



## **Tarea 1:** **“Herramientas De Visualización De Datos”**

Visualización De Datos  
Campus Santiago San Joaquín  
Semestre 2023-1  
Prof. Jose Luis Martí

Integrantes:

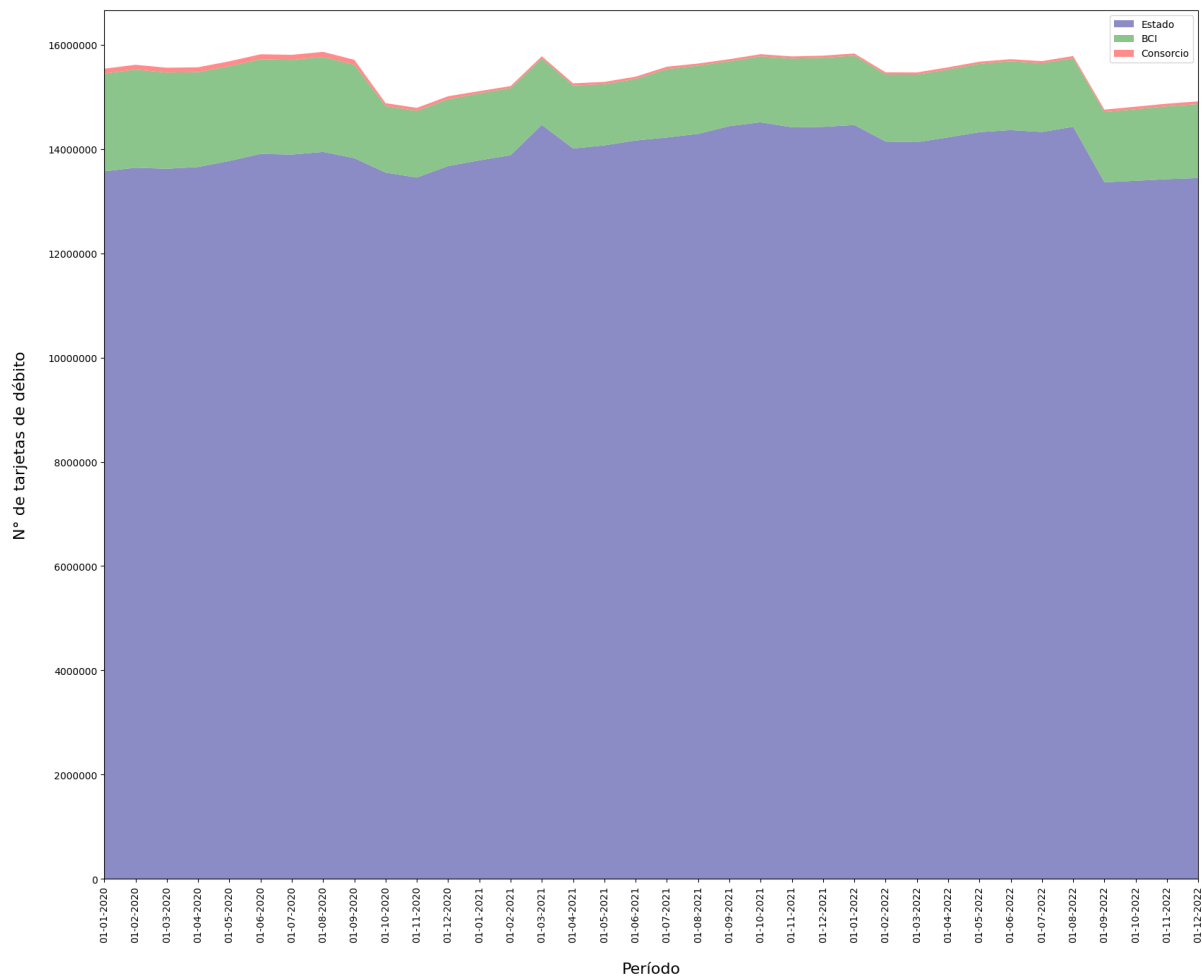
- Debora Alayo
- Javier Perez
- Alan Zapata

## Parte 1: Bancos

Los bancos elegidos para la tarea son Banco Estado, BCI y Consorcio. Cada uno ganador en un aspecto diferente respecto a los demás.

