

November 6, 2024

## 1 SPRINT 10 Tomar decisiones de negocio (tienda online)

### 2 Indice

1. Introduccion
2. Carga y limpieza de datos
3. Eleccion de hipotesis
  - ICE
  - RICE
4. Analisis de prueba A/B
  - Ingreso acumulado por grupos A/B
  - Tamaño de pedido promedio (por dia) por grupo A/B
  - Diferencia relativa en el tamaño de pedido promedio para el grupo B en comparacion con el grupo A
  - Tasa de conversion para cada grupo
  - Tasa de conversion diaria de los grupos
  - Numero de pedidos por usuario
  - Precio de pedidos
  - Significancia estadística de la diferencia en la conversión entre los grupos
  - Significancia estadística de la diferencia en el tamaño promedio de pedido entre los grupos
5. Conclusion general

#### 2.1 Introducción

En el dinámico entorno del comercio electrónico, optimizar las estrategias de venta es crucial para maximizar los ingresos y mejorar la experiencia del cliente. Este proyecto tiene como objetivo identificar las hipótesis más prometedoras para incrementar las ventas en una tienda online y evaluar la efectividad de una prueba A/B implementada para validar una de estas hipótesis.

En la primera parte del proyecto, se aplicarán los frameworks ICE y RICE para priorizar un conjunto de hipótesis diseñadas para aumentar los ingresos. Estos frameworks permitirán evaluar el potencial de cada hipótesis considerando factores como el impacto esperado, la confianza en su éxito y el esfuerzo requerido para implementarla.

A continuación, se analizarán los resultados de una prueba A/B realizada para validar una de las hipótesis priorizadas. A través del análisis de datos de pedidos y visitas, se evaluará el impacto de

la variante B en comparación con la variante A en términos de ingresos, tamaño promedio de los pedidos, tasa de conversión y otros indicadores clave de rendimiento.

El objetivo final de este proyecto es proporcionar recomendaciones basadas en datos para optimizar las estrategias de marketing y ventas de la tienda online, y contribuir al crecimiento del negocio.

## 2.2 Carga y limpieza de datos

```
[1]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import datetime as dt
import scipy.stats as stats
```

```
[2]: hipotesis = pd.read_csv("/datasets/hypotheses_us.csv", sep=";")
pedidos = pd.read_csv("/datasets/orders_us.csv")
visitas = pd.read_csv("/datasets/visits_us.csv")
```

```
[3]: hipotesis.columns = hipotesis.columns.str.lower()
hipotesis.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9 entries, 0 to 8
Data columns (total 5 columns):
#   Column          Non-Null Count  Dtype
---  -
0   hypothesis       9 non-null     object
1   reach            9 non-null     int64
2   impact           9 non-null     int64
3   confidence       9 non-null     int64
4   effort           9 non-null     int64
dtypes: int64(4), object(1)
memory usage: 488.0+ bytes
```

```
[4]: pedidos.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1197 entries, 0 to 1196
Data columns (total 5 columns):
#   Column          Non-Null Count  Dtype
---  -
0   transactionId    1197 non-null   int64
1   visitorId        1197 non-null   int64
2   date             1197 non-null   object
3   revenue          1197 non-null   float64
4   group            1197 non-null   object
dtypes: float64(1), int64(2), object(2)
```

memory usage: 46.9+ KB

```
[5]: pedidos = pedidos.rename(columns={'transactionId': 'transaction_id',  
    ↪ 'visitorId': 'visitor_id'})
```

```
[6]: pedidos['date'] = pd.to_datetime(pedidos['date'])  
print(pedidos.duplicated().value_counts())  
print(pedidos.isnull().sum())
```

```
False      1197  
dtype: int64  
transaction_id    0  
visitor_id        0  
date              0  
revenue           0  
group             0  
dtype: int64
```

```
[7]: visitas.info()
```

```
<class 'pandas.core.frame.DataFrame'>  
RangeIndex: 62 entries, 0 to 61  
Data columns (total 3 columns):  
#   Column  Non-Null Count  Dtype  
---  -----  -  
0   date     62 non-null      object  
1   group    62 non-null      object  
2   visits   62 non-null      int64  
dtypes: int64(1), object(2)  
memory usage: 1.6+ KB
```

```
[8]: visitas['date'] = pd.to_datetime(visitas['date'])  
print(visitas.duplicated().value_counts())  
print(visitas.isnull().sum())
```

```
False      62  
dtype: int64  
date        0  
group        0  
visits       0  
dtype: int64
```

## 2.3 Eleccion de hipotesis

### 2.3.1 ICE

```
[9]: hipotesis['ICE'] = (hipotesis['impact'] * hipotesis['confidence']) / hipotesis['effort']

hipotesis_ice = hipotesis.sort_values('ICE', ascending=False)
hipotesis_ice
```

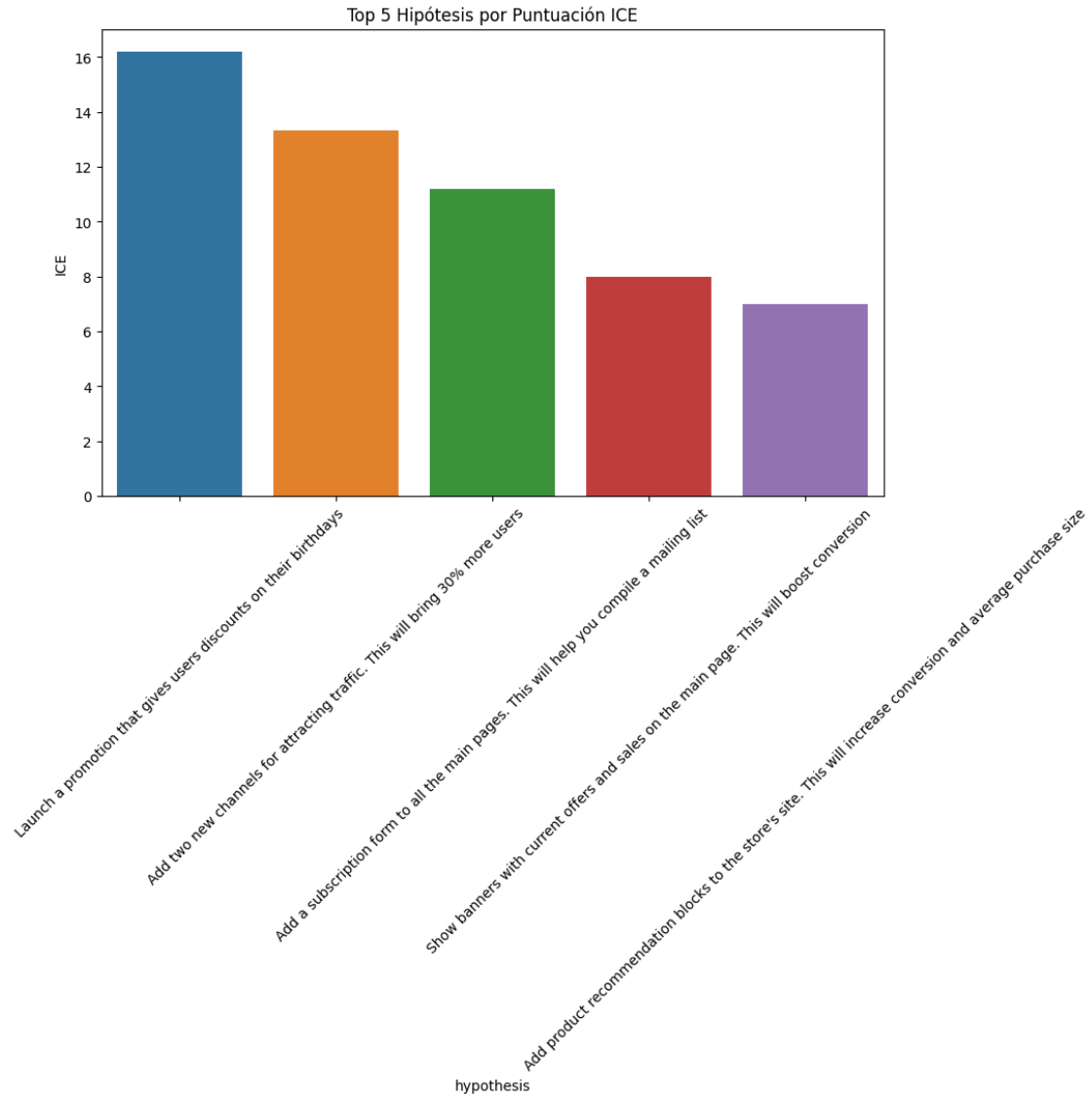
```
[9]:
```

	hypothesis	reach	impact	\
8	Launch a promotion that gives users discounts ...	1	9	
0	Add two new channels for attracting traffic. T...	3	10	
7	Add a subscription form to all the main pages...	10	7	
6	Show banners with current offers and sales on ...	5	3	
2	Add product recommendation blocks to the store...	8	3	
1	Launch your own delivery service. This will sh...	2	5	
5	Add a customer review page. This will increase...	3	2	
3	Change the category structure. This will incre...	8	3	
4	Change the background color on the main page. ...	3	1	

	confidence	effort	ICE
8	9	5	16.200000
0	8	6	13.333333
7	8	5	11.200000
6	8	3	8.000000
2	7	3	7.000000
1	4	10	2.000000
5	2	3	1.333333
3	3	8	1.125000
4	1	1	1.000000

```
[10]: plt.figure(figsize=(10, 6))
sns.barplot(x='hypothesis', y='ICE', data=hipotesis_ice.head(5))
plt.title('Top 5 Hipótesis por Puntuación ICE')
plt.xticks(rotation=45)
plt.show()
```



