

## CONTENIDO

<b>1. Serie de tiempo con pre-procesamiento</b>	<b>2</b>
1.1. Lectura de todas las tablas (DBF) con for . . . . .	2
1.2. Lapply . . . . .	2
1.3. Mapply . . . . .	3
1.4. Tapply . . . . .	5
1.5. Serie de tiempo . . . . .	5
1.6. Creación de pdf . . . . .	7

## 1. Serie de tiempo con pre-procesamiento

```
setwd("C:\\Users\\81799\\Downloads\\tmod_vic-main")
dir()

## [1] "descriptor_11_12.csv"      "descriptor_13_20.csv"
## [3] "FD_2011.xls"              "FD_2012.xls"
## [5] "FD_2013.xlsx"             "FD_2014.pdf"
## [7] "FD_2015.pdf"              "FD_2016.pdf"
## [9] "FD_2017.pdf"              "FD_2018.pdf"
## [11] "FD_2019.pdf"              "FD_2020.pdf"
## [13] "tmod_vic_2011.DBF"         "tmod_vic_2012.DBF"
## [15] "tmod_vic_2013.dbf"         "tmod_vic_2014.dbf"
## [17] "tmod_vic_2015.dbf"         "ts_delitos_por_tipo.pdf"

library(foreign) #Sirve para leer archivos .dbf
library(tidyverse) #Carpeta para ciencia de datos en R

## - Attaching packages ----- tidyverse 1.3.1 -
## v ggplot2 3.3.5      v purrr 0.3.4
## v tibble 3.1.6       v dplyr 1.0.8
## v tidyr 1.2.0        v stringr 1.4.0
## v readr 2.1.2        v forcats 0.5.1
## - Conflicts ----- tidyverse_conflicts() -
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()      masks stats::lag()

tabla1 <- read.dbf("tmod_vic_2011.DBF") #Pero quiero leer todos los archivos juntos
```

### 1.1. Lectura de todas las tablas (DBF) con for

```
setwd("C:\\Users\\81799\\Downloads\\tmod_vic-main")
indices <- c(13:17) #13,14,15,16 y 17
tablas <- list()
for(i in indices){
  tablas[[i]] <- read.dbf(dir()[i]) #Leer los archivos del 13 al 17
}
dim(tablas[[13]]) #Me dice cuantas filas y columnas tiene el DataFrame

## [1] 27186    138

dim(tablas[[14]]) #Dice cuantas filaas y columnas tiene el DataFrame

## [1] 32493    134
```

### 1.2. Lapply

```
setwd("C:\\Users\\81799\\Downloads\\tmod_vic-main")
lectura <- function(x){
  read.dbf(dir()[x])
}
tablas2 <- lapply(indices, lectura) #indices =c(13:17)
```

```

dim(tablas2[[1]])

## [1] 27186 138

class(tablas2[[1]]$FAC_DEL) #Nos dice la clase de la columna FAC_DEL

## [1] "factor"

tablas2[[1]]$FAC_DEL[1] #Nos da el 1er valor de la tabla 1, fila 1 columna FAC_DEL

## [1] 446
## 3301 Levels: 100 1000 1001 1002 1003 1004 1005 1006 1007 1008 1009 101 ... 999

class(tablas2[[1]]$BPCOD[1])

## [1] "factor"

#Convierto de clase factor a numeric
tablas2[[1]]$FAC_DEL <- as.numeric(as.character(tablas2[[1]]$FAC_DEL ))
class(tablas2[[1]]$FAC_DEL )

## [1] "numeric"

summary(tablas2[[1]]$BP1_1) #Nos indica los meses que ocurrieron los delitos

## 01 02 03 04 05 06 07 08 09 10 11 12 88 99
## 1569 2128 2171 1680 2058 2623 1936 2132 2143 2365 3110 3148 75 48

tablas2[[1]] <- tablas2[[1]][!(tablas2[[1]]$BP1_1 %in% c("88","99")),] #Me quedo con los meses
summary(tablas2[[1]]$BP1_1)

## 01 02 03 04 05 06 07 08 09 10 11 12 88 99
## 1569 2128 2171 1680 2058 2623 1936 2132 2143 2365 3110 3148 0 0

class(tablas2[[1]]$BP1_1)

## [1] "factor"

tablas2[[1]]$BP1_1 <- as.factor(paste(tablas2[[1]]$BP1_1,2010,sep = "/" )) #aparecera 01/2010,
class(tablas2[[1]]$BP1_1)

## [1] "factor"

