

Médicos por Especialidad

Alicia Perdices Guerra

3 de mayo, 2021

Contents

1.PROCESAMIENTO DE LOS DATOS.

- En primer lugar leemos el fichero:

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

- Realicemos una breve inspección de los datos

```
str(medicos_e)
```

```
## 'data.frame': 1800 obs. of 6 variables:
## $ TIME : int 2010 2010 2010 2010 2010 2010 2010 2010 2010 2010 ...
## $ GEO : Factor w/ 36 levels "Albania","Austria",...: 3 3 3 3 3 4 4 4 4 4 ...
## $ UNIT : Factor w/ 1 level "Number": 1 1 1 1 1 1 1 1 1 1 ...
## $ MED_SPEC : Factor w/ 5 levels "General paediatricians",...: 3 2 4 5 1 3 2 4 5 1 ...
## $ Value : Factor w/ 1139 levels ":", "0.00", "1 001.00",...: 210 210 2 354 66 806 723 532 4 ...
## $ Flag.and.Footnotes: Factor w/ 8 levels "","b","bd","be",...: 1 1 1 1 1 1 1 1 1 1 ...
```

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

```
## [1] "TIME" "GEO" "UNIT"
## [4] "MED_SPEC" "Value" "Flag.and.Footnotes"
```

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

```
## [1] 1800
```

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

```
## [1] 6
```

*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.
- **MED_SPEC**: variable cualitativa. Indica la especialidad médica.
- **Value**: Variable cuantitativa. Indica el número de médicos por especialidad médica y por países.
- **Fal.and.footnotes**. Notas sobre etiquetas. Eliminamos esta columna.

*Años de las mediciones:

```
unique(medicos_e$TIME)
```

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

*Países:

```
unique(medicos_e$GEO)
```

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

*Unidad de las mediciones:

```
unique(medicos_e$UNIT)
```

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

*Variable que indica la especialidad médica

```
unique(medicos_e$MED_SPEC)
```

```
## [1] Generalist medical practitioners
## [2] General practitioners
## [3] Other generalist medical practitioners
## [4] Specialist medical practitioners
## [5] General paediatricians
## 5 Levels: General paediatricians ... Specialist medical practitioners
```

- Eliminamos la columna Fal.and.footnotes.

```
medicos_e<-medicos_e[,-6]
```

- 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.

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

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

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

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

```
##
##      0      5      6     10     11     12     13     14     15     16
##    71      9      1      2      3      1      1      2      5      1
##   18     19     20     21     23     24     25     26     27     34
##      1      1      1      1      2      3      2      2      2      1
##   35     37     44     46     48     55     59     63     67     68
##      1      5      1      1      1      1      2      1      1      1
##   74     76     79     81     82     83     84     85     87     90
##      2      1      1      2      1      3      2      2      2      1
##   91     92     96     98    101    102    104    106    117    119
##      2      2      1      1      1      1      1      1      1      1
##  120    122    123    130    132    133    145    146    147    154
##      1      2      1      1      1      1      1      1      1      1
##  157    161    163    165    166    167    168    169    172    173
##      2      1      4      1      1      1      2      2      3      2
##  179    181    182    183    184    185    188    189    190    191
##      2      1      3      1      2      2      4      1      2      2
##  195    197    204    209    213    215    218    220    222    228
##      2      1      2      3      2      2      2      2      1      1
##  232    238    241    242    245    246    248    249    253    255
##      1      1      2      1      2      3      3      2      2      1
##  256    262    271    273    278    284    303    315    324    335
##      1      1      1      3      2      2      3      2      1      2
##  336    340    346    349    360    373    379    384    394    396
##      2      2      2      2      2      1      2      1      3      4
##  397    398    400    408    414    415    419    421    424    426
##      1      1      2      2      2      1      1      1      2      1
##  428    430    431    435    436    441    445    451    453    455
##      1      1      1      1      1      2      1      1      1      1
##  457    463    467    469    472    473    485    489    490    492
##      1      1      2      1      1      1      1      2      1      1
##  494    496    498    505    509    512    524    534    540    574
##      1      1      2      2      1      1      1      2      1      1
##  587    605    614    620    648    658    659    660    673    680
##      1      1      1      1      1      1      1      1      2      2
##  692    696    697    703    706    707    708    710    714    716
##      1      1      1      2      1      1      1      1      1      1
##  720    722    725    726    735    744    748    751    752    754
##      1      1      1      1      2      1      1      1      1      2
##  760    778    786    788    796    798    802    804    817    818
```

