FACULTAD DE CIENCIAS
LICENCIATURA CIENCIA DE DATOS

“PROYECTO FINAL”

Reporte de Análisis de Datos: Series de Televisión IMDb

JULIETTE NINNIE LAZO VAZQUEZ

MATRÍCULA: 371701

MATERIA: PROGRAMACIÓN PARA CIENCIA DE DATOS.

VIERNES 23 DE MAYO DEL 2025.

INTRODUCCIÓN

Este proyecto se basa en un dataset que contiene información de diversas series de televisión extraída de IMDb. A través del análisis de variables como la calificación promedio, la cantidad de votos, la duración de los episodios y los géneros predominantes, se busca identificar patrones, tendencias y preferencias del público. Además, se visualizan estos datos para facilitar su interpretación y extraer conclusiones significativas sobre lo que hace exitoso a un programa de televisión.

El análisis no solo ofrece una perspectiva general sobre el comportamiento del contenido televisivo en los últimos años, sino que también sirve como base para explorar cómo ciertos elementos —como el género o los actores protagonistas— influyen en la recepción y popularidad de una serie.

DESARROLLO

1. Carga de datos

Se utilizó la librería `pandas` para cargar el dataset en formato CSV:

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
```

```
df = pd.read_csv("imdb_tvshows.csv.xls")
```

El archivo contiene información sobre series de televisión, incluyendo su título, duración, género, calificación, votos y años de emisión.

2. Limpieza y transformación de los datos

Se aplicó la función cleaning(df) que realizó las siguientes tareas:

- Estandarización de nombres de columnas a minúsculas y sin espacios.
- Conversión de columnas numéricas como `rating` y `episodeduration(in minutes)` a tipo float.
- Eliminación de filas con valores nulos en columnas clave (`title`, `rating`, `genres`, `duration`).

- Extracción del género principal (primer elemento de la lista de géneros).
- Extracción del año de inicio desde la columna `years`.

Código:

```python
df = cleaning(df)  # Limpieza
```

```python
def cleaning(df):
    """
    Realiza una limpieza básica del DataFrame:
    - Elimina filas con valores nulos en columnas importantes.
    - Convierte columnas a tipos de datos adecuados.
    """
    df = df.copy()  # copia del df para no modificar el original

    # normalizamos nombres de columnas para evitar errores de preprocesamiento
    df.columns = df.columns.str.strip().str.lower()
    # mantenimiento para valores nan
    df.replace("NaN", pd.NA, inplace=True)

    if "title" not in df.columns:
        posibles_titles = [col for col in df.columns if "title" in col]
        if posibles_titles:
            df.rename(columns={posibles_titles[0]: "title"}, inplace=True)
```

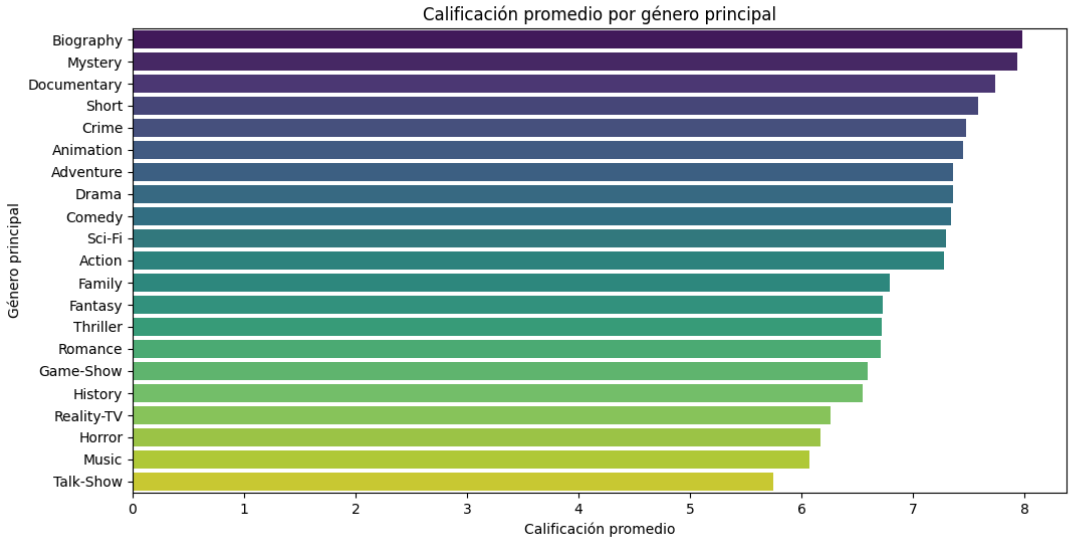
```python
# columnas relevantes para mi analisis
columnas_necesarias = ["title", "episodeduration(in minutes)", "genres", "rating"]

