

NOMBRE COMPLETO: Ing. Hilaría Adima Vásquez Durán

NAO ID: 3033

FECHA: 16 de agosto de 2024

NOMBRE DE LA TRAYECTORIA

EN LA QUE ESTÁS ENROLADO: DATA ANALYST CORE

Título del Reto:

Programación R al elaborar reportes estadísticos

SPRINT 2 DESARROLLO R

Crea un conjunto de visualizaciones de los datos de la farmacéutica Rossman Pharm, a partir de programas en R, para complementar el análisis y explorar la crisis en los países donde dicha empresa tiene presencia, que cumplan las siguientes especificaciones:

EJERCICIO 1

Crea un programa ("2_a_evolution_sales.R") que genere visualizaciones que permitan apreciar la evolución de las ventas (en millones de dólares) considerando todos los datos históricos desde su existencia. El programa deberá crear una visualización general y otra que permita ver la apertura de dicha evolución por país; las imágenes resultantes deberán denominarse "2_a_evolution_sales_general.png" y "2_a_evolution_sales_by_country.png",

DESARROLLO DEL CÓDIGO

Cargar las bibliotecas `tidyvers` , `readxl` y `lubridate`

```
library(tidyverse)
library(readxl)
library(lubridate)
```

```
> library(tidyverse)
— Attaching core tidyverse packages — tidyverse 2.0.0 —
✓ dplyr      1.1.4    ✓ readr      2.1.5
✓ forcats   1.0.0    ✓ stringr   1.5.1
✓ ggplot2    3.5.1    ✓ tibble    3.2.1
✓ lubridate  1.9.3    ✓ tidyr     1.3.1
✓ purrr      1.0.2
— Conflicts — tidyverse_conflicts() —
✖ dplyr::filter() masks stats::filter()
✖ dplyr::lag()     masks stats::lag()
i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflict
s to become errors
> library(readxl)
> library(lubridate)
>
```

Se define el directorio de trabajo donde se encuentran los archivos de datos.

```
data path <- "/Users/Usuario/Desktop/Help ME"
setwd(data path)
```

Se leen los datos de ventas desde un archivo CSV y los datos de tiendas desde un archivo Excel.

```
# Leer los datos de ventas
sales data <- 'sales.csv'
sales <- read.csv2(sales data, sep = ";", header = TRUE)

# Leer los datos de tiendas
stores info data <- 'stores info.xlsx'
stores <- read_excel(stores info data, sheet = "data")
```

```
>
> data_path <- "/Users/Usuario/Desktop/Help ME"
> setwd(data_path)
> # Leer los datos de ventas
> sales_data <- 'sales.csv'
> sales <- read.csv2(sales_data, sep = ";", header = TRUE)
>
> # Leer los datos de tiendas
> stores_info_data <- 'stores_info.xlsx'
> stores <- read_excel(stores_info_data, sheet = "data")
> □
```

#Se convierte la `date` al tipo de dato `Date` usando la función `lubridate` de `lubridate`

```
sales_all$date <- ymd(sales_all$date)
```

Se realiza una unión (join) de las tablas de ventas y tiendas basándose en la columna `store`.

```
sales_all <- sales %>% left_join(stores, by = c("store"))
```

```
>
> sales_all <- sales %>% left_join(stores, by = c("store"))
> 
```

Se convertirá las Ventas a Millones de Dólares

```
sales_all$sales_million <- sales_all$sales / 1e6
```

Agrupar las ventas por año y mes, y se calcula la suma total de ventas en millones de dólares para cada mes.

```
sales_by_month <- sales_all %>%
  group_by(year = year(date), month = month(date)) %>%
  summarise(total_sales_million = sum(sales_million), .groups = 'drop')
```

```
> sales_all$date <- ymd(sales_all$date)
> sales_all$sales_million <- sales_all$sales / 1e6
> sales_by_month <- sales_all %>%
+   group_by(year = year(date), month = month(date)) %>%
+   summarise(total_sales_million = sum(sales_million), .groups = 'drop')
> 
```

Crear una visualización (gráfico de líneas) que muestra la evolución de las ventas en millones de dólares a lo largo del tiempo.

```

evolution_general_plot <- ggplot(sales_by_month, aes(x = as.Date(paste
  geom_line(color = "blue") +
  labs(title = "Evolución de las Ventas en Millones de Dólares",
    x = "Fecha",
    y = "Ventas (Millones de USD)") +
  theme_minimal()