### 2.3.2 RICE

```
[11]: hipotesis['RICE'] = (hipotesis['reach'] * hipotesis['impact'] *
    ↪ hipotesis['confidence']) / hipotesis['effort']

hipotesis_rice = hipotesis.sort_values('RICE', ascending=False)
hipotesis_rice
```

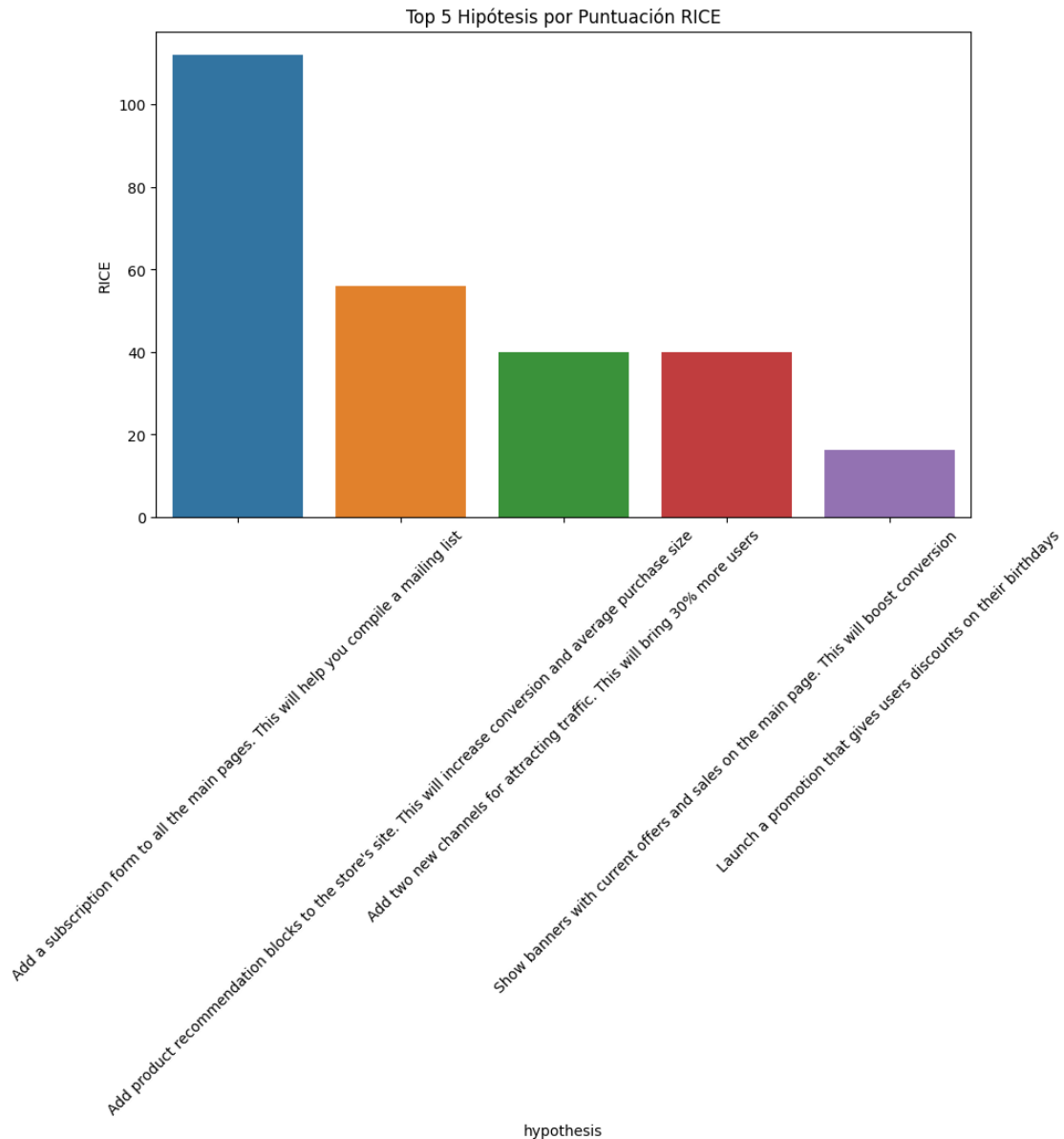
```
[11]:
```

	hipotesis	reach	impact	\
7	Add a subscription form to all the main pages...	10	7	
2	Add product recommendation blocks to the store...	8	3	
0	Add two new channels for attracting traffic. T...	3	10	

6	Show banners with current offers and sales on ...	5	3
8	Launch a promotion that gives users discounts ...	1	9
3	Change the category structure. This will incre...	8	3
1	Launch your own delivery service. This will sh...	2	5
5	Add a customer review page. This will increase...	3	2
4	Change the background color on the main page. ...	3	1

	confidence	effort	ICE	RICE
7	8	5	11.200000	112.0
2	7	3	7.000000	56.0
0	8	6	13.333333	40.0
6	8	3	8.000000	40.0
8	9	5	16.200000	16.2
3	3	8	1.125000	9.0
1	4	10	2.000000	4.0
5	2	3	1.333333	4.0
4	1	1	1.000000	3.0

```
[12]: plt.figure(figsize=(10, 6))
sns.barplot(x='hypothesis', y='RICE', data=hipotesis_rice.head(5))
plt.title('Top 5 Hipótesis por Puntuación RICE')
plt.xticks(rotation=45)
plt.show()
```



Se usaron dos metodos para tomar una decision de cual de las hipotesis seria mas conveniente de probar, entonces, ICE se centra en el impacto a corto plazo, la confianza en la hipótesis y el esfuerzo requerido. Prioriza las ideas que tienen un alto potencial de éxito y son relativamente fáciles de implementar, sin embargo si compramos el esfuerzo de metodo ICE, son pruebas que seria mas costosas para la empresa. Por otro lado, RICE añade el alcance al análisis, considerando cuántos usuarios se verán afectados por la hipótesis. Esto significa que una hipótesis con un impacto menor pero que alcance a muchos más usuarios podría tener una puntuación RICE más alta que otra con un impacto mayor pero un alcance más limitado.

Entonces, como el interes es alcanzar a una audiencia más amplia (ingresos), RICE puede ser una mejor opciones hacia hipótesis que pueden ayudar a aumentar los ingresos, por ende, sera mejor

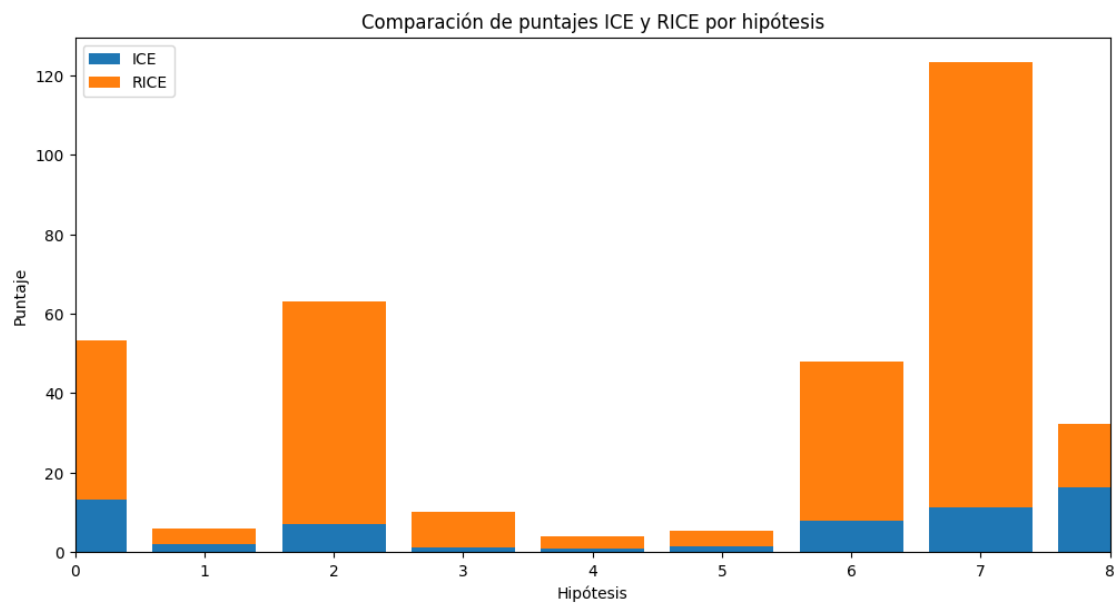
aplicar la hipótesis mejor puntuada del metodo RICE, ya que podria tener un mejor alcance, es decir, la cantidad de usuarios que seran afectados. y ademas, tendria un costo menor.

