

Dashboard con Power BI

LUIS ARDÉVOL MESA

El objetivo de esta práctica es crear un dashboard permita avanzar en conocimiento sobre aspectos relacionados con el tráfico aéreo global. En este caso, se centrará la atención en una región concreta, Canarias. Para ello, se usará la base de datos proporcionada, *backup_vuelos*, que contiene información sobre rutas aéreas, aeropuertos y aerolíneas.

No se entrará en detalles acerca de la carga de los datos en Power BI; esta simplemente se realiza mediante la importación de la base de datos de PostgreSQL existente en nuestra máquina virtual, *backup_vuelos*, haciendo uso del puerto 5432.

Las Islas Canarias son un conocido destino turístico no solo en España, sino en Europa. Sin embargo, el gran número de turistas que recibe, así como una orientación poco sostenible de este sector están llevando a las islas y a su gente al límite. El objetivo de este dashboard es mostrar el tráfico aéreo de Canarias, así como otros indicadores relacionados con la conectividad aérea de las islas. Esto busca ofrecer una visión detallada (considerando los datos disponibles) del impacto que puede tener el tráfico aéreo en la región.

El dashboard generado se compone de 4 visuales: un diagrama de cuerdas que muestra el número de conexiones interinsulares, para poner en perspectiva con el resto de conexiones que se mostrarán más adelante; un gráfico de barras apiladas generado con *R*, que representa la presencia de las distintas aerolíneas en el archipiélago; un mapa coroplético generado con *Python*, que muestra el número de rutas directas a Canarias por países; y un mapa de flujo que especifica cada una de las conexiones nacionales e internacionales (excluyendo las interinsulares) de las islas.

El número de aeropuertos en Canarias es reducido, por lo que para filtrar los aeropuertos se recurrirá a los propios IDs de los mismos: 1051 - Fuerteventura, 1052 - El Hierro, 1053 - La Palma, 1054 - Gran Canaria, 1055 - Lanzarote, 1056 - Tenerife (sur), 1057 - Tenerife (norte) y 5659 - La Gomera. Este método, si bien más directo, no sería aplicable a escalas mayores.

1. DIAGRAMA DE CUERDAS

Para mostrar las conexiones interinsulares, lo primero es filtrar las mismas en la tabla de rutas. Para ello, se crea una nueva tabla, “RutasCanariasFiltradas”, que contenga solo este tipo de rutas. Esto se hace de forma sencilla como

```
RutasCanariasFiltradas = FILTER('public routes', 'public routes'[
    source_airport_id] IN {1051, 1052, 1053, 1054, 1055, 1056, 1057,
    5659} && 'public routes'[destination_airport_id] IN {1051, 1052,
    1053, 1054, 1055, 1056, 1057, 5659})
```

Una vez creada la tabla, se puede hacer la representación. Este diagrama no viene por defecto en *Power BI*, pero se puede añadir en la obtención de más objetos visuales; concretamente, se usa “Chord” de *Microsoft Corporation*. Este visualización requiere tres campos: el origen, el destino y el valor. Para que sea cómodo de

leer, se quiere que las etiquetas correspondan con los nombres de los aeropuertos, no los IDs. Esto se consigue creando dos nuevas columnas en la tabla “RutasCanariasFiltradas”: “NombreOrigen” y “NombreDestino”, que relacionan el ID del aeropuerto con su nombre

```
NombreOrigen = RELATED('public airports origen'[name])
NombreDestino = RELATED('public airports destino'[name])
```

Para los valores, sería interesante que aparecieran ordenados por el número de rutas que los conectan. Para ello, se crea nueva columna en la tabla “RutasCanariasFiltradas”, “OrdenOrigen”, que dé el orden necesario a la visualización

```
OrdenOrigen = SWITCH(TRUE(), RutasCanariasFiltradas[source_airport_id] = 1051, 1, RutasCanariasFiltradas[source_airport_id] = 1052, 2,
RutasCanariasFiltradas[source_airport_id] = 1053, 3,
RutasCanariasFiltradas[source_airport_id] = 1054, 4,
RutasCanariasFiltradas[source_airport_id] = 1055, 5,
RutasCanariasFiltradas[source_airport_id] = 1056, 6,
RutasCanariasFiltradas[source_airport_id] = 1057, 7,
RutasCanariasFiltradas[source_airport_id] = 5659, 8)
```

