Article

Estudio del gasto y duración media de los viajes de los turistas extranjeros en distintas comunidades autónomas

Alejandro León Líndez[1], Adrian Lizzadro Pla[2], Marta Medina Muñiz[3]

[1] Máster en Ciencia de Datos; alelin@alumni.uv.es
[2] Máster en Ciencia de Datos; alizpla@alumni.uv.es
[3] Máster en Ciencia de Datos; memuiz@alumni.uv.es
* Correspondence: email@email.com; Tel.: +XX-000-00-0000.

Simple Summary: Resumen del trabajo

Abstract: abstract

Keywords: keyword 1; keyword 2; keyword 3 (list three to ten pertinent keywords specific to the article, yet reasonably common within the subject discipline.).

1. Introducción

2. Carga de librerías e importación del fichero

Antes de comenzar, eliminamos todas las variables guardadas.

```
rm(list = ls())  # Borrado de todas las variables
```

```
library(readr)  # Librería para importación de datos
library(dplyr)  # Librería para arreglo de datos
```

```
##
## Adjuntando el paquete: 'dplyr'
## The following objects are masked from 'package:stats':
##
##     filter, lag
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(tidyr)  # Librería para arreglo de datos
library(ggplot2)  # Librería para gráficas
```

Importamos los datos

```
gastos <- read_delim("data/Gasto_turistas_internacionales_segun_comunidad_paisresi
    delim = ";", escape_double = FALSE, trim_ws = TRUE)
```

```
## Rows: 3072 Columns: 6
## -- Column specification --------------------------------------------------------
## Delimiter: ";"
## chr (5): País de residencia, Total Nacional y CCAA, Tipo de dato, Gastos y d...
## dbl (1): Periodo
##
## 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 messag
```

Citation: Estudio del gasto y duración media de los viajes de los turistas extranjeros en distintas comunidades autónomas. *Data* **2023**, *1*, 0. https://doi.org/

Received:
Revised:
Accepted:
Published:

3. Preparación de los datos

3.1. Tranformación a tidydata

Observamos que las variables no están correctamente colocadas en columnas.

```
tipos_datos <- unique(gastos$`Gastos y duración media de los viajes`)
tipos_datos  # Vemos cuantas variables tenemos que transformar a columnas
```

```
## [1] "Gasto total"                    "Gasto medio por persona"
## [3] "Gasto medio diario por persona" "Duración media de los viajes"
```

```
gastos1 <- subset(gastos, gastos$`Gastos y duración media de los viajes` ==
    tipos_datos[1])
colnames(gastos1)[colnames(gastos1) == "Total"] <- "Gasto_total"
gastos1 <- subset(gastos1, select = -`Gastos y duración media de los viajes`)
```

```
gastos2 <- subset(gastos, gastos$`Gastos y duración media de los viajes` ==
    tipos_datos[2])
colnames(gastos2)[colnames(gastos2) == "Total"] <- "Gasto_medio_persona"
gastos2 <- subset(gastos2, select = -`Gastos y duración media de los viajes`)
```

```
gastos3 <- subset(gastos, gastos$`Gastos y duración media de los viajes` ==
    tipos_datos[3])
colnames(gastos3)[colnames(gastos3) == "Total"] <- "Gasto_medio_diario_persona"
gastos3 <- subset(gastos3, select = -`Gastos y duración media de los viajes`)
```

```
gastos4 <- subset(gastos, gastos$`Gastos y duración media de los viajes` ==
    tipos_datos[4])
colnames(gastos4)[colnames(gastos4) == "Total"] <- "Duracion_media"
gastos4 <- subset(gastos4, select = -`Gastos y duración media de los viajes`)
```

```
# Compruebo que en todas las columnas salvo la última todos
# las filas son iguales para poder hacer merge de los dos
# datasets correctamente
any(gastos1[1:length(nrow(gastos1) - 1)] != gastos2[1:length(nrow(gastos2) -
    1)])
```

```
## [1] FALSE
```

```
any(gastos3[1:length(nrow(gastos1) - 1)] != gastos4[1:length(nrow(gastos2) -
    1)])
```

```
## [1] FALSE
```

```
# Uno los dos datasets en un único dataset con el que
# trabajar

gastos12 <- merge(gastos1, gastos2, by = c(colnames(gastos1[1:length(gastos1) -
    1])))
gastos34 <- merge(gastos3, gastos4, by = c(colnames(gastos3[1:length(gastos3) -
    1])))
```

```
# Compruebo que en todas las columnas salvo la última todos
# las filas son iguales para poder hacer merge de los dos
# datasets correctamente
any(gastos12[1:length(nrow(gastos12) - 2)] != gastos34[1:length(nrow(gastos34) -
    2)])
```

```
## [1] FALSE
```

```
# Uno los dos datasets
datos <- merge(gastos12, gastos34, by = c(colnames(gastos12[1:4])))
```