### 2.3.3 Grafico para comparar ICE y RICE

```
[13]: plt.figure(figsize=(12, 6))
plt.bar(hipotesis.index, hipotesis['ICE'], label='ICE')
plt.bar(hipotesis.index, hipotesis['RICE'], bottom=hipotesis['ICE'],
        label='RICE')

plt.xlabel('Hipótesis')
plt.xlim(0,8)
plt.ylabel('Puntaje')
plt.title('Comparación de puntajes ICE y RICE por hipótesis')
plt.legend()

plt.show()
```



## 2.4 Analisis de prueba A/B

### 2.4.1 Ingreso acumulado por grupos A/B

```
[14]: print(pedidos.head(5))
print(visitas.head(5))

grupos = pedidos[['date', 'group']].drop_duplicates()
```



```

#datos diarios acumulados agregados sobre los pedidos
agregar_pedidos = grupos.apply(lambda x: pedidos[np.logical_and(pedidos['date'] ≤
    ≤ x['date'], pedidos['group'] == x['group'])].agg({'date' : 'max', 'group' :
    → 'max', 'transaction_id' : pd.Series.nunique, 'visitor_id' : pd.Series.
    → nunique, 'revenue' : 'sum'}), axis=1).sort_values(by=['date', 'group'])

#datos diarios acumulados agregados sobre los visitantes
agregar_visitas= grupos.apply(lambda x: visitas[np.logical_and(visitas['date'] ≤
    ≤ x['date'], visitas['group'] == x['group'])].agg({'date' : 'max', 'group' :
    → 'max', 'visits' : 'sum'}), axis=1).sort_values(by=['date', 'group'])

# fusion de las dos tablas
fusion = agregar_pedidos.merge(agregar_visitas, left_on=['date', 'group'],
    → right_on=['date', 'group'])
fusion.columns = ['date', 'group', 'orders', 'buyers', 'revenue', 'visits']

print(fusion.head(5))

```

	transaction_id	visitor_id	date	revenue	group
0	3667963787	3312258926	2019-08-15	30.4	B
1	2804400009	3642806036	2019-08-15	15.2	B
2	2961555356	4069496402	2019-08-15	10.2	A
3	3797467345	1196621759	2019-08-15	155.1	B
4	2282983706	2322279887	2019-08-15	40.5	B

	date	group	visits
0	2019-08-01	A	719
1	2019-08-02	A	619
2	2019-08-03	A	507
3	2019-08-04	A	717
4	2019-08-05	A	756

	date	group	orders	buyers	revenue	visits
0	2019-08-01	A	24	20	2356.8	719
1	2019-08-01	B	21	20	1620.0	713
2	2019-08-02	A	44	38	3860.3	1338
3	2019-08-02	B	45	43	4243.8	1294
4	2019-08-03	A	68	62	5675.5	1845

```

[15]: # DataFrame con pedidos acumulados e ingresos acumulados por día, grupo A
ingreso_acumulado_A = fusion[fusion['group']=='A'][['date', 'revenue', 'orders']]

# DataFrame con pedidos acumulados e ingresos acumulados por día, grupo B
ingreso_acumulado_B = fusion[fusion['group']=='B'][['date', 'revenue', 'orders']]

# gráfico de ingresos del grupo A
plt.plot(ingreso_acumulado_A['date'], ingreso_acumulado_A['revenue'], label='A')

# gráfico de ingresos del grupo B

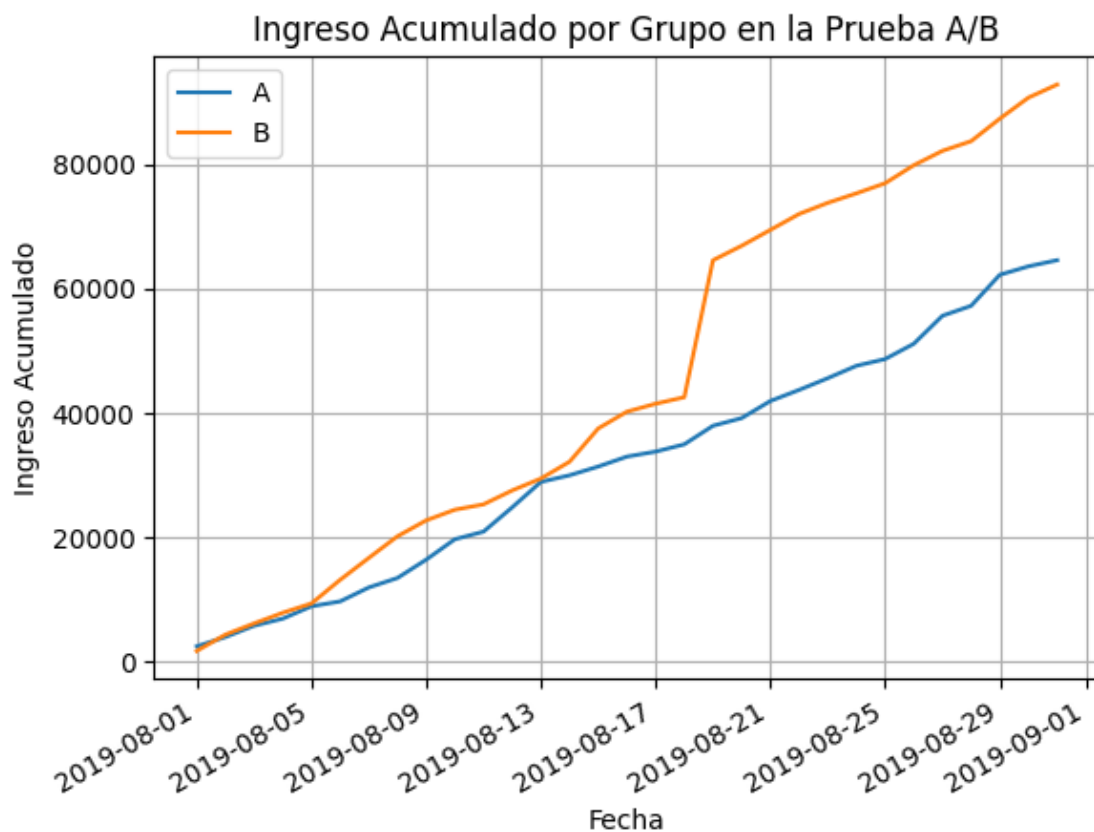
```

```
plt.plot(ingreso_acumulado_B['date'], ingreso_acumulado_B['revenue'], label='B')

plt.title('Ingreso Acumulado por Grupo en la Prueba A/B')
plt.xlabel('Fecha')
plt.ylabel('Ingreso Acumulado')
plt.legend(title='Grupo')
plt.grid(True)
plt.gca().xaxis.set_major_formatter(plt.matplotlib.dates.
    ↳DateFormatter('%Y-%m-%d'))
plt.gcf().autofmt_xdate()

plt.legend()
```

[15]: <matplotlib.legend.Legend at 0x7f97795be850>



El grupo B está generando más ingresos al menos hasta el punto de datos final del gráfico, el grupo B ha superado al grupo A en términos de ingresos acumulados. Existe una diferencia significativa entre los grupos, la distancia entre las líneas de los dos grupos sugiere una diferencia en los ingresos.

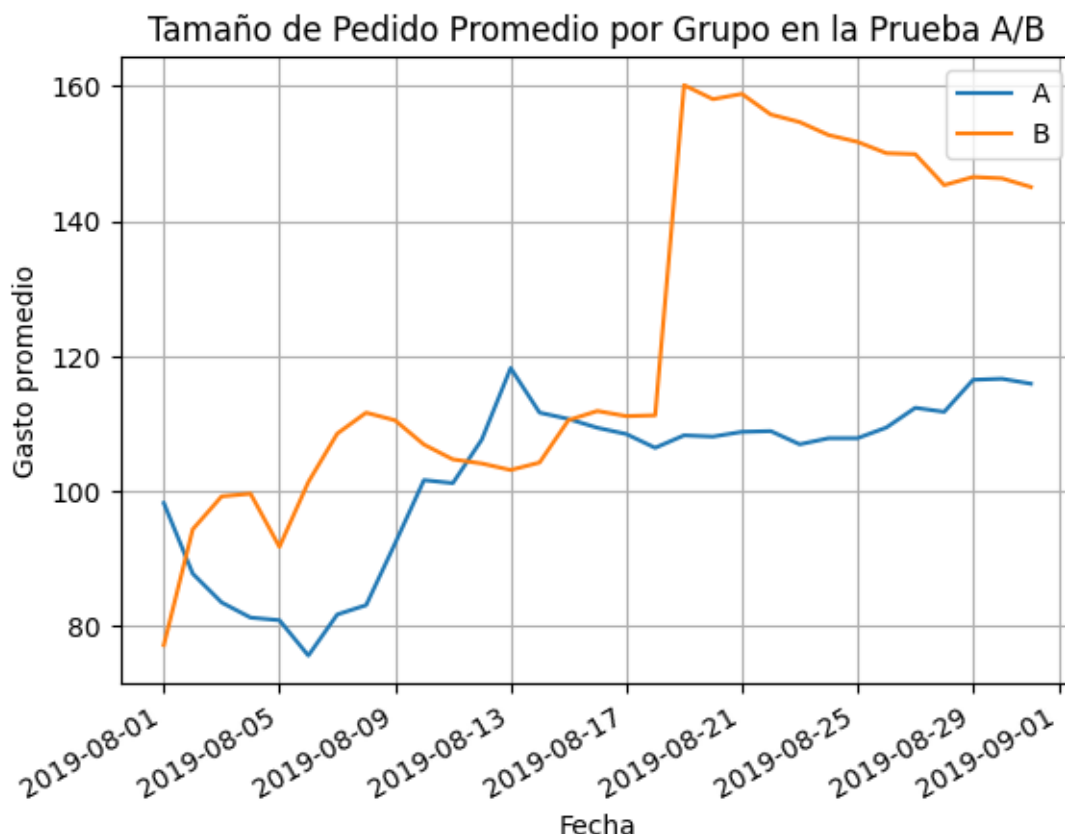
El grupo B podría ser una mejor opción, como el objetivo principal de la prueba A/B es maximizar

los ingresos, los resultados preliminares sugieren que la variante B podría ser más efectiva.