Así, introduciendo “NombreOrigen” en el campo “From”, “NombreDestino” en el campo “To” y “OrdenOrigen” en el campo “Values”, se obtiene el diagrama de cuerdas deseado. Al posicionarse encima de cualquier isla, aparecerá el número de rutas, y al posicionarse encima de las cuerdas, se mostrará el número de rutas (ida y vuelta) para esa conexión.

Este gráfico (figura 1) resulta importante para comparar el tráfico interinsular con el tráfico exterior. Se puede ver como Tenerife Sur posee muy pocas conexiones entre islas (y con solo tres islas). Esto contrasta con el hecho de que este es el segundo aeropuerto con más tráfico de las islas, con números muy similares al primero (Gran Canaria); se verá bien reflejado en los siguientes gráficos.

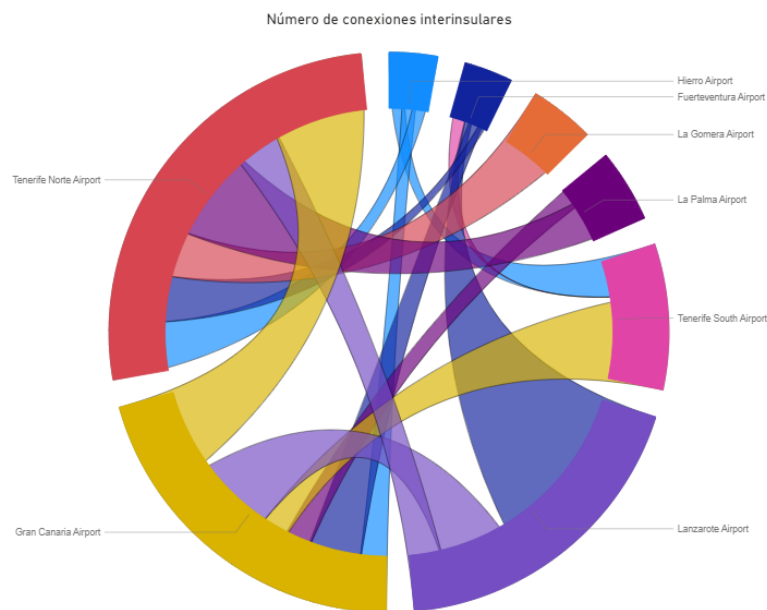


Fig. 1. Conexiones interinsulares en Canarias.

2. GRÁFICO DE BARRAS APILADAS

Para mostrar la presencia de las distintas aerolíneas en Canarias, se recurre a un gráfico de barras apiladas. Para hacer un desglose por islas, es necesario crear nuevas medidas en la tabla de rutas: se crean medidas del número de rutas de cada aerolínea con destino en cada isla, con nombres tipo “VuelosACE” (para Lanzarote), “VuelosFTV” (para Fuerteventura), etc.

```
VuelosACE = CALCULATE(COUNT('public routes'[airline_id]), FILTER('
  public routes', 'public routes'[destination_airport_id] = 1055))
```