### 1.1 Banco Estado

Para la primera visualización se usa un sub-dataset, en base al proporcionado por el [CMF](#), que recoge los datos de Tarjetas de débito y ATM para los distintos bancos de Chile. De esta información, se trabajó solo con los bancos de interés (Estado, BCI, Consorcio) y con los años 2020 a 2022 y solo en el ámbito de las tarjetas de débito. Con esto se obtienen la cantidad de tarjetas de débito activas en el período para cada banco, para su futuro uso en la visualización.



Dentro de esta visualización se compara la cantidad de tarjetas de débito activas por cada banco en un período específico. A simple vista, se puede distinguir claramente que Banco Estado posee mayor cantidad de tarjetas de débito activas en todos los períodos, en comparación a BCI y Consorcio. Esto se aprecia en el área asociada a cada banco, donde notoriamente es más grande la de Banco Estado, seguida de BCI y en último lugar, Consorcio, que tiene tan pocas tarjetas de débito asociadas que su área visualmente parece más bien una curva/línea.

Las marcas utilizadas son áreas apiladas, donde la altura de cada área representa el número de tarjetas de débito en un momento determinado.

Los canales son las etiquetas de categoría relacionadas a un color, que identifican a los bancos en sí y los colores también permiten distinguirlos entre ellos. Otros canales son los ejes del gráfico, el vertical para el atributo cuantitativo y el horizontal para los períodos.

### 1.1.1 Código

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt

datos = pd.read_csv('oficial.csv', sep=';', index_col=0)
series = datos.values.transpose()
cols = ['#7f7fbf', '#7fbf7f', '#ff7f7f']

x = datos.index.transpose()
serie1 = series[0]
serie2 = series[1]
serie3 = series[2]
y = np.vstack([serie1, serie2, serie3])

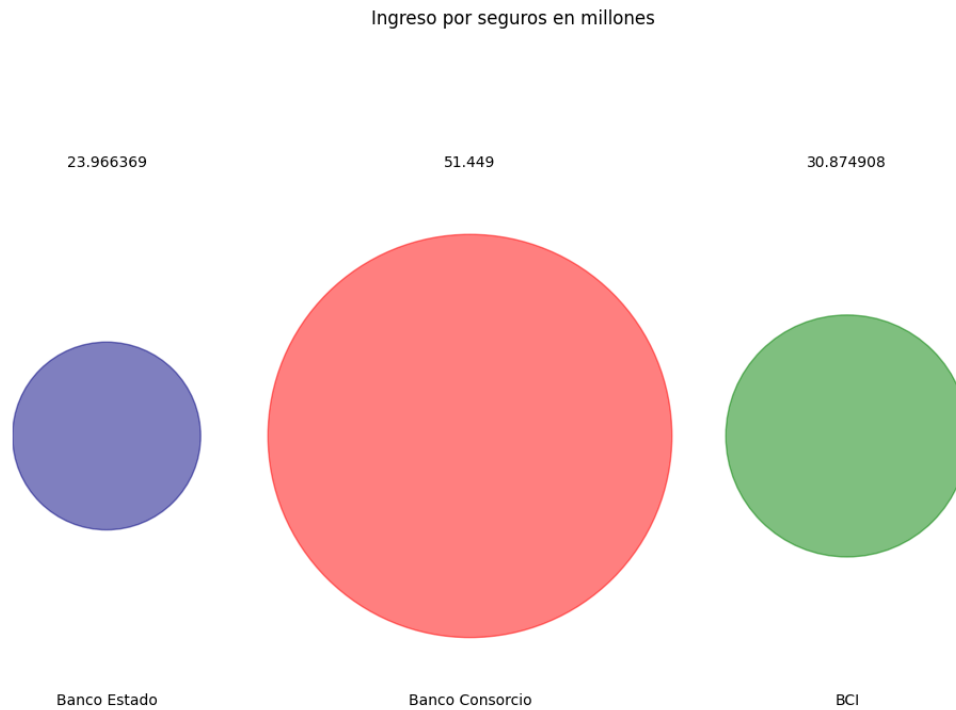
fig, ax = plt.subplots(figsize=(20, 16))
ax.stackplot(x, y, labels = ["Estado", "BCI", "Consortio"], colors = cols,
alpha = 0.9)
ax.legend(loc = 'upper right')
ax.set(xlim = (min(x), max(x)), xticks = x)