#### 2.4.2 Tamaño de pedido promedio (por día) por grupo A/B

```
[16]: ingreso_acumulado_A= fusion[fusion['group'] == 'A'][['date', 'revenue',  
↳ 'orders']]  
ingreso_acumulado_B = fusion[fusion['group'] == 'B'][['date', 'revenue',  
↳ 'orders']]  
  
plt.plot(ingreso_acumulado_A['date'], ingreso_acumulado_A['revenue']/  
↳ ingreso_acumulado_A['orders'], label='A')  
plt.plot(ingreso_acumulado_B['date'], ingreso_acumulado_B['revenue']/  
↳ ingreso_acumulado_B['orders'], label='B')  
  
plt.title('Tamaño de Pedido Promedio por Grupo en la Prueba A/B')  
plt.xlabel('Fecha')  
plt.ylabel('Gasto promedio')  
plt.legend(title='Grupo')  
plt.grid(True)  
plt.gca().xaxis.set_major_formatter(plt.matplotlib.dates.  
↳ DateFormatter('%Y-%m-%d'))  
plt.gcf().autofmt_xdate()  
  
plt.legend()
```

```
[16]: <matplotlib.legend.Legend at 0x7f9779538c40>
```



A pesar de las fluctuaciones, podemos observar algunas tendencias generales:

El grupo B comienza con un tamaño de pedido promedio significativamente menor que el grupo A, pero experimenta un crecimiento rápido en los primeros días de la prueba, hacia mediados del mes, el grupo B alcanza un pico y luego comienza a disminuir gradualmente, el grupo A, por su parte, mantiene un tamaño de pedido promedio más estable a lo largo de todo el período, con algunas fluctuaciones menores, al final del período, ambos grupos convergen hacia un tamaño de pedido promedio similar.

Conjeturas:

El crecimiento inicial del grupo B podría deberse a un efecto novedad. Los usuarios del grupo B, al ser expuestos a una nueva experiencia o diseño, podrían haber realizado compras de mayor valor y La disminución gradual del tamaño promedio del pedido en el grupo B podría indicar que los usuarios se han adaptado a la nueva experiencia y sus compras han vuelto a niveles más normales, sin embargo, es posible que existan segmentos de usuarios dentro de cada grupo que respondan de manera diferente a las variaciones de la prueba. Por ejemplo, usuarios nuevos versus usuarios recurrentes, o usuarios que realizan compras de mayor o menor valor.

### 2.4.3 Diferencia relativa en el tamaño de pedido promedio para el grupo B en comparación con el grupo A

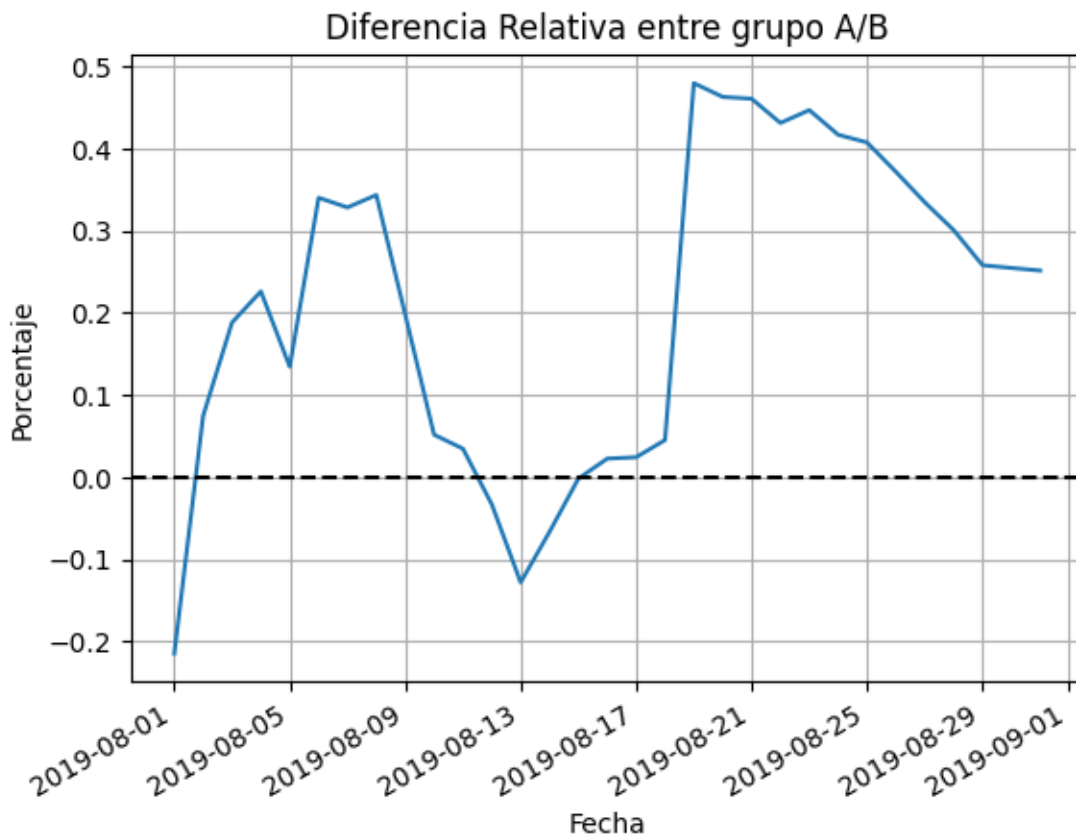
```
[17]: merge_ingreso_acumulado = ingreso_acumulado_A.merge(ingreso_acumulado_B,
    ↪left_on='date', right_on='date', how='left', suffixes=['A', 'B'])

# gráfico de diferencia relativa para los tamaños de compra promedio
plt.plot(merge_ingreso_acumulado['date'], (merge_ingreso_acumulado['revenueB']/
    ↪merge_ingreso_acumulado['ordersB'])/(merge_ingreso_acumulado['revenueA']/
    ↪merge_ingreso_acumulado['ordersA'])-1)

plt.title('Diferencia Relativa entre grupo A/B')
plt.xlabel('Fecha')
plt.ylabel('Porcentaje')
plt.grid(True)
plt.gca().xaxis.set_major_formatter(plt.matplotlib.dates.
    ↪DateFormatter('%Y-%m-%d'))
plt.gcf().autofmt_xdate()

plt.axhline(y=0, color='black', linestyle='--')
```

[17]: <matplotlib.lines.Line2D at 0x7f97794a5c10>



El grupo B ha tenido un desempeño superior en términos de tamaño promedio del pedido durante gran parte del período. Sin embargo, esta ventaja se ha reducido significativamente hacia el final de la prueba.

#### 2.4.4 Tasa de conversion para cada grupo

```
[18]: fusion['conversion'] = fusion['orders']/fusion['visits']

# datos en el grupo A
ingreso_acumulado_A = fusion[fusion['group']=='A']

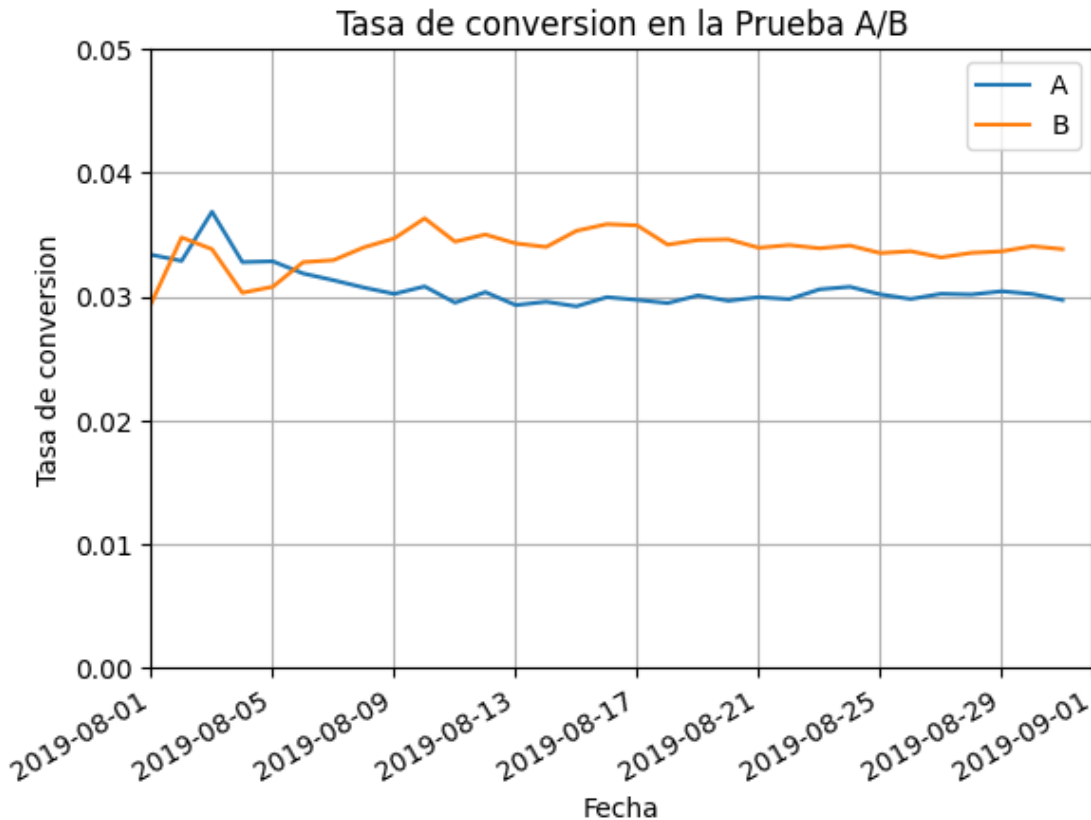
# datos en el grupo B
ingreso_acumulado_B = fusion[fusion['group']=='B']

plt.plot(ingreso_acumulado_A['date'], ingreso_acumulado_A['conversion'],
         ↪label='A')
plt.plot(ingreso_acumulado_B['date'], ingreso_acumulado_B['conversion'],
         ↪label='B')
plt.axis([pd.to_datetime('2019-08-01'), pd.to_datetime('2019-09-01'), 0, 0.05])

plt.title('Tasa de conversion en la Prueba A/B')
plt.xlabel('Fecha')
plt.ylabel('Tasa de conversion')
plt.legend(title='Grupo')
plt.grid(True)
plt.gca().xaxis.set_major_formatter(plt.matplotlib.dates.
    ↪DateFormatter('%Y-%m-%d'))
plt.gcf().autofmt_xdate()

plt.legend()
```

```
[18]: <matplotlib.legend.Legend at 0x7f9779494cd0>
```



