middle Service Class" = 3 <- 1,
            "Lower Service Class"         = 4 <- 2,
            "Routine Non-Manual"          = 5 <- c(3,11),
            "Technicians, Supervisors"    = 6 <- 7,
            "Skilled Workers"             = 7 <- 8,
            "Semi-/Unskilled Workers"     = 8 <- 9
        )
```

- **`recode()`** is generic, with methods for classes **`"item"`**, **`"factor"`**, and **`"vector"`**
- For items:
  - Left of "**=**": new value labels (optional)
  - Left of "**<-**": new codes
  - Right of "**<-**": old codes

Distinction of logical conditions

```
age.categ <- conditions(
         "18-29" = age >= 18 & age < 30,
         "30-39" = age >= 30 & age < 40,
         "40-49" = age >= 40 & age < 50,
         "50-59" = age >= 50 & age < 60,
         "60+  " = age >= 60
        )
```

Distinction of logical conditions

```
byear.categ <- conditions(
                "    -1919" = birthyear < 1920,
                "1920-1929" = birthyear < 1930,
                "1930-1939" = birthyear < 1940,
                "1940-1949" = birthyear < 1950,
                "1950-1959" = birthyear < 1960,
                "1960-1969" = birthyear < 1970,
                "1970-1979" = birthyear < 1980,
                "1980+    " = birthyear >=1980
            )
```

Distinction of logical conditions

```
genTable(range(birthyear,na.rm=TRUE)~byear.categ,
         data=classd.churchat.data)
```

```
 byear.categ
      -1919 1920-1929 1930-1939 1940-1949 1950-1959
1      1891      1920      1930      1940      1950
2      1919      1929      1939      1949      1959
 byear.categ
  1960-1969 1970-1979 1980+
1      1960      1970  1980
2      1969      1979  1987
```

Distinction of logical conditions

```
age.categ <- conditions(
         "18-29" = age >= 18 & age < 30,
         "30-39" = age >= 30 & age < 40,
         "40-49" = age >= 40 & age < 50,
         "50-59" = age >= 50 & age < 60,
         "60+  " = age >= 60
        )
```

- **`conditions()`** in this case results in a *factor*.
- Left of “**=**”: labels of the factor levels.
- Right of “**=**”: logical conditions that define the factor levels.
- Works like a series of **`ifelse`** or a vectorized version of **`switch`** (with different syntax!).

Codebooks

```
codebook(classd.churchat.data)
```

```
========================================================================

   year 'YEAR'

------------------------------------------------------------------------

   Storage mode: integer
   Measurement: interval

            Min:  1980.000
            Max:  2006.000
           Mean:  1993.104
       Std.Dev.:     7.697
       Skewness:     0.009
       Kurtosis:    -1.051
```

Codebooks

```
codebook(classd.churchat.data)
```

```
========================================================================

   east.west 'REGION OF INTERVIEW: WEST - EAST'

------------------------------------------------------------------------

   Storage mode: integer
   Measurement: nominal
   Missing values: 0

   Values and labels     N     Percent

          1   'West' 37714    78.7  78.7
          2   'East' 10233    21.3  21.3
```

Codebooks

```
codebook(classd.churchat.data)
```

```
========================================================================

   birthyear 'RESPONDENT: YEAR OF BIRTH'

------------------------------------------------------------------------

   Storage mode: integer
   Measurement: interval
   Missing values: 0, 9997-Inf

         Values and labels    N     Percent

       9997 M 'REFUSED'       13           0.0
       9999 M 'NO ANSWER'     56           0.1
              (unlab.vld.) 47878   100.0  99.9

            Min:  1891.000
            Max:  1987.000
           Mean:  1945.920
```

Codebooks

```
codebook(classd.churchat.data)
```

```
========================================================================

   byear.categ

------------------------------------------------------------------------

   Storage mode: integer
   Measurement: nominal

   Values and labels    N    Percent

     1   '    -1919' 4327    9.0  9.0
     2   '1920-1929' 5746   12.0 12.0
     3   '1930-1939' 7587   15.8 15.8
     4   '1940-1949' 8018   16.7 16.7
     5   '1950-1959' 9174   19.1 19.1
     6   '1960-1969' 8700   18.1 18.1
     7   '1970-1979' 3318    6.9  6.9
     8   '1980+    ' 1077    2.2  2.2
```

UNIVERSITÄT MANNHEIM

Codebooks

```
codebook(classd.churchat.data)
```

```
========================================================================

   SPDn 'SPD vs. other or no vote'

------------------------------------------------------------------------

   Storage mode: integer
   Measurement: nominal
   Valid values: 0, 1

   Values and labels     N     Percent

         1   'SPD'   12611    32.9  26.3
         0   'Other' 25773    67.1  53.8
        NA M          9563          19.9
```

Behind the scences

The class "data.set"

```
showClass("data.set")
```

```
Slots:

Name:           document
Class: character or NULL

Extends:
Class "data.frame", directly
Class "oldClass", by class "data.frame", distance 2
```

The class `"data.set"`

- `"data.set"` objects are a variant of data frames, especially desined to contain `"item"` objects.
- Such an object results from importing an SPSS or Stata file or a subset of it.
- `"data.set"` can be coerced into data frames. When that happens all `"item"` objects are converted into “ordinary” data vectors or factors.

The class **"item"**

The **"item"** class is used to represent items in a survey questionaire and the answers obtained for it.