ax.set_xlabel('Período', fontsize=16, labelpad=20)
ax.set_ylabel('N° de tarjetas de débito', fontsize=16, labelpad=20)

plt.xticks(rotation=90)
plt.locator_params(axis='x', tight=True)
plt.ticklabel_format(style='plain', axis='y')

plt.show()
```

## 1.2 Banco Consortio



Este gráfico muestra el número de ingresos gracias a seguros en el año 2022 por parte de los bancos: [BancoEstado](#), [BCI](#), y, [Consortio](#).

El radio de los círculos está relacionado directamente con este número de ingresos, por lo cual, podemos notar que el banco consorcio es el que registra un ingreso mayor gracias a la cobranza y venta de seguros, seguido por el banco bci y en último lugar de la comparación el banco estado, nótese que no necesariamente significa que alguno de estos bancos tuvo menos o más ingresos que otros en general, solamente significa que ganó menos o más gracias al apartado de seguros.

La información se pudo obtener gracias a que cada banco cuenta con una subempresa que corresponde a la corredora de seguros por lo que viendo los estados de resultados de cada empresa podemos ver cuánto ganó cada banco gracias a los seguros.

Las marcas utilizadas en esta visualización son puntos para representar cada banco, mientras que los canales visuales corresponden al color a modo de identidad, para representar cada banco, y su tamaño, que muestra la cantidad de ingresos por seguros

### 1.2.1 Código

```
import matplotlib.pyplot as plt

values = [23966.369 , 51449.000, 30874.908]
bancos = ["Banco Estado", "Banco Consorcio", "BCI"]

sizes = values
colors = ['#000080', 'red', 'green'] centers = [(sizes[0]/2, 0),
((sizes[0]+sizes[1]/1.5), 0), (sum(sizes), 0)]

fig, ax = plt.subplots()
```

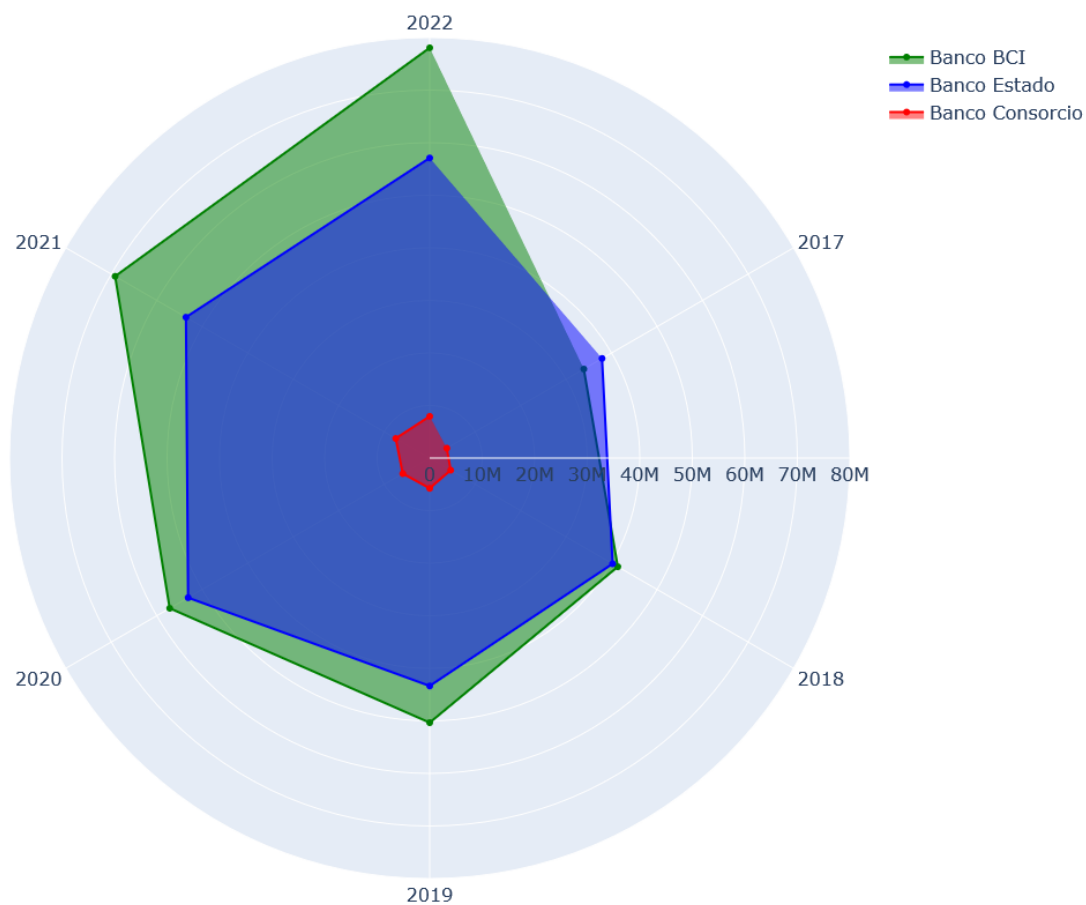
```
for bn, c, s, color in zip(bancos,centers, sizes, colors):
    circle = plt.Circle(c, s/2, color=color, alpha = 0.5)
    ax.add_artist(circle)
    label = ax.annotate(bn, xy=(c[0],-max(sizes)/1.5), fontsize=10,
ha="center")
    label = ax.annotate(s, xy=(c[0],max(sizes)/1.5), fontsize=10,
ha="center")