Eliminamos filas que no vamos a emplear en el estudio (tasa de variación) y columnas redudantes o que contienen información irrelevante,

```
unique(datos$`Tipo de dato`)
```

```
## [1] "Dato base"               "Tasa de variación anual"
```

```
# Nos quedamos únicamente con los datos base, quitando las
# tasas de variación
datos <- subset(datos, datos$`Tipo de dato` == unique(datos$`Tipo de dato`)[1])
# Quito la columna irrelevante
datos <- subset(datos, select = -`Tipo de dato`)
```

```
rm(gastos, gastos1, gastos2, gastos3, gastos34, gastos4, gastos12)  # Eliminamos
# algunos variables auxiliares innecesarias a posteriori
```

3.2. Transformación de clases

Todas las variables han sido importadas como tipo carácter.Vamos a transformar las variables Gasto_total, Gasto_medio_persona, Gasto_medio_diario_persona y Duracion_media a numérico, donde previamente transformamos la cadena de carácteres a una que sea interpretable como número para poder aplicar la función as.numeric() correctamente.

```
# lapply(datos,class)
```

```
# Quitar punto de miles
datos[, 4:ncol(datos)] <- lapply(datos[, 4:ncol(datos)], function(x) gsub("\\.",
    "", x))
# Sustituir coma decimal por punto decimal
datos[, 4:ncol(datos)] <- lapply(datos[, 4:ncol(datos)], function(x) gsub(",",
    ".", x))
# Transformar a numerico
datos[, 4:ncol(datos)] <- lapply(datos[, 4:ncol(datos)], function(x) as.numeric(x)
# Comprobamos la clase lapply(datos,class)
```

Comprobamos si al realizar la transformación se han introducido datos NA.

```
any(is.na(datos))
```

```
## [1] FALSE
```

A continuación, transformamos a factor las variables País de residencia y Comunidades Autónomas.

```r
# unique(datos$`País de residencia`) Transformacion a
# factor de paises de residencia
datos$`País de residencia` <- as.factor(datos$`País de residencia`)
```

Quitamos algunas filas irrelevantes que contienen datos que consisten en la suma total de los datos de la columna de Comunidades Autónomas para el análisis exploratorio. Sin embargo, los recogeremos en una variable a parte para presentarlo en una visualización más compleja.

```r
# Guardamos los datos totales por si resultan útiles en el
# futuro
datos_totales <- subset(datos, datos$`País de residencia` ==
    "Total" | datos$`Total Nacional y CCAA` == "Total")
```

```r
# Quitar filas de total de columna comunidades autonomas
datos_total_por_residencia <- subset(datos, datos$`Total Nacional y CCAA` ==
    "Total")
datos <- subset(datos, datos$`Total Nacional y CCAA` != "Total")
```

```r
# Transformacion a factor de nombres de comunidades
# unique(datos$`Total Nacional y CCAA`)
datos$`Total Nacional y CCAA` <- as.factor(datos$`Total Nacional y CCAA`)
# Cambiamos de nombre los niveles del factor
levels(datos$`Total Nacional y CCAA`) <- c("Andalucía", "Illes Balears",
    "Canarias", "Cataluña", "Comunitat Valenciana", "Comunidad de Madrid",
    "Otras CCAA")
```

3.3. Cambios de nombres de las columnas

Vamos a renombrar las columnas de manera apropiada (sin espacios y con nombres representativos).

```r
# colnames(datos)
nombres_columnas <- c("Pais", "CCAA", "Periodo", colnames(datos)[4:7])
colnames(datos) <- nombres_columnas
colnames(datos)
```

```
## [1] "Pais"                      "CCAA"
## [3] "Periodo"                   "Gasto_total"
## [5] "Gasto_medio_persona"       "Gasto_medio_diario_persona"
## [7] "Duracion_media"
```