##	2	1	2	2	2	2	1	1	1	1
##	827	828	833	835	842	855	857	864	869	875
##	1	1	2	1	1	2	1	1	1	1
##	891	898	912	919	925	926	928	932	933	937
##	1	2	2	1	1	1	1	1	1	1
##	940	946	950	959	960	964	971	979	980	984
##	1	1	1	2	1	1	1	1	1	1
##	990	992	995	1001	1002	1004	1010	1015	1016	1025
##	1	1	1	1	1	1	1	2	2	1
##	1026	1029	1043	1044	1048	1049	1052	1053	1057	1058
##	1	1	1	1	1	1	2	1	1	1
##	1059	1061	1064	1069	1070	1072	1073	1076	1081	1083
##	1	1	1	1	1	1	1	1	1	1
##	1085	1097	1104	1105	1107	1113	1122	1125	1127	1132
##	1	1	1	1	1	1	1	1	1	1
##	1143	1151	1158	1178	1183	1185	1204	1213	1217	1233
##	1	1	1	1	1	1	1	1	1	1
##	1234	1237	1246	1252	1262	1275	1284	1285	1292	1296
##	1	1	1	1	1	1	1	1	1	1
##	1309	1311	1313	1315	1323	1328	1332	1335	1338	1340
##	1	2	1	1	1	2	2	1	2	1
##	1350	1352	1356	1357	1359	1361	1364	1378	1379	1383
##	1	1	1	2	2	1	1	1	2	1
##	1391	1394	1397	1400	1405	1411	1412	1414	1420	1423
##	1	3	1	1	2	2	2	3	1	1
##	1424	1425	1426	1429	1448	1459	1467	1473	1475	1482
##	1	1	1	1	2	1	1	1	1	1
##	1484	1500	1532	1542	1548	1556	1563	1584	1586	1603
##	1	1	1	1	1	1	1	1	1	2
##	1608	1610	1618	1634	1635	1645	1648	1681	1692	1725
##	1	1	1	1	1	1	1	1	1	1
##	1735	1770	1773	1817	1818	1822	1823	1843	1856	1861
##	1	2	1	1	1	1	2	1	1	1
##	1869	1890	1911	1914	1919	1933	1934	1943	1967	1973
##	1	1	1	1	1	1	1	1	2	1
##	1983	2016	2017	2020	2032	2038	2039	2063	2070	2076
##	1	1	1	1	2	1	1	1	1	1
##	2085	2087	2093	2112	2135	2200	2226	2228	2246	2266
##	1	1	1	1	1	1	1	1	1	1
##	2278	2286	2291	2298	2302	2305	2311	2329	2371	2381
##	1	1	1	1	1	1	1	1	1	1
##	2383	2402	2422	2423	2426	2452	2478	2481	2492	2501
##	2	1	1	1	1	1	1	1	1	1
##	2534	2541	2552	2560	2561	2566	2577	2598	2600	2625
##	1	1	2	1	1	1	1	1	1	1
##	2632	2636	2638	2655	2668	2679	2697	2706	2709	2710
##	1	1	1	1	1	1	1	1	1	1
##	2769	2819	2828	2836	2865	2873	2877	2884	2937	2962
##	1	1	1	1	1	1	1	1	1	1
##	2971	2975	3000	3011	3085	3103	3188	3248	3279	3292
##	1	1	1	1	1	1	1	1	1	1
##	3293	3299	3309	3318	3321	3326	3335	3343	3350	3361
##	1	1	1	1	1	2	1	1	1	2
##	3369	3371	3391	3406	3425	3440	3456	3487	3494	3502

