

finalproject3

```
knitr::opts_chunk$set(echo = TRUE)
library(magrittr)
library(dplyr)
```

Attaching package: 'dplyr'

The following objects are masked from 'package:stats':

filter, lag

The following objects are masked from 'package:base':

intersect, setdiff, setequal, union

```
library(ggplot2)
```

```
poblacion <- read.csv("59011.csv", header = TRUE, sep = "\t")
poblacion$Date <- poblacion$Periodo
head(poblacion)
```

	Nacionalidad	Periodo	Total	Date
1	Española	1 de enero de 2025	19.700	1 de enero de 2025
2	Española	1 de octubre de 2024	25.200	1 de octubre de 2024
3	Española	1 de julio de 2024	20.600	1 de julio de 2024
4	Española	1 de abril de 2024	18.500	1 de abril de 2024
5	Española	1 de enero de 2024	19.500	1 de enero de 2024
6	Española	1 de octubre de 2023	23.600	1 de octubre de 2023

```
#vector de reemplazo de meses
meses <- c("enero" = "01", "febrero" = "02", "marzo" = "03", "abril" = "04",
           "mayo" = "05", "junio" = "06", "julio" = "07", "agosto" = "08",
           "septiembre" = "09", "octubre" = "10", "noviembre" = "11", "diciembre" = "12")

# Reemplazar nombres de meses por su número
for (mes in names(meses)) {
  poblacion$Date <- gsub(mes, meses[mes], poblacion$Date)
}
head(poblacion)
```

	Nacionalidad	Periodo	Total	Date
1	Española	1 de enero de 2025	19.700	1 de 01 de 2025
2	Española	1 de octubre de 2024	25.200	1 de 10 de 2024
3	Española	1 de julio de 2024	20.600	1 de 07 de 2024
4	Española	1 de abril de 2024	18.500	1 de 04 de 2024
5	Española	1 de enero de 2024	19.500	1 de 01 de 2024
6	Española	1 de octubre de 2023	23.600	1 de 10 de 2023

```
# Quitar la palabra "de"
poblacion$Date<- gsub(" de ", "-", poblacion$Date)

# Convertir a clase Date
poblacion$Julian <- julian(as.Date(poblacion$Date, format = "%d-%m-%Y"))
#####
head(poblacion)
```

	Nacionalidad	Periodo	Total	Date	Julian
1	Española	1 de enero de 2025	19.700	1-01-2025	20089
2	Española	1 de octubre de 2024	25.200	1-10-2024	19997
3	Española	1 de julio de 2024	20.600	1-07-2024	19905
4	Española	1 de abril de 2024	18.500	1-04-2024	19814
5	Española	1 de enero de 2024	19.500	1-01-2024	19723
6	Española	1 de octubre de 2023	23.600	1-10-2023	19631

```
summary(poblacion)
```

Nacionalidad	Periodo	Total	Date
Length:488	Length:488	Length:488	Length:488
Class :character	Class :character	Class :character	Class :character