ax.set_aspect('equal')
plt.title("Ingreso por seguros en millones")
ax.set_xlim([0, sum(sizes)*1.2])
ax.set_ylim([-max(sizes), max(sizes)])

plt.axis("off")
plt.show()
```

### 1.3 Banco BCI

Total de activos por banco al 31 de Diciembre de los ultimos 6 años(En M\$)



Para el último de los gráficos comparativos de los bancos, se extrajo el monto total de activos por cada banco al 31 de Diciembre de cada año, rescatado desde los informes de estados financieros de los bancos [Consortio](#), [Estado](#) y [BCI](#).

Dentro de la presente visualización se puede apreciar como el Banco Consorcio nunca está cerca de superar a ninguno de los otros dos bancos en estudio, además que el año 2017 fue el

único año dentro del cual el Banco Estado supero al Banco BCI, dejándole el puesto desde el año 2018, diferencia la cual llego a su máximo, dentro de los periodos estudiados, a fines del año 2022. Esta información es util para quienes buscan invertir con o en algun banco en Chile, debido a la solidez en activos que muestran el Banco Estado y BCI

Las marcas utilizadas son puntos unidos por líneas formando un área radial en torno al centro.

Los canales son los colores asignables a cada banco en la etiqueta del gráfico, colores que identifican a cada banco y además facilitan ver las zonas donde se interceptan las áreas, ademas a medida que los puntos se alejan del centro, aumenta la cantidad de activos totales del banco al que pertenece dichos puntos respectivamente.

### 1.3.1 Código

```
import plotly.graph_objects as go

colores = ['#c11f1f', '#730000', '#1d9c34']

lineas = []
bci = [78049119, 69158634, 57156299, 50336620, 41350875, 33883396]
estado = [57090784, 53586003, 53119190, 43354976, 40221529, 37890236]
consorcio = [7939144, 7468745, 5916530, 5756872, 4610750, 3752719]
years=["2022", "2021", "2020", "2019", "2018", "2017"]

