# PYTHON, CARTO, POWER BI Y GOOGLE DATA STUDIO

ENEB | Escuela de Negocios Europea de Barcelona

*BIG DATA & ANALYTICS*

# ÍNDICE


## TEMA 1. Interpretación de datos

1.1. INTRODUCCIÓN A LA VISUALIZACIÓN DE DATOS ..... 9

1.1.1. ¿Qué se entiende por visualización de datos? ..... 9

1.1.2. ¿Por qué se crean visualizaciones? ..... 11

1.1.3. Cuarteto de Anscombe ..... 11

1.2. PROCESO DE VISUALIZACIÓN ..... 13

1.2.1. Estructura ..... 13

1.2.2. Combinación e interacción ..... 15

1.2.3. Recomendaciones ..... 15

1.3. TIPOS DE GRÁFICOS Y CÓMO ELEGIRLOS ..... 16

1.3.1. Tipos de gráficos ..... 16

1.4. EL ERROR EN LA REPRESENTACIÓN DE LA INFORMACIÓN ..... 27

1.4.1. Errores comunes en la representación de un gráfico ..... 27

IDEAS CLAVE ..... 29

GLOSARIO ..... 31

BIBLIOGRAFÍA ..... 33

## TEMA 2. Visualización interactiva con Python

2.1. MATPLOTLIB ..... 37

2.1.1. Introducción a Matplotlib ..... 37

2.1.2. Instalación de librerías ..... 37

2.1.3. Gráfico de líneas (*sparkline*) ..... 38

2.1.4. Gráficos de barras ..... 44

2.1.5. Gráficos de tarta ..... 46
2.1.6. Histograma ..... 47
2.1.7. Gráfico de caja (*boxplots*) ..... 48
2.1.8. Gráfico de dispersión (*scatter plot*) ..... 49
2.1.9. Subplots ..... 50
2.2. SEABORN ..... 52
2.2.1. Introducción a Seaborn ..... 52
2.2.2. Instalación de Seaborn ..... 52
2.2.3. Gráfico de barras ..... 53
2.2.4. Gráfico de barras agrupadas ..... 54
2.2.5. Gráfico histograma ..... 56
2.2.6. Gráfico de caja (*boxplots*) ..... 56
2.2.7. Gráfico de dispersión ..... 57
2.2.8. Gráfico de tarta (*pie chart*) ..... 58
2.2.9. *Subplots* ..... 59
2.2.10. Gráfico de líneas ..... 60
2.3. PLOTLY ..... 61
2.3.1. Introducción a Plotly ..... 61
2.3.2. Instalación de librerías ..... 61
2.3.3. Gráfico de líneas (*line chart*) ..... 63
2.3.4. Gráfico de barras (*bar chart*) ..... 64
2.3.5. Gráfico de barras agrupadas (*bar chart*) ..... 65
2.3.6. Gráfico de caja (*boxplots*) ..... 66
2.3.7. Histograma ..... 67
2.3.8. Gráfico de tarta (*pie chart*) ..... 68
2.3.9. Gráfico de anillos ..... 69
2.3.10. Gráfico de dispersión ..... 70

IDEAS CLAVE ..... 71

GLOSARIO ..... 73

BIBLIOGRAFÍA ..... 75

## TEMA 3. Introducción a CARTO

3.1. INTRODUCCIÓN A CARTO ..... 79

3.1.1. ¿Qué es CARTO? ..... 79

3.1.2. Primeros pasos ..... 79

3.1.3. Crea tu primer mapa ..... 80

3.1.4. Explora tu *dataset* ..... 81

3.2. TIPOS DE FUENTES DE INFORMACIÓN Y TIPOS DE GRÁFICOS ..... 83

3.2.1. Tipos de fuentes de información ..... 83

3.3. CASO PRÁCTICO ..... 85

3.3.1. Selección de bases de datos ..... 85

3.3.2. Creación de nuestro mapa ..... 86

3.3.3. Editado de nuestro mapa ..... 89

3.3.4. Insertar *Widgets* ..... 90

IDEAS CLAVE ..... 95

GLOSARIO ..... 97

BIBLIOGRAFÍA ..... 99

## TEMA 4. Microsoft Power BI

4.1. INTRODUCCIÓN A POWER BI ..... 103

4.1.1. ¿Qué es Power BI? ..... 103

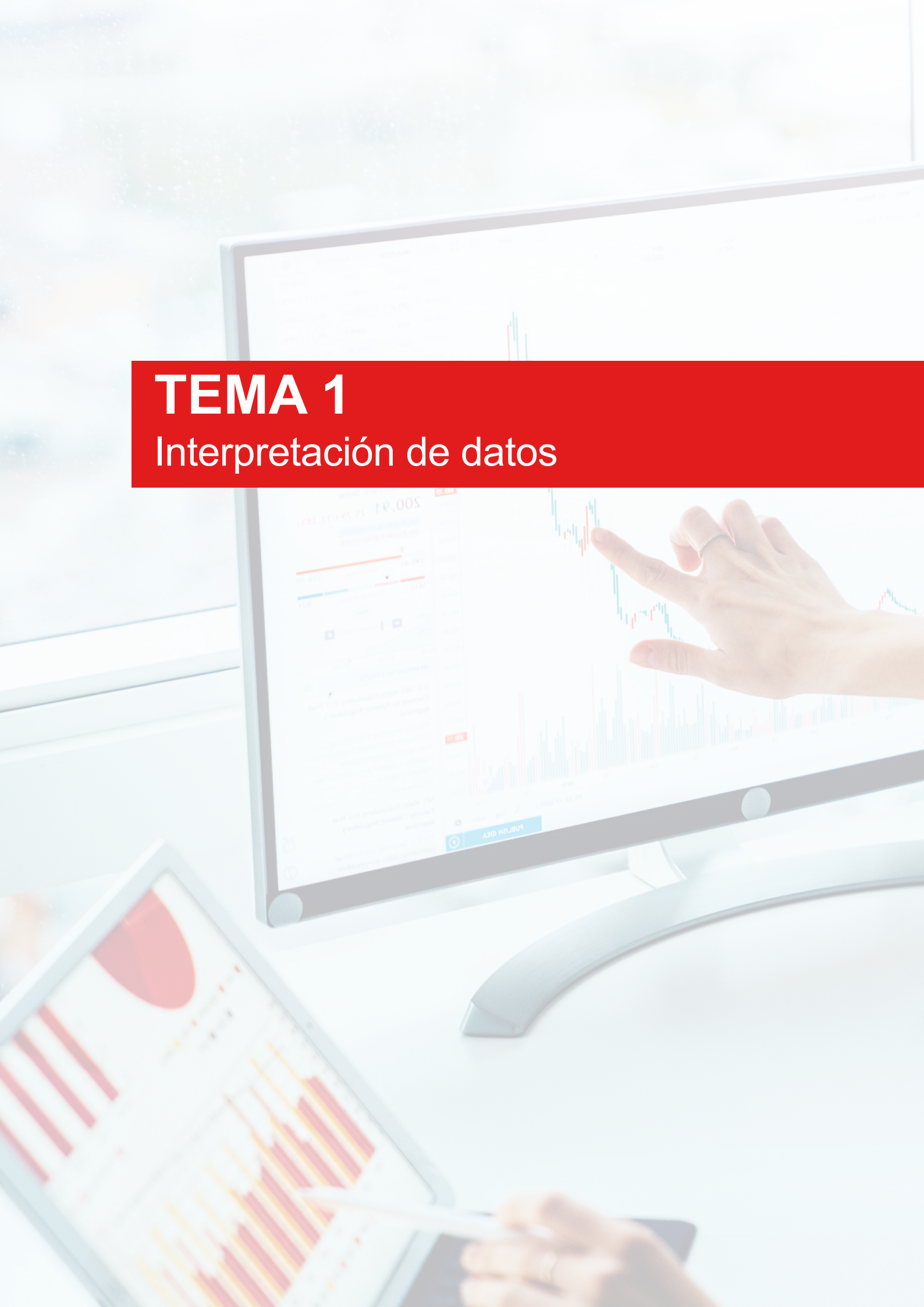
4.1.2. Estructura de Power BI ..... 103

4.1.3. Instalación de Power BI Desktop ..... 103

4.1.4. Power BI Desktop ..... 105
4.1.5. Conectar a datos ..... 106
4.1.6. Modelado ..... 108
4.1.7. Relación de tablas ..... 109
4.1.8. Creación de visuales ..... 110
4.1.9. Compartir informes ..... 111
4.1.10. Power BI Service ..... 112
4.2. TIPOS DE FUENTES DE DATOS Y TIPOS DE GRÁFICOS ..... 114
4.2.1. Tipos de fuentes de datos ..... 114
4.2.2. Tipos de visualizaciones ..... 118
4.3. CASO PRÁCTICO DE ETL ..... 126
4.4. CASO PRÁCTICO DE CONSTRUCCIÓN DE *DASHBOARD* ..... 131
IDEAS CLAVE ..... 137
GLOSARIO ..... 139
BIBLIOGRAFÍA ..... 141

## TEMA 5. Google Data Studio

5.1. ¿QUÉ ES GOOGLE DATA STUDIO? ..... 145
5.2. CARACTERÍSTICAS DE LA HERRAMIENTA E INTERFAZ ..... 146
5.3. FUENTES DE DATOS EN DATA STUDIO ..... 152
5.4. INFORMES EN DATA STUDIO ..... 153
IDEAS CLAVE ..... 161
GLOSARIO ..... 163
BIBLIOGRAFÍA ..... 165



# TEMA 1
## Interpretación de datos

# OBJETIVOS

- Estudiar la evolución de la teoría de la visualización a través de la evolución de las formas de representación.
- Conocer los fundamentos en el desarrollo de una visualización.
- Analizar los distintos tipos de visualización de datos.
- Conocer los riesgos y las mejores prácticas a la hora de llevar a cabo la visualización.

# 1.1. INTRODUCCIÓN A LA VISUALIZACIÓN DE DATOS

## 1.1.1. ¿QUÉ SE ENTIENDE POR VISUALIZACIÓN DE DATOS?

La recopilación de datos desde siempre ha sido de gran valor, ya que nos permite analizar y entender que ha ocurrido en el pasado.

Sin embargo, únicamente estos datos no nos permiten tener una idea clara de lo que ha ocurrido. El diseño nos ayuda a conseguir este objetivo, ya que el **diseño** es "la actividad creativa que tiene por fin proyectar objetos que sean útiles y estéticos" (Antonacci, 2022).

Por lo tanto, entendemos como visualización de datos a la presentación visual de estos.

La visualización de datos tiene principalmente tres aplicaciones:

1. **Ahorrar tiempo.** Vivimos en un mundo en el que el tiempo es un factor determinante, por lo que nos convendrá una visualización que sea capaz de transmitirnos su mensaje de un simple vistazo nos hará ahorrar tiempo.

2. **Personalizar los procesos.** Las organizaciones buscan la personalización de sus procesos y, para ello, además de datos, utilizan diseño.

3. **Asimilar el cambio tecnológico.** La tecnología evoluciona muy rápido, de tal forma que no siempre es fácil asimilar este cambio. El diseño acerca la tecnología a los usuarios permitiendo que este avance tecnológico sea útil, accesible y valioso.

**IMPORTANTE**

El objetivo principal de la **visualización de datos** es comunicar un mensaje de manera sencilla para el público objetivo. Este mensaje tiene que llamar a la acción: si el mensaje no llega a la acción, el valor del análisis es relativo. Lo que realmente aporta valor es la acción no el análisis.

Florence Nightingale fue una extraordinaria enfermera y es conocida como la creadora de la enfermería moderna. Además de esto, fue una gran analista de datos.

Durante la guerra de Crimea, que tuvo lugar en la década de 1850, Nightingale sirvió al ejército inglés como supervisora del hospital militar en Turquía. Comenzó a recopilar datos para su posterior visualización en tablas y gráficos, y fue la impulsora del gráfico *cowcomb* (similar a las áreas de los polos de la Tierra).

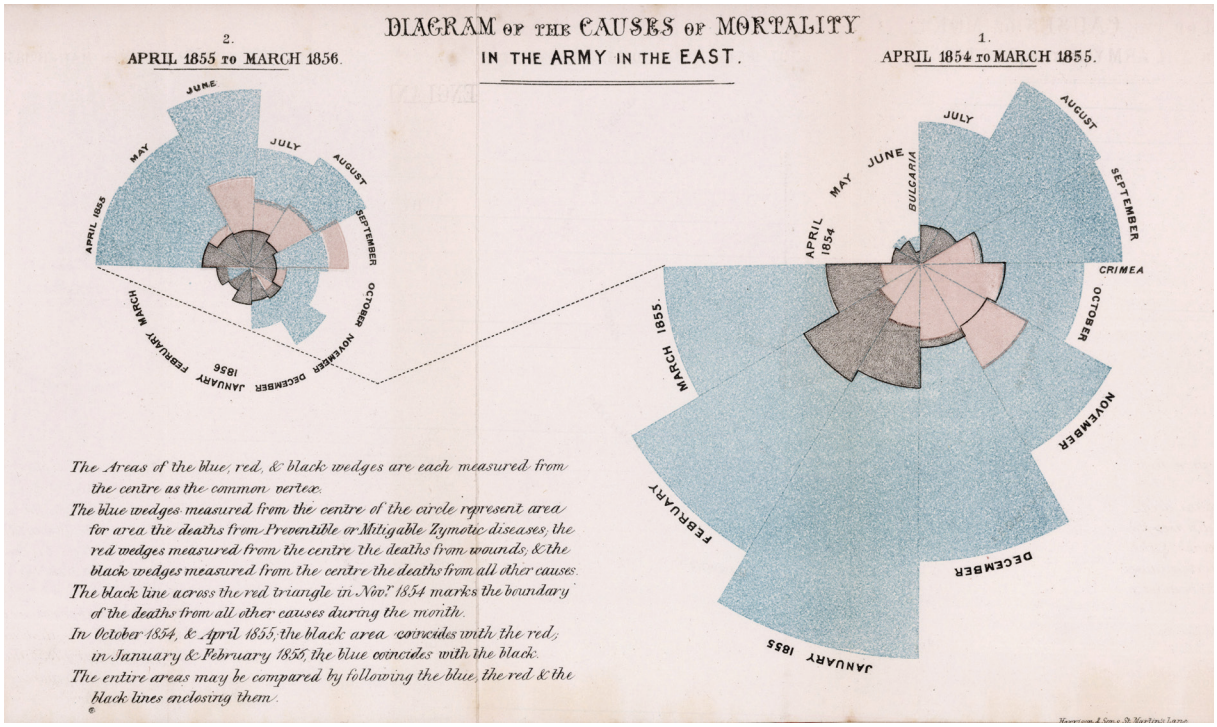



Imagen 1. Gráfico *cowcomb*.

Gracias a sus análisis y a las evidencias que mostraban sus visualizaciones, fue capaz de demostrar tendencias estacionales en los soldados que eran evitables con un conjunto de medidas básicas de higiene y orden. Esto mostró el grave problema en la sanidad que recibían y las condiciones de los hospitales.

Esas malas condiciones y la falta de cuidados eran, en realidad, la causa de la muerte de muchos soldados, y no las heridas de guerra, como se creía (la tasa de mortalidad por herida de guerra era de 1/10).

Una vez averiguado esto gracias a sus análisis, las cifras de mortalidad descendieron de forma drástica. Su trabajo de investigación fue todo un éxito, y por ello fue conmemorada por el ejército inglés, el mismo que fue bastante reticente a la hora de digerir las conclusiones que el trabajo de Nightingale mostró, ya que esto quería decir que las prácticas desempeñadas por el ejército no eran eficientes (Sloane, 2016, p. 191).

**IMPORTANTE**

El objetivo es crear visualizaciones que llamen a la acción del cambio.

El trabajo de un analista consiste en encontrar *insights* que llamen a la acción. Cuanto más visuales sean estos *insights*, antes se iniciará el cambio. Se denomina *insights* a la habilidad de entender y profundizar sobre algún tema poco entendible o complicado, es decir son aquellas ideas que actúan como palanca a la hora de tomar una decisión.

El objetivo de un *insight* es la aclaración de un problema que no era evidente y que nos orienta a tomar una acción. Esto nos ayuda a entender un problema y nos dirige a ejecutar algo. Convertir los *insights* en acciones es cómo conseguimos la ventaja competitiva a través de los datos, donde una buena visualización juega un papel crucial a la hora de fomentar el cambio.

### 1.1.2. ¿POR QUÉ SE CREAN VISUALIZACIONES?

**CITA**

Como dijo Henryk Ibsen, “una imagen vale más que mil palabras” (Ibsen, 1918).

Esto se debe a que nuestro cerebro retiene el 10 % de lo que oímos, 20 % de lo que leemos y 90 % de lo que vemos y hacemos (Dale, 1954).

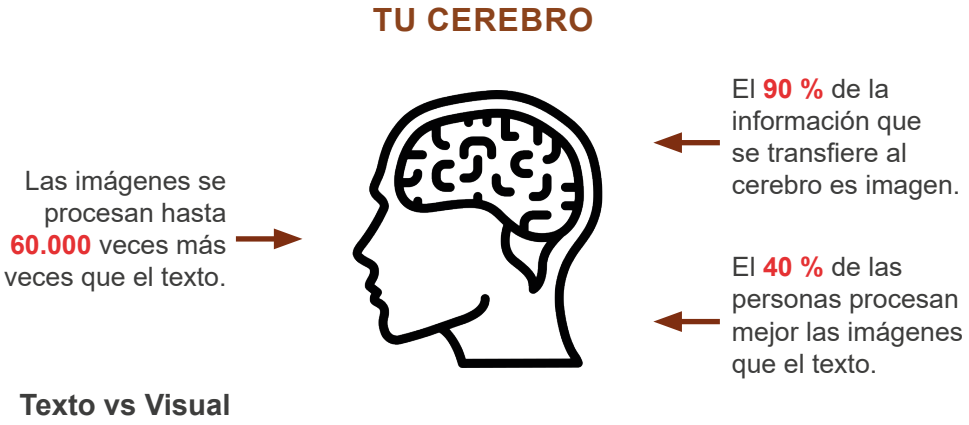



Imagen 2. Retención por imagen.

Nuestro cerebro es capaz de visualizar 60 000 veces más rápido los elementos visuales que un texto, lo que es crucial en el mundo de los datos, ya que gracias a esto somos capaces de encontrar patrones y relaciones más rápidamente en imágenes que en textos.

### 1.1.3. CUARTETO DE ANSCOMBE

Como ya se ha visto, la visualización de datos es un gran aliado a la hora de explicar tendencias, identificar valores atípicos y buscar *insights* en grandes volúmenes de datos. Por ejemplo, pensemos en las conclusiones que pueden sacarse a simple vista de la siguiente tabla:

| I | | II | | III | | IV | |
|---|---|---|---|---|---|---|---|
| x | y | x | y | x | y | x | y |
| 10 | 8.04 | 10 | 9.14 | 10 | 7.46 | 8 | 6.58 |
| 8 | 6.95 | 8 | 8.14 | 8 | 6.77 | 8 | 5.76 |

| I | | II | | III | | IV | |
|---|---|---|---|---|---|---|---|
| x | y | x | y | x | y | x | y |
| 13 | 7.58 | 13 | 8.74 | 13 | 12.7 | 8 | 7.71 |
| 9 | 8.81 | 9 | 8.77 | 9 | 7.11 | 8 | 8.84 |
| 11 | 8.33 | 11 | 9.26 | 11 | 7.81 | 8 | 8.47 |
| 14 | 9.96 | 14 | 8.1 | 14 | 8.84 | 8 | 7.04 |
| 6 | 7.24 | 6 | 6.13 | 6 | 6.08 | 8 | 5.25 |
| 4 | 4.26 | 4 | 3.1 | 4 | 5.39 | 19 | 12.5 |
| 12 | 10.8 | 12 | 9.13 | 12 | 8.15 | 8 | 5.56 |
| 7 | 4.82 | 7 | 7.26 | 7 | 6.42 | 8 | 7.91 |
| 5 | 5.68 | 5 | 4.74 | 5 | 5.73 | 8 | 6.89 |

Fuente: elaboración propia.

El cuarteto de Anscombe consiste en cuatro conjuntos de datos, cada conjunto tiene dos variables x e y, estos conjuntos comparten una serie de estadísticos: media, varianza, correlación y recta de regresión.

El objetivo de Frank Anscombe es demostrar la importancia de mirar gráficamente un conjunto de datos antes de sacar conclusiones. Tal y como podemos observar, a pesar de que dichos conjuntos tienen las mismas propiedades estadísticas, visualmente son completamente diferentes (Anscombe, 1973):

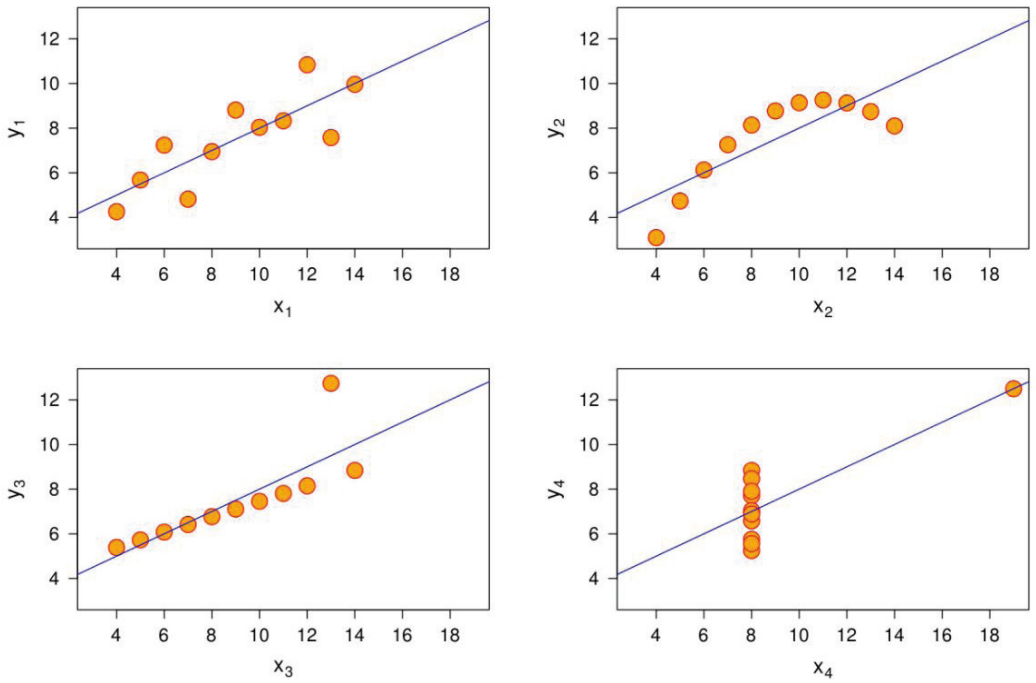



Imagen 3. Cuarteo de Anscombe.

Por lo tanto, aunque hay veces que vemos claramente cómo se comporta un conjunto de datos, tan solo con sus valores estadísticos y los valores de su tabla, es importante representarlos gráficamente para ver si cumplen la expectativa.

## 1.2. PROCESO DE VISUALIZACIÓN

Los datos sin orden y una correcta visualización son poco más que datos, para encontrar los *insights* que nos ayuden a entender el problema Ben Fry recomienda seguir una serie de pasos detallados en su libro *Visualizing Data* (2008).

Antes de comenzar el proceso, tenemos que preguntarnos:

- ¿Cuál es el objetivo de guardar esta información?
- ¿Qué tiene de interesante?
- ¿Qué *insights* puede mostrarnos?

Lo más importante para conocer los datos es identificar el problema que queremos resolver. En lugar de pensar en la información que ya tenemos, tenemos que pensar en cómo vamos a utilizarla, y en si es realmente valiosa la respuesta de esta pregunta. Debemos tener en cuenta que cuanto más específicas sean estas preguntas, más claro será el objetivo.

Una de las habilidades más destacadas a la hora de conocer los datos es formular buenas preguntas. Una buena pregunta demuestra que tenemos interés por solventar el problema y, por ende, por encontrar el insight que convenza a los demás para tomar decisiones; por lo tanto, en este apartado se valora más la curiosidad que una visión matemática.

**IMPORTANTE**

El éxito de una buena visualización, al igual que cualquier otro tipo de comunicación, consiste en la habilidad de hacer entender el mensaje a la audiencia.

Para conseguir nuestro objetivo, Ben Fry estableció los siguientes pasos: *acquire, parse, filter, mine, represent, refine* y *interact.*

### 1.2.1. ESTRUCTURA

- **Acquire:** en este paso, se obtienen los datos, independientemente de la fuente. Es importante darles una estructura para que su búsqueda en grandes bases de datos sea fácil y rápida.
- **Parse:** se le da forma y estructura al significado de los datos con el objetivo de ordenarlos en categorías. En este paso es muy importante darle a cada columna el tipo de dato que requiere, por ejemplo, String, Float, Character, Integer, Index.

- **Filter:** este paso consiste en quedarnos solo con los datos de nuestro interés. Para ello, en algunos casos será necesario quedarnos solo con una parte del texto de una columna o descartar parte de los datos o, si nuestro objetivo es modelizar, probablemente tengamos que normalizar los datos (0 1).

- **Mine:** en este paso, se aplican metodologías estadísticas (*data mining*) para buscar patrones en un contexto matemático.

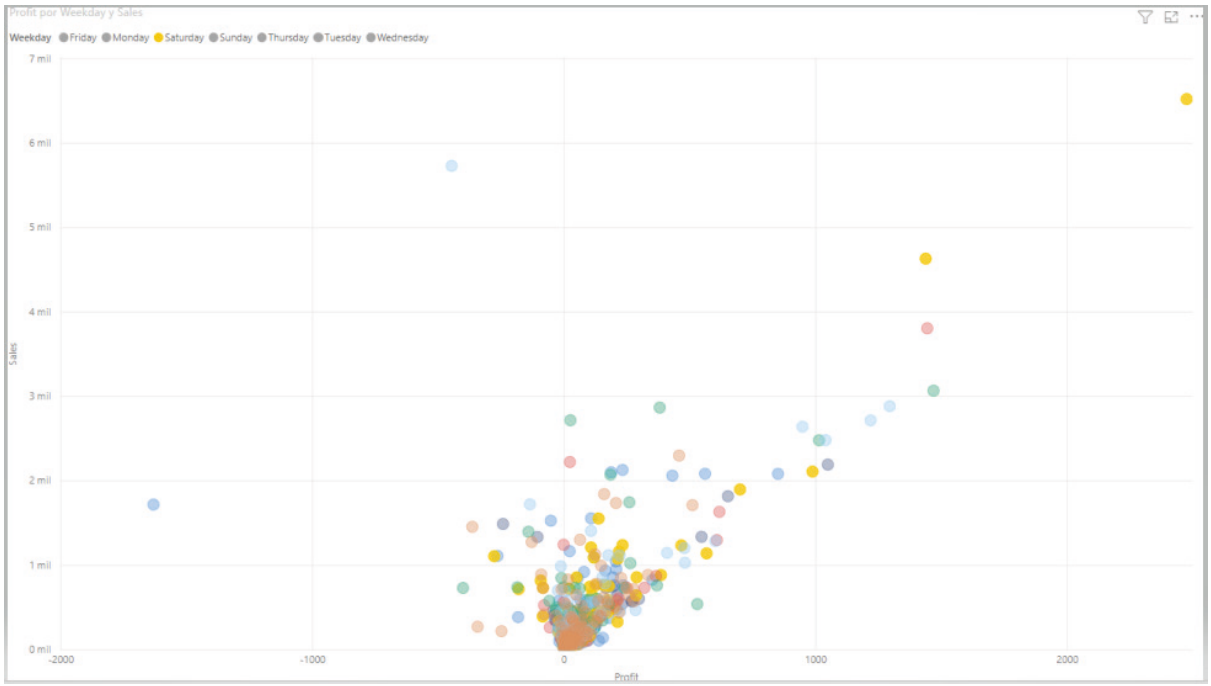



Imagen 4. Patrones de *data mining*.

En este caso, estamos analizando las ventas por país, viendo la relación entre beneficio y cantidad y observamos que, de los días de la semana, representados por colores, los sábados son los que proporcionan mejores resultados.

- **Represent:** en función del tipo de dato y del objetivo que queramos abordar, es importante elegir correctamente el visual para el conjunto de datos.

- **Refine:** este paso consiste en mejorar la anterior representación con el objetivo de captar la atención de nuestra audiencia. Al establecer relaciones entre distintos campos/tablas, podemos bajar o subir el nivel de detalle de nuestra visualización.

- **Interact:** se hace que los gráficos sean interactivos en función del objeto que queramos analizar. Esto es muy importante, ya que, de esta forma, le damos la opción al usuario de que sea él quien interactúe con las visualizaciones.

**IMPORTANTE**

No obstante, aplicar uno o todos los pasos dependerá del *business case*. Por lo general en el ámbito empresarial, este flujo será elaborado por varios miembros. En estos casos, es muy importante que ninguno de los miembros pierda el objetivo predeterminado, que ha sido elaborado en fase de preguntas.

## 1.2.2. COMBINACIÓN E INTERACCIÓN

Tal y como podemos apreciar en la **Imagen 5**, cuanto más tarde se realicen cambios en el proceso, más implicaciones y modificaciones se obtendrán en los pasos primarios. Por ello, es muy importante tener claras las necesidades para trasladarlo en los pasos primarios. Esto hará que el proceso de trabajo sea mucho más productivo.

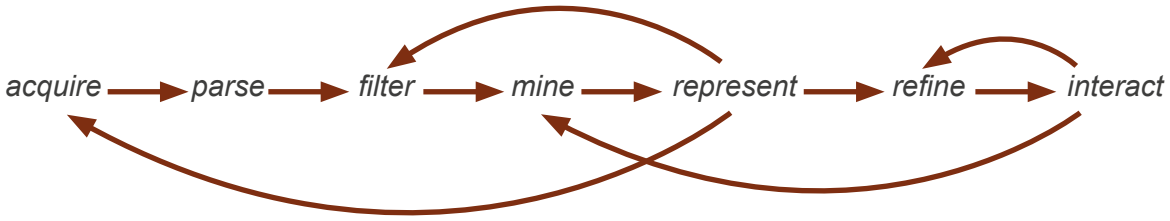



Imagen 5. Interacciones entre las siete fases. Fuente: Fry, B. (2008). *Visualizing Data*. O'Reilly.

En equipos compuestos por diferentes perfiles (ingenieros, negocio y diseñadores), es importante plantearnos las siguientes preguntas:

- A la hora de adquirir la información, ¿cuál es su cadencia de actualización?
- En cuanto a la visualización, ¿cómo consideramos cuándo hay una nueva categoría?

## 1.2.3. RECOMENDACIONES

En este apartado, se presentarán una serie de principios que ayudan a las presentaciones de visualización de datos:

- **Cada proyecto tiene unas necesidades únicas:** cada visualización es única, por lo que es importante enfatizar aquellos puntos que puedan ayudar a solventar el problema.
- **Menos es más:** mostrar toda la información que hemos recopilado es un error muy común, pues esto hace que nuestra audiencia se distraiga de lo realmente importante: los *insights* que harán actuar a nuestra audiencia. Un claro ejemplo es el mapa del parte meteorológico: si mezclamos corrientes atmosféricas, temperaturas máximas y mínimas, el usuario estará perdido, ya que dispondrá de demasiada información.
- **La importancia de conocer nuestra audiencia:** para ello, debemos conocer cuál es el objetivo de nuestra audiencia con esta visualización, qué pretenden aprender y qué papel desempeñan. En función de su rol, podremos profundizar más o menos en el contenido.

# 1.3. TIPOS DE GRÁFICOS Y CÓMO ELEGIRLOS

A la hora de desarrollar una visualización es primordial hacernos las siguientes preguntas:

- **Serie temporal.** ¿Los datos forman parte de una serie temporal?
- **Jerarquía.** ¿Conforman una jerarquía? (por ejemplo: país, región, provincia, municipio…). De estas, ¿cuáles hay que representar? ¿Cuáles son más importantes?
- **Variación entre datos.** ¿Existe una gran variación entre los datos?
- **Categorización de los datos.** ¿Son medidas cuantitativas o cualitativas?
  - Variables cuantitativas: aquellas variables medibles (por ejemplo, salario, cantidad).
  - Variables cualitativas: son aquellas variables no medibles (por ejemplo, región, provincia, color…), ya que estas variables no son numéricas, sino que son categóricas.

A continuación, veremos en detalle los principales tipos de gráficos.

## 1.3.1. TIPOS DE GRÁFICOS

### BÁSICOS

#### BARRAS

Se trata de una forma de representar gráficamente un conjunto de datos o valores mediante barras rectangulares de longitud proporcional a los valores representados (Der, 2014).

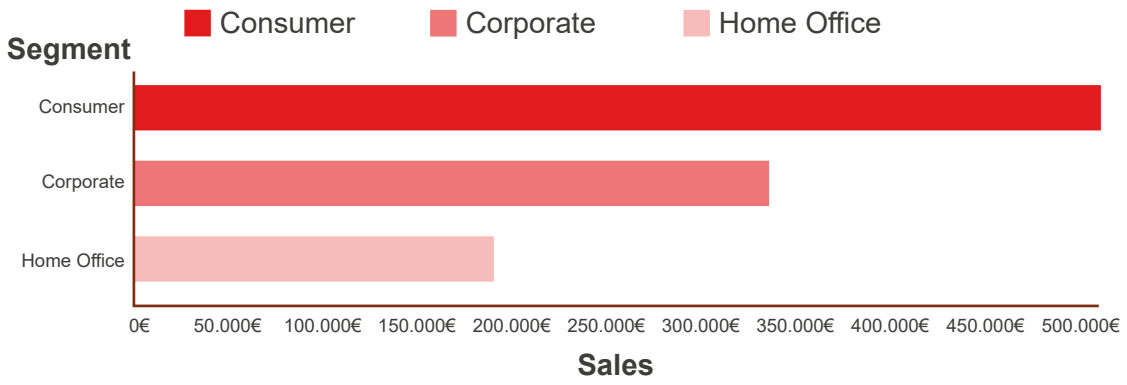



Imagen 6. Gráfico de barras. Fuente: elaboración propia.

- **Ventajas:** es útil y fácil de crear, e ideal para variables discretas.
- **Desventajas:** no se pueden representar variables continuas.

## BARRAS APILADAS

Es un diagrama compuesto por un rectángulo dividido en porciones, de tal manera que cada una de estas porciones es proporcional a la magnitud que representa (García Mendoza, 2009).

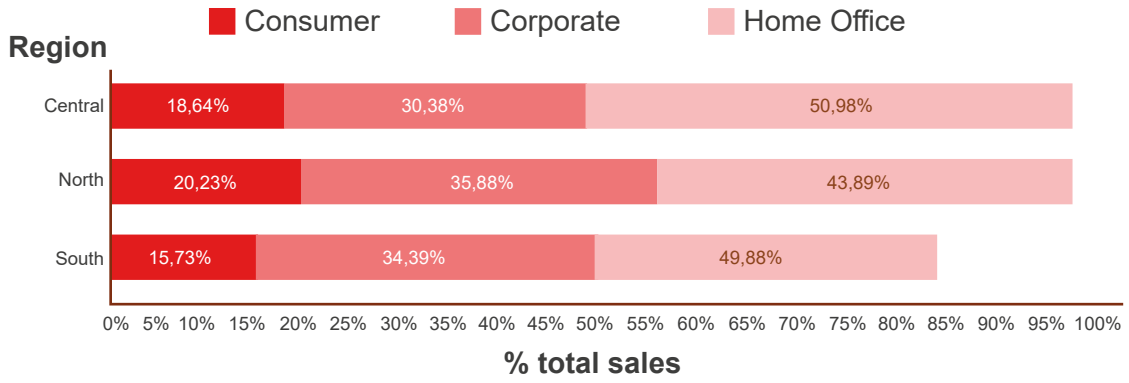



Imagen 7. Gráfico de barras apiladas. Fuente: elaboración propia.

- **Ventajas:** múltiples categorías y series en una misma área.
- **Desventajas:** no es buena idea cuando hay muchas categorías.

## DIAGRAMA BIKINI

Se trata de una forma de representar gráficamente un conjunto de datos o valores mediante barras rectangulares de longitud proporcional a los valores representados, divididos por categorías desde el eje superior.

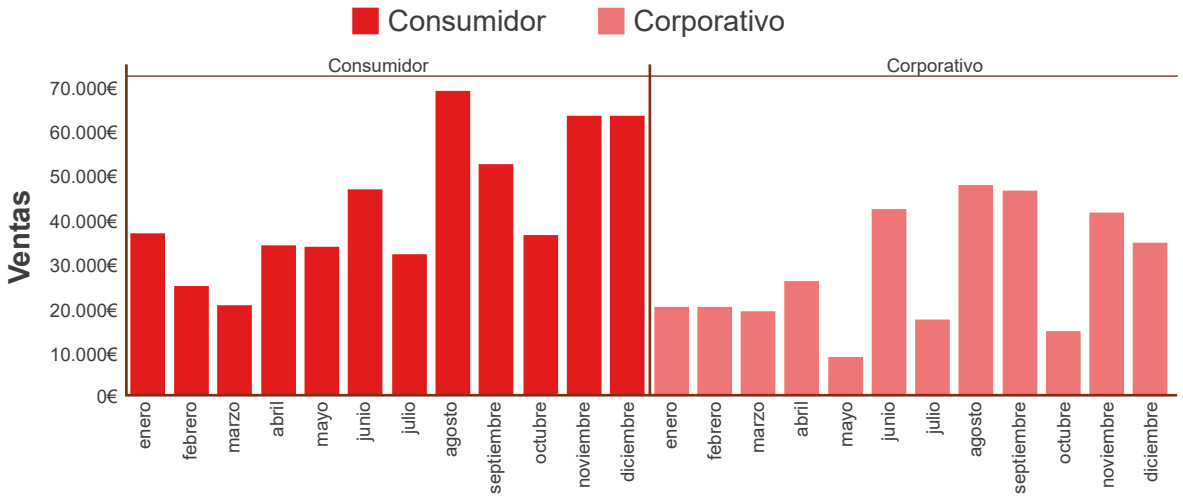



Imagen 8. Gráfico de diagrama de bikini. Fuente: elaboración propia.

- **Ventajas:** es útil y fácil de crear, ideal para variables discretas.
- **Desventajas:** no se pueden representar variables continuas.

## TARTA

Estos gráficos nos permiten ver la distribución interna de los datos que representan una variable categórica en forma de porcentajes sobre un total (García Mendoza, 2009).

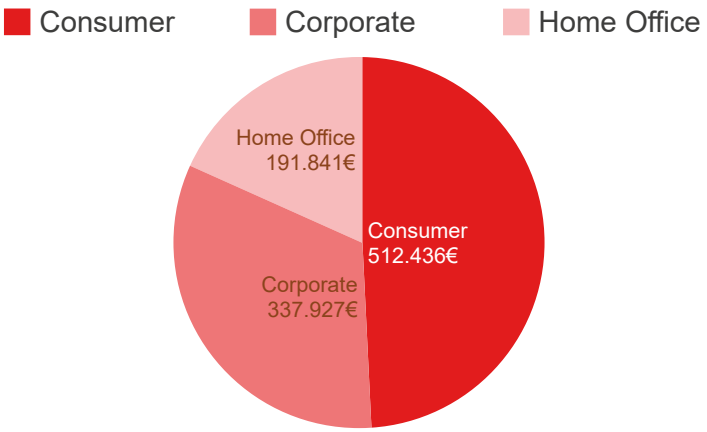



Imagen 9. Gráfico de tarta. Fuente: elaboración propia.

- **Ventajas:** ideal para visualizar categorías de una forma rápida y sencilla.
- **Desventajas:** no se recomienda cuando hay demasiadas categorías. En este caso, es recomendable agruparlas para tener una visión más general.

## HISTOGRAMA

Representación en forma de barras donde la longitud de las barras indica el valor del eje X, este eje tiene unos valores que se agrupan entre sí.

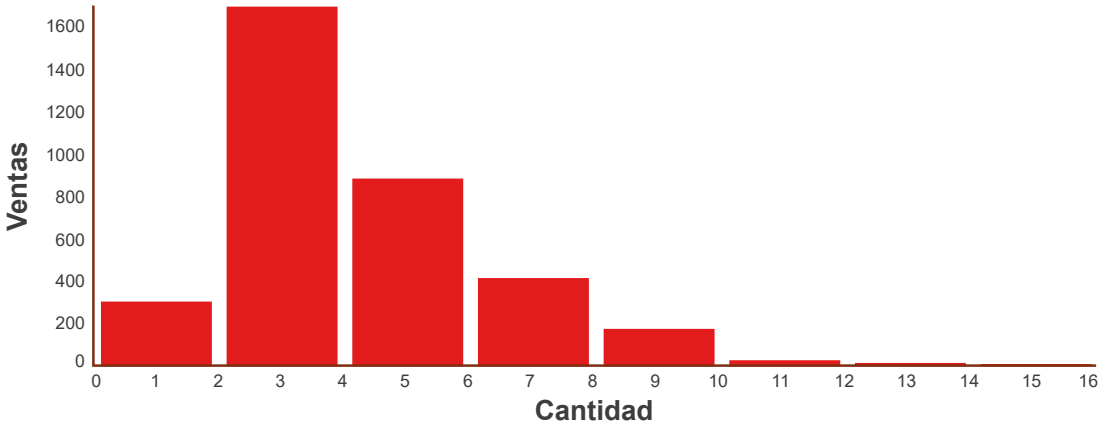



Imagen 10. Gráfico histograma. Fuente: elaboración propia.

- **Ventajas:** ayuda a visualizar la distribución de una serie numérica.
- **Desventajas:** no se le pueden añadir variables categóricas.

### *SPARKLINE*

Los valores se representan mediante un punto, que se unen mediante líneas para facilitar su visualización (García Mendoza, 2009).

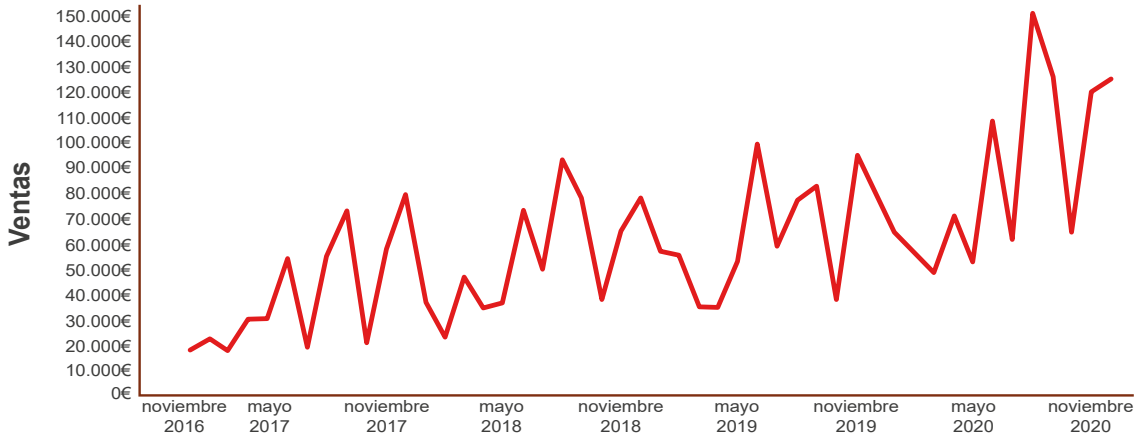



Imagen 11. Gráfico *sparkline*. Fuente: elaboración propia.

- **Ventajas:** como otros gráficos que se han visto anteriormente, es fácil de interpretar y realizar, ideal para representar series temporales.
- **Desventajas:** dificultad si hay demasiadas variables mezcladas entre sí.

## ÁREAS

### ÁREA APILADA

Se utiliza para mostrar las diferentes partes que integran un total. Los valores del componente mayor se localizan primero y luego se les une con una línea.

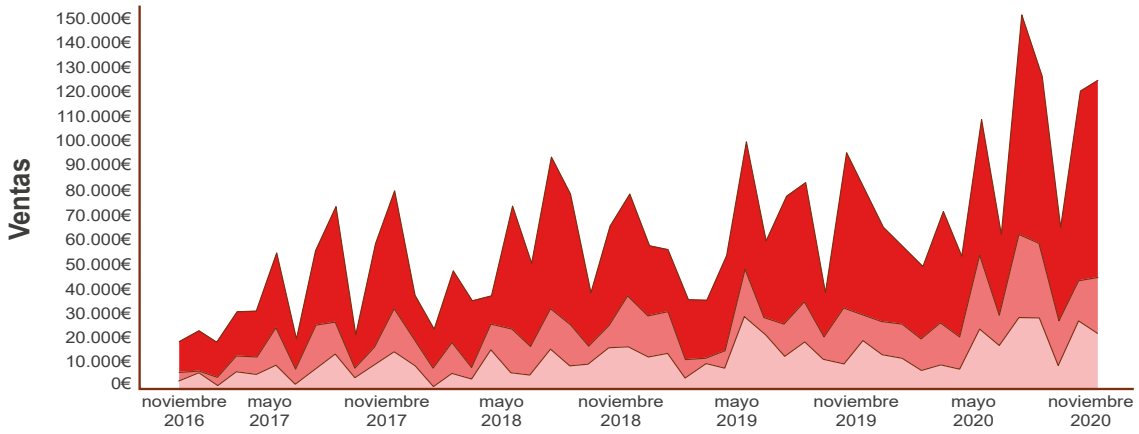



Imagen 12. Gráfico de áreas. Fuente: elaboración propia.

- **Ventajas:** visualización de cómo varía una categoría respecto a otra a lo largo de un eje temporal.
- **Desventajas:** cuando las variables cambian entre sí (pasa de la posición 1 a 2) este paso no se aprecia claramente, ya que las áreas no se mezclan entre sí.

### *STREAMGRAPH*

Se utiliza para mostrar las diferentes partes que integran un total. Los valores del componente mayor se localizan primero y luego se les une con una línea. Estos valores confluyen sobre un eje central, de forma que podemos ver cómo varía a lo largo del eje.

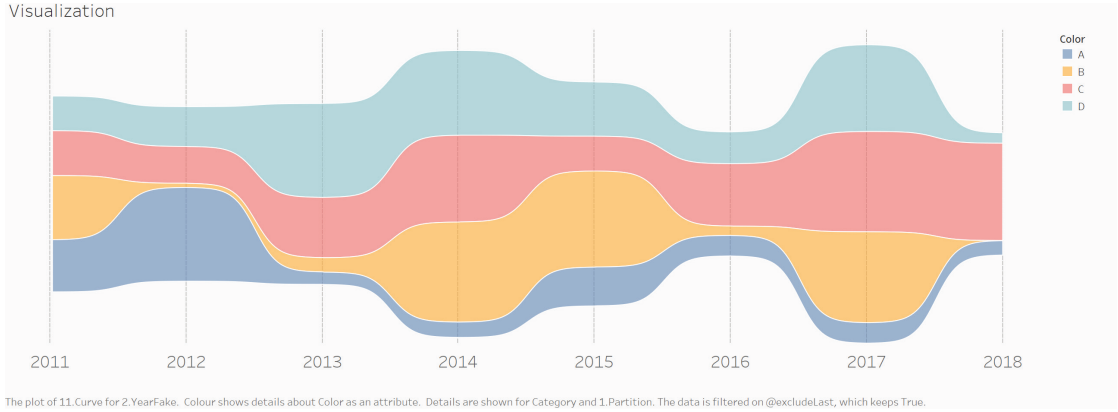



Imagen 13. Gráfico *streamgraph*. Fuente: elaboración propia.

- **Ventajas:** ayuda a representar cambios porcentuales a lo largo de un eje temporal.
- **Desventajas:** difícil elaboración y no se pueden utilizar demasiadas categorías.

## JERARQUÍA

### *TREEMAP*

Se utiliza para visualizar como está conformado una categoría en forma rectangular, el tamaño de estos rectángulos hace referencia al volumen.

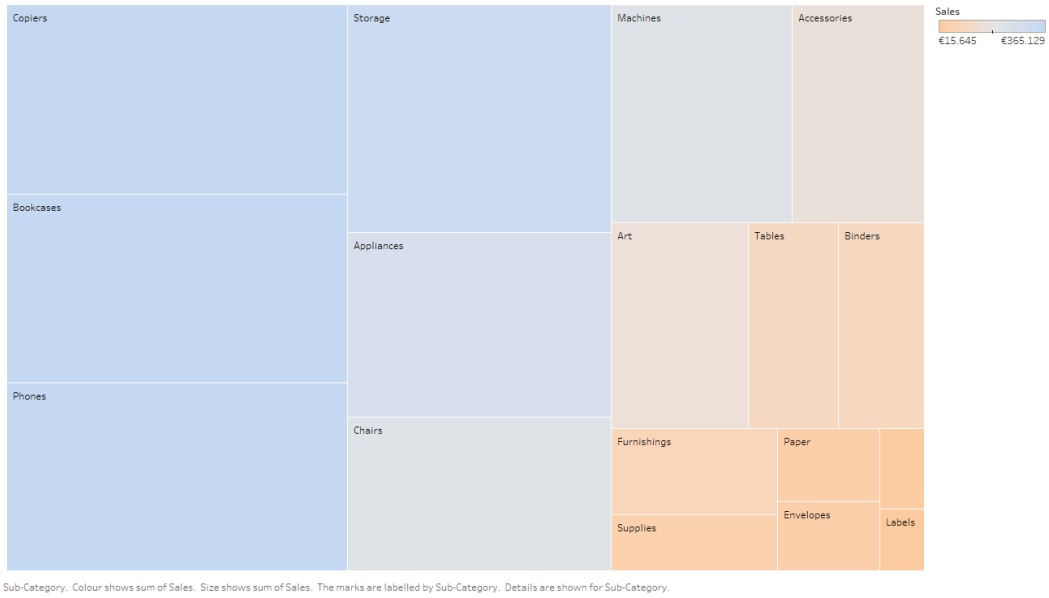



Imagen 14. Gráfico *treemap.* Fuente: elaboración propia.

