

Personal Sanitario No Enfermería

Alicia Perdiges Guerra

3 de mayo, 2021

Contents

1.PROCESAMIENTO DE LOS DATOS.

- En primer lugar leemos el fichero:

```
sanitarios_ne<-read.csv("C:/temp/PersonalSanitario_No_enfermeria.csv",sep= ",")
```

- Realicemos una breve inspección de los datos

```
str(sanitarios_ne)
```

```
## 'data.frame': 1600 obs. of 7 variables:
## $ TIME : int 2010 2010 2010 2010 2010 2010 2010 2010 2010 2010 ...
## $ GEO : Factor w/ 40 levels "Albania","Austria",...: 10 10 10 10 11 11 11 11 3 3 ...
## $ UNIT : Factor w/ 1 level "Number": 1 1 1 1 1 1 1 1 1 1 ...
## $ WSTATUS : Factor w/ 1 level "Practising": 1 1 1 1 1 1 1 1 1 1 ...
## $ ISCO08 : Factor w/ 4 levels "Dentists","Medical doctors",...: 2 1 3 4 2 1 3 4 2 1 ...
## $ Value : Factor w/ 733 levels ":", "1 102.00",...: 1 1 1 1 1 1 1 1 354 651 ...
## $ Flag.and.Footnotes: Factor w/ 6 levels "","b","be","e",...: 1 1 1 1 1 1 1 1 1 1 ...
```

```
colnames(sanitarios_ne) #Nombre de las variables
```

```
## [1] "TIME" "GEO" "UNIT"
## [4] "WSTATUS" "ISCO08" "Value"
## [7] "Flag.and.Footnotes"
```

```
nrow(sanitarios_ne) #Número de registros
```

```
## [1] 1600
```

```
ncol(sanitarios_ne) #Número de variables
```

```
## [1] 7
```

*Observamos las siguientes variables:

- **TIME**: variable cuantitativa. Indica el año en el que se ha realizado la medida, en este caso el valor de la variable “Value”. Se ha cargado bien como número entero.
- **GEO**: variable cualitativa. Indica el país o región en el que se ha realizado la medida. Se ha cargado bien como factor.
- **UNIT**: variable cualitativa. Indica la medida de la variable valor. Se ha cargado bien como factor.
- **WSTATUS**: variable cualitativa. Indica el estatus laboral del personal sanitario (no enfermería)
- **ISCO08**: Variable cualitativa. Indica si la variable “Value” se refiere a Médicos, Dentistas, Fisioterapeutas o Farmacéuticos.
- **Value**: Variable cuantitativa. Indica el número de profesionales sanitarios por países.
- **Flag.and.footnotes**. Notas sobre etiquetas. Eliminamos esta columna.

*Años de las mediciones:

```
unique(sanitarios_ne$TIME)
```

```
## [1] 2010 2011 2012 2013 2014 2015 2016 2017 2018 2019
```

*Países:

```
unique(sanitarios_ne$GEO)
```

```
## [1] European Union - 27 countries (from 2020)
## [2] European Union - 28 countries (2013-2020)
## [3] Belgium
## [4] Bulgaria
## [5] Czechia
## [6] Denmark
## [7] Germany (until 1990 former territory of the FRG)
## [8] Estonia
## [9] Ireland
## [10] Greece
## [11] Spain
## [12] France
## [13] France (metropolitan)
## [14] Croatia
## [15] Italy
## [16] Cyprus
## [17] Latvia
## [18] Lithuania
## [19] Luxembourg
## [20] Hungary
## [21] Malta
## [22] Netherlands
## [23] Austria
## [24] Poland
## [25] Portugal
## [26] Romania
## [27] Slovenia
## [28] Slovakia
## [29] Finland
## [30] Sweden
## [31] Iceland
## [32] Liechtenstein
## [33] Norway
## [34] Switzerland
## [35] United Kingdom
## [36] Montenegro
## [37] North Macedonia
## [38] Albania
## [39] Serbia
## [40] Turkey
## 40 Levels: Albania Austria Belgium Bulgaria Croatia Cyprus Czechia ... United Kingdom
```