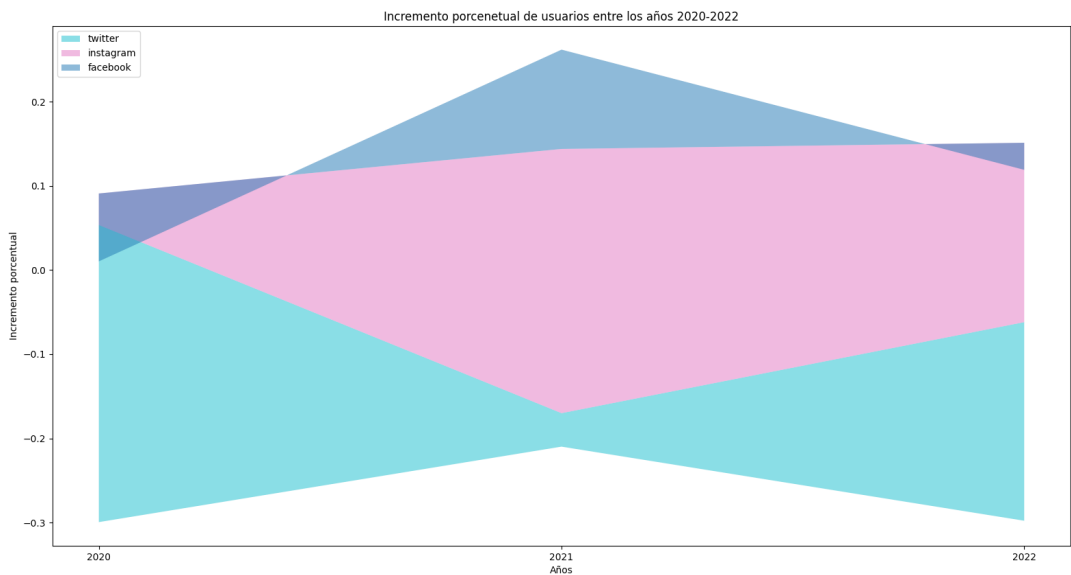
fig = go.Figure()
Activos= "Activos totales del año= "
fig.add_trace(go.Scatterpolar(
    r=bci,
    theta=years,
    fill='toself',
    name='Banco BCI',
    line_color='green'
))
fig.add_trace(go.Scatterpolar(
    r=estado,
    theta=years,
    fill='toself',
    name='Banco Estado',
    line_color='blue'
))
fig.add_trace(go.Scatterpolar(
    r=consorcio,
    theta=years,
    fill='toself',
    name='Banco Consorcio',
    line_color='red'
))
fig.update_layout(
```

```
polar=dict(  
    radialaxis=dict(  
        visible=True,  
        range=[0,80000000]  
    ),  
  
    showlegend=True,  
    title='Total de activos por banco al 31 de Diciembre de los ultimos 6  
años (En M$) ',  
    font_size=16,  
    legend_font_size=16,  
    polar_radialaxis_tickprefix='',  
    polar_angularaxis_rotation=90  
)  
  
fig.show()
```

## Parte 2: Redes Sociales

Las redes sociales elegidas para la tarea son Facebook, Instagram y Twitter. Cada una ganadora en un aspecto diferente respecto a las demás

### 2.1 Twitter



el siguiente gráfico muestra el incremento porcentual de usuarios, entonces, podemos notar cómo en términos porcentuales twitter ha crecido mas que facebook e instagram en el último año, esto es porque al no contarlo como la cantidad total de usuarios no consideramos si una red social creció un número fijo, sino que es relativo en cuanto a la cantidad total de usuarios que tenía previamente además podemos ver que facebook en el 2020 y 2022 tiene un porcentaje negativo, lo cual significa que decreció, instagram por otra parte, creció, pero no tanto como twitter, esto se ve reflejado en la cantidad de espacio que ocupa en el gráfico.

cabe destacar que si una categoría se encuentra invertida (señalada con un color un poco más oscuro dado que se contrapone a las otras categorías) esta decreció (como facebook en 2020 y

2022), caso contrario, la red creció, los datos comienzan a partir del 2019 para poder calcular la cantidad de nuevos usuarios respecto al total.

en este gráfico las marcas utilizadas corresponden a puntos que indican el porcentaje de usuarios por años, además de su enlace que corresponde a una línea que conecta dichos puntos, y finalmente una figura, que está rellena y es aquella que describen los enlaces, los canales visuales corresponden al color, a modo de identidad, y el área generada por la figura que crece con el incremento de usuarios, o decremento de estos.

La información presentada se encuentra a partir de los siguientes enlaces para los años: [2019](#), [2020](#), [2021](#), y, [2022](#).

### 2.1.1 Código

```
import numpy as np
import matplotlib.pyplot as plt
from matplotlib.ticker import MultipleLocator

twus = [251,339.6,353.1,436.4]
igus = [895,928.5,1220,1480]
fbus = [2121,1950,2180,2110]
years = [2020,2021,2022 ]
tl = np.diff(twus) / twus[:-1]
il = np.diff(igus) / igus[:-1]
fl = np.diff(fbus) / fbus[:-1]

print("twitter ",tl, "\ninstagram", il, "\nfacebook", fl)
colors = ['tab:cyan', 'tab:pink', 'tab:blue']

fig, ax = plt.subplots(figsize=(3, 6))

