

Merge Bases

Felipe Gonzalez Arango

23 de abril de 2019

Contents

Preparar environment para el analisis	1
Cargar paquetes	1
Importar bases de datos	2
Limpiar y organizar bases de datos	3
Merge	3
Items ACIDI/VOCA	4
Reconciliacion	4
Disculpas	4
Violencia	4
Rencor	4
Memoria Historica	5
Item de memoria historica con expectativas (Bicchieri)	5
Estadisticos de los items creados	5
MERGE con base de datos CEDE general	6
Creacion de indices importantes	6

Objetivo de este documento

Esta pagina tiene dos objetivos. El primero de ellos es importar las bases de datos y hacer los merge necesarios para sus analisis.

El segundo es crear los items importantes del cuestionario de reconciliacion del ACDIVOCA.

Preparar environment para el analisis

Script para organizar las bases de datos con las que vamos a trabajar en el proyecto (aqui se hace merge de las bases CEDE con el acdivoca y se definen las variables importantes)

Cargar paquetes

```
library(readstata13)
library(tidyr)
library(dplyr)
library(ggplot2)
library(data.table)
library(readxl)
library(doBy)
library(vtable)
library(plyr)
library(gridExtra)
library(grid)
```

```
library(purrr)
library(psych)
```

Importar bases de datos

```
#Set Working Directory
setwd("C:/Users/felig/Dropbox/Proyecto Juan Camilo")
rm(list=ls())

#Importar datos ACDIVOCA
datos_acdivoca <- read_excel('ACDI-VOCA - Base de Datos y Tabulados_F.xlsx')

#Ojo siempre abrir desde el dropbox de Felipe González porque se hizo el cambio de variables chaacter
datos_acdivoca_num <- as.data.table(
  read_excel('ACDI-VOCA - Base de Datos y Tabulados_F.xlsx',sheet=2))
#Importar datos CEDE
datos_cede_generales <- read.dta13("PANEL_CARACTERISTICAS_GENERALES.dta",
  encoding="latin1", fromEncoding="UTF-8",
  convert.factors = TRUE,
  nonint.factors=TRUE, generate.factors = TRUE)
datos_cede_conflicto <- read.dta13("PANEL_CONFLICTO_Y_VIOLENCIA.dta",
  encoding="latin1", fromEncoding="UTF-8",
  convert.factors = TRUE, generate.factors = TRUE,
  nonint.factors=TRUE)
datos_cede_salud <- read.dta13("PANEL_SALUD_Y_SERVICIOS.dta",
  encoding="latin1", fromEncoding="UTF-8",
  convert.factors = TRUE, generate.factors = TRUE,
  nonint.factors=TRUE)
datos_cede_educacion <- read.dta13("PANEL_DE_EDUCACION.dta",
  encoding="latin1", fromEncoding="UTF-8",
  convert.factors = TRUE, generate.factors = TRUE,
  nonint.factors=TRUE)
datos_cede_gobierno <- read.dta13("PANEL_BUEN_GOBIERNO.dta",
  encoding="latin1", fromEncoding="UTF-8",
  convert.factors = TRUE, generate.factors = TRUE,
  nonint.factors=TRUE)
datos_cede_tierras <- read.dta13("PANEL_AGRICULTURA_Y_TIERRA.dta",
  encoding="latin1", fromEncoding="UTF-8",
  convert.factors = TRUE, generate.factors = TRUE,
  nonint.factors=TRUE)
```

Aquí revisamos, por encima, el número de municipios que tiene cada base de datos

```
#Contar municipios dentro de las bases
length(unique(datos_cede_generales$codmpio)) # --> 1122 municipios

## [1] 1122
length(unique(datos_cede_conflicto$codmpio)) # --> 1123 municipios

## [1] 1123
```

```
length(unique(datos_cede_educacion$codmpio)) # --> 1122 municipios

## [1] 1122

length(unique(datos_cede_gobierno$codmpio)) # ---> 1123 municipios

## [1] 1123

length(unique(datos_cede_salud$codmpio)) # ---> 1258 municipios

## [1] 1258

length(unique(datos_cede_tierras$codmpio)) # ---> 1126 municipios

## [1] 1126

length(unique(datos_acdivoca$MPIO_IK))          # --> 44 municipios

## [1] 44
```