- **Ventajas:** se pueden visualizar gran cantidad de datos, y permite clasificar en varias categorías jerárquicas.
- **Desventajas:** no visualiza los datos en varias magnitudes, solo admite valores positivos.

## *CIRCLEPACKING*

Representación de distintas categorías en forma de burbujas, asignándole tamaño a cada una de las burbujas en función de una variable cuantitativa.

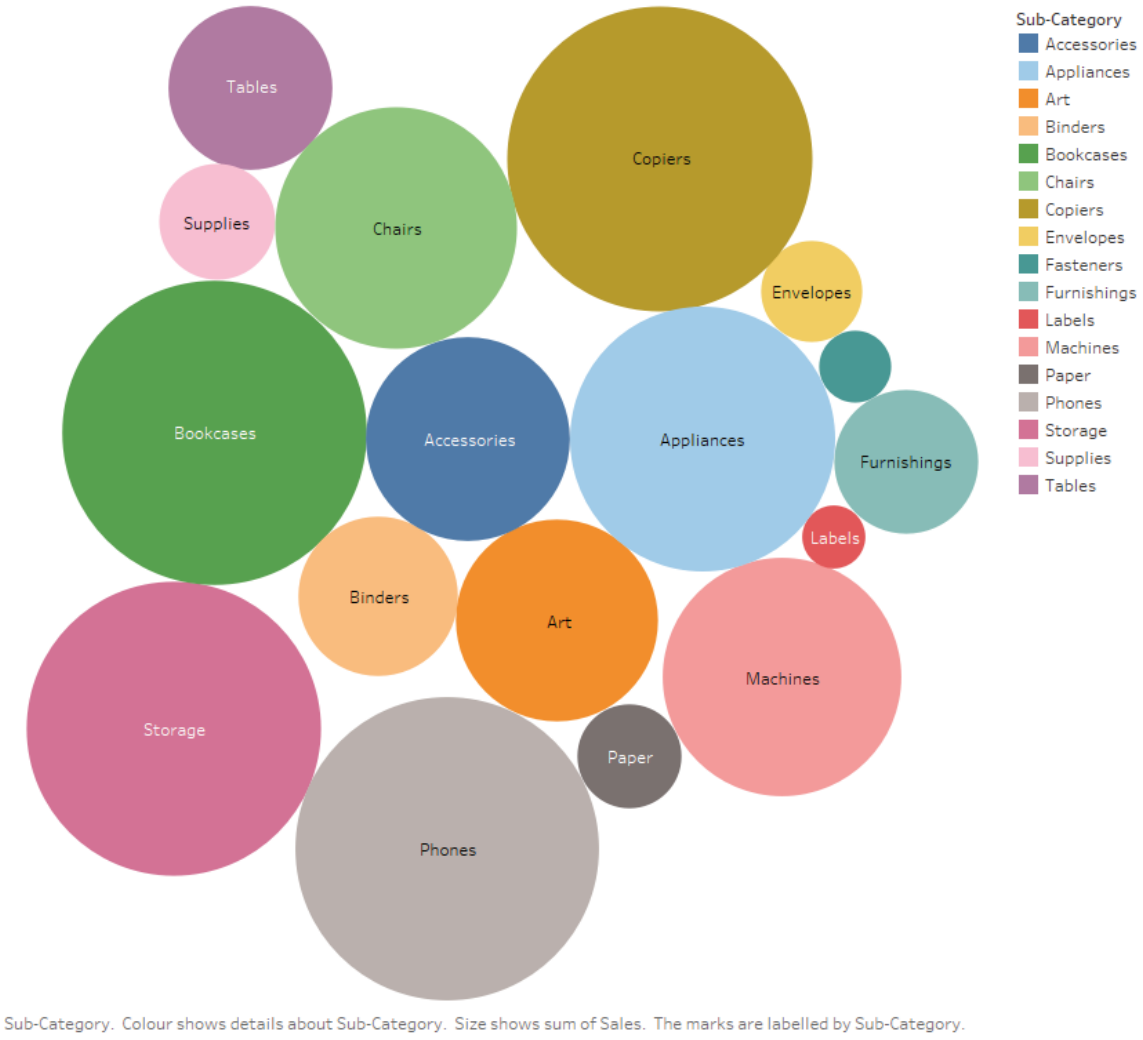



Imagen 15. Gráfico de *circlepacking.* Fuente: elaboración propia.

- **Ventajas:** permite la visualización de la distribución de la variable cuantitativa en cada una de las categorías.
- **Desventajas:** no permite comparar categorías entre sí con valores similares.

### *SUNBURST*

Es una ampliación de gráfico de tarta que incluye distintos niveles jerárquicos (por ejemplo, continente-países).

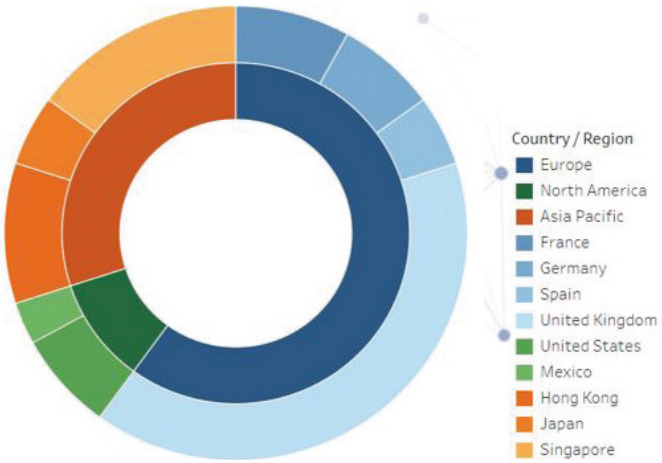



Imagen 16. Gráfico *sunburst.* Fuente: Hoang, s. f.

- **Ventajas:** es buena elección para visualizar los niveles jerárquicos.
- **Desventajas:** no se pueden comparar categorías entre sí con valores similares.

## MULTIVARIABLE

### *SCATTERPLOT*

Muestra la asociación entre dos variables cuantitativas, donde cada relación se ve visualizada en un punto.

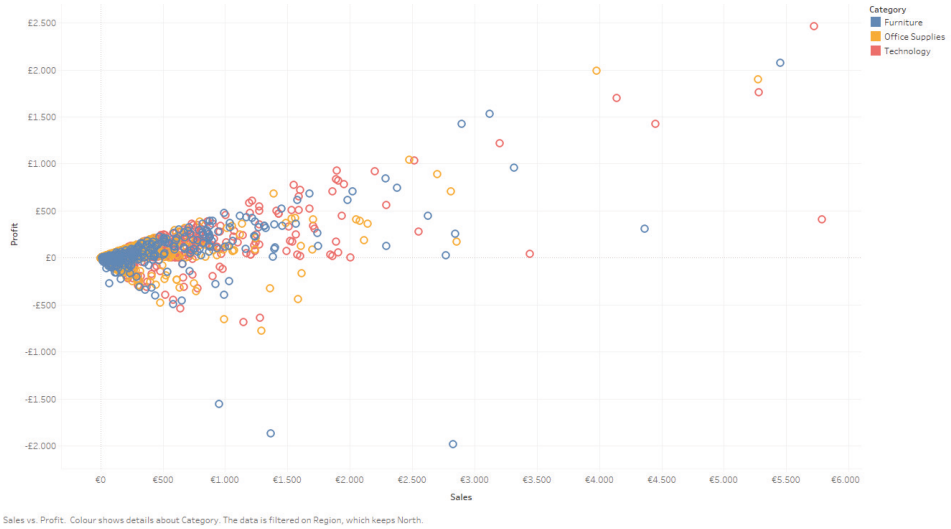



Imagen 17. Gráfico *scatterplot.* Fuente: elaboración propia.

- **Ventajas:** es perfecto para mostrar la tendencia entre dos variables.
- **Desventajas:** correlación no significa causalidad.

### *BUBBLE CHART*

Muestra la relación entre dos variables, donde el tamaño de la burbuja lo indica una tercera variable.

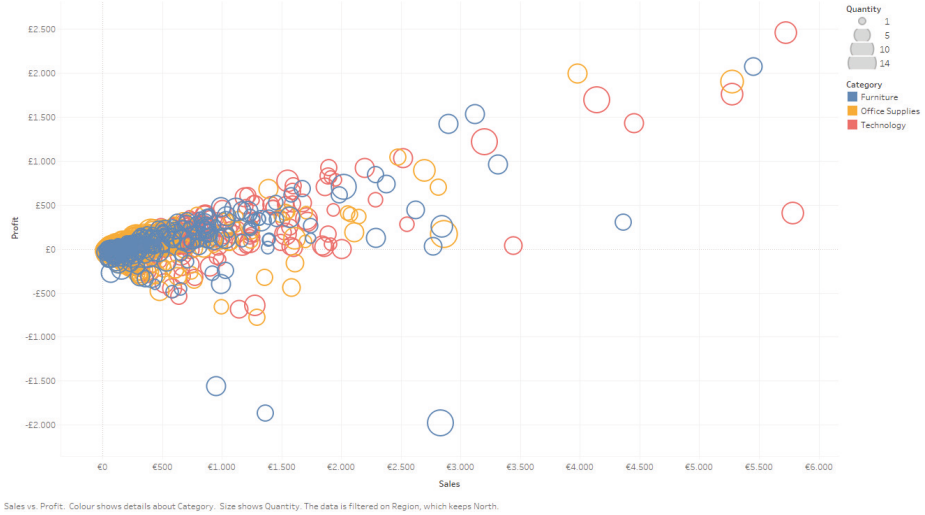



Imagen 18. Gráfico de burbujas. Fuente: elaboración propia.

- **Ventajas:** muestra la relación entre tres o más variables.
- **Desventajas:** demasiadas burbujas pueden distraer a la audiencia del *insight* que queremos mostrar.

## GRAFOS

### DIAGRAMA DE RED

Muestra la relación existente entre varios puntos.

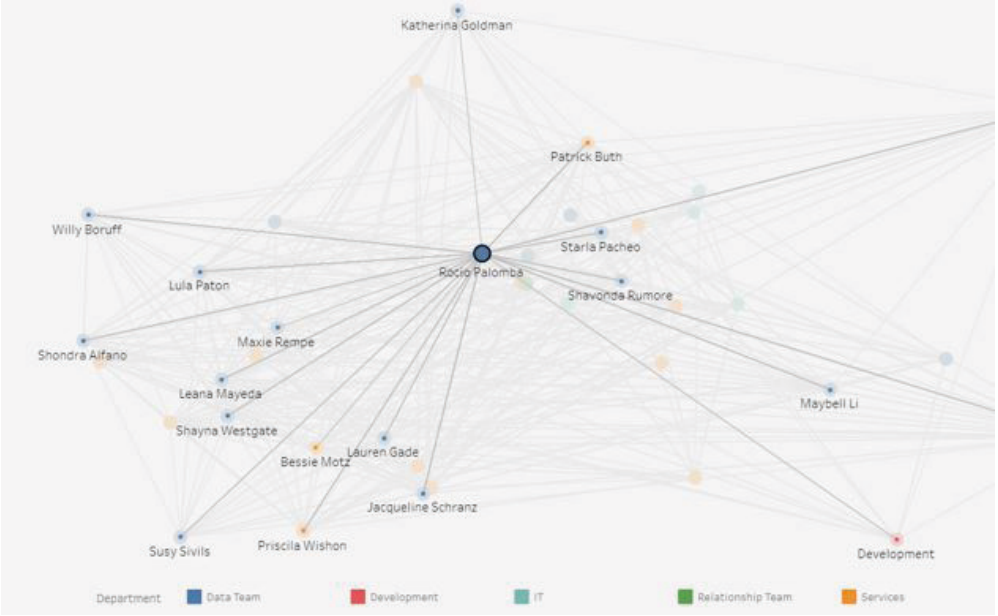



Imagen 19. Gráfico de diagrama de red. Fuente: Hiscock, s. f.

- **Ventajas:** este tipo de gráfico ayuda a visualizar interacciones entre puntos (por ejemplo, en redes sociales, el impacto de un *tweet*).
- **Desventajas:** aunque está pensado para albergar cantidades ingentes de datos, si la muestra se dispara es difícil ver las relaciones entre sí.

## DIAGRAMA DE CUERDAS

Muestra, de forma circular, las relaciones entre una o varias entidades. El grosor de las relaciones se representa por una variable categórica.

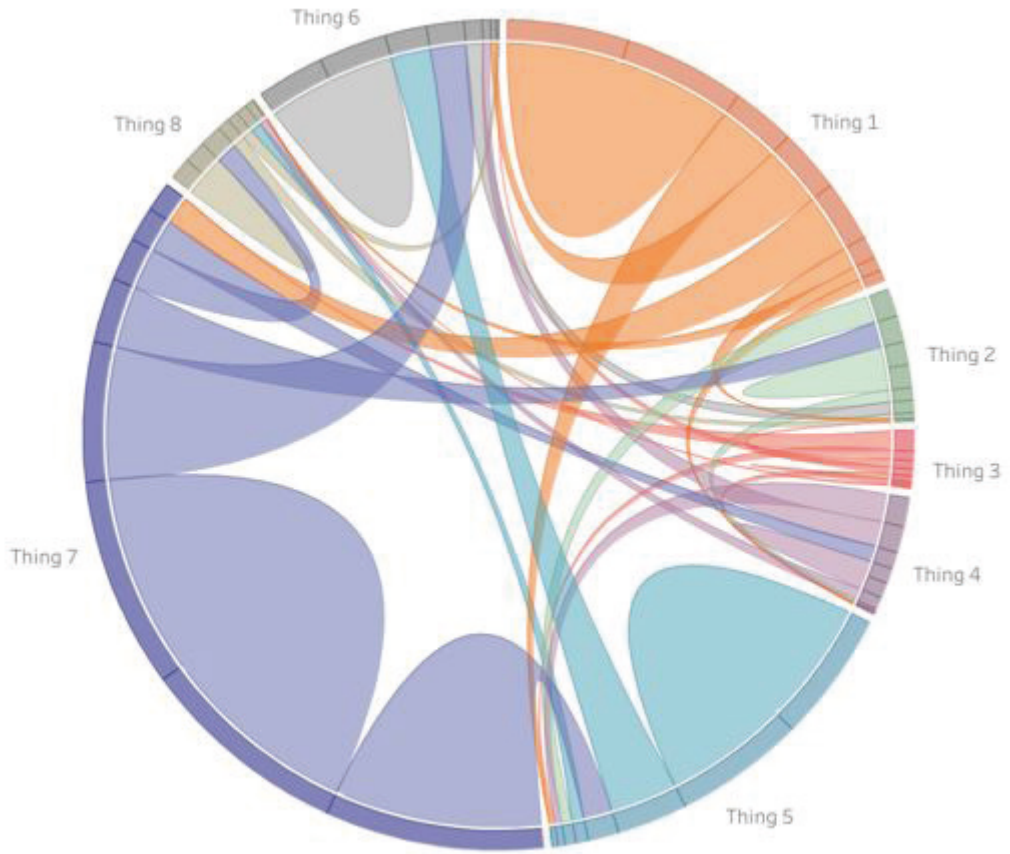



Imagen 20. Gráfico de diagrama de cuerdas. Fuente: Salvaterra, s. f.

- **Ventajas:** ayuda a obtener una idea general de las relaciones entre las variables.
- **Desventajas:** son solo prácticas para tener una idea general, por lo que no es aconsejables para bajar a nivel de detalle.

### DIAGRAMA DE SANKEY

Muestra, de forma rectangular, las relaciones entre una o varias entidades entre dos extremos. El grosor de las relaciones se representa mediante una variable categórica.

Imagen 21. Gráfico de diagrama de Sankey. Fuente: Flerlage s. f.

- **Ventajas:** ideal para tener una idea general de las relaciones entre las variables.
- **Desventajas:** son solo prácticas para tener una idea general, no son aconsejables para bajar a nivel de detalle.

## GEOGRÁFICOS

### COROPLETAS

Visualización en forma de polígono de un área predeterminada en un mapa.

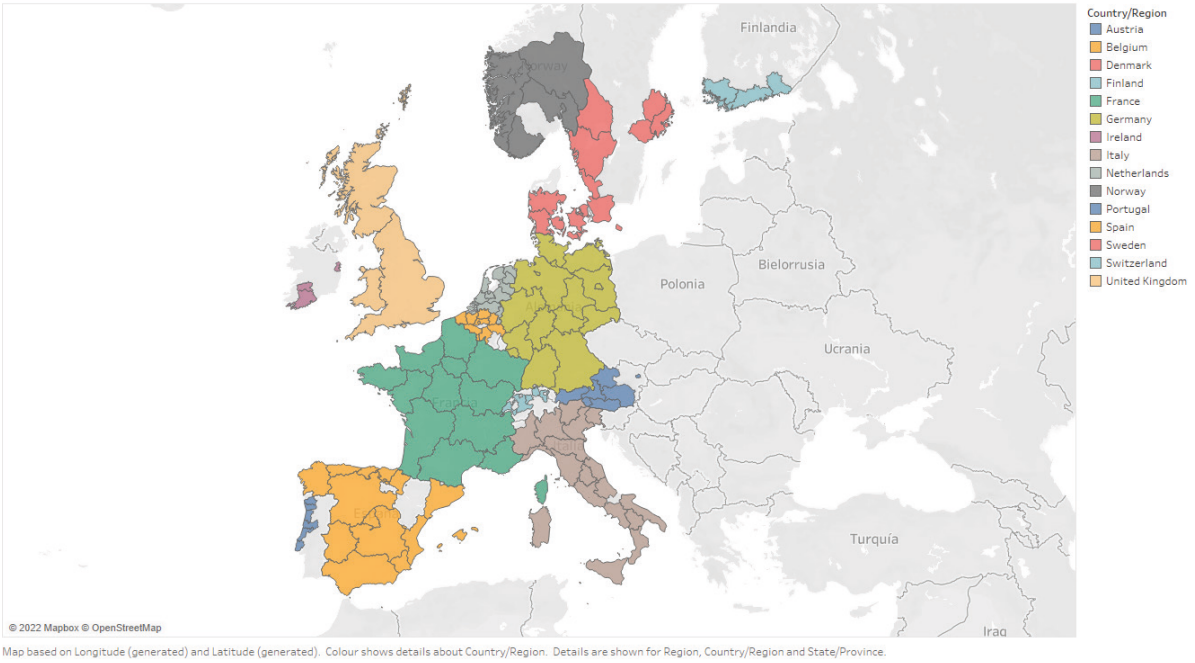



Imagen 22. Mapa coroplético. Fuente: elaboración propia.

- **Ventajas:** permite visualizar una categoría de forma agrupada y geoposicionada, y permite subir y bajar el nivel de detalle de forma jerarquizada.
- **Desventajas:** solo se puede representar una variable cuantitativa.

## MAPA DE BURBUJAS

Visualización en forma de burbuja de un punto predeterminado por una latitud y longitud en un mapa. La amplitud de la burbuja representa una variable categórica.

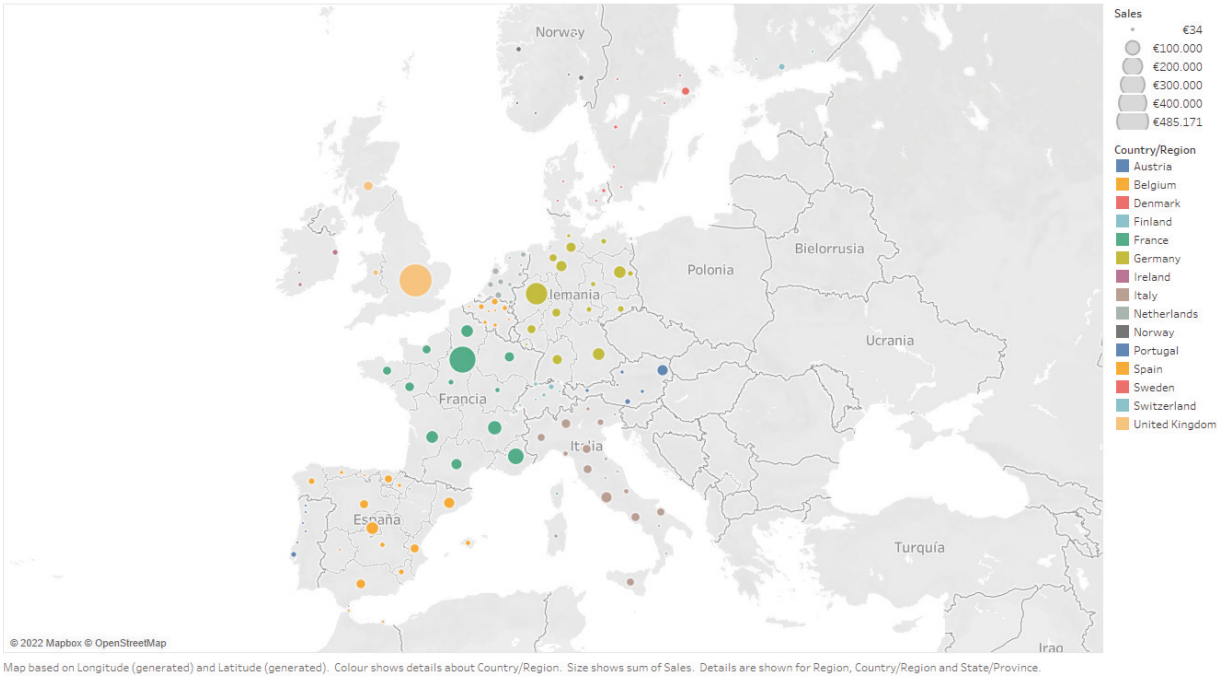



Imagen 23. Mapa de burbujas. Fuente: elaboración propia.

- **Ventajas:** ayuda a visualizar la distribución de una variable cuantitativa.
- **Desventajas:** es complicado saber con exactitud el valor de cada punto.

## MAPA DE CONEXIONES

Establece conexiones entre uno o más puntos georreferenciados mediante latitud y longitud en mapa.

Imagen 24. Mapa de conexiones. Fuente: Tableau. s. f.

- **Ventajas:** ideal para medir de forma visual las interacciones entre varios puntos.
- **Desventajas:** si disponemos de una muestra demasiado grande, no visualizaremos con detalle cada uno de los puntos.

# 1.4. EL ERROR EN LA REPRESENTACIÓN DE LA INFORMACIÓN

## 1.4.1. ERRORES COMUNES EN LA REPRESENTACIÓN DE UN GRÁFICO

Según García Mendoza (2009), los principales errores que se suelen cometer son los siguientes:

- Mostrar de una forma clara una leyenda.
- Utilizar títulos extremadamente extensos.
- Presentar gráficos muy cargados o sumamente complejos de interpretar.
- Omitir los rótulos de los ejes o las unidades de medida.
- Presentar demasiadas alternativas en un solo gráfico. En este sentido, no se deben permitir más de cinco o seis elementos en un gráfico de línea, un gráfico de barras no debería de tener más de quince barras, y uno circular, no más de ocho componentes.

- Seleccionar adecuadamente la escala, generalmente las escalas comienzan por 0.
- Omitir la leyenda donde se han usado claves o símbolos.

Tal y como ya se ha comentado, a la hora de desarrollar un gráfico es importante enfatizar el *insight* que queremos transmitir.

El siguiente ejemplo muestra de forma clara la importancia de determinar de forma conjunta entre negocio y análisis el *insight* que queremos transmitir.

## EJEMPLO

En la imagen a continuación se muestran los impactos recibidos por los aviones de combate que han regresado a la base.

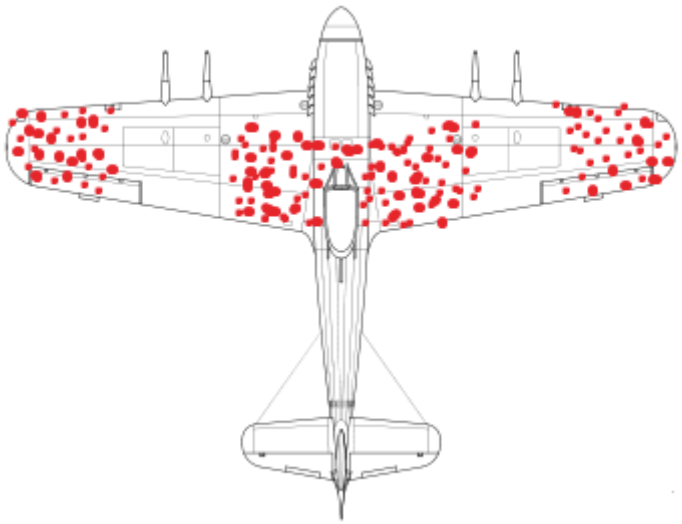


Imagen 25. Estudio de avión que ha regresado a base.

A simple vista, podríamos decir que, reforzando aquellos puntos que han recibido impacto, podemos mejorar la tasa de éxito de nuestro avión combate. No obstante, esta afirmación sería errónea, ya que esta imagen es de los aviones de combate que lograron regresar a la base.

El estadístico Abraham Wald indicó que las zonas que hay que reforzar son aquellas que no han sido atacadas, ya que indican cuáles son los puntos críticos para que un avión no vuelva a la base.

# IDEAS CLAVE

- La importancia del diseño en una visualización es clave.
- El objetivo de la visualización es remarcar los *insights* que permitan a la audiencia ejercer de palanca de cambio.
- Siempre que sea posible, es preferible utilizar visualizaciones para transmitir un mensaje, pues se retiene más fácilmente una imagen que un texto.
- Es importante conocer los datos estadísticos de nuestras variables. Por ello, antes de tomar una decisión al respecto, es conveniente representar dichos datos de manera gráfica.
- A la hora de desarrolla una visualización es importante hacernos las siguientes preguntas: ¿cuál es el objetivo de guardar esta información? ¿Qué tiene de interesante? ¿Qué *insights* puede mostrarnos?
- A la hora de elegir un gráfico, es muy importante conocer el *insight* que queremos transmitir.

# GLOSARIO

- ***Insight.*** Habilidad de entender y profundizar sobre algún tema poco entendible o complicado.
- **Variables categóricas.** Aquella variable que permite clasificar una serie de datos por medio de valores fijos asociados a una cualidad o categoría concreta.
- **Variables discretas.** Aquella variable estadística que, a diferencia de la cualitativa, puede expresarse a través de cifras. Puede analizarse con métodos estadísticos.

# BIBLIOGRAFÍA


Anscombe, F. J. (1973). Graphs in Statistical Analysis. *American Statistician*, 27(1), 17-21.

Dale, E. (1954). *Audiovisual Methods in Teaching*. Dryden Press.

Der, G. (2014). *A Handbook of Statistical Graphics Using SAS ODS*. Chapman and Hal.

Flerlage, K. (s. f.). Sankey with adjustable whitspace. https://public.tableau.com/app/profile/ken.flerlage/viz/HorizontalSankeywithAdjustableWhitespace/Sankey

Fry, B. (2008). *Visualizing Data*. O'Reilly.

García Mendoza, S. (2009). *Tipos de gráficos estadísticos*. Instituto nacional de estadística e informática.

Hiscock, S. (s. f.). Tableau How-to. https://public.tableau.com/app/profile/sophie.hiscock/viz/NetworkCharts1/NetworkVisualisation

Hoang, T. (s.f.). Sunbursts. https://public.tableau.com/app/profile/toan.hoang/viz/Sunbursts_1/Sunbursts

Ibsen, H. (1918). A picture is worth a thousand words. *San Antonio Light.*

Salvaterra, N. (s. f.). Chord template. https://public.tableau.com/views/ChordDiagramTemplate/Chord?%3Aembed=y&%3AshowVizHome=no

Sloane, P. (2016). *Think like an innovator*. Pearson.

Tableau (s. f.). *Create Maps that Show Paths Between Origins and Destinations in Tableau*. Tableau. https://help.tableau.com/current/pro/desktop/en-us/maps_howto_origin_destination.htm

# TEMA 2

## Visualización interactiva con Python

# OBJETIVOS

- Aprender a elaborar visualizaciones gráficas con librería base (Matplotlib).
- Aprender a elaborar gráficas personalizadas (Matplolib–Seaborn).
- Aprender a elaborar gráficas interactivas (Plotly).

En este tema se explorarán las diferentes opciones a la hora de representar una visualización en Python.

Primero se verá Matplotlib, que es una librería bastante simple pero que contiene numerosas funcionalidades.

Posteriormente pasaremos a Seaborn, que se basa en la estructura de Matplotlib pero que permite crear visualizaciones más interesantes. Tiene la ventaja de que su estructura es bastante menos compleja que Matplotlib.

Por último, profundizaremos en Plotly, cuyo uso principal es crear visualizaciones interactivas (ideal para crear *dashboard* interactivos). Su estructura es bastante más compleja que las anteriormente mencionadas, ya que no está ligada directamente con Pandas, sino que lo está con JavaScript. No obstante, el sobresfuerzo tiene su recompensa, pues nos permite crear gráficos interactivos. Además, existe la opción de trabajar con la librería Cufflinks, que nos permitirá conectarnos directamente con nuestro *dataset*.

Esta última parte es muy interesante, ya que de esta forma podríamos publicar nuestro *dashboard* en una página web y compartir el contenido con nuestros compañeros de equipo; no obstante, para mejorar la privacidad del contenido, se recomienda crear una interfaz de *logeo* donde podamos controlar quién accede a nuestra información.

Todo el contenido se desarrollará en Jupyter *Notebook*s, proporcionado por Anaconda.

# 2.1. MATPLOTLIB

## 2.1.1. INTRODUCCIÓN A MATPLOTLIB

Matplotlib es una librería gráfica estándar de Python. Es una de las librerías más sencillas de usar. Está funcionalmente integrada con Pandas y Numpy para mejorar la eficiencia de visualización. Matplotlib proporciona al usuario el total control sobre las fuentes y tipo de gráficos.

En términos de rendimiento, es especialmente eficiente cuando realizamos análisis exploratorios gracias a su integración con Pandas, ya que podemos pasar de un *dataframe* a un gráfico fácilmente.

Matplotlib es ideal para crear gráficos básicos como:

- Gráficos de dispersión.
- Gráficos de barras.
- Gráficos *sparkline.*

## 2.1.2. INSTALACIÓN DE LIBRERÍAS

A la hora de comenzar a trabajar en nuestro *notebook*, lo primero que tenemos que hacer es instalar aquellas librerías que vayamos a utilizar, en este caso:

```
!pip install matplotlib
```

**IMPORTANTE**

Esto sería solo necesario en el caso que no la tuviéramos previamente instalada, usualmente entornos como Anaconda o Anaconda Mini suelen disponer de estas librerías.

Posteriormente, cargamos dicha librería:

```
import matplotlib.pyplot as plt
```

### 2.1.3. GRÁFICO DE LÍNEAS (*SPARKLINE*)

Para realizar un gráfico de líneas, necesitamos un eje X y un eje Y. En el siguiente ejemplo vamos a visualizar las ventas de un producto a lo largo de los años:

```
x = [2016, 2017, 2018, 2019, 2020, 2021] #años

y = [542, 543, 650, 578, 648, 750] #ventas


plt.plot(x, y)

plt.savefig('IMG1.png')

plt.show()
```

El resultado sería el siguiente:

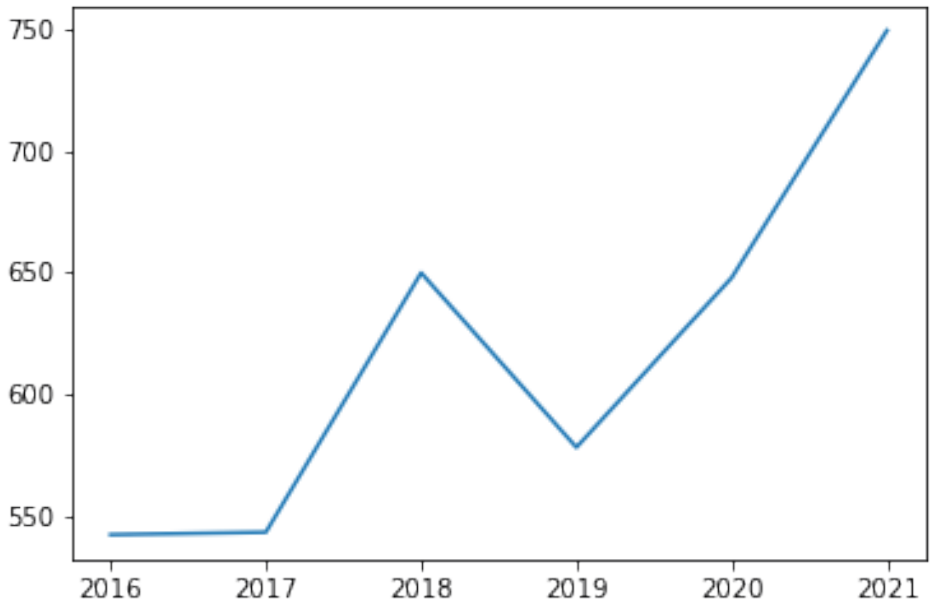



Imagen 1. Gráfico de líneas (*sparkline*) simple.

Como se puede observar, el gráfico es bastante básico, aunque lo podemos customizar aplicando una serie de modificaciones:

```
x = [2016, 2017, 2018, 2019, 2020, 2021] #años

y = [542, 543, 650, 578, 648, 750] #ventas

plt.plot(x, y, marker ='o', linestyle='--', linewidth =2, color='lightcoral')

plt.xlabel('Años')

plt.ylabel('Ventas producto 4535')

plt.title('Ventas anuales producto 4535')

plt.yticks([0, 542, 750, 800])

plt.rcParams["figure.figsize"] = (10,8)

x = [2016, 2017, 2018, 2019, 2020, 2021] #años

y = [542, 543, 650, 578, 648, 750] #ventas

plt.savefig('IMG2.png')

plt.show()
```

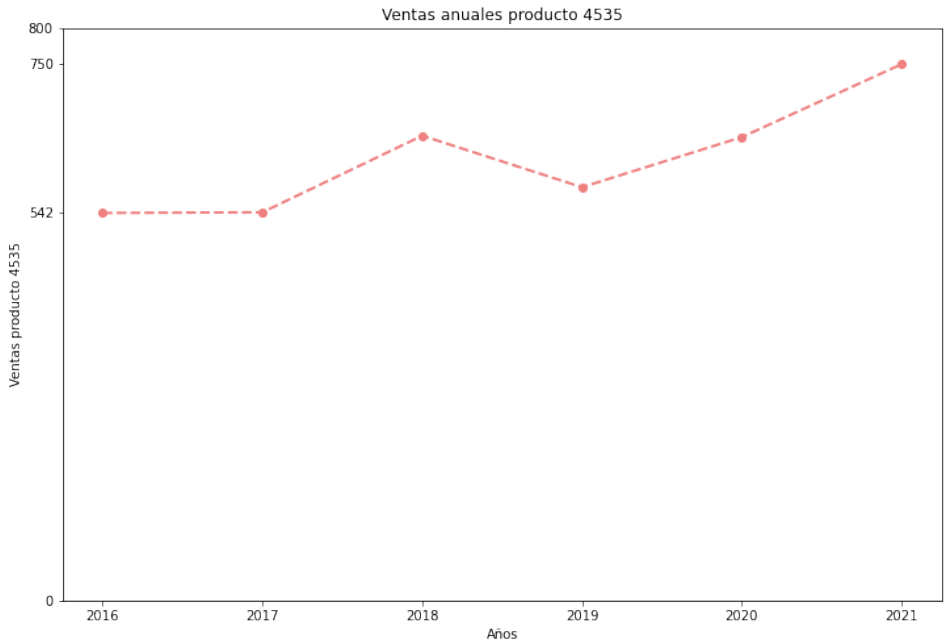



Imagen 2. Gráfico de líneas (*sparkline*).

Viendo este ejemplo que podemos desarrollar, vamos a ver las distintas partes del código que elaboran nuestro ejemplo:

- **plt.plot.** Se trata de la fórmula como elemental del gráfico. Principalmente, encontramos los siguientes elementos:
  - **Variables de entrada** (posición primera y segunda). Por ejemplo, x, y.
  - **marker.** Identifica el tipo de marcador que queremos que aparezca en nuestra representación. Estos serían los principales:

| marker | symbol | description |
|---|---|---|
| "." | • | point |
| ",” | · | pixel |
| "o" | ● | circle |
| "v" | ▼ | triangle_down |
| "^" | ▲ | triangle_up |
| "<" | ◀ | triangle_left |
| ">" | ▶ | triangle_right |
| "1" | ⊤ | tri_down |
| "2" | ⊥ | tri_up |
| "3" | ⊣ | tri_left |
| "4" | ⊢ | tri_right |
| "8" | ● | octagon |
| "s" | ■ | square |
| "p" | ⬟ | pentagon |
| "P" | ✚ | plus (filled) |
| "*" | ★ | star |
| "h" | ⬢ | hexagon1 |
| "H" | ⬣ | hexagon2 |
| "+" | + | plus |
| "x" | × | x |
| "X" | ✖ | x (filled) |
| "D" | ◆ | diamond |
| "d" | ⧫ | thin_diamond |
| "\|" | \| | vline |
| "_" | — | hline |

Imagen 3. Principales marcadores.

- **linestyle.** Identifica el tipo de línea. Estos son los principales ejemplos:

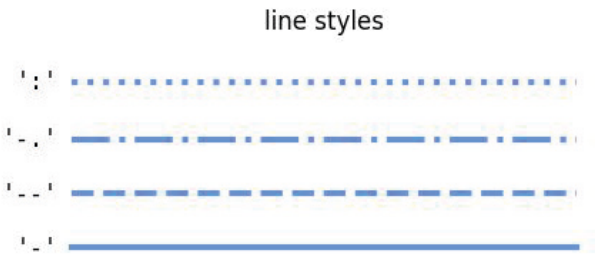



Imagen 4. Tipos de línea.

- **linewidth.** Indica el grosor de la línea mediante un campo numérico.
- **color.** Indica el color de la línea y puntos. Estos son los colores disponibles:

Imagen 5. Colores disponibles en Matplotlib.

- **plt.xlabel().** Indica el nombre del eje X.
- **plt.ylabel().** Indica el nombre del eje Y.
- **plt.title().** Indica el nombre del título.
- **plt.yticks().** Delimita el eje Y en este caso para que aparezcan los datos mencionados en una lista.
- **plt.rcParams["figure.figsize"] = (10,8).** Representa el tamaño de la imagen, en este caso, 10*6.
- **plt.savefig().** Comando para guardar la imagen.
- **plt.show().** Visualiza la imagen.

A continuación, incluyamos una segunda variable:

```
#Producto 4535

x = [2016, 2017, 2018, 2019, 2020, 2021] #años

y = [542, 543, 650, 578, 648, 750] #ventas

#Producto 4538

x2 = [2016, 2017, 2018, 2019, 2020, 2021] #años

y2 = [252, 350, 485, 526, 580, 780] #ventas

plt.plot(x, y, marker ='o', linestyle='--', linewidth =2, color='li-
ghtcoral', label='4535')

plt.plot(x2, y2, marker ='d', linestyle='-', linewidth =2, color='me-
diumorchid', label='4538')

plt.xlabel('Años')

plt.ylabel('Ventas productos')

plt.title('Ventas anuales productos')

plt.legend(loc='best')

plt.yticks([0, 542, 750, 800])

plt.rcParams["figure.figsize"] = (10,8)

plt.savefig('IMG6.png')

plt.show()
```

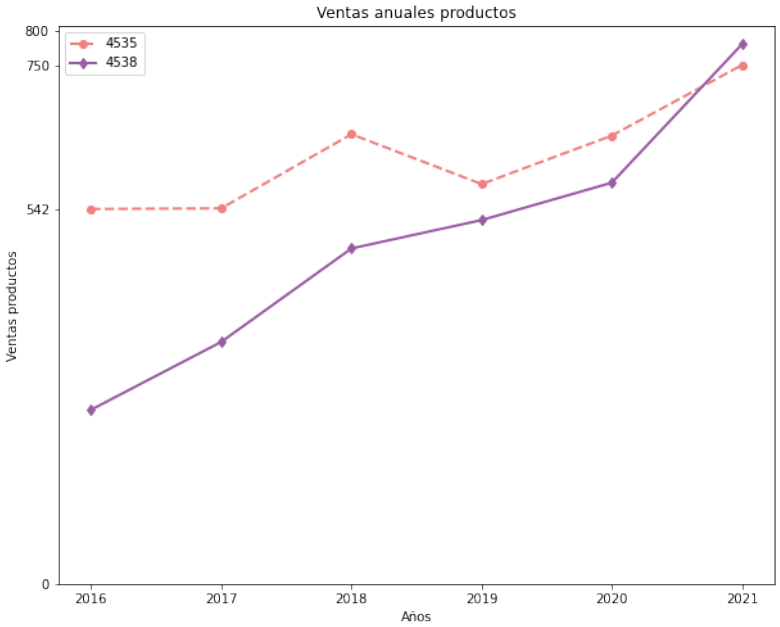



Imagen 6. Gráfico de líneas, dos variables.

Esta visualización incorpora una leyenda que indica cada una de las variables. Para ello, tenemos que hacer dos modificaciones:

- **plt.plot().** Añadir una nueva línea para la nueva línea de productos siguiendo la misma estructura comentada.
- **plt.plot().** Tenemos que añadir label en cada una de las líneas e identificarlas.
- **plt.legend(loc='best').** Por defecto estará marcado mejor; no obstante, se puede modificar su posición mediante la columna de la izquierda:

| LOCATION STRING | TRADUCCIÓN |
|---|---|
| best | Mejor |
| upper right | Arriba derecha |
| upper left | Arriba izquierda |
| lower left | Abajo izquierda |
| lower right | Abajo derecha |
| right | Derecha |
| center left | Centro izquierda |

| LOCATION STRING | TRADUCCIÓN |
|---|---|
| center right | Centro derecha |
| lower center | Abajo centro |
| upper center | Arriba centro |
| center | Centro |

Fuente: elaboración propia.

### 2.1.4. GRÁFICOS DE BARRAS

Para realizar un gráfico de barras, necesitamos un eje X y un eje Y, siendo la variable del eje X de tipo categórica. En el siguiente ejemplo veremos las ventas de unos productos:

```
x = ['2356', '2358', '2362', '6935']

y = [1250, 825, 1326, 450]

colors = ['lightcoral', 'wheat', 'paleturquoise', 'thistle']

plt.bar(x, y, color = colors , edgecolor='grey', width = 0.75)

plt.savefig('IMG7.png')

plt.show()
```

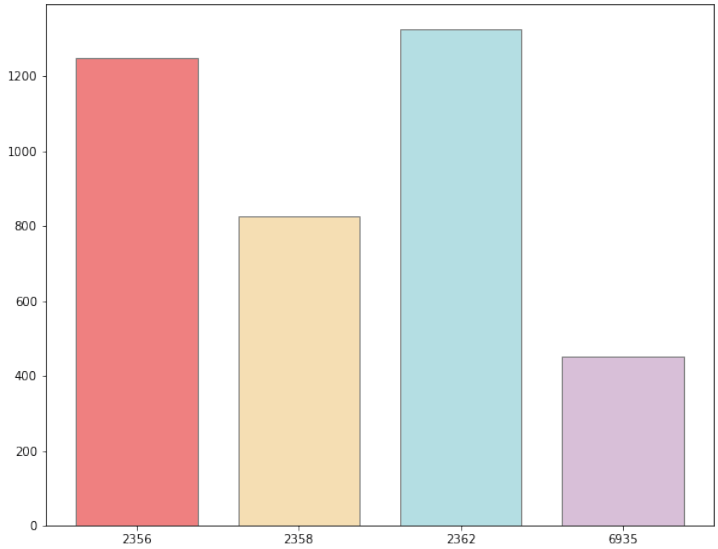



Imagen 7. Gráficos de barras.

El procedimiento es similar a la creación de un gráfico de líneas, con los siguientes cambios:

- La función principal cambia a plt.bar.
- Como son varias columnas, tendremos que elegir un color para cada uno de los tipos de productos mediante una lista. En caso de que no se pongan tantos colores como categorías, se repetirá la lista desde el principio.
- Mediante el parámetro width podremos determina el grosor de las columnas.

Si queremos realizar un gráfico de barras horizontales, podemos realizarlo fácilmente cambiando en la función principal plt.bar por plt.barh.

```
x = ['2356', '2358', '2362', '6935']

y = [1250, 825, 1326, 450]

colors = ['lightcoral', 'wheat', 'paleturquoise', 'thistle']

plt.barh(x, y, color = colors , edgecolor='grey')

plt.savefig('IMG8.png')

plt.show()
```

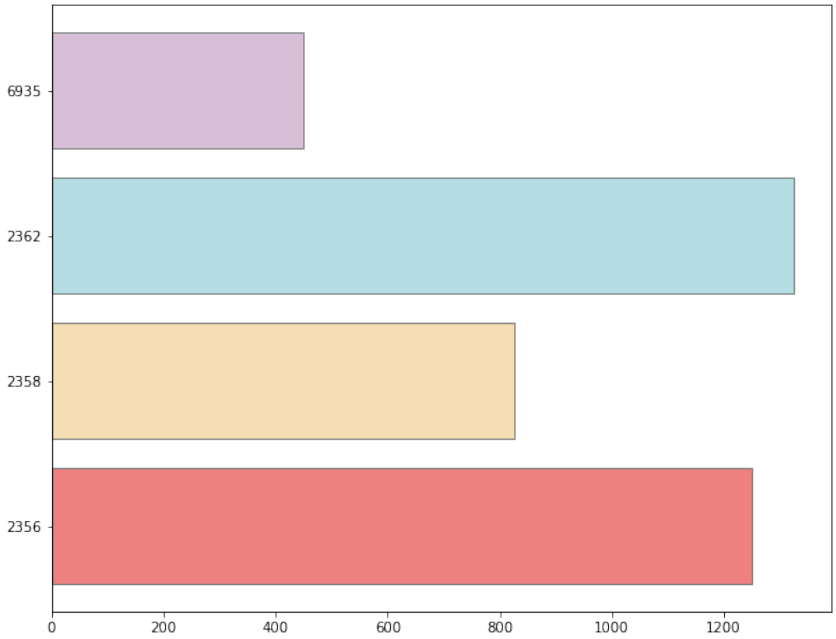



Imagen 8. Gráfico de barras horizontales.

### 2.1.5. GRÁFICOS DE TARTA

Para realizar un gráfico de tarta, necesitamos un eje X y un eje Y, siendo la variable del eje Y de tipo categórica (justo al contrario que en el gráfico de barras). En el siguiente ejemplo veremos las ventas de unos productos:

```
x = ['2356', '2358', '2362', '6935']

y = [1250, 825, 1326, 450]

colors = ['lightcoral', 'wheat', 'paleturquoise', 'thistle']

explode = (0.1, 0, 0, 0)

plt.pie(y, explode =explode, labels = x, autopct='%1.1f%%', colors=
colors, shadow=True, startangle=90)

plt.axis('equal')

plt.savefig('IMG9.png')

plt.show()
```

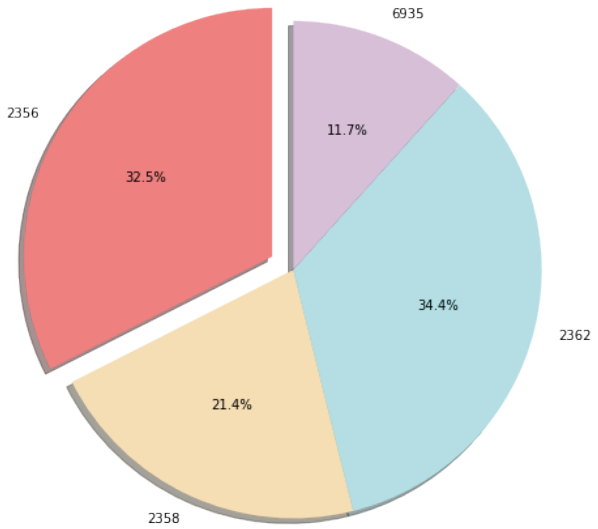



Imagen 9. Gráfico de tarta.

Los principales cambios que encontramos son los siguientes:

- La función principal cambia a plt.pie.
  - La variable categórica aparece en el argumento labels.
  - explode indica que posición de la variable categórica queremos que tenga el efecto de salida mediante la lista explode.

- autopct indicamos numéricamente la variable y.
- color cambia a colors.
- shadow da efecto sombra a la visualización
- stratangle indica la rotación que tiene el gráfico.

• plt.axis('equal') permite asegurar que el gráfico tiene forma redonda.

### 2.1.6. HISTOGRAMA

Para realizar un gráfico de histograma, necesitamos un eje X y un eje Y, siendo la variable del eje Y de tipo numérica y la variable X bins (agrupación de valores). En el siguiente ejemplo veremos las ventas en función de los rangos de importes, de tal forma que las columnas superiores indican que se ha vendido más en ese rango:

```
x = [150, 957, 125, 126, 250, 350, 450, 480, 497, 50, 75, 850, 230,
75, 25, 85, 652, 280, 754, 985, 874, 852]

bins = [200, 400, 600, 800, 1000]

plt.hist(x, bins, edgecolor ="grey", color= 'paleturquoise')

plt.savefig('IMG10.png')

plt.show()
```

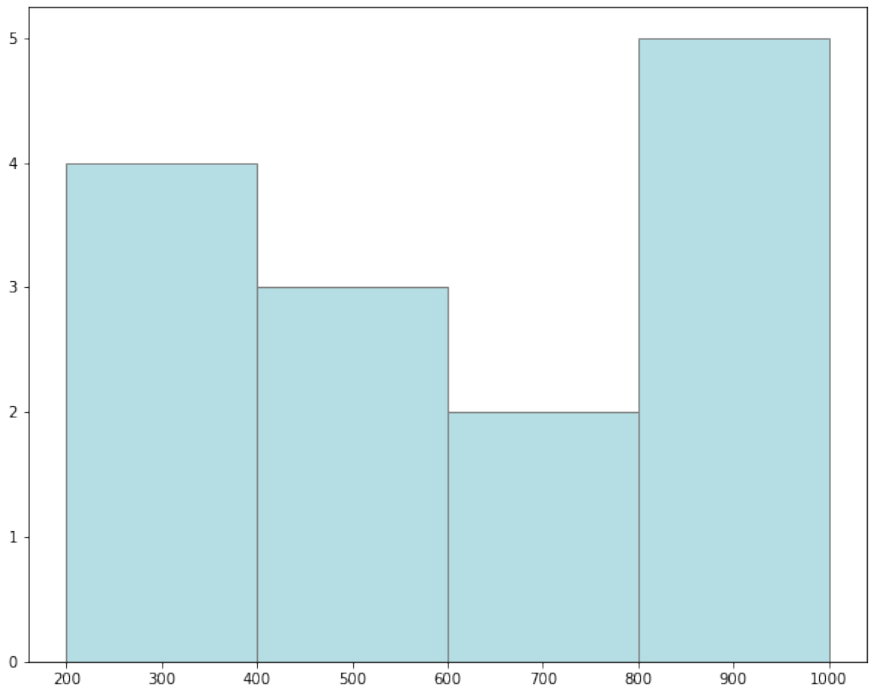



Imagen 10. Gráfico histograma.