```

### 1.3. Mapply

```

setwd("C:\\Users\\81799\\Downloads\\tmod_vic-main")
preprocesado <- function(x,y){
  x[, "FAC_DEL"] <- as.numeric(as.character(x$FAC_DEL)) #Convierte a numerico
  x <- x[!(x$BP1_1 %in% c("88","99")),] #Quita las que tengan BP1_1, 88 y 99
  x$BP1_1 <- as.factor(paste(x$BP1_1,y,sep="/")) #Fecha mes/año
  x #Es como nuestro return (x es la tabla en este caso)
}
years <- c(2010:2014)
tmod <- mapply(preprocesado, tablas2, years)

```

```
length(tmod) #Nos dice cuantos elementos que tiene

## [1] 5

summary(tmod[[1]]$BP1_1) #Aquí nos da el 2010

## 01/2010/2010 02/2010/2010 03/2010/2010 04/2010/2010 05/2010/2010 06/2010/2010
##          1569          2128          2171          1680          2058          2623
## 07/2010/2010 08/2010/2010 09/2010/2010 10/2010/2010 11/2010/2010 12/2010/2010
##          1936          2132          2143          2365          3110          3148

summary(tmod[[2]]$BP1_1) #Aquí nos da el 2011

## 01/2011 02/2011 03/2011 04/2011 05/2011 06/2011 07/2011 08/2011 09/2011 10/2011
##    2025    2682    2636    2028    2317    2637    2116    2402    2368    2707
## 11/2011 12/2011
##    3451    3654

setwd("C:\\Users\\81799\\Downloads\\tmod_vic-main")
library(tidyverse)
descriptor_1 <- read_tsv("descriptor_11_12.csv", col_names = T)

## Rows: 14 Columns: 2
## - Column specification -----
## Delimiter: "\t"
## chr (2): CODIGO, DESCRIPCION
##
## i Use 'spec()' to retrieve the full column specification for this data.
## i Specify the column types or set 'show_col_types = FALSE' to quiet this message.
descriptor_2 <- read_tsv("descriptor_13_20.csv", col_names = T)

## Rows: 15 Columns: 2
## - Column specification -----
## Delimiter: "\t"
## chr (2): CODIGO, DESCRIPCION
##
## i Use 'spec()' to retrieve the full column specification for this data.
## i Specify the column types or set 'show_col_types = FALSE' to quiet this message.
dim(tmod[[1]])

## [1] 27063 138

tmod[[1]]$BPCOD[1]

## [1] 02
## Levels: 01 02 03 04 05 06 07 08 09 10 11 12 13 14

tmod[[1]] <- left_join(tmod[[1]], descriptor_1, by = c("BPCOD" = "CODIGO"))
#Al aplicar left_join() es importante que sean de las mismas clases
class(tmod[[1]]$BPCOD)

## [1] "character"

class(descriptor_1$CODIGO)

## [1] "character"

dim(tmod[[1]]) #Se agrega una columna con el nombre de descripción

## [1] 27063 139
```

```
tmod[[2]] <- left_join(tmod[[2]],descriptor_1, by = c("BPCOD" = "CODIGO"))
tmod[[3]] <- left_join(tmod[[3]],descriptor_2, by = c("BPCOD" = "CODIGO"))
tmod[[4]] <- left_join(tmod[[4]],descriptor_2, by = c("BPCOD" = "CODIGO"))
tmod[[5]] <- left_join(tmod[[5]],descriptor_2, by = c("BPCOD" = "CODIGO"))
```