El resto tienen una estructura completamente análoga, cambiando el ID. Como valores para el gráfico, se pasarán estas medidas, junto con el nombre de las aerolíneas, es decir, la columna “name” de la tabla “airlines”. Con esto, se forma un dataframe que, manipulando un poco en R, da como resultado el gráfico mostrado. El código de R usado es el siguiente: simplemente ordena el eje x (aerolíneas) por cantidad de rutas y ajusta los parámetros del plot haciendo uso de la librería “ggplot2”.

```
1 # El código siguiente, que crea un dataframe y quita las filas duplicadas, siempre
  se ejecuta y actúa como un preambulo del script:
2
3 # dataset <- data.frame(undefined, undefined.1, undefined.2, undefined.3, undefined
  .4, undefined.5, undefined.6, undefined.7, undefined.8)
4 # dataset <- unique(dataset)
5
6 # Pegue o escriba aquí el código de script:
7 library(tidyrr)
8 library(ggplot2)
9 library(dplyr)
10
11 data_long <- dataset %>%
12   gather(key = "Isla", value = "Vuelos", VuelosACE:VuelosVDE) %>%
13   group_by(name) %>%
14   mutate(total_vuelos = sum(Vuelos, na.rm = TRUE)) %>%
15   ungroup() %>%
16   arrange(desc(total_vuelos))
17
18 ggplot(data_long, aes(x = factor(name, levels = unique(name)), y = Vuelos, fill =
  Isla)) +
19   geom_bar(stat = "identity") +
20   theme_minimal() +
21   labs(title = "Presencia de aerolíneas en Canarias",
22        x = "Aerolínea", y = "Número de vuelos") +
23   theme(axis.text.x = element_text(angle = 45, hjust = 1, size = 8),
24         plot.title = element_text(hjust = 0.5)) +
25   scale_fill_manual(values = c("VuelosACE" = "#ca3142", "VuelosFTV" = "#009eff",
    "VuelosLPA" = "#4ad986", "VuelosTFN" = "#a226ff", "VuelosTFS" = "#fe9dff", "
    VuelosGMZ" = "#ffaa2c", "VuelosSPC" = "yellow", "VuelosVDE" = "#00ffbe"),
26   labels = c("VuelosACE" = "Lanzarote", "VuelosFTV" = "
    Fuerteventura", "VuelosLPA" = "Gran Canaria", "VuelosTFN" = "Tenerife (norte)",
    "VuelosTFS" = "Tenerife (sur)", "VuelosGMZ" = "La Gomera", "VuelosSPC" = "La
    Palma", "VuelosVDE" = "El Hierro"))
```

Este gráfico (figura 2) es importante para ver la presencia de las distintas aerolíneas en Canarias. La principal aerolínea es Ryanair, que destaca de forma notable sobre el resto. Esto se debe al gran número de rutas nacionales e internacionales que implementa esta aerolínea en el archipiélago canario. Se puede ver además como de las 10 principales, solo 2 son españolas. Además, se destaca la presencia de una mayoría de aerolíneas que no operan a nivel nacional, aportando solo rutas internacionales.