### 2.1.7. GRÁFICO DE CAJA (*BOXPLOTS*)

Este gráfico se utiliza principalmente para detectar valores atípicos en una distribución. En nuestro caso, veremos la distribución ventas:

```
#Boxplots

x = [2500, 150, 957, 125, 126, 250, 350, 450, 480, 497, 50, 75, 850,
230, 75, 25, 85, 652, 280, 754, 985, 874, 852]


plt.boxplot(x, labels= ["Ventas España"])

plt.savefig('IMG11.png')

plt.show()
```

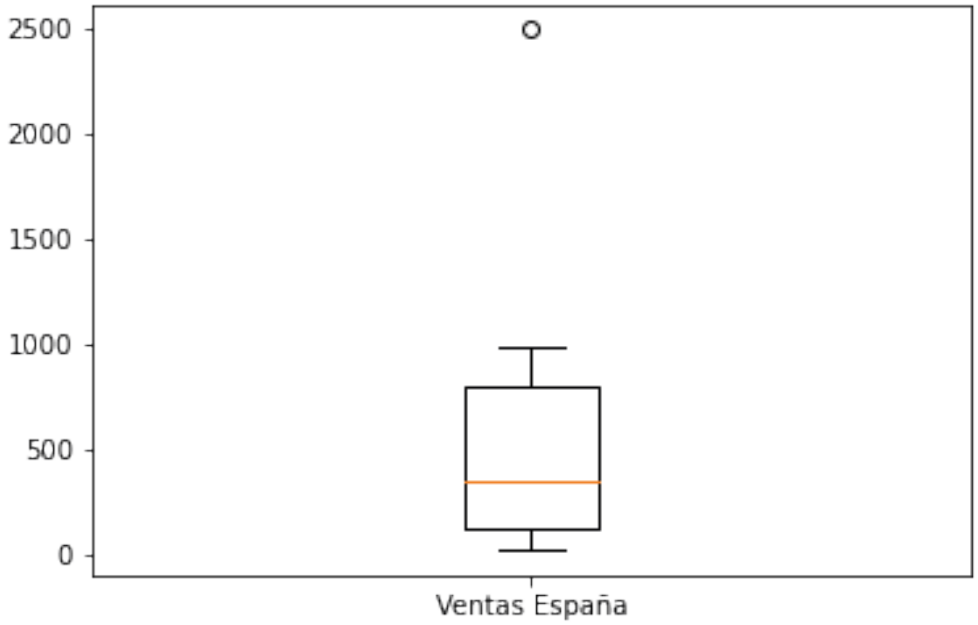



Imagen 11. Gráfico de caja (*boxplot*).

En caso de que queramos representar más de una variable, esto se haría añadiendo esa variable, y aparecería dentro de plt.boxplot entre [], formato lista:

```
#Boxplots

Ventas_España = [2500, 150, 957, 125, 126, 250, 350, 450, 480, 497,
50, 75, 850, 230, 75, 25, 85, 652, 280, 754, 985, 874, 852]

Ventas_Alemania = [2500, 1550, 9757, 1425, 1426, 1250, 3250, 4510,
2480, 1497, 150, 715, 850, 230, 75, 25, 858, 652, 280, 754, 9850, 874,
1852]

colors = ['lightcoral', 'thistle']
```

```
plt.boxplot([Ventas_España, Ventas_Alemania], labels= ["Ventas España",
"Ventas_Alemania"])

plt.savefig('IMG12.png')

plt.show()
```

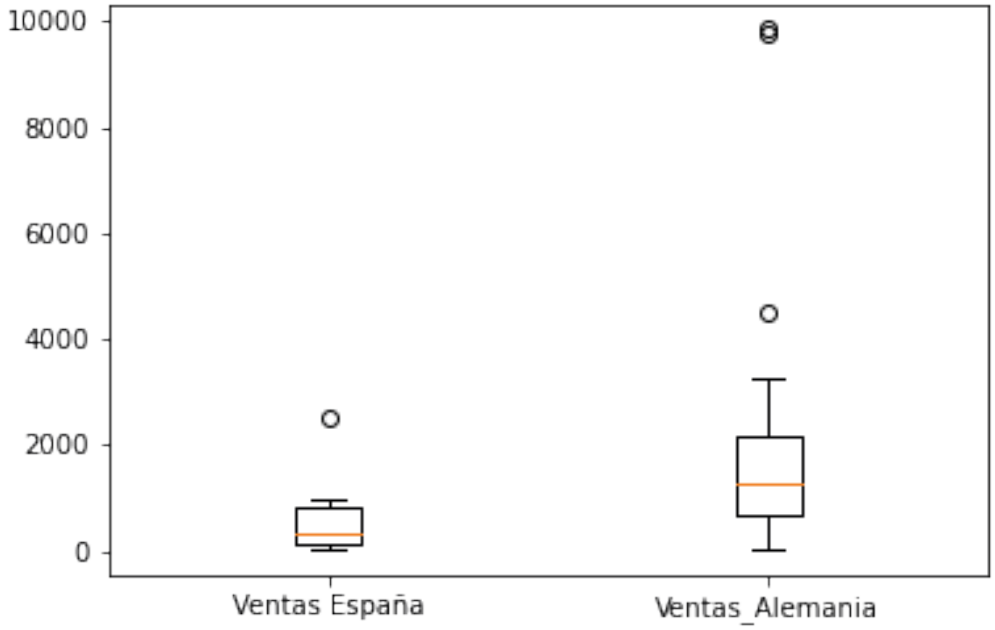



Imagen 12. Gráfico de caja (*boxplot*).

## 2.1.8. GRÁFICO DE DISPERSIÓN (*SCATTER PLOT*)

En el gráfico de dispersión observamos la relación entre dos variables, en este caso vamos a observar la relación existente entre las ventas y el beneficio por ventas.

```
Ventas_España = [2500, 150, 957, 125, 126, 250, 350, 450, 480, 497,
50, 75, 850, 230, 75, 25, 85, 652, 280, 754, 985, 874, 852]

Benefico_España =[688, 25, 550, 58, 23, 26, 85, 165, 187, 201, 1, 322,
145, 25, 2, 9, 65, 85, 63, 96, 256, 365, 202]

plt.scatter(Ventas_España, Benefico_España)

plt.xlabel('Ventas_España')

plt.ylabel('Benefico_España')

plt.title('Relación venta - beneficio España')

plt.savefig('IMG13.png')

plt.show()
```

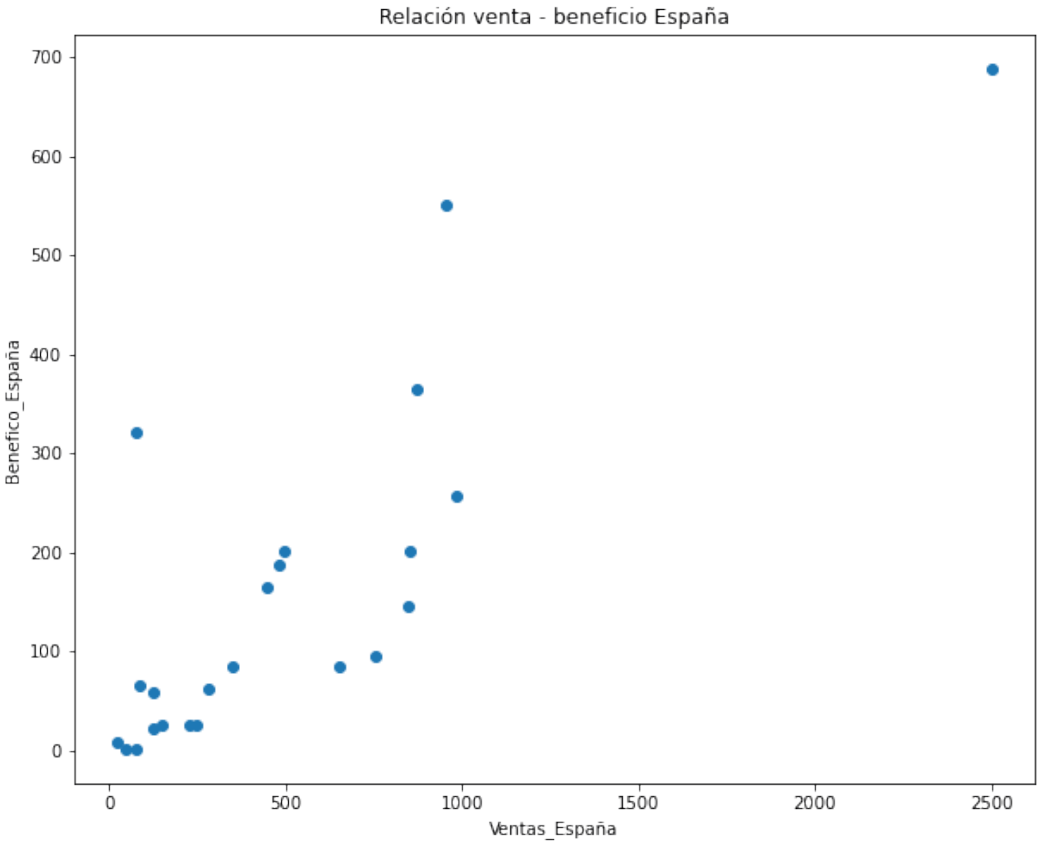



Imagen 13. Gráfico de dispersión (*scatter plot*).

## 2.1.9. SUBPLOTS

Los *subplots* son gráficos que están uno al lado del otro. Estos gráficos son especialmente útiles cuando queremos hacer comparaciones entre dos o más tipos de gráficos. En este caso, vamos a visualizar la distribución de las ventas en España mediante un gráfico de caja y la relación entre las variables ventas y beneficio.

```
#Subplots

#Subplots

x = [2500, 150, 957, 125, 126, 250, 350, 450, 480, 497, 50, 75, 850,
230, 75, 25, 85, 652, 280, 754, 985, 874, 852]

Ventas_España = [2500, 150, 957, 125, 126, 250, 350, 450, 480, 497,
50, 75, 850, 230, 75, 25, 85, 652, 280, 754, 985, 874, 852]

Benefico_España =[688, 25, 550, 58, 23, 26, 85, 165, 187, 201, 1, 322,
145, 25, 2, 9, 65, 85, 63, 96, 256, 365, 202]

fig, ax = plt.subplots(1,2, sharey=True)

ax[0].boxplot(x, labels= ["Ventas España"])
```

```
ax[1].scatter(Benefico_España, Ventas_España)
plt.xlabel('Ventas_España')
plt.ylabel('Benefico_España')
plt.title('Relación venta - beneficio España')

plt.savefig('IMG13.1.png')
plt.show()
```

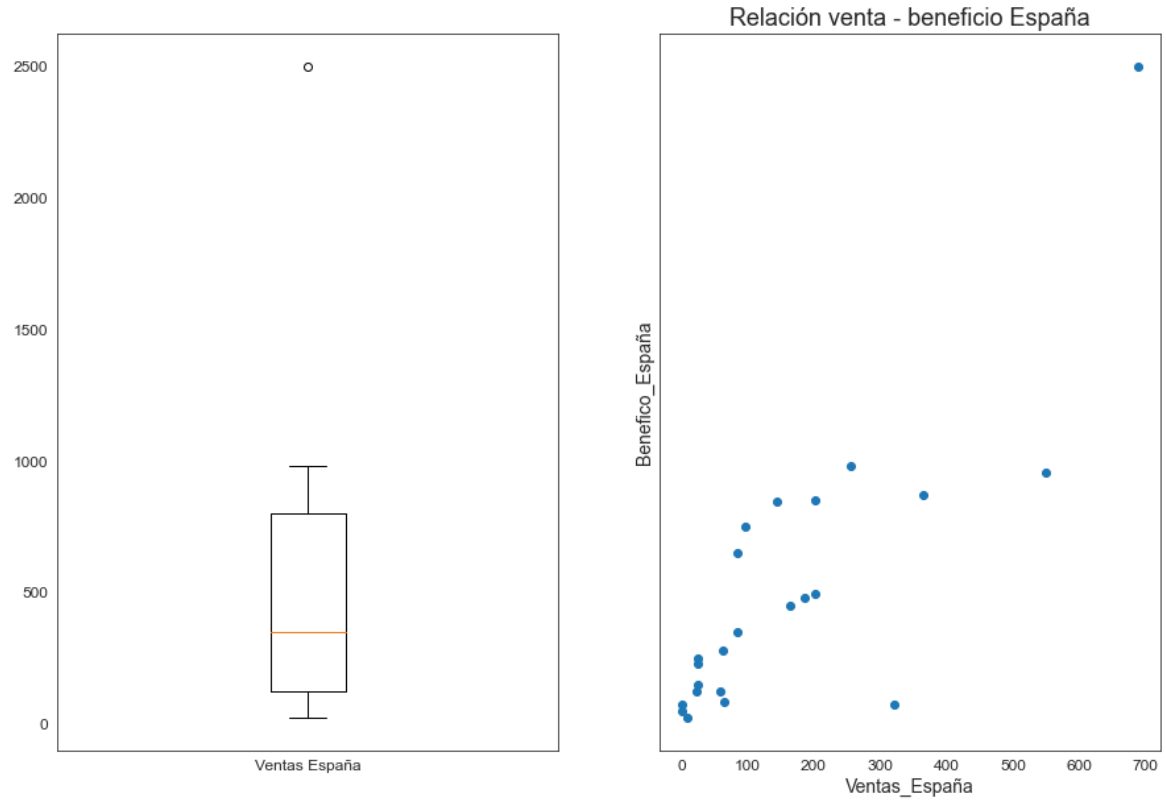



Imagen 14. Gráficos combinados.

**IMPORTANTE**

Mediante la función fig, ax = plt.subplots(), podemos determinar los siguientes parámetros:

- Primera posición: indica el número de filas que queremos visualizar.
- Segunda posición: indica el número de columnas que queremos visualizar.
- Sharey: si indicamos true ambos gráficos compartirán eje Y.

# 2.2. SEABORN

## 2.2.1. INTRODUCCIÓN A SEABORN

Seaborn es una librería de visualización de Python basada en Matplotlib integrada con las librerías Numpy, Pandas y datos estructurales. A continuación, veremos las principales diferencias con Matplotlib:

- **Funcionalidad:** mientras que Matplotlib está diseñado para elaborar gráficos simples, con Seaborn podemos profundizar más en el desarrollo de las visualizaciones, así como en su personalización con el uso de paletas preestablecidas, las cuales son combinaciones de colores preestablecidas.

- **Sintaxis:** la sintaxis de Seaborn es ligeramente más simple y fácil de aprender. A continuación, veremos la elaboración de un mismo gráfico de barras simple con las dos librerías:

  **Ejemplo:**

  – Matplolib: matplotlib.pyplot.bar(x_axis, y_axis)

  – Seaborn: seaborn.barplot(x_axis, y_axis)

- **Matrices y marcos de datos:** Matplotlib trata los datos como objetos (variables), mientras que Seaborn trata conjuntamente a todo el *dataset*, de forma que si queremos realizar la representación de dos columnas no tendríamos que aislarlas (crear variables sobre el *dataset*), ya que solo tendríamos que mencionar el *dataset* y nombrar cada eje con el nombre de la columna que queremos representar.

Para el desarrollo de los ejemplos de visualizaciones mediante Seaborn, vamos a analizar las ventas de un supermercado europeo. Para realizar las visualizaciones, previamente tenemos que agrupar los datos en función del objetivo de nuestra visualización.

## 2.2.2. INSTALACIÓN DE SEABORN

```
!pip install seaborn
```

```
import seaborn as sns
```

Para que todos los gráficos tengan la misma estructura, vamos a incluir los siguientes parámetros:

```
sns.set_style('white') # Opciones disponibles: darkgrid, white grid,
dark, white and ticks
```

```
plt.rc('axes', titlesize=18) # Fuente título eje
```

```
plt.rc('axes', labelsize=14) # Fuente etiquetas ejes x e y
```

```
plt.rc('xtick', labelsize=12) # Tamaño de fuente los puntos sobre las
etiquetas X.
```

```
plt.rc('ytick', labelsize=12) # Tamaño de fuente los puntos sobre las etiquetas Y.

plt.rc('legend', fontsize=12) # Tamaño leyenda

plt.rc('font', size=12) # Control por defecto del tamaño del texto
```

### 2.2.3. GRÁFICO DE BARRAS

En este ejemplo, visualizaremos las ventas medias de cada una de las compras por país. Para ello, previamente realizaremos la siguiente agrupación:

```
# Preprocessing

barplot = data.groupby(['Country/Region'], as_index=False).mean().round(2)[['Country/Region', 'Sales']]

#Gráfico de barras

plt.figure(figsize=(20,8), tight_layout=True)

ax = sns.barplot(x=barplot['Country/Region'], y=barplot['Sales'], palette='pastel')

ax.set(title='Venta media por país', xlabel='Country/Region', ylabel='Avg. Ventas')

for i in ax.containers:

    ax.bar_label(i,)

plt.savefig('IMG14.png')

plt.show()
```

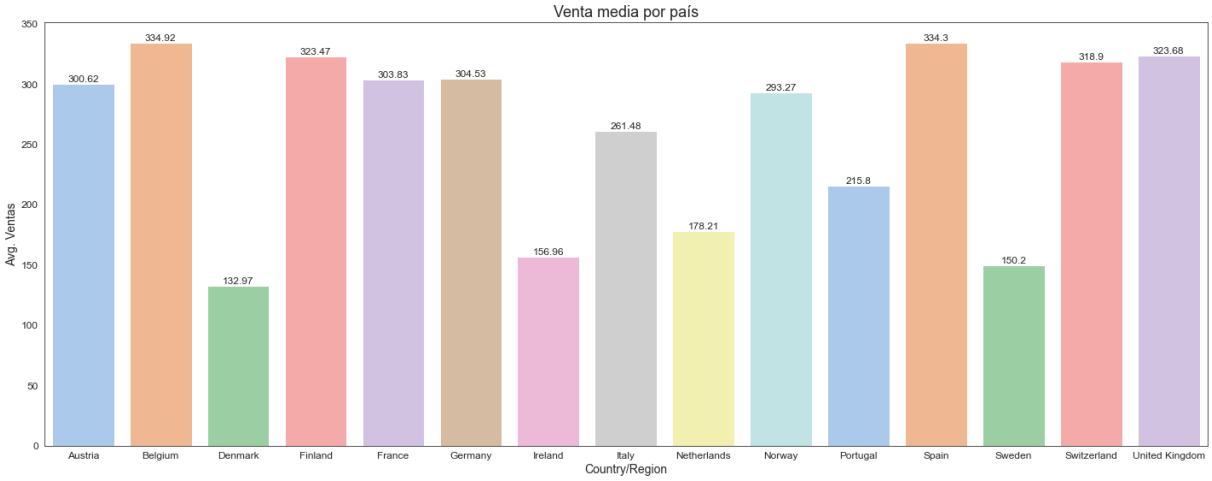



Imagen 15. Gráfico de barras.

Su estructura está formada por:

- plt.figure(). Determina el formato de la visualización, se compone por:
  - figsize: determina el tamaño de la visualización.
  - tight_layout: en el caso de que tengamos más de un gráfico, si marcamos True, ajusta automáticamente los gráficos entre sí.
- ax. Es la función principal para realizar la representación. Está formada por:
  - sns.barplot: mediante barplot determinamos que es un gráfico de barras, se compone por:
    - » x: eje X, para ello nombramos la variable dentro del *dataset*.
    - » y: eje Y, para ello nombramos la variable dentro del *dataset*.
    - » pallete: determinamos la paleta de colores disponibles.

**+ INFO**

Puedes consultar este *link* para obtener más información al respecto: https://seaborn.pydata.org/tutorial/color_palettes.html#palette-tutorial

- ax.set(). Determinamos la información recogida en título (title), eje X (xlabel) y eje Y (ylabel).

**IMPORTANTE**

Para visualizar en la parte superior del gráfico los valores de cada columna, utilizamos la siguiente función:

```
for i in ax.containers:
        ax.bar_label(i,)
```

## 2.2.4. GRÁFICO DE BARRAS AGRUPADAS

En este caso, vamos a visualizar el numero de registros por segmento de venta y país. Para ello, previamente realizamos la siguiente agrupación:

```
# Preprocessing

barplot = data[data['Segment'].isin(['Consumer', 'Corporate', 'Home
Office'])]

barplot = barplot.groupby(['Country/Region', 'Segment'], as_index=False).
count()
```

```
# Gráfico de barras agrupadas

plt.figure(figsize=(20, 8), tight_layout=True)

ax = sns.barplot(x=barplot['Country/Region'], y=barplot['Order ID'],
hue=barplot['Segment'], palette='pastel')

ax.set(title='N° de registros por segmento/ país' ,xlabel='País', ylabel='Count')

ax.legend(title='Segmento', title_fontsize='13', loc='upper right')

plt.savefig('IMG15.png')

plt.show()
```

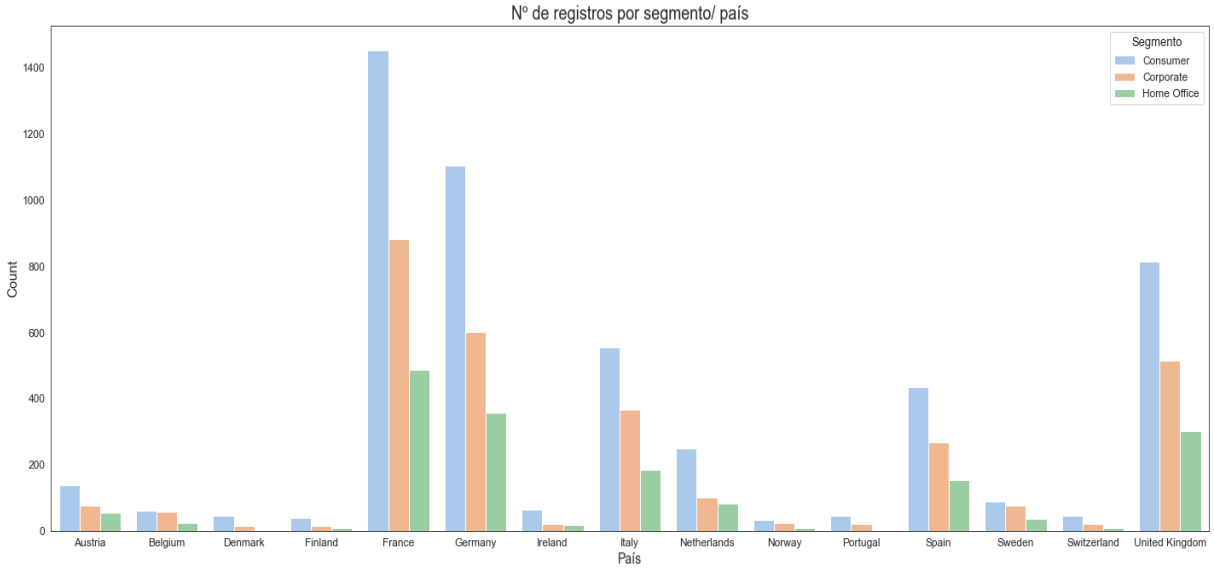



Imagen 16. Gráfico de barras apiladas.

A diferencia del gráfico anterior, dentro del apartadado ax= sns.barplot incluimos la variable hue, que nos permite categorizar nuestro gráfico.

**+ INFO**

En algunas versiones nos encontramos con el siguiente error:

Type of error: module 'seaborn' has no attribute 'histplot'

Su solución es:

1. !pip install seaborn --upgrade
2. Reiniciar el *notebook* mediante Kernel/Restart

## 2.2.5. GRÁFICO HISTOGRAMA

En el siguiente ejemplo veremos cómo está distribuida la cantidad de ventas por venta.

```
# Gráfico histograma

ax = sns.histplot(data=data, x='Quantity', bins=7, color=sns.color_
palette('Set2')[2], linewidth=2)

ax.set(title='Histogram', xlabel='Cantidad', ylabel='Count')

plt.show()
```

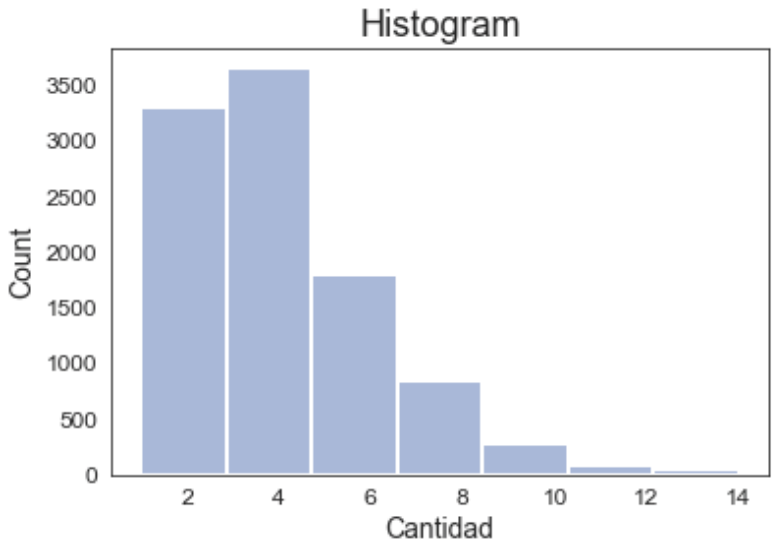



Imagen 17. Gráfico histograma.

En este tipo de gráfico es importante determinar el número de partes en las que queremos separar el eje X, esto lo tenemos que indicar en ax= sns.histplot(bins=).

## 2.2.6. GRÁFICO DE CAJA (*BOXPLOTS*)

En el siguiente ejemplo vamos a ver cómo están distribuidas la cantidad de ventas en cada compra por países respecto a sus valores estadísticos (valor mínimo, cuartil 1, cuartil 2, cuartil 3, valores máximos y valores extremos).

```
#Gráfico de caja (boxplots)

plt.figure(figsize=(18,8), tight_layout=True)

ax=sns.boxplot(data=data,x='Country/Region',y='Quantity',palette='Set2',
linewidth=2.5)

ax.set(title='Boxplot', xlabel='', ylabel='Quantity')

plt.savefig('IMG17.png')

plt.show()
```

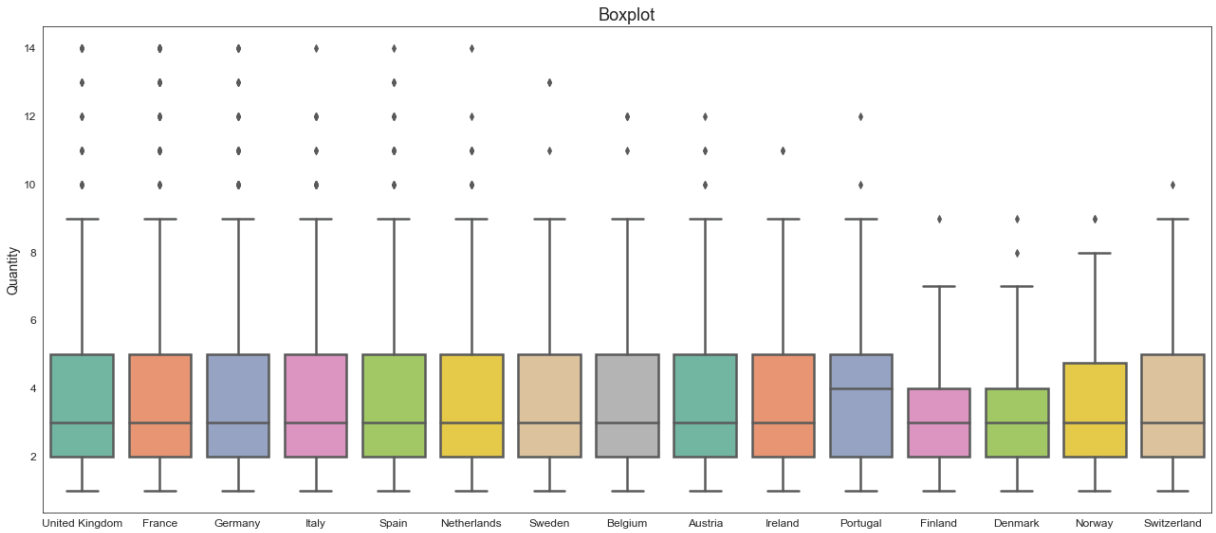



Imagen 18. Gráfico de caja (*boxplot*).

En este ejemplo vemos otra forma de utilizar la función ax=sns, ya que, utilizando el parámetro data, luego podemos utilizar los parámetros x e y para los ejes X e Y.

## 2.2.7. GRÁFICO DE DISPERSIÓN

Mediante este gráfico de dispersión podemos ver la relación existente entre dos variables y una tercera. En este caso vamos a analizar la relación entre ventas y beneficio por país, que se determina por el color del punto.

```
#Gráfico de dispersión

plt.figure(figsize=(20,8), tight_layout=True)

ax = (Seaborn, 2022)(data=data, x='Sales', y='Profit', hue='Country/
Region', palette='Set2', s=160)

ax.set(xlabel='Sales', ylabel='Profit')

ax.legend(title='Country/Region', title_fontsize = 12)

plt.savefig('IMG18.png')

plt.show()
```

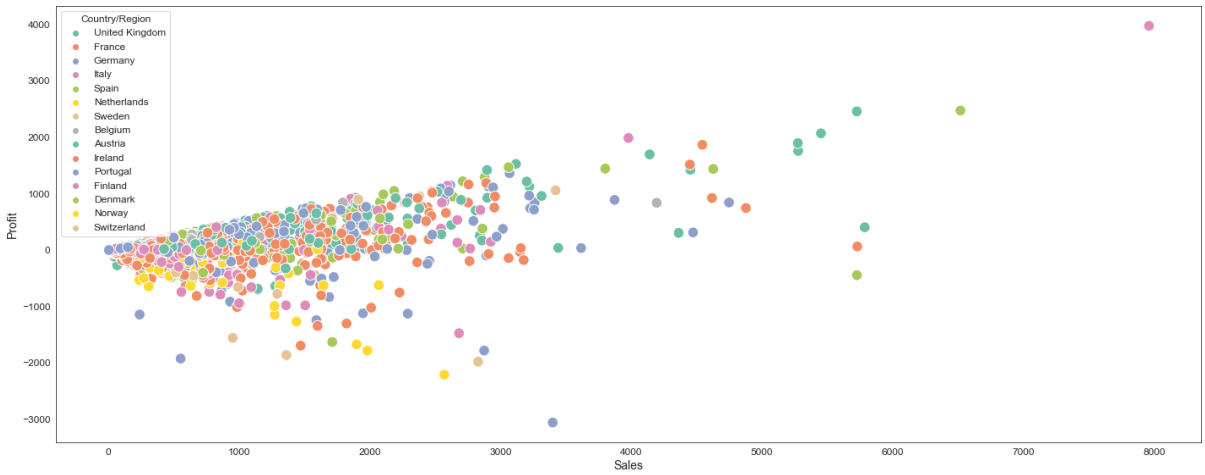



Imagen 19. Gráfico de dispersión.

**¿SABÍAS QUE...?**

Como particularidad con respecto al resto de visualizaciones, dentro de la fórmula ax= sns.scatterplot() mediante el parámetro s podemos determinar el tamaño del punto.

## 2.2.8. GRÁFICO DE TARTA (*PIE CHART*)

En este gráfico de tarta vamos a representar el porcentaje de ventas por productor (solo los diez principales productores). Para ello, previamente tenemos que realizar las siguientes transformaciones en nuestro *dataset*:

```
piechart = data[data['Country/Region']=='Spain']
```

```
piechart = piechart.sort_values('Sales', ascending=False)[['Manufacturer',
'Sales']]
```

```
piechart = piechart[:11]
```

Seaborn como tal no dispone de un tipo de gráfico puramente de tarta. No obstante, está basado en Matplotlib, por lo que podemos elaborarlo de la misma forma que en Seaborn, pero incorporando la paleta de colores propia de Seaborn.

```
# Gráfico de tarta (Pie chart)
```

```
colors = sns.color_palette('pastel')
```

```
plt.figure(figsize=(12, 8), tight_layout=True)
```

```
explode_list = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0.2]
```

```
plt.pie(piechart['Sales'], labels=piechart['Manufacturer'], autopct='%.0f
%%', explode=explode_list, pctdistance=.7,     colors=colors, shadow=True)
```

```
plt.title('Ventas por productor', weight='bold')
plt.savefig('IMG19.png')
plt.show()
```

Imagen 20. Gráfico de tarta.

## 2.2.9. *SUBPLOTS*

Los *subplots* son los gráficos que están uno al lado del otro. Estos son especialmente útiles cuando queremos realizar comparaciones entre dos o más tipos de gráficos. En este caso, vamos a visualizar la distribución de las ventas por los diez primeros proveedores para los puntos de venta en España y Francia con un gráfico de tarta. Para ello, previamente tenemos que realizar las siguientes modificaciones en nuestro *dataset*:

```
# Preprocessing
piechart = data[data['Country/Region']=='Spain']
piechart = piechart.sort_values('Sales', ascending=False)[['Manufacturer',
'Sales']]
piechart = piechart[:11]
piechart2 = data[data['Country/Region']=='France']
piechart2 = piechart2.sort_values('Sales', ascending=False)[['Manufacturer',
'Sales']]
piechart2 = piechart2[:11]
```

```
#subplots
fig, ax = plt.subplots(1,2, figsize=(12, 5), tight_layout=True)

colors = sns.color_palette('pastel')

explode_list = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0.2]

ax[0].pie(piechart['Sales'], labels=piechart['Manufacturer'], autopct='%.0f
%%', explode=explode_list, pctdistance=.7, colors=colors, shadow=True)

ax[0].set_title('España', weight='bold')

explode_list1 = [0.1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]

ax[1].pie(piechart2['Sales'], labels=piechart2['Manufacturer'], autopct='%.0f
%%', explode=explode_list1, pctdistance=.7, colors=colors, shadow=True)

ax[1].set_title('Francia', weight='bold')

plt.savefig('IMG20.png')

plt.show()
```

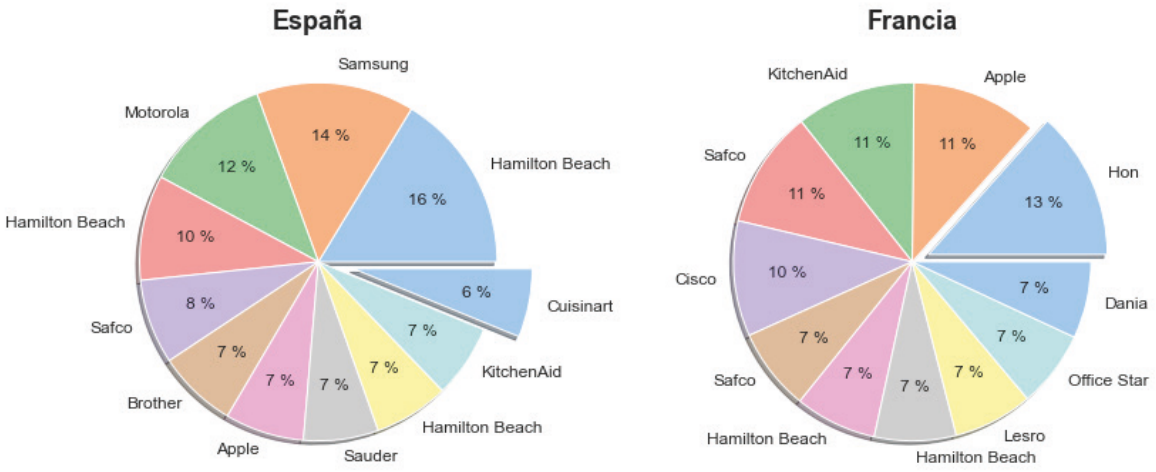



Imagen 21. *Subplots*.

## 2.2.10. GRÁFICO DE LÍNEAS

Mediante esta visualización, veremos las ventas por país desde 2017 a 2020. Para ello, previamente realizaremos las siguientes modificaciones en nuestro *dataset*:

```
# Preposessing

barplot = data.groupby(['Country/Region', 'year'], as_index=False).
mean().round(2)[['year','Country/Region', 'Sales']]

barplot = barplot.sort_values(by=['year', 'Country/Region'])

# Gráfico de líneas (line chart)

plt.figure(figsize=(20, 8))
```

```
palette = sns.color_palette("pastel", 15)

plt.xticks([2017, 2018, 2019, 2020])

ax = sns.lineplot(data=barplot, x= 'year', y= 'Sales' ,linewidth=2.5,
hue= "Country/Region", style='Country/Region',

     markers=True, dashes=False, palette = palette)

ax.set(xlabel='Year', ylabel='Sales', title='Rating troughtout the
years')

ax.legend(title='Players', title_fontsize = 8, loc='upper right')

plt.savefig('IMG21.png')

plt.show()
```

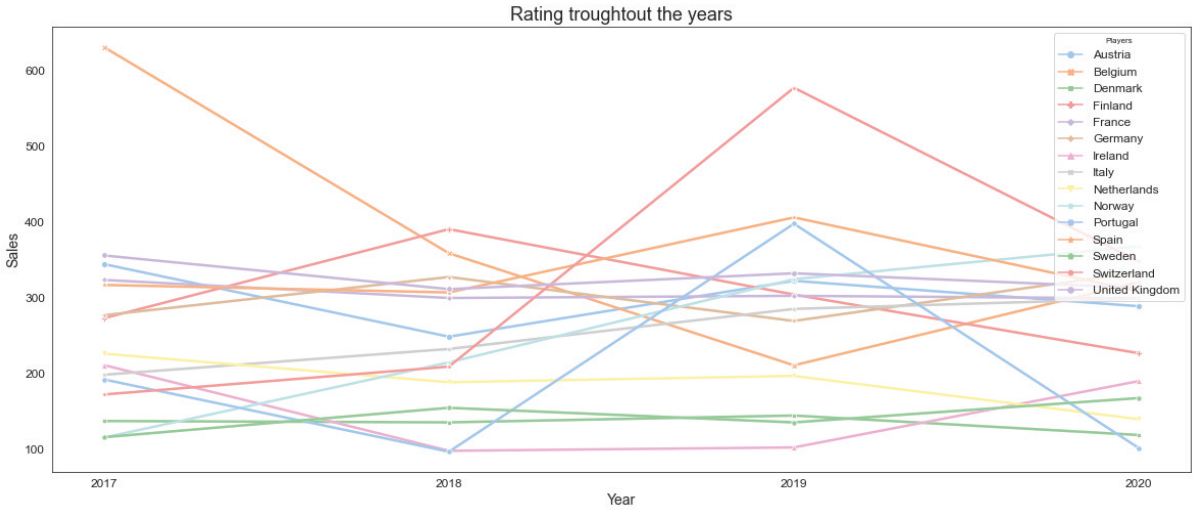



Imagen 22. Gráfico de líneas (*line chart*).

# 2.3. PLOTLY

## 2.3.1. INTRODUCCIÓN A PLOTLY

Plotly es una librería con base en JavaScript que permite crear visuales interactivas. El hecho que Plotly sea una herramienta con base en JavaScript hace que su uso sea algo más complejo que Matplotlib o Seaborn. No obstante, vamos a utilizar el comando iplot, que mediante la librería cufflinks hace que Plotly sea interactivo con nuestro *dataset* basado en Pandas.

## 2.3.2. INSTALACIÓN DE LIBRERÍAS

```
#Instalación de librerías

!pip install cufflinks
```

```
#Importamos librerias

import pandas as pd

import cufflinks as cf

from IPython.display import display, HTML

cf.set_config_file(sharing='public',theme='white',offline=True)
```

En caso de que queramos ver otros temas (background del gráfico) disponibles, podemos ver las opciones mediante:

```
#Temas disponibles

cf.getThemes()
```

**+ INFO**

En el caso que queramos que la visualización de nuestras visualizaciones sea privada tenemos que cambiar el parámetro de sharing a private.

El *dataset* que vamos a analizar en este ejemplo hace referencia a la tasa de mortalidad de los países del sur de la Unión Europea (España, Francia, Portugal e Italia). Para ello, hemos analizado el *dataset* que proporciona el Centro Europeo para la Prevención y el Control de las Enfermedades (ECDC, 2022).

En este *dataset* vienen dados todos los países de la UE con información dializada. En nuestro caso, tan solo nos interesan los países del sur de Europa (España, Francia, Portugal e Italia), con su información agrupada por meses. De tal forma, procedemos a realizar los siguientes cambios:

```
#Carga de dataset

data = pd.read_excel(r'C:\Users\sfernan\Downloads\PROJECTS\
Redacción\E-Learning\Visualización de datos\Data\data.xlsx')

#Cambio a formato fecha y creación de variable periodo.

data['dateRep'] = pd.to_datetime(data['dateRep'],format="%d/%m/%Y")

data['Periodo']= data['dateRep'].dt.strftime('%Y%m')

#Cambio de nombre a determinadas columnas

data= data.rename(columns={'countriesAndTerritories':'country',
'continentExp': 'continent'})

#Filtrado de países

data = data[data['country'].isin(['Spain', 'France', 'Portugal',
'Italy'])]
```

```
#Creación del nuevo dataset agrupado por Periodo y país

data = data.groupby(['Periodo', 'country'], as_index=False).
agg({'deaths':'sum', 'popData2020':'mean'})

#Creación de la variable de estudio, tasa de mortalidad per cápita

data['Mortalidad'] = data['deaths']/data['popData2020']

#Limpieza de dataset

data = data.drop(columns=['deaths','popData2020'])

data = data.sort_values(by='Periodo')

#Elminar valores na

data= data.dropna()

#Pivotizar nuestro dataset

data= data.pivot(index='Periodo', columns='country', values='Morta-
lidad')
```

### 2.3.3. GRÁFICO DE LÍNEAS (*LINE CHART*)

En esta visualización, representaremos cómo ha fluctuado la tasa de mortalidad desde el inicio de la pandemia a la actualidad.

```
# Gráfico de líneas

data.iplot(kind='line', xTitle='Año', yTitle= 'CasosPerCápita'

 , title = 'Casos Per Cápita Países sur Europa')
```

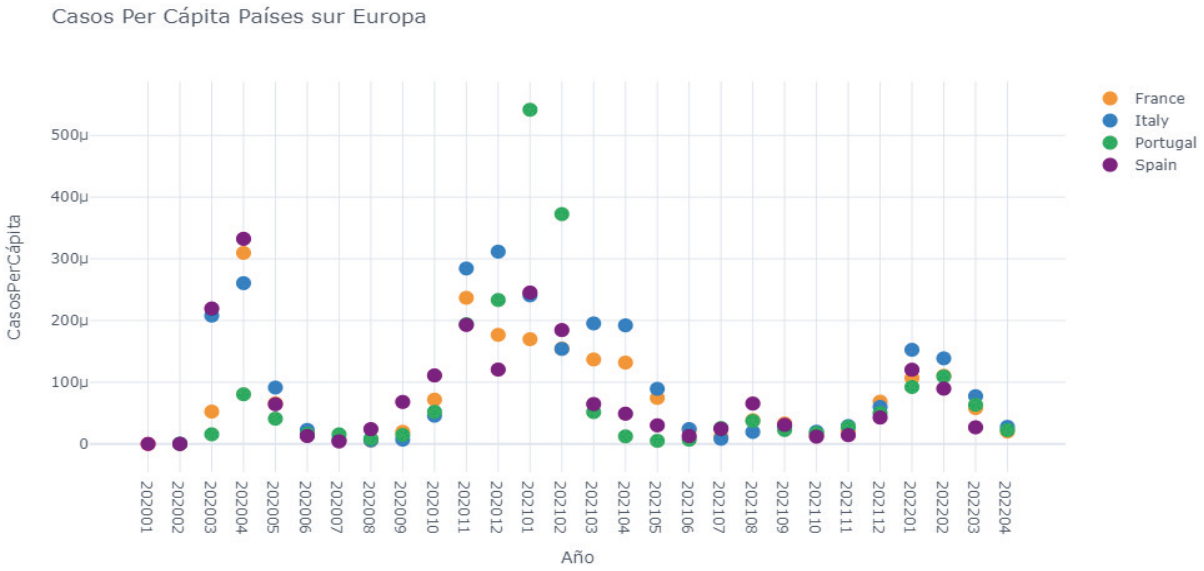



Imagen 23. Gráfico de líneas (*line chart*).

La creación de este gráfico está compuesta por:

- data: indicamos nuestro *dataset*.
- iplot(): es la función con la que visualizamos el gráfico, está formada por:
  - kind(): tipo de gráfico.
  - xTitle: título eje X.
  - yTitle: título eje Y.
  - title: título principal.

Como se puede ver, su creación es bastante simple: la parte más compleja es la creación de *dataset* que sea acorde al gráfico que vamos a representar.

**¿SABÍAS QUE...?**

Si hacemos clic en los nombres de las leyendas, vamos quitando las variables de la representación y, si volvemos a hacer clic, volverá a aparecer. Por otro lado, a diferencia del resto de librerías, esta permite que al pasar el cursor por el gráfico podamos ver los valores que lo conforman.

### 2.3.4. GRÁFICO DE BARRAS (*BAR CHART*)

En esta visualización vamos a representar cómo ha impactado la tasa de mortalidad a los países seleccionados en el periodo 03/2020 (inicio de la pandemia). Para ello, previamente tenemos que realizar los siguientes cambios en el *dataset*:

```
# Preposessing

data_202003= data[data.index.isin(['202003'])]

data_202003= data_202003.T

# Gráfico de barras

data_202003.iplot(kind='bar', xTitle='Países', yTitle= 'CasosPerCápita',
title = 'Casos Per Cápita Países sur Europa para 202003', color='green')
```

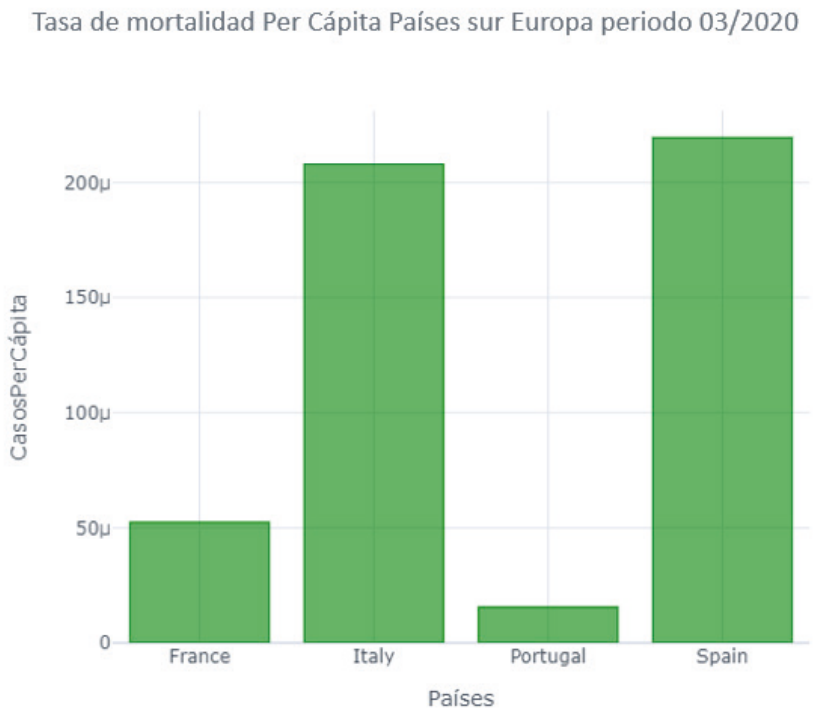



Imagen 24. Gráfico de barras (*bar chart*).

Para la realización de este gráfico, tenemos que modificar el parámetro kind a bar.

## 2.3.5. GRÁFICO DE BARRAS AGRUPADAS (*BAR CHART*)

En esta visualización vamos a representar como ha impactado la tasa de mortalidad a los países seleccionados en unos periodos determinados. Para ello, previamente tenemos que realizar los siguientes cambios en el *dataset*:

```
#Multiples graficos de barra

data_quarter=  data[data.index.isin(['202003','202006','202009','2020
012','202103','202106','202109','202112','202203'])]

fig=data_quarter.iplot(kind='bar',xTitle='Países',yTitle='CasosPerCápita',
title = 'Casos Per Cápita Países sur Europa por Trimestres')
```

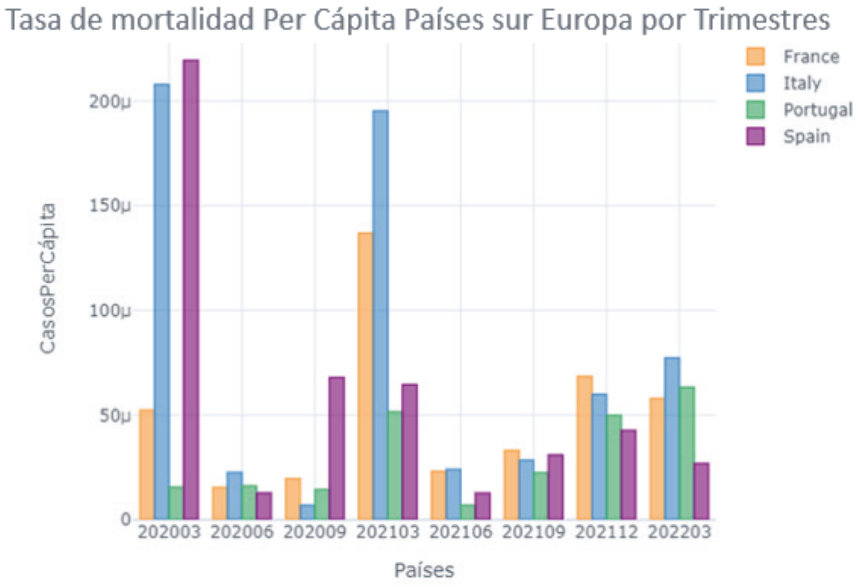



Imagen 25. Gráfico de barras agrupadas.