*Unidad de las mediciones:

```
unique(sanitarios_ne$UNIT)
```

```
## [1] Number
## Levels: Number
```

*Variable que indica el estatus laboral del personal sanitario (no enfermería).

```
unique(sanitarios_ne$WSTATUS)
```

```
## [1] Practising
## Levels: Practising
```

*Variable que indica cómo agrupamos el personal de sanitario para las mediciones.

```
unique(sanitarios_ne$ISCO08)
```

```
## [1] Medical doctors Dentists Pharmacists Physiotherapists
## Levels: Dentists Medical doctors Pharmacists Physiotherapists
```

- Eliminamos la columna Fal.and.footnotes.

```
sanitarios_ne<-sanitarios_ne[,-7]
```

- Tendríamos que convertir la columna Value a numérico porque se ha cargado como factor y es erróneo. El resto de variables tienen el tipo correcto.

```
sanitarios_ne$Value<-as.character(sanitarios_ne$Value)
sanitarios_ne$Value<-(gsub(',', '.', sanitarios_ne$Value) )
sanitarios_ne$Value<-(gsub(' ', '', sanitarios_ne$Value) )
sanitarios_ne$Value<-as.numeric(sanitarios_ne$Value)
```

```
## Warning: NAs introducidos por coerción
```

- Comprobamos que valores tenemos en la columna **Value**:

```
table(sanitarios_ne$Value, useNA = "ifany")
```

```
##
##      8      9     10     12     15     16     20     23     24     44     46
##      1      1      1      1      2      1      1      1      1      1      1
##     48     50     51     52    104    111    116    119    120    124    135
##      3      1      3      1      1      1      1      1      2      1      1
##    138    144    149    153    155    157    160    166    171    181    182
##      1      1      1      2      1      1      1      1      1      1      1
##    184    186    187    188    190    195    197    201    206    214    223
##      1      1      1      1      2      1      1      1      1      1      1
##    232    270    272    274    278    279    283    285    289    299    301
##      1      1      1      1      2      1      1      1      1      1      1
##    335    374    380    382    397    406    416    422    436    441    454
##      1      2      1      2      1      1      1      1      1      1      1
##    460    469    476    493    506    550    558    581    605    624    750
##      1      1      1      1      1      1      1      1      2      1      1
##    761    772    783    785    798    816    829    832    839    842    847
##      1      1      1      1      1      1      1      1      1      1      1
##    868    876    882    883    939    948    955    956    963    976    1102
##      1      1      1      1      1      1      1      1      1      1      1
##   1118    1131    1142    1159    1168    1179    1189    1190    1193    1196    1198
##      1      1      2      1      1      1      1      1      1      1      1
##   1223    1239    1241    1244    1249    1257    1259    1260    1265    1277    1279
##      1      1      1      1      1      2      1      1      1      1      1
##   1280    1292    1295    1296    1319    1324    1326    1330    1337    1350    1356
##      1      1      1      1      1      1      1      1      1      1      1
##   1358    1361    1365    1373    1380    1381    1388    1392    1400    1402    1404
##      1      1      1      1      1      1      1      1      1      1      2
##   1411    1419    1421    1422    1433    1440    1445    1455    1456    1459    1466
##      1      1      1      1      1      1      1      1      1      1      3
##   1488    1490    1492    1538    1548    1566    1574    1589    1603    1612    1613
```