##	1	1	1	1	1	1	1	1	1	1
##	3507	3508	3510	3523	3531	3585	3605	3631	3642	3647
##	1	1	1	1	1	1	1	1	1	1
##	3692	3701	3710	3716	3718	3773	3787	3891	3921	3923
##	1	1	1	1	1	1	1	1	1	1
##	3942	3948	3950	3963	3965	4008	4012	4035	4039	4058
##	1	1	1	1	1	1	1	1	1	1
##	4081	4083	4087	4133	4135	4140	4187	4199	4204	4218
##	1	1	1	1	1	2	1	1	1	1
##	4220	4258	4272	4278	4307	4317	4322	4323	4374	4390
##	1	1	1	2	1	1	1	1	1	2
##	4393	4396	4407	4409	4418	4433	4437	4455	4477	4495
##	1	1	1	1	1	1	1	1	1	1
##	4514	4517	4521	4525	4531	4561	4566	4571	4572	4576
##	1	1	1	1	1	1	1	1	1	1
##	4582	4595	4616	4629	4649	4650	4651	4691	4697	4714
##	1	1	1	1	1	1	1	1	1	1
##	4721	4747	4748	4758	4761	4781	4811	4812	4821	4839
##	1	1	1	1	1	1	1	1	1	1
##	4851	4880	4883	4884	4900	4909	4910	4925	4945	4956
##	1	1	1	1	1	1	1	1	1	1
##	4981	4987	5008	5012	5014	5018	5043	5058	5066	5087
##	1	1	1	1	1	1	1	1	1	1
##	5097	5119	5128	5136	5143	5206	5210	5282	5305	5423
##	1	1	1	1	1	1	1	1	1	1
##	5497	5533	5753	5907	5973	6011	6061	6084	6106	6191
##	1	1	1	2	2	1	1	1	2	1
##	6201	6239	6270	6346	6350	6358	6380	6388	6411	6440
##	2	1	2	2	2	1	1	2	2	1
##	6480	6495	6507	6523	6534	6550	6595	6604	6609	6614
##	1	1	1	1	1	1	1	1	1	1
##	6627	6637	6644	6668	6712	6724	6730	6752	6780	6792
##	1	1	1	1	1	1	1	1	1	1
##	6835	6837	6849	6855	6861	6869	6888	6977	6981	7031
##	1	1	1	1	1	1	1	1	2	1
##	7066	7096	7107	7108	7125	7163	7166	7167	7232	7287
##	1	1	1	1	1	1	1	1	1	1
##	7298	7346	7362	7369	7371	7386	7401	7420	7429	7466
##	1	1	1	4	2	2	1	1	1	1
##	7490	7516	7585	7665	7666	7724	7773	7795	7843	7873
##	1	1	1	1	1	1	1	1	1	1
##	7918	7965	7970	8057	8075	8134	8151	8214	8270	8272
##	1	1	1	1	1	1	1	1	1	1
##	8285	8312	8322	8356	8387	8418	8439	8441	8449	8484
##	1	1	1	1	1	1	1	1	1	1
##	8514	8526	8557	8610	8618	8715	8746	8763	8765	8900
##	1	1	1	1	1	1	1	1	1	1
##	8954	8955	8965	8986	9039	9084	9110	9170	9191	9203
##	1	1	1	1	1	1	1	1	1	1
##	9225	9339	9358	9401	9432	9443	9566	9570	9571	9597
##	1	1	1	1	1	1	1	1	1	1
##	9601	9628	9683	9728	9760	9784	9786	9800	9841	9857
##	1	1	1	1	1	1	1	1	1	1
##	9862	9907	9910	9916	9939	9941	9960	9997	10011	10049