Para la realización de este gráfico, tenemos que modificar el parámetro kind a bar.

## 2.3.6. GRÁFICO DE CAJA (*BOXPLOTS*)

Este gráfico se utiliza principalmente para detectar valores atípicos en una distribución. En nuestro caso, vamos a ver la distribución de la tasa de mortalidad:

```
#Boxplot
data['Spain'].iplot(kind='box', xTitle='', yTitle= 'CasosPerCápita',
title = 'Tasa de moralidad Per Cápita España')
```

Imagen 26. Gráfico de caja (boxplot).

Para la realización de este gráfico, tenemos que modificar el parámetro kind a box.

**IMPORTANTE**

En caso de que queramos representar más de una variable, solo tenemos que visualizar un *dataset* completo.

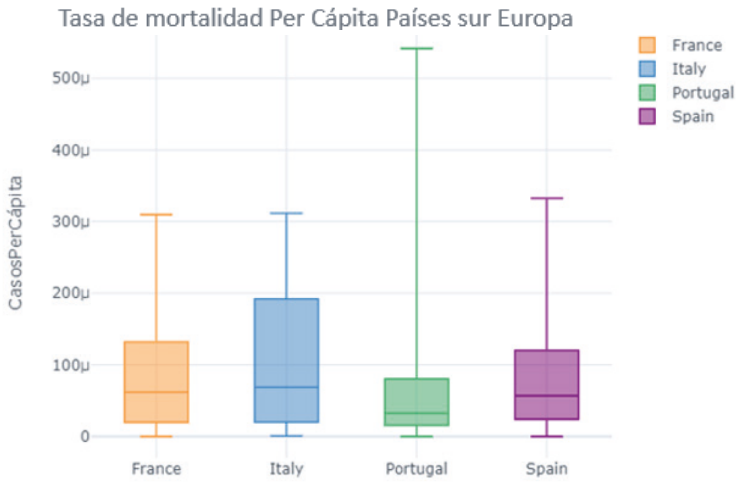



Imagen 27. Gráfico de caja (*boxplot*).

## 2.3.7. HISTOGRAMA

Para realizar un gráfico de histograma, necesariamente necesitamos un eje X y un eje Y, siendo la variable del eje Y de tipo numérica y la variable X bins (agrupación de valores). En este caso, Plotly automáticamente nos equidistribuye la variable bins.

```
#Histogram

data['Spain'].iplot(kind='hist', xTitle='Tasa Mortalidad', yTitle=
'Casos', title = 'Tasa de moralidad Per Cápita Países sur Europa')
```

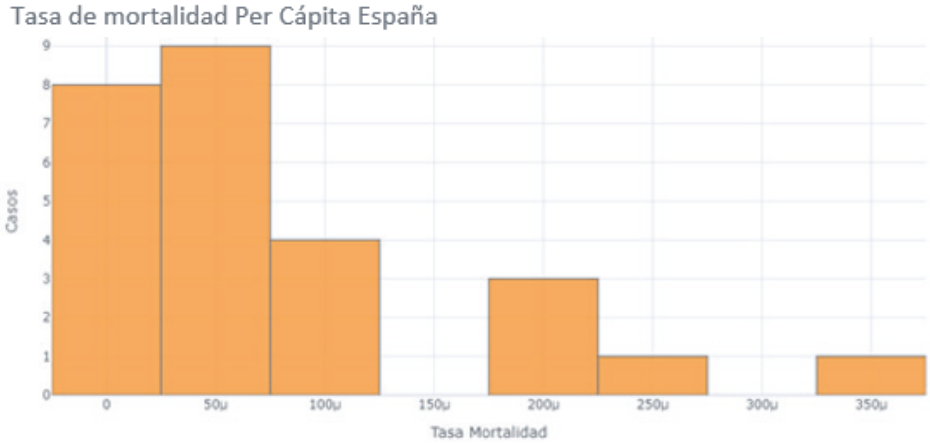



Imagen 28. Histograma.

Para la realización de este gráfico, tenemos que modificar el parámetro kind a hist.

### 2.3.8. GRÁFICO DE TARTA (*PIE CHART*)

En este gráfico de tarta, vamos a ver representar la tasa de mortalidad de un periodo concreto (202003):

```
#Piechart

data_202003= data_202003.reset_index()

data_202003.iplot(kind='pie',   labels='country',   values='202003',
title = 'Tasa de moralidad Per Cápita Países sur Europa')
```

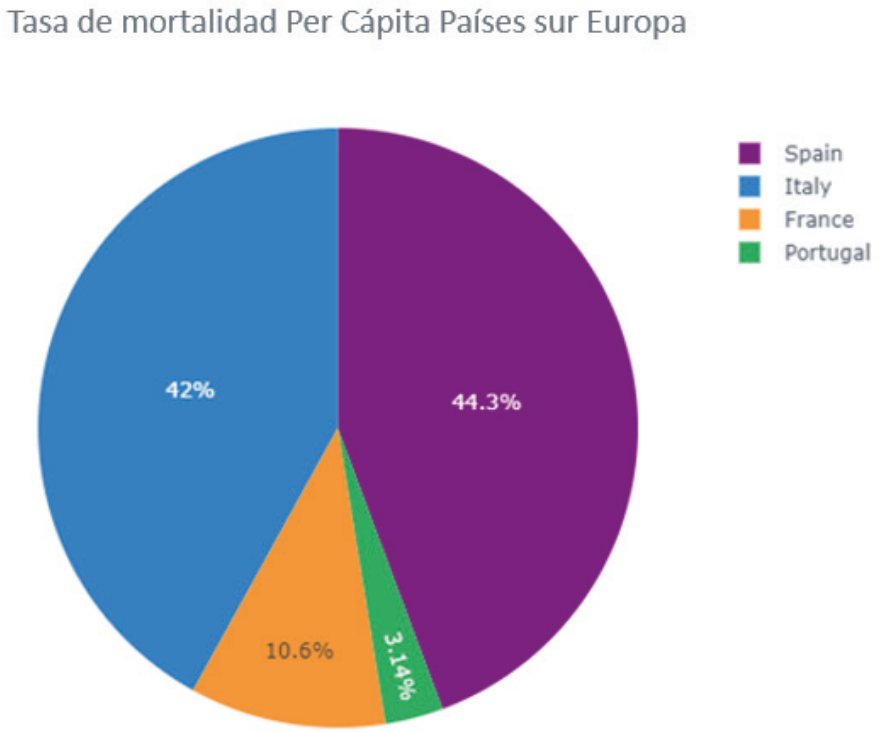



Imagen 29. Gráfico de tarta (*pie chart*).

Para la realización de este gráfico, tenemos que modificar el parámetro kind a pie.

### 2.3.9. GRÁFICO DE ANILLOS

El desarrollo y objetivo es el mismo que en el gráfico de tarta, la única diferencia es visual ya que este tipo de gráfico es más limpio a la hora de su representación gráfica al no contener nada en su interior.

```
#Gráfico de anillos

data_202003= data_202003.reset_index()

data_202003.iplot(kind='pie',   labels='country',   values='202003',
hole=.8, title = 'Tasa de moralidad Per Cápita Países sur Europa')
```

Su desarrollo es igual al del gráfico de tarta, pero incluyendo el parámetro hole. Por otro lado, podemos controlar el grosor de la parte blanca central mediante el parámetro hole.

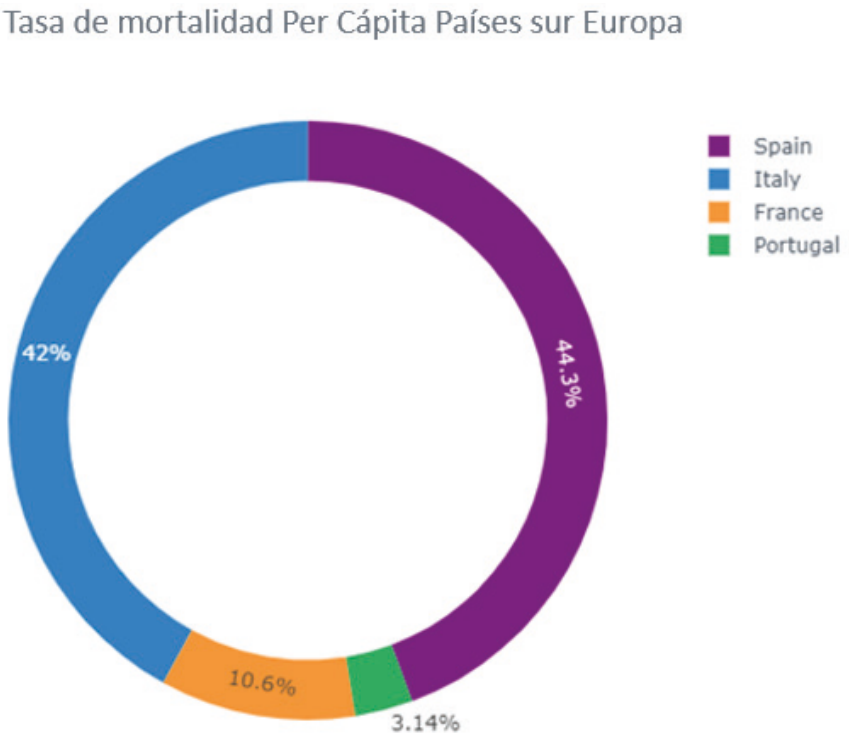



Imagen 30. Gráfico de anillos.

Para la realización de este gráfico, tenemos que modificar el parámetro kind a pie.

## 2.3.10. GRÁFICO DE DISPERSIÓN

Mediante el gráfico de dispersión podemos ver la relación existente entre dos variables y una tercera. En este caso, vamos a analizar la relación entre tiempo y tasa de mortalidad por país, siendo esta determinada por el color del punto.

```
#Gráfico de dispersión

scatter=data.iplot(kind='scatter', mode= 'markers', xTitle='Año',
yTitle= 'CasosPerCápita', title = 'Tasa de moralidad Per Cápita Países
sur Europa')
```

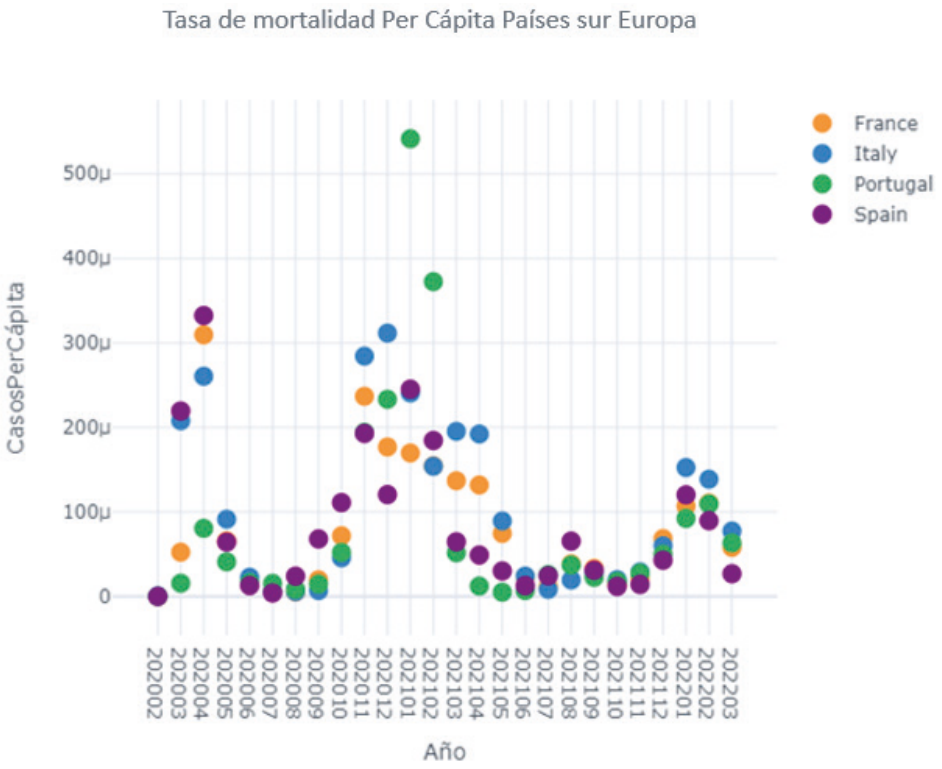



Imagen 31. Gráfico de dispersión (*scatter plot*).

Para la realización de este gráfico, tenemos que modificar el parámetro kind a scatter.

# IDEAS CLAVE

- Si vamos a realizar una representación gráfica donde no queremos depurar el diseño, pero y no queremos invertir demasiado tiempo en adaptar nuestro *dataset* al gráfico, Matplotlib es la mejor opción.
- Si queremos invertir más tiempo en depurar nuestra imagen, por los diversos elementos que lo componen, Seaborn es la mejor opción.
- Si queremos realizar visualizaciones interactivas, Plotly es la mejor opción.

# GLOSARIO

- ***Dashboard.*** Es un conjunto de visualizaciones gráficas y KPI (*key performance indicators*) que nos ayudan a entender una serie de datos. Estos elementos pueden ser interactivos entre sí.

- ***Dataframe.*** Estructura de datos con dos dimensiones en la cual se puede guardar distintos tipos de datos en columnas.

- ***Dataset.*** Conjunto de datos habitualmente tabulados, donde cada columna de la tabla representa una variable y cada fila representa un miembro determinado del conjunto de datos.

- **Numpy.** Librería de Python especializada en el cálculo numérico y el análisis de datos, con un rendimiento eficiente y un gran volumen de datos.

- **Pandas.** Librería de Python especializada en el manejo y análisis de estructuras de datos.

# BIBLIOGRAFÍA

ECDC. (s. f.). *COVID-19 situation update for the EU/EEA*. www.ecdc.europa.eu. Retrieved from: https://www.ecdc.europa.eu/en/publications-data/data-daily-new-cases-covid-19-eu-eea-country

Matplotlib. (s. f.). matplotlib.pyplot.legend. https://matplotlib.org/stable/api/_as_gen/matplotlib.pyplot.legend.html

Matplotlib. (s. f.). matplotlib.pyplot.plot. https://matplotlib.org/stable/api/_as_gen/matplotlib.pyplot.plot.html

Ploty. (s. f.). Jupyter Notebook Tutorial in Python. https://plotly.com/python/ipython-notebook-tutorial/

Seaborn. (s. f.). Seaborn: statistical data visualization. https://seaborn.pydata.org/

# TEMA 3
## Introducción a CARTO

# OBJETIVOS

- Conocer el uso de la herramienta de análisis de datos geográficos CARTO.
- Descubrir los principales tipos de datos que procesa CARTO.
- Crear mapas basado en análisis georreferenciados, así como elementos gráficos.
- Desarrollar un caso práctico.

# 3.1. INTRODUCCIÓN A CARTO

## 3.1.1. ¿QUÉ ES CARTO?

CARTO es una plataforma web de análisis de datos georreferenciados. Esta plataforma permite a las compañías realizar informes espaciales para crear rutas de reparto más eficientes, análisis del comportamiento del consumidor, analizar la localización de un nuevo emplazamiento, un lanzamiento de un nuevo producto…

Dependiendo del uso de CARTO, este se distribuye en varios apartados:

- ***Workspace:*** entorno para conectar nuestro modelo a diferentes bases de datos (públicas y propios).
- ***Builder:*** será nuestro lienzo de trabajo donde desarrollaremos nuestros *dashboard* a partir de mapas interactivos. Es el entorno donde integramos las bases de datos que hemos registrado en el entorno "*Workspace*".
- ***Platform APIs and libraries:*** es el apartado para los desarrolladores, este apartado permite la conectividad con todas las librerías de CARTO.
- ***CARTOframes:*** es una librería de Python que nos permite integrar CARTO *maps* en nuestros *workflows*.
- ***Data observatory:*** es el catálogo de bases de datos geoespaciales, esto nos permite añadir además de nuestras bases de datos (públicas y propios) el catálogo de datos (públicas y premium).

Una vez hemos creado nuestra visualización podemos publicar estos informes en una página web, o compartir el *link* del *dashboard*. En este curso nos centraremos en el uso de los primeros apartados.

## 3.1.2. PRIMEROS PASOS

Es posible acceder a CARTO mediante su web: https://carto.com/

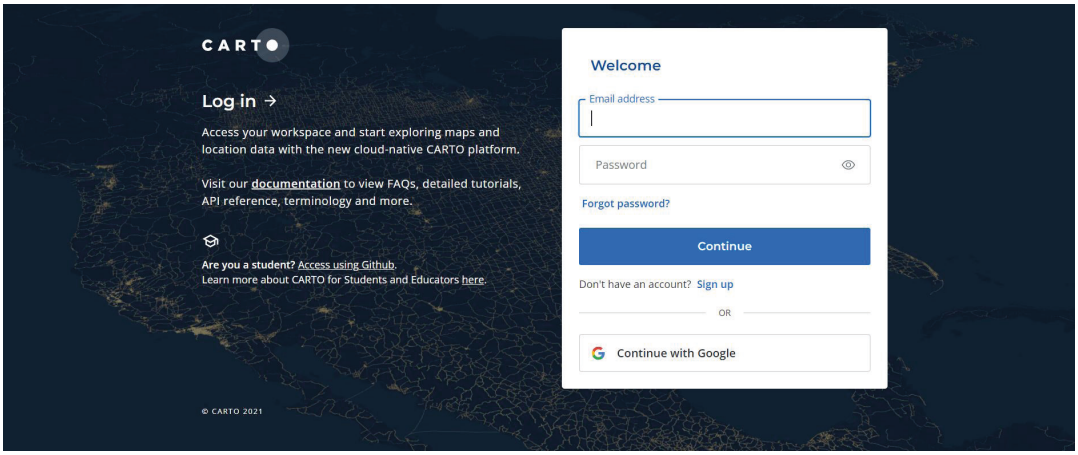



Imagen 1. Creación de cuenta.

El registro, se puede realizar de forma fácil mediante tu cuenta de Google.

Nos solicitará: el nombre de la compañía, información personal (nombre, apellidos, país, móvil) e información general sobre su compañía (número de empleados, sector).

### 3.1.3. CREA TU PRIMER MAPA

Desde el menú principal, en el panel izquierdo, haz clic en *maps*, posteriormente haz clic en "*Create your first map*".

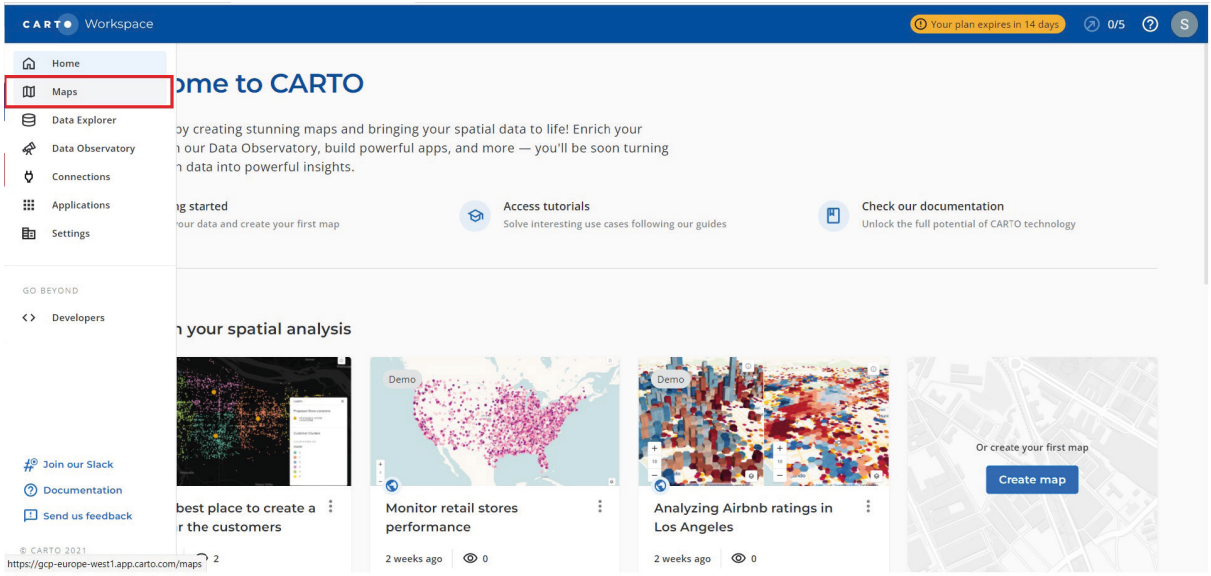



Imagen 2. Menú principal.

Desde el menú de "*Builder*" haz clic en "Add source from …":

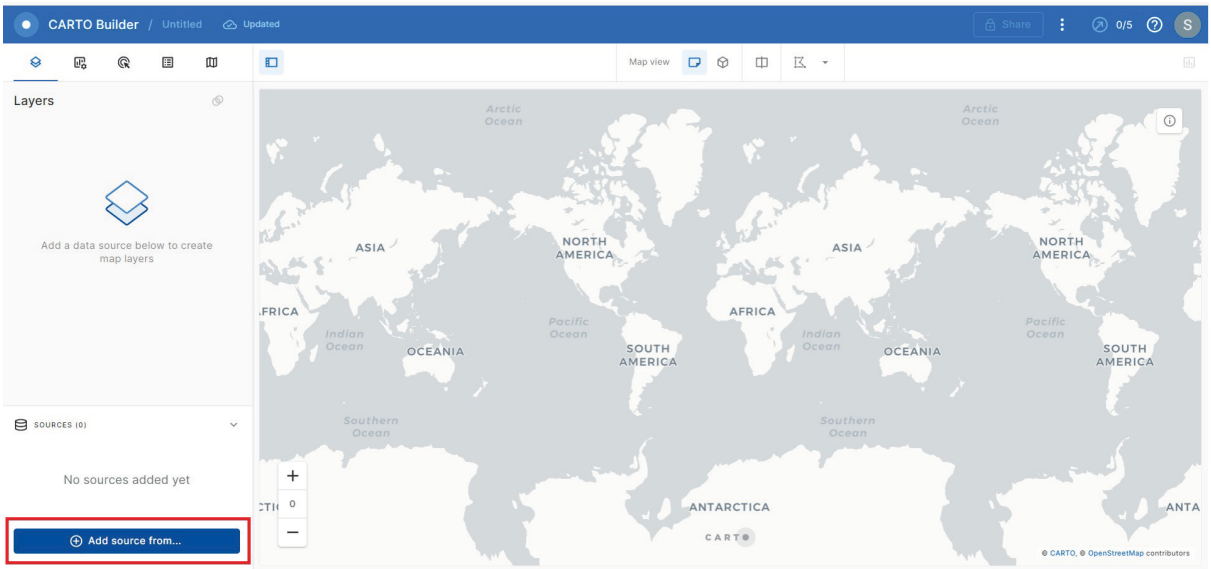



Imagen 3. Menú Builder.

Selecciona "*Custom query (SQL)*" y "*Type your own query*".

En este caso, como prueba, vamos a crear un centroide respecto a unos puntos geométricos, de manera que hacemos clic en "*Create Centroids of Geometries*" y posteriormente clic en "*Add Source*".

En este apartado podemos seleccionar bases de datos propias que hemos subido previamente, tal y como veremos en un ejemplo práctico.

### 3.1.4. EXPLORA TU *DATASET*

Una vez hemos seleccionado nuestro *dataset*, vamos a analizar los datos, desde el menú de "*builder*" ahora nos encontramos con los siguientes campos:

1. **Panel de navegación:** desde este panel podemos navegar por las siguientes ventanas:

   a. ***Layers:*** donde podemos visualizar y personalizar las capas que hemos cargado en nuestro mapa.

      » Personalización de una capa: hacemos clic en la capa que deseamos personalizar, desde "*Fill color*" hacemos clic en el botón de "*More options*" para que aparezca el apartado de "*Color base on*", donde seleccionamos la medida que queremos visualizar, por ejemplo, población por metros cuadrados. Si hacemos clic en "*Palette*" podemos elegir el color que queremos aplicar.

   b. ***Widgets:*** los *widgets* son herramientas visuales que nos permiten crear gráficos interactivos, de tal forma podemos insertar un KPI numérico, un histograma, gráfico de barras, histograma, gráfico de líneas, tablas.

**+ INFO**

En función del tipo de *dataset* que vamos a analizar, nos aparecerán unos tipos de gráficos disponibles, para desplazarnos de uno a otro haremos clic sobre uno de ellos y luego con las teclas de navegación de nuestro teclado ( <-- o -->).

   c. ***Interactions:*** desde este apartado determinamos los elementos que vamos a visualizar en nuestro mapa mediante un *Tooltip* de un punto concreto.

**+ INFO**

Si habilitamos la casilla "*Coordinates*", en el *Tooltip* nos aparecerá, la latitud y longitud de cada punto seleccionado.

   d. ***Legend:*** desde este apartado podemos renombrar el nombre de las legendas. A su vez, podemos seleccionar si queremos o no mostrar las leyendas en el mapa. Esto nos permitiría visualizar o no las capas desde el logo del ojo.

e. ***Base map:*** podemos seleccionar y personalizar las siguientes bases de mapas:

» CARTO: basado en *openstreetmap*, solo dispone de un tipo de mapa.

» Google Maps: basado en el mapa de Google Maps, dispone de varios tipos de mapas (carretera, básico, oscuro, carreteras, satélite y de terreno).

» Amazon location: podemos cargar nuestro mapa creado en Amazon location de Amazon Web Services (AWS).

2. **Capas:** desde este apartado podemos duplicar, renombrar o ocultar las distintas capas.

3. **Bases de datos:** muestra las bases de datos disponibles.

4. **Mapa:** muestra la información que hemos seleccionado en las capas.

5. **SQL editor:** desde aquí podemos seleccionar y modificar las bases de datos que posteriormente representaremos como una capa en el mapa.

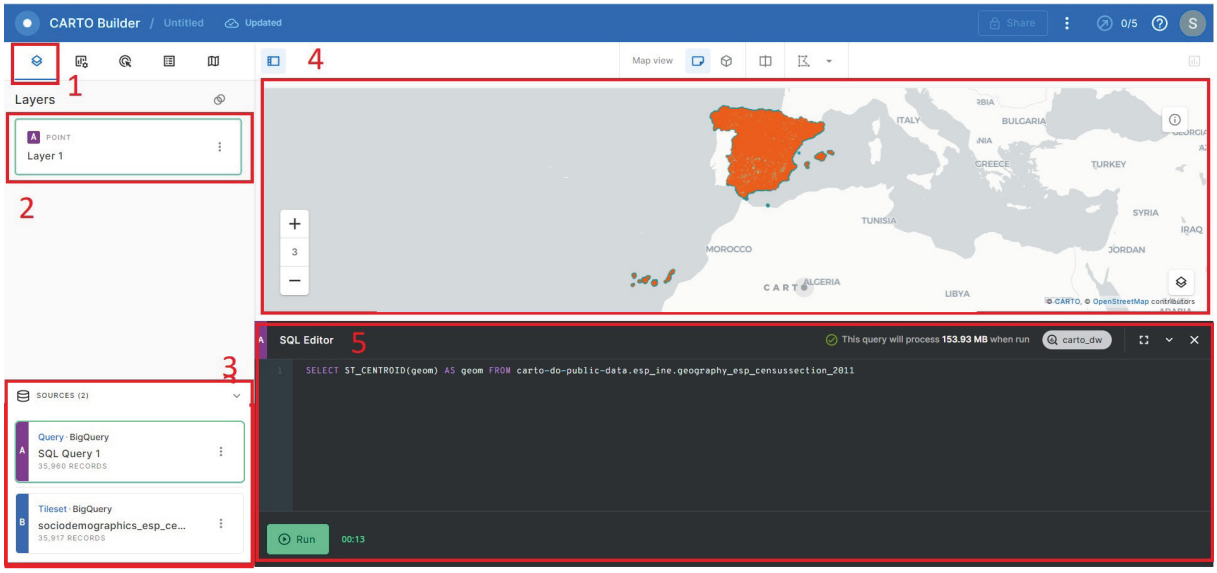



Imagen 4. Menú *builder.*

# 3.2. TIPOS DE FUENTES DE INFORMACIÓN Y TIPOS DE GRÁFICOS

## 3.2.1. TIPOS DE FUENTES DE INFORMACIÓN

CARTO nos permite crear una tabla geoespacial en su propia base de datos (CARTO *Data Warehouse*), en Google *Cloud* (*BigQuerry*) o *Snowflake* importando nuestros archivos de forma local (desde nuestro PC) o vía URL. Una vez hemos realizado la conexión, el resultado lo podemos visualizar en una tabla (*Data Explorer*), posteriormente utilizar "*Builder*" y aplicaciones externas de CARTO para realizar los mapas.

Los tipos de archivos soportados son: *CSV, KML, TAB, GeoJSON, GeoPackge* y *Shapefiles* (en formato ZIP). Estos archivos no pueden superar los 512 MB, en caso de ser necesario, pueden contactar directamente con el equipo de CARTO en este email: support@CARTO.com.

Para los archivos *CSV*, CARTO detectará automáticamente las columnas de tipo Geometría o las creará desde las columnas latitud y longitud. Estas columnas pueden tener los siguientes nombres:

- Para geometría: *geo, Geo, geometry, the_geom, wkt, wkb.*
- Latitud: *latitude, lat, Latitude.*
- Longitud: *longitude, lon, Lon, Longitude, lng, Lng.*

Los campos de delimitación admitidos son: coma (,) , semi-coma (; ), o tabulación.

Desde CARTO *Builder*, en el apartado "*Layers*" hacemos clic en "*Add source from…*", donde desde la pestaña "*Import file taba*" seleccionamos el archivo que queremos subir o el *link*.

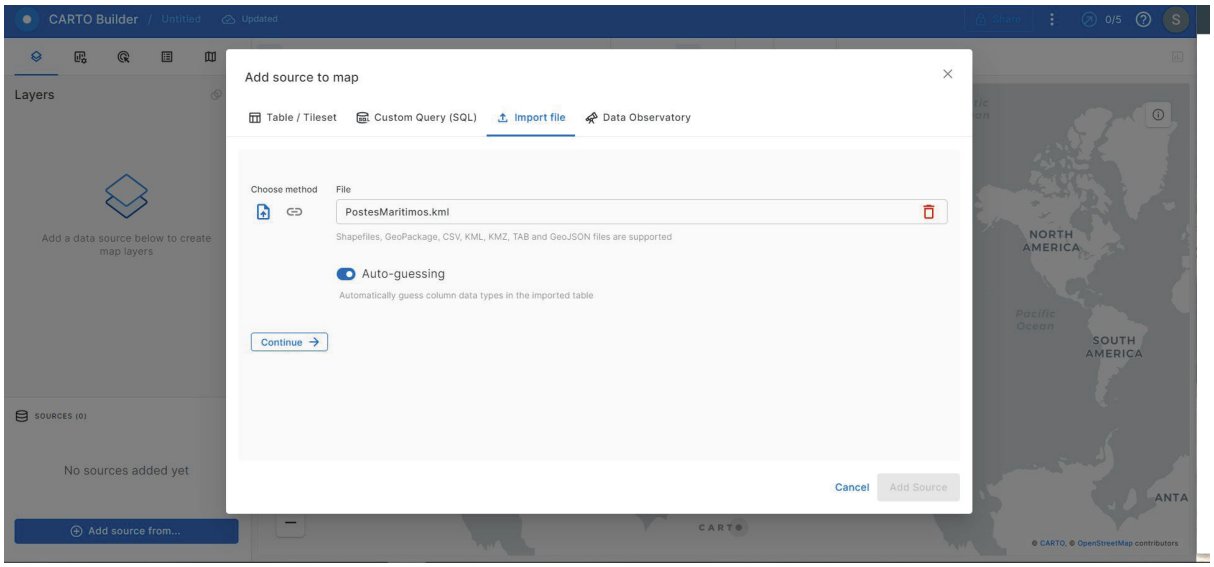



Imagen 5. Subida de archivo.

Tras este paso, hacemos clic en "*Continue*", en la siguiente ventana tenemos que seleccionar donde almacenamos nuestro archivo, en este caso haremos clic en "*organization data*" y posteriormente "*shared_us*". Tras esto hacemos clic en "*Continue*".

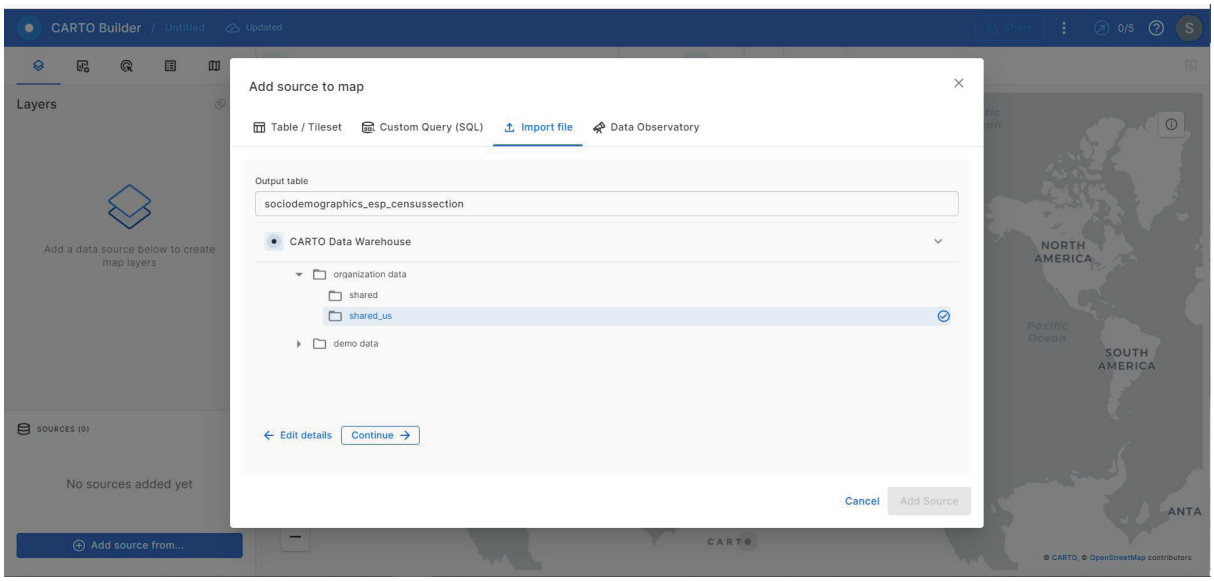



Imagen 6. Almacenamiento BBDD.

En esta última ventana veremos el resultado de nuestro proceso de importación.

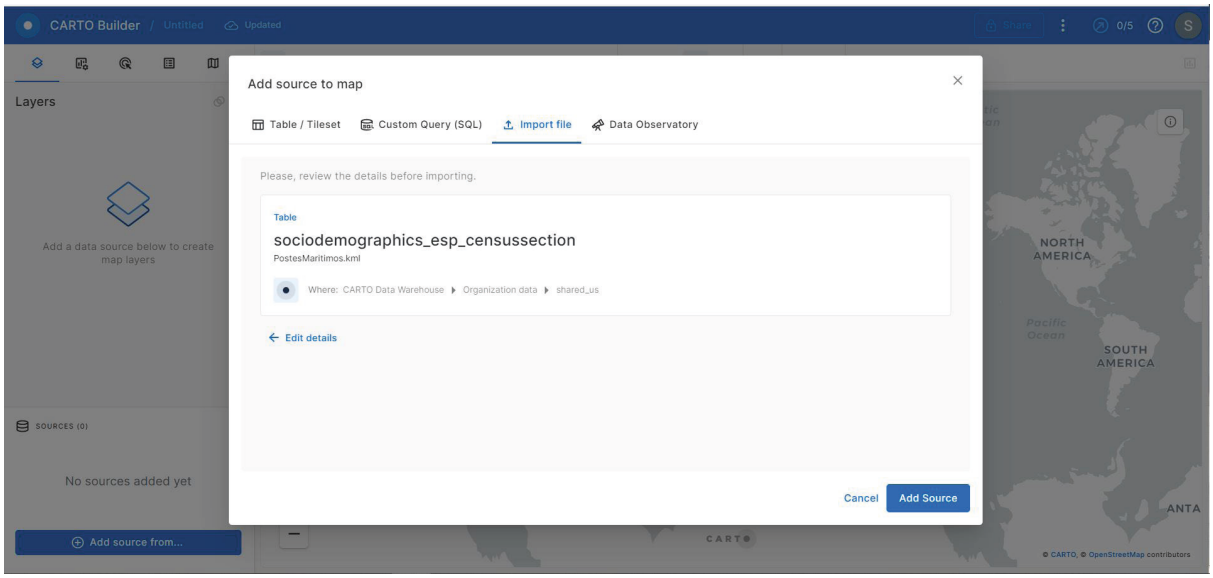



Imagen 7.

# 3.3. CASO PRÁCTICO

En este caso práctico vamos a analizar los datos públicos del Ayuntamiento de Málaga, una ciudad del sur de España, ubicada en Andalucía. Veremos la disponibilidad de aparcamientos públicos, carriles bici, puntos de aparcamientos de bici y densidad demográfica.

Los datos los hemos obtenido del portal de datos abiertos del ayuntamiento de Málaga (https://datosabiertos.malaga.eu/organization) y formato *KML – EPSG 25830* desde el apartado de datos demográfico de las bases de datos propias de CARTO.

## 3.3.1. SELECCIÓN DE BASES DE DATOS

Tal y como hemos comentado, CARTO soporta varios tipos de datos, en este ejemplo vamos a trabajar con el formato *KML- EPSG 25830*, para subir los archivos tenemos dos vías, desde la sección de mapas (tal y como hemos visto en el punto 1.2.1.) o desde *Data Explorer*, en este caso lo haremos desde este último.

Desde el menú principal de CARTO, hacemos clic en el panel lateral izquierdo *Data Explorer*, tras ello hacemos clic en el logo de subir archivos:

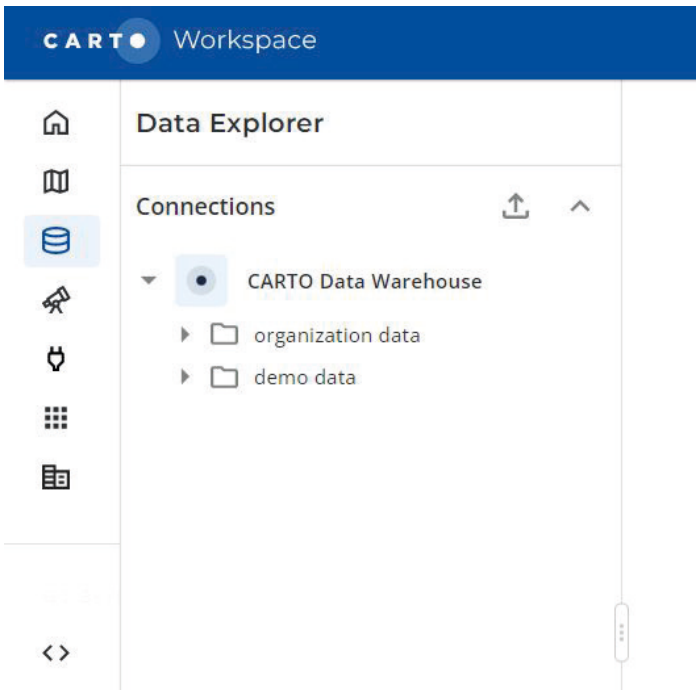



Imagen 8.

En este primer paso, seleccionamos el método en el apartado izquierdo (archivo o *link*) para este ejemplo, marcamos la opción de importar desde archivo, posteriormente buscamos en el navegador el archivo y le damos a continuar.

En este paso, buscamos la carpeta donde guardar el archivo, en este caso sería "*organization data/ shared_us*". En el campo de "*Output table*" podemos cambiar el nombre de la tabla. Finalmente hacemos clic en "*Save here*".

Por último, en el apartado 3 de "*Confirmation*", nos aparecerá un resumen de los ajustes que hemos seleccionado. Si todo es conforme, hacemos clic en "*Import*". Una vez se ha completado la barra verde, aparecerá un mensaje bajo la barra indicando "*Imported succesfully*", posteriormente hacemos clic en "*Done*".

Una vez hemos creado nuestra tabla, podemos previsualizarla desde el menú de "*Data Explorer*", seleccionando la carpeta donde lo hemos guardado.

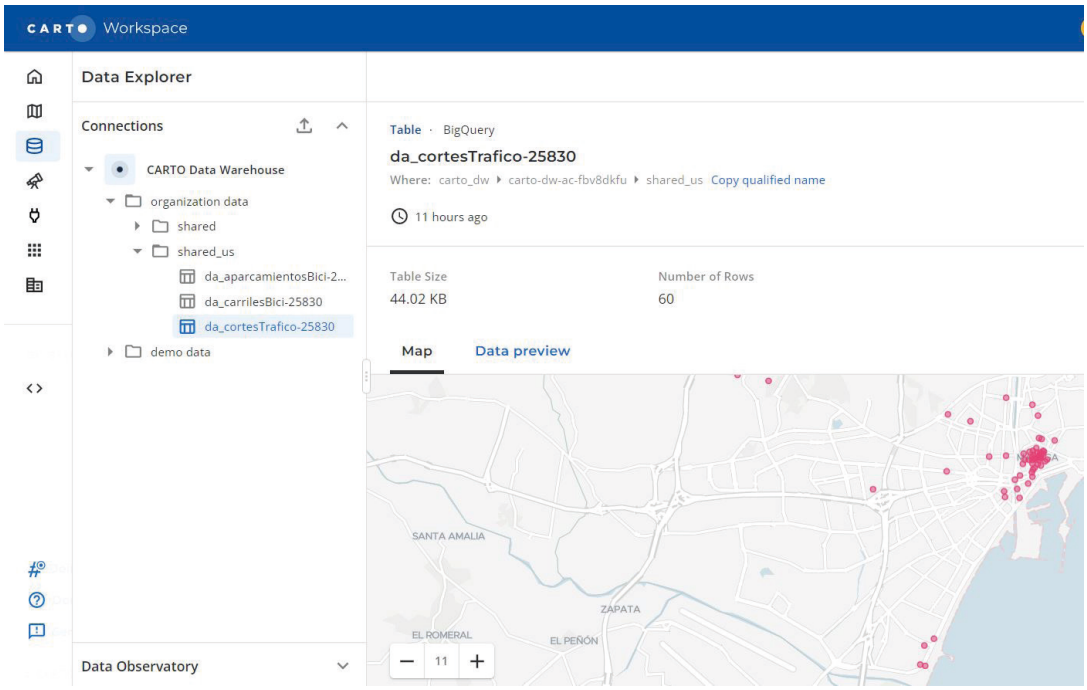



Imagen 9. Tabla importada.

## 3.3.2. CREACIÓN DE NUESTRO MAPA

En este ejemplo vamos a analizar cómo está distribuida la red de puntos de aparcamientos para bicicletas en función de la densidad demográfica de la zona, así como la red de carril bici. De esta forma veremos qué zonas están acondicionadas para el uso de la bicicleta como medio de transporte. Para ello, elaboraremos un *dashboard* que nos permita ver cómo están distribuida la red de carreteras por barrios en función de su densidad demográfica.

Para crear un nuevo mapa / *dashboard* o visualizar los previamente creados, tenemos que hacer clic en el apartado izquierdo de "*Maps*". Desde este menú podemos crear un nuevo mapa / *dashboard* haciendo clic en el panel superior derecho "*+ New map*".

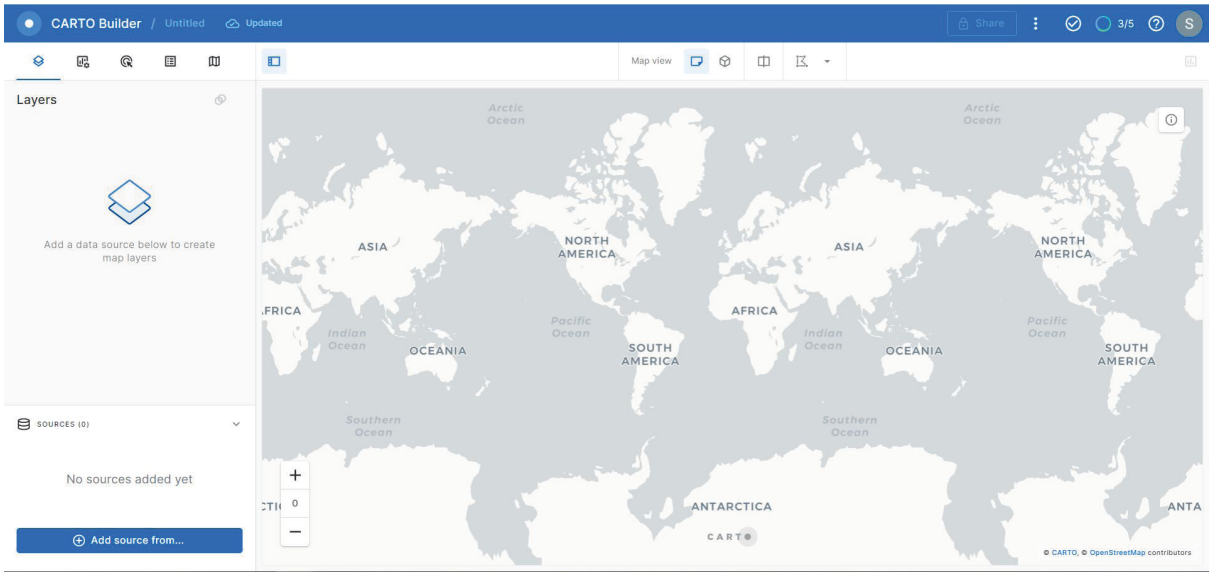



Imagen 10. Menú CARTO *Builder.*

Desde el menú de CARTO *Builder*, en el apartado de "*Layers*", en el panel inferior izquierdo bajo, hacemos clic en "*Add source from…*".

Como previamente hemos añadido nuestra tabla de datos en *Data Explorer*, ahora en el menú de selección seleccionamos "*Table/Tileset*", buscamos nuestra tabla en la ruta donde previamente la hemos creado y finalmente hacemos clic en "*Add Source*".

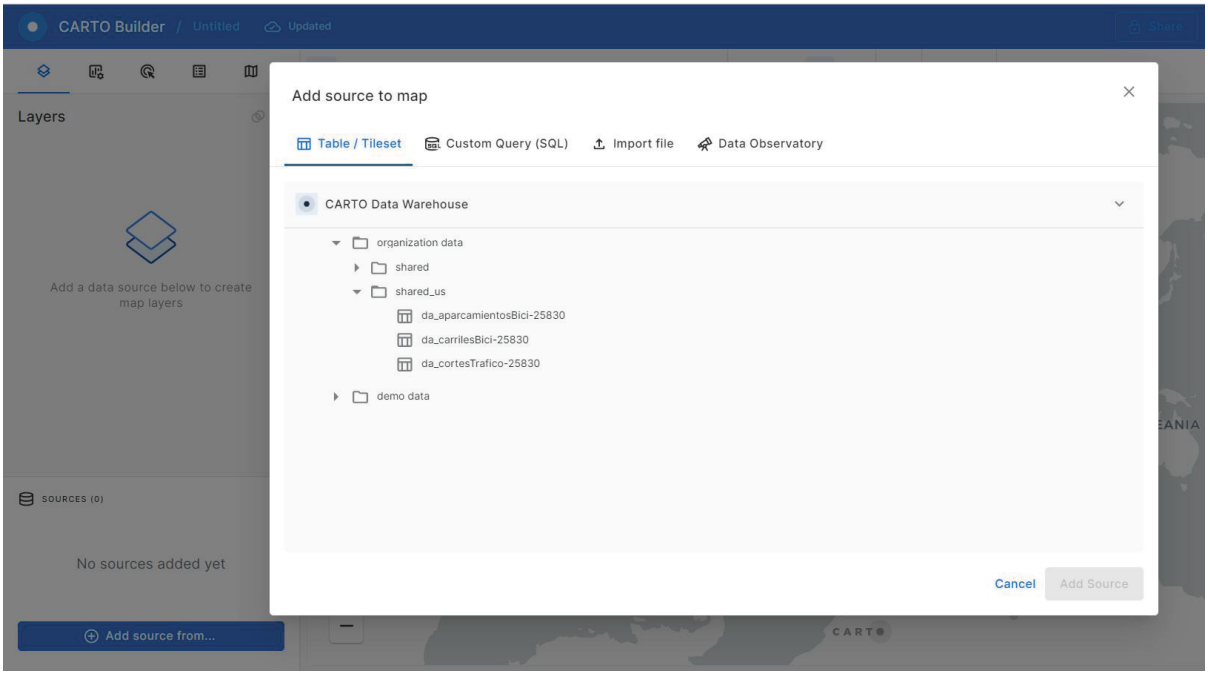



Imagen 11. Menú para añadir una capa.

Nos aparecerá una nueva capa, la que podemos renombrar desde los tres puntos situados a su derecha.