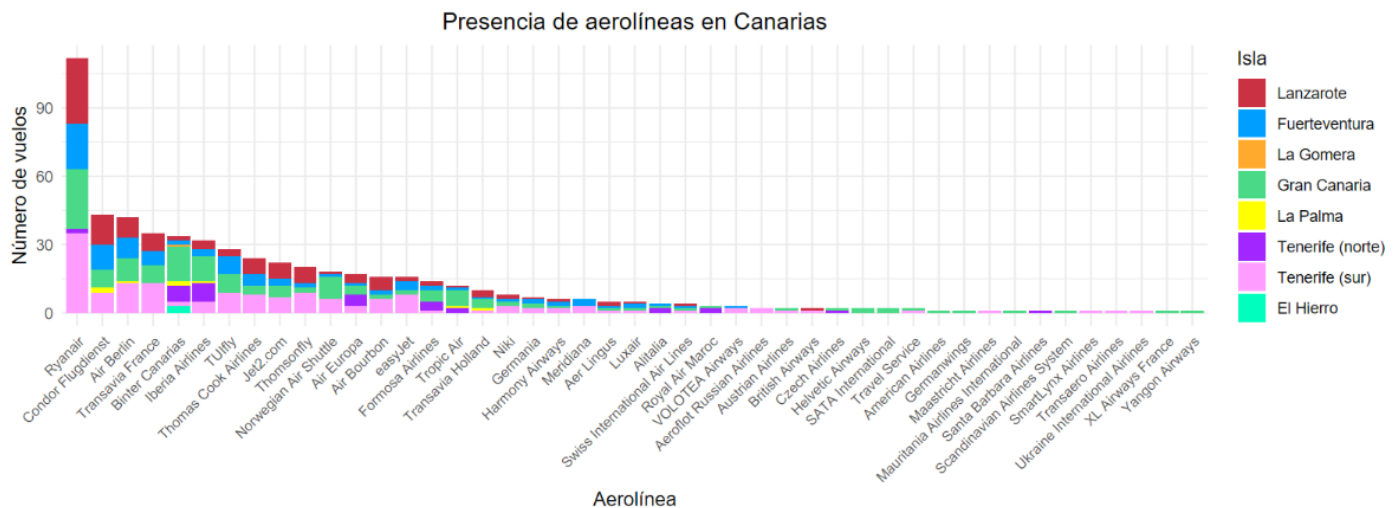


Fig. 2. Presencia de aerolíneas en Canarias.

3. MAPA COROPLÉTICO

Este mapa se realizará como visual de *Python*, ya que existen numerosas librerías que permiten la creación de mapas coropléticos de forma muy sencilla. El único cálculo necesario en este caso es el número de vuelos directos a Canarias por país. Para ello, se crea una nueva tabla, “Vuelos por País”, que contiene la magnitud que buscada, así como el país de origen

```
Vuelos por Pais = SUMMARIZE(FILTER('public routes', 'public routes'[
    destination_airport_id] IN {1051, 1052, 1053, 1054, 10551, 1056,
    1057, 5659}), 'public airports origen'[country], "Cantidad de
    Vuelos", COUNT('public routes'[airline_id]))
```

Esta tabla tendrá por tanto dos columnas, “country” y “Cantidad de Vuelos”; estos serán los campos a introducir en valores para generar el visual. En cuanto al código python empleado, se usó la librería plotly, que permite la creación de mapas interactivos. No obstante, *Power BI* no soporta la visualización de estos mapas, por lo que se guarda la representación como imagen plana y se muestra usando matplotlib. A continuación, el código *Python* empleado:

```
1 # El codigo siguiente, que crea un dataframe y quita las filas duplicadas, siempre
  # se ejecuta y actua como un preambulo del script:
2
3 # dataset = pandas.DataFrame(Cantidad de Vuelos, country)
4 # dataset = dataset.drop_duplicates()
5
6 # Pegue o escriba aqui el codigo de script:
7
8 import plotly.express as px
9 import matplotlib.pyplot as plt
10 import matplotlib.image as mpimg
11
12
13 fig = px.choropleth(dataset, locations = 'country', locationmode = 'country names',
14                     color = "Cantidad de Vuelos", hover_name = 'country',
15                     color_continuous_scale = px.colors.sequential.Sunsetdark,
16                     title = 'Numero de rutas directas a Canarias por paises')
```

```

17 fig.update_layout(margin = {"r":0, "t":0, "l":0, "b":0},
18                   title = {'y':0.87, 'x':0.5, 'xanchor':'center'},
19                   coloraxis_colorbar = dict(len=0.7, y=0.5, title="Rutas directas")
20 )
21 fig.write_image('choropleth_map.png')
22
23 img = mpimg.imread('choropleth_map.png')
24 plt.figure(figsize=(15,10))
25 plt.imshow(img)
26 plt.axis('off')
27 plt.show()

```

Esta visualización (figura 3) muestra algo curioso: existe un mayor número de conexiones con Reino Unido que con el propio territorio nacional, que presenta un número de conexiones similar al de Alemania. Esto evidencia las fuentes principales del turismo en Canarias y el crecimiento descontrolado de este sector.