3.4. Instance engineering

Vamos a trabajar con valores de las filas para obtener un dataset más apropiado al estudio que deseamos realizar. En primer lugar, queremos deshacernos del nivel "Total" en la variable Pais y transformarla en otro llamado "Otros" en que en el resto de variables contenga la información referente al resto de paises distintos de los cuáles poseemos datos concretos.

```r
# Cambiar el nombre de las filas
paises <- levels(datos$Pais)
paises[paises == "Total"] <- "Otros"
levels(datos$Pais) <- paises
```

3.4.1. Instance engineering de Gasto_total

Para la variable Gasto_total, las filas correspondientes a "Otros" de la variable Pais y las filas correspondientes a "Total" se relacionan de la siguiente manera con $pais \in \{Alemania, Francia, Italia, Pases\ Nrdicos, Reino\ Unido, Otros\}$:

$$\text{Gasto total}_{\text{Pais = total}} = \sum_{pais} \text{Gasto total}_{pais}$$

Luego el gasto total de "Otros" es el gasto de "Total" menos el gasto de cada uno de los otros 5 paises disponibles por separado.

```
comunidades <- levels(datos$CCAA)
anos <- unique(datos$Periodo)
for (i in comunidades) {
    for (j in anos) {
        aux <- datos[datos$CCAA == i & datos$Periodo == j, "Gasto_total"]
        datos[datos$CCAA == i & datos$Periodo == j & datos$Pais ==
            "Otros", "Gasto_total"] <- aux[length(aux)] - sum(aux[1:length(aux) -
            1])
    }
}
```

3.4.2. Instance engineering de Gasto_medio_persona, Gasto_medio_diario_persona y Duracion_media

Como en estas variables se trata de una media, no podemos emplear el método usado para Gasto_total. En su lugar consideramos una media ponderada. Si consideramos que el número de paises totales es 194 tenemos que para $pais \in \{Alemania, Francia, Italia, Pases\ Nrdicos, R$

$$\overline{Media\ Total} = \frac{5}{194} * \sum_{pais} Valor\ medio_{pais} + \frac{194-5}{194} * Valor\ Medio_{otros}$$

De esta manera, concemos el valor de $\overline{Media\ Total}$ y de $Valor\ medio_{pais}$ y podemos calcular el valor de los valores medios para las filas "Otros" de la variable Pais, otorgandoles un mayor peso de manera proporcional al número de países considerados en esta categoría.

```
# Duracion media
for (i in comunidades) {
    for (j in anos) {
        aux <- datos[datos$CCAA == i & datos$Periodo == j, "Duracion_media"]
        datos[datos$CCAA == i & datos$Periodo == j & datos$Pais ==
            "Otros", "Duracion_media"] <- (aux[length(aux)] -
            (5/194) * sum(aux[1:length(aux) - 1])) * 194/(194 -
            5)
    }
}
```

```
# Gasto_medio_persona
for (i in comunidades) {
    for (j in anos) {
        aux <- datos[datos$CCAA == i & datos$Periodo == j, "Gasto_medio_persona"]
        datos[datos$CCAA == i & datos$Periodo == j & datos$Pais ==
            "Otros", "Gasto_medio_persona"] <- (aux[length(aux)] -
            (5/194) * sum(aux[1:length(aux) - 1])) * 194/(194 -
            5)
    }
}
```

```
# Gasto_medio_diario_persona
for (i in comunidades) {
    for (j in anos) {
        aux <- datos[datos$CCAA == i & datos$Periodo == j, "Gasto_medio_diario_per
        datos[datos$CCAA == i & datos$Periodo == j & datos$Pais ==
            "Otros", "Gasto_medio_diario_persona"] <- (aux[length(aux)] -
            (5/194) * sum(aux[1:length(aux) - 1])) * 194/(194 -
            5)
    }
}

rm(i, j, anos, comunidades, paises, aux)  # Borrar variables auxiliares
```