##	1	1	1	1	1	2	1	1	1	1
##	10083	10098	10127	10135	10182	10184	10212	10355	10445	10454
##	1	1	1	1	1	1	1	1	1	1
##	10464	10498	10564	10568	10648	10693	10775	10781	10877	11079
##	1	1	1	1	1	1	1	1	1	1
##	11200	11243	11274	11288	11306	11366	11723	11728	11896	12026
##	1	1	1	1	1	1	1	1	1	1
##	12029	12037	12067	12070	12133	12184	12189	12222	12228	12280
##	1	1	1	1	1	1	1	1	2	1
##	12290	12333	12334	12363	12389	12475	12483	12529	12560	12576
##	2	1	1	2	1	1	2	1	2	1
##	12610	12645	12655	12693	12735	12837	12846	12854	12892	12929
##	1	1	1	2	1	1	1	1	1	2
##	12992	13047	13178	13192	13219	13227	13322	13403	13617	13621
##	2	1	2	1	1	1	1	1	1	1
##	13657	13744	13745	13758	13766	13834	13888	13924	14023	14093
##	1	1	1	1	1	1	1	1	1	1
##	14130	14210	14275	14509	14525	14531	14615	14618	14633	14689
##	1	1	1	1	1	1	1	1	1	1
##	14792	14891	14893	15006	15020	15091	15211	15270	15285	15460
##	1	1	1	1	1	1	1	1	1	1
##	15601	15682	15740	15768	15820	15908	15923	16052	16741	16968
##	1	1	1	1	1	1	1	1	1	1
##	17086	17114	17197	17211	17270	17282	17354	17472	17475	17484
##	1	1	1	1	1	1	1	1	1	1
##	17495	17651	17718	17958	18210	18350	18650.5	18945	19153	19198
##	1	1	1	1	1	1	1	1	1	1
##	19492	19534	19536	19639	19831	20094	20103	20108	20269	20313
##	1	1	1	1	1	2	1	1	1	1
##	20389	20532	20564	20848	20881	20901	20948.5	20956	20993	21107
##	1	1	1	1	1	1	1	1	1	1
##	21170	21174	21424	21600	21606	21630	21646.5	21658	21917	21926
##	1	1	1	1	1	1	1	1	1	1
##	21954.5	21974	22062	22125	22180	22196	22300	22465	22518	22669
##	1	1	1	1	1	1	1	1	1	1
##	22744	22866	22902	22936.5	22949	23246	23266	23301	23459	23464
##	1	1	1	1	1	1	1	1	1	1
##	23502	23702	23725	23748	23814	23817.5	24054	24110	24135	24218
##	1	1	1	1	1	1	1	1	1	1
##	24248	24296	24498	24521	24905	24931	24940	25123	25303	25387
##	1	1	1	1	1	1	1	1	1	1
##	25611	25615	25642	25903	26100	26490	26814	27060	27076	27664
##	1	1	1	1	1	1	1	1	1	1
##	27730	28394	28557	28708	28994	29761	30292	30376	30802	31080
##	1	1	1	1	1	1	1	1	1	1
##	31183	31238	31405	32151	32193	32568	32654	33057	33173	33435
##	1	1	1	1	1	1	1	1	1	1
##	34000	34073	34441	34525	34620	34675	34722	34735	34797	34880
##	1	1	1	1	2	1	2	2	2	2
##	35017	35061	35378	35524	35621	35798	36905	37980	39537	39552
##	2	2	2	1	1	2	1	1	1	1
##	40396	40764	41040	41077	41116	41401	41542	41783	41840	42309
##	1	1	2	2	1	2	2	1	2	1
##	42987	43731	44279	44442	44646	44655	44937	45203	45437	45878

```
##      2      1      1      1      2      1      1      1      1      1
## 46061 46107 47999 48641 48688 49569 49824 49838 49875 50008
##      1      2      2      2      2      2      2      2      2      2
## 50080 50169 50595 51096 51520 51596 52839 52944 52998 53114
##      2      2      2      2      1      1      1      1      1      1
## 53258 53463 53525 53610 53691 53856 53994 54063 55161 56739.5
##      1      1      1      1      1      1      1      1      1      1
## 56831 56902 56961.5 57360 57652.5 57892.5 58474.5 58939.5 59399 60214
##      1      1      1      1      1      1      1      1      1      1
## 60255 61205 62129 62864 63522 64583 65717 69696 72380 72400
##      1      1      1      1      1      1      1      1      1      1
## 73819 73943 74113 74376 75252 75390 75678 78601 78608 79847
##      1      1      1      1      1      1      1      1      1      1
## 80226 81411 82370 82757 84189 88371 92735 93140 93283 93944
##      1      1      1      1      1      1      1      1      1      1
## 94074 94262 94322 94328 94889 94923 96613 98720 101998 103262
##      1      1      1      1      1      1      1      1      1      2
## 104440 106780 107542 108501 108759 108870 110185 111837 112464 112699
##      1      1      1      1      1      1      1      1      1      1
## 113206 113527 115039 115392 116273 117141 117414 118033 118356 120479
##      1      1      1      1      1      1      1      1      1      1
## 121561 122061 124110 127025 130719 132657 136097 138726 145367 173440
##      1      1      1      1      1      1      1      1      1      1
## 176627 177625 179492 181393 182426 185579 187303 187821 189481 229532
##      1      1      1      1      1      1      1      1      1      1
## 235971 242000 246799 252469 259528 264908 269784 274644 <NA>
##      1      1      1      1      1      1      1      1      390
```