##	1	1	1	1	1	1	1	1	1	1	1
##	1636	1643	1650	1656	1683	1720	1743	1780	1850	1855	1925
##	1	1	2	1	1	1	1	1	2	1	1
##	1930	1932	1965	1987	2135	2399	2441	2456	2486	2529	2606
##	1	1	1	1	1	1	1	1	1	1	1
##	2628	2644	2669	2678	2685	2689	2754	2758	2787	2788	2803
##	1	1	1	1	1	1	1	1	1	1	1
##	2839	2849	2851	2879	2880	2931	2972	2979	2982	3004	3024
##	1	1	1	1	2	1	1	1	1	1	1
##	3042	3046	3062	3063	3072	3119	3120	3130	3158	3162	3196
##	1	1	1	1	1	1	1	1	1	1	1
##	3209	3223	3225	3308	3325	3327	3341	3347	3367	3433	3443
##	1	1	1	1	1	1	1	1	1	1	1
##	3465	3470	3506	3544	3566	3571	3578	3599	3619	3635	3640
##	1	1	1	1	1	1	1	1	1	1	1
##	3651	3696	3709	3710	3863	3948	3988	4009	4100	4109	4123
##	1	1	1	1	1	1	1	1	1	1	1
##	4141	4162	4180	4181	4191	4200	4208	4217	4234	4241	4247
##	1	1	1	1	1	1	1	1	1	1	1
##	4293	4319	4321	4337	4340	4342	4343	4361	4372	4380	4393
##	1	1	1	1	1	1	1	1	1	1	1
##	4395	4418	4419	4434	4450	4454	4468	4492	4527	4544	4548
##	1	1	1	2	1	1	1	1	1	1	1
##	4569	4590	4596	4605	4642	4685	4703	4743	4797	4853	4893
##	1	1	1	1	1	1	1	1	1	1	1
##	4906	4954	4973	4979	5009	5027	5121	5223	5228	5236	5257
##	1	1	1	1	1	1	1	1	1	1	1
##	5258	5292	5299	5324	5339	5416	5579	5609	5683	5700	5712
##	1	1	1	1	1	1	1	1	1	1	1
##	5741	5821	5823	5830	5863	5886	5889	5891	5918	5936	5941
##	1	1	1	1	1	1	1	1	1	1	1
##	5944	5963	5997	5999	6020	6061	6083	6104	6194	6203	6222
##	1	1	1	1	1	1	1	1	1	1	1
##	6224	6225	6246	6259	6295	6324	6327	6349	6367	6383	6389
##	1	1	1	1	1	1	1	1	1	1	1
##	6396	6408	6412	6423	6456	6517	6589	6591	6682	6724	6750
##	1	1	1	1	1	1	1	1	1	1	1
##	6870	6965	7039	7054	7139	7172	7263	7268	7280	7282	7293
##	1	1	1	1	1	1	1	1	1	1	1
##	7308	7320	7347	7353	7372	7426	7429	7461	7506	7524	7547
##	1	1	1	1	1	1	1	1	1	1	1
##	7557	7594	7605	7612	7674	7675	7689	7730	7744	7777	7809
##	1	1	1	1	1	1	1	1	1	1	1
##	7840	7841	7842	7844	7869	7900	7906	7921	7930	7955	7965
##	1	1	1	1	1	1	1	1	1	1	1
##	8001	8058	8068	8080	8108	8173	8225	8290	8291	8315	8379
##	1	1	1	1	1	1	1	1	1	1	1
##	8396	8410	8478	8516	8614	8705	8711	8730	8788	9340	9353
##	1	1	1	1	1	1	1	1	1	1	1
##	9435	9470	12215	12226	12304	12322	12367	12407	12450	12490	12491
##	1	1	1	1	1	1	1	1	1	1	1
##	12549	12603	12604	12605	12629	12631	12650	12772	12812	12845	12881
##	1	1	1	1	1	1	1	1	1	1	1
##	12887	12906	12959	13027	13033	13088	13280	13302	13308	13324	13331