A pesar de las pequeñas diferencias iniciales, ambos grupos mostraron un comportamiento similar a lo largo del tiempo. Esto podría indicar que los cambios introducidos no tuvieron un impacto significativo en la tasa de conversión de uno u otro grupo. el grupo B parece tener una tasa de conversión ligeramente superior al grupo A. Sin embargo, esta diferencia se reduce a medida que avanza el tiempo, y al final del período, ambas líneas se encuentran muy cercanas.

#### 2.4.5 Tasa de conversion diaria de los grupos

```
[19]: merge_acumulado_conversion = ingreso_acumulado_A[['date', 'conversion']].
      ↪ merge(ingreso_acumulado_B[['date', 'conversion']], left_on='date',
      ↪ right_on='date', how='left', suffixes=['A', 'B'])

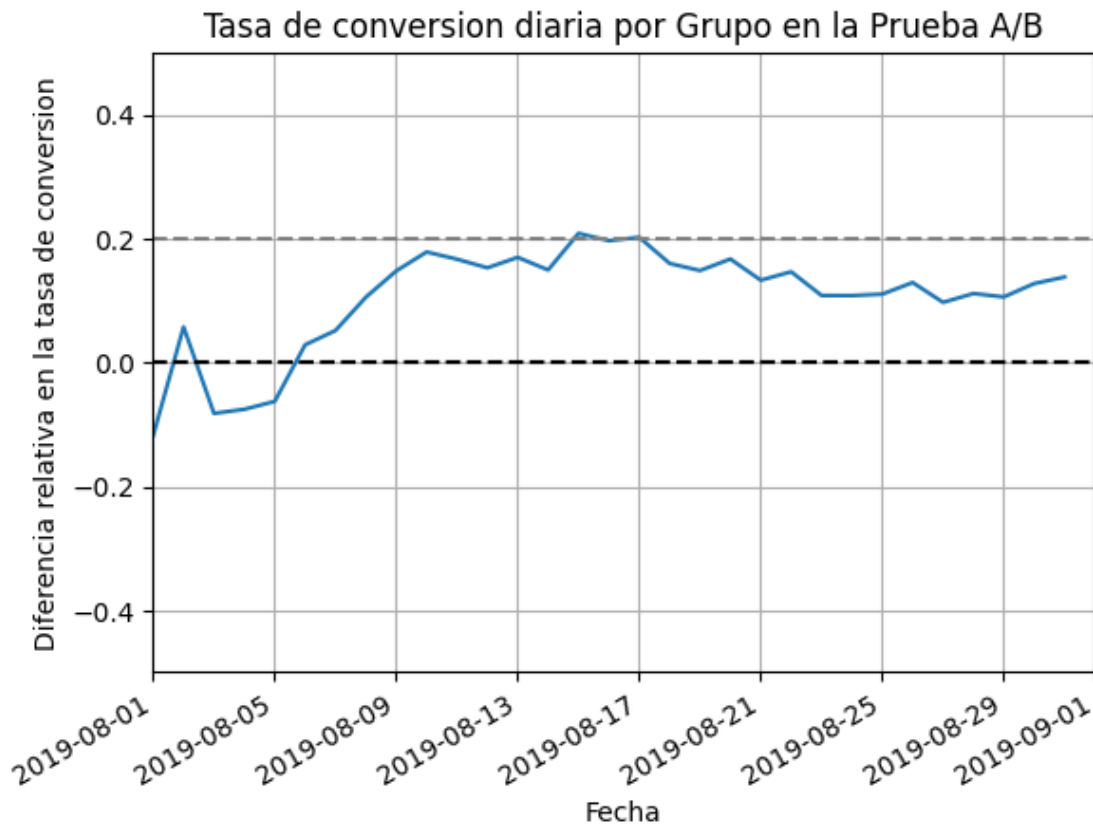
plt.plot(merge_acumulado_conversion['date'],
      ↪ merge_acumulado_conversion['conversionB']/
      ↪ merge_acumulado_conversion['conversionA']-1)
plt.axhline(y=0, color='black', linestyle='--')
plt.axhline(y=0.2, color='grey', linestyle='--')

plt.title('Tasa de conversion diaria por Grupo en la Prueba A/B')
```

```
plt.xlabel('Fecha')
plt.ylabel('Diferencia relativa en la tasa de conversion')
plt.grid(True)
plt.gca().xaxis.set_major_formatter(plt.matplotlib.dates.
    ↪DateFormatter('%Y-%m-%d'))
plt.gcf().autofmt_xdate()

plt.axis([pd.to_datetime('2019-08-01'), pd.to_datetime('2019-09-01'), -0.5, 0.
    ↪5])
```

[19]: (18109.0, 18140.0, -0.5, 0.5)



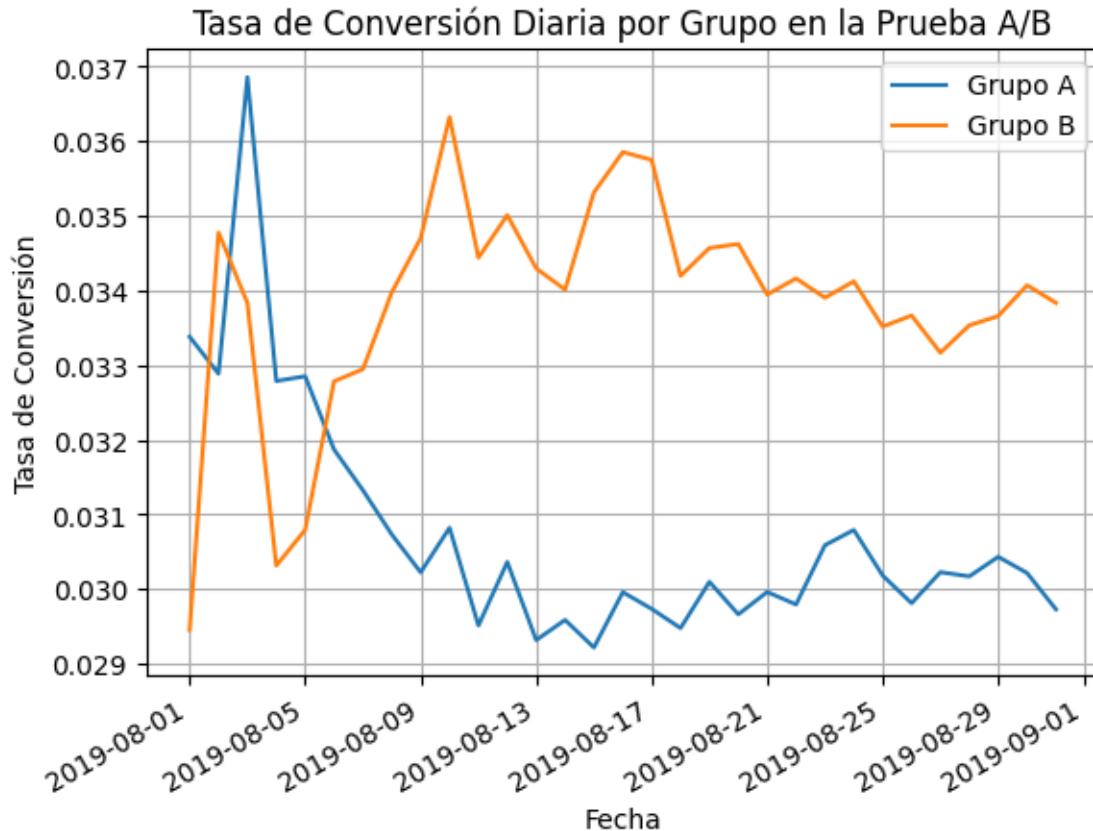
La línea que representa la diferencia relativa presenta muchas fluctuaciones, lo que indica que la relación entre las tasas de conversión de ambos grupos varía significativamente de un día a otro. A pesar de las fluctuaciones, parece haber una tendencia general a que la tasa de conversión del grupo B sea ligeramente inferior a la del grupo A. Esto se observa porque la línea se encuentra principalmente por debajo del eje horizontal (que representa una diferencia relativa de 0).