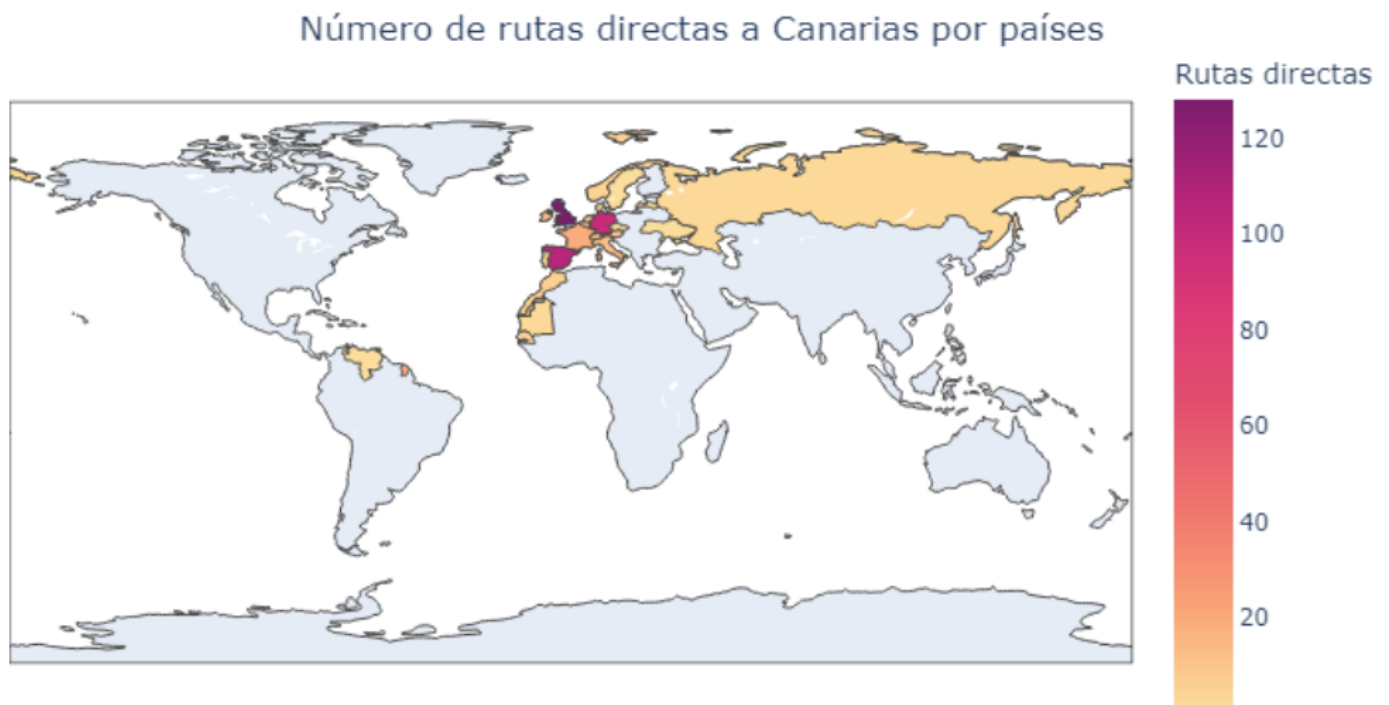


Fig. 3. Rutas directas a Canarias por países.

4. MAPA DE FLUJO

Para crear este mapa se comienza con un proceso similar al visto para el diagrama de cuerdas: filtrar las rutas. Esta vez, se busca filtrar las rutas con destino Canarias y origen fuera de las islas. Para ello, se crea una nueva tabla, "RutasFiltradas",

```

RutasFiltradas = FILTER('public routes', NOT('public routes'[
    source_airport_id] IN {1051, 1052, 1053, 1054, 1055, 1056, 1057,
    5659})) && 'public routes'[destination_airport_id] IN {1051, 1052,
    1053, 1054, 1055, 1056, 1057, 5659})