##	1	1	1	1	1	1	1	1	1	1	1
##	13430	13504	13508	13534	13643	13779	13876	13885	14029	14075	14084
##	1	1	1	1	1	1	1	1	1	1	1
##	14248	14280	14490	14666	14846	15178	15359	15389	15497	15660	15962
##	1	1	1	1	1	1	1	1	1	1	1
##	16030	16090	16231	16262	16285	16366	16614	17025	17063	17104	17116
##	1	1	1	1	1	1	1	1	1	1	1
##	17511	17541	17759	20114	20670	20744	20773	20824	20853	21108	21238
##	1	1	1	1	1	1	1	1	1	1	1
##	21458	21614	21874	21882	22332	22754	22848	22902	23619	23688	24301
##	1	1	1	1	1	1	1	1	1	1	1
##	24583	25203	25538	26276	26290	26843	27446	27747	27963	28121	28384
##	1	1	1	1	1	1	1	1	1	1	1
##	28599	28686	28801	28891	29038	29268	29330	29492	29500	29625	29803
##	1	1	1	1	1	1	1	1	1	1	1
##	30037	30327	30486	30641	31313	31515	31748	31815	32182	32298	32543
##	1	1	1	1	1	1	1	1	1	1	1
##	32583	32681	32791	32999	33078	33218	33268	33353	33676	33785	34020
##	1	1	1	1	1	1	1	1	1	1	1
##	34348	34621	34760	34762	34834	34867	35000	35069	35568	35592	35708
##	1	1	1	1	1	1	1	1	1	1	1
##	35762	36324	36743	36940	37661	37769	38171	38624	38776	38817	39839
##	1	1	1	1	1	1	1	1	1	1	1
##	40105	40634	40641	40899	41268	41507	41740	41993	42084	42281	42302
##	1	1	1	1	1	1	1	1	1	1	1
##	42602	42898	42919	43000	43026	43126	43167	43333	43767	44002	44816
##	1	1	1	1	1	1	1	1	1	1	1
##	45596	46337	47038	47604	47610	48042	48559	48638	49200	49552	49731
##	1	1	1	1	1	1	1	1	1	1	1
##	50305	50456	50604	50656	50778	51153	51315	51431	52004	52100	52362
##	1	1	1	1	1	1	1	1	1	1	1
##	52430	52568	52828	53261	53480	53720	53867	54380	54493	54567	54807
##	1	1	1	1	1	1	1	1	1	1	1
##	55400	55633	55975	56167	56542	57329	57762	58329	58640	59144	59333
##	1	1	1	1	1	1	1	1	1	1	1
##	59704	60233	61730	63233	66427	66974	67812	68329	68474	68732	69145
##	1	1	1	1	1	1	1	1	1	1	1
##	69216	69269	69346	69520	69679	69939	69971	70025	70074	70136	70247
##	1	1	1	1	1	1	1	1	1	1	1
##	70277	70500	70786	70934	71093	71953	83201	84221	85025	85246	87687
##	1	1	1	1	1	1	1	1	1	1	1
##	88437	90284	91730	166250	169919	171747	173976	175033	176665	177352	177665
##	1	1	1	1	1	1	1	1	1	1	1
##	177731	178600	178833	179267	180444	180633	182534	185692	188166	188783	196767
##	1	1	1	1	1	1	1	1	1	1	1
##	196784	199920	201784	204129	206159	207789	209367	211162	212337	226384	230621
##	1	1	1	1	1	1	1	1	1	1	1
##	231481	233102	234918	235889	239642	240301	241512	242595	303645	311223	317390
##	1	1	1	1	1	1	1	1	1	1	1
##	325407	332695	338129	344755	351195	357401	<NA>				
##	1	1	1	1	1	1	847				

- Observamos que tenemos **847 valores perdidos**. Guardamos en la variable **idx** los índices de los registros con valores **NA** de la variable **Value**.

```
idx<-which(is.na(sanitarios_ne$Value))
length(idx)
```