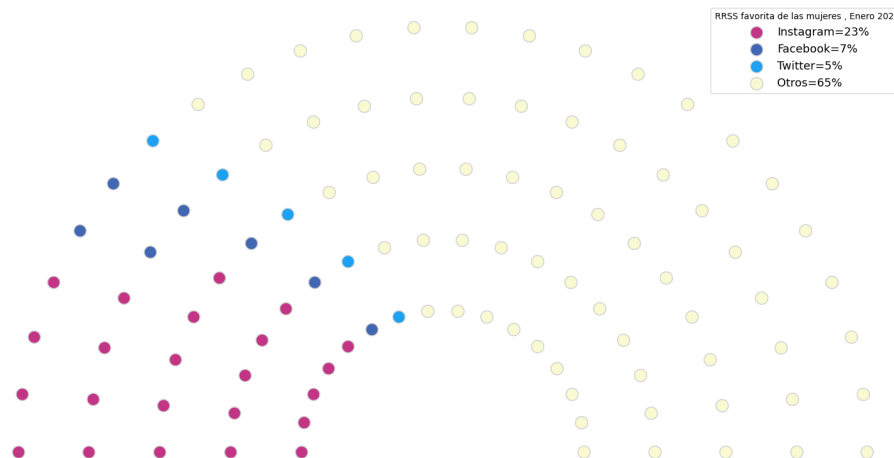
path = ax.stackplot(years, tl, il, fl, baseline='wiggles',
                    labels=['twitter', 'instagram', 'facebook'], colors =colors,
                    alpha =0.5)

ax.legend(loc='upper left')

ax.set_xlabel('Años')
ax.set_ylabel('Incremento porcentual')
ax.set_title('Incremento porcentual de usuarios entre los años
2020-2022')
ax.xaxis.set_major_locator(MultipleLocator(1))
plt.show()
```



## 2.2 Instagram



En este segundo gráfico comparativo de las redes sociales, se ilustró las preferencias de redes sociales de las mujeres entre 16 a 24 años en el mundo a partir de la información investigada por [DataReportal](#). Podemos observar que es ampliamente preferida Instagram, mientras que Facebook no alcanza a representar  $\frac{1}{3}$  de las preferencias del ganador, mientras que Twitter no alcanza ni  $\frac{1}{4}$  de las preferencias.

Es sabido que uno de los principales objetivos de las empresas en el mundo es obtener más clientes, lo cual lo hacen, entre otras cosas, mediante publicidad. La información presentada en el gráfico es particularmente útil para aquellas empresas que presentan clientas potenciales en el rango etario de 16 a 24 años, ya que, si es que se busca encontrar a alguien en ese rango, se sabe que el 23% de las mujeres prefieren Instagram.

Las marcas son puntos de colores, donde cada color representa a una red social en particular.

Los canales visuales corresponden a la cantidad de puntos del color representante de la red social, a mayor cantidad de puntos, mayor es el porcentaje de mujeres en el rango etario de la muestra que prefieren dicha red social

### 2.2.1 Código

```
import seaborn as sns
from poli_sci_kit import utils
import matplotlib.pyplot as plt

default_sat = 0.95

rrss=["Instagram", "Facebook", "Twitter", "Otras"]
porcentajes = [23, 7, 5, 65]
colores = ['#C13584', '#4267B2', '#1DA1F2', '#FAFAD2']

allocations=porcentajes
labels=rrss
colors=colores
style="semicircle"
```



```
num_rows=5
marker_size=200
df_seat_lctns=None

sns.set_palette(colors)

df_seat_lctns = utils.gen_parl_points(
    allocations=allocations,
    labels=labels,
    style=style,
    num_rows=num_rows,
    speaker=False,
)