```

La visualización usada es “Flow Map” de *Weiwei Cui*, y no se encuentra disponible en los visuales por defecto. Esta representación requiere simplemente de valores de origen y destino para generar las trayectorias o rutas, modificables en color y anchura aportando otros campos como filtros. Los valores de origen y destino se deben dar como puntos (latitud, longitud), por lo que se crean nuevas columnas en la tabla generada para calcular, primero las coordenadas:

```
LatitudOrigen = RELATED('public airports origen'[latitude])
LongitudOrigen = RELATED('public airports origen'[longitude])
LatitudDestino = RELATED('public airports destino'[latitude])
LongitudDestino = RELATED('public airports destino'[longitude])
```

y posteriormente la concatenación de estas, para obtener el formato esperado por el mapa de flujo

```
Origen = CONCATENATE(RutasFiltradas[LatitudOrigen], CONCATENATE(", ",
    RutasFiltradas[LongitudOrigen]))

Destino = CONCATENATE(RutasFiltradas[LatitudDestino], CONCATENATE(",
    ", RutasFiltradas[LongitudDestino]))
```

En este caso, debido al gran volumen de rutas desde la península Ibérica y Reino Unido, escalar las trayectorias resultaba inviable para la correcta visualización del mapa, por lo que se puso una cota superior a este escalado, adecuada para poder visualizar el máximo número de rutas posibles. Esto se consigue ajustando el ancho de las trayectorias: se usa una escala lineal con mínimo 1 y máximo 5.

Este mapa, al ser un visual de *Power BI*, permite interacción. De este modo, y para aportar un mayor conocimiento acerca del tráfico aéreo, se muestran los nombres del origen y el destino: al situarse sobre cualquier trayectoria, se pueden ver los distintos puntos de origen que convergen a un mismo destino. Para conseguir esto, se crean dos nuevas columnas, “NombreOrigen” y “NombreDestino”, que relacionan el ID del aeropuerto con su nombre, como se hizo en el diagrama de cuerdas, y se introducen en el apartado “Tooltips” del visual:

```
NombreOrigen = RELATED('public airports origen'[name])
NombreDestino = RELATED('public airports destino'[name])
```

Este gráfico (figura 4) hace explícita la variedad de conexiones que existen con Canarias, y la distribución de las mismas a lo largo de tres continentes. Si bien el flujo principal proviene de Europa, desde todos los países de Europa occidental y parte de Europa del este, también existen rutas hacia el este africano y Sudamérica.

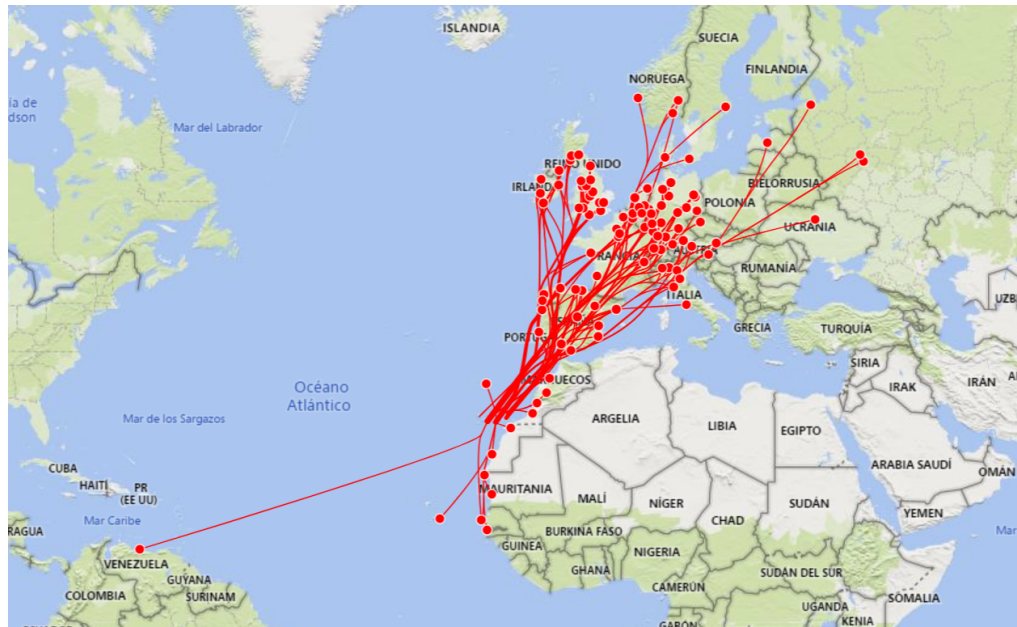


Fig. 4. Rutas aéreas a Canarias.

