4fda4bab-7623-4452-bd52-be7bf8307135-Copy1

November 26, 2024

1 Proyecto final. Coronavirus (SQL)

1. Introducción
2. Numero de libros publicados despues de 1 de enero del 2000
3. Número de reseñas de usuarios y la calificación promedio para cada libro
4. Editorial que ha publicado el mayor número de libros con más de 50 páginas
5. Autor que tiene la más alta calificación
6. Número promedio de reseñas de texto entre los usuarios que calificaron más de 50 libros
7. Conclusión general

1.1 Introducción

La pandemia de COVID-19 aceleró significativamente la adopción de hábitos de lectura digital. Ante este nuevo panorama, surgieron numerosas plataformas de libros electrónicos que compiten por captar la atención de los lectores. El presente proyecto tiene como objetivo analizar una base de datos de una de estas plataformas para identificar patrones de consumo, preferencias de los usuarios y oportunidades de mejora. A través de este análisis, se busca generar insights valiosos que permitan desarrollar una propuesta de valor diferenciada para un nuevo producto en este mercado altamente competitivo.

La industria editorial está experimentando una transformación digital sin precedentes. Para destacar en este mercado dinámico, es crucial comprender las necesidades y expectativas de los lectores. De esta forma aprovechar oportunidades para desarrollar un nuevo producto que ofrezca una experiencia de lectura más personalizada y satisfactoria. Al analizar datos sobre libros, autores, editoriales y reseñas de usuarios, se busca identificar nichos de mercado desatendidos y características que diferencien a un nuevo producto de la competencia.

[1]:
```
import pandas as pd
from sqlalchemy import create_engine
import seaborn as sns
import matplotlib.pyplot as plt
```

[2]:
```
db_config = {
 'user': 'practicum_student', # username
 'pwd': 'QnmDH8Sc2TQLvy2G3Vvh7', # password
 'host': 'yp-trainers-practicum.cluster-czs0gxyx2d8w.us-east-1.rds.amazonaws.
 ↪com',
 'port': 5432, # connection port
 'db': 'data-analyst-final-project-db' # the name of the database
```

```
 }
connection_string = 'postgresql://{}:{}@{}:{}/{}'.format(db_config['user'],

db_config['pwd'],

db_config['host'],
db_config['port'],

db_config['db'])

engine = create_engine(connection_string, connect_args={'sslmode':'require'})
```

[3]:
```
# Consulta primeras filas de cada tabla
query_books = "SELECT * FROM books LIMIT 5"
query_authors = "SELECT * FROM authors LIMIT 5"
query_publishers = "SELECT * FROM publishers LIMIT 5"
query_ratings = "SELECT * FROM ratings LIMIT 5"
query_reviews = "SELECT * FROM reviews LIMIT 5"

# Ejecutar las consultas y crear DataFrames
df_books = pd.read_sql(query_books, engine)
df_authors = pd.read_sql(query_authors, engine)
df_publishers = pd.read_sql(query_publishers, engine)
df_ratings = pd.read_sql(query_ratings, engine)
df_reviews = pd.read_sql(query_reviews, engine)

# Imprimir las primeras filas de cada DataFrame
print("Primeras 5 filas de la tabla books:")
print(df_books.head())

print("\nPrimeras 5 filas de la tabla authors:")
print(df_authors.head())

print("Primeras 5 filas de la tabla publishers:")
print(df_publishers.head())

print("\nPrimeras 5 filas de la tabla reviews:")
print(df_reviews.head())
```

```
Primeras 5 filas de la tabla books:
   book_id  author_id                                              title  \
0        1        546                                  'Salem's Lot
1        2        465                1 000 Places to See Before You Die
2        3        407  13 Little Blue Envelopes (Little Blue Envelope…
3        4         82  1491: New Revelations of the Americas Before C…
4        5        125                                               1776
```

```
   num_pages publication_date  publisher_id
0        594       2005-11-01            93
1        992       2003-05-22           336
2        322       2010-12-21           135
3        541       2006-10-10           309
4        386       2006-07-04           268

Primeras 5 filas de la tabla authors:
   author_id                          author
0          1                      A.S. Byatt
1          2  Aesop/Laura Harris/Laura Gibbs
2          3                 Agatha Christie
3          4                   Alan Brennert
4          5        Alan Moore/David   Lloyd
Primeras 5 filas de la tabla publishers:
   publisher_id                          publisher
0             1                                Ace
1             2                           Ace Book
2             3                          Ace Books
3             4                       Ace Hardcover
4             5  Addison Wesley Publishing Company

Primeras 5 filas de la tabla reviews:
   review_id  book_id       username  \
0          1        1   brandtandrea
1          2        1     ryanfranco
2          3        2       lorichen
3          4        3  johnsonamanda
4          5        3    scotttamara

                                                text
0  Mention society tell send professor analysis. …
1  Foot glass pretty audience hit themselves. Amo…
2  Listen treat keep worry. Miss husband tax but …
3  Finally month interesting blue could nature cu…
4  Nation purpose heavy give wait song will. List…
```