Una vez hemos concluido el pre-procesamiento de los datos, vamos a ver un resumen de las variables.

```
summary(datos)
```

```
##               Pais                        CCAA       Periodo
##  Alemania       :56   Andalucía            :48   Min.   :2016
##  Francia        :56   Illes Balears        :48   1st Qu.:2018
##  Italia         :56   Canarias             :48   Median :2020
##  Países Nórdicos:56   Cataluña             :48   Mean   :2020
##  Reino Unido    :56   Comunitat Valenciana:48   3rd Qu.:2021
##  Otros          :56   Comunidad de Madrid :48   Max.   :2023
##                       Otras CCAA           :48
##   Gasto_total       Gasto_medio_persona Gasto_medio_diario_persona
##  Min.   :   22.28   Min.   : 383.0      Min.   : 56.0
##  1st Qu.:  396.06   1st Qu.: 832.8      1st Qu.:105.0
##  Median :  849.69   Median : 980.8      Median :130.5
##  Mean   : 1775.89   Mean   : 983.7      Mean   :133.0
##  3rd Qu.: 2375.74   3rd Qu.:1138.0      3rd Qu.:154.0
##  Max.   :14281.08   Max.   :1612.5      Max.   :277.0
##
##  Duracion_media
##  Min.   : 3.610
##  1st Qu.: 5.793
##  Median : 7.465
##  Mean   : 7.586
##  3rd Qu.: 9.040
##  Max.   :14.270
##
```

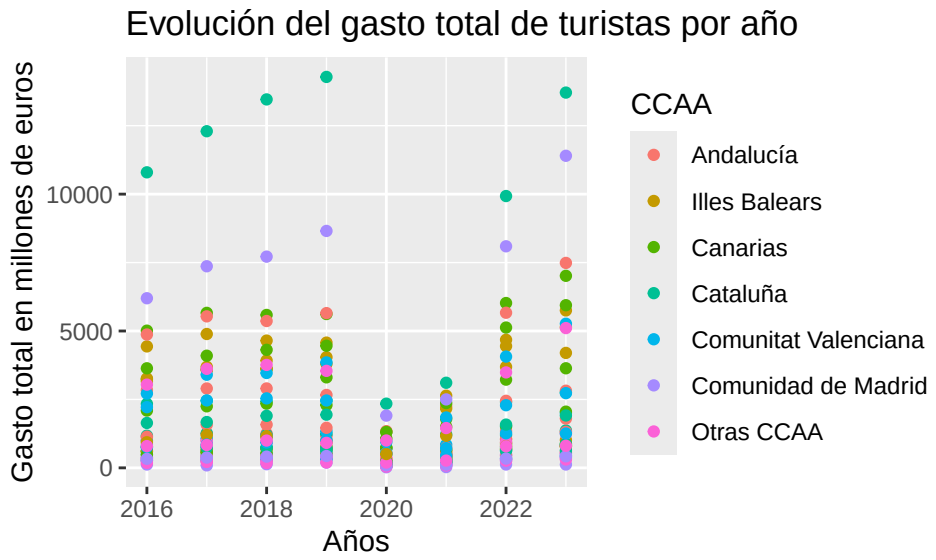
Table 1. 10 primeras filas del conjunto de datos procesados

	Pais	CCAA	Periodo	Gasto_total	Gasto_medio_persona	Gasto_medio_diario_persona	Duracion_media
1	Alemania	Andalucía	2016	1121.33	1132	102	11.14
2	Alemania	Andalucía	2017	1258.89	1123	105	10.70
3	Alemania	Andalucía	2018	1256.69	1165	114	10.18
4	Alemania	Andalucía	2019	1168.38	1048	117	8.96
5	Alemania	Andalucía	2020	255.96	1056	102	10.34
6	Alemania	Andalucía	2021	464.01	1040	100	10.39
7	Alemania	Andalucía	2022	1045.65	1211	113	10.69
8	Alemania	Andalucía	2023	1314.95	1267	130	9.71
17	Alemania	Illes Balears	2016	4436.99	961	129	7.44
18	Alemania	Illes Balears	2017	4890.66	1013	133	7.63