# depuración: imprime las columnas antes de eliminar nulos
print("Current Columns:", df.columns.tolist())
# eliminacion de nulos
df.dropna(subset=columnas_necesarias, inplace=True)
# columna de duración a numérico, forzando errores a NaN para limpieza posterior
df["episodeduration(in minutes)"] = pd.to_numeric(df["episodeduration(in minutes)"], errors="coerce")
# columna de rating a numérico, similar al paso anterior
df["rating"] = pd.to_numeric(df["rating"], errors="coerce")
# elimina filas que tengan valores nulos en cualquier columna, asegurando integridad de datos
df.dropna(inplace=True)
return df
```

3. Análisis y visualización de datos

3.1. Calificación promedio por género principal

Se agruparon los datos por el género principal y se calculó el promedio de calificación.

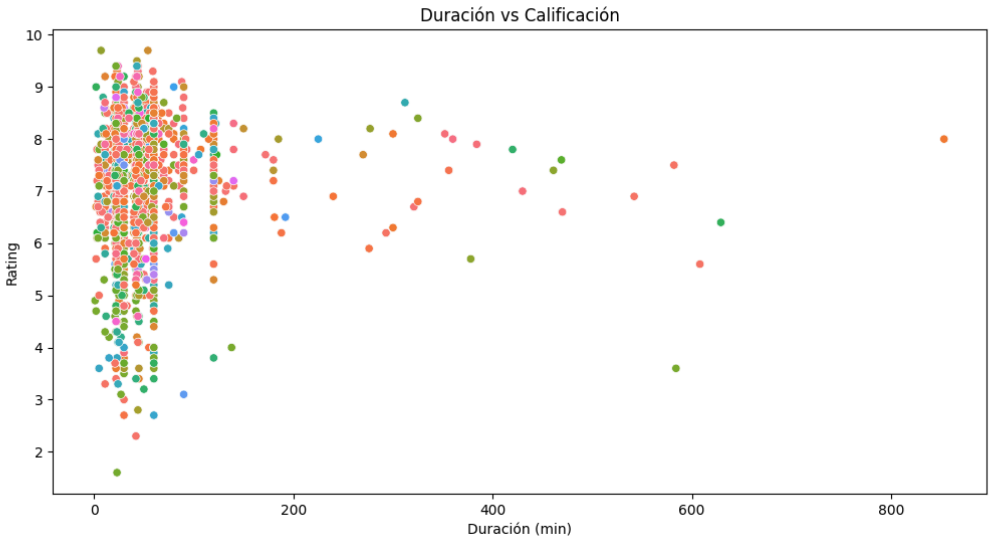
```python
def rating_vs_genre(df):
    """
    Analiza la calificación promedio según el género principal.
    """
    # género principal de la columna 'genres' (primer género listado)
    df["generoprincipal"] = df["genres"].apply(lambda x: x.split(",")[0] if isinstance(x, str) else x)
    #  género principal y la calificación promedio (de mauyor a menor)
    promedio_genero = df.groupby("generoprincipal")["rating"].mean().sort_values(ascending=False)
    # plt horizontal para la calificación promedio por género
    sns.barplot(x=promedio_genero.values, y=promedio_genero.index, palette="viridis")
    plt.title("Calificación promedio por género principal")
    plt.xlabel("Calificación promedio")
    plt.ylabel("Género principal")
    plt.show()
```



3.2. Relación entre duración y calificación

Se creó un gráfico de dispersión para visualizar si existe relación entre la duración de los episodios y su calificación.

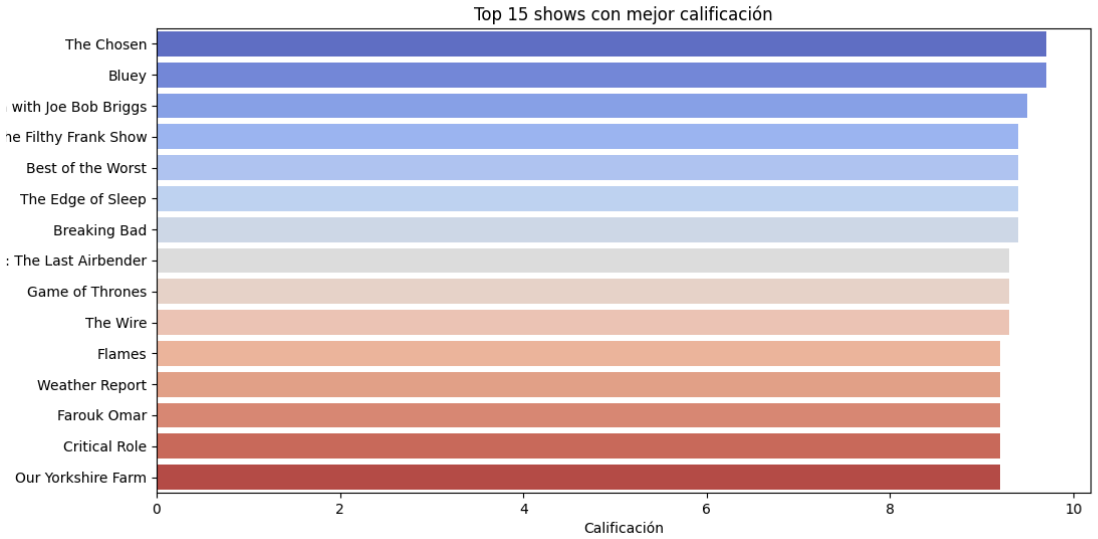
```python
def rating_vs_duration(df):
    """
    Grafica la relación entre duración del episodio y calificación.
    """
    # scatterplot para observar la relación entre duración y rating, coloreando por género
    sns.scatterplot(data=df, x="episodeduration(in minutes)", y="rating", hue="genres", legend=False)
    plt.title("Duración vs Calificación")
    plt.xlabel("Duración (min)")
    plt.ylabel("Rating")
    plt.show()
```



3.3. Top 15 series con mayor calificación

Se seleccionaron las 15 series mejor calificadas y se graficaron con barras horizontales.

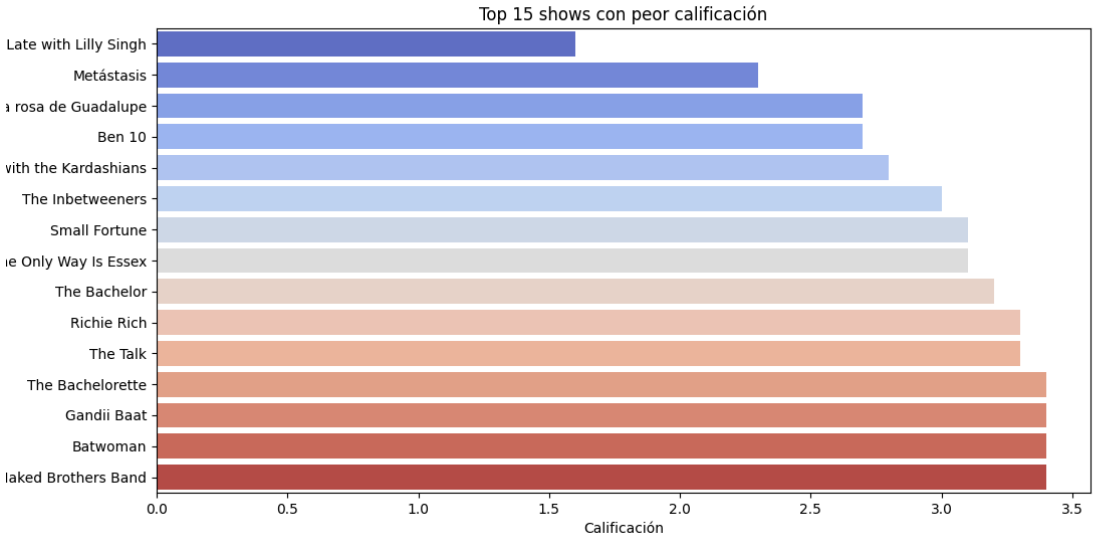
```python
def rating_vs_show(df):
    """
    Muestra los shows mejor calificados.
    """
    # 15 shows con mayor rating para análisis
    df = df.sort_values("rating", ascending=False).head(15)
    # plt horizontal mostrando los ratings de los top 15 shows
    sns.barplot(x="rating", y="title", data=df, palette="coolwarm")
    plt.title("Top 15 shows con mejor calificación")
    plt.xlabel("Calificación")
    plt.ylabel("Show")
    plt.show()
```



3.4. Top 15 series con menor calificación

Se seleccionaron las 15 series menor calificadas y se graficaron con barras horizontales.

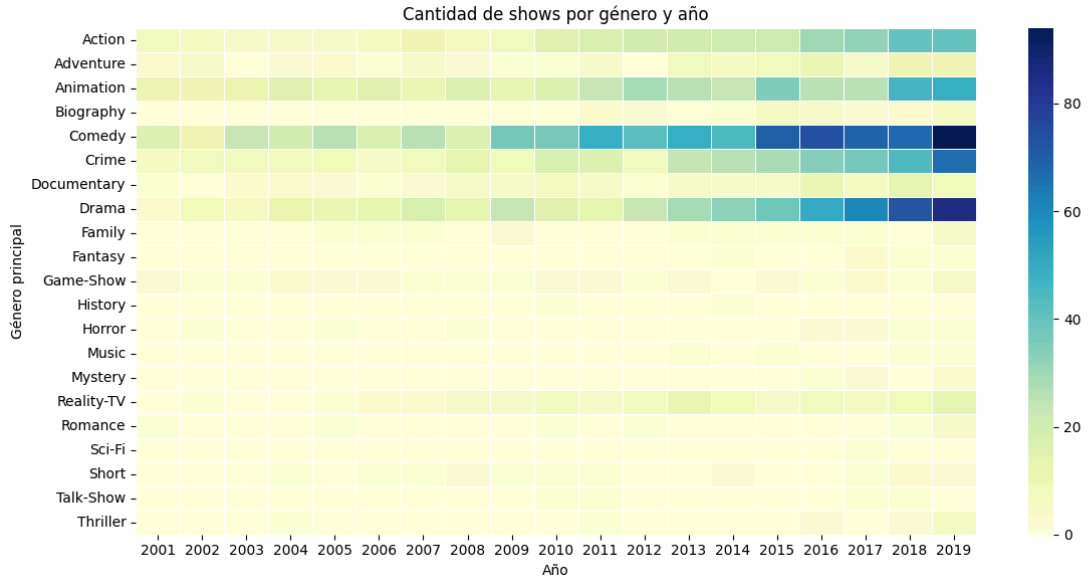
```python
def rating_vs_show_2(df):
    """
    Muestra los shows con peor calificación.
    """
    # 15 shows con menor rating para análisis
    df = df.sort_values("rating", ascending=True).head(15)
    # plt horizontal mostrando los ratings de los top 15 shows
    sns.barplot(x="rating", y="title", data=df, palette="coolwarm")
    plt.title("Top 15 shows con peor calificación")
    plt.xlabel("Calificación")
    plt.ylabel("Show")
    plt.show()
```



3.5. Evolución por año y género

Se graficó un heatmap mostrando cuántos shows se estrenaron por género y año. También se graficó una línea de tendencia con el promedio de calificación por año.

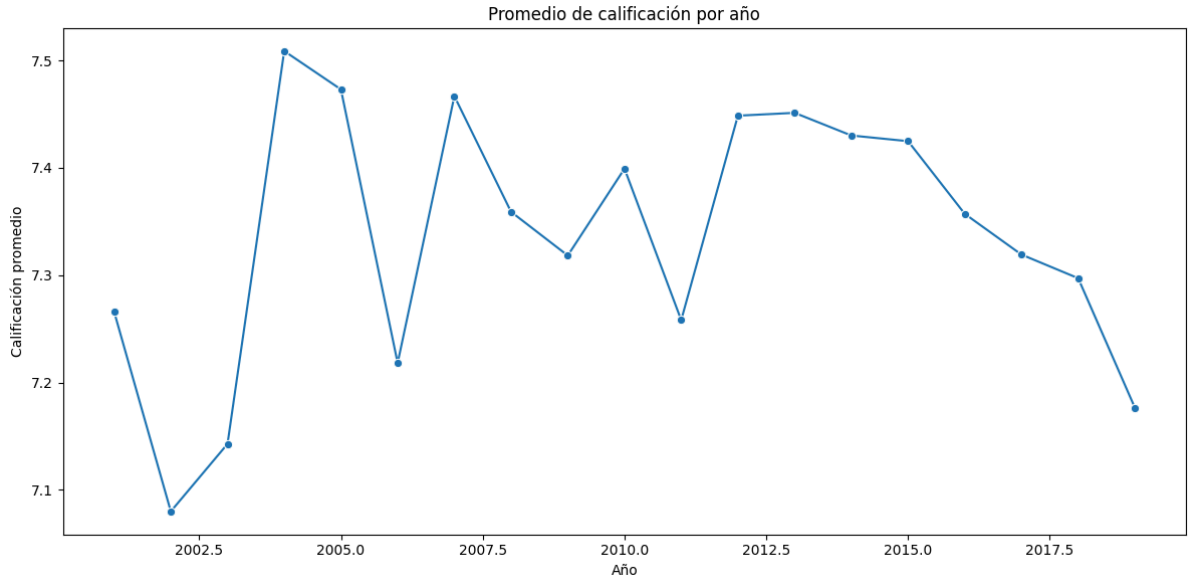
```
# heatmap para visualizar la cantidad de shows por género y año
sns.heatmap(conteo.T, cmap="YlGnBu", linewidths=0.5)
plt.title("Cantidad de shows por género y año")
plt.xlabel("Año")
plt.ylabel("Género principal")
plt.tight_layout()
plt.show()
```



3.6. Gráficos estadísticos generales

Se generaron los siguientes gráficos:
- Histograma de la distribución de calificaciones (rating).
- Histograma de la duración de episodios.
- Conteo de frecuencia por género principal.

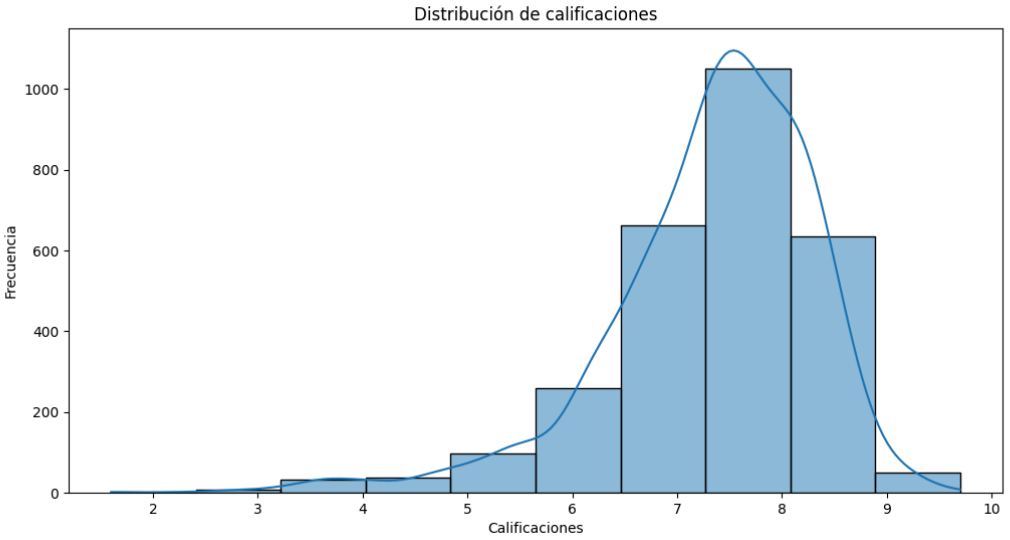
```
# calificación promedio por año para detectar tendencias temporales
rating_por_ano = df.groupby('year_inicio')['rating'].mean()

# plt de líneas para mostrar evolución del rating promedio a lo largo de los años
sns.lineplot(x=rating_por_ano.index, y=rating_por_ano.values, marker="o")
plt.title("Promedio de calificación por año")
plt.xlabel("Año")
plt.ylabel("Calificación promedio")
plt.tight_layout()
plt.show()
```



3.7. Histograma de la distribución de calificaciones (rating)

El histograma muestra cómo se distribuyen las calificaciones de los shows. La mayoría de las calificaciones se encuentra entre 7 y 8, lo que indica que la tendencia general es hacia evaluaciones medias-altas, con pocas calificaciones extremas.

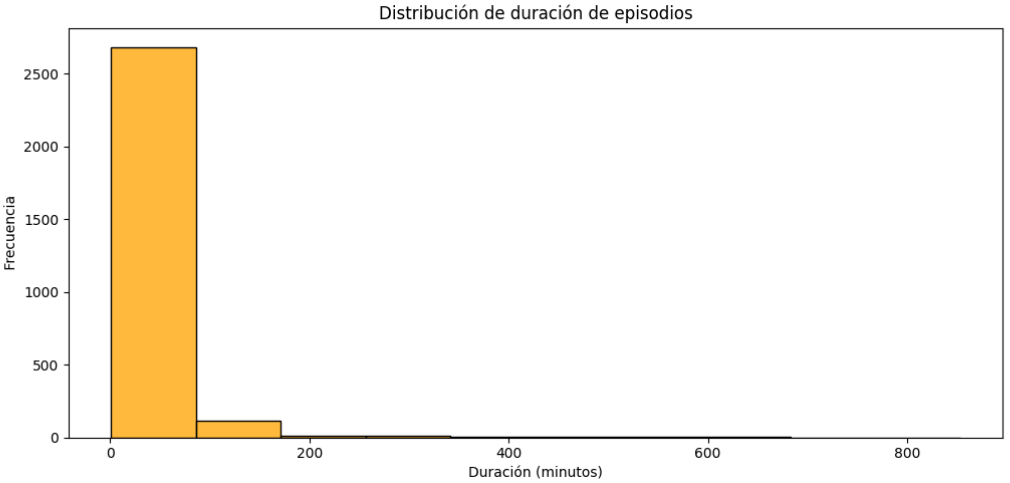
```
# histograma con KDE para visualizar cómo se distribuyen las calificaciones
sns.histplot(df["rating"], bins=10, kde=True)
plt.title("Distribución de calificaciones")
plt.xlabel("Calificaciones")
plt.ylabel("Frecuencia")
plt.show()
```



3.8. Histograma de la duración de episodios

El histograma representa la frecuencia de las diferentes duraciones de los episodios. Se observa que la mayor parte de los episodios dura entre 20 y 60 minutos, mientras que existen pocos episodios con duraciones mucho mayores, mostrando una distribución asimétrica hacia la derecha.

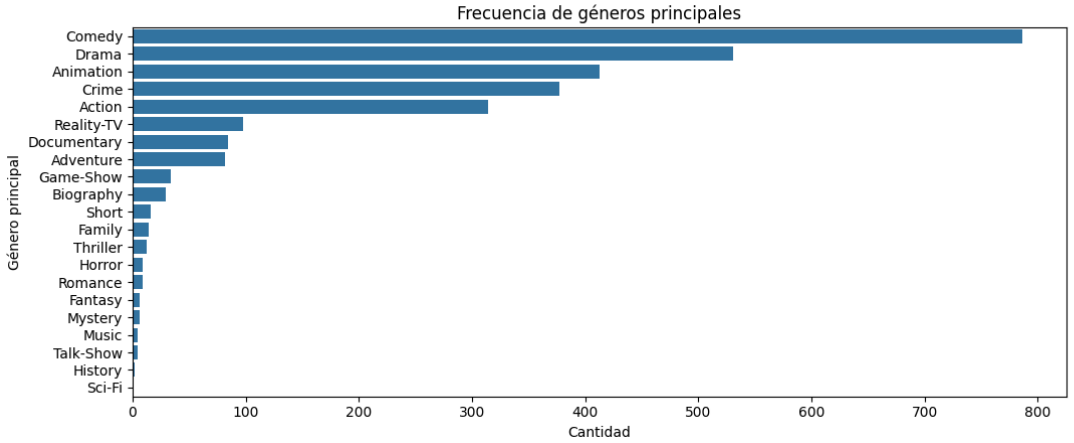
```
# histograma para observar la distribución de la duración de episodios
sns.histplot(df["episodeduration(in minutes)"], bins=10, color='orange')
plt.title("Distribución de duración de episodios")
plt.xlabel("Duración (minutos)")
plt.ylabel(" Frecuencia ")
plt.show()
```



3.9. Conteo de frecuencia por género principal

Esta gráfica ilustra cuántos shows corresponden a cada género principal. Comedia es el género más frecuente, seguido por drama y animación, lo que revela una clara preferencia por estos géneros en la base de datos analizada, mientras que otros géneros son considerablemente menos comunes.

```
# género principal para conteo
df["generoprincipal"] = df["genres"].apply(lambda x: x.split(",")[0])
# conteo y gráfico de barras horizontal con frecuencia de cada género principal
sns.countplot(y="generoprincipal", data=df, order=df["generoprincipal"].value_counts().index)
plt.title("Frecuencia de géneros principales")
plt.xlabel("Cantidad")
plt.ylabel("Género principal")
plt.show()
```



4. Conclusiones

El análisis estadístico realizado proporciona una visión general del comportamiento de los shows en la base de datos. Se observa que los géneros más comunes son la comedia, el drama y la animación, lo que indica una preferencia clara por estos tipos de contenido en las producciones analizadas. La distribución de las calificaciones muestra que la mayoría de los shows tiene valoraciones medias-altas (entre 7 y 8), aunque hay pocos casos de puntuaciones muy bajas o muy altas.

En cuanto a la duración, la mayoría de los episodios suele durar entre 20 y 60 minutos, con pocos casos de episodios extraordinariamente largos, lo que sugiere que los formatos estándar son preferidos por creadores y audiencia. Además, las calificaciones más altas no siempre corresponden a episodios con mayor duración,

lo que implica que la calidad percibida por el público no depende necesariamente del tiempo de cada capítulo.

El análisis temporal revela que años como 2005 y 2011 mostraron incrementos en la calidad promedio, reflejando posiblemente una mayor inversión o innovación en la producción durante esos periodos. Finalmente, se concluye que tanto el género principal como el año de producción pueden influir en el éxito de una serie, ya que la popularidad y la valoración del público varían siguiendo estas características.

5. Librerías utilizadas

- pandas – Manipulación de datos.
- seaborn – Visualización gráfica.
- matplotlib.pyplot – Personalización y presentación de gráficas.