En este ejemplo vamos a trabajar con las siguientes capas:

- **Carril bici:** esta tabla contiene todas las vías de carril bici en la ciudad de Málaga (https://datosabiertos.malaga.eu/dataset/carriles-bici)

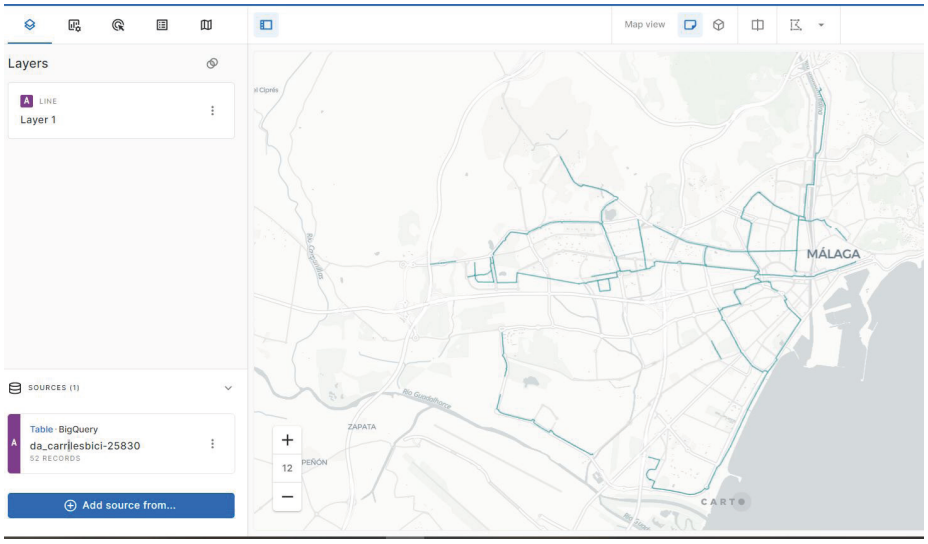



Imagen 12. Capa de carril bici.

- **Aparcamiento de bici:** contiene todos los puntos de aparcamientos para bicicletas para la ciudad de Málaga.

- **Censo demográfico de Málaga:** base de datos interna de CARTO, desde “demo data / demo_tilesets /sociodemographics_esp_censusection”.

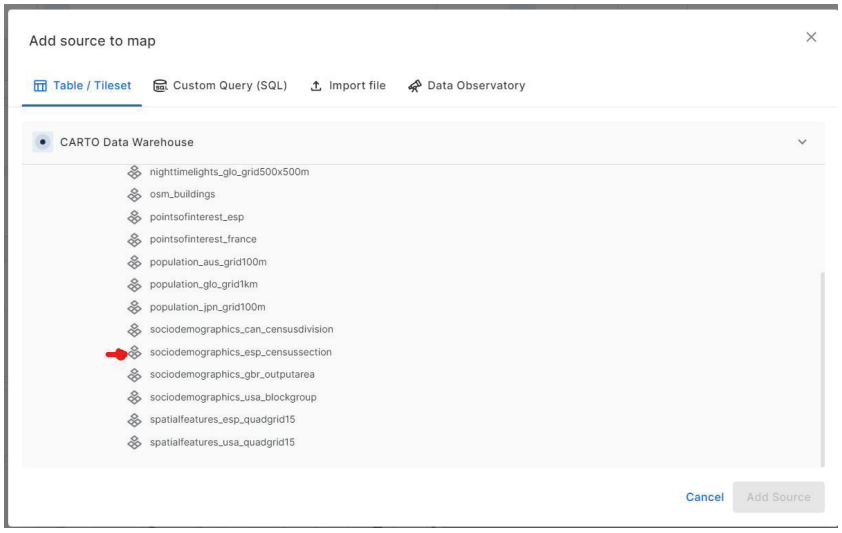



Imagen 13. Capa sociodemográfica.

Es importante que, una vez que has añadido la capa, la renombres con un nombre adecuado, para ello haremos clic en los tres puntos que aparecen en cada una de las capas del apartado “*Layers*”.

### 3.3.3. EDITADO DE NUESTRO MAPA

Para editar/personalizar una capa, tenemos que acceder a la misma mediante doble clic en la capa.

- **Carril bici**

Imagen 14. Capa carril bici editada.

Esta capa cuenta con líneas (carril bici) y puntos (inicio/fin carril bici). Podemos cambiar los puntos mediante "*Fill Color*" y el radio mediante "*Radious*", como en este caso no nos interesan estos puntos, le daremos un valor de "*Radious*" muy pequeño.

Respecto a las líneas, podemos modificar su color mediante "*Stroke Color*" y su grosor con "*Stroke Width*".

- **Aparcamientos bici**

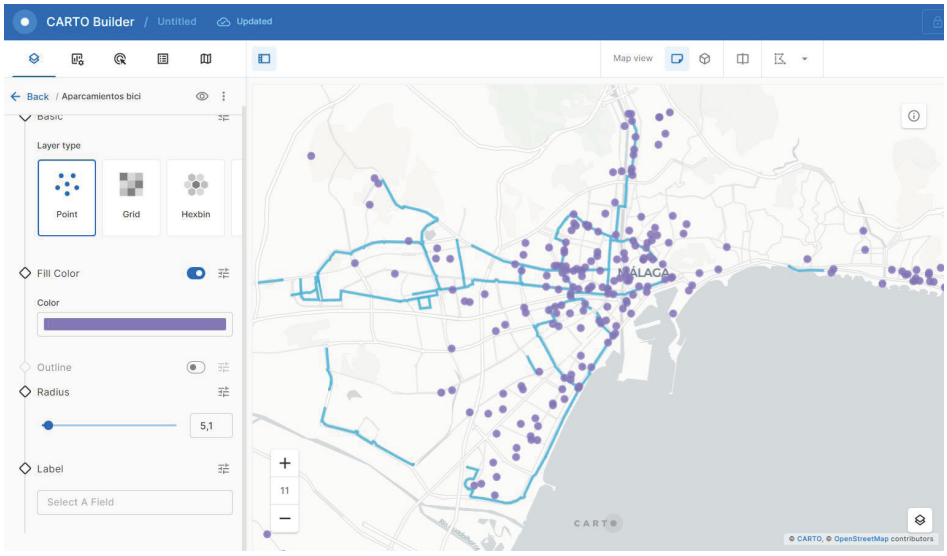



Imagen 15. Aparcamientos bici.

En este caso, tan solo aumentaremos el radio y cambiaremos el color.

- **Densidad demográfica**

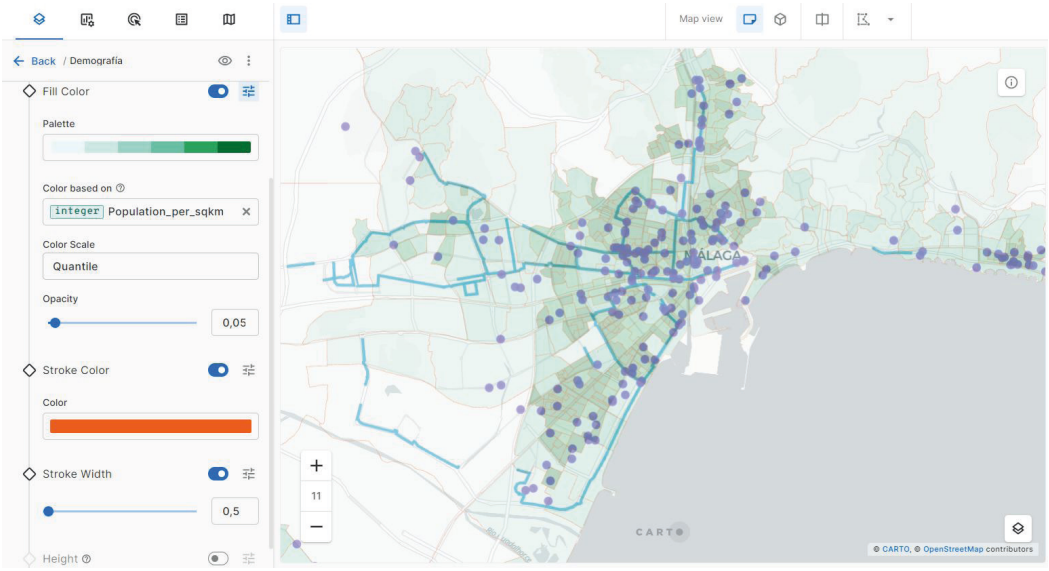



Imagen 16. Densidad demográfica.

En este caso, vamos a basar el color en función de una variable. Para ello tenemos que hacer clic en el botón de ajustes del apartado "*Fill Color*" y en "*Color base don*" seleccionamos la variable, en este caso "*Population_per_sqkm*".

Como en nuestro caso queremos que esta sea una variable secundaria respecto a la ubicación del carril bici y los puntos de estacionamiento, le daremos opacidad a esta capa mediante el parámetro de opacidad "*Opacity*", bajándolo hasta un 0.05.

Respecto a la delimitación de los perímetros, le bajaremos el grosor de la línea en el apartado de "*Stroke Color*".

## 3.3.4. INSERTAR *WIDGETS*

En este apartado insertaremos algunos *widgets* (KPI) que nos ayudarán a tener una imagen global de la zona visible. Si acercamos o alejamos el mapa, estos *widgets* se recalcularán de nuevo de forma automática.

Para insertar estos *widgets*, desde el panel de navegación hacemos clic en "Widgets" y posteriormente en el botón +, dónde nos aparecerá la tabla que queremos de base.

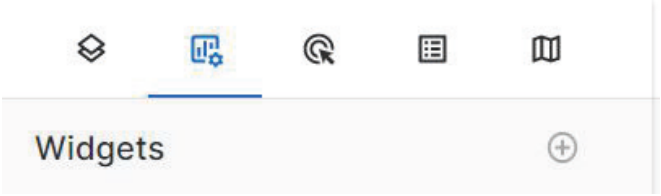



Imagen 17. *Widget.*

Para calcular la media de la densidad demográfica, seleccionamos previamente el tipo de *widget*, en este caso "*Formula*", posteriormente el tipo de operación "*AVG*", donde la variable con la que queremos operar es "*population_per_sqkm*". Finalmente, seleccionamos el formato del campo, en este caso seleccionamos el formato con dos decimales.

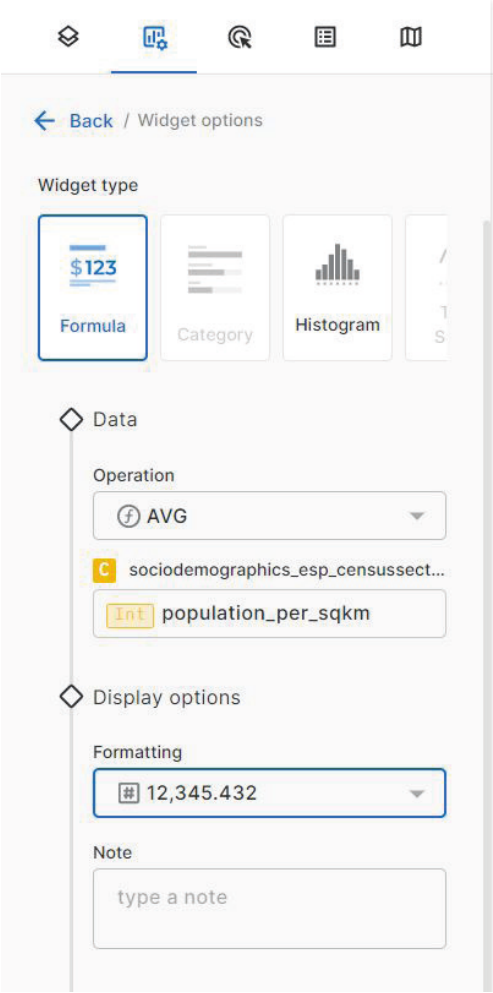



Imagen 18. Densidad demográfica.

**IMPORTANTE**

Tener en cuenta que el formato número es el formato americano, el cual es al contrario que el sistema europeo. Donde para separar decimales, utilizan punto "." y para separar miles utilizan coma ",".

Para añadir un *widget* con el número de puntos de aparcamiento de bicicletas disponibles, primero seleccionamos el tipo "*Formula*". En el apartado de data seleccionamos la operación de "*Count*". En este caso no tenemos que relacionar con ninguna columna, ya que el cálculo se basa en el conteo de números de registros que tiene nuestra tabla.

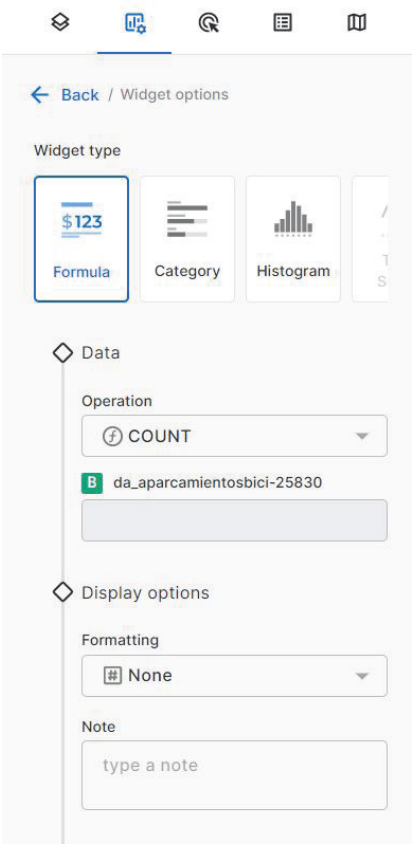



Imagen 19. Nº de puntos de aparcamiento.

El resultado del *dashboard* sería el siguiente:

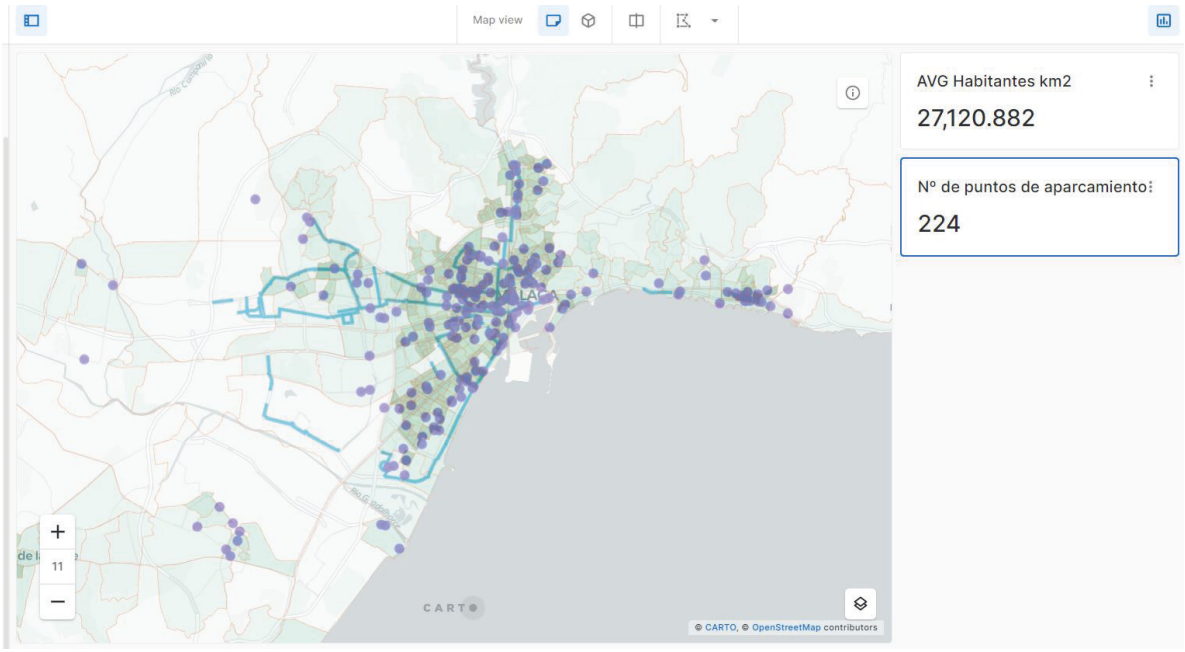



Imagen 20. *Dashboard.*

De este *dashboard* podemos sacar varias conclusiones:

- La zona este y oeste-centro de Málaga no están correctamente conectadas mediante carril bici con el resto de las redes, a pesar de disponer de puntos de estacionamiento.

- Los puntos de estacionamiento están correctamente situados en aquellas zonas que cuenta con una mayor presión demográfica, excepto la zona de San Rafael, que aún tenido una tasa mayor a 31.011 habitante por km2 a la media de otras zonas, todavía no dispone de ningún punto de estacionamiento.

Por último, le pondremos un nombre a nuestro *dashboard*, haciendo clic en el panel superior donde por predefinido aparece *"Untitled".*

# IDEAS CLAVE

- CARTO es una herramienta que, además de analizar nuestros dataset, nos permite añadir datasets muy valiosos sobre nuestro sector (datos demográficos, socioeconómicos, …).
- CARTO nos permite crear visualizaciones en nuestros mapas. A medida que avancemos en un área estos se actualizarán.
- CARTO nos permite visualizar patrones que en una tabla o gráfico no sería posible.

# GLOSARIO

- ***Workflows.*** Un flujo de trabajo que permite automatizar un proceso.
- **Datos geoespaciales.** Los datos geoespaciales son datos basados en una ubicación especifica de la superficie de la Tierra. Pueden proporcionar información sobre las relaciones entre variables y revelar patrones y tendencias. (IBM, 2022).
- **Centroide.** Punto donde se considera concentrada el área total de una figura, donde se supone está ubicado el centro geométrico del cuerpo plano y homogéneo. (Massieu, 2022).
- ***Tooltip.*** Mensaje que aparece flotante cuando desplazamos nuestro puntero sobre un punto del gráfico.

# BIBLIOGRAFÍA

CARTO. (22 de 04 de 2022). *CARTO User Manual*. Obtenido de: https://docs.carto.com/carto-user-manual/maps/introduction/

CARTO. (23 de 04 de 2022). *Getting started with CARTO*. Obtenido de: https://docs.carto.com/get-started/#:~:text=CARTO%20is%20the%20leading%20Location,store%20placements%2C%20and%20much%20more.

IBM. (25 de 04 de 2022). *Definición de datos geoespaciales*. Obtenido de: https://www.ibm.com/ar-es/topics/geospatial-data

Málaga, A. d. (22 de 04 de 2022). *Portal de datos abiertos, Ayuntamiento de Málaga*. Obtenido de: https://datosabiertos.malaga.eu/

Massieu, W. (25 de 04 de 2022). *Instituto Politécnico Nacional.* Obtenido de: https://www.ipn.mx/assets/files/cecyt11/docs/Practicas/Fisica/Fisica1/practica-7.pdf

# TEMA 4

## Microsoft Power BI

# OBJETIVOS

- Adquirir conocimiento sobre el entorno Power BI.
- Conocer las principales fuentes de datos y visuales gráficos que podemos crear en Power BI.
- Conocer el proceso ETL en Power BI.
- Crear un *dashboard* completo en Power BI.

# 4.1. INTRODUCCIÓN A POWER BI

## 4.1.1. ¿QUÉ ES POWER BI?

Power BI es un servicio de análisis de datos de Microsoft, orientado a proporcionar visualizaciones interactivas y capacidades de inteligencia empresarial con una interfaz lo suficientemente simple como para que los usuarios finales puedan crear por sí mismos sus propios informes y paneles.

El origen de estos datos puede ser desde un archivo CSV o una hoja Excel hasta un entorno de bases de datos en la nube.

Power BI nos permite conectarnos a estas fuentes de datos de una forma fácil e intuitiva para, posteriormente, crear visualizaciones donde compartir los *insights* que nos muestran los datos.

## 4.1.2. ESTRUCTURA DE POWER BI

Como se ha comentado, Power BI está formado por varios elementos que interaccionan entre sí. Sus principales componentes serían los siguientes:

- **Power BI Desktop.** Es la aplicación de escritorio local de Windows.
- **Power BI Service.** Es un servicio SaaS (*software* como servicio) *online* que nos permite visualizar, compartir y crear contenido en la web de Power BI Service.
- **Aplicaciones Móviles.** Visualización de *dashboards* desde la aplicación móvil.

**¿SABÍAS QUE...?**

**Power BI Desktop** solamente está disponible para Windows de forma oficial.

## 4.1.3. INSTALACIÓN DE POWER BI DESKTOP

Para instalar Power BI, disponemos de dos opciones:

1. **Página web de Microsoft.** Podemos descargarlo desde el siguiente *link*: https://powerbi.microsoft.com/es-es/downloads/.

   Es importante realizar la descarga desde Opciones avanzadas de descarga, ya que esto nos permitirá seleccionar el tipo de aplicación (64 o 32 bits), así como el idioma de descarga.

Imagen 1. Descarga Power BI Desktop (Web).

2. **Microsoft Store.** Desde la propia aplicación de Microsoft Store de nuestro escritorio, buscamos “Power BI Desktop”.

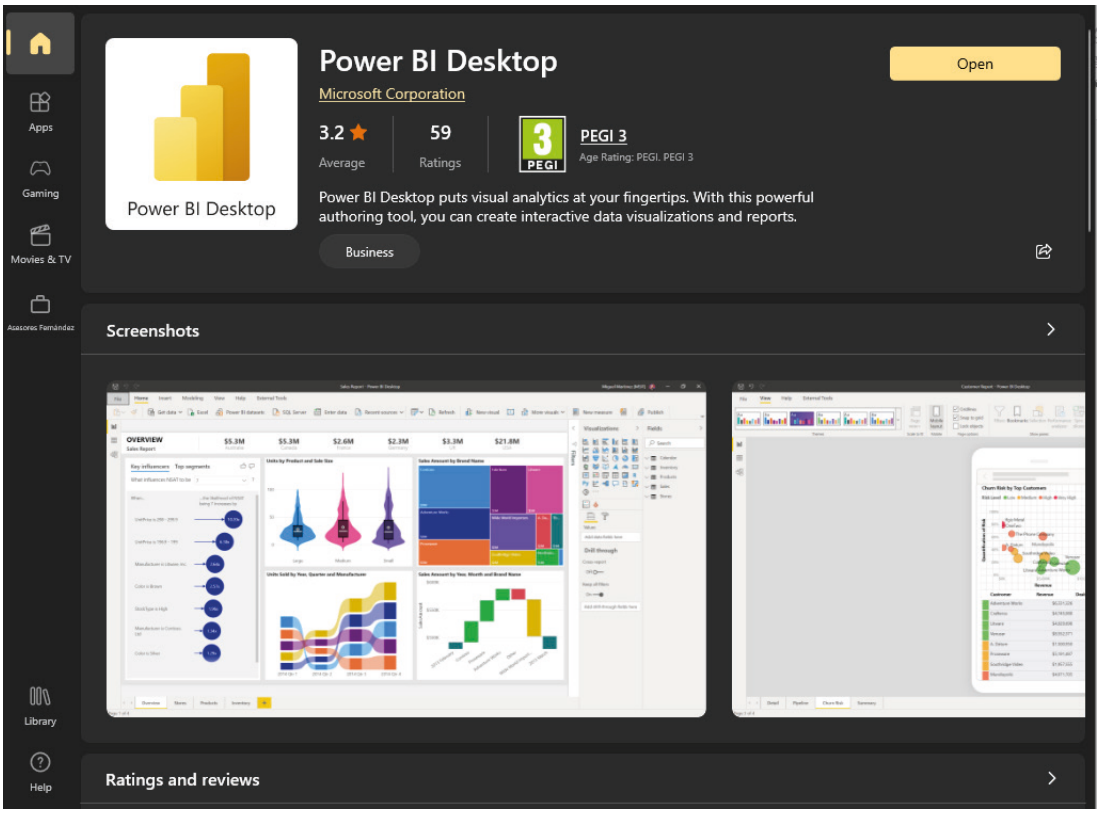



Imagen 2. Descarga Power BI Desktop (Microsoft Store).

**¿SABÍAS QUE...?**

Power BI se actualiza prácticamente con carácter mensual, con las solicitudes que los usuarios han solicitado. Por ello, recomendamos siempre la opción 1 de descarga, ya que nos permite tener un control de versiones.

## 4.1.4. POWER BI DESKTOP

Power BI Desktop es una aplicación de uso en equipo local que permite conectarse a los datos, transformarlos y visualizarlos de forma gratuita. Desde el apartado de modelado podemos conectar y combinar varias fuentes de datos. Una vez tenemos nuestro conjunto de datos, podemos crear elementos con el objetivo de mostrarlos en su conjunto en un *dashboard*. Podemos distribuir *dashboards* de forma interna o externa en nuestra organización.

**+ INFO**

Una práctica muy común es crear los *dashboards* en Power BI Desktop y distribuirlos desde Power BI Service.

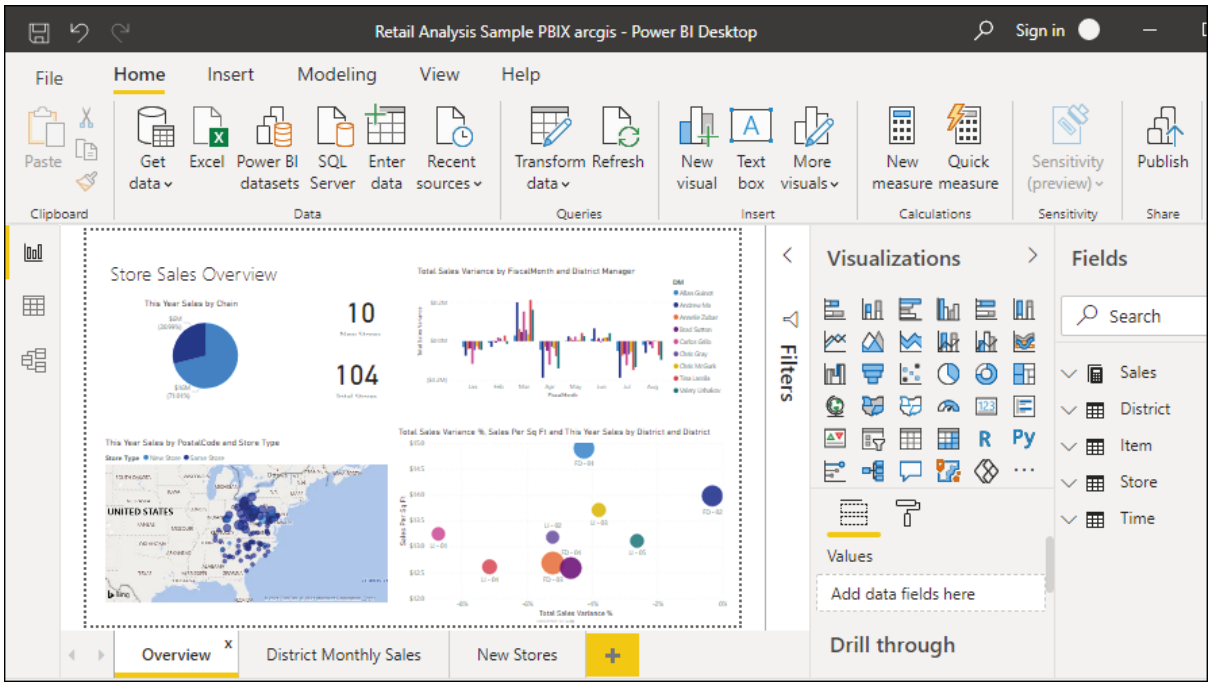



Imagen 3. Power BI Desktop.

Power BI Desktop cuenta con tres vistas disponibles. Estas se encuentran en el panel izquierdo del lienzo de trabajo. Estas vistas aparecen con el siguiente orden:

- **Informe.** Es donde se encuentra el lienzo de trabajo. Aquí podemos crear visualizaciones gráficas y *dashboards*.
- **Datos.** En esta vista visualizamos las tablas, medidas que se usan en el modelo asociado al informe. Desde este menú podemos modificar el tipo de datos, así como crear nuevos campos.
- **Modelo.** En esta vista se ven y se administran las relaciones entre las tablas de datos del modelo.

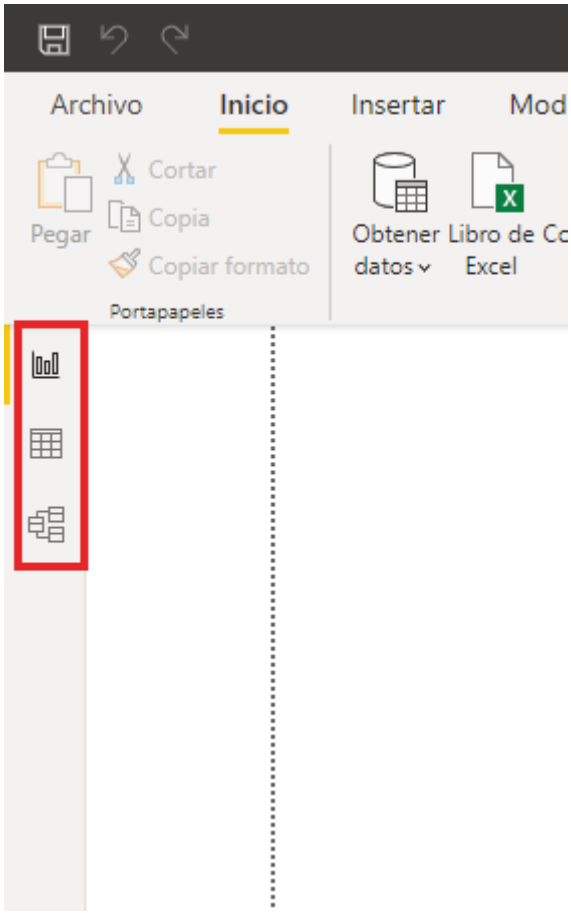



Imagen 4. Vistas Power Bi Desktop.

## 4.1.5. CONECTAR A DATOS

El primer paso para comenzar a trabajar con Power BI Desktop es conectarnos a nuestros datos.

Hay dos formas de hacerlo:

1. Una vez abierto Power BI Desktop, nos aparece la opción Obtener datos.

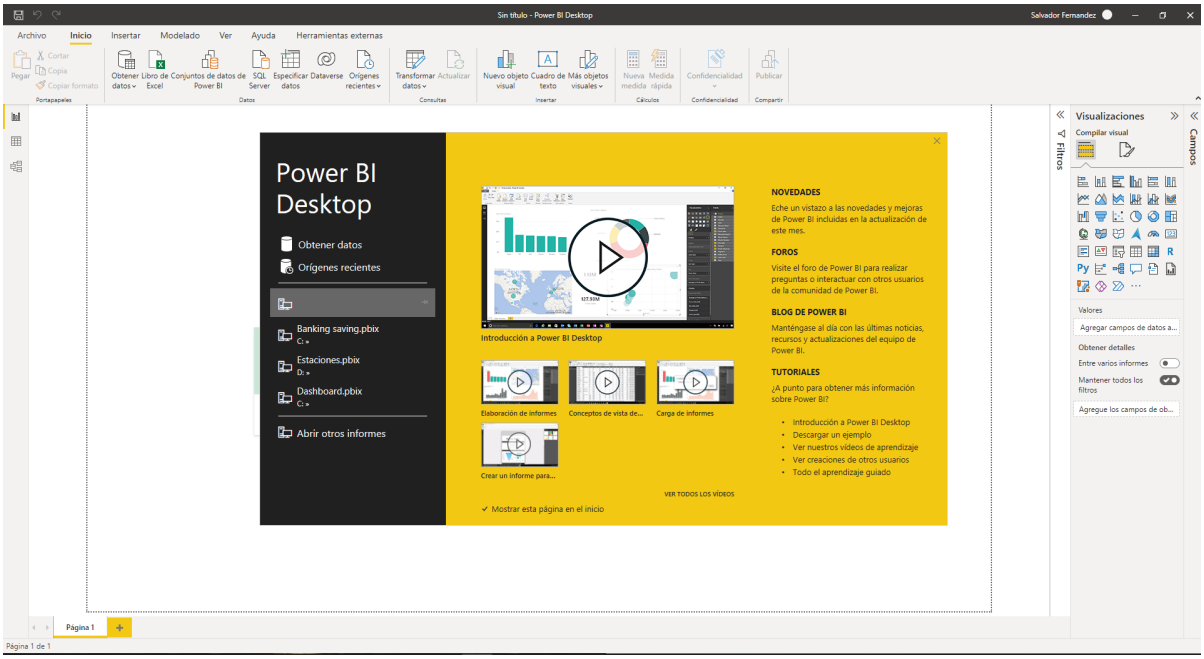



Imagen 5. Obtener datos.

2. Desde la vista de informe, hacemos clic en la cinta superior en el apartado Obtener datos.

   Desde esta ventana nos aparecerán todas las opciones de conexión de datos disponibles categorizadas por tipo de conexión.

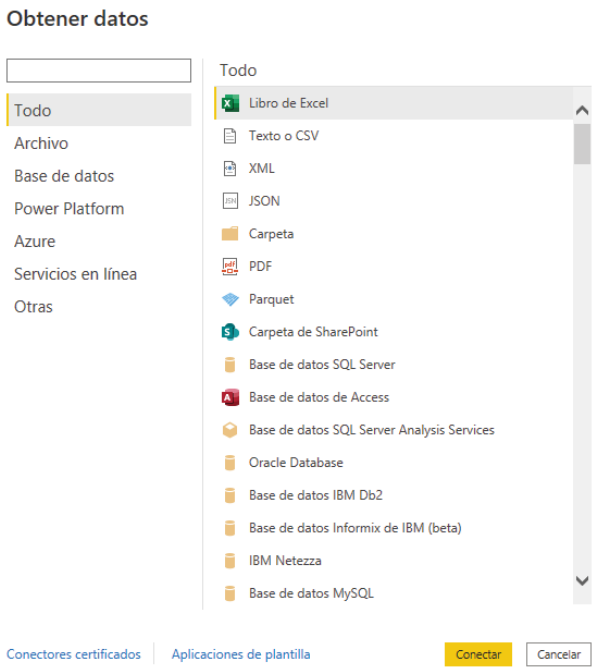



Imagen 6. Tipos de datos.

En este caso, vamos a analizar los datos de los goleadores de la selección española de futbol desde una fuente de datos pública (Wikipedia).

Para ello, en el panel de búsqueda, buscamos “web”, seleccionamos en el panel derecho y hacemos clic en Conectar.

Como es una página web, nos solicitará la URL. Hacemos clic en Aceptar.

Al tratarse de una página web, nos puede solicitar un usuario y contraseña. En este caso no lo hará, ya que es una base de datos pública.

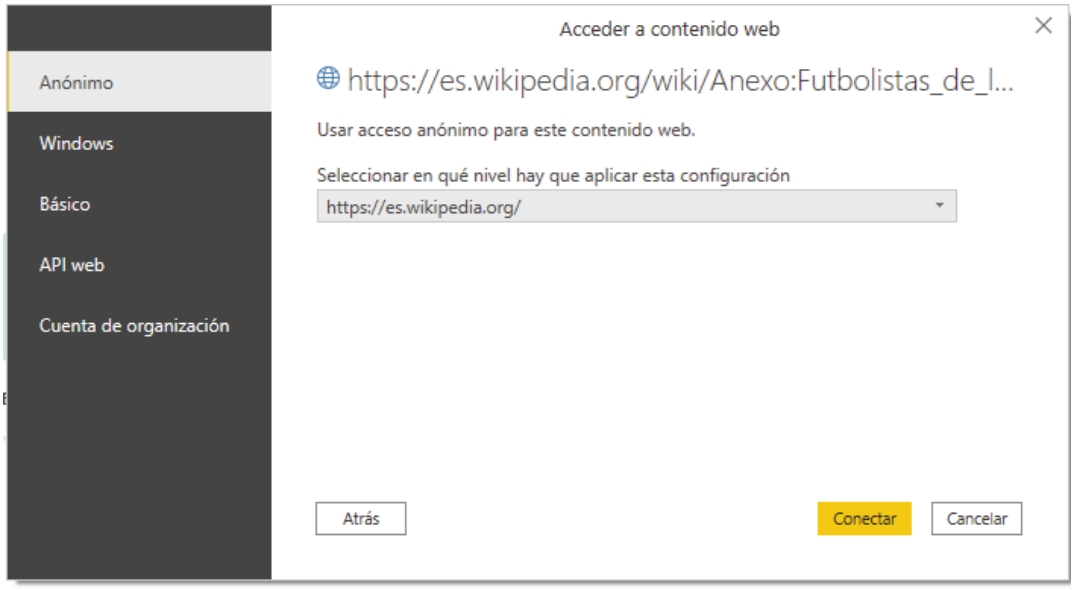



Imagen 7. Conexión URL.

Tras seleccionar la tabla o tablas, hacemos clic en Transformar datos, donde se abrirá una nueva ventana con el apartado de modelado.

## 4.1.6. MODELADO

Desde el apartado de modelado podemos transformar, limpiar los datos mediante el editor de Power Query integrado. Con este editor podemos realizar las siguientes modificaciones:

- Cambiar tipo de datos.
- Quitar columnas.
- Promover una fila.
- Combinar varios orígenes de datos.

Al apartado de modelado podemos entrar siempre que queramos desde el lienzo principal, Inicio, Transformar Datos:

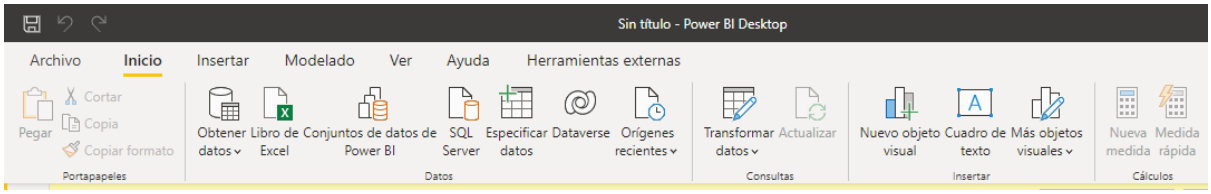



Imagen 8. Transformar datos.

Cada paso que realizamos queda registrado en el panel derecho en el editor de Power Query. Cada vez que se conecta al origen de datos se ejecutan esos pasos de modo que los datos siempre tengan la forma especificada. Esto nos permite automatizar procesos de descarga de documentos o edición de archivos.

Una vez los datos tienen la forma adecuada, le damos a Cerrar y aplicar.

## 4.1.7. RELACIÓN DE TABLAS

Una vez hemos cargado las tablas, tenemos que crear la relación entre ellas. Para ello, acudimos al apartado de modelo, donde nos aparecerá una relación que automáticamente ha elaborado Power BI.

Si queremos eliminar esta relación, hacemos botón derecho sobre la línea y hacemos clic en Eliminar. Para crear una nueva relación, seleccionaremos el campo y lo arrastraremos hasta el campo de la otra tabla.

Si queremos modificar la relación entre los campos (uno a uno, uno a varios, etc.) lo haremos haciendo doble clic sobre la línea. Este sería el resultado de nuestro ejemplo:

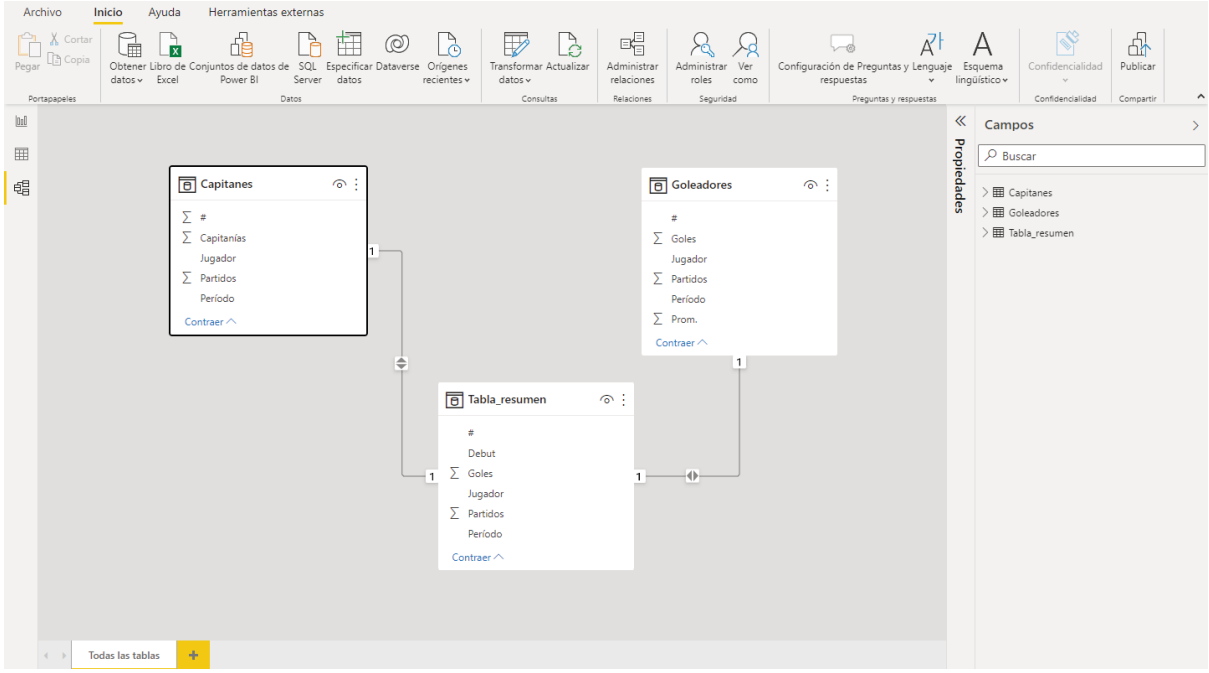



Imagen 9. Modelo de tablas.

## 4.1.8. CREACIÓN DE VISUALES

Una vez que hemos creado nuestro modelo de datos, podemos pasar al lienzo de trabajo. En él tenemos dos formas de crear un visual:

- Haciendo doble clic en el objeto visual desde el panel de visualizaciones y arrastrando el campo de la tabla.
- Arrastrando el campo de la tabla y luego seleccionando el objeto visual deseado desde el panel de visualizaciones.

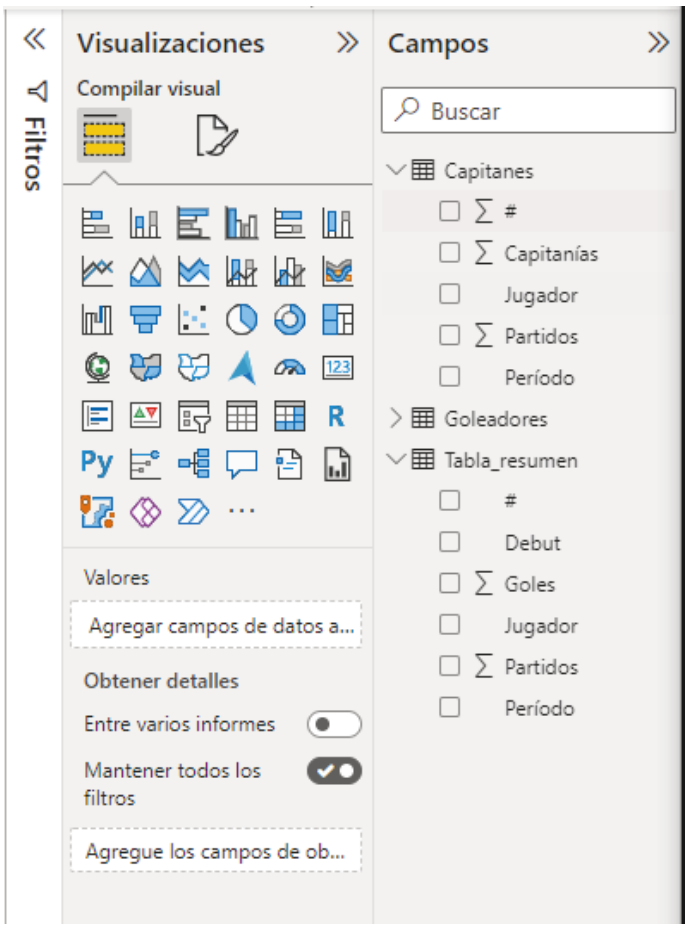



Imagen 10. Panel de creación visual.

Al conjunto de visualizaciones se le denomina informe o *dashboard*. Podemos crear o navegar desde los diversos informes desde el panel inferior.

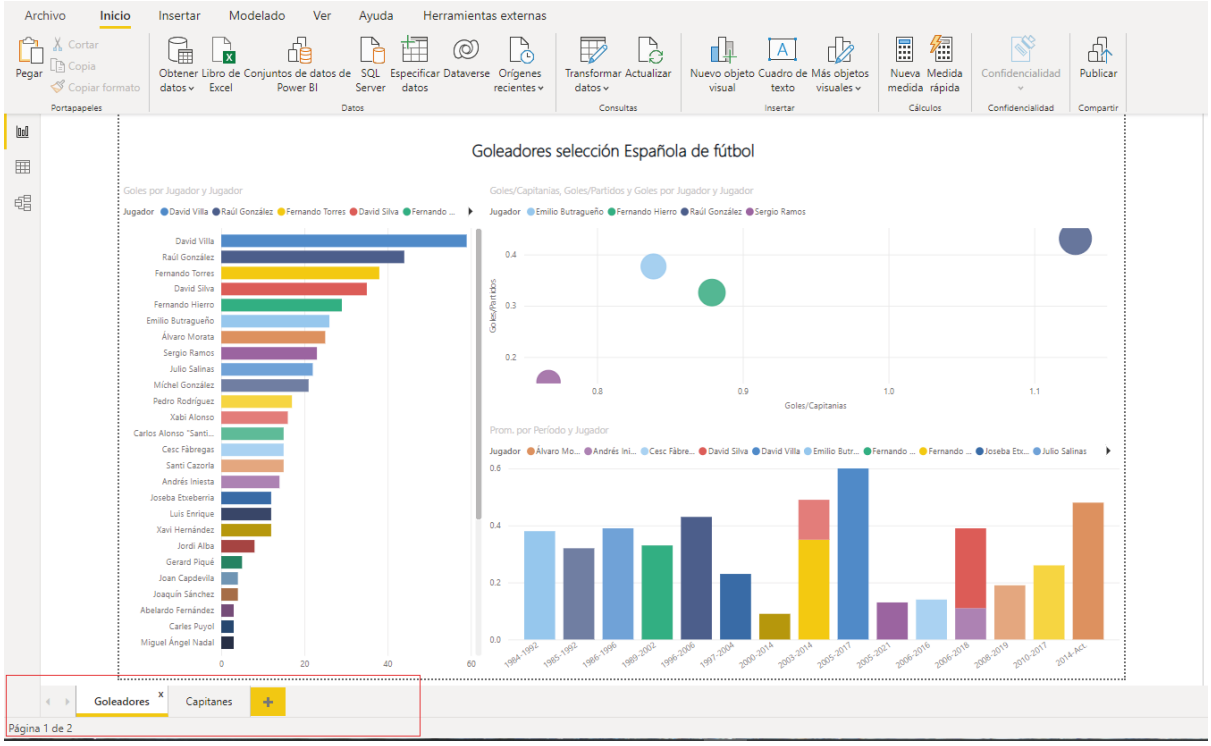



Imagen 11. Panel de navegación.

## 4.1.9. COMPARTIR INFORMES

Una vez hemos terminado nuestro informe, procedemos a guardar el archivo mediante la cinta de botones en la parte superior izquierda. Tras ello, procedemos a publicar el informe.

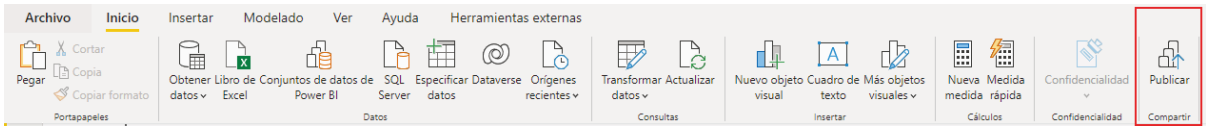



Imagen 12. Publicación del informe.

Nos solicitarán iniciar sesión en nuestra cuenta de Microsoft.

**IMPORTANTE**

Para usar el servicio de Power BI Service necesitas una cuenta con mínimo Office 365 E3 o una cuenta de dominio privado (@sony.es, @planeta, etc.).

Una vez hemos iniciado sesión, seleccionaremos el apartado donde subimos nuestros informes.

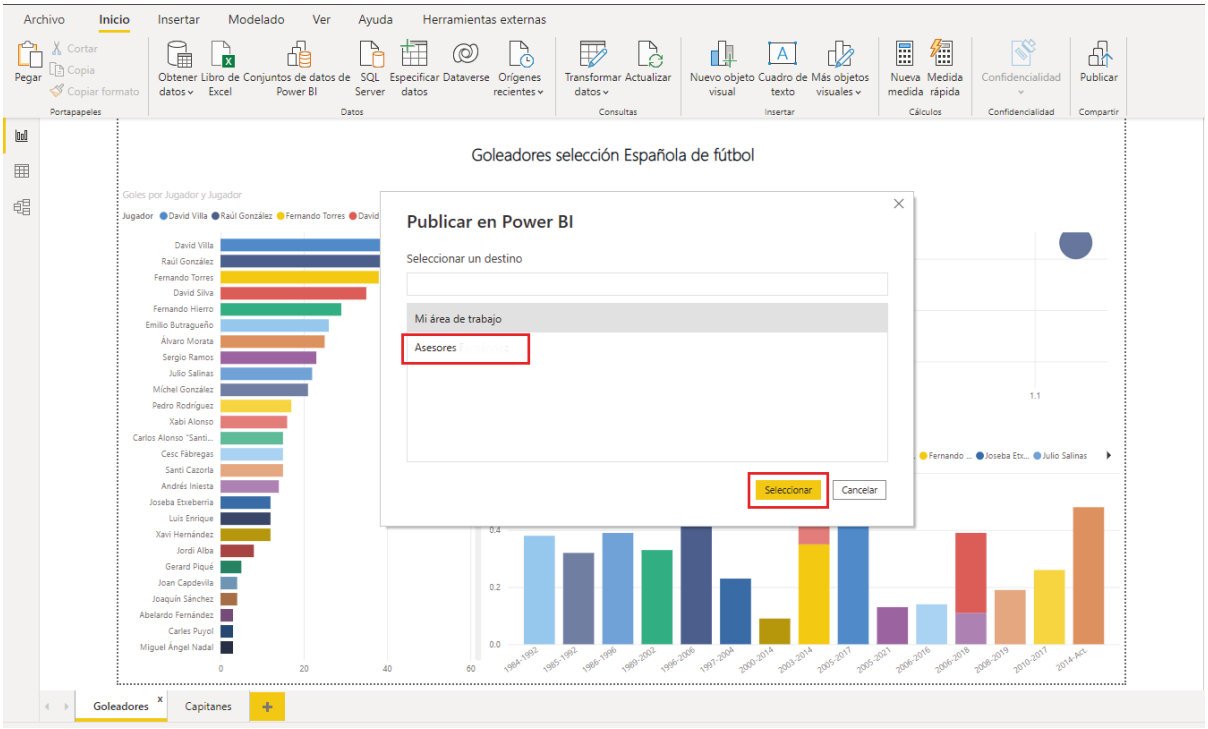



Imagen 13. Publicar en Power BI.