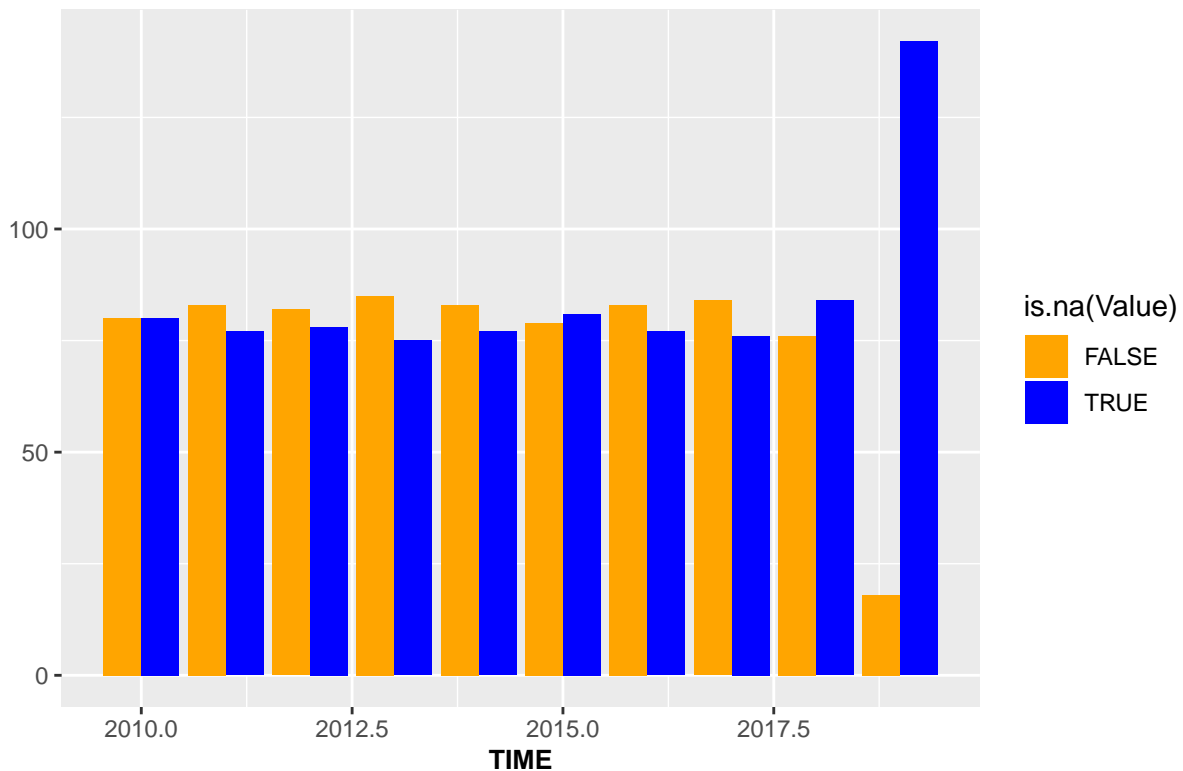
```
## [1] 847
```

- Grafiquemos la información que contiene la variable **Value**

```
library(ggplot2)
library(scales)
g = ggplot(sanitarios_ne, aes(TIME, fill=is.na(Value)) ) +
labs(title = "Valores Nulos")+ylab("") +
theme(plot.title = element_text(size = rel(2), colour = "blue"))

g+geom_bar(position="dodge") + scale_fill_manual(values = alpha(c("orange", "blue"), 1)) +
theme(axis.title.x = element_text(face="bold", size=10))
```

Valores Nulos



- En caso de detectar algún valor anómalo (en nuestro caso los NAS) en las variables tendríamos que realizar una imputación de esos valores o bien sustituyéndolos por la media o usando el algoritmo KNN (k-Nearest Neighbour) con los 3 vecinos más cercanos usando la distancia que consideremos, en este caso usaremos Gower(Mediana), por ser una medida más robusta frente a extremos.

```
library(VIM)
```

```
## Loading required package: colorspace
```

```
## Loading required package: grid
```

```
## VIM is ready to use.
```