No hay una diferencia clara y consistente entre los grupos, las fluctuaciones son demasiado grandes para afirmar con certeza que un grupo es superior al otro.



```
[20]: # Grafico con tasas de conversión individuales y diferencia relativa
plt.plot(merge_acumulado_conversion['date'],
         ↪merge_acumulado_conversion['conversionA'], label='Grupo A')
plt.plot(merge_acumulado_conversion['date'],
         ↪merge_acumulado_conversion['conversionB'], label='Grupo B')

plt.title('Tasa de Conversión Diaria por Grupo en la Prueba A/B')
plt.xlabel('Fecha')
plt.ylabel('Tasa de Conversión')
plt.legend()
plt.grid(True)
plt.gca().xaxis.set_major_formatter(plt.matplotlib.dates.
         ↪DateFormatter('%Y-%m-%d'))
plt.gcf().autofmt_xdate()
plt.show()
```



Ambos grupos muestran una cierta variabilidad en la tasa de conversión a lo largo del tiempo. Sin embargo, parece que el grupo B tiene una tendencia ligeramente ascendente, mientras que el grupo A se mantiene más estable. En general, el grupo B parece tener una tasa de conversión ligeramente superior al grupo A, especialmente hacia el final del período. Aunque las diferencias

no son muy grandes, el grupo B parece tener una tasa de conversión ligeramente superior al grupo A, especialmente hacia el final del período.

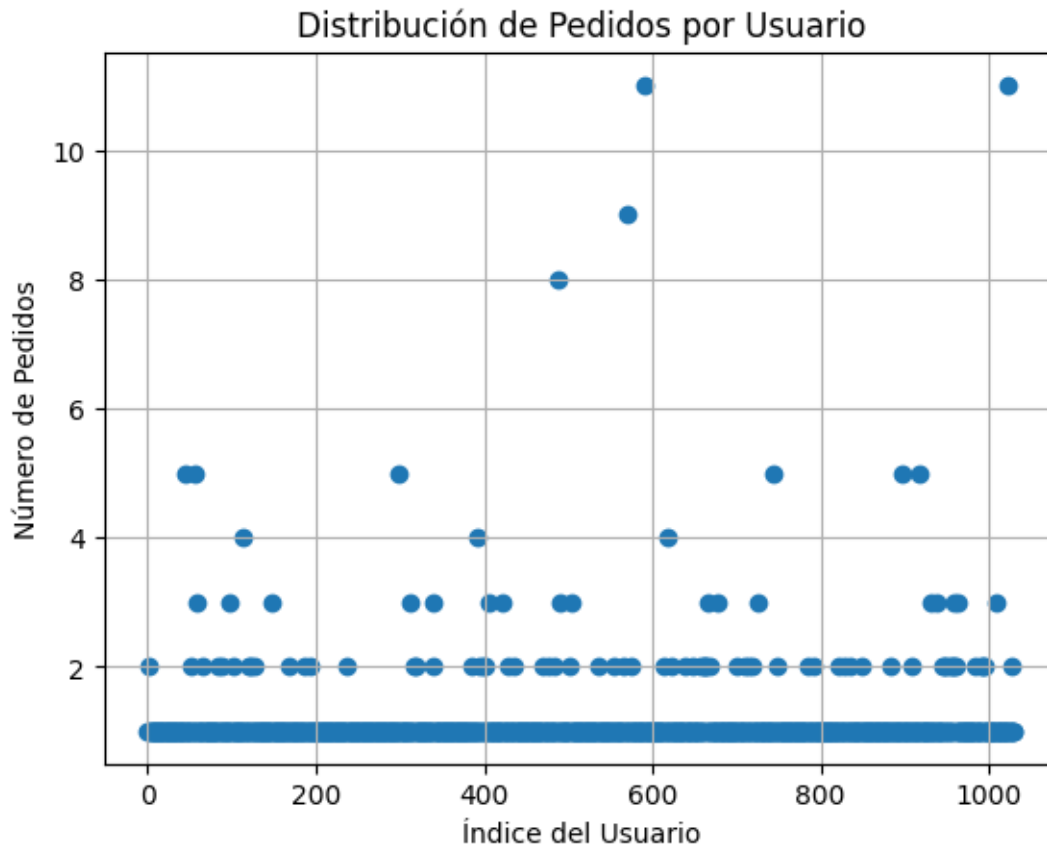
#### 2.4.6 Número de pedidos por usuario.

```
[21]: pedidos_por_usuario = (pedidos.drop(['group', 'revenue', 'date'], axis=1)
    .groupby('visitor_id', as_index=False)
    .agg({'transaction_id': pd.Series.nunique}))

pedidos_por_usuario.columns = ['visitor_id', 'orders']

print(pedidos_por_usuario.sort_values(by='orders', ascending=False).head(10))
plt.scatter(pedidos_por_usuario.index, pedidos_por_usuario['orders'])
plt.xlabel('Índice del Usuario')
plt.ylabel('Número de Pedidos')
plt.title('Distribución de Pedidos por Usuario')
plt.grid(True)
plt.show()
```

	visitor_id	orders
1023	4256040402	11
591	2458001652	11
569	2378935119	9
487	2038680547	8
44	199603092	5
744	3062433592	5
55	237748145	5
917	3803269165	5
299	1230306981	5
897	3717692402	5



El análisis de la distribución de pedidos por usuario revela una gran heterogeneidad en el comportamiento de compra de los usuarios. La mayoría de los usuarios realizan un número reducido de pedidos, lo que sugiere oportunidades para incrementar la frecuencia de compra. Sin embargo, un pequeño pero significativo grupo de usuarios realiza un número significativamente mayor de pedidos. De hecho, el 95% de los usuarios realiza menos de 2 pedidos, mientras que el 99% realiza menos de 4 pedidos, lo que indica la presencia de una larga cola en la distribución y la existencia de un segmento de clientes altamente valiosos

```
[22]: print(np.percentile(pedidos_por_usuario['orders'], [ 95, 99]))
```

```
[2. 4.]
```

#### 2.4.7 Precios de pedidos

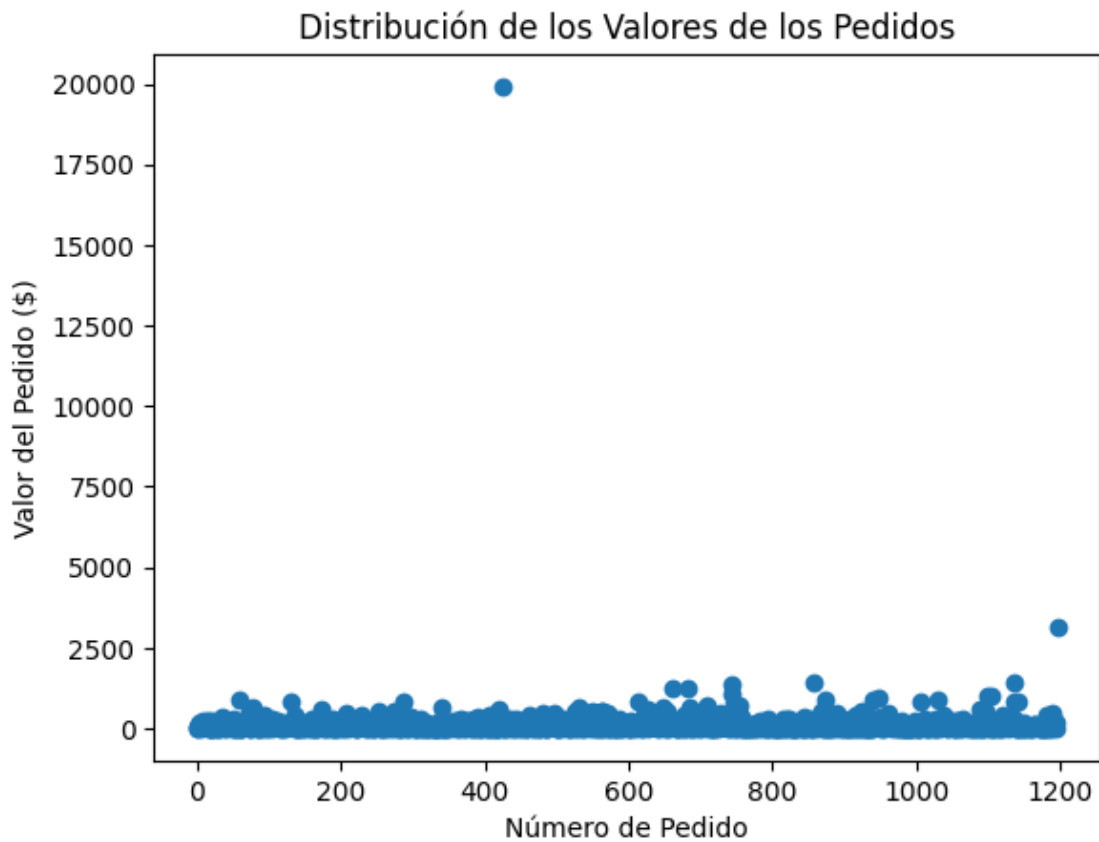
```
[23]: print(pedidos.sort_values(by='revenue',ascending=False).head(10))
```

```
x_values = pd.Series(range(0,len(pedidos['revenue'])))
plt.scatter(x_values, pedidos['revenue'])
plt.xlabel('Número de Pedido')
plt.ylabel('Valor del Pedido ($)')
```

```
plt.title('Distribución de los Valores de los Pedidos')
```

	transaction_id	visitor_id	date	revenue	group
425	590470918	1920142716	2019-08-19	19920.4	B
1196	3936777065	2108080724	2019-08-15	3120.1	B
858	192721366	1316129916	2019-08-27	1450.2	A
1136	666610489	1307669133	2019-08-13	1425.8	A
744	3668308183	888512513	2019-08-27	1335.6	B
662	1811671147	4266935830	2019-08-29	1220.2	A
682	1216533772	4266935830	2019-08-29	1220.2	B
743	3603576309	4133034833	2019-08-09	1050.0	A
1103	1348774318	1164614297	2019-08-12	1025.8	A
1099	316924019	148427295	2019-08-12	1015.9	A

```
[23]: Text(0.5, 1.0, 'Distribución de los Valores de los Pedidos')
```



```
[24]: print(np.percentile(pedidos['revenue'], [95, 99]))
```

```
[435.54  900.904]
```