Limpiar y organizar bases de datos

```
#Limpiar el código ID de la base acdivoca. Hay ciertos municipios que empiezan con un '0' y hay que qu
datos_acdivoca$MPIO_IK <- ifelse(startsWith(datos_acdivoca$MPIO_IK,"0"),
                                sub("0{1}", "", datos_acdivoca$MPIO_IK,
                                    ignore.case=TRUE),
                                datos_acdivoca$MPIO_IK)
datos_acdivoca_num$MPIO_IK <- ifelse(startsWith(as.character(datos_acdivoca_num$MPIO_IK),"0"),
                                     sub("0{1}", "", datos_acdivoca_num$MPIO_IK, ignore.case=TRUE),
                                     datos_acdivoca_num$MPIO_IK)
setnames(datos_acdivoca, 'MPIO_IK', 'codmpio') #Para que el merge sea más fácil
setnames(datos_acdivoca_num, 'MPIO_IK', 'codmpio')
# Meter municipio en código
Nombre_cod_municipio <- datos_acdivoca[,c('Municipio','codmpio')]
Nombre_cod_municipio <- datos_acdivoca_num[,c('Municipio','codmpio')]
Nombre_cod_municipio$codmpio <- as.character(Nombre_cod_municipio$codmpio)
#Eliminar las filas repetidas
Nombre_cod_municipio <- Nombre_cod_municipio[!duplicated(Nombre_cod_municipio),]
```

Merge

Aquí hacemos el merge de las bases de datos. Mas específicamente, vamos a reducir las bases de datos del CEDE para que solo tengan los 44 municipios que tienen los datos del ACDIVOCA. El resto de municipios los podemos obviar.

```
#Merge CONFLICTO con 44 municipios (Así queda con los mismos municipios del acdivoca)
cede_conflicto_44 <- merge(datos_cede_conflicto, Nombre_cod_municipio, by='codmpio', all.y=T)
#Merge EDUCACION con 44 municipios (Así queda con los mismos municipios del acdivoca)
cede_educacion_44 <- merge(datos_cede_educacion, Nombre_cod_municipio, by='codmpio', all.y=T)
#Merge GENERAL con 44 municipios (Así queda con los mismos municipios del acdivoca)
cede_general_44 <- merge(datos_cede_generales, Nombre_cod_municipio, by='codmpio', all.y=T)
#Merge GOBIERNO con 44 municipios (Así queda con los mismos municipios del acdivoca)
cede_gobierno_44 <- merge(datos_cede_gobierno, Nombre_cod_municipio, by='codmpio', all.y=T)
#Merge SALUD con 44 municipios (Así queda con los mismos municipios del acdivoca)
cede_salud_44 <- merge(datos_cede_salud, Nombre_cod_municipio, by='codmpio', all.y=T)
#Merge TIERRAS con 44 municipios (Así queda con los mismos municipios del acdivoca)
```

```
cede_tierras_44 <- merge(datos_cede_tierras, Nombre_cod_municipio, by='codmpio', all.y=T)

# Convertir ano en numero
cede_conflicto_44$ano <- as.Date(cede_conflicto_44$ano, '%Y', na.rm=T)
cede_general_44$ano <- as.Date(cede_general_44$ano, '%Y', na.rm=T)
```

Items ACIDI/VOCA

Primero recodificamos los valores para que la respuesta “Totalmente en desacuerdo” tome el valor -2, la respuesta “En desacuerdo” tome el valor -1 y así sucesivamente hasta que la respuesta “Totalmente de acuerdo” tome el valor 2.

```
#Recodificar valores de los Ítems para que sean más interpretables (código para los no invertidos y l
datos_acdivoca_num$memoria <- datos_acdivoca_num$p075_1
datos_acdivoca_num <- datos_acdivoca_num %>%
  mutate_at(c("p024_1", "p024_2", "p024_3", "p024_4", "p024_5",
              "p024_6", "p024_7", "p024_8", "p053",
              "p058", "memoria"),
            recode, "1"=-2, "2"=-1, "4"=1, "5"=2, "3"=0) %>%
  mutate_at(c("p059"), recode, "1"=2, "2"=1,
            "4"=-1, "5"=-2, "3"=0)
```

Reconciliation

```
# Pregunta p024 -- Preguntas de Reconciliacion. Todos son Directos, se suman para sacar un indice más
datos_acdivoca_num$reconciliacion <- datos_acdivoca_num %>%
  select(p024_1, p024_2, p024_3, p024_4, p024_5, p024_6, p024_7, p024_8) %>%
  rowSums()
```

Disculpas

```
#Pregunta p053 -- Pregunta sobre las disculpas. Solo es una entonces good
datos_acdivoca_num$disculpas <- datos_acdivoca_num$p053
```

Violencia

```
#Pregunta p058 y p059. Ambas de Violencia. La segunda tiene escala invertida. La Primera no. Se hacen l
datos_acdivoca_num$violencia <- datos_acdivoca_num %>%
  select(p058, p059) %>%
  rowSums()
```