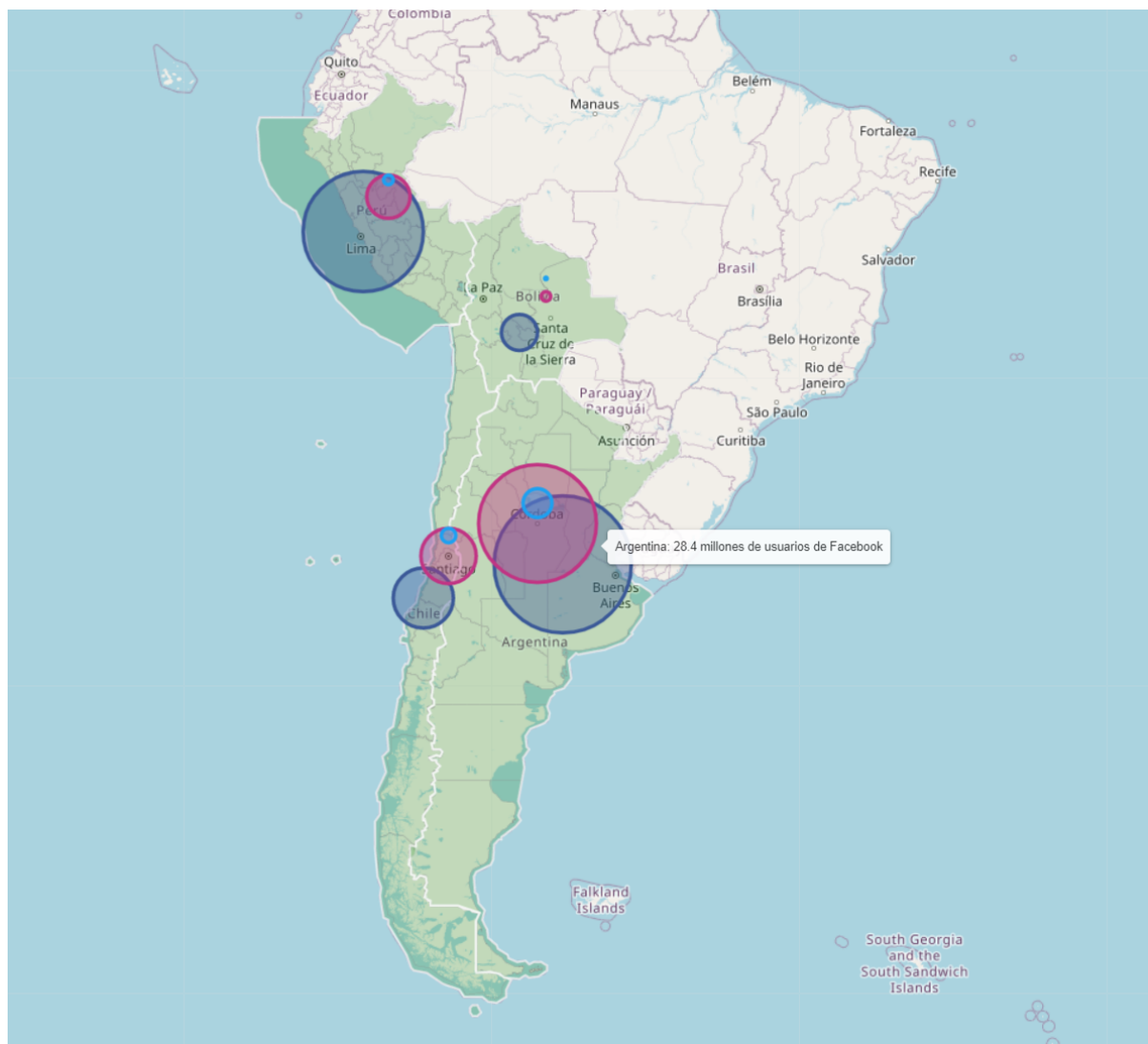
for g, lbl in enumerate(labels):
    df_subsetted = df_seat_lctns[df_seat_lctns["group"] == lbl]

    ax = sns.scatterplot(
        data=df_subsetted,
        x="x_loc",
        y="y_loc",
        color=colors[g],
        marker="o",
        s=marker_size,
        edgecolor="#D2D2D3",
    )

ax.axis("equal")
ax.axis("off")