Mode :character Mode :character Mode :character Mode :character

Julian

Min. :19448

1st Qu.:19608

Median :19768

Mean :19768

3rd Qu.:19928

Max. :20089

```
(poblacion$Total <- as.numeric(poblacion$Total))
```

Warning: NAs introduced by coercion

[1]	19.7	25.2	20.6	18.5	19.5	23.6	19.9	18.4	NA	NA	NA	NA	NA	NA	NA
[16]	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
[31]	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
[46]	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	8.9	9.0	10.3	9.7
[61]	9.6	8.8	10.2	10.2	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
[76]	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
[91]	NA	NA	NA	NA	NA	NA	NA	5.7	NA	6.0	5.7	6.2	6.3	7.1	NA
[106]	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
[121]	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
[136]	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
[151]	NA	NA	6.4	7.0	7.3	6.1	NA	5.2	5.9	11.5	NA	NA	NA	NA	NA
[166]	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
[181]	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
[196]	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
[211]	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
[226]	NA	NA	NA	NA	NA	NA	NA	27.7	25.5	25.1	26.0	25.8	24.4	21.5	23.2
[241]	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
[256]	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
[271]	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
[286]	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
[301]	NA	NA	NA	NA	7.3	5.6	6.4	7.3	7.0	5.5	7.6	10.1	NA	NA	NA
[316]	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
[331]	NA	NA	NA	NA	NA	NA	7.3	7.3	8.3	7.4	7.9	7.3	8.8	8.6	NA
[346]	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA	NA
[361]	43.4	34.6	36.9	39.2	42.6	34.0	37.7	44.3	NA	NA	NA	NA	NA	NA	NA
[376]	NA	6.8	NA	NA	NA	6.2	NA	NA	NA	NA	NA	5.5	NA	NA	NA

```
[391]  NA   NA 18.8 15.8 15.2 15.6 17.2 13.0 14.6 18.8   NA   NA   NA   NA   NA
[406]  NA   NA   NA 30.5 21.2 21.4 22.6 27.3 15.3 18.1 21.5   NA   NA   NA   NA
[421]  NA   NA   NA   NA   NA   NA   NA   NA   NA   NA   NA   NA   NA   NA   NA
[436]  NA   NA   NA   NA   NA   NA   NA   NA   NA   NA   NA   NA   NA   NA   NA
[451]  NA   NA   NA   NA   NA   NA   NA   NA   NA   NA   NA   NA   NA   NA   NA
[466]  NA   NA   NA   NA   NA   NA   NA   NA   NA   NA   NA   NA   NA   NA   NA
[481]  NA   NA   NA   NA   NA   NA   NA   NA   NA
```

```
poblacion <- poblacion[!is.na(poblacion$Total), ]
```

```
# grafico
(x_axis <- max(poblacion$Total, na.rm = T))
```