- Observamos que tenemos **390 valores perdidos**. Guardamos en la variable **idx** los índices de los registros con valores **NA** de la variable **Value**.

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

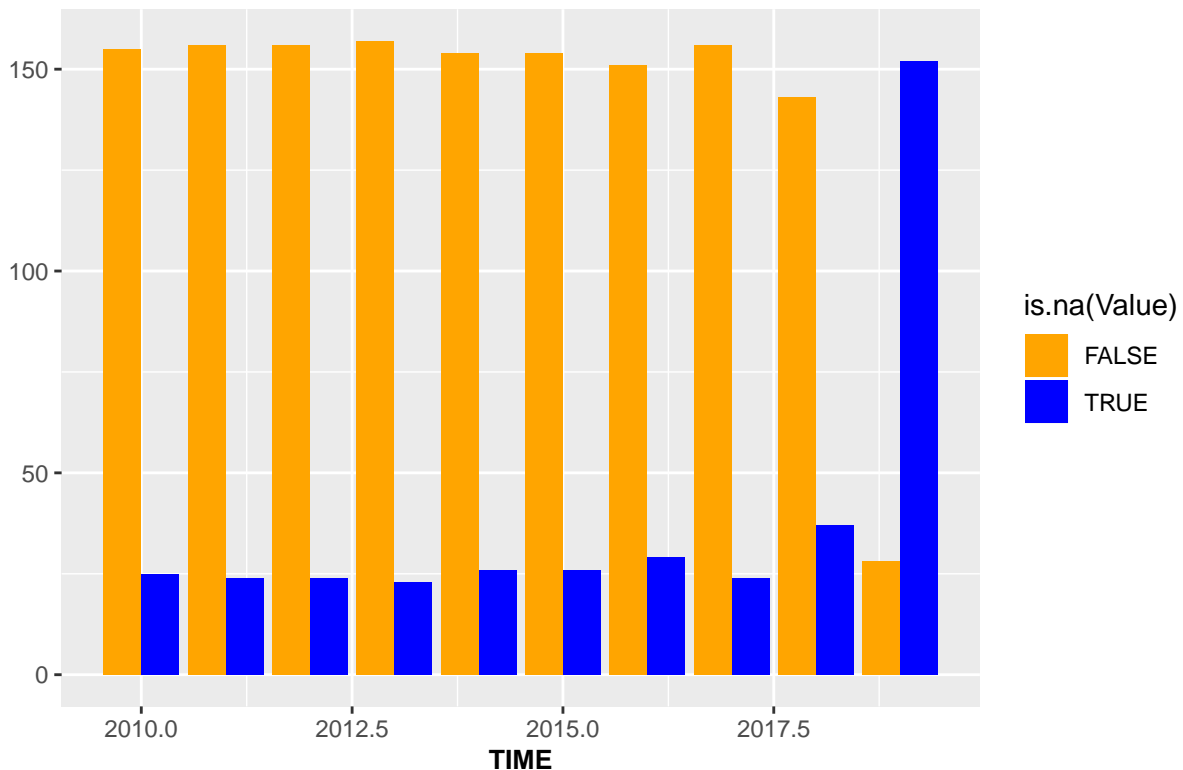
```
## [1] 390
```

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