```
showClass("item")
```

```
Slots:

Name:          value.labels         value.filter          measurement
Class: value.labels or NULL value.filter or NULL    character or NULL

Name:            annotation
Class:           annotation

Known Subclasses: "integer.item", "double.item", "character.item"
```

Value labels

- Via the **"value.labels"** slot, character string labels can be attached to certain values of an item.
- If an **"item"** object is coerced into a factor, the labels become the labels of the factor levels.
- Value labels of an item can be manipulated via **labels(x)** and **labels(x)<-y**

Value filters

- Value filters, the contents of the **`"value.filter"`** slot, allow to distinguish between “valid” values and “missing” values of an item.
- If an item is coerced into an “ordinary” vector or factor, “missing” values are automatically replaced by **`NA`**.
- **`"value.filter"`** objects come in three flavours, that is, classes: **`"valid.values"`**, **`"valid.range"`**, and **`"missing.values"`**.
- Value filters of an item can be manipulated via **`valid.values(x)`**, **`valid.range(x)`**, **`missing.values(x)`**, **`value.filter(x)`**, **`valid.values(x)<-y`**, etc.

Measurement level

- The “measurement level” of an survey item is represented by the **`"measurement"`** slot, which may be **`"nominal"`**, **`"ordinal"`**, **`"interval"`**, or **`"ratio"`**.
- The **`"measurement"`** slot of an item governs how it is converted if the containing **`"data.set"`** object is coerced into a data frame:
  - Items with "nominal" measurement level are changed into unordered factors,
  - "ordinal" items are changed into *ordered* factors.
  - "interval" and "ratio" scale items are changed into numeric vectors.
  - The measurement level of an item can be manipulated by **`measurement(x)`** and **`measurement(x)<-y`**.

Annotations

- The **"annotations"** slot can be used to attach arbitrary information to an **"item"** object. They can be manipulated using **annotation(x)** and **annotation(x)<-y**.
- These annotations should be a *named* character vector.
- Elements of such a character vector named as "description" or "wording" are special, however:
  - "description" strings correspond to SPSS and Stata's "variable labels". One can use **description(x)** and **description(x)<-y**.
  - "wording" strings are to contain the question wording of a survey item. One can use **wording(x)** and **wording(x)<-y** for this type of annotations.

Data analysis

Simple (conditional) sample statistics

```
genTable(range(birthyear,na.rm=TRUE)~byear.categ,
         data=classd.churchat.data)
```

```
 byear.categ
      -1919 1920-1929 1930-1939 1940-1949 1950-1959
1      1891      1920      1930      1940      1950
2      1919      1929      1939      1949      1959
 byear.categ
  1960-1969 1970-1979 1980+
1      1960      1970      1980
2      1969      1979      1987
```

Simple (conditional) sample statistics

```
genTable(range(birthyear,na.rm=TRUE)~byear.categ,
         data=classd.churchat.data)
```

```
aggregate(range(birthyear,na.rm=TRUE)~byear.categ,
         data=classd.churchat.data,sort=TRU)
```

Simple (conditional) sample statistics

```
aggregate(range(birthyear,na.rm=TRUE)~byear.categ,
         data=classd.churchat.data,sort=TRU)
```

```
      byear.categ  Min  Max
2           -1919 1891 1919
1       1920-1929 1920 1929
4       1930-1939 1930 1939
14      1940-1949 1940 1949
10      1950-1959 1950 1959
6       1960-1969 1960 1969
12122   1970-1979 1970 1979
2415    1980+     1980 1987
```

By, a convenient variant of **by**

```
glms <- By(~east.west,
   glm(SPDn~classd*decade,family="binomial",
           contrasts=list(
            classd=contr.treatment(levels(classd),
                                    base=7))),
   data=within(classd.churchat.data,
               SPDn <- as.integer(SPDn))
   )
```

- Instead of a list of factors **By** uses a formula.
- The second argument may be a function or an expression.
- It has an optional **data=** argument, the data source for both the formula and the expression evaluated.

By, a convenient variant of **by**

```
glms
```

```
east.west: West

Call:  glm(formula = SPDn ~ classd * decade,
           family = "binomial",
           contrasts = list(classd = contr.treatment(
                    levels(classd), base = 7)))