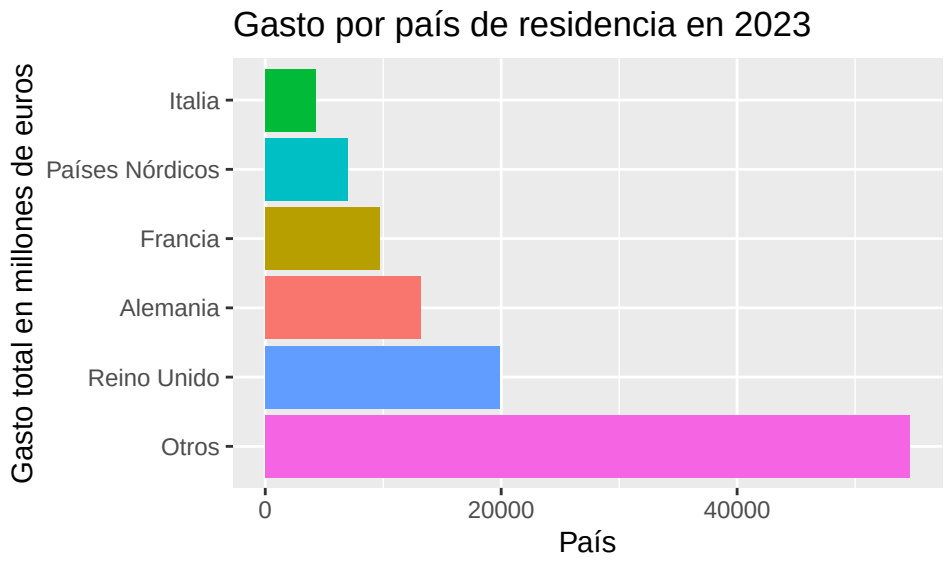
NOTA: Hay dos conjuntos de datos para representar: datos : quitando totales para poder ver relaciones por pais, comunidad autonoma etc datos_totales: Aqui estan guardados todos los datos que he modificado o borrado del dataset original referentes a valores totales por si se necesitan en algun momento

4. Prueba de ggplot

```
# Visualización de los datos e interpretación de los
# posibles patrones

# Visualización previa para tener una idea de ciertas
# variables de los datos
ggplot(datos, aes(x = Periodo, y = Gasto_total, color = CCAA,
    group = 1)) + geom_point() + labs(title = "Evolución del gasto total de turist
    x = "Años", y = "Gasto total en millones de euros")
```

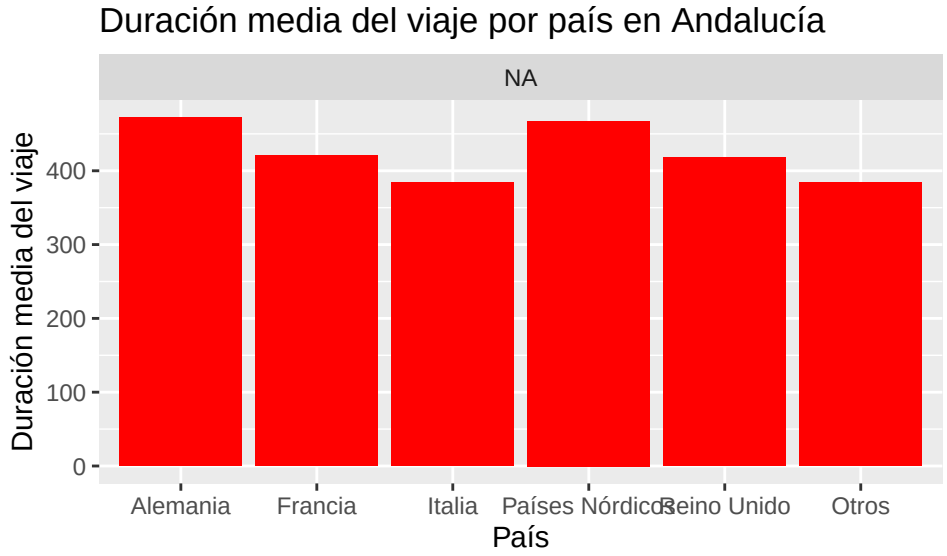



```
# Nos centramos en un año en concreto y visualizamos el
# gasto total por país en ese año
ggplot(datos[datos$Periodo == 2023, ], aes(x = reorder(Pais,
    -Gasto_total), y = Gasto_total)) + geom_bar(stat = "identity",
    aes(fill = Pais)) + coord_flip() + labs(title = "Gasto por país de residencia
    x = "Gasto total en millones de euros", y = "País") + theme(legend.position =
```