## 1.4. Tapply

Quiero fijarme por tipo de delito y por mes cuantos delitos se cometieron

```
#tapply sirve para hacer cálculos discriminando
class(tmod[[1]]$FAC_DEL)

## [1] "numeric"

del_2010 <- tapply(tmod[[1]]$FAC_DEL, list(tmod[[1]]$BP1_1, tmod[[1]]$DESCRIPCION), function(x){
  sum(x, na.rm = T)
})
del_2010 <- data.frame(rownames(del_2010), del_2010, TOTAL = rowSums(del_2010, na.rm = TRUE))
del_2010
```

##	rownames.del_2010.	Agr_Fis	Amen	Ext	Fr_B	Fr_C	Hos_S			
##	01/2010/2010	01/2010/2010	70230	85385	245756	56955	71850	50137		
##	02/2010/2010	02/2010/2010	96353	104262	410498	94557	77040	46877		
##	03/2010/2010	03/2010/2010	78964	109599	424317	88604	79622	53580		
##	04/2010/2010	04/2010/2010	64842	121895	339422	52496	70104	40405		
##	05/2010/2010	05/2010/2010	65898	124840	437295	70488	54332	96767		
##	06/2010/2010	06/2010/2010	115559	165510	618637	87871	97116	47516		
##	07/2010/2010	07/2010/2010	47570	107707	420127	73659	56875	22688		
##	08/2010/2010	08/2010/2010	98186	129341	480523	86887	87382	59704		
##	09/2010/2010	09/2010/2010	144103	175413	445688	92440	87414	34333		
##	10/2010/2010	10/2010/2010	108805	179364	478331	94806	93562	67033		
##	11/2010/2010	11/2010/2010	134299	267006	643469	105570	116568	94940		
##	12/2010/2010	12/2010/2010	154719	249842	524302	96292	105874	110147		
##	O_R	Otros	R_call	R_cas	R_p_veh	R_t_veh	Sec	Vio_S	TOTAL	
##	01/2010/2010	92804	17561	278813	104359	134364	37583	2723	3211	1251731
##	02/2010/2010	100378	19130	425417	136996	188933	34825	7879	558	1743703
##	03/2010/2010	103440	21702	419158	129562	195342	42767	5140	448	1752245
##	04/2010/2010	92143	25916	319260	101353	138640	38798	611	164	1406049
##	05/2010/2010	85808	20819	420240	116049	174195	30865	6186	2793	1706575
##	06/2010/2010	151853	26442	462150	174909	223569	31488	7651	840	2211111
##	07/2010/2010	88539	30497	396331	119606	180594	31790	15436	755	1592174
##	08/2010/2010	87497	30584	416537	103546	194254	35046	19424	NA	1828911
##	09/2010/2010	106006	20822	471568	113817	198895	26679	4502	2267	1923947
##	10/2010/2010	129461	23604	435171	152976	219623	31475	22042	1624	2037877
##	11/2010/2010	165697	37630	691967	181641	287167	55120	12937	2339	2796350
##	12/2010/2010	217324	45235	872855	163438	224132	38191	9969	126	2812446