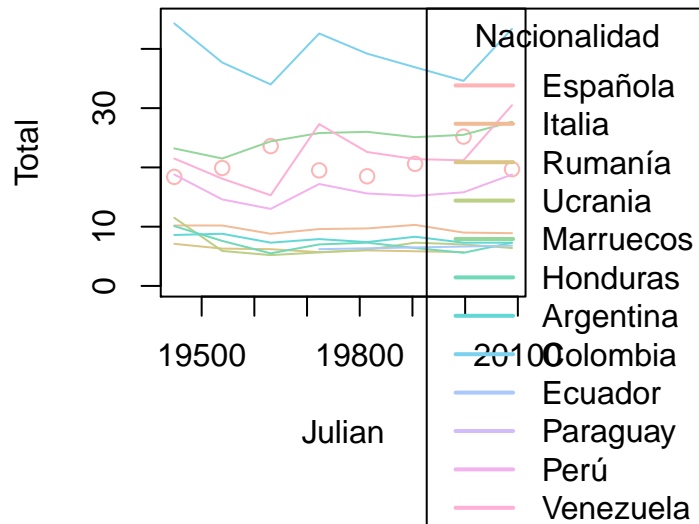
```
[1] 44.3
```

```
(ylim_range <- c(0, ceiling(x_axis/5)*5))
```

```
[1] 0 45
```

```
total_color <- hcl.colors(length(unique(poblacion$Nacionalidad)), palette = "Set3")
par(mar = c(6, 8, 4, 10))
#plot_total <- barplot(poblacion$Total, names.arg = poblacion$Nacionalidad,
#                      las = 2, col = total_color, horiz = TRUE, xaxt = "n",
#                      main = "inmigracion extranjero")
plot(Total ~ Julian, data = poblacion, subset = Nacionalidad == unique(poblacion$Nacionalidad),
for(i in 2:length(unique(poblacion$Nacionalidad))){
  lines(Total ~ Julian, data = poblacion, subset = Nacionalidad == unique(poblacion$Nacionalidad))
}
axis(side = 1, at = seq(0, max(ylim_range), by = 5))

legend("topright",
      legend = unique(poblacion$Nacionalidad),
      col = total_color,
      lwd = 2,
      title = "Nacionalidad",
      xpd = TRUE,
      inset = c(-0.5, 0))
```



```
# col = total_color[1] la l especifica que quiero las lineas de colores y en el for loop la
#xpd = TRUE: permite que la leyenda se dibuje fuera del área del gráfico
```

```
#| lablabel: create_new_columns
#| fig.width: 10
#| fig.height: 5
#pais <- split(poblacion, f = poblacion$Nacionalidad)
#print(pais)
#america <- c(pais$Argentina, pais$Colombia, pais$Ecuador, pais$Honduras, pais$Paraguay, pais$Peru)
#print(america)

poblacion$Zona <- dplyr::case_when(
  poblacion$Nacionalidad %in% c("Española", "Italia", "Rumanía", "Ucrania") ~ "Europa",
  poblacion$Nacionalidad %in% c("Colombia", "Argentina", "Venezuela", "Ecuador", "Perú", "Honduras") ~ "America",
  poblacion$Nacionalidad %in% c("Marruecos") ~ "Africa",
  TRUE ~ "Otras")

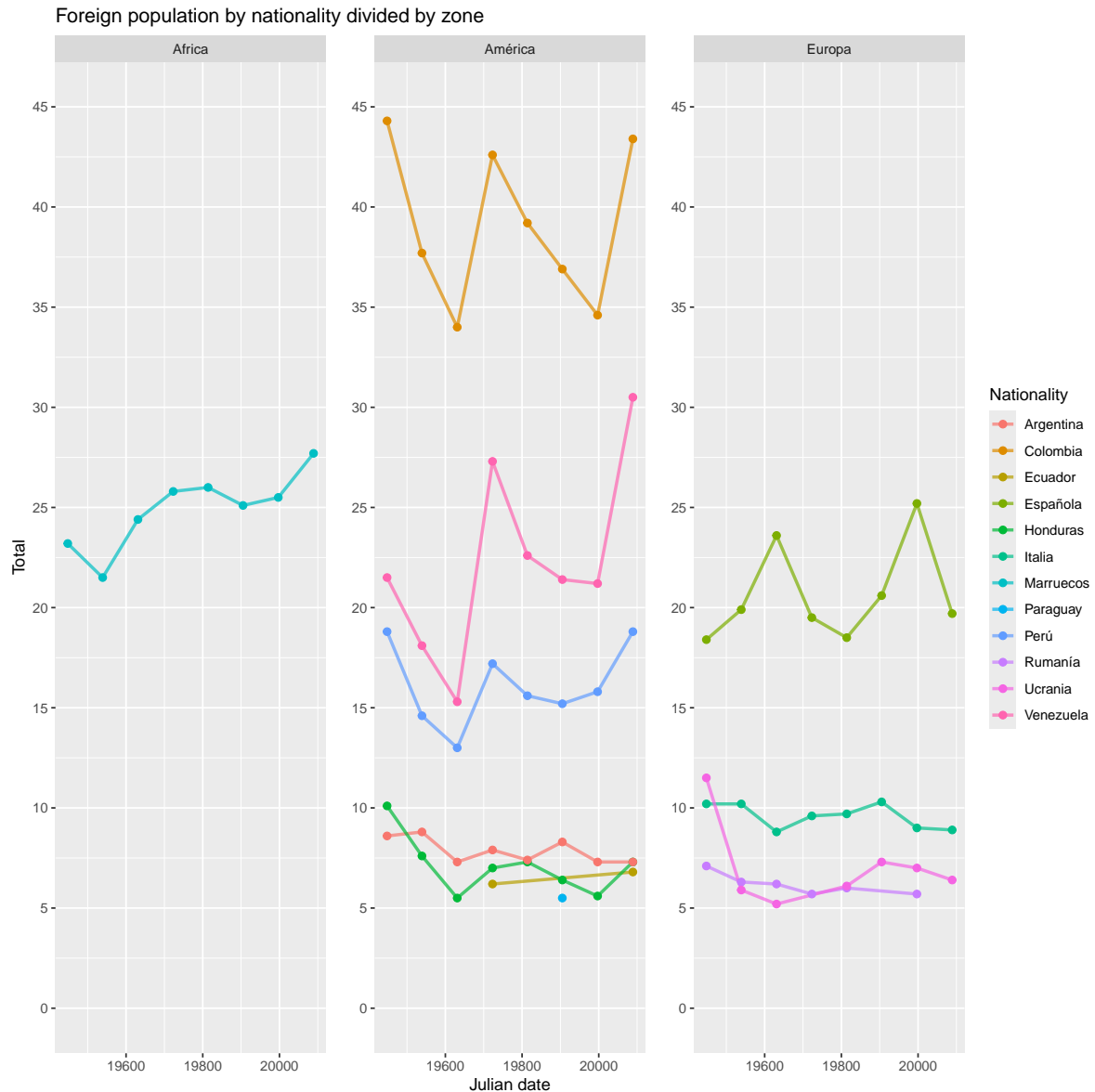
#countries <- unique(poblacion$Nacionalidad)
#length(countries)
#df1 <- poblacion[poblacion$Nacionalidad %in% countries[1:length(countries) / 3], ]
```

```

y_lim <- max(poblacion$Total, na.rm = T)
ylim_range <- c(0, ceiling(y_lim/5)*5)
point_plot <- ggplot(poblacion, aes(x = Julian, y = Total, color = Nacionalidad)) +
  geom_line(alpha = 0.7, linewidth= 1) +
  geom_point(alpha = 1, size = 2) +
  facet_wrap(~ Zona, scales = "free_y") +
  scale_y_continuous(
    limits = ylim_range,
    breaks = seq(0, ylim_range[2], by = 5)
  ) +
  labs(
    title = "Foreign population by nationality divided by zone",
    x = "Julian date",
    y = "Total",
    color = "Nationality"
  ) +
  theme(
    panel.spacing = unit(0.6, "cm")
  )

print(point_plot)