## 4.1.10. POWER BI SERVICE

Una vez hemos publicado nuestro *report* en nuestro *Workspace*, nos aparecerá la siguiente ventana, compuesta por los siguientes elementos:

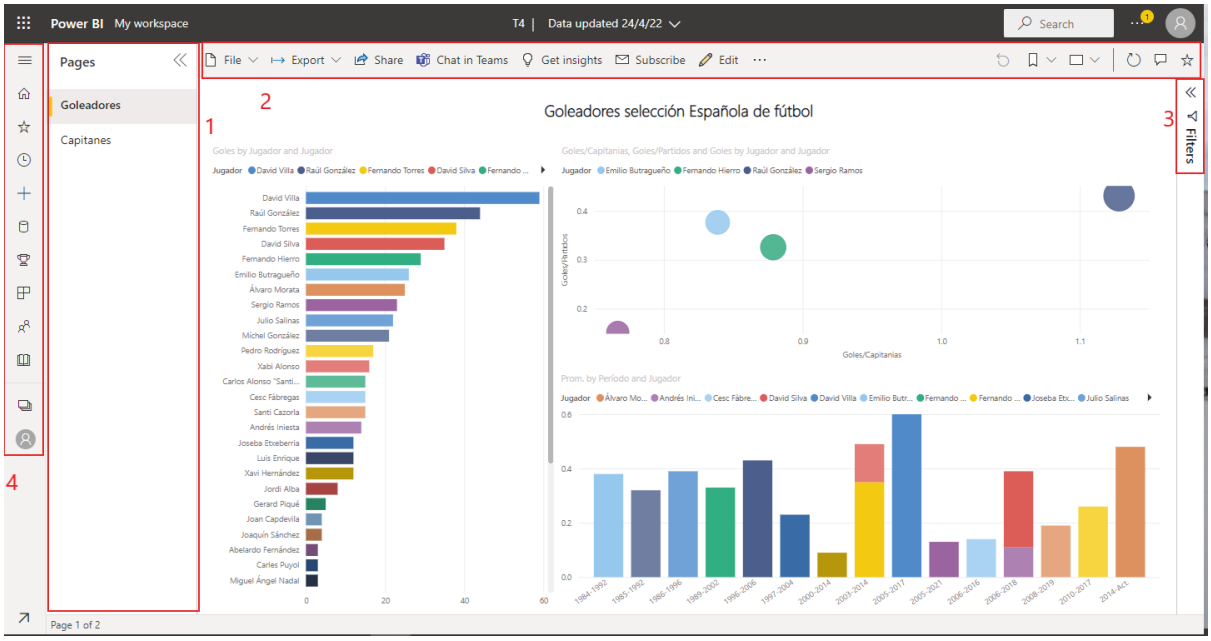



Imagen 14. Power BI Service.

1. **Panel de navegación.**

2. **Barra de trabajo.** Desde aquí podemos descargar el archivo para su uso en Power BI Desktop, exportarlo (PDF, PowerPoint y PDF), compartir (solo para usuarios PRO), Chat in Teams (solo para usuarios PRO), obtener *insights* (solo para usuarios PRO), suscribirnos (solo para usuarios PRO) o editar en Power BI Service.

**¿SABÍAS QUE...?**

Es posible compartir el *dashboard* de forma pública desde File/Embed Report/Publish to web (*public*).

La información que se publique será pública, por lo que cualquiera que tenga la URL tendrá acceso al *dashboard* y a los datos del *dataset*.

3. ***Filters.*** Los mismos filtros que tenemos en el Power BI Desktop.

4. **Menú de navegación Power BI Service.**

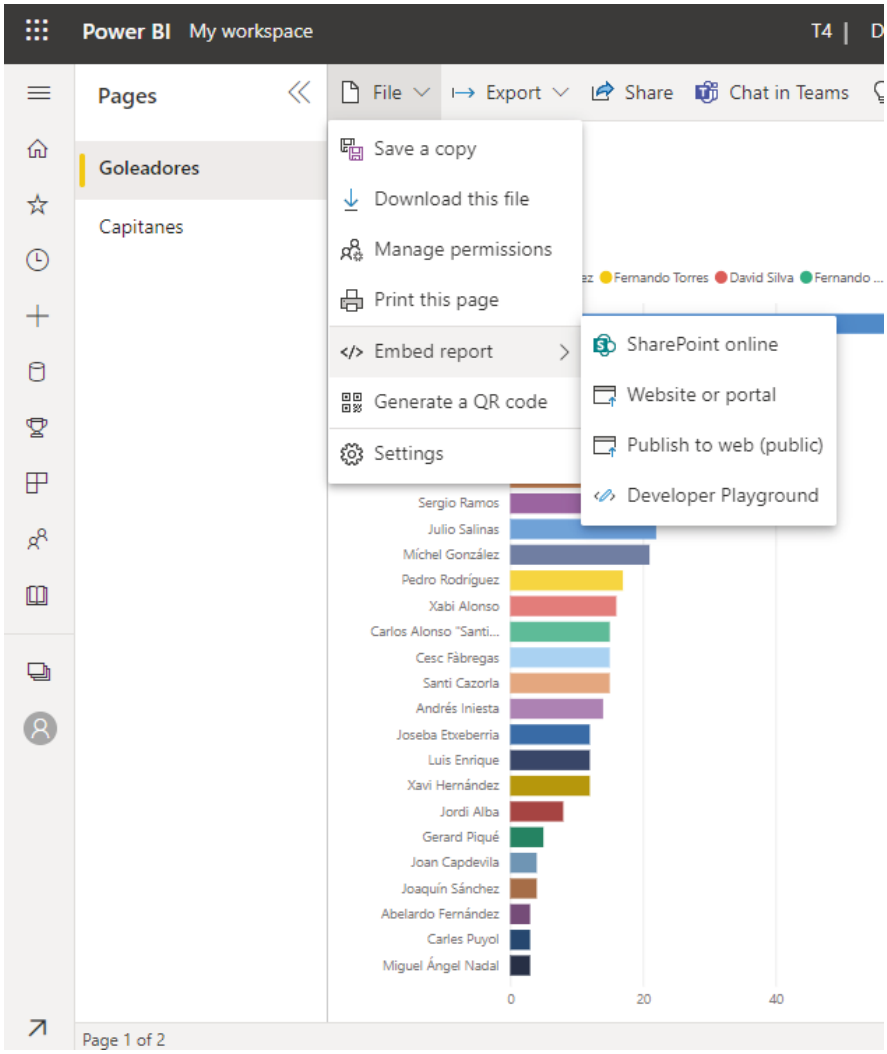



Imagen 15. Publicación de forma pública.

# 4.2. TIPOS DE FUENTES DE DATOS Y TIPOS DE GRÁFICOS

## 4.2.1. TIPOS DE FUENTES DE DATOS

El cuadro de orígenes de datos se agrupa en las siguientes categorías:

- **Todo.**
- **Archivo:** fichero Excel, CSV, texto, PDF, JSON, Carpeta, etc.
- **Bases de datos:** SQL, Access, Google, Oracle, IBM, MySQL, Amazon Redshift, SAP HANA...
- **Power Platform:** conjuntos de datos de Power BI, flujos de datos Power BI, Common Data Service, Dataverse, Flujo de datos.
- **Azure:** SQL Database, Cosmos DB, Databricks, Cost Management, etc.
- **Servicios en línea:** lista de SharePoint, Dynamics 365, Business Central, Google Analytics, Adobe Analytics, Objetos e informes de Salesforce, etc.
- **Otros:** Web, SharePoint, OData, Active Directory, Microsoft Exchange, Archivo Hadoop (HDFS), Spark, ODBC, Solver, Paxata, Hive LLAP, etc.
- **Aplicaciones plantilla:** las aplicaciones plantilla dependen del tipo de organización. Accederemos mediante el panel inferior.

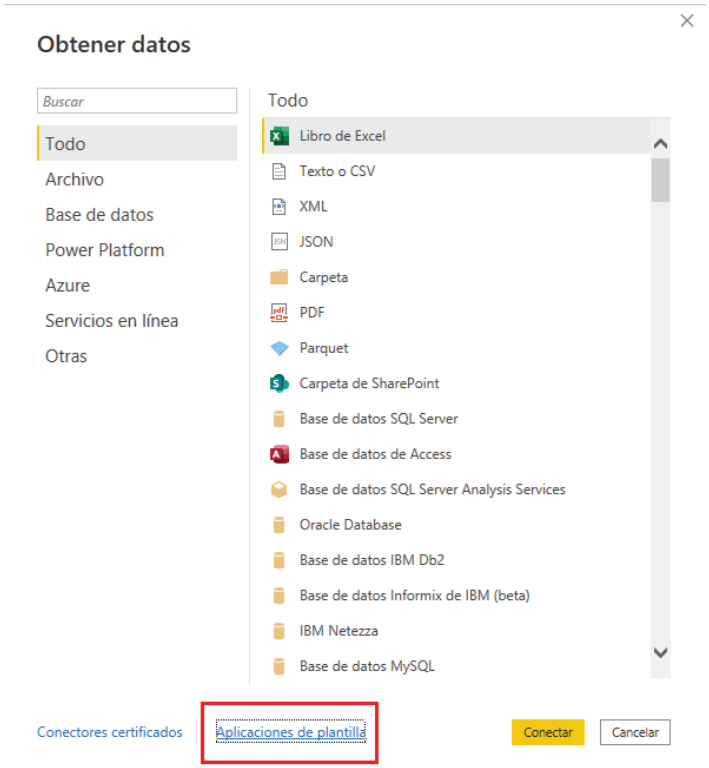



Imagen 16. Aplicaciones plantilla.

- **Archivos PBIDS:** los archivos PBIDS son las conexiones de otro archivo Power BI Desktop, que mantiene todos los datos y conexiones. Podemos encontrarlas como Conjunto de datos de Power BI en la barra de navegación.

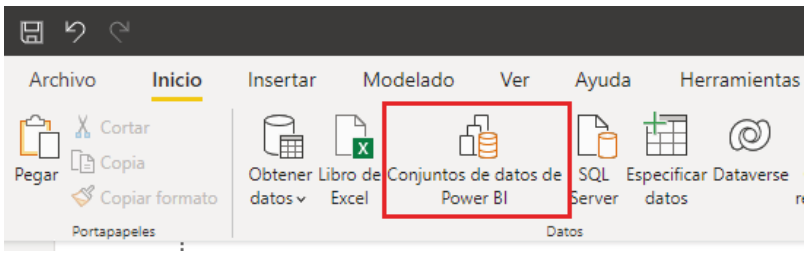



Imagen 17. Carga formato PBIDS.

### 4.2.1.1. EJEMPLO DE CONEXIÓN A UN ARCHIVO CSV

Desde Power BI podemos conectarnos a un archivo CSV desde la cinta principal en Obtener datos/Texto o CSV.

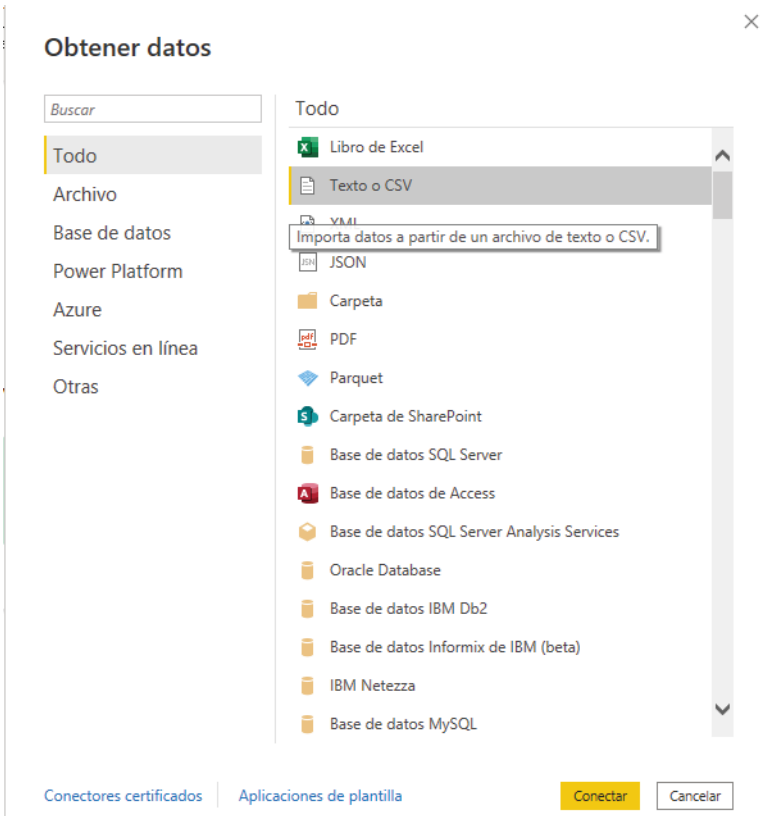
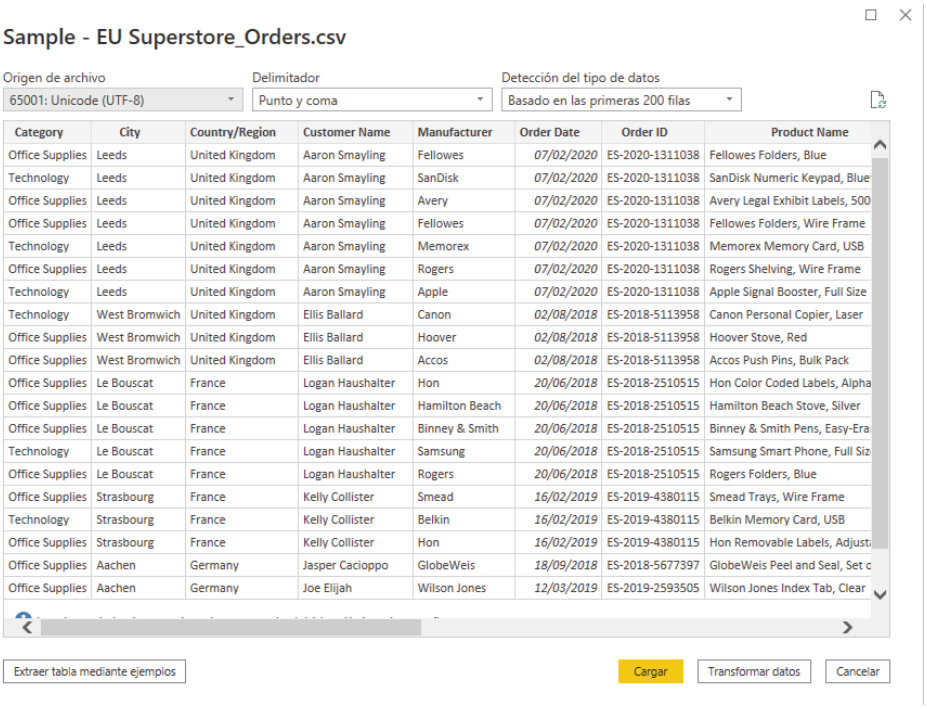



Imagen 18. Carga archivo CSV.

Seleccionamos el archivo y nos aparecerá el siguiente menú, donde tendremos que verificar que el resultado es el deseado, si no lo es así podemos modificar el tipo de lectura de archivo CSV mediante el cuadrado de Origen de archivo. Tras esto hacemos clic en Transformar datos y, si el resultado es el esperado, hacemos clic Guardar y cerrar.




| Category | City | Country/Region | Customer Name | Manufacturer | Order Date | Order ID | Product Name |
|---|---|---|---|---|---|---|---|
| Office Supplies | Leeds | United Kingdom | Aaron Smayling | Fellowes | 07/02/2020 | ES-2020-1311038 | Fellowes Folders, Blue |
| Technology | Leeds | United Kingdom | Aaron Smayling | SanDisk | 07/02/2020 | ES-2020-1311038 | SanDisk Numeric Keypad, Blue |
| Office Supplies | Leeds | United Kingdom | Aaron Smayling | Avery | 07/02/2020 | ES-2020-1311038 | Avery Legal Exhibit Labels, 500 |
| Office Supplies | Leeds | United Kingdom | Aaron Smayling | Fellowes | 07/02/2020 | ES-2020-1311038 | Fellowes Folders, Wire Frame |
| Technology | Leeds | United Kingdom | Aaron Smayling | Memorex | 07/02/2020 | ES-2020-1311038 | Memorex Memory Card, USB |
| Office Supplies | Leeds | United Kingdom | Aaron Smayling | Rogers | 07/02/2020 | ES-2020-1311038 | Rogers Shelving, Wire Frame |
| Technology | Leeds | United Kingdom | Aaron Smayling | Apple | 07/02/2020 | ES-2020-1311038 | Apple Signal Booster, Full Size |
| Technology | West Bromwich | United Kingdom | Ellis Ballard | Canon | 02/08/2018 | ES-2018-5113958 | Canon Personal Copier, Laser |
| Office Supplies | West Bromwich | United Kingdom | Ellis Ballard | Hoover | 02/08/2018 | ES-2018-5113958 | Hoover Stove, Red |
| Office Supplies | West Bromwich | United Kingdom | Ellis Ballard | Accos | 02/08/2018 | ES-2018-5113958 | Accos Push Pins, Bulk Pack |
| Office Supplies | Le Bouscat | France | Logan Haushalter | Hon | 20/06/2018 | ES-2018-2510515 | Hon Color Coded Labels, Alpha |
| Office Supplies | Le Bouscat | France | Logan Haushalter | Hamilton Beach | 20/06/2018 | ES-2018-2510515 | Hamilton Beach Stove, Silver |
| Office Supplies | Le Bouscat | France | Logan Haushalter | Binney & Smith | 20/06/2018 | ES-2018-2510515 | Binney & Smith Pens, Easy-Era |
| Technology | Le Bouscat | France | Logan Haushalter | Samsung | 20/06/2018 | ES-2018-2510515 | Samsung Smart Phone, Full Siz |
| Office Supplies | Le Bouscat | France | Logan Haushalter | Rogers | 20/06/2018 | ES-2018-2510515 | Rogers Folders, Blue |
| Office Supplies | Strasbourg | France | Kelly Collister | Smead | 16/02/2019 | ES-2019-4380115 | Smead Trays, Wire Frame |
| Technology | Strasbourg | France | Kelly Collister | Belkin | 16/02/2019 | ES-2019-4380115 | Belkin Memory Card, USB |
| Office Supplies | Strasbourg | France | Kelly Collister | Hon | 16/02/2019 | ES-2019-4380115 | Hon Removable Labels, Adjust |
| Office Supplies | Aachen | Germany | Jasper Cacioppo | GlobeWeis | 18/09/2018 | ES-2018-5677397 | GlobeWeis Peel and Seal, Set c |
| Office Supplies | Aachen | Germany | Joe Elijah | Wilson Jones | 12/03/2019 | ES-2019-2593505 | Wilson Jones Index Tab, Clear |

Imagen 19. Carga archivo CSV.

### 4.2.1.2. EJEMPLO DE CONEXIÓN WEB

Copiamos la URL donde se encuentra la tabla que queremos importar. Desde Power BI podemos conectarnos a un archivo CSV desde la cinta principal en Obtener datos/Web, y pegar la URL. Tras ello, hacemos clic en Cargar datos.

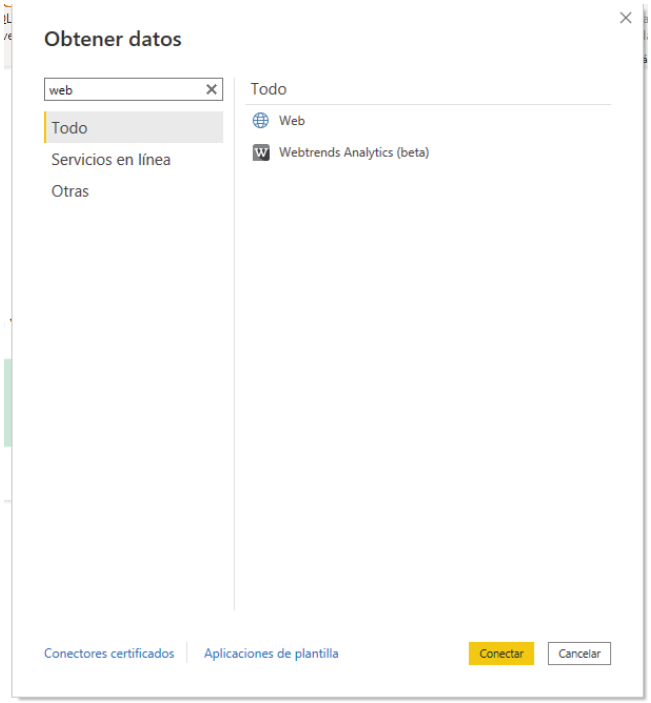



Imagen 20. Carga de datos web.

**¿SABÍAS QUE...?**

Si para acceder a la página web necesitamos iniciar sesión con nuestro usuario y contraseña, podremos añadir esta información durante el mismo proceso.

### 4.2.1.3. EJEMPLO DE CONEXIÓN A UN ARCHIVO GITHUB

GitHub es una de repositorios de *notebooks*, así como datos. Durante la pandemia se hizo viral el uso de esta web. Nos permite actualizar nuestro *dashboard* de forma automática, ya que, mediante el conector de web, una vez el autor de Github actualiza los datos, podemos actualizarlos de forma sencilla.

Primero, acudimos al siguiente link, donde se encuentra el repositorio: https://github.com/owid/covid-19-data/blob/master/public/data/owid-covid-data.csv. Es importante que, una vez tengamos abierto el CSV, hagamos clic en View raw.

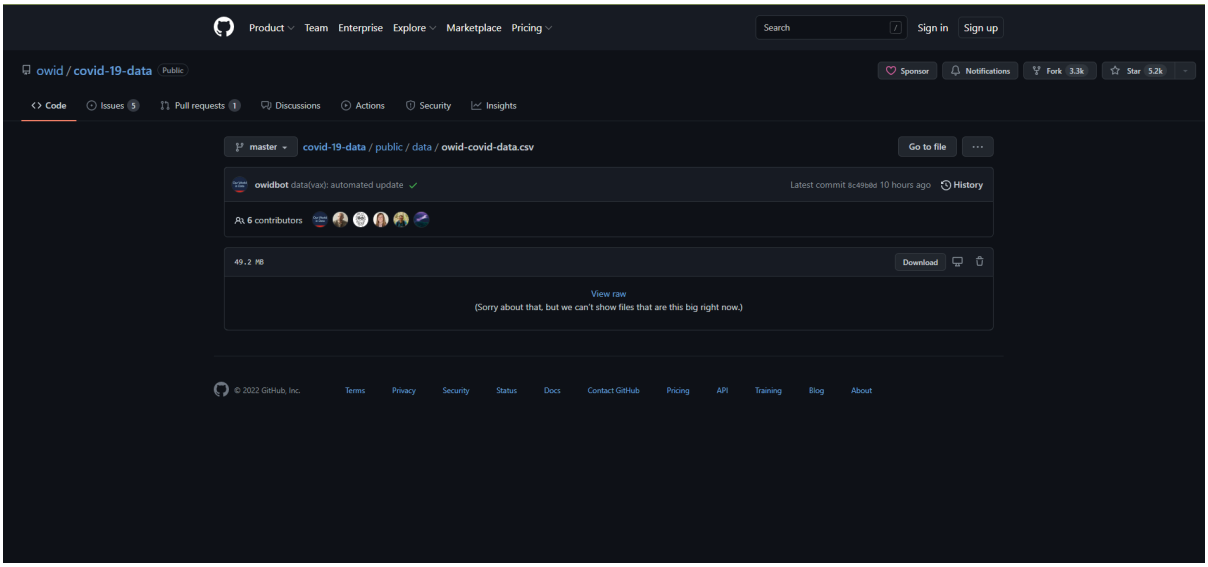



Imagen 21. Repositorio casos COVID-19.

Una vez estamos en la ventana donde tenemos el archivo en RAW, podemos copiar la URL de la página.

Los siguientes pasos son iguales al punto 4.2.1.2. ya que se trata de un archivo en línea (por lo tanto, en URL), seleccionamos la URL que nos aparece desde la ventana de RAW.

### 4.2.1.4. EJEMPLO DE CONEXIÓN A UN ARCHIVO EXCEL *ONLINE*

La conexión a documentos Excel se actualizan en una web con una determinada cadencia. Los podemos actualizar de forma automática con la conexión web. En este ejemplo, vamos a conectarnos a la web del Ministerio de Fomento, donde, con carácter mensual, actualizan los datos de visados de viviendas.

Imagen 22. Conexión a documento Excel *online*.

Tal y como podemos ver, en la parte inferior aparece el nombre del documento Excel, así como su formato (.XLS). Hay que tener en cuenta que Microsoft Power BI no es muy eficiente con el formato .XLS, ya que es un formato de Excel obsoleto. No obstante, si son pocos archivos, no debería dar problemas.

Hacemos clic con el botón derecho sobre el link donde se encuentra la URL y la copiamos. Esta será la URL que utilizaremos en nuestro proceso en Power BI Desktop.

El resto del proceso es igual al punto 4.2.1.2. ya que se trata de un archivo en línea.

## 4.2.2. TIPOS DE VISUALIZACIONES

### GRÁFICOS DE ÁREAS

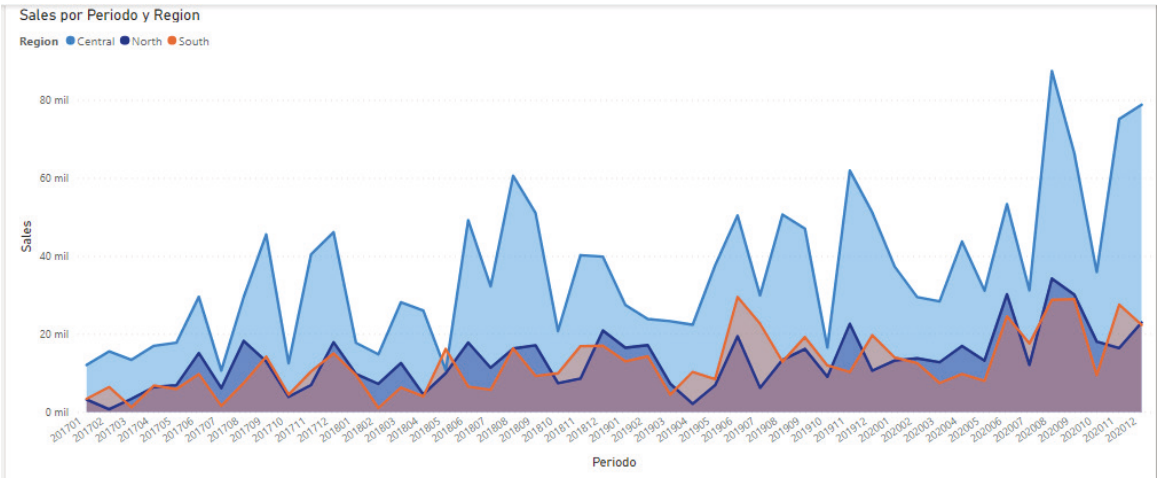



Imagen 23. Gráfico de áreas.

## GRÁFICOS DE BARRAS Y COLUMNAS

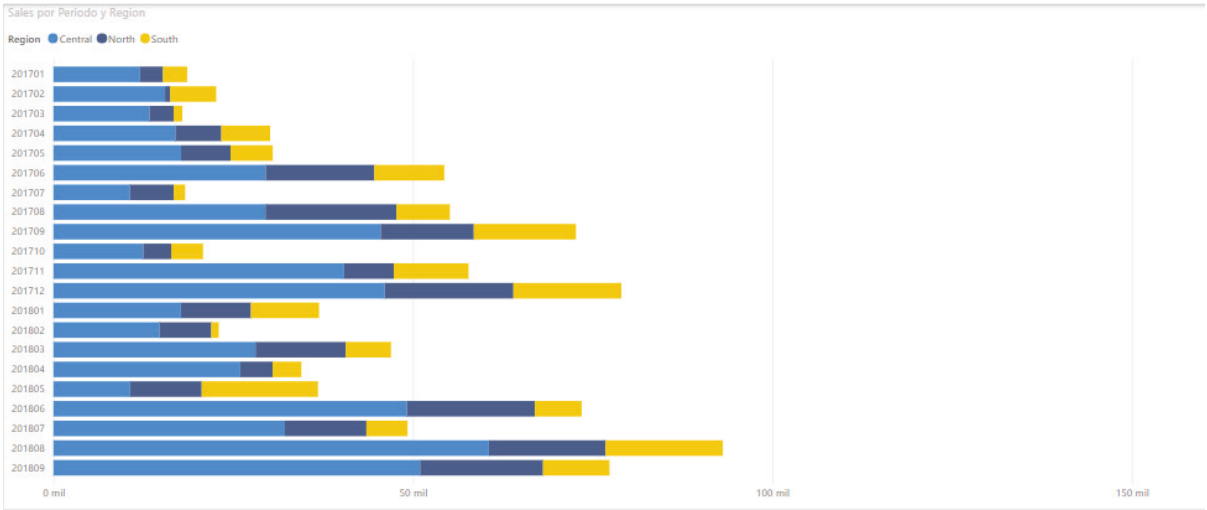



Imagen 24. Gráfico de barras y columnas.

## KPI

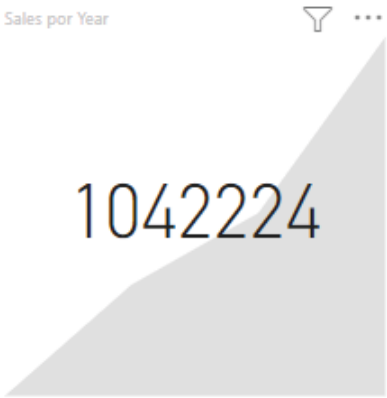



Imagen 25. KPI.

## KPI SOLO

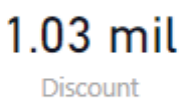



Imagen 26. KPI solo.

## GRÁFICOS COMBINADOS

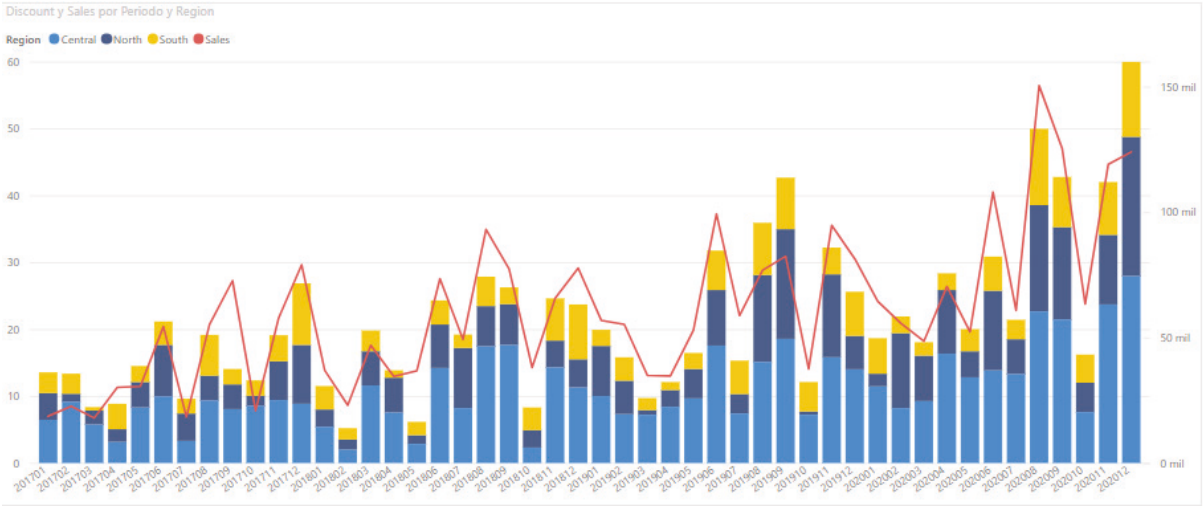



Imagen 27. Gráficos combinados.

## JERARQUÍA

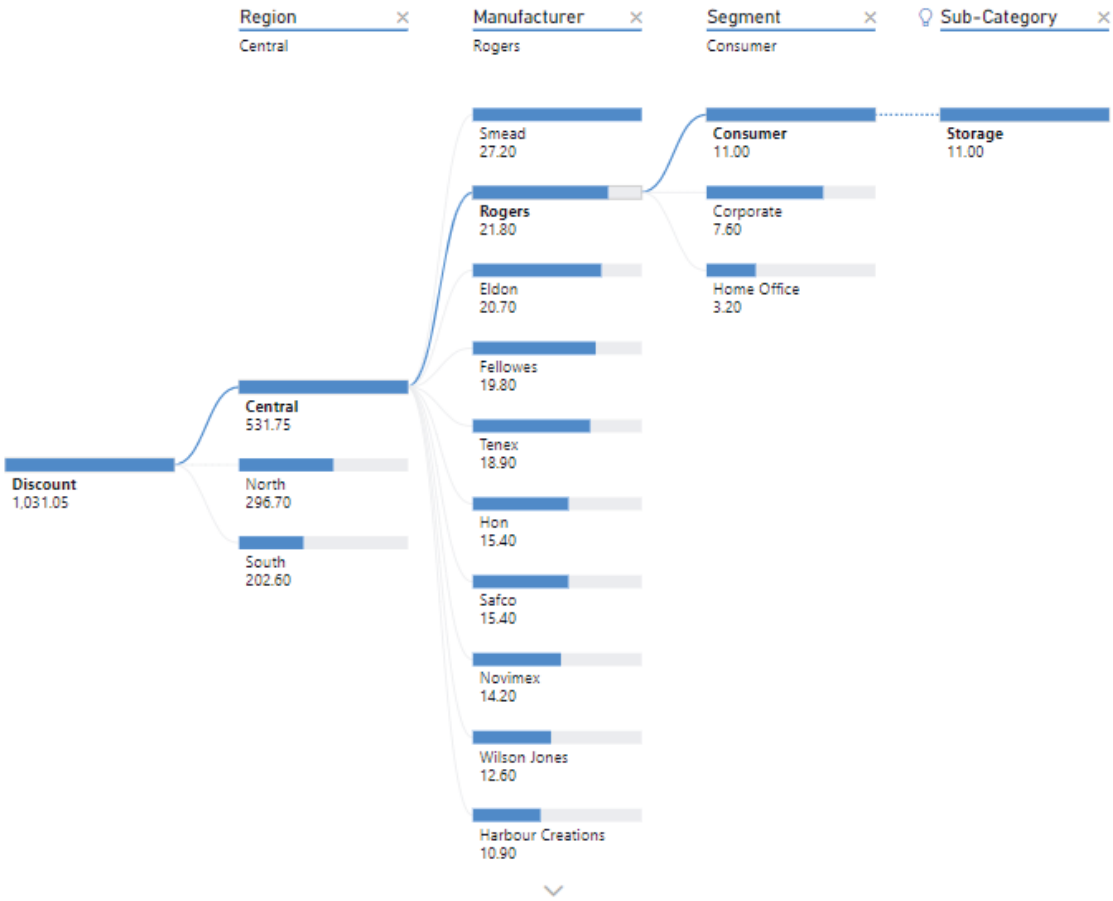



Imagen 28. Gráfico de jerarquía.

## GRÁFICO DE ANILLO

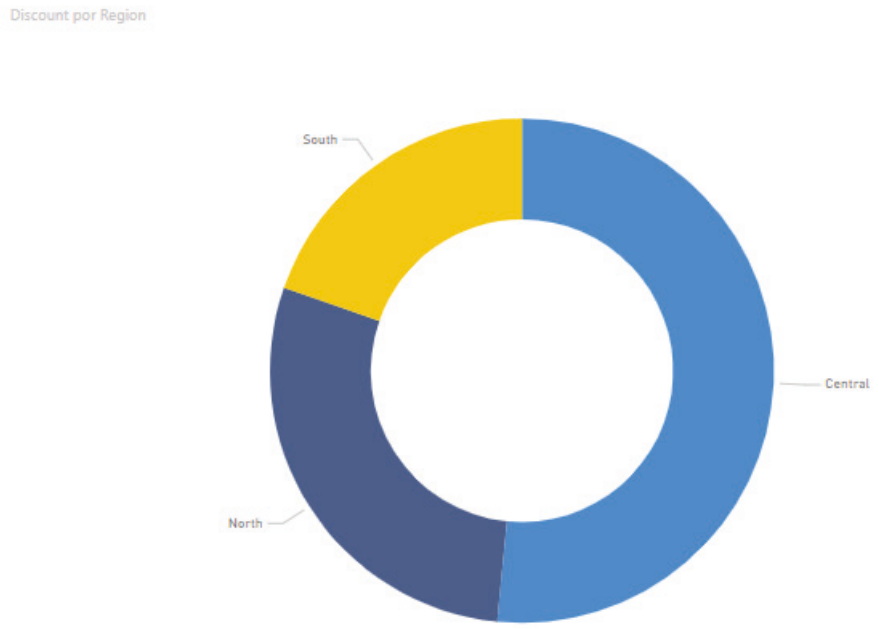



Imagen 29. Gráfico de anillos.

## GRÁFICO DE EMBUDO

Imagen 30. Gráfico de embudo.

## GRÁFICO DE MEDIDOR

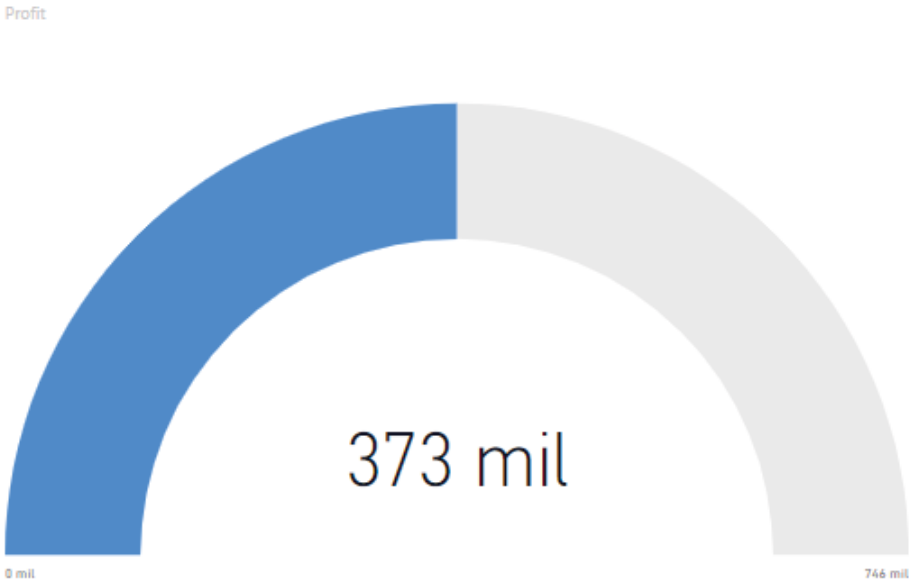



Imagen 31. Gráfico medidor.

## GRÁFICO DE INFLUENCIADORES CLAVE

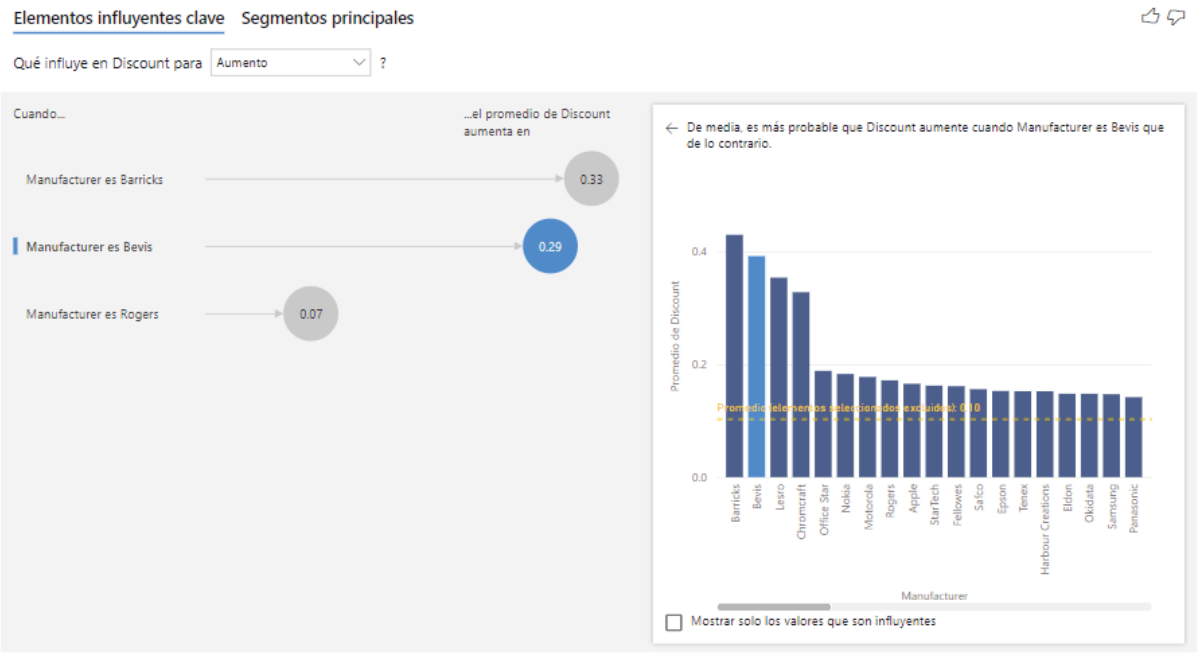



Imagen 32. Gráfico de influenciadores.

## GRÁFICO DE LÍNEAS

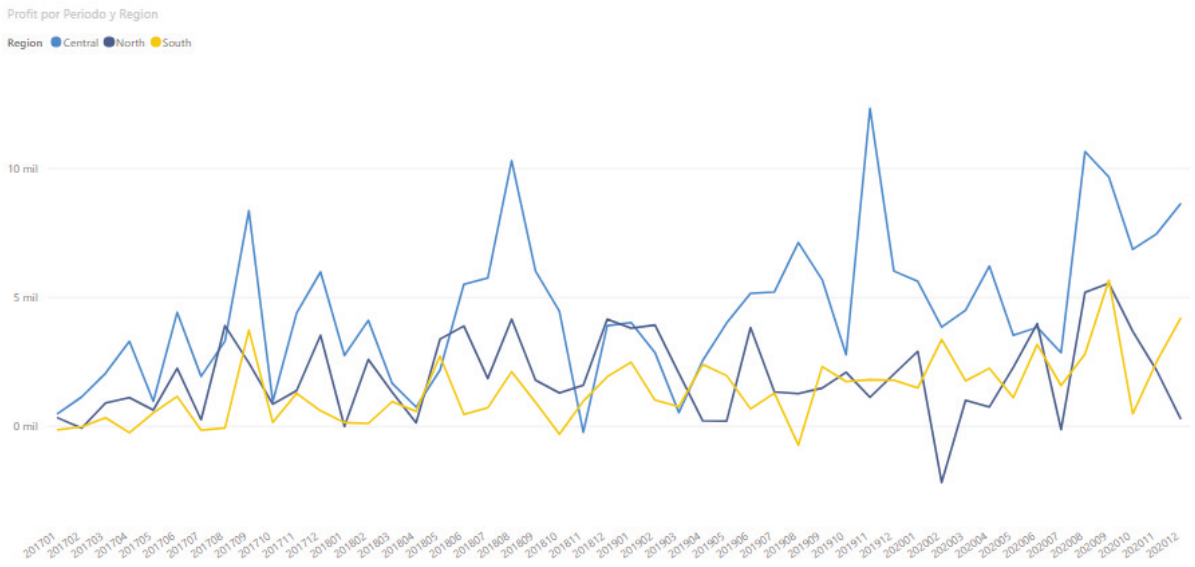



Imagen 33. Gráfico de líneas.

## MAPAS DE BURBUJAS

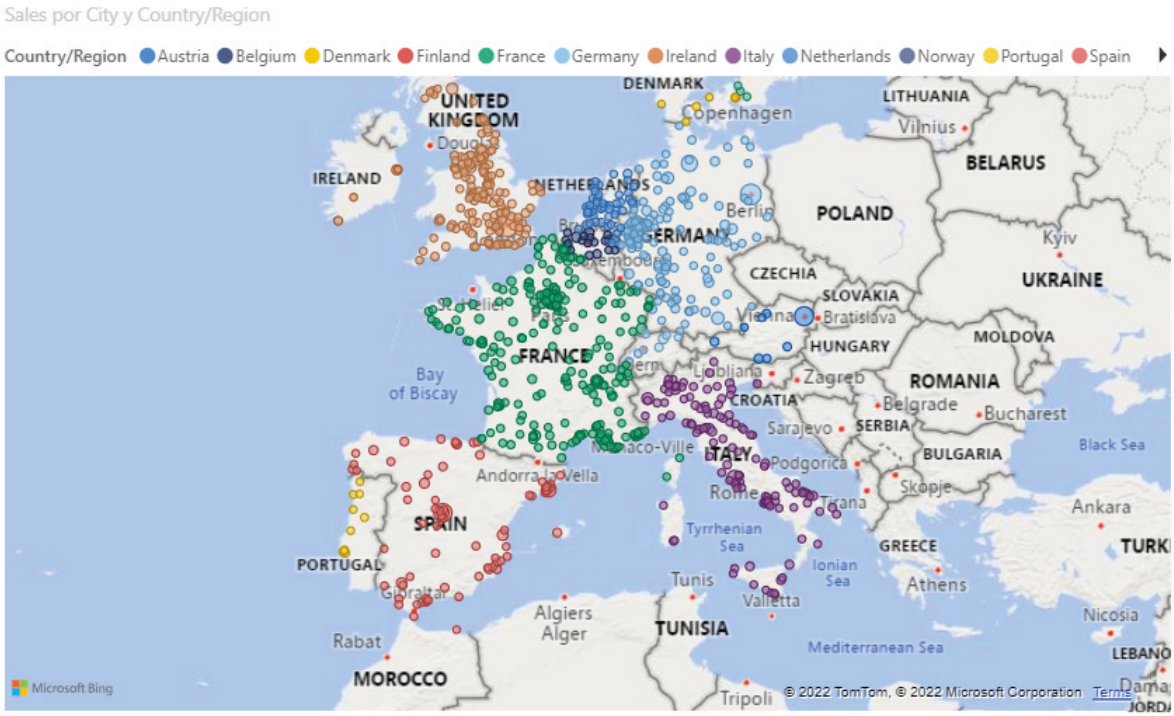



Imagen 34. Mapa de burbujas.

## MAPA COROPLÉTICO

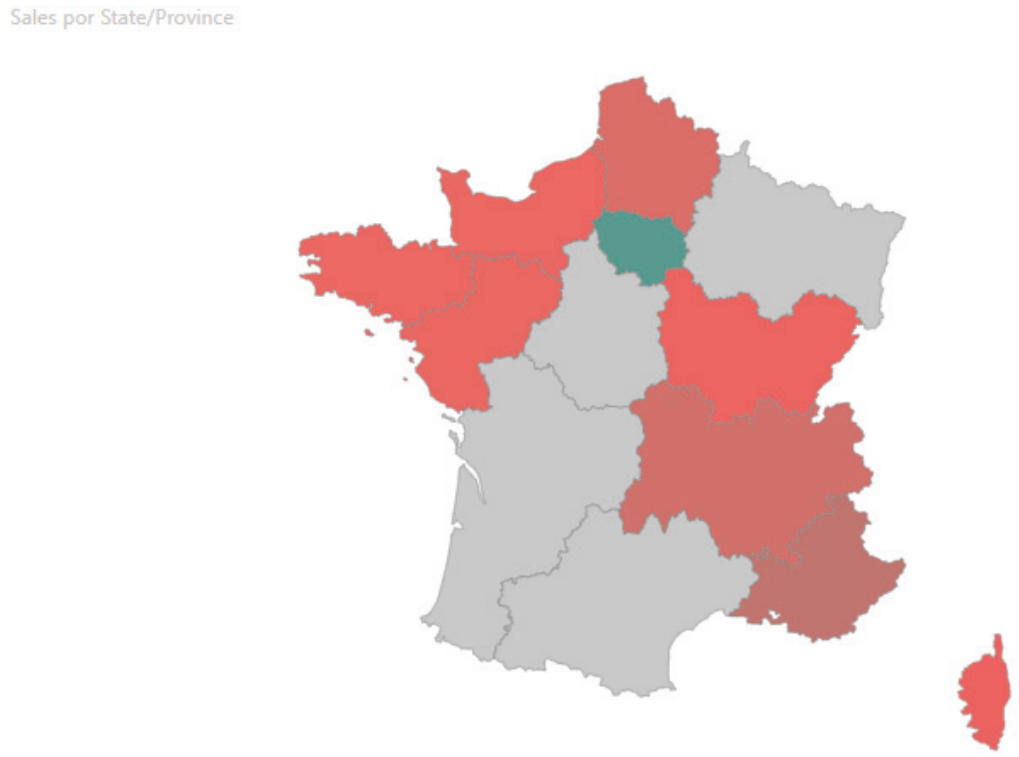



Imagen 35. Mapa coroplético.