plt.legend(labels=['Instagram=23%', 'Facebook=7%', 'Twitter=5%',
'Otros=65%'], title="RRSS favorita de las mujeres , Enero
2023", fontsize=13)
plt.show()
```

## 2.3 Facebook



Para esta visualización se arma un dataset en base a los datos proporcionados por *datareportal*, que recolectan estadísticas de las redes sociales por países en períodos anuales (para este caso, se utilizaron los informes de 2022). Se juntan los datos de usuarios para [Facebook](#), [Instagram](#) y [Twitter](#), para los países adyacentes a Chile, incluyéndolo ([Perú](#), [Bolivia](#), [Argentina](#) y [Chile](#)). La visualización es interactiva y corresponde a un archivo html (abrir en cualquier navegador aunque de preferencia Chrome, donde fue probado), disponible en el [enlace](#) (si es necesario se debe ajustar el zoom).

La visualización es un mapa donde sobre cada país se ubican tres burbujas y cada una es una representación de la cantidad de usuarios de una red social en ese país. Mientras más grande sea una burbuja, mayor cantidad de usuarios representa. En todos estos países, se observa que Facebook gana en cantidad de usuarios, aunque en Argentina la diferencia con Instagram es más sutil, pero existente.

Las marcas visuales son las burbujas, mientras los canales son los colores de las burbujas y los nombres de los países y sus formas.

### 2.3.1 Código

```
import folium

geojson_destacados = 'export.geojson'

mapa = folium.Map(location=[-14.235004,-51.92528], zoom_start=4)

países = [
    {'nombre': 'Chile', 'ubicacion': [-33.448889, -70.6692655]},
    {'nombre': 'Argentina', 'ubicacion': [-31.429771, -64.184019]},
    {'nombre': 'Perú', 'ubicacion': [-9.189967, -75.015152]},
    {'nombre': 'Bolivia', 'ubicacion': [-16.290154, -63.588652]}
]

usuarios_facebook = [12.5, 28.40, 24.8, 7.65]
usuarios_instagram = [11.65, 24.7, 8.9, 1.85]
usuarios_twitter = [2.9, 5.9, 2.2, 0.0004169]

desplazamiento = 1 * (5 / mapa.options['zoom'])

folium.GeoJson(geojson_destacados, style_function=lambda feature: {
    'fillColor': 'green',
    'color': 'white',
    'weight': 2,
    'opacity': 0.7
}).add_to(mapa)

for i in range(len(países)):
    if(países[i]['nombre']=='Argentina'):
        folium.CircleMarker(
            location=[países[i]['ubicacion'][0]-2*desplazamiento,
países[i]['ubicacion'][1]+1.5*desplazamiento],
            radius=usuarios_facebook[i]*2,
            color='#3b5998',
            fill=True,
            fill_color='#3b5998',
            fill_opacity=0.4,
            tooltip=f"{países[i]['nombre']}: {usuarios_facebook[i]} millones
de usuarios de Facebook"
        ).add_to(mapa)

    if(países[i]['nombre']!='Argentina'):
        folium.CircleMarker(
            location=[países[i]['ubicacion'][0]-2*desplazamiento,
países[i]['ubicacion'][1]-1.5*desplazamiento],
            radius=usuarios_facebook[i]*2,
            color='#3b5998',
            fill=True,
```



```
        fill_color='#3b5998',
        fill_opacity=0.4,
        tooltip=f"{países[i]['nombre']}: {usuarios_facebook[i]} millones
de usuarios de Facebook"
    ).add_to(mapa)

    folium.CircleMarker(
        location=[países[i]['ubicacion'][0], países[i]['ubicacion'][1]],
        radius=usuarios_instagram[i]*2,
        color='#C13584',
        fill=True,
        fill_color='#C13584',
        fill_opacity=0.4,
        tooltip=f"{países[i]['nombre']}: {usuarios_instagram[i]} millones
de usuarios de Instagram"
    ).add_to(mapa)

    folium.CircleMarker(
        location=[países[i]['ubicacion'][0]+desplazamiento,
países[i]['ubicacion'][1]],
        radius=usuarios_twitter[i]*2,
        color='#1DA1F2',
        fill=True,
        fill_color='#1DA1F2',
        fill_opacity=0.4,
        tooltip=f"{países[i]['nombre']}: {usuarios_twitter[i]} millones de
usuarios de Twitter"
    ).add_to(mapa)
bounds = [[-22.6, -81.2], [-0.7, -58.4]]
mapa.fit_bounds(bounds)
mapa.save('mapa.html')

mapa
```

## Anexo

Link a repositorio con los archivos de código y un csv utilizado :

[https://github.com/Forgettts/VDD\\_2023-1/tree/main/Tarea%201/Graficos%20Tarea%201](https://github.com/Forgettts/VDD_2023-1/tree/main/Tarea%201/Graficos%20Tarea%201)