El análisis de los percentiles revela que el 95% de los pedidos tienen un valor inferior a 435.6,

mientras que el 99% se encuentra por debajo de 900.9. Esto confirma la concentración de los valores en la parte baja de la distribución. La presencia de pocos valores atípicos por encima de estos percentiles indica la existencia de un pequeño grupo de clientes que realizan compras de mayor valor.

```
[25]: Q1 = pedidos['revenue'].quantile(0.25)
      Q3 = pedidos['revenue'].quantile(0.75)
      IQR = Q3 - Q1

      # límites
      lower_limite = Q1 - 1.5 * IQR
      upper_limite = Q3 + 1.5 * IQR

      print("Rango intercuartílico (IQR):", IQR)
      print("Límite inferior:", lower_limite)
      print("Límite superior:", upper_limite)

      # valores atípicos
      outliers = pedidos[(pedidos['revenue'] < lower_limite) | (pedidos['revenue'] >
      ↪ upper_limite)]
      print("Valores atípicos:\n", outliers)

      # gráfico de caja
      sns.boxplot(x=pedidos['revenue'])
      plt.title('Gráfico de Caja de Ingresos por Pedido')
      plt.xlabel('Ingresos')
      plt.show()
```

Rango intercuartílico (IQR): 109.60000000000001

Límite inferior: -143.6

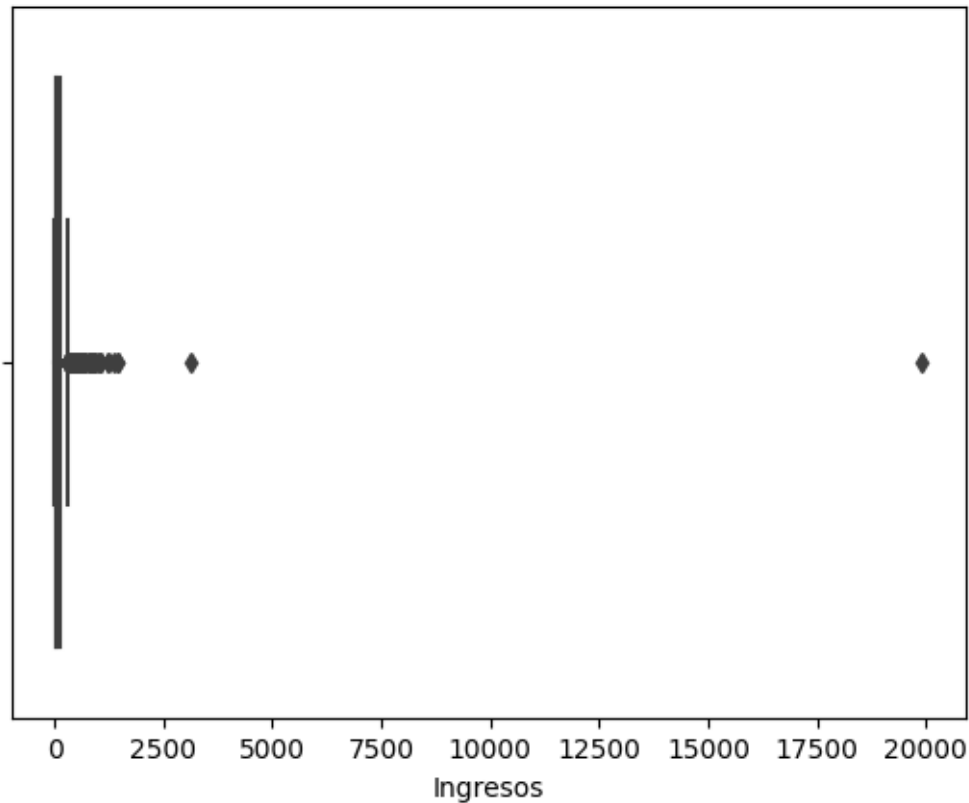
Límite superior: 294.8

Valores atípicos:

	transaction_id	visitor_id	date	revenue	group
34	2153010762	834683205	2019-08-15	355.5	A
38	3276214732	1110007955	2019-08-15	300.3	B
60	1421016313	4256040402	2019-08-16	875.5	B
69	890243194	3055165772	2019-08-16	300.3	A
73	1295850747	2947100995	2019-08-01	455.8	A
...	...	...	...	...	...
1137	2183069967	759473111	2019-08-13	810.4	A
1142	3815014015	2606908679	2019-08-13	815.8	B
1181	1303723408	3218863961	2019-08-14	390.5	B
1189	4082620617	393266494	2019-08-14	490.4	B
1196	3936777065	2108080724	2019-08-15	3120.1	B

[112 rows x 5 columns]

Gráfico de Caja de Ingresos por Pedido



Dado que hay algunos valores atípicos muy altos en comparación con el resto de los datos, es probable que estos estén sesgando los resultados.

#### 2.4.8 Significancia estadística de la diferencia en la conversión entre los grupos

```
[26]: pedidos_por_usuario_A = (pedidos[pedidos['group'] == 'A']
    .groupby('visitor_id', as_index=False)
    .agg({'transaction_id': pd.Series.nunique}))

pedidos_por_usuario_A.columns = ['visitor_id', 'orders']

pedidos_por_usuario_B = (pedidos[pedidos['group'] == 'B']
    .groupby('visitor_id', as_index=False)
    .agg({'transaction_id': pd.Series.nunique}))

pedidos_por_usuario_B.columns = ['visitor_id', 'orders']

muestra_A = pd.concat([pedidos_por_usuario_A['orders'], pd.Series(0, index=np.
    ↳ arange
```

```

↪(visitas[visitas['group'] == 'A']['visits'].sum()
↪len(pedidos_por_usuario_A['orders'])),name='orders',),],axis=0,)

muestra_B = pd.concat([pedidos_por_usuario_B['orders'],pd.Series(0,index=np.
↪arange
↪(visitas[visitas['group'] == 'B']['visits'].sum()
↪len(pedidos_por_usuario_B['orders']) ),name='orders',),],axis=0,)

p_valor = stats.mannwhitneyu(muestra_A, muestra_B)[1]
print(f"El valor p obtenido es de {p_valor:.3f}. Un valor p menor a 0.05
↪sugiere que existe una diferencia estadísticamente significativa entre los
↪grupos.")

diferencia_medias = muestra_B.mean() / muestra_A.mean() - 1
print(f"La diferencia relativa en el promedio de pedidos entre los grupos es de
↪{diferencia_medias:.3f}. Esto indica que el grupo B tiene, en promedio, un
↪{diferencia_medias*100:.2f}% más de pedidos que el grupo A.")

```

El valor p obtenido es de 0.017. Un valor p menor a 0.05 sugiere que existe una diferencia estadísticamente significativa entre los grupos.

La diferencia relativa en el promedio de pedidos entre los grupos es de 0.138. Esto indica que el grupo B tiene, en promedio, un 13.81% más de pedidos que el grupo A.

Con un valor p de 0.017, este valor indica la probabilidad de obtener una diferencia tan extrema o más extrema si no existiera una diferencia real entre los grupos. Un p-valor menor a 0.05 sugiere que podemos rechazar la hipótesis nula de que no hay diferencia entre los grupos y concluir que existe una diferencia estadísticamente significativa.

Por otro lado la diferencia en las medias de 0.138 nos indica que, en promedio, los usuarios del grupo B realizaron un 13.8% más de pedidos que los usuarios del grupo A.

Basándonos en los resultados obtenidos, podemos concluir que existe una diferencia estadísticamente significativa en la tasa de conversión entre los grupos A y B. Los usuarios del grupo B tienen una probabilidad significativamente mayor de realizar una compra en comparación con los usuarios del grupo A.

## 2.4.9 Significancia estadística de la diferencia en el tamaño promedio de pedido entre los grupos

```

[27]: p_valor = stats.mannwhitneyu(pedidos[pedidos['group']=='A']['revenue'],
↪pedidos[pedidos['group']=='B']['revenue'])[1]

```

```

print(f"El valor p obtenido es de {p_valor:.3f}. Un valor p menor a 0.05
↳ sugiere que no existe una diferencia estadísticamente significativa entre
↳ los grupos.")

diferencia_medias = pedidos[pedidos['group']=='B']['revenue'].mean()/
↳ pedidos[pedidos['group']=='A']['revenue'].mean()-1
print(f"La diferencia relativa en el promedio de pedidos entre los grupos es de
↳ {diferencia_medias:.3f}. Esto indica que el grupo B tiene, en promedio,
↳ {diferencia_medias*100:.2f}% de pedidos mas grandes que el grupo A.")