## GRÁFICO DE DISPERSIÓN

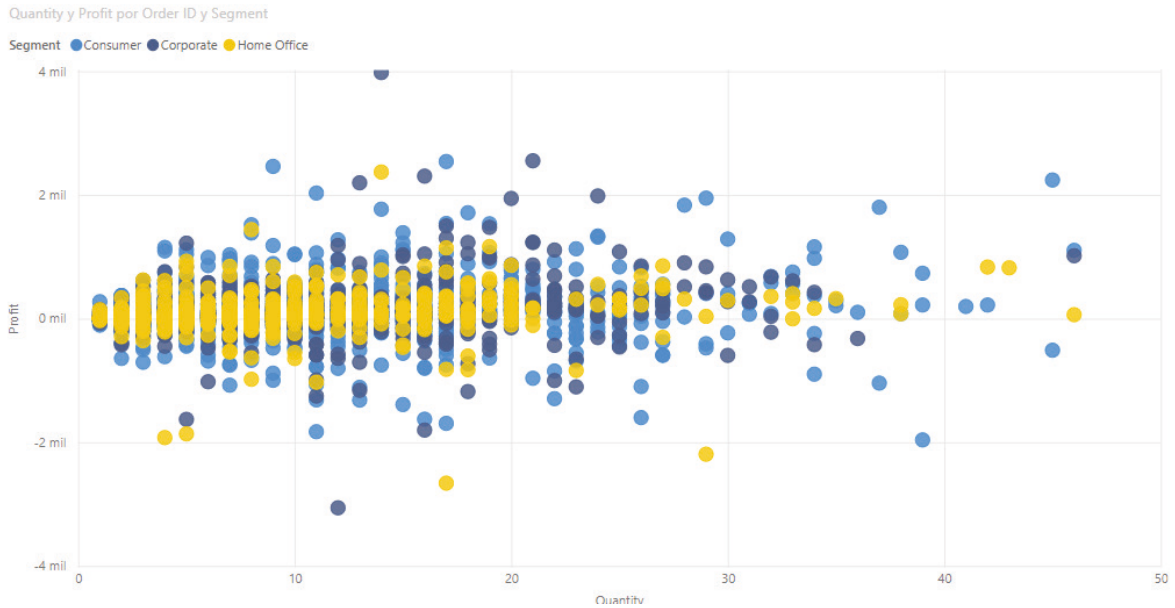



Imagen 36. Gráfico de dispersión.

## TABLA

| Country/Region | Central | North | South | **Total** |
|---|---|---|---|---|
| Austria | 81168 | | | **81168** |
| Belgium | 49233 | | | **49233** |
| Denmark | | 8643 | | **8643** |
| Finland | | 20702 | | **20702** |
| France | 858930 | | | **858930** |
| Germany | 628857 | | | **628857** |
| Ireland | | 16638 | | **16638** |
| Italy | | | 289717 | **289717** |
| Netherlands | 77520 | | | **77520** |
| Norway | | 20529 | | **20529** |
| Portugal | | | 15106 | **15106** |
| Spain | | | 287162 | **287162** |
| Sweden | | 30490 | | **30490** |
| Switzerland | 24874 | | | **24874** |
| United Kingdom | | 528570 | | **528570** |
| **Total** | **1720582** | **625572** | **591985** | **2938139** |

Imagen 37. Tabla.

## GRÁFICO DE CINTA

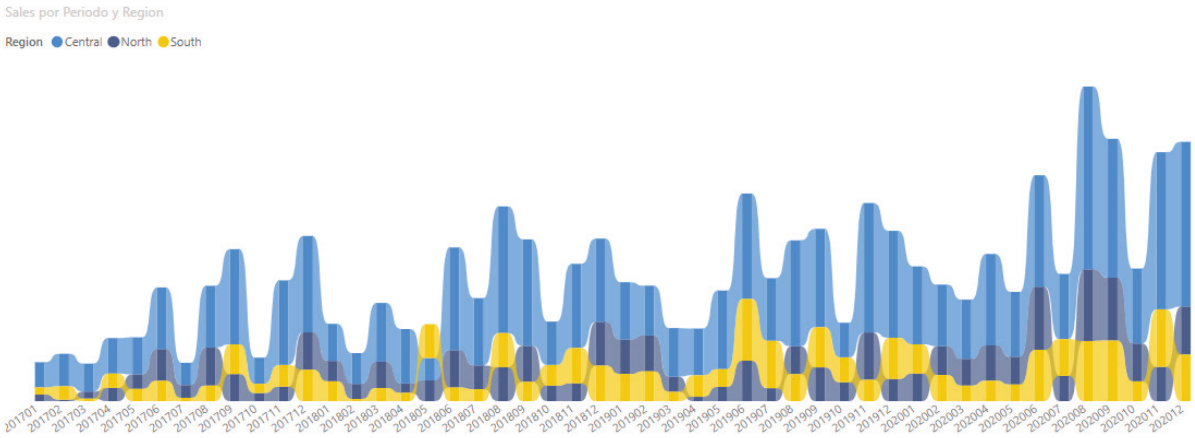



Imagen 38. Gráfico de cinta.

### GRÁFICO DE RECTÁNGULO

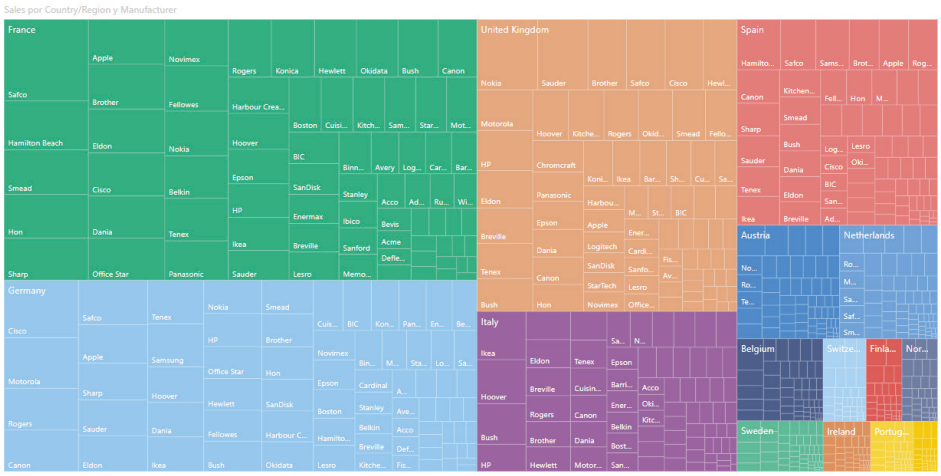



Imagen 39. Gráfico de rectángulos.

## 4.3. CASO PRÁCTICO DE ETL

En este caso práctico vamos a crear un *dashboard* para visualizar el precio de combustible en España. Para ello, nos basaremos en la información que nos proporciona la plataforma de datos abiertos del gobierno de España. Esta información se actualiza diariamente. Desde el *link* https://datos.gob.es/es/catalogo/e05068001-precio-de-carburantes-en-las-gasolineras-espanolas podemos ver todos los tipos de formatos y archivos disponibles. En nuestro caso, nos quedaremos con el formato XLS del fichero Precio de carburantes en gasolineras españolas. Necesitamos la URL del fichero, de forma que hacemos clic con el botón principal del ratón en Copiar dirección de enlace.

Los datos y enlaces que aparecen en la siguiente captura pueden variar de la información que aparezca en el momento de la consulta.

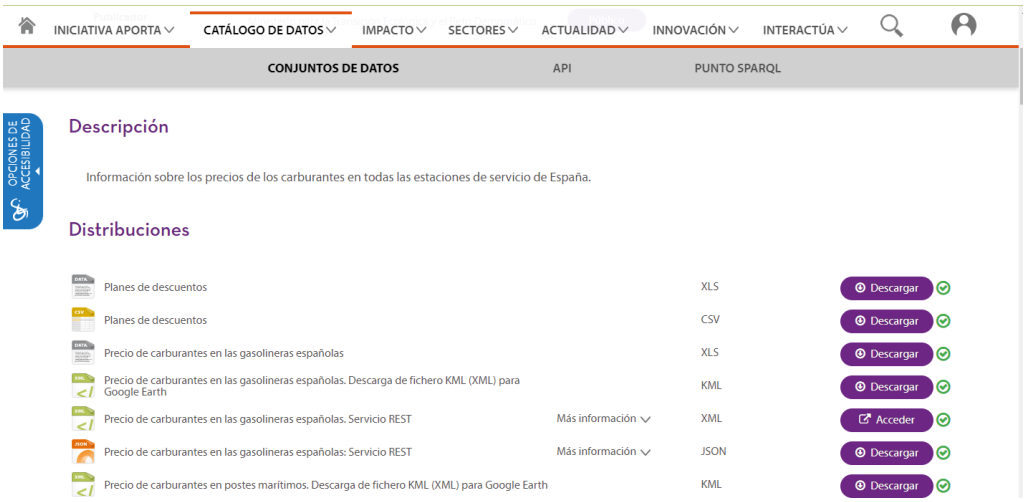



Imagen 40. Plataforma de datos.

A continuación, ejecutaremos Power BI. Haremos clic en el enlace de Obtener datos que vemos en la pantalla de bienvenida, buscaremos el conector web, haremos clic en Conectar y, en la pantalla que se muestra, pegaremos la URL (con control + V). Comprobaremos que el fichero en cuestión es un libro Excel, pues la URL termina con el nombre y extensión del fichero: /preciosEESS_es.xls

Ya en el editor de consultas (Power Query) nos encontraremos con la siguiente tabla:

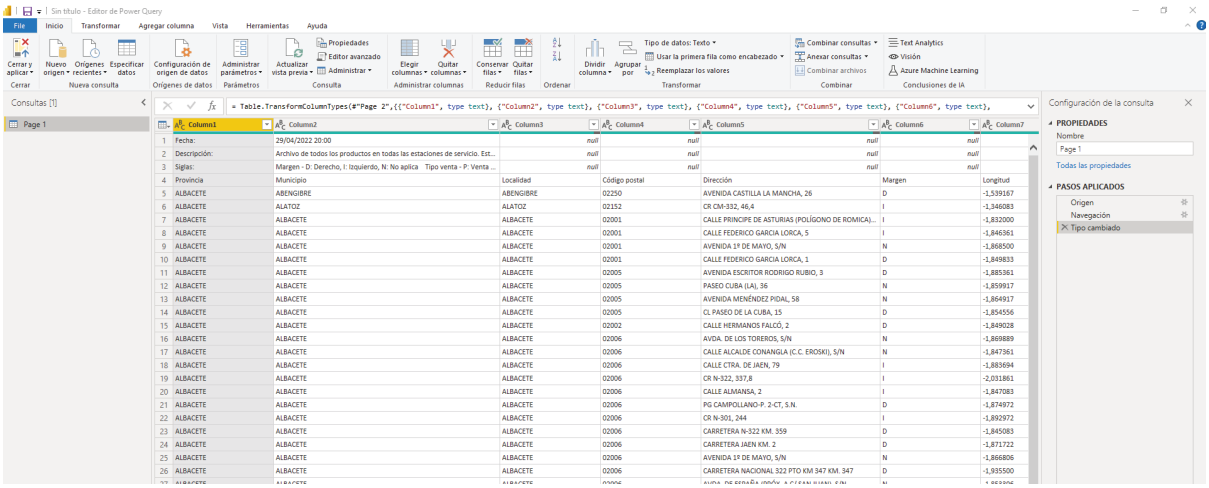


Imagen 41. Power Query.

Deberemos “limpiar” los datos de esta tabla siguiendo los siguientes pasos:

- Tal y como vemos en la primera fila, aparece la última fecha de actualización. Para conservar este dato tenemos que duplicar la página (botón derecho sobre la pestaña, duplicar) y eliminar el resto del contenido.

  Para eliminar una fila, seleccionamos la primera fila y, luego, en el panel superior, desde Inicio, en el apartado de reducir filas, seleccionamos Conservar filas/Conservar filas superiores indicando 1, ya que solo queremos conservar la primera fila.

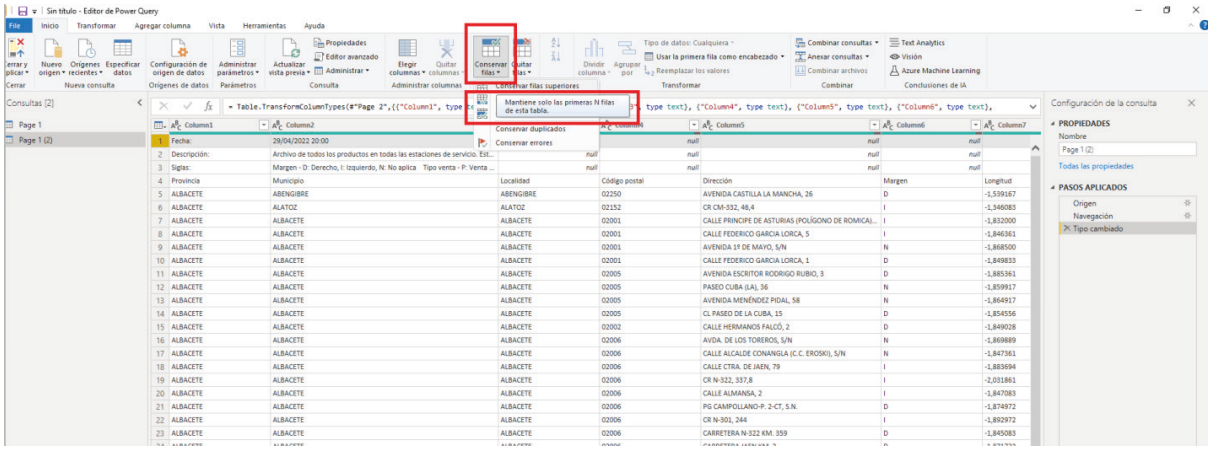


Imagen 42. Conservar filas superiores.

Respecto a las columnas, tenemos que quedarnos solo con las dos primeras. Para ello, hacemos clic en la fila que queremos mantener. En este caso, como son dos, mantenemos la tecla Ctrl del teclado para mantener la segunda, hacemos clic con el botón derecho sobre la columna y seleccionamos Quitar otras columnas.

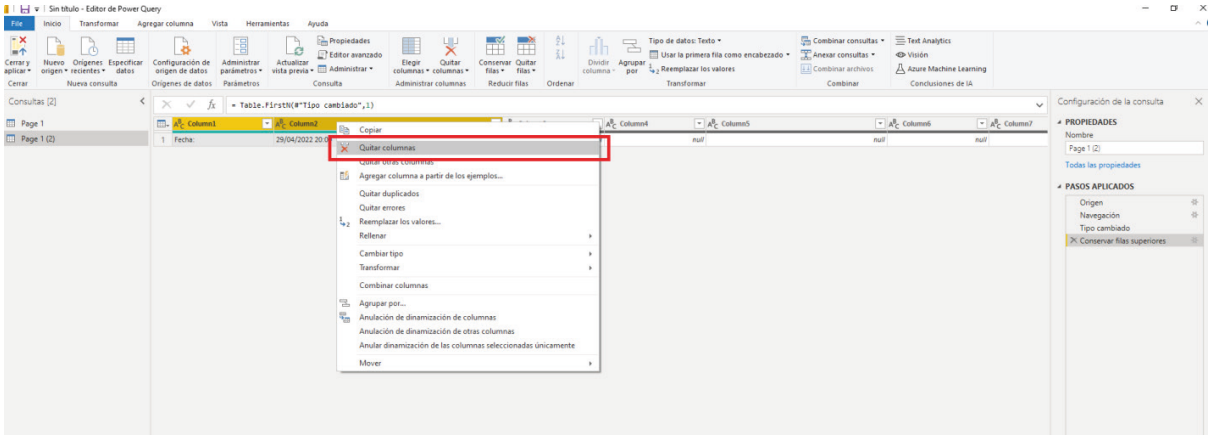



Imagen 43. Quitar otras columnas.

Tras este paso, nos quedarán dos columnas. El siguiente paso sería transponer esas columnas. Para ello, desde la barra de navegación superior, desde Transformar, hacemos clic en Transponer.

El siguiente paso es promover la primera fila como encabezado. Para ello, desde el panel de navegación Inicio, seleccionamos en el apartado Usar la primera fila como encabezado de Transformar.

Por último, vamos a renombrar esta página haciendo botón derecho sobre la página Cambiar nombre.

**+ INFO**

Si hacemos doble clic sobre la pestaña, podemos cambiarle el nombre.

**IMPORTANTE**

Tal y como podemos observar en el panel derecho, a medida que hemos realizado las modificaciones, estas aparecen registradas. Haciendo clic en ellas podemos navegar hacia ese paso. Desde ese punto que hemos seleccionado podemos insertar otro paso; no obstante, eso puede perjudicar los siguientes pasos, por lo que se desaconseja realizar esta práctica si no se tiene mucho manejo en la herramienta.

- En la pestaña original, realizaremos los siguientes pasos:
  - Eliminar las tres primeras columnas desde el panel de navegación Inicio, Reducir filas, Quitar filas superiores, seleccionar 3.
  - Usar primera fila como encabezado, tal y como vimos anteriormente.
  - De los campos que contienen precios, eliminaremos todos salvo los correspondientes a la gasolina 95 E5, gasolina 98 E5, gasóleo A y gasóleo B. Se sugiere cambiar los nombres de las cuatro columnas para que solo muestren el nombre del producto, y no la palabra "precio":

| Gasolina 95 E5 | Gasolina 98 E5 | Gasóleo A | Gasóleo B | |
|---|---|---|---|---|
| 1779 | null | 1839 | 1330 | Nº |
| 1849 | 2000 | 1890 | null | REI |
| 1869 | 2011 | 1939 | null | BP |
| null | null | 1929 | null | CEI |
| 1839 | 1939 | 1889 | null | CAI |
| 1737 | null | 1777 | null | PLE |
| 1788 | null | 1798 | null | A& |

Imagen 44. Columnas Precio.

  - A continuación, deberemos despivotar estas cuatro columnas de precios para dejarlas en dos columnas: una con la etiqueta del producto correspondiente y otra con el precio. Para ello, seleccionamos las columnas, hacemos clic con el botón derecho sobre una columna y seleccionamos Anulación de dinamización de columnas.
  - Una vez despivotadas, cambiaremos las etiquetas de las dos columnas resultantes a Producto y Precio. Asigna al campo Precio un tipo Número decimal fijo (equivalente a formato de moneda). Cambiaremos el tipo de formato desde la configuración regional haciendo clic en el símbolo junto al nombre la columna.

    Repetiremos el mismo proceso de cambio de tipo para los campos de Latitud y Longitud.
  - En el caso que aparezca los datos sin decimales, tenemos que crear una nueva columna dividiendo la columna Precio recién renombrada /1000. Para ello, desde el panel superior en Agregar columna, Columna personalizada.

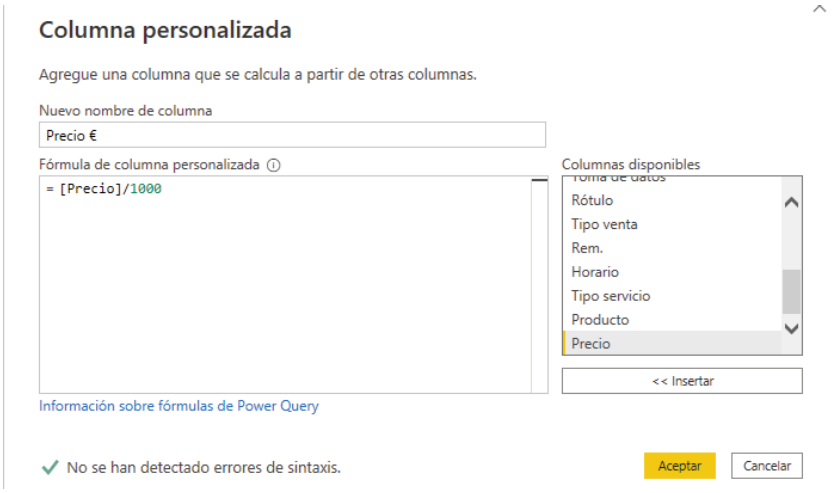



Imagen 45. Creación de columna.

Finalmente, eliminamos la columna Precio.

- Podemos eliminar también las columnas Margen, Tipo venta, Rem. y Tipo servicio.
- Cambiaremos el nombre del campo Rótulo por Proveedor.
- Por último, cambiamos el nombre de esta consulta de Page 1 a Estaciones.

- Creación de logos. Sin salir del editor de consultas, nos conectamos al segundo fichero. Para ello, hacemos clic en Nuevo origen (en la barra de herramientas), volvemos a escoger el conector web e introducimos la siguiente URL: https://raw.githubusercontent.com/salvadorfergo/Logos_Gasolineras/main/estaciones.csv

- Escogemos la configuración adecuada para la lectura del fichero (delimitador, básicamente) y confirmamos la lectura haciendo clic en el botón Aceptar.

  Ahora deberemos limpiar esta tabla, asegurándonos de que los nombres de los campos son los adecuados.

  Comprobamos que esta tabla solo contiene los logos de siete proveedores (aquellos con mayor número de estaciones de servicio en España). Cambiemos el nombre de esta tabla a Logos.

  Tras esto, hacemos clic en guardar y cerrar.

- Una vez se han cargado las tablas, desde el panel de navegación izquierdo seleccionamos el segundo botón de Datos y seleccionamos la tabla Logos para seleccionar la columna Logo y en el panel superior desde Herramientas de columnas en Categoría de datos modificar a Dirección URL de la imagen.

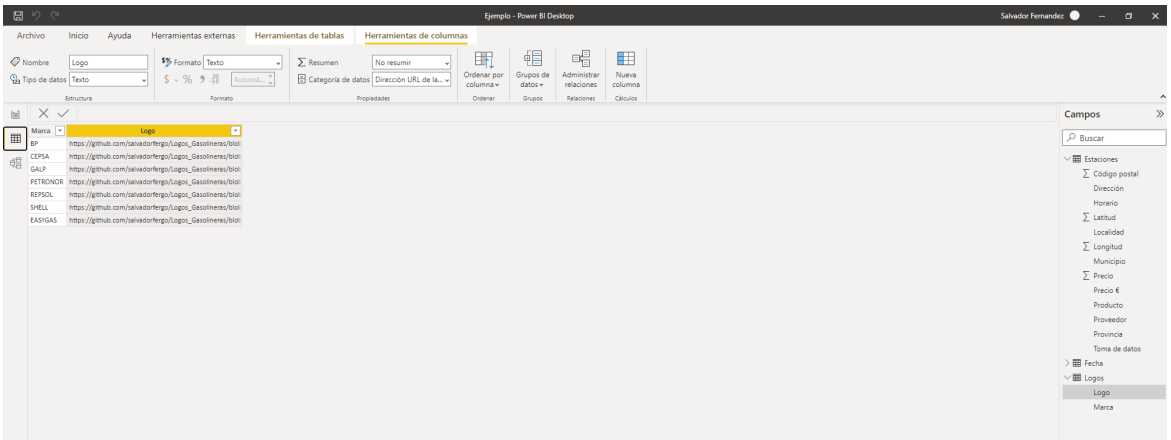



Imagen 46. URL.

- Finalmente, desde el apartado de Modelo del panel de navegación izquierdo, relacionamos el campo Proveedor de la tabla Estaciones con el campo Marca de la tabla Logos arrastrando ambos campos.

  Si hacemos clic en la relación, podemos ver qué tipo de relación se ha creado de forma automática.

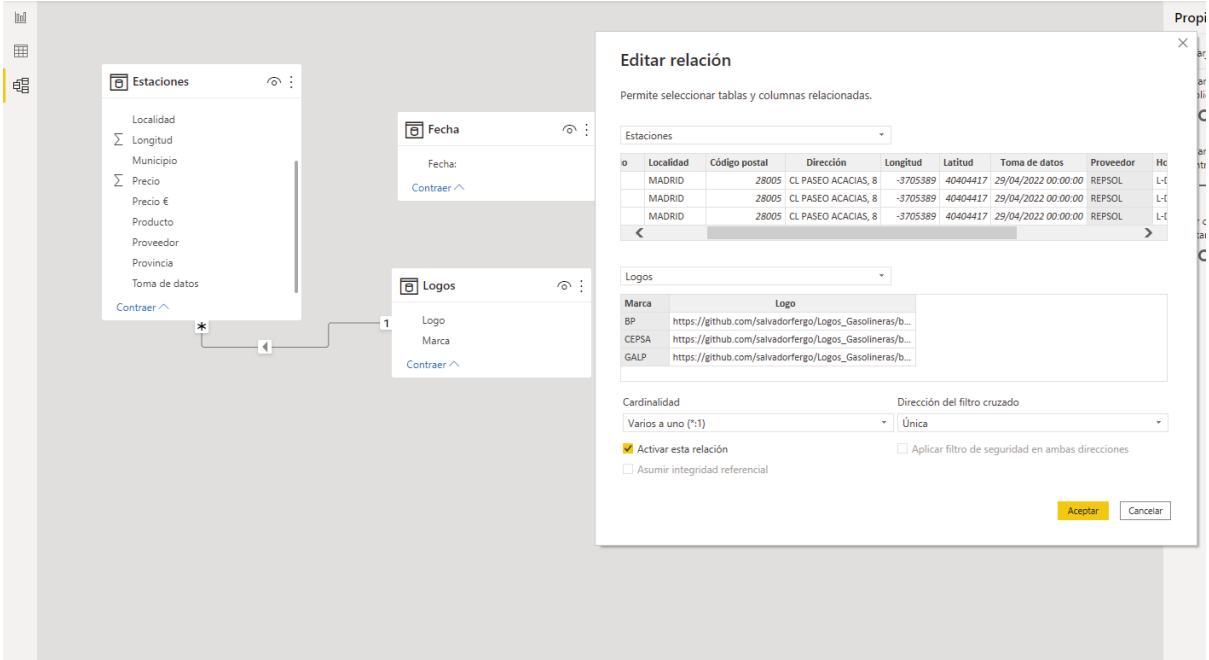



Imagen 47. Modelo de datos.

## 4.4. CASO PRÁCTICO DE CONSTRUCCIÓN DE *DASHBOARD*

Tras ver el desarrollo ETL, a continuación vamos a ponerlo en práctica.

Antes de crear un *dashboard*, se recomienda crear un *mock up* en papel o en PowerPoint.

Imagen 48. Mock up.

De esta forma, tendremos las ideas más claras a la hora de elaborar nuestro *dashboard*.

Tal y como se ha comentado, a la hora de crear un gráfico tenemos que seleccionar el tipo de gráfico en el panel de gráficos y arrastrarlo a la hoja de trabajo.

1. **Comenzaremos con el seleccionador de logos.** Para ello, seleccionamos un segmentador de datos. Con el visual ya en la hoja de trabajo desde el panel de edición en el Configurador de la segmentación, seleccionamos Horizontal.

Imagen 49. Segmentador de datos.

2. **Campo de última fecha seleccionada.** Seleccionamos el campo Fecha: haciendo doble clic. Le editamos el texto desde el apartado Valores y Encabezados de columna en el panel de navegación.

3. **Mapa.** Seleccionamos la visualización Mapa e insertamos los campos Latitud y Longitud. Como leyenda, seleccionamos el campo Proveedor.

**IMPORTANTE**

Previamente tenemos que cambiar el tipo de dato para Longitud y Latitud desde el apartado del panel de navegación lateral izquierdo Datos.

Ajustamos el tamaño de las burbujas desde el panel de edición, apartado Burbujas.

4. **Gráfico de barras precio por producto.** Seleccionamos el visual Gráfico de barras agrupadas, como eje X vamos a utilizar la columna Producto. Como eje Y vamos a crear la medida AVG_Precio desde la tabla de Estaciones, seleccionamos en la parte derecha los tres puntos y hacemos clic en Nueva medida.

```
1 AVG_Precio = VAR ProveedorEASYGAS = SELECTEDVALUE(Estaciones[Producto])
2     VAR AVG_PRICE = AVERAGE(Estaciones[Precio])-0.2
3     VAR Result = IF(ProveedorEASYGAS ="EASYGAS", AVG_PRICE*0.8, AVG_PRICE)
4     RETURN Result
```

Imagen 50. Variable AVG_Precio.

5. **Segmentador de provincia y municipio:** este segmentador nos permite ir bajando de niveles y utilizarlo como filtro.

6. **Tabla detalles:** mediante esta tabla podemos visualizar el detalle de proveedor, localidad y precio medio. Además, vamos a añadir una barra condicional en la columna de AVG_ Precio. Para ello, desde el apartado de selección de visuales, hacemos clic en el campo AVG_Precio en la pestaña y seleccionamos Formato condicional y Barra de datos.

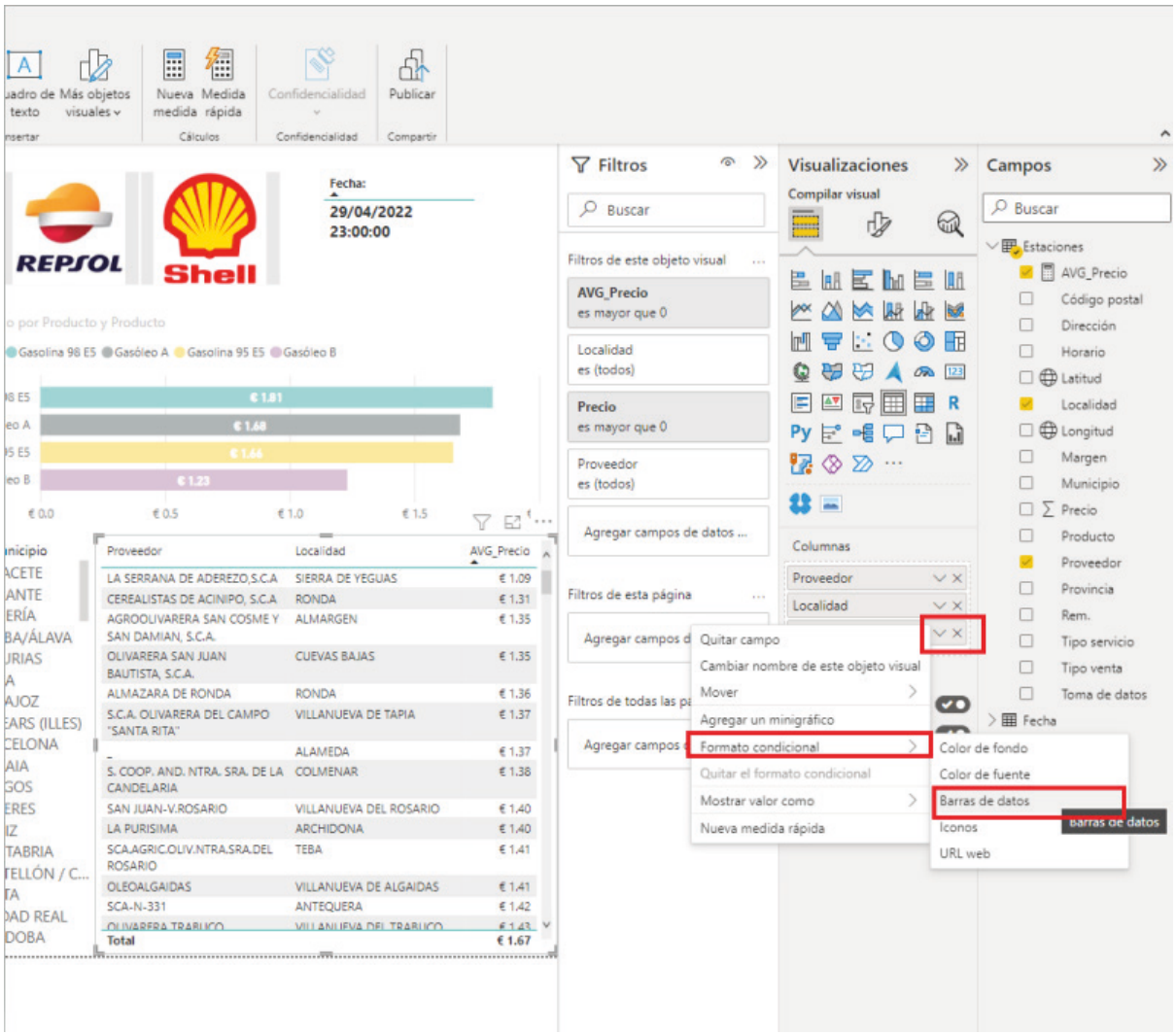



Imagen 51. Formato condicional.

7. **Insertar *tooltip*:** el *tooltip* en el mapa nos dará conocimiento del punto geolocalizado. Para ello, tenemos que crear una nueva página y, desde el panel superior de diseño, hacemos clic en Configuración del lienzo, y en tipo seleccionamos Información sobre herramienta. Además, desde el apartado de Información de la página marcamos la opción Permitir el uso como información sobre herramientas.

   Ahora comenzamos añadir la información que creamos oportuna, en este caso, mediante el visual Tarjeta nos muestra la información individualizada.

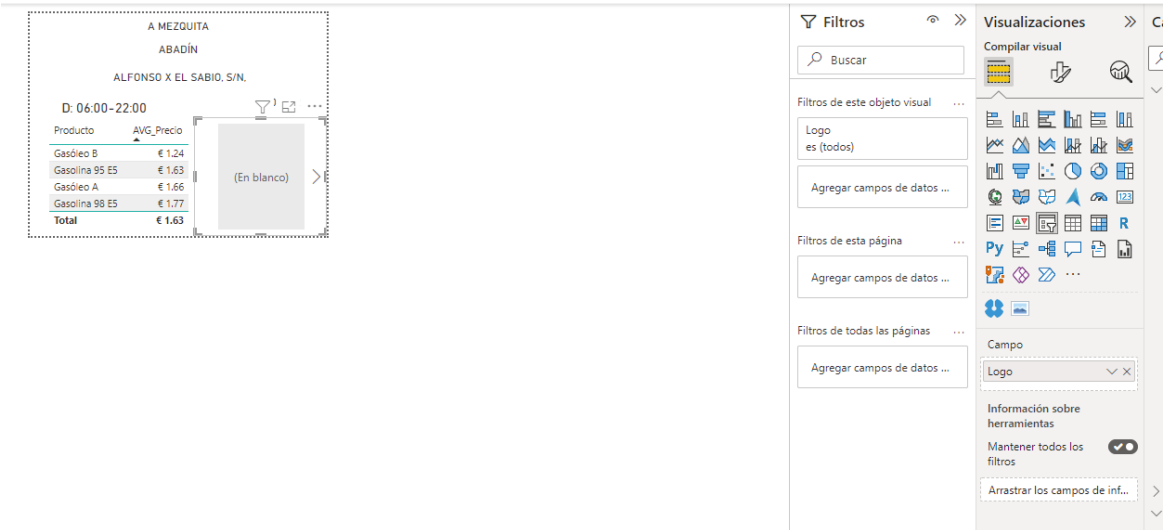



Imagen 52. *Tooltip.*

El siguiente paso es conectar nuestro mapa al *tooltip*. Para ello, hacemos clic en el mapa y desde el menú de Formato visual seleccionamos General y marcamos la opción Información sobre herramientas. En tipo seleccionamos Página del informe y en página seleccionamos la página que hemos creado con el *tooltip*.

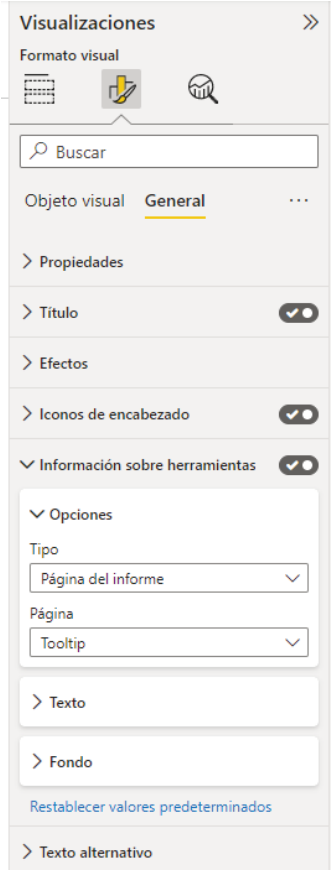



Imagen 53. Mapa tooltip.

Con esto ya tendríamos nuestro *dashboard*, que podemos actualizar haciendo clic en el botón superior de Actualizar. Si hay información actualizada, cambiará la fecha de actualización.

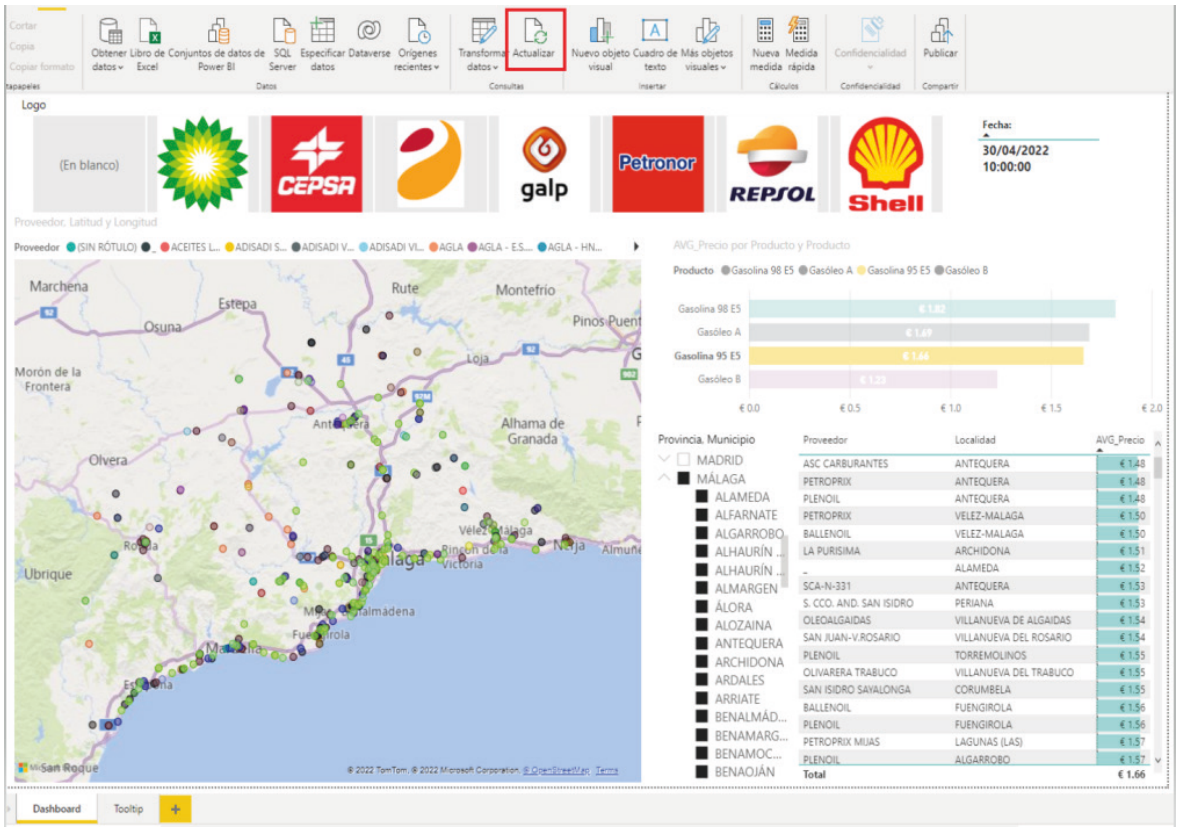



Imagen 54. Actualización *dashboard*.

Por último, publicaremos el *dashboard* desde el panel superior Inicio y Publicar. Guardamos los cambios, seleccionamos Mi área de trabajo y hacemos clic en Seleccionar.

Veremos el siguiente mensaje:

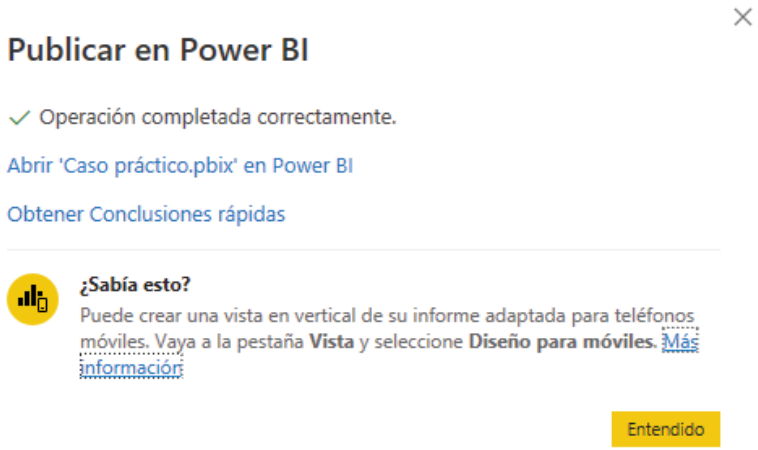



Imagen 55. Publicación Power BI.

Hacemos clic en Abrir ‘Caso práctico.pbix’ en Power BI.

A continuación, veremos nuestro *reporte* publicado en Power BI Service.

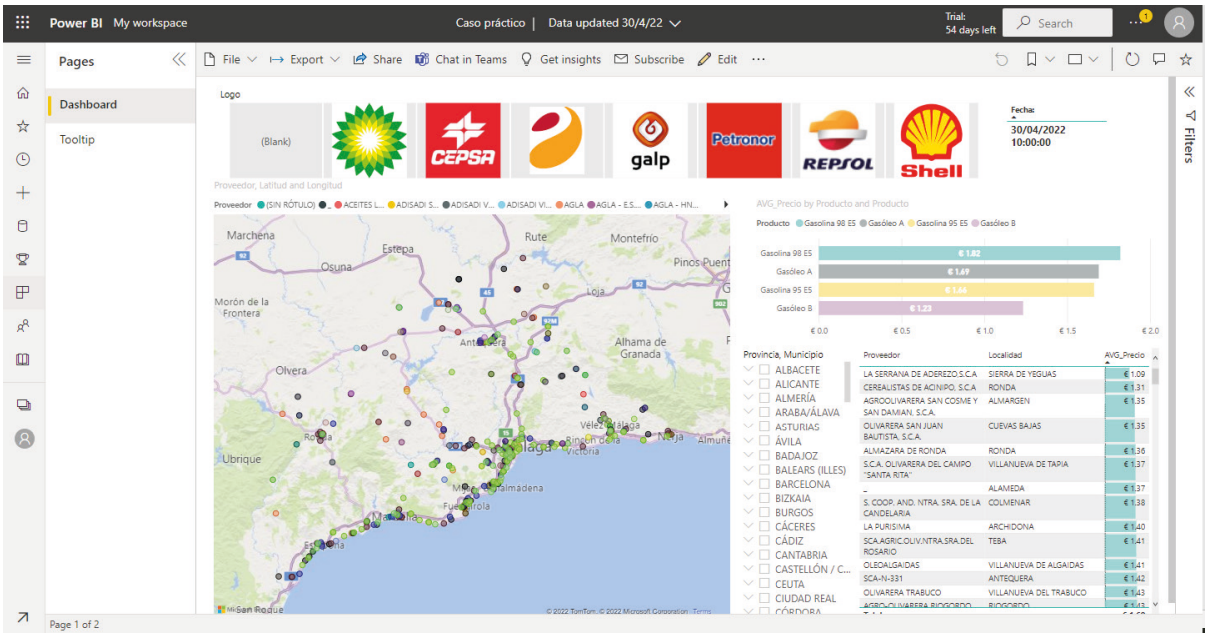



Imagen 56. Power BI Service.

Desde aquí podemos compartir nuestro *dashboard* o incrustarlo en una página web.

# IDEAS CLAVE

- Microsoft Power BI Desktop es una aplicación local que permite desarrollar el flujo completo de un análisis de datos, desde el proceso ETL hasta su publicación.
- Microsoft Power BI Service es un entorno *cloud* que permite editar *dashboard* creados en Power BI Desktop y distribuirlo en diferentes niveles.
- El ecosistema Power BI permite automatizar el proceso de actualización y distribución de un *dashboard.*

# GLOSARIO

- **GitHub.** Entorno cloud destinado a desarrolladores que le ayuda a almacenar y administrar su código. También sirve para almacenar datos a pequeña escala.
- **Mock-up.** Creación de un montaje del producto final de forma esquemática y resumida.
- **Power Query.** Motor de preparación y transformación de datos mediante una interfaz gráfica, disponible en varios productos de Microsoft.
- **Proceso ETL.** Conjunto de acciones que permiten extraer datos de diversas fuentes para, una vez limpiados, guardarlos de forma estandarizada en una base de datos nueva más fácil de manipular.

# BIBLIOGRAFÍA

MdCloud. (s. f.). *Procesos ETL: definición, desarrollo y aplicaciones*. MdCloud. https://blog.mdcloud.es/procesos-etl-definicion-desarrollo-aplicaciones/

Microsoft. (s. f.). *¿Qué es el servicio Power BI?* Microsoft Build. https://docs.microsoft.com/es-es/power-bi/fundamentals/power-bi-service-overview

Microsoft. (s. f.). *Format. Microsoft Build*. https://docs.microsoft.com/en-us/dax/format-function-dax

Microsoft. (s. f.). *Orígenes de datos en Power BI Desktop*. Microsoft Build. https://docs.microsoft.com/es-es/power-bi/connect-data/desktop-data-sources

Microsoft. (s.f.). *Tipos de visualización en Power BI.* Microsoft Build. https://docs.microsoft.com/es-es/power-bi/visuals/power-bi-visualization-types-for-reports-and-q-and-a

# TEMA 5

## Google Data Studio

# OBJETIVOS

- Comprender el uso de Google Data Studio.
- Conocer los niveles de información de datos en Google Data Studio.
- Conocer los beneficios de las estadísticas de datos que aporta Google Data Studio.
- Conocer algunas desventajas que presenta Google Data Studio.
- Conocer las opciones y principales funcionalidades de Google Data Studio.
- Saber cómo agregar conexiones de datos en Google Data Studio.
- Conocer los tipos de gráficos que se pueden realizar con Google Data Studio.
- Conocer los tipos de fuentes de datos que permite Google Data Studio.
- Aprender diferentes funcionalidades hasta la generación del informe.
- Conocer las opciones disponibles para los informes en Google Data Studio.

# 5.1. ¿QUÉ ES GOOGLE DATA STUDIO?

Google Data Studio es un programa gratuito de Google que permite crear paneles e informes dinámicos. Los informes creados por esta la plataforma son fáciles de entender, compartir y personalizar.

El análisis web siempre ha sido un desafío para muchos especialistas en marketing digital. No solo a nivel interno para el propio equipo de trabajo, sino además para el usuario.

Realizar informes profesionales que respalden los datos recopilados es en realidad una labor compleja. Muchas veces, crear un informe significa descargar minuciosamente datos de Google Analytics y llevarlos a una hoja de cálculo de Excel para crear gráficos, etc. Y en otros casos, se pierde mucho tiempo capturando gráficos y tablas de datos.

Con la herramienta Google Data Studio, Google nos permite crear informes profesionales, muy completos y bastante fáciles de entender.

Google Data Studio ayuda a crear informes y paneles personalizados utilizando datos de las diferentes herramientas de Google y otras fuentes externas.

El buscador ha lanzado al mercado Data Studio 360 Payment Tool, una plataforma de informes y visualización de datos enfocada más a empresas. Data Studio es un software gratuito y mejorado respecto a Data Studio 360.

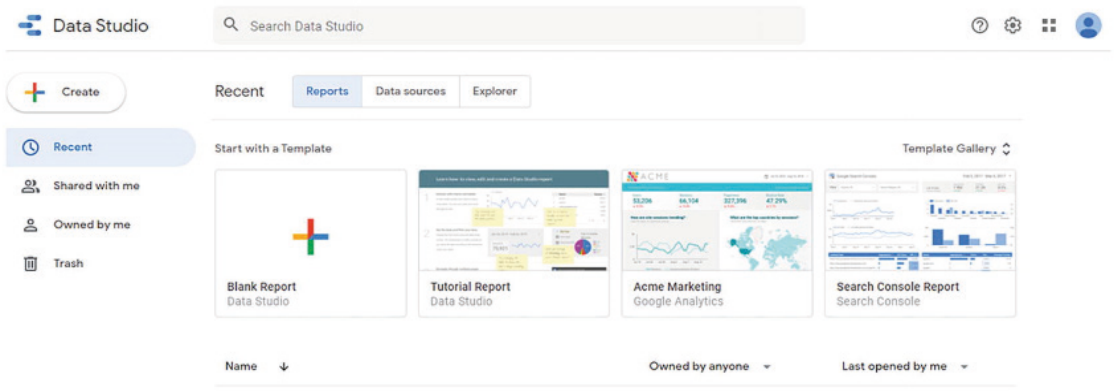



Imagen 1. Interfaz de Data Studio. Fuente: Data Studio.

Beneficios clave de las estadísticas de datos:

- Conectar fácilmente a la plataforma múltiples fuentes de datos.
- Transformar los datos para presentarlos de forma amigable.
- Ahorrar tiempo y recursos al crear informes.
- Customizar los gráficos.
- Actualizar los datos en tiempo real.

- Generar informes de forma automática.
- Editar presentaciones de informes y compartirlas.
- Trabajar en equipo desde diferentes ordenadores de forma simultánea en un mismo informe.

Por el contrario, también podemos señalar las principales desventajas de Data Studio. Y es que, aunque se trata de un programa muy poderoso, especialmente considerando que es gratuito, la plataforma tiene ciertos inconvenientes que seguramente tenderán a desaparecer:

- Solo permite conectar datos de sus propias fuentes o por medio de bases de datos externas. Sería interesante que permitiera la recogida de información por medio de otras fuentes como Facebook.
- Su interfaz no se encuentra optimizada para dispositivos móviles.
- No permite calcular métricas con diferentes fuentes de datos, es decir, no se pueden combinar entre sí.

## 5.2. CARACTERÍSTICAS DE LA HERRAMIENTA E INTERFAZ

Google Data Studio es una herramienta cómoda y visual. Presenta una interfaz sencilla con dos tipos de elementos:

- **Informes**. Con los *dashboards* disponibles o los nuevos que se vayan a crear. Solo hay que conectar los datos a su fuente para que en el informe se actualicen directamente cuando haya variaciones.
- **Fuentes de datos**. Incluye las conexiones con datos que se encuentran habilitadas para ser usadas por los *dashboards*. Las conexiones son los elementos más importantes, ya que proporcionan los datos.