## 1.5. Serie de tiempo

```
library(base)
library(zoo)
```

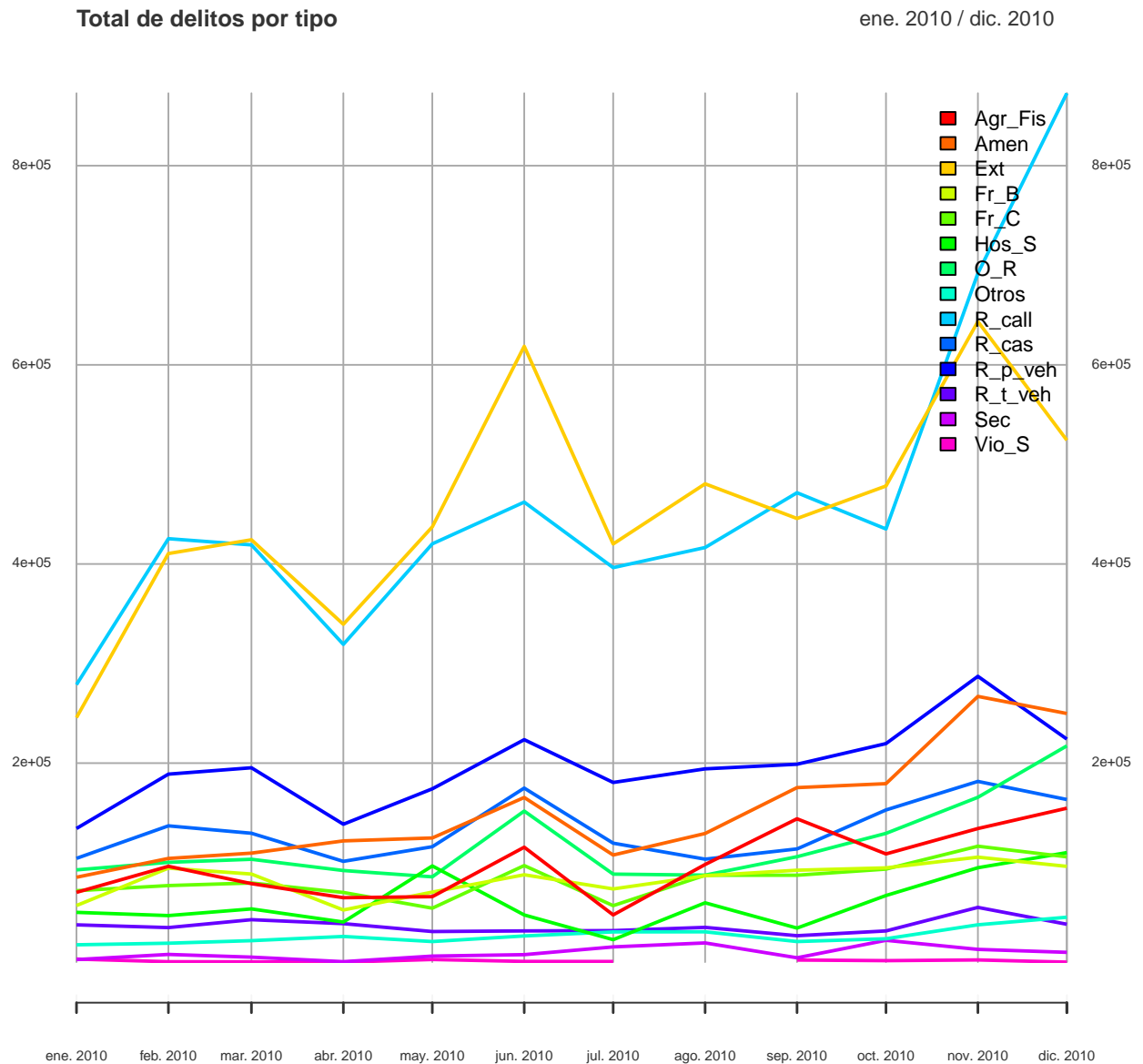
```
##
## Attaching package: 'zoo'
## The following objects are masked from 'package:base':
##
##      as.Date, as.Date.numeric

library(tidyverse)
library(dplyr)
library(xts)

##
## Attaching package: 'xts'
## The following objects are masked from 'package:dplyr':
##
##      first, last

# del_2010[, -1] Quitar la 1er columna
# as.yearmon() es una función que solo acepta meses y año
del_2010 <- xts(del_2010[, -1],
               order.by = as.yearmon(del_2010[, 1], format = "%m/%Y") )

color <- rainbow(ncol(del_2010))
plot.xts(del_2010[, -15], col = color,
         legend.loc = "topright",
         cex = 0.5,
         cex.axis = 0.7,
         main = "Total de delitos por tipo")
```



## 1.6. Creación de pdf

```
pdf("ts_delitos_por_tipo.pdf")
plot.xts(del_2010[,-15], col = color,
         legend.loc = "topright",
         cex = 0.5,
         cex.axis = 0.7,
         main = "Total de delitos por tipo")
dev.off()

## pdf
## 2
```