```

El valor p obtenido es de 0.692. Un valor p menor a 0.05 sugiere que no existe una diferencia estadísticamente significativa entre los grupos.

La diferencia relativa en el promedio de pedidos entre los grupos es de 0.252. Esto indica que el grupo B tiene, en promedio, 25.17% de pedidos mas grandes que el grupo A.

En este caso, con un p-valor de 0.692, indica que hay una probabilidad del 69.2% de obtener una diferencia tan extrema o más extrema si no existiera una diferencia real entre los grupos. Un p-valor tan alto sugiere que no podemos rechazar la hipótesis nula. Es decir, no hay evidencia suficiente para concluir que exista una diferencia estadísticamente significativa en el tamaño promedio del pedido entre los grupos A y B.

La diferencia en las medias de 0.252 nos indica que, en promedio, los pedidos del grupo B son un 25.2% más grandes que los pedidos del grupo A. Sin embargo, dado el alto p-value, esta diferencia no es estadísticamente significativa. En conclusión y basados en los resultados, podemos concluir que no hay evidencia suficiente para afirmar que el cambio realizado en uno de los grupos haya tenido un impacto significativo en el tamaño promedio del pedido.

```

[28]: usuarios_con_muchos_pedidos = pd.
↳ concat([pedidos_por_usuario_A[pedidos_por_usuario_A['orders'] >
↳ 4]['visitor_id'],
↳
↳ pedidos_por_usuario_B[pedidos_por_usuario_B['orders'] > 4]['visitor_id']],
↳ axis = 0)

usuarios_con_pedidos_costosos = pedidos[pedidos['revenue'] > 700]['visitor_id']

anomalias = pd.concat([usuarios_con_muchos_pedidos,
↳ usuarios_con_pedidos_costosos], axis = 0).drop_duplicates().sort_values()

print(anomalias.head(5))
print(anomalias.shape)

```

```

1099    148427295
1137    759473111
949     887908475
744     888512513
709     950626008

```



Name: visitor\_id, dtype: int64  
(24,)

En total hay 24 usuarios anómalos que realizaron más de un pedido o el costo de sus pedidos fueron mayor a 700\$

#### 2.4.10 Significancia estadística de la diferencia en la conversión entre los grupos (sin anomalías)

```
[29]: filtro_muestra_A = pd.concat([pedidos_por_usuario_A[np.
    ↪ logical_not(pedidos_por_usuario_A['visitor_id']
    ↪ isin(anomalias))] ['orders'],
                                   pd.Series(0, index=np.
    ↪ arange(visitas[visitas['group']=='A']['visits'].sum()
                                   - len(pedidos_por_usuario_A['orders'])), name='orders']], axis=0)

filtro_muestra_B = pd.concat([pedidos_por_usuario_B[np.logical_not(
    ↪ pedidos_por_usuario_B['visitor_id']
    ↪ isin(anomalias))] ['orders'],
                                   pd.Series(0, index=np.
    ↪ arange(visitas[visitas['group']=='B']['visits'].sum()
                                   - len(pedidos_por_usuario_B['orders'])), name='orders']], axis=0)

print("Valor p: {0:.3f} (Un valor p menor a 0.05 indica que la diferencia entre
    ↪ los grupos es estadísticamente significativa.)".format(stats.
    ↪ mannwhitneyu(filtro_muestra_A, filtro_muestra_B)[1]))
print(" {0:.3f}".format(filtro_muestra_B.mean()/filtro_muestra_A.mean()-1))
print("Diferencia de medias: El grupo B tiene, en promedio, un 16.3% más de
    ↪ conversiones que el grupo A. Esto sugiere que la modificación realizada en
    ↪ el grupo B ha tenido un impacto positivo en la tasa de conversión")
```

Valor p: 0.011 (Un valor p menor a 0.05 indica que la diferencia entre los grupos es estadísticamente significativa.)  
0.163

Diferencia de medias: El grupo B tiene, en promedio, un 16.3% más de conversiones que el grupo A. Esto sugiere que la modificación realizada en el grupo B ha tenido un impacto positivo en la tasa de conversión

Luego de filtrar los datos se obtuvo un p-valor de 0.011, esto sigue siendo menor a 0.05, lo que indica que aún existe una diferencia estadísticamente significativa en la tasa de conversión entre los grupos A y B, incluso después de eliminar los usuarios anómalos. Esto sugiere que la diferencia en la tasa de conversión no se debe únicamente a la presencia de outliers.

En el cálculo de diferencia en las medias 0.163, donde los usuarios del grupo B tienen, en promedio,

un 16.3% más de conversiones que los usuarios del grupo A.

En consecuencia la persistencia de una diferencia estadísticamente significativa en la tasa de conversión después de eliminar las anomalías refuerza la conclusión de que la modificación realizada en el grupo B tuvo un impacto positivo en la tasa de conversión. Si bien las anomalías no eran la causa principal de la diferencia en la tasa de conversión, su eliminación ha ayudado a obtener resultados más precisos y confiables.

#### 2.4.11 Significancia estadística de la diferencia en el tamaño promedio de pedido entre los grupos (sin anomalías)

```
[30]: # valor p calculado con la prueba U de Mann-Whitney
print("Valor p: {0:.3f}".format(stats.mannwhitneyu(pedidos[np.
    ↪ logical_and(pedidos['group']=='A',
                                                    np.
    ↪ logical_not(pedidos['visitor_id']
                                                    ↪
    ↪ .isin(anomalias)))]['revenue'],
                                pedidos[np.
    ↪ logical_and(pedidos['group']=='B',
                                                    np.
    ↪ logical_not(pedidos['visitor_id']
                                                    ↪
    ↪ .isin(anomalias)))]['revenue'])[1]))

# diferencia relativa en el ingreso medio entre los grupos B y A
print("Diferencia de medias: {0:.3f}".format(pedidos[np.
    ↪ logical_and(pedidos['group']=='B',
                                                    np.
    ↪ logical_not(pedidos['visitor_id']
                                                    ↪
    ↪ .isin(anomalias)))]['revenue'].mean() /
                                pedidos[np.logical_and(pedidos['group']=='A',
                                                    np.
    ↪ logical_not(pedidos['visitor_id'].isin(anomalias)))]['revenue'].mean() - 1))
```

Valor p: 0.637

Diferencia de medias: 0.047

Por último el valor P fue de 0.637, este valor es muy alto, lo que indica que no hay evidencia suficiente para rechazar la hipótesis nula, es decir, no existe una diferencia estadísticamente significativa en el tamaño promedio del pedido entre los grupos A y B, incluso después de eliminar las anomalías.

Y la diferencia en las medias 0.047 aunque hay una pequeña diferencia en los promedios, esta diferencia no es estadísticamente significativa debido al alto valor de p.

Entonces, no hay evidencia de que el tratamiento aplicado al grupo B haya tenido un impacto significativo en el tamaño promedio de los pedidos. La eliminación de las anomalías no ha modificado significativamente esta conclusión

## 2.5 Conclusion general

Basado en el análisis de la prueba A/B realizada, se puede concluir que la variante B ha demostrado ser más efectiva en términos de tasa de conversión. Los usuarios expuestos a la variante B mostraron una mayor probabilidad de realizar una compra en comparación con aquellos expuestos a la variante A. Aunque no se observó una diferencia estadísticamente significativa en el tamaño promedio de los pedidos, la variante B presentó un incremento en este indicador.

Una mayor tasa de conversión significa que un mayor porcentaje de visitantes están realizando compras. Esto se traduce directamente en un mayor número de clientes y, por lo tanto, en un mayor potencial de ingresos.

Recomendaciones:

Implementar la variante B dado el desempeño superior de la variante B en términos de tasa de conversión, se recomienda implementar esta versión en toda la plataforma para maximizar las ventas a corto plazo.

Es fundamental continuar monitoreando el desempeño de la variante B a lo largo del tiempo, realizando análisis periódicos para identificar posibles tendencias y ajustes necesarios. Explorar si existen diferencias significativas en el comportamiento de los usuarios en función de segmentos demográficos (edad, género, ubicación) o de comportamiento (frecuencia de compra, valor promedio del carrito) para personalizar aún más la experiencia de compra.

Continuar realizando pruebas A/B para evaluar nuevas variaciones de la variante B y explorar otras variables que puedan influir en la tasa de conversión, como el diseño de la página, el copywriting, la estructura de la información y la experiencia de pago. Basándose en los resultados de las pruebas y en los datos de seguimiento, implementar mejoras en la experiencia de usuario para reducir la tasa de abandono y aumentar la probabilidad de conversión. Esto puede incluir la simplificación del proceso de compra, la mejora de la velocidad de carga de la página y la personalización de las recomendaciones de productos.

Investigar los factores que pueden estar influyendo en el tamaño promedio del pedido, como la oferta de productos complementarios, la implementación de descuentos o promociones, y la personalización de las recomendaciones de productos.

Realizar un seguimiento a largo plazo del impacto de la implementación de la variante B en métricas clave como el valor de vida del cliente (CLV) y la tasa de retención.