Con Data Studio se pueden crear informes sobre datos de una variedad de fuentes sin necesidad de escribir código.

Con Información de datos, se puede:

- Ver datos por gráficos y tablas con numerosas opciones de personalización.
- Conectarse fácilmente a múltiples fuentes de datos.
- Compartir información con el equipo.
- Participar en informes colaborativos con el equipo.
- Acelerar la generación de informes con informes de muestra integrados.

## CONCEPTOS CLAVE DE GOOGLE DATA STUDIO

- **Informes**. La característica principal de esta herramienta es que permite crear informes totalmente personalizables. Estos incluyen un panel en el que se pueden ver métricas y gráficos que proporcionarán el cuerpo para el tablero final.

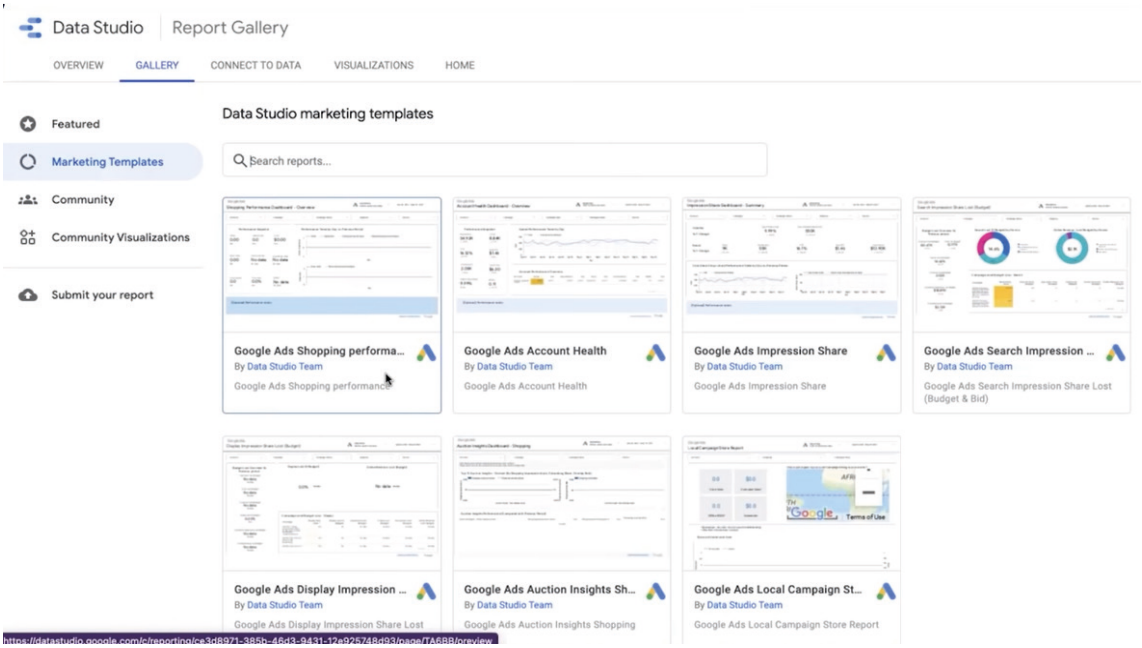



Imagen 2. Plantillas para informes. Fuente: Data Studio.

- **Fuentes de datos**. De ellas se obtienen los datos para generar informes. Se puede usar Google Analytics, Search Console, SQL, Google Sheets, Google Ads y más.

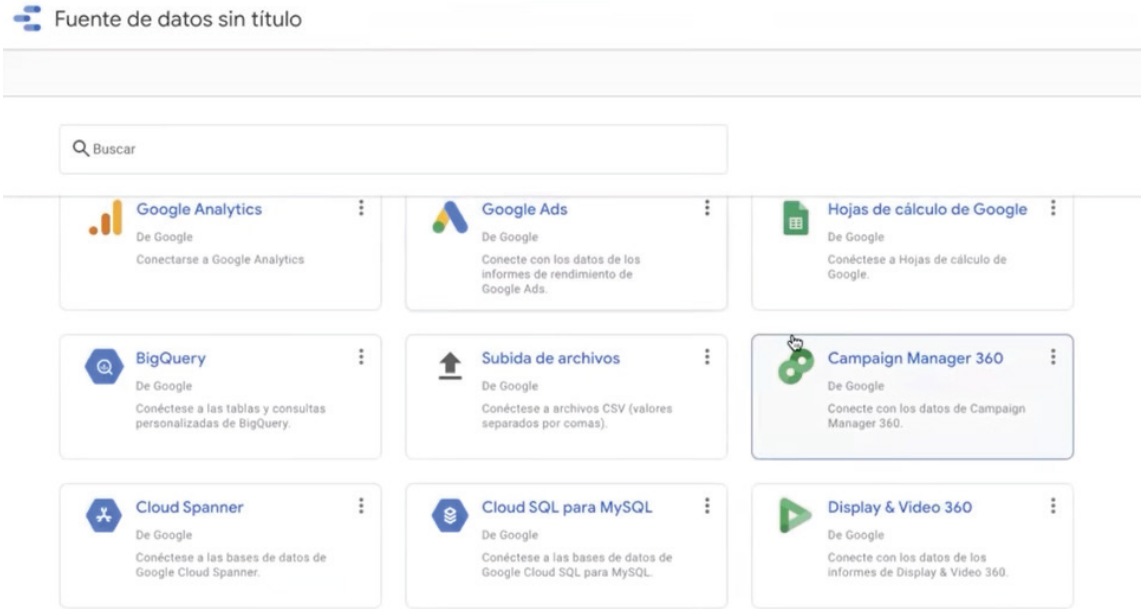



Imagen 3. Posibles fuentes de datos en Data Studio. Fuente: Data Studio.

- **Gráficos**. Cuando se haya creado un informe y se conecte a las fuentes de datos, solo hay que introducir los datos y las métricas en un gráfico. Para ello, se seleccionan las fuentes de datos que se quieran usar, con la opción "Crear fuentes de datos" y, con los datos cargados, se eligen los valores que se quieren usar para, finalmente, seleccionar "Añadir al informe".

  Pueden ser de diferentes tipos:

  – ***Scorecards* o tarjeta de resultados**. Representan el valor que tiene una métrica específica, aunque no permite desglosar por dimensión; normalmente se seleccionan las métricas clave del rendimiento de la web, datos como el total de ventas, la tasa media de rebote, etc.

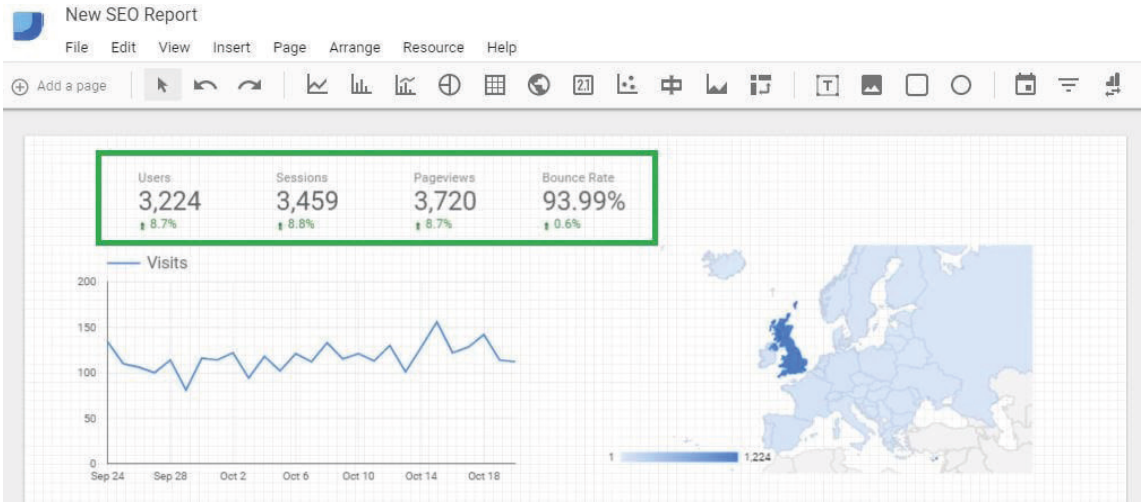



Imagen 4. Scorecard. Fuente: Data Studio.

  – **Mapas geográficos**. Cambios en datos o métricas basadas en áreas geográficas.

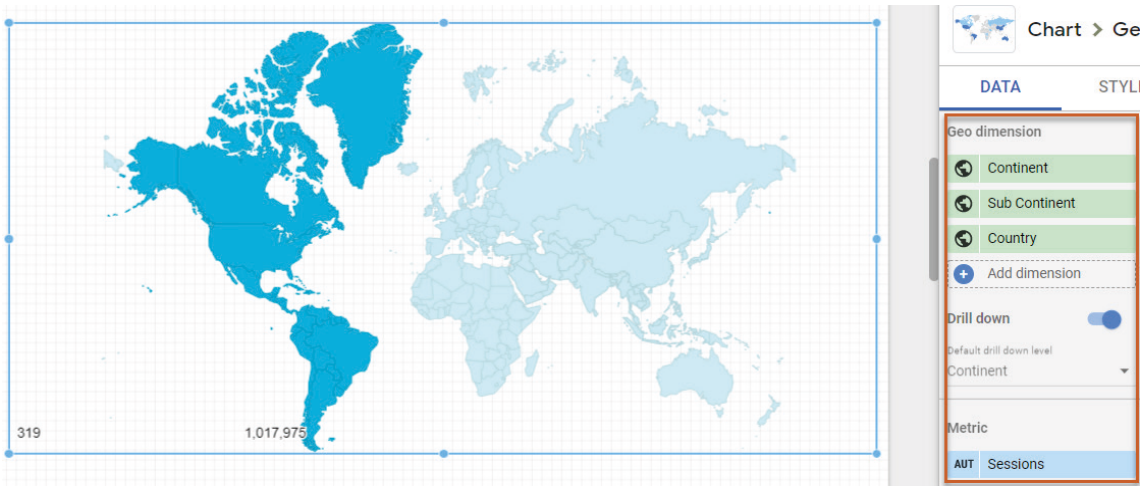



Imagen 5. Mapa según métricas. Fuente: Data Studio.

  – **Tabla dinámica**. Muestra los datos tanto en filas como en columnas, permitiendo emplear tantas métricas y dimensiones como se deseen; sin embargo, cuantas más se agreguen, más difícil resultara el análisis.

Cada columna es la representación de una cierta dimensión o métrica y cada fila incluye los datos registrados. De este modo, se pueden reorganizar los datos de una tabla al cambiar los de un segundo parámetro (por ejemplo, sexo, rebote, localización, compras).

Para realizar una tabla dinámica basta con seleccionar la opción Tabla dinámica del menú Ver, seleccionar la fuente del tráfico y seleccionar las métricas a usar.

| | | Year / Quarter / Revenue Per User | |
|---|---|---|---|
| | | | 2017 |
| Country | Gender | Q3 | Q4 |
| United States | male | $271.09 | $58.28 |
| | female | $158.54 | $22.04 |
| Hong Kong | male | - | $158.54 - |
| | female | - | - |
| Canada | male | $74.06 | $11.54 |
| | female | $75.52 | $9.95 |
| Venezuela | | - | - |
| Australia | female | - | $64.36 |
| | male | $55.75 | - |

Imagen 6. Tabla dinámica. Fuente: Data Studio.

– **Serie temporal**. Representa la evolución de los datos disponibles durante un período de tiempo específico.

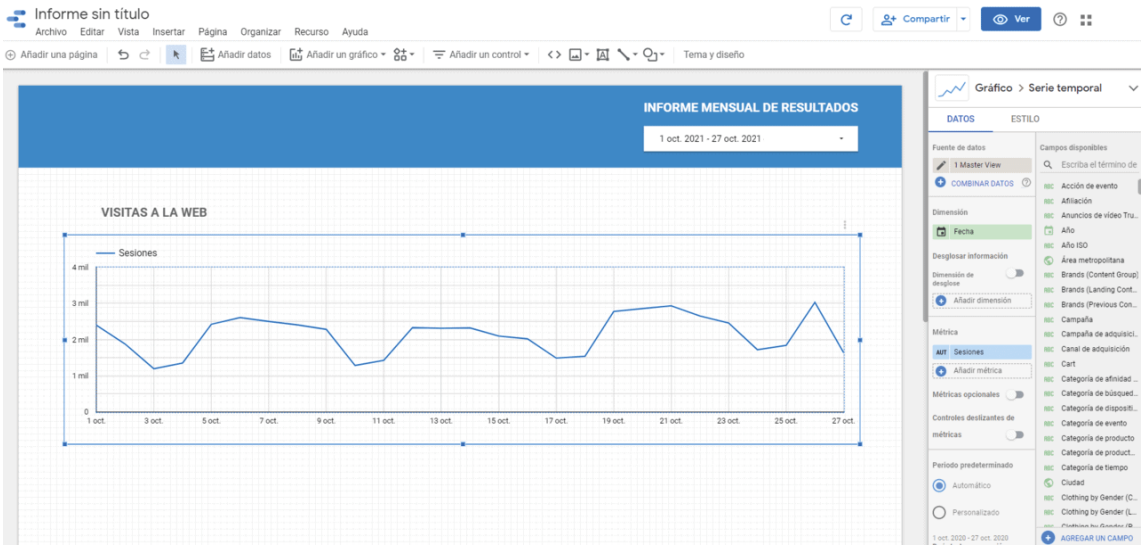



Imagen 7. Serie temporal. Fuente: Data Studio.

- **Otras representaciones gráficas**. Son diferentes representaciones que incluyen datos según la métrica que se pretende analizar. Podemos encontrar:

  » **Líneas**. Se usan mucho para las sesiones que hubo en un periodo de tiempo concreto, también para los usuarios activos en un periodo de tiempo concreto. Son especialmente útiles para descubrir tendencias y, en especial, para destacar tendencias al alza o a la baja.

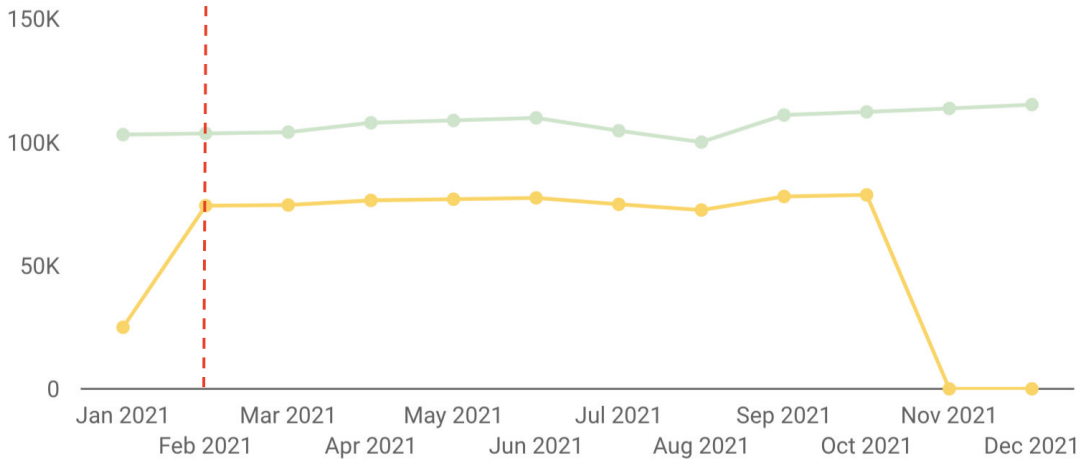



Imagen 8. Gráfico de líneas. Fuente: Data Studio.

  » **Barras**. Se usan mucho para representar datos demográficos tales como la edad y sexo de los visitantes o de los que realizan compras. No arrojan mucha luz sobre cómo se ha llegado a esos valores y cuál es la tendencia de los mismos.

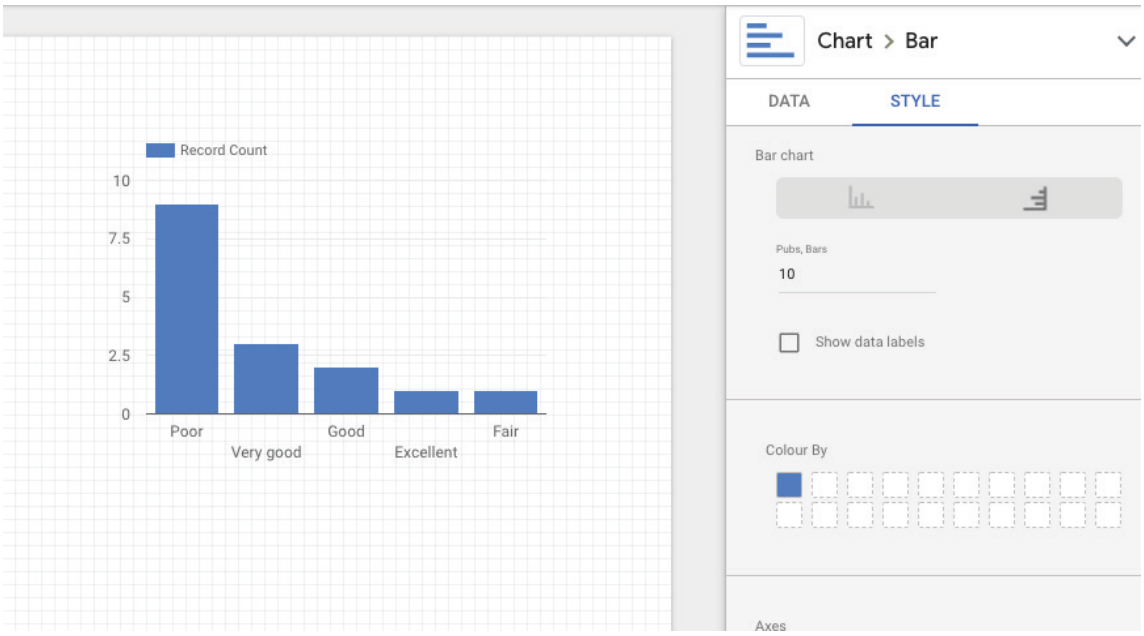



Imagen 9. Gráfico de barras. Fuente: Data Studio.

» **Círculos**. Tienen el mismo uso que las barras: se utilizan para representar valores en una única dimensión (edad, sexo, localización, etc.).

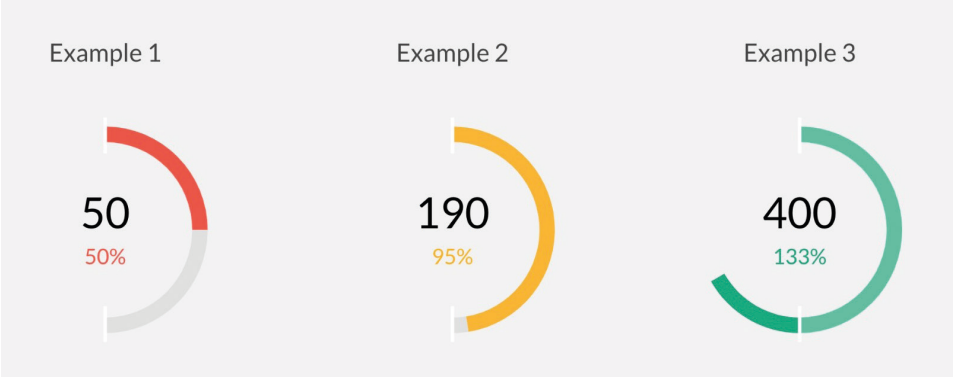



Imagen 10. Gráfico de círculos. Fuente: Data Studio.

» **Dispersiones**. Permiten usar dos métricas, una por eje; se puede incluso añadir una tercera métrica. Son muy útiles para mostrar las relaciones entre los distintos bloques de datos y, especialmente, con valores numéricos. Por ejemplo, resultaría interesante añadir las métricas Sesiones, Duración media de sesión y Porcentaje de nuevas sesiones.

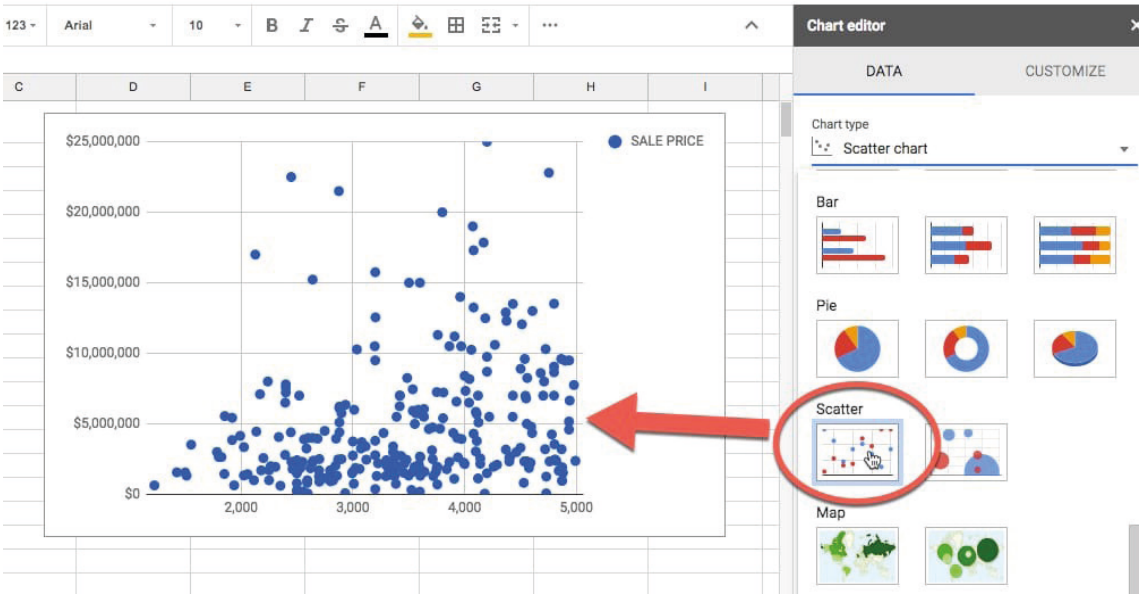



Imagen 11. Gráfico de dispersión. Fuente: Data Studio.

» **Gráficos de tarta**. Al igual que los gráficos circulares, solo se usarán para mostrar una única dimensión (edad, sexo…). Útiles para mostrar la información de una forma más visual, pero no para realizar comparaciones y análisis complejos.

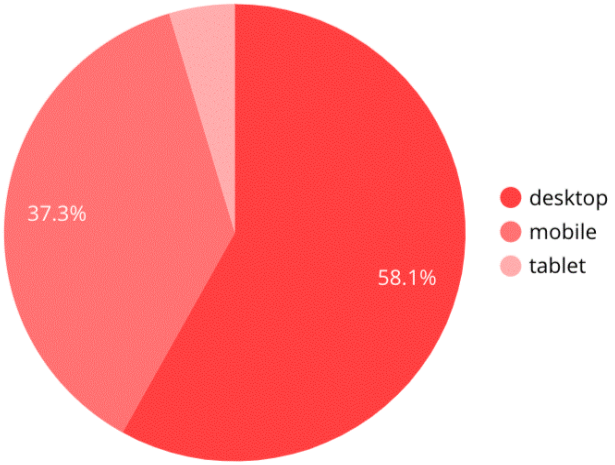



Imagen 12. Gráfico de tarta. Fuente: Data Studio.

- **Funciones interactivas**. Se pueden agregar funciones interactivas a los informes para brindar más vitalidad y claridad a los datos.

Imagen 13. Funciones interactivas. Fuente: Data Studio.

## 5.3. FUENTES DE DATOS EN DATA STUDIO

El primer paso para comenzar a usar Google Data Studio es registrarte en su sitio web. Se trata de un interfaz muy claro e intuitivo. En la esquina superior derecha hay un botón llamado Crear que nos permite crear un informe, conexión de datos o buscador.

Finder o **Buscador** está diseñado para encontrar la información lo más fácil y rápidamente posible. Destacan las plantillas que ofrece para presentar la información en los informes.

Data Studio admite muchas conexiones de datos (actualmente hasta 18, pero este número se incrementa con frecuencia ya que la herramienta se actualiza periódicamente).

# 5.4. INFORMES EN DATA STUDIO

Después de agregar la conexión de datos, puede diseñar el informe. Data Studio permite utilizar una plantilla prediseñada o elegir una en blanco.

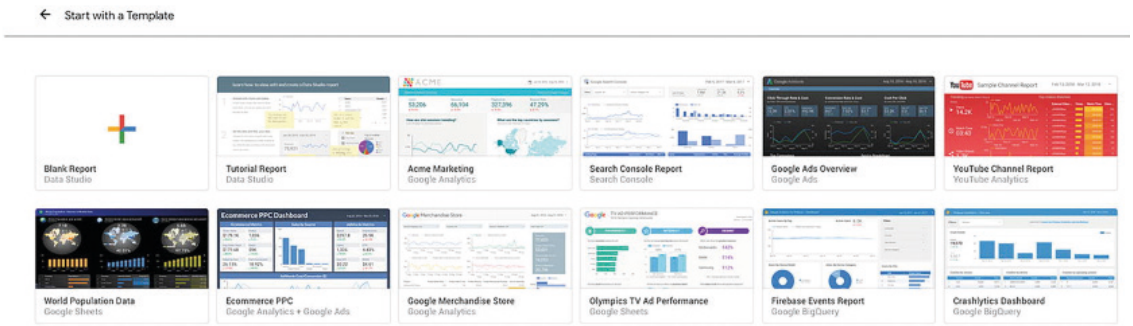



Imagen 14. Plantillas de informe predeterminadas. Fuente: Data Studio.

Si se elige una plantilla en blanco, se puede crear el informe desde cero y agregar los widgets que mejor se adapten a sus necesidades. No olvide dar un nombre al informe.

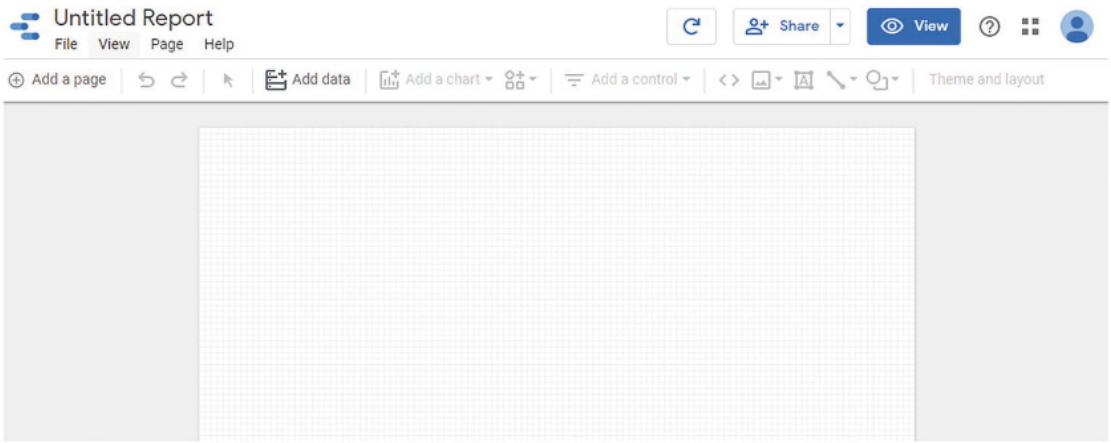



Imagen 15. Plantilla en blanco. Fuente: Data Studio.

**Tableros *(dashboard)***. Permiten agregar valor al informe. Basta con seleccionar de entre las opciones disponibles en el menú desplegable Agregar gráfico.

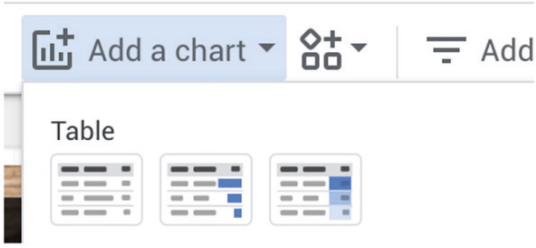



Imagen 16. Opciones de tablero. Fuente: Data Studio.

Se puede usar el **editor de estilo** para establecer bordes, redondear esquinas y ajustar el relleno del tablero. Se debe tratar de mantener el estilo simple para que el informe sea lo más claro y sencillo posible.

Los tableros pueden representar diferentes cantidades de datos. Con la opción **Añadir datos** se permite agregar datos al informe, pudiendo conectarse de otra fuente de datos o de una propia.

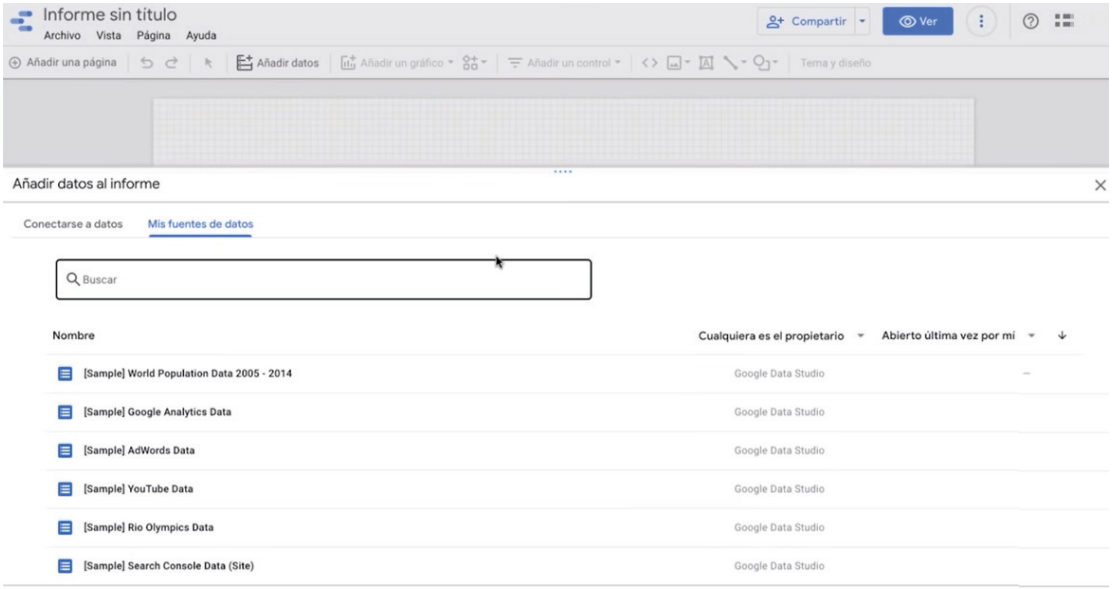



Imagen 17. Añadir datos a un informe. Fuente: Data Studio.

Las **dimensiones** son atributos de datos.

Las **métricas** son los datos cuantitativos.

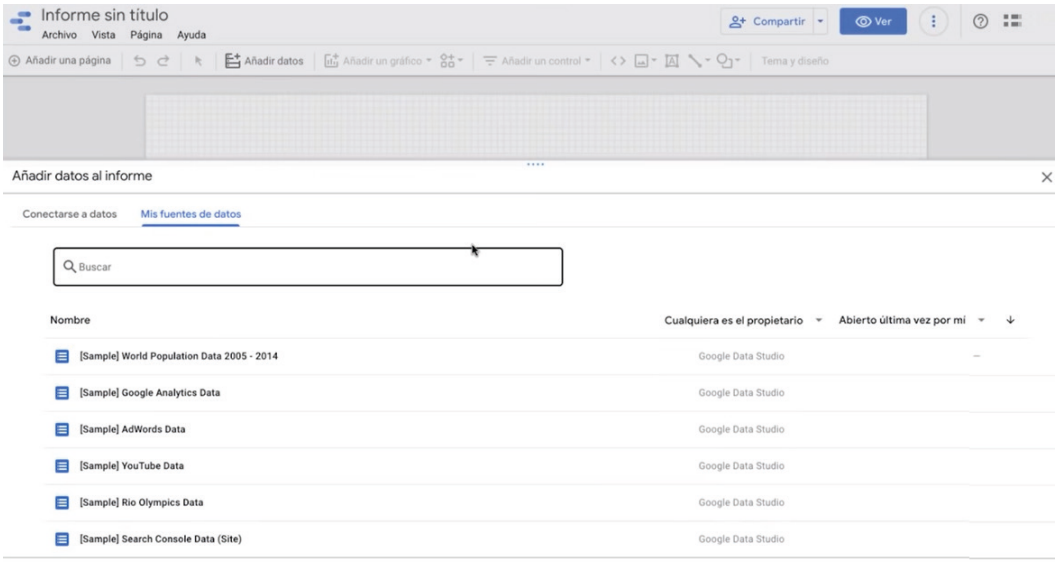



Imagen 18.

El siguiente paso es **agregar gráficos** para completar el informe. Para ello se debe seleccionar el tipo de gráfico deseado desde el menú desplegable Agregar gráfico.

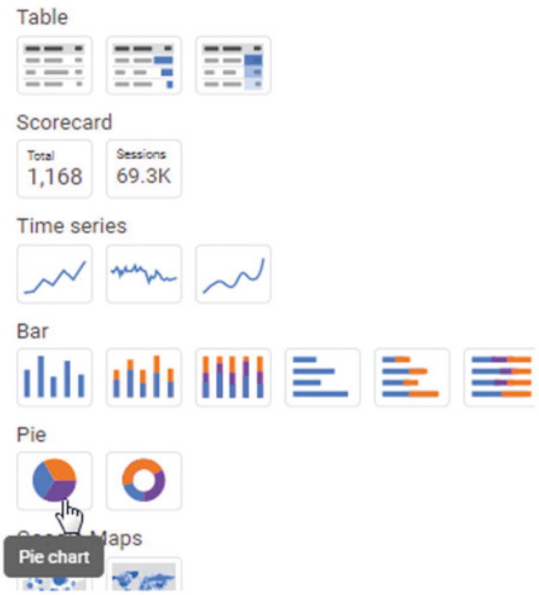



Imagen 19. Opciones de gráficos para los informes. Fuente: Data Studio.

Se pueden agregar gráficos de mapas, tablas, cuadros de mando, series temporales, gráficos de barras, gráficos circulares, gráficos de líneas, gráficos de áreas, gráficos de dispersión, gráficos dinámicos y algunos otros más.

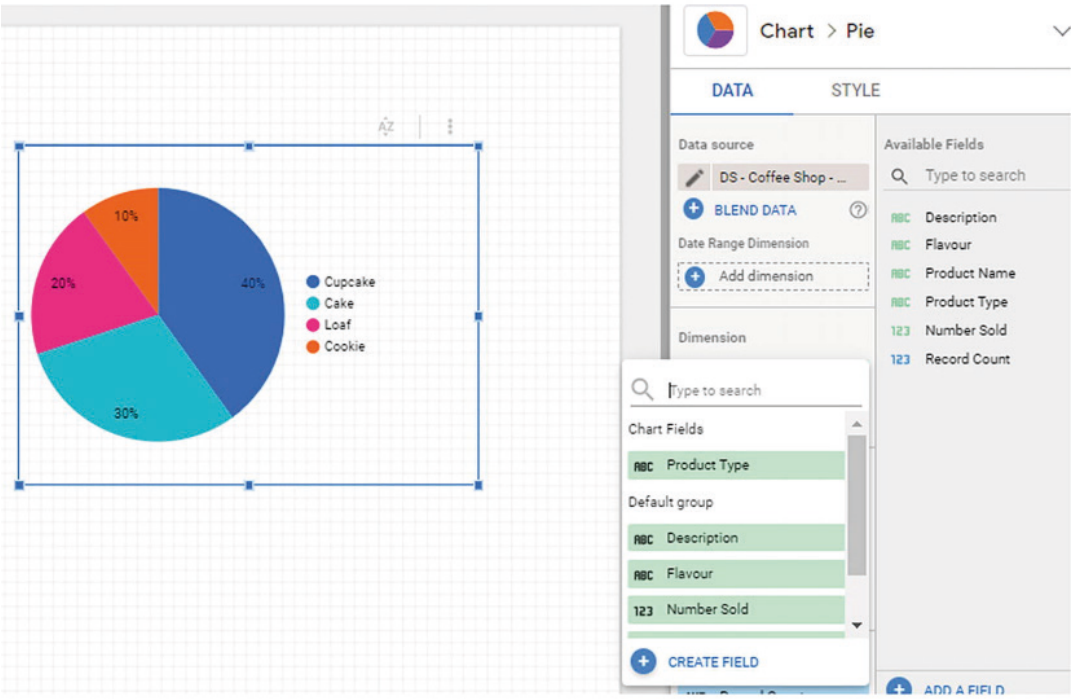



Imagen 20. Ejemplo de gráfico. Fuente: Data Studio.

Tras haber diseñado el informe y agregado los gráficos que se necesiten, ya está casi todo listo. Google Studio una herramienta simple y sencilla. En este momento ya se podrá previsualizar el informe que se ha creado seleccionando el botón Ver/ View.

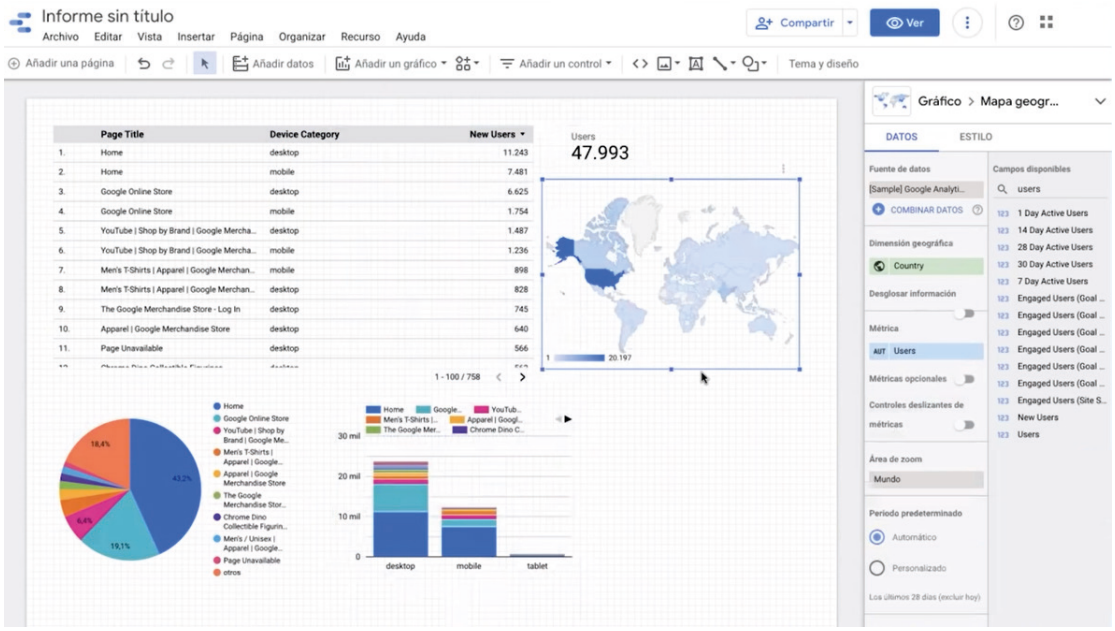



Imagen 21. Informe con gráficos. Fuente: Data Studio.

Si se desea cambiar algo, el informe se puede **Editar** con la opción situada la esquina superior derecha.

Ahora que hemos creado el informe final, es el momento de compartirlo con otros. Las opciones para compartir a las que podemos acceder en Data Studio son:

- **Descargar**. La primera opción es descargar el informe en formato PDF. Está disponible haciendo clic en el icono de descarga situado en la parte superior de la consola. Se puede hacer que el archivo ignore los colores personalizados, incluir un enlace a un informe en línea y proteger el informe con una contraseña.

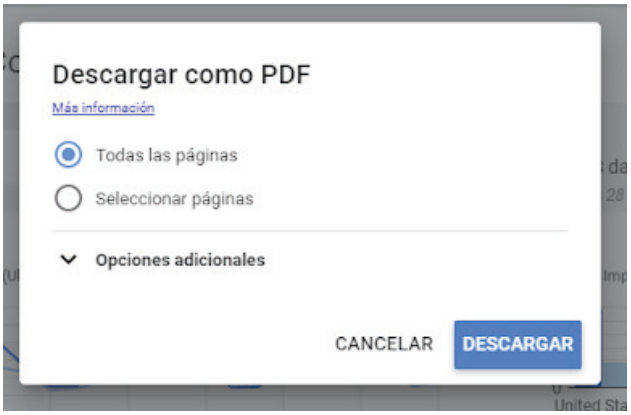



Imagen 22. Descarga de un informe como PDF. Fuente: Data Studio.

- Otra opción es configurar la **entrega automática de correo electrónico**. Para hacer esto basta con pulsar en el icono con forma de reloj en la parte superior de la pantalla para ver la ventana emergente. Así, Data Studio enviará por *email* una versión en formato PDF del informe a la dirección de correo electrónico que elija con periodicidad diaria, semanal o mensual.

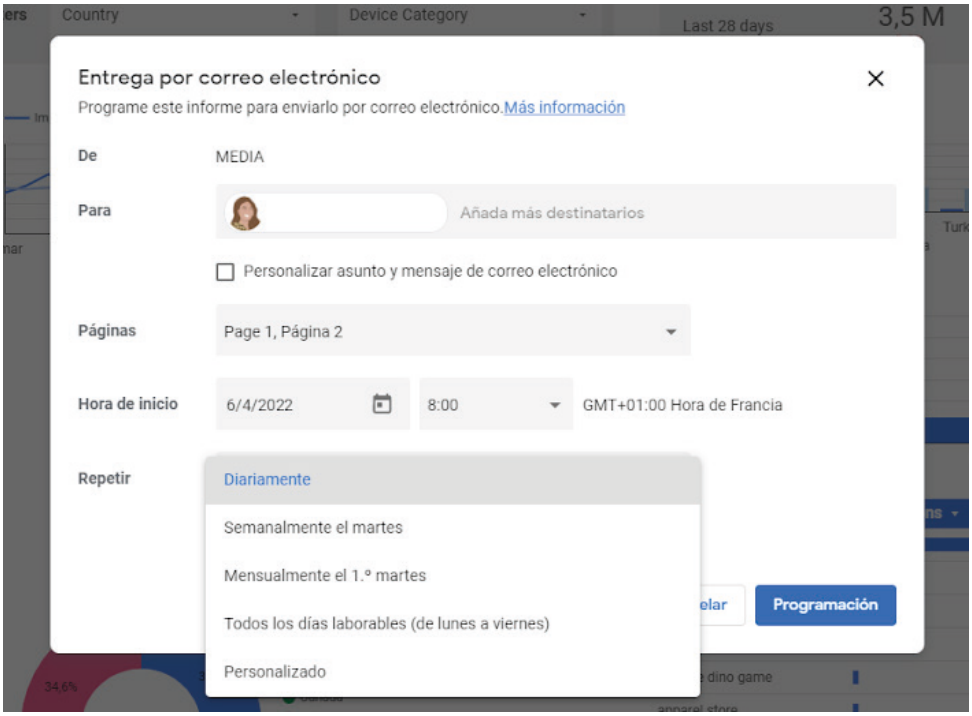



Imagen 23. Envío de un informe por correo electrónico. Fuente: Data Studio.

- También se puede **crear un enlace del informe**. A través del botón de Enlace, se abrirá una ventana emergente con una URL que puede copiar y compartir según sea necesario.

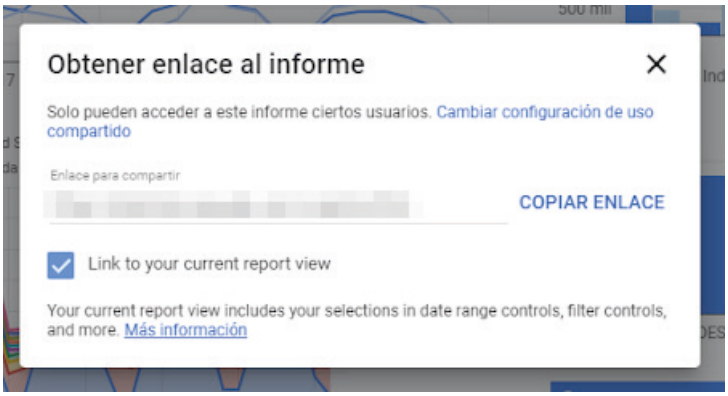



Imagen 24. Generar un enlace de visualización de un informe . Fuente: Data Studio.

- Finalmente, también se puede **compartir** como cualquier otro documento de Google Drive. Al hacer clic en el ícono Compartir, se abrirá la ventana que permite realizar esta operación.

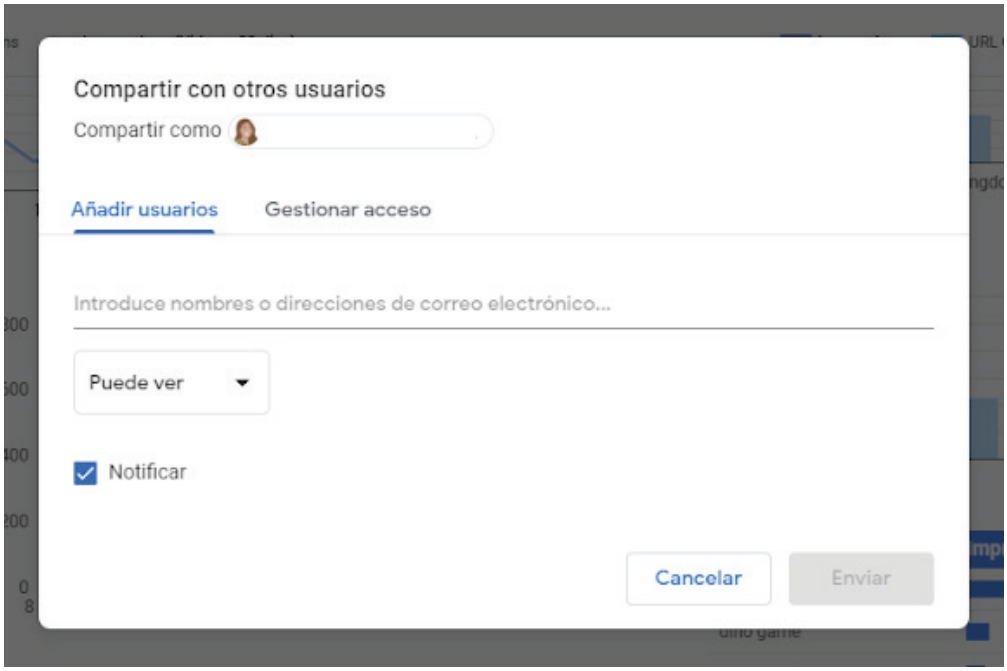



Imagen 25. Compartir un informe. Fuente: Data Studio.

Dada la importancia del análisis de datos dentro del marketing *online*, hay muchas otras herramientas alternativas a Google Data Studio, casi siempre son de pago, entre las que destacan:

- **Databox**: es una herramienta orientada al análisis de KPI en dispositivos móviles. Resulta algo cara, pero cuenta con un buen soporte.
- **Kibana**: con una interfaz muy intuitiva y analíticas muy flexibles, es adecuada para pequeñas empresas. También resulta algo cara, pero cuenta con un buen soporte y es de fácil de uso.
- **DashThis**: también es de fácil uso y cuenta con una interfaz muy atractiva para facilitar la labor a usuarios con poco tiempo para aprender a usarla. Además, permite personalizarla para instalar y, por tanto, pagar solo por lo que se va a usar.
- **Grow.com**: es muy potente para pequeñas y medianas empresas. Al ser un aplicativo web, no requiere instalar nada en ningún equipo.
- **Microsoft Power BI**: con un precio desde 9 € al mes, es una herramienta útil cuando se pretende procesar grandes cantidades de datos.
- **Looker**: es una herramienta especialmente útil para los análisis de comportamientos de usuarios y la captura de datos de múltiples fuentes. Es muy escalable.

La elección de una u otra herramienta debe realizarse de acuerdo con los parámetros de quien la vaya a usar (grado de experiencia en herramientas de análisis de datos, tiempo de aprendizaje, tiempo para su uso), las necesidades reales y el precio que se está dispuesto a pagar.

En los siguientes enlaces se pueden visualizar tutoriales sobre Google Data Studio:

https://support.google.com/datastudio/answer/6390659?utm_source=in-product&utm_medium=feature-panel&utm_campaign=videos

https://www.youtube.com/watch?v=RtbVBeu3gtk

# IDEAS CLAVE

- Google Data Studio es una herramienta gratuita de Google que ayuda a los usuarios a crear informes y paneles personalizados utilizando datos de las diferentes herramientas de marketing de Google y otras fuentes externas.
- La creación de informes totalmente personalizables es la característica principal de esta herramienta. Dentro de ellos, encontrará un panel en el que puede ver métricas y gráficos que proporcionarán el cuerpo para el tablero final.
- Cuando se haya creado un informe y lo haya conectado a diferentes fuentes de datos, todo lo que tiene que hacer es reflejar la información y las métricas en un gráfico. Pueden ser de diferentes tipos.
- Una vez finalizado el informe se podrá descargar, enviar por email, crear un enlace al informe o simplemente compartirlo por Google Drive.

# GLOSARIO

- **Fuentes de datos.** Instancias específicas de una cierta conexión con sus respectivos datos. Dan un esquema o estructura de los campos para crear informes.
- **Gráficos.** Representación en forma gráfica de los datos, de forma que su visualización resulta más sencilla y evidente.
- **Informes.** Presentaciones de información que permiten tomar decisiones o facilitar un cambio.
- ***Scorecards.*** Muestran el resumen de una cierta métrica. Si se usan en Data Studio con indicadores clave, permiten medir el rendimiento del negocio y del mercado.

# BIBLIOGRAFÍA

Kumar, Deepak. (s.f.). *The art of visualizing data in Google Data Studio.* Publicación independiente.

Kemp, Grant y White, Gerry. (2020). *Google Data Studio for Beginners Start Making Your Data Actionable.* Publicación Independiente.

ENEB | Escuela de Negocios Europea de Barcelona