1.2 Numero de libros publicados despues de 1 de enero del 2000

[4]:
```
# Consulta para contar los libros publicados después del 1 de enero de 2000
query = """SELECT COUNT(*) AS num_libros_post_2000
FROM books
WHERE publication_date > '2000-01-01'"""

# Ejecutar la consulta y almacenar el resultado
publicaciones_2000 = pd.read_sql(query, engine)
```

```
# Mostrar el resultado
print("Número de libros publicados después del 1 de enero de 2000:")
print(publicaciones_2000)
```

```
Número de libros publicados después del 1 de enero de 2000:
   num_libros_post_2000
0                   819
```

El numero total de libros que fueron publicados despues del primero de enero del 2000 son 819 libros

1.3 Número de reseñas de usuarios y la calificación promedio para cada libro.

[5]:
```
# Consulta para obtener el número de reseñas y la calificación promedio por libro
query = """SELECT
    b.book_id,
    b.title,
    COUNT(r.rating_id) AS num_reseñas,
    AVG(r.rating) AS calificacion_promedio
FROM
    books b
INNER JOIN ratings r ON b.book_id = r.book_id
GROUP BY b.book_id, b.title;"""

# Ejecutar la consulta y almacenar el resultado
reseña_calificacion = pd.read_sql(query, engine)

# Mostrar el resultado
print(reseña_calificacion)
```

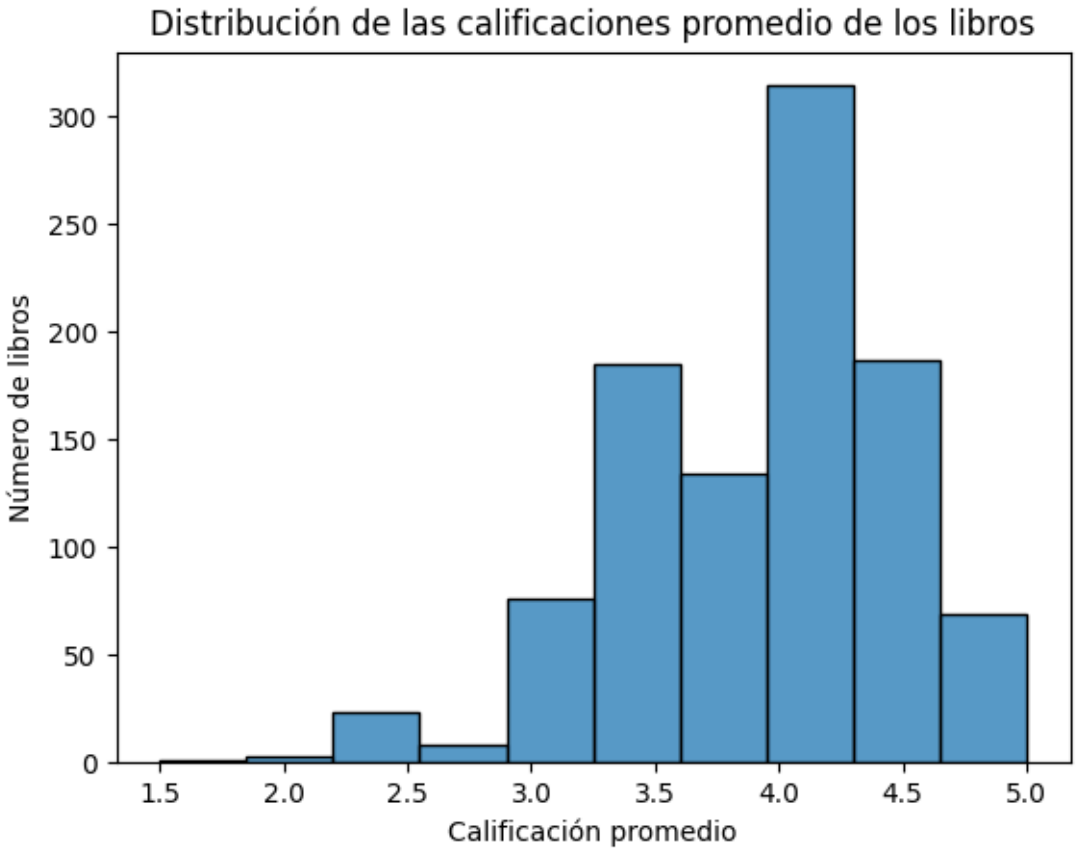
```
     book_id                                              title  num_reseñas  \
0        652          The Body in the Library (Miss Marple  #3)            2
1        273                                        Galápagos            2
2         51                            A Tree Grows in Brooklyn           12
3        951  Undaunted Courage: The Pioneering First Missio…            2
4        839                                       The Prophet            7
..       …                                                   …          …
995       64                                Alice in Wonderland           13
996       55          A Woman of Substance (Emma Harte Saga #1)            2
997      148                                        Christine            7
998      790  The Magicians' Guild (Black Magician Trilogy  #1)            2
999      828                            The Plot Against America            2

     calificacion_promedio
0                 4.500000
1                 4.500000
2                 4.250000
```

```
3                4.000000
4                4.285714
..                    …
995              4.230769
996              5.000000
997              3.428571
998              3.500000
999              3.000000

[1000 rows x 4 columns]
```