```
library(ggplot2)
library(scales)
g = ggplot(medicos_e, 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(medicos_e, variable=c("Value"),k=3)
```

```
medicos_e<-output
```

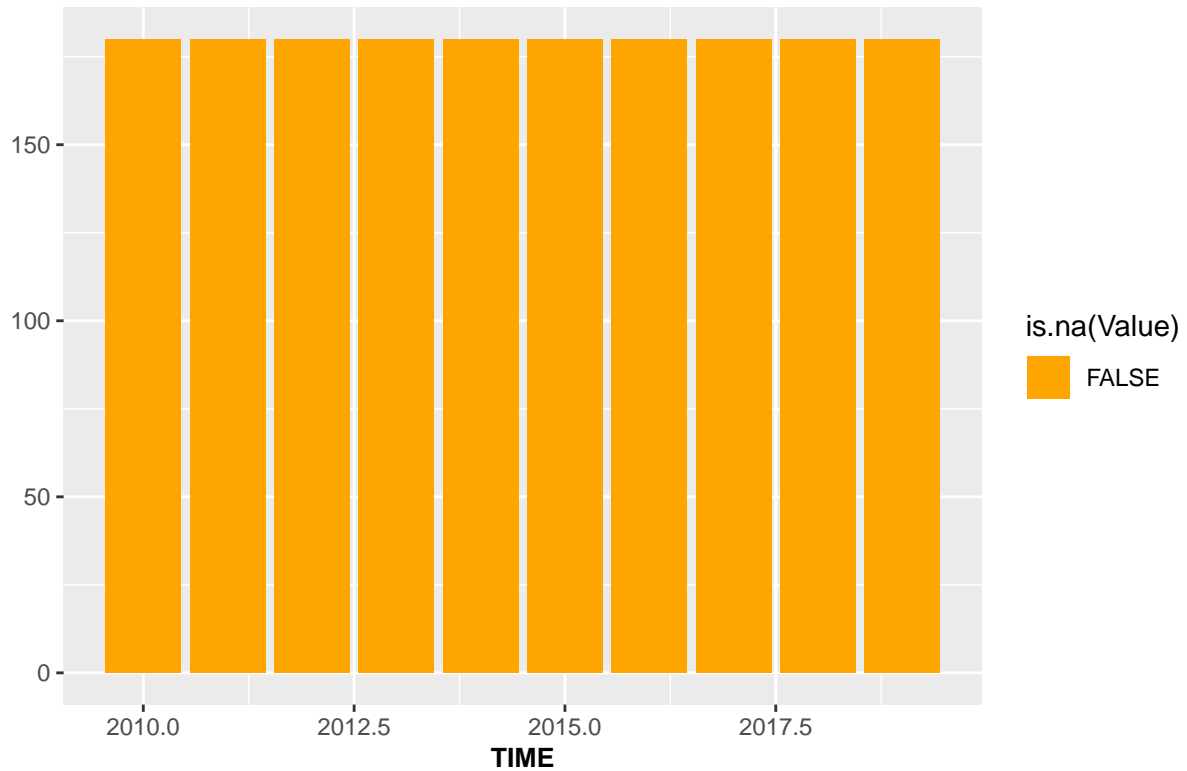
- Comprobamos que no tenemos valores nulos después de la imputación