```

```

> evolution_general_plot <- ggplot(sales_by_month, aes(x = as.Date(paste(year,$
+   geom_line(color = "blue") +
+   labs(title = "Evolución de las Ventas en Millones de Dólares",
+     x = "Fecha",
+     y = "Ventas (Millones de USD)") +
+   theme_minimal()

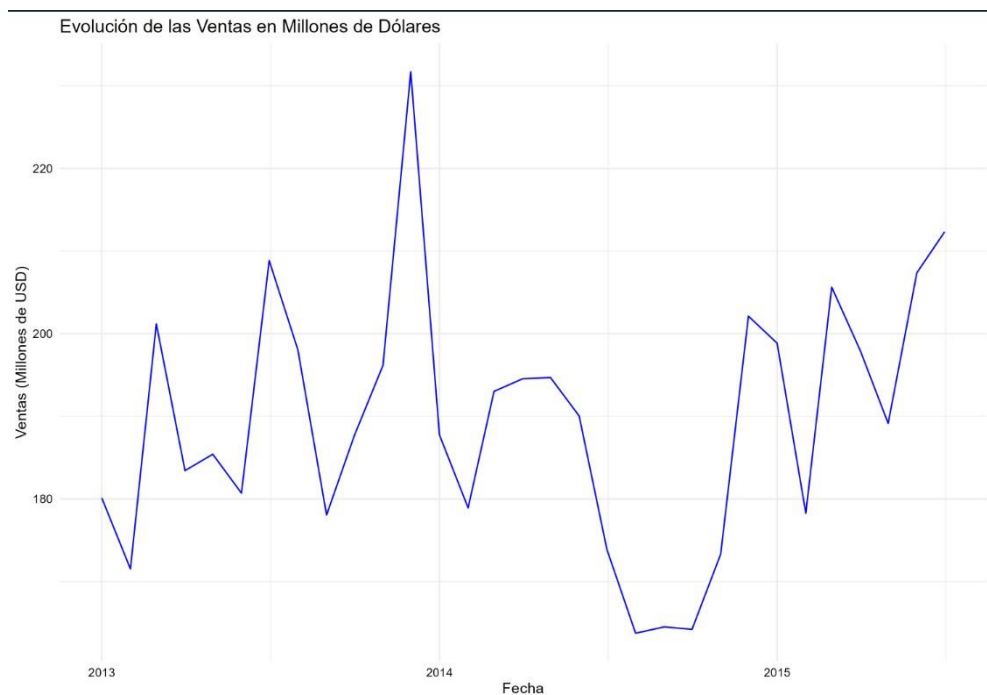
```

```

ggsave("2_a_evolution_sales_general.png", plot = evolution_general_plot,

```

Guardar la Visualización General



AGRUPAR VENTAS POR MES Y PAÍS

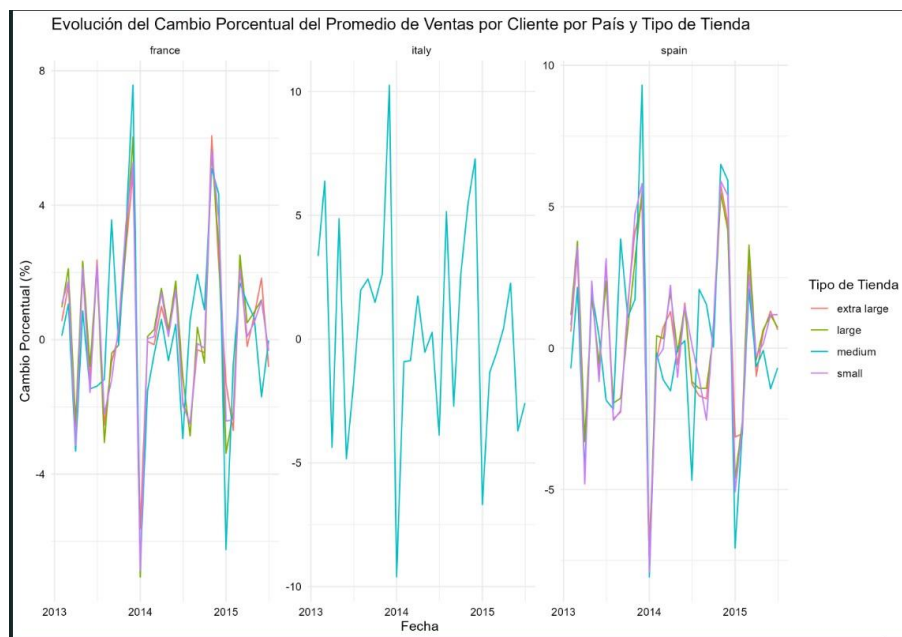
```
sales_by_month_country <- sales_all %>%
  group_by(country, year = year(date), month =
month(date)) %>% summarise(total_sales_million =
sum(sales_million), .groups = 'drop')
```

Crear Visualización de la Evolución de las Ventas por País

```
evolution_by_country_plot <-
ggplot(sales_by_month_country, aes(x = as.Date(
+ labs(title = "Evolución de las Ventas en Millones de
Dólares por País" x = "Fecha", y = "Ventas
(Millones de USD)", color = "País") +
theme_minimal() + theme(plot.background =
element_rect(fill = "white", color = NA),
panel.background = element_rect(fill = "white", color =
NA), plot.title = element_text(color = "black"),
axis.title.x = element_text(color = "black"),
axis.title.y = element_text(color = "black"),
axis.text = element_text(color = "black"),
plot.caption = element_text(color = "black"),
strip.text = element_text(color = "black") # Asegura que
los títulos)
```

```
ggsave("2_a_evolution_sales_by_country.png", plot =
evolution_by_count
```

```
> ggsave("2_a_evolution_sales_general.png", plot = evolution_general_plot, wid$
> sales_by_month_country <- sales_all %>%
+ group_by(country, year = year(date), month = month(date)) %>%
+ summarise(total_sales_million = sum(sales_million), .groups = 'drop')
> evolution_by_country_plot <- ggplot(sales_by_month_country, aes(x = as.Date($
+ geom_line() +
+ labs(title = "Evolución de las Ventas en Millones de Dólares por País",
+ x = "Fecha",
+ y = "Ventas (Millones de USD)",
+ color = "País") +
+ theme_minimal() +
+ facet_wrap(~ country, scales = "free_y")
>
```



Guardar la Visualización por País

```
cat("Las figuras '2_a_evolution_sales_general.png' y '2_a_evolution_sales_by_country.png' han sido creadas exitosamente.")
```

```
> ggsave("2_a_evolution_sales_by_country.png", plot = evolution_by_country_plot)
> cat("Las figuras '2_a_evolution_sales_general.png' y '2_a_evolution_sales_by_country.png' han sido creadas exitosamente.")
```