[7]:
```
# Crear un histograma de las calificaciones
sns.histplot(data=reseña_calificacion, x='calificacion_promedio', bins=10)
plt.title('Distribución de las calificaciones promedio de los libros')
plt.xlabel('Calificación promedio')
plt.ylabel('Número de libros')
plt.show()
```



La distribución de las calificaciones muestra una tendencia hacia las valoraciones positivas, con un pico en el rango de 4 a 4.5 estrellas. Esto sugiere que el contenido de la plataforma es en general

bien recibido por los usuarios.

1.4 Editorial que ha publicado el mayor número de libros con más de 50 páginas

[8]:
```
# Consulta para encontrar la editorial con más libros de más de 50 páginas
query = """SELECT
    p.publisher,
    COUNT(*) AS num_libros_mas_50_paginas
FROM
    books b
INNER JOIN publishers p ON b.publisher_id = p.publisher_id
WHERE
    b.num_pages > 50
GROUP BY
    p.publisher
ORDER BY
    num_libros_mas_50_paginas DESC
LIMIT 1;"""

# Ejecutar la consulta y almacenar el resultado
editorial_mas_50_paginas = pd.read_sql(query, engine)

# Mostrar el resultado
print(editorial_mas_50_paginas)
```

```
       publisher  num_libros_mas_50_paginas
0  Penguin Books                         42
```

La editorial Penguin Books ha publicado el mayor numero de libros con mas de 50 paginas con un total de 42 libros.

1.5 Autor que tiene la más alta calificación

[9]:
```
# Consulta para encontrar al autor con la calificación promedio más alta
query = """
SELECT
    a.author,
    AVG(r.rating) AS promedio_calificacion
FROM
    books b
INNER JOIN authors a ON b.author_id = a.author_id
INNER JOIN ratings r ON b.book_id = r.book_id
GROUP BY
    a.author
HAVING
    COUNT(r.rating) >= 50
ORDER BY
    promedio_calificacion DESC
```

```
LIMIT 1;"""

# Ejecutar la consulta y almacenar el resultado
resultado = pd.read_sql(query, engine)

# Mostrar el resultado
print(resultado)
```

```
           author  promedio_calificacion
0  Diana Gabaldon                    4.3
```

La autora con las mas altas calificaciones es Diana Gabaldon en libros con las de 50 paginas. Tiene un promedio de calificacion de 4.3/5

1.6 Número promedio de reseñas de texto entre los usuarios que calificaron más de 50 libros

[10]:
```
# Consulta
query = """
SELECT
    AVG(num_text_reviews) AS promedio_reseñas_texto
FROM (
    SELECT
        r.username,
        COUNT(CASE WHEN rv.text IS NOT NULL THEN 1 END) AS num_text_reviews
    FROM
        ratings r
    LEFT JOIN reviews rv ON r.book_id = rv.book_id AND r.username = rv.username
    GROUP BY
        r.username
    HAVING
        COUNT(*) > 50
) AS subconsulta;"""

# Ejecutar la consulta y almacenar el resultado
promedio_reseñas_texto = pd.read_sql(query, engine)

# Mostrar el resultado
print(promedio_reseñas_texto)
```

```
   promedio_reseñas_texto
0               24.333333
```

El numero promedio de reseñas de texto son 24 reseñas por libro entre los usuraios que calificaron mas de 50 libros.

1.7 Conclusion general

Recomendaciones para un Nuevo Producto:

El presente estudio ha revelado valiosos insights sobre los hábitos de lectura y las preferencias de los usuarios de una plataforma de libros digitales en el contexto post-pandémico. Los resultados obtenidos permiten identificar oportunidades y desafíos para el desarrollo de un nuevo producto en este mercado.

Hallazgos Clave:

*Contenido Actualizado: La preferencia por libros publicados después del año 2000 indica una demanda de contenido fresco y relevante.

*Calidad Percibida: La alta valoración promedio de los libros sugiere que la plataforma ofrece un catálogo de calidad.

*Liderazgo Editorial: Penguin Books destaca como una editorial líder en la publicación de obras de mayor extensión, lo que posiciona a esta editorial como un socio estratégico.

*Autores Influyentes: Autores como Diana Gabaldon ejercen una gran influencia en las decisiones de compra de los usuarios.

*Comunidades Activas: Los usuarios que han calificado más de 50 libros son una comunidad activa y comprometida, generando un gran volumen de reseñas de texto.

Se podria realizar una personalización Avanzada:

Recomendaciones basadas en autores y editoriales: Sugerir libros de autores y editoriales con alta valoración. Recomendaciones basadas en reseñas: Utilizar el análisis de sentimientos para recomendar libros similares a los que el usuario ha disfrutado. Listas de lectura personalizadas: Crear listas de lectura personalizadas basadas en los intereses y el historial de lectura del usuario.