Rencor

Este índice es interesante. Se hace una formula para sacar un índice entre 0 y 15. Un valor bajito representa una persona con mucho rencor mientras que un valor alto representa a una persona con culpa. La formula es la siguiente:

$$Rencor = \frac{Otro_i}{Uno_i} \times \frac{(Otro_i + Uno_i)}{2}$$

Así: $0 < Rencor < 15$

#Pregunta p071a y p071b. Este es chevere. Indice de rencor. Se hace una formula para hablar de la "gravedad"

```
datos_acdivoca_num$rencor <- datos_acdivoca_num %>%
  select(p071_1, p071_2) %>%
  dplyr::mutate(rencor=(p071_2/p071_1)*((p071_1+p071_2)/2)) %>%
  select(rencor) %>%
  as.matrix(sapply(., as.numeric)) %>%
  as.numeric()
```

Memoria Historica

#Pregunta p075_1 es de memora historica.

```
datos_acdivoca_num$memoria <- datos_acdivoca_num$p075_1
```

Item de memoria historica con expectativas (Bicchieri)

Este item tiene la misma formula que la del indice de rencor solo que se hace con las preguntas de memoria historica. La formula es la misma.

$$Memoria_{Bicchieri} = \frac{Otro_i}{Uno_i} \times \frac{(Otro_i + Uno_i)}{2}$$

Asi: $0 < Memoria_{Bicchieri} < 15$

#indice de memoria histórica y expectativas cf. Bicchieri.

```
datos_acdivoca_num$memoria_expectativa <- datos_acdivoca_num %>%
  select(p075_1, p075_2) %>%
  dplyr::mutate(memoria_expectativa=(p075_2/p075_1)*((p075_1+p075_2)/2)) %>%
  select(memoria_expectativa) %>%
  as.matrix(sapply(., as.numeric)) %>%
  as.numeric()
```

Estadisticos de los items creados

Aqui exponemos las estadísticas descriptivas de los items que se acaban de crear

#Otra forma de ver los estadisticos

```
datos_acdivoca_num %>%
  select(memoria, reconciliacion, disculpas,
         violencia, rencor, memoria_expectativa) %>%
  describe(fast = T) %>%
  print()
```

##	vars	n	mean	sd	min	max	range	se
## memoria	1	11966	3.33	1.08	1.0	5	4.0	0.01
## reconciliacion	2	11966	3.41	6.22	-16.0	16	32.0	0.06
## disculpas	3	11966	0.80	1.02	-2.0	2	4.0	0.01
## violencia	4	11966	-1.23	1.46	-4.0	4	8.0	0.01
## rencor	5	11966	3.67	1.06	0.6	15	14.4	0.01
## memoria_expectativa	6	11966	3.38	1.25	0.6	15	14.4	0.01

MERGE con base de datos CEDE general

Finalmente, vamos a pegar esta base de datos (el CEDE general) con el resto porque esta presenta la informacion mas general de todos los municipios (aÃ±o, pib, poblacion, etc.)

```
#Collapse estadisticos por municipio

reconciliacion_agreg <- summaryBy(reconciliacion ~ Municipio,
                                datos_acdivoca_num, FUN=c(sum, median, mean, sd), na.rm=F )
disculpas_agreg <- summaryBy(disculpas ~ Municipio,
                             datos_acdivoca_num, FUN=c(sum, median, mean, sd), na.rm=F )
violencia_agreg <- summaryBy(violencia ~ Municipio,
                             datos_acdivoca_num, FUN=c(sum, median, mean, sd), na.rm=F )
rencor_agreg <- summaryBy(rencor ~ Municipio,
                          datos_acdivoca_num, FUN=c(sum, median, mean, sd), na.rm=F )
memoria_agreg <- summaryBy(memoria ~ Municipio,
                           datos_acdivoca_num, FUN=c(sum, median, mean, sd), na.rm=F )
memoria_expectativa_agreg <- summaryBy(memoria_expectativa ~ Municipio,
                                       datos_acdivoca_num, FUN=c(sum, median, mean, sd), na.rm=F )

# Merge con base de datos general -----
#Vamos a limpiar un poco la base general para que el merge sea mas facil de entender
cede_general_44 <- cede_general_44 %>%
  select(-c(Municipio)) %>% #Quitar variable que tienen las otras bases
  dplyr::rename(ano_base_general=ano) #Renombrar la variable ano asociada con la base general

#Merge con las bases
cede_conflicto_44 <- merge(cede_conflicto_44, cede_general_44, by="codmpio")
cede_educacion_44 <- merge(datos_cede_educacion, cede_general_44, by='codmpio')
cede_gobierno_44 <- merge(datos_cede_gobierno, cede_general_44, by='codmpio')
cede_salud_44 <- merge(datos_cede_salud, cede_general_44, by='codmpio')
cede_tierras_44 <- merge(datos_cede_tierras, cede_general_44, by='codmpio')
```

Por ultimo limpiamos las variables que ya no se vuelven a utilizar

```
#Limpiar environment

rm(datos_cede_conflicto,datos_cede_educacion,
    datos_cede_generales,datos_cede_gobierno,
    datos_cede_salud,datos_cede_tierras, cede_general_44)
```

Creacion de indices importantes