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



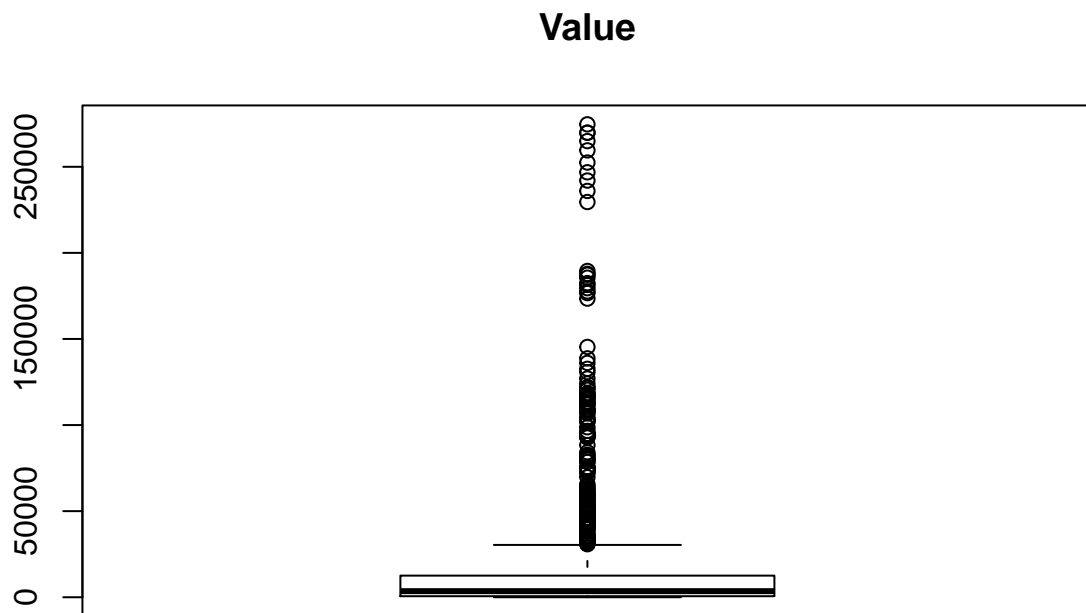
```
g+geom_bar(position="dodge") + scale_fill_manual(values = alpha(c("orange", "blue"), 1)) +  
theme(axis.title.x = element_text(face="bold", size=10))
```

Valores Nulos



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

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



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

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

```
##
## 2010 2011 2012 2013 2014 2015 2016 2017 2018 2019
## 180 180 180 180 180 180 180 180 180 180
```

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

```
##
##
## Albania
## 50
## Austria
## 50
## Belgium
## 50
## Bulgaria
## 50
## Croatia
## 50
## Cyprus
## 50
## Czechia
## 50
## Denmark
## 50
## Estonia
```

##	50
##	Finland
##	50
##	France
##	50
##	Germany (until 1990 former territory of the FRG)
##	50
##	Greece
##	50
##	Hungary
##	50
##	Iceland
##	50
##	Ireland
##	50
##	Italy
##	50
##	Latvia
##	50
##	Liechtenstein
##	50
##	Lithuania
##	50
##	Luxembourg
##	50
##	Malta
##	50
##	Montenegro
##	50
##	Netherlands
##	50
##	North Macedonia
##	50
##	Norway
##	50
##	Poland
##	50
##	Portugal
##	50
##	Romania
##	50
##	Serbia
##	50
##	Slovenia
##	50
##	Spain
##	50
##	Sweden
##	50
##	Switzerland
##	50
##	Turkey
##	50
##	United Kingdom

```
##
```

50

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

```
##
```

```
## Number
```

```
## 1800
```

```
table(medicos_e$MED_SPEC, useNA = "ifany")
```

```
##
```

```
##          General paediatricians
```

```
General practitioners
```

```
##          360
```

```
360
```

```
## Generalist medical practitioners Other generalist medical practitioners
```

```
##          360
```

```
360
```

```
## Specialist medical practitioners
```

```
##          360
```

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(medicos_e, file="Medicos_x_especialidad_clean.csv", row.names = FALSE)
```