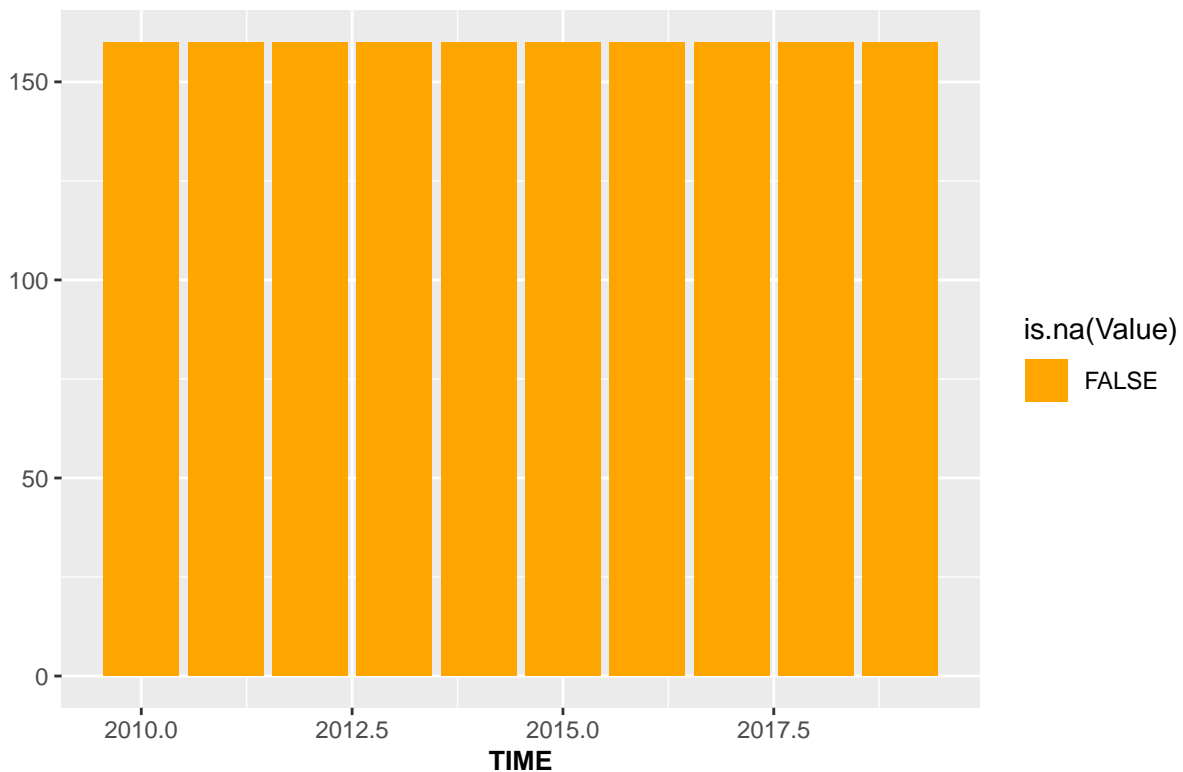
```
## Suggestions and bug-reports can be submitted at: https://github.com/statistikat/VIM/issues
##
## Attaching package: 'VIM'
## The following object is masked from 'package:datasets':
##
##      sleep
output<-kNN(sanitarios_ne, variable=c("Value"),k=3)
sanitarios_ne<-output
```