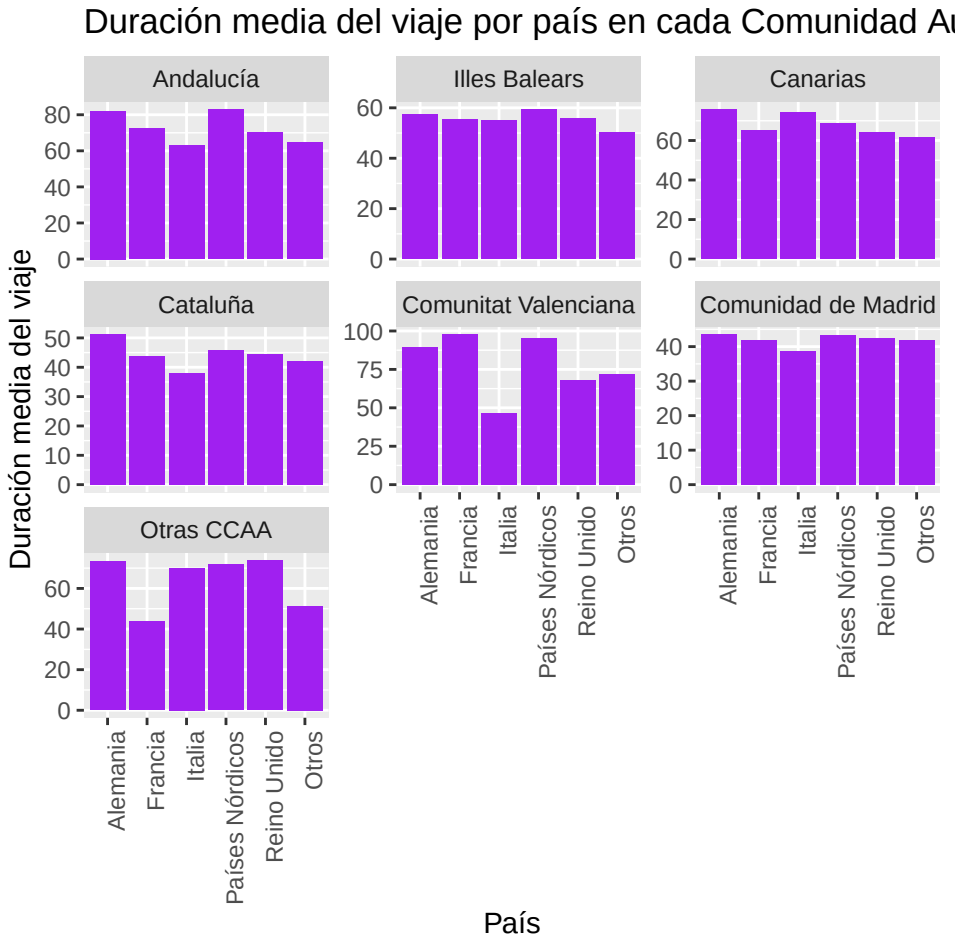
109

```
# Para analizar únicamente una variable por país en una
# sola Comunidad Autónoma.
ggplot(datos, aes(x = Pais, y = Duracion_media)) + geom_bar(stat = "identity",
    fill = "red") + facet_grid(. ~ CCAA["i"], scales = "free") +
    labs(title = "Duración media del viaje por país en Andalucía",
        x = "País", y = "Duración media del viaje")
```



110

```
# Para todas las Comunidades Autónomas por separado
ggplot(datos, aes(x = Pais, y = Duracion_media)) + geom_bar(stat = "identity",
    fill = "purple") + facet_wrap(~CCAA, scales = "free_y") +
    labs(title = "Duración media del viaje por país en cada Comunidad Autónoma",
        x = "País", y = "Duración media del viaje") + theme(axis.text.x = element_
    hjust = 1))
```