```



```
#y_lim para crear yaxis con el total de la poblacion, ceiling para redondear al siguiente mu.
#aes = los datos que quiero que use en el ggplot
#color= nacionalidades para que cada pais tenga un color diferente
#facet_wrap = ~Zona, quiero tres graficos dependiendo de la zona.
# free_y cada zona tenga propio eje y
# scale_continuous, hacer el set up de yaxis, limits = de donde a donde quiero que vaya, bre
#panel.spacing = dejar espacio entre los graficos.
```

```
poblacion2 <- read.csv("export.csv", header = TRUE, sep = ";")
head(poblacion2)
```

	Año	Fecha	Población.residente		
1	2021	1 de enero	47.400.798		
2	NA	1 de abril	47.356.065		
3	NA	1 de julio	47.346.836		
4	NA	1 de octubre	47.424.595		
5	2022	1 de enero	47.486.727		
6	NA	1 de abril	47.609.145		
		Variación.respecto.al.trimestre.anterior	Variación.anual....		
1			NA		
2			-44.733		
3			-9.229		
4			77.759		
5			62.132		0,18
6			122.418		0,53

```
poblacion2$Año <- c("2021", "2021", "2021", "2021", "2022", "2022", "2022", "2022", "2023", "
```

```
#convertir en number
```

```
poblacion2$Date <- poblacion2$Fecha
```

```
meses <- c("enero" = "01", "febrero" = "02", "marzo" = "03", "abril" = "04",
           "mayo" = "05", "junio" = "06", "julio" = "07", "agosto" = "08",
           "septiembre" = "09", "octubre" = "10", "noviembre" = "11", "diciembre" = "12")
```

```
# Reemplazar nombres de meses por su número
```

```
for (mes in names(meses)) {
  poblacion2$Date <- gsub(mes, meses[mes], poblacion2$Date)
}
head(poblacion2)
```