Coefficients:
                              (Intercept)
                                  0.00636
                       classdAgricultural
                                 -1.94385
                 classdPetty Bourgeoisie
                                 -1.34712
       classdHigher/Middle Service Class
                                 -0.86830
```

Collecting and displaying model estimates using **mtable**

```
mtab.glms <- mtable(glms,
               factor.style="($l)",
               summary.stats=c("Deviance","N"),
               coef.style="horizontal")
mtab.glms
```

Collecting and displaying model estimates using `mtable`

```
Calls:
West: glm(formula = SPDn ~ classd * decade, family = "binomial",
    contrasts = list(classd = contr.treatment(levels(classd), base = 7)))
East: glm(formula = SPDn ~ classd * decade, family = "binomial",
    contrasts = list(classd = contr.treatment(levels(classd), base = 7)))

=============================================================================================
                                                             West                East
---------------------------------------------------------------------------------------------
(Intercept)                                            0.006     (0.061)   -0.469***  (0.101)
Agricultural/Skilled Workers                          -1.944***  (0.242)    0.028     (0.265)
Petty Bourgeoisie/Skilled Workers                     -1.347***  (0.132)   -0.798***  (0.242)
Higher/Middle Service Class/Skilled Workers           -0.868***  (0.102)   -0.128     (0.163)
Lower Service Class/Skilled Workers                   -0.560***  (0.083)    0.061     (0.138)
Routine Non-Manual/Skilled Workers                    -0.280**   (0.107)    0.122     (0.199)
Technicians, Supervisors/Skilled Workers              -0.111     (0.106)    0.209     (0.212)
Semi-/Unskilled Workers/Skilled Workers                0.180     (0.112)   -0.415     (0.256)
decade                                                -0.329***  (0.046)   -0.703***  (0.135)
Agricultural/Skilled Workers x decade                  0.437**   (0.169)    0.051     (0.412)
Petty Bourgeoisie/Skilled Workers x decade             0.216*    (0.093)    0.652*    (0.266)
Higher/Middle Service Class/Skilled Workers x decade   0.170*    (0.072)    0.177     (0.218)
Lower Service Class/Skilled Workers x decade           0.217***  (0.061)    0.205     (0.182)
Routine Non-Manual/Skilled Workers x decade            0.106     (0.078)    0.226     (0.253)
Technicians, Supervisors/Skilled Workers x decade      0.022     (0.079)    0.030     (0.273)
Semi-/Unskilled Workers/Skilled Workers x decade      -0.330***  (0.082)    0.727*    (0.309)
---------------------------------------------------------------------------------------------
Deviance                                                    22124.9             5462.2
N                                                           17995               4542
=============================================================================================
```

Collecting and displaying model estimates using `mtable`

```
toLatex(mtab.glms)
```

	West		East	
(Intercept)	0.006	(0.061)	−0.469***	(0.101)
Agricultural/Skilled Workers	−1.944***	(0.242)	0.028	(0.265)
Petty Bourgeoisie/Skilled Workers	−1.347***	(0.132)	−0.798***	(0.242)
Higher/Middle Service Class/Skilled Workers	−0.868***	(0.102)	−0.128	(0.163)
Lower Service Class/Skilled Workers	−0.560***	(0.083)	0.061	(0.138)
Routine Non-Manual/Skilled Workers	−0.280**	(0.107)	0.122	(0.199)
Technicians, Supervisors/Skilled Workers	−0.111	(0.106)	0.209	(0.212)
Semi-/Unskilled Workers/Skilled Workers	0.180	(0.112)	−0.415	(0.256)
decade	−0.329***	(0.046)	−0.703***	(0.135)
Agricultural/Skilled Workers × decade	0.437**	(0.169)	0.051	(0.412)
Petty Bourgeoisie/Skilled Workers × decade	0.216*	(0.093)	0.652*	(0.266)
Higher/Middle Service Class/Skilled Workers × decade	0.170*	(0.072)	0.177	(0.218)
Lower Service Class/Skilled Workers × decade	0.217***	(0.061)	0.205	(0.182)
Routine Non-Manual/Skilled Workers × decade	0.106	(0.078)	0.226	(0.253)
Technicians, Supervisors/Skilled Workers × decade	0.022	(0.079)	0.030	(0.273)
Semi-/Unskilled Workers/Skilled Workers × decade	−0.330***	(0.082)	0.727*	(0.309)
Deviance	22124.9		5462.2	
N	17995		4542	

Outlook

Outlook

- Variants of "item" and "data.set" that reside in external files rather than in main memory and use contemporaneous **`largefile`** facilities allowing for files of size larger than 2 GB. (currently being implemented).
- Interfacing with Thomas Lumleys **`survey`** package (still a TODO).