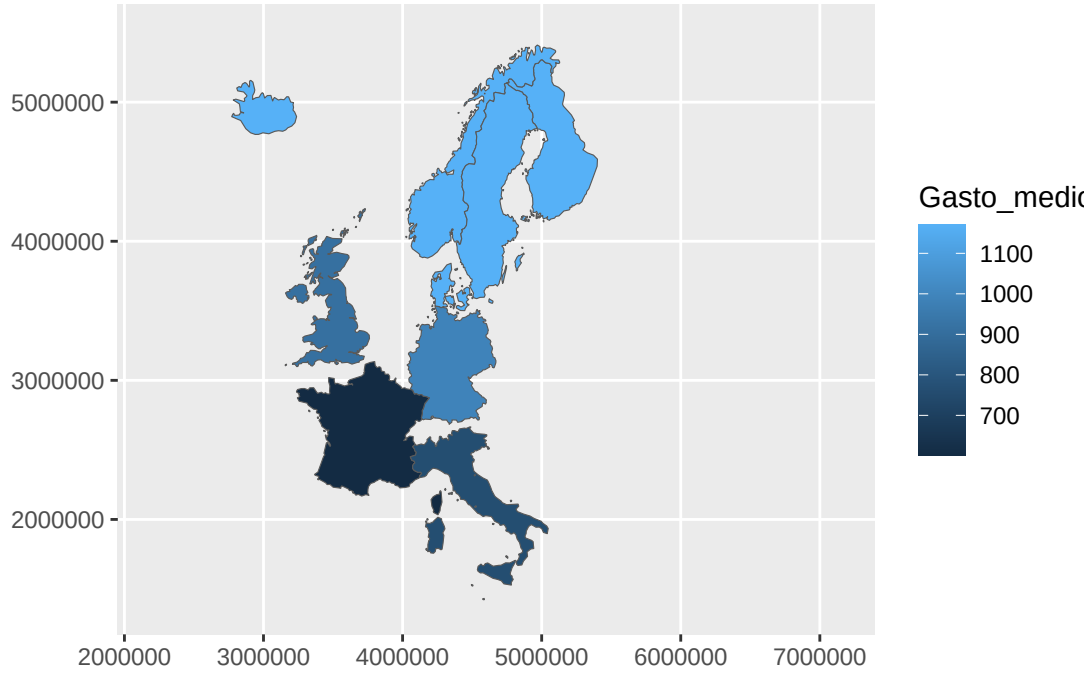


Filtraremos los datos totales por el Periodo y recogeremos solo la variable que nos sirve para representar la magnitud de valores en el mapa de visualización, en este caso el gasto medio por persona

```
gastos_medio_residencia <- datos_total_por_residencia %>%
    select(`País de residencia`, Gasto_medio_persona, Periodo) %>%
    filter(Periodo == "2016" & `País de residencia` != "Total")

gastos_medio_residencia_PN <- gastos_medio_residencia %>%
    select(`País de residencia`, Gasto_medio_persona, Periodo) %>%
    filter(`País de residencia` == "Países Nórdicos")

# Introducimos 4 filas iguales sobre los países nórdicos
# para separar con el join estos y poder representarlos en
# el mapa

gastos_medio_residencia <- rbind(gastos_medio_residencia, gastos_medio_residencia_
    gastos_medio_residencia_PN, gastos_medio_residencia_PN, gastos_medio_residenci

# install.packages('sf')
library(sf)
```

```
## Linking to GEOS 3.12.1, GDAL 3.8.4, PROJ 9.3.1; sf_use_s2() is TRUE
```

```r
# install.packages('dplyr')
library(dplyr)
# install.packages('ggplot2')
library(ggplot2)
# install.packages('giscoR')
library(giscoR)

año_ref <- 2016

# Datos
countries <- gisco_get_countries(year = año_ref, resolution = 20) %>%
    select(CNTR_ID, NAME_ENGL, geometry) %>%
    st_transform(3035)

paises_english <- c("Germany", "France", "Italy", "Norway", "Finland",
    "Sweden", "Denmark", "Iceland", "United Kingdom")
countries_filtered <- countries %>%
    filter(NAME_ENGL %in% paises_english)

# Incluir columna de identificación de los países para unir
# con las coordenadas de los mapas

CNTR_ID <- c("DE", "FR", "IT", "DK", "UK", "IS", "FI", "NO",
    "SE")
gastos_medio_residencia <- cbind(gastos_medio_residencia, CNTR_ID)


gastos_medio_viz <- gastos_medio_residencia %>%
    left_join(countries, by = join_by(CNTR_ID == CNTR_ID))



# Mapa base
ggplot(gastos_medio_viz) + geom_sf(aes(geometry = geometry, fill = Gasto_medio_per
    xlim(c(2200000, 7150000)) + ylim(c(1380000, 5500000))
```




```
# Marta: Te comenté esta parte porque me estaba dando
# problemas al hacer knit no se por qué

# Gráfico ggplot(gastos_medio_viz) + # Primera capa con
# todos los países geom_sf(aes(geometry= geometry), data =
# gastos_medio_viz, fill = 'grey80', color = NA) + #
# Establece límites xlim(c(2200000, 7150000)) +
# ylim(c(1380000, 5500000))
```