	Año	Fecha	Población.residente
1	2021	1 de enero	47.400.798
2	2021	1 de abril	47.356.065
3	2021	1 de julio	47.346.836
4	2021	1 de octubre	47.424.595
5	2022	1 de enero	47.486.727
6	2022	1 de abril	47.609.145

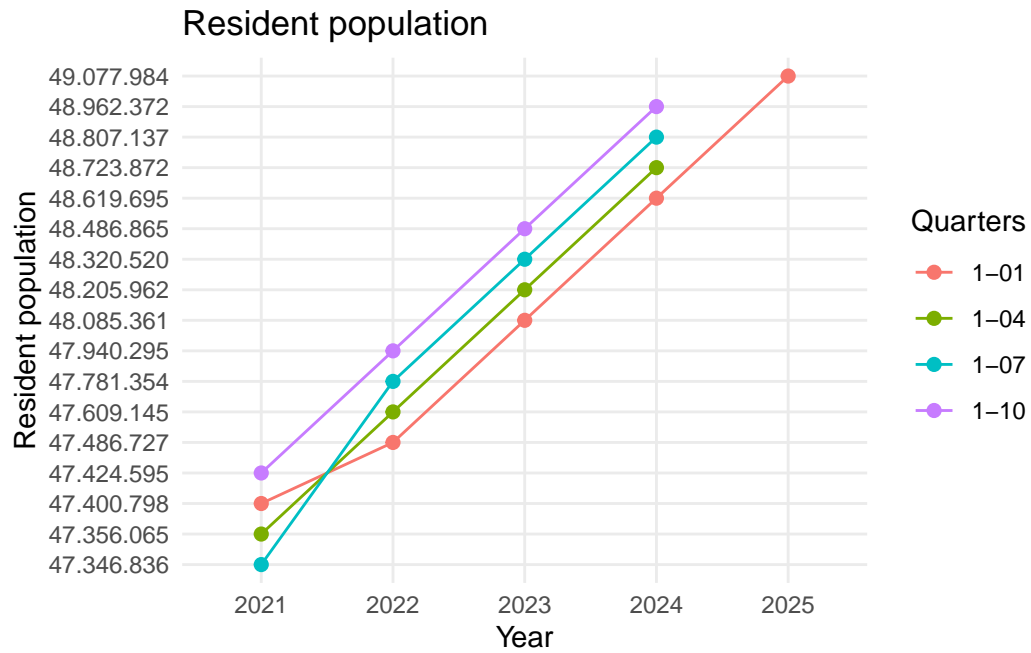
	Variación.respecto.al.trimestre.anterior	Variación.anual....	Date
1	NA		1 de 01
2	-44.733		1 de 04
3	-9.229		1 de 07
4	77.759		1 de 10
5	62.132	0,18	1 de 01
6	122.418	0,53	1 de 04

```
# Quitar la palabra "de"
poblacion2$Date<- gsub(" de ", "-", poblacion2$Date)

# Convertir a clase Date
poblacion2$Julian2 <- julian(as.Date(poblacion2$Date, format = "%d-%m"))
```

```
plot3 <- ggplot(poblacion2, aes(x = Año, y = Población.residente, group = Date, color = Date))
  geom_point(size = 2) +
  geom_line() +
  # scale_color_gradient(low = "blue", high = "red") +
  labs(title = "Resident population",
        x = "Year",
        y = "Resident population",
        color = "Quarters") +
  theme_minimal()

print(plot3)
```



	Category	Subcategory	Population	Variation	Growth (%)
0	TOTAL		49077984	115612	0.24
1	Nationality	Spanish	42225636	14819	0.04
2	Nationality	Foreign	6852348	100793	1.49
3	Country of Birth	Spain	39698012	-20505	-0.05
4	Country of Birth	Foreign	9379972	136117	1.47