5. RESULTADO FINAL

A continuación se muestra la composición de los cuatro visuales en un único dashboard (figura 5). Este dashboard permite una visión global del tráfico aéreo en Canarias, mostrando las conexiones interinsulares, la presencia de las distintas aerolíneas, las rutas directas a Canarias por países y las rutas aéreas a Canarias, arrojando un poco de claridad sobre el insostenible modelo turístico actual del archipiélago.

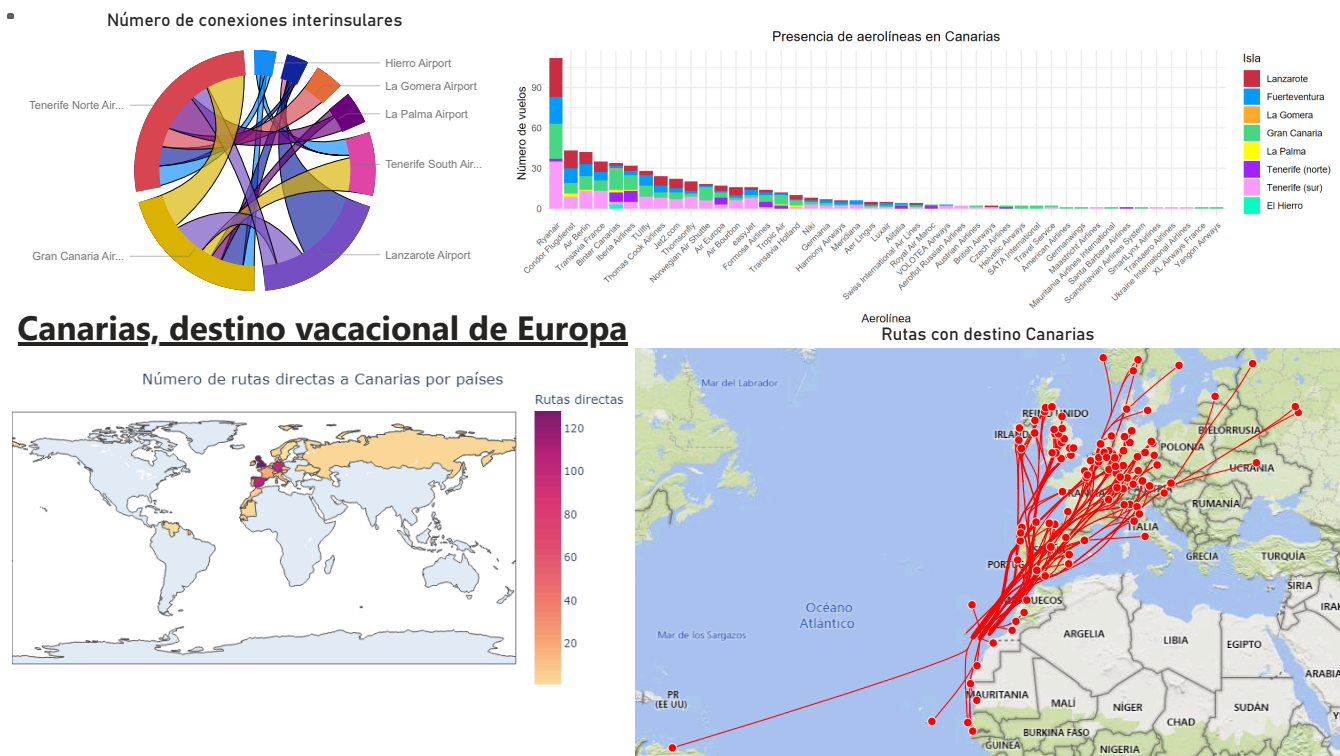


Fig. 5. Canarias, destino vacacional de Europa.