- Comprobamos que no tenemos valores nulos después de la imputación

```
g = ggplot(sanitarios_ne, aes(TIME, fill=is.na(Value)) ) +
labs(title = "Valores Nulos")+ylab("") +
theme(plot.title = element_text(size = rel(2), colour = "blue"))

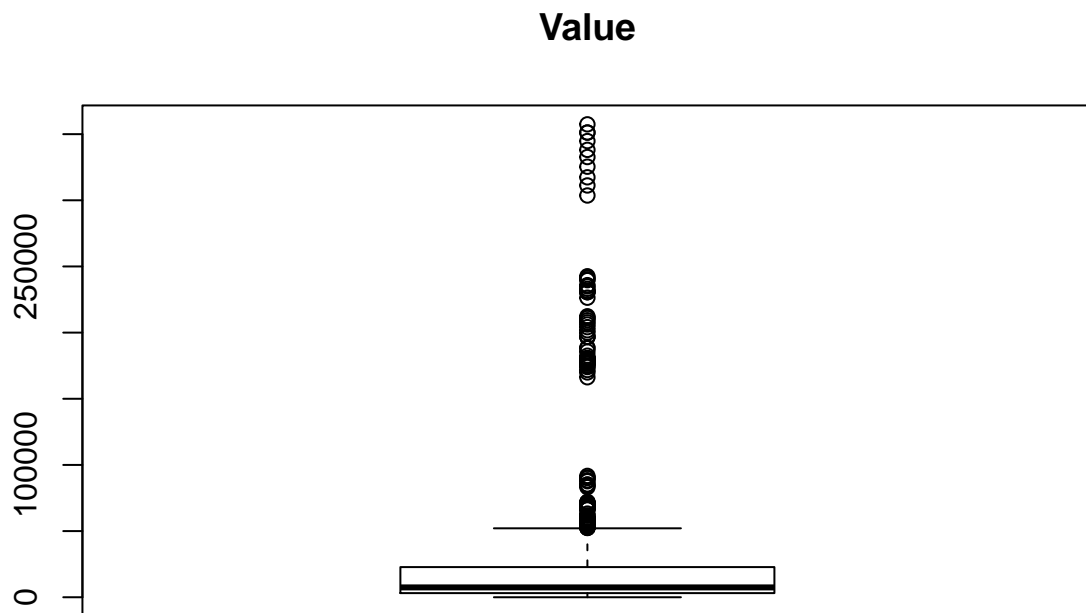
g+geom_bar(position="dodge") + scale_fill_manual(values = alpha(c("orange", "blue"), 1)) +
theme(axis.title.x = element_text(face="bold", size=10))
```

Valores Nulos



- Con el siguiente gráfico, observaremos que la variable **Value** tiene outliers o valores extremos

```
boxplot(sanitarios_ne$Value, main="Value")
```



- Por otro lado, revisamos para el resto de columnas si tenemos valores NA.(desconocidos o perdidos)

```
table(sanitarios_ne$TIME, useNA = "ifany")
```

```
##
## 2010 2011 2012 2013 2014 2015 2016 2017 2018 2019
## 160 160 160 160 160 160 160 160 160 160
```

```
table(sanitarios_ne$GEO, useNA = "ifany")
```

```
##
##                Albania
##                 40
##                Austria
##                 40
##                Belgium
##                 40
##                Bulgaria
##                 40
##                Croatia
##                 40
##                Cyprus
##                 40
##                Czechia
##                 40
##                Denmark
##                 40
##                Estonia
```


##		40
##	European Union - 27 countries (from 2020)	
##		40
##	European Union - 28 countries (2013-2020)	
##		40
##	Finland	
##		40
##	France	
##		40
##	France (metropolitan)	
##		40
##	Germany (until 1990 former territory of the FRG)	
##		40
##	Greece	
##		40
##	Hungary	
##		40
##	Iceland	
##		40
##	Ireland	
##		40
##	Italy	
##		40
##	Latvia	
##		40
##	Liechtenstein	
##		40
##	Lithuania	
##		40
##	Luxembourg	
##		40
##	Malta	
##		40
##	Montenegro	
##		40
##	Netherlands	
##		40
##	North Macedonia	
##		40
##	Norway	
##		40
##	Poland	
##		40
##	Portugal	
##		40
##	Romania	
##		40
##	Serbia	
##		40
##	Slovakia	
##		40
##	Slovenia	
##		40
##	Spain	

```
##
##
##
##
##
##
##
##
##
##
```

```
table(sanitarios_ne$UNIT, useNA = "ifany")
```

```
##
## Number
## 1600
```

```
table(sanitarios_ne$WSTATUS, useNA = "ifany")
```

```
##
## Practising
## 1600
```

```
table(sanitarios_ne$ISCO08, useNA = "ifany")
```

```
##
## Dentists Medical doctors Pharmacists Physiotherapists
## 400 400 400 400
```

Observamos que no existen ahora valores perdidos después de la imputación. La suma de las cantidades de cada variable, suman el total.

- Finalmente, creamos un fichero con toda la información corregida.

```
write.csv(sanitarios_ne, file="PersonalSanitario_No_enfermeria_clean.csv", row.names = FALSE)
```