



**DATA ANALYTICS – FLEX**

**COMISION #51455**

**PROYECTO FINAL**

FECHA DE PRESENTACIÓN: 23/06/2023

FURTADO, JULIO CESAR

# Rotten Tomatoes

Reseña y puntaje  
sus variables



## TABLA DE CONTENIDO

<b>I.</b>	<b>Historial de versiones</b>	<b>04</b>
<b>II.</b>	<b>Introducción</b>	<b>05</b>
II.I	Descripción de la temática de los datos	05
II.II	Objetivo	05
II.III	Alcance	06
II.IV	Herramientas a implementar	06
<b>III.</b>	<b>Base de datos</b>	<b>06</b>
III.I	Dataset	06
III.II	Diagramas de entidad-relación	07
III.II.I	Nivel conceptual	07
III.II.II	Nivel lógico	08
III.III	Listado de tablas	09
<b>IV.</b>	<b>Maqueta de dashboard inicial</b>	<b>13</b>
<b>V.</b>	<b>Documentación en Power BI</b>	<b>18</b>
V.I	Estructura inicial de datos	18
V.II	Modelo de datos relacional	18
V.III	Transformaciones aplicadas	19
V.IV	Tablas de medidas y medidas calculadas	30
<b>VI.</b>	<b>Conclusión y futuras líneas</b>	<b>32</b>
<b>VII.</b>	<b>Bibliografía</b>	<b>34</b>

## I. HISTORIAL DE VERSIONES

Número de versión	Fecha	Acciones
Versión 1.0	25/04/2023	<ul style="list-style-type: none"> <li>*Definición del proyecto</li> <li>*Selección del dataset</li> <li>*Armado general</li> </ul>
Version 2.0	08/05/2023	<ul style="list-style-type: none"> <li>*Definición de alcance y objetivos</li> <li>*Definición de usuario y aplicación</li> <li>*DERs y estructuras de tablas</li> <li>*Mínimo formateo de documentación</li> </ul>
Version 3.0	05/06/2023	<ul style="list-style-type: none"> <li>*Desarrollo de Maquetado inicial sin estilos del dashboard</li> <li>*Documentación de Power BI inicial</li> <li>*Documentación de Modelo de Datos Relacional</li> <li>*Historial de transformaciones aplicadas para la creación de cada tabla</li> <li>* Medidas calculadas generadas con fórmulas (Tabla de Calendario y Medidas, junto con Parámetro)</li> </ul>
Version 3.1	06/06/2023	<ul style="list-style-type: none"> <li>*Mínimo formateo de proyecto en Power BI</li> <li>*Reemplazo de vistas previas de boceto en el documento</li> </ul>
Version 4.0	14/06/2023	<ul style="list-style-type: none"> <li>*Desarrollo final en Power BI</li> <li>*Modificación de gráficos DERs</li> <li>*Modificación de tablas descriptas</li> <li>*Cambio de mockups de tablero</li> <li>*Cambio de formato de portada</li> </ul>
Version 4.1	15/06/2023	<ul style="list-style-type: none"> <li>*Modificación final de índice</li> <li>*Formato general</li> <li>*Conclusion y líneas finales</li> </ul>
Version 4.2	17/06/2023	<ul style="list-style-type: none"> <li>*Guardado en portada para inicio</li> <li>*Corrección ligera de portada</li> <li>*Tooltips funcionales</li> <li>*Filtro agregado en "Rating"</li> <li>*Tablas agregadas en "Dirección/Guión" para mejorar el uso</li> <li>*Agregadas medianas para ayudar a la conclusión</li> <li>*Documentación actualizada</li> </ul>

## **II. INTRODUCCIÓN**

### **II.I DESCRIPCIÓN TEMÁTICA DE LOS DATOS**

Rotten Tomatoes es uno de los medios líderes para consultar puntuaciones para quienes disfrutan de consumir cine o shows de TV. El sitio toma como base información acerca de las críticas que hayan emitido los profesionales del rubro en diversos medios, generando a partir de ellas promedios y rankings brindando al usuario una interfaz sencilla y resumida, categorizada de diversas maneras, para ser utilizada, por ejemplo, como recomendación.

Cuenta también con las reseñas de usuarios, que a la vez que los críticos especializados, generan un puntaje de audiencia, otro indicador útil como parámetro para decidir.

### **II.II OBJETIVO**

En este trabajo se intentará descubrir si existe un patrón entre la opinión de la crítica especializada con el puntaje de audiencia, así como también si hay otros factores implicados (como por ejemplo la combinación de género y época de lanzamiento, productoras, directores consagrados, etc.), y lo recaudado por cada película.

Por cuestiones de disponibilidad de los datos y naturaleza de la hipótesis, el análisis estará enfocado únicamente en las películas, dejando de lado las series de televisión.

En síntesis, se observará la información de películas valoradas en el sitio de Rotten Tomatoes, con puntaje de audiencia y de crítica, así como género, fechas de lanzamiento en cines, fecha de lanzamiento en streaming, recaudación de taquilla, productoras y más.

## **II.III ALCANCE**

El análisis a desarrollar podrá utilizarse a nivel estratégico por productoras/distribuidoras, para encontrar nichos posibles en función de las opiniones y las tendencias mostradas. También podría ser útil para cinéfilos buscando implementar una ruta de nivel táctico que les permita extender su conocimiento del tema, o clarificar la información que brinda el sitio tomando de base los criterios mencionados en el punto anterior.

## **II.IV HERRAMIENTAS A IMPLEMENTAR**

- \* Microsoft Excel: Para la primer lectura de los datos
- \* Draw.IO: Para la creación de los DERs
- \* Microsoft Power BI: Para el procesamiento de los datos, y la creación del dashboard

## **III. BASE DE DATOS**

### **III.I DATASET**

Para este trabajo se tomó información desde el sitio de Kaggle, donde un usuario puso a disposición los datos extraídos de Rotten Tomatoes con última fecha de actualización el 12 de abril de 2023.

Los datos están separados en dos archivos, donde el principal contiene información completa de las películas a analizar, y el secundario posee información detallada de las críticas recibidas por cada película.

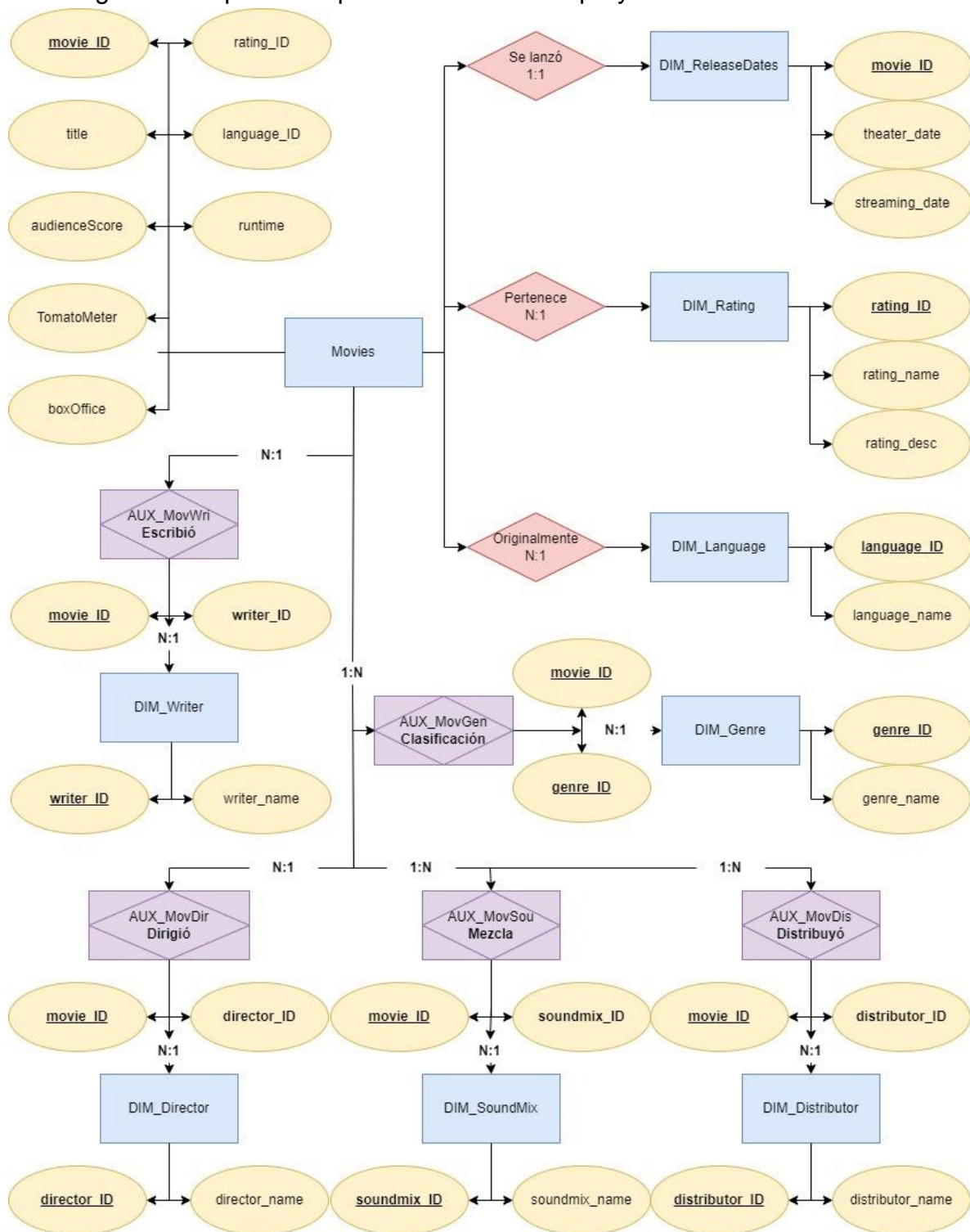
Se adjunta en carpeta el dataset seleccionado para el proyecto.

*Nota: No se utilizará la información relacionada a las reseñas extendidas por probarse impráctico y no sumar a la hipótesis actual.*

## III.II DIAGRAMA DE ENTIDAD-RELACIÓN

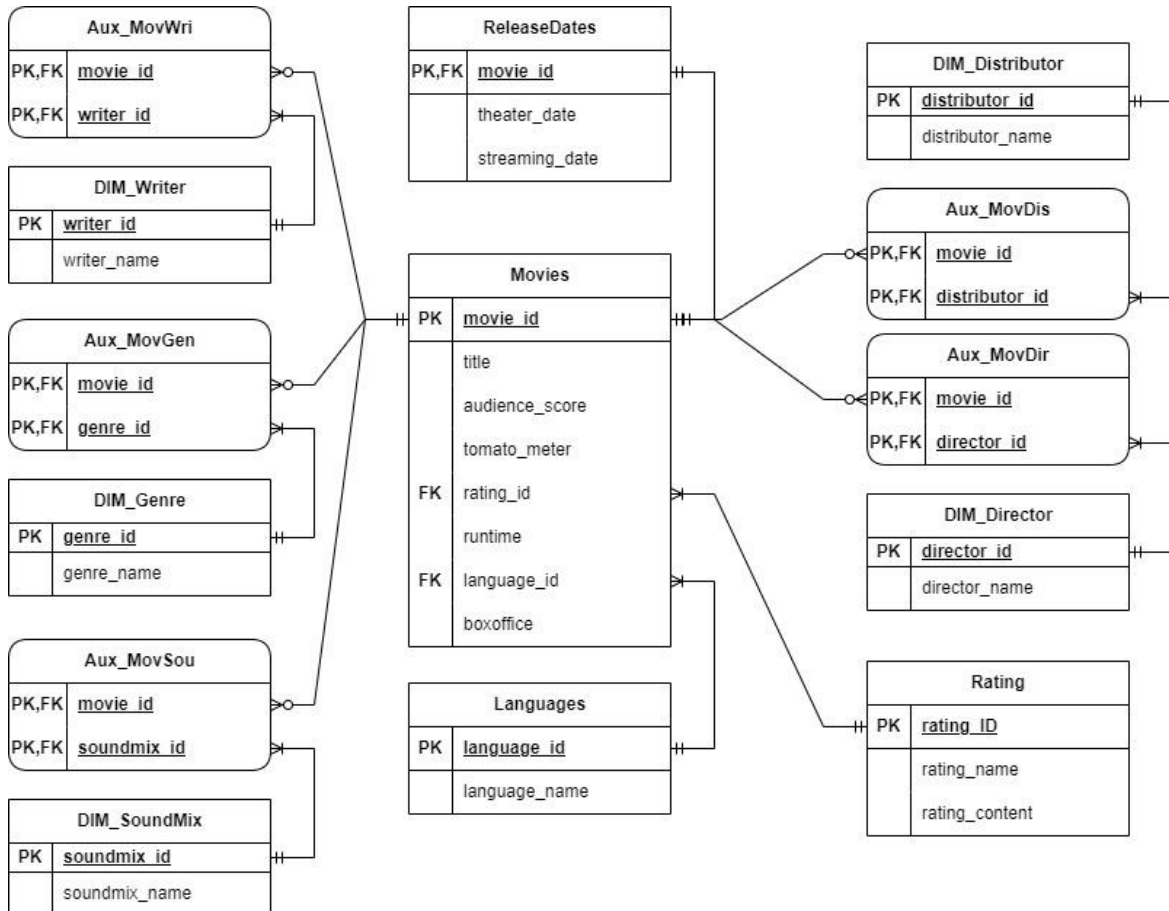
### III.II.I NIVEL CONCEPTUAL

En el siguiente esquema se presenta el DER del proyecto.



### III.II.II NIVEL LÓGICO

A continuación se adjunta el nivel lógico del gráfico, para poder apreciar cada tabla con sus PK y FKs correspondientes.



*Nota: El modelo está armado en función de los datos disponibles, por lo que la cardinalidad está ajustada a esos parámetros, dejando de lado la conceptualidad general.*



### III.III LISTADO DE TABLAS

A continuación se presenta el listado de tablas que componen el DER con su respectiva descripción, claves primarias y foráneas, tipo de dato y relación entre tablas.

TABLA1: MOVIES		
Tipo de Clave	Campo	Tipo de Campo
PK	movie_ID	nvarchar(100)
-	title	nvarchar(100)
-	audienceScore	int
-	tomatoMeter	int
FK	rating_ID	int
-	runtime	int
FK	language_ID	int
-	boxOffice	int

La tabla de Movies contiene la información de las películas con su id único, título, puntaje de audiencia, puntaje de la crítica, id de rating que la vincula con la tabla Rating, su duración, id de lenguaje que la vincula con la tabla Language, y la recaudación.

TABLA2: DIM_RATING		
Tipo de Clave	Campo	Tipo de Campo
PK	rating_ID	int
-	rating_name	nvarchar(100)
-	rating_content	nvarchar(100)

La tabla de Rating refiere a 'edad de admisión recomendada' de las películas e incluye su id único que la vincula con la tabla Movies, el nombre del rating, y el contenido específico por el que le valió tal clasificación.

TABLA3: DIM_LANGUAGES		
Tipo de Clave	Campo	Tipo de Campo
PK	language_ID	int
-	language_name	nvarchar(100)

Languages almacena el idioma original de lanzamiento de cada película. Trae el id de cada lenguaje como clave primaria, y el nombre de cada lenguaje.

TABLA4: AUX_MOVGEN		
Tipo de Clave	Campo	Tipo de Campo
PK,FK	movie_ID	nvarchar(100)
PK,FK	genre_ID	int

*Esta tabla, así como todas cuyo nombre comienza en 'Aux' serán desarrolladas para evitar una relación muchos a muchos.* En este caso, es una tabla auxiliar para vincular películas a géneros (de ahí el nombre Aux\_MovGen, y como tal, ofrece como clave primaria (y foránea) los campos ID las mencionadas tablas, para evitar un índice o concatenar un campo que luego no se usará.

TABLA5: DIM_GENRE		
Tipo de Clave	Campo	Tipo de Campo
PK	genre_ID	int
-	genre_name	nvarchar(100)

La tabla Genre describe cada género con un id único como clave primaria y el nombre del género.

TABLA6: AUX_MOVDIR		
Tipo de Clave	Campo	Tipo de Campo
PK,FK	movie_ID	nvarchar(100)
PK,FK	director_ID	int

*Esta tabla, así como todas cuyo nombre comienza en 'Aux' serán desarrolladas para evitar una relación muchos a muchos.* En este caso, es una tabla auxiliar para vincular películas a directores (de ahí el nombre Aux\_MovDir, y como tal, ofrece como clave primaria (y foránea) los campos ID las mencionadas tablas, para evitar un índice o concatenar un campo que luego no se usará.

TABLA7: DIM_DIRECTOR		
Tipo de Clave	Campo	Tipo de Campo
PK	director_ID	int
-	director_name	nvarchar(100)

La tabla Director incluye a cada director con un id único como clave primaria y el nombre del director.

TABLA8: AUX_MOVDIS		
Tipo de Clave	Campo	Tipo de Campo
PK,FK	movie_ID	nvarchar(100)
PK,FK	director_ID	int

*Esta tabla, así como todas cuyo nombre comienza en 'Aux' serán desarrolladas para evitar una relación muchos a muchos. En este caso, es una tabla auxiliar para vincular películas a distribuidoras/productoras (de ahí el nombre Aux\_MovDis, y como tal, ofrece como clave primaria (y foránea) los campos ID las mencionadas tablas, para evitar un índice o concatenar un campo que luego no se usará.*

TABLA9: DISTRIBUTOR		
Tipo de Clave	Campo	Tipo de Campo
PK	distributor_ID	int
-	distributor_name	nvarchar(100)

La tabla Distributor describe cada distribuidora con un id único como clave primaria y el nombre de la distribuidora.

TABLA10: AUX_MOVWRI		
Tipo de Clave	Campo	Tipo de Campo
PK,FK	movie_ID	nvarchar(100)
PK,FK	writer_ID	int

*Esta tabla, así como todas cuyo nombre comienza en 'Aux' serán desarrolladas para evitar una relación muchos a muchos. En este caso, es una tabla auxiliar para vincular películas a escritores (de ahí el nombre Aux\_MovWri, y como tal, ofrece como clave primaria (y foránea) los campos ID las mencionadas tablas, para evitar un índice o concatenar un campo que luego no se usará.*

TABLA11: WRITER		
Tipo de Clave	Campo	Tipo de Campo
PK	writer_ID	int
-	writer_name	nvarchar(100)

La tabla Writer incluye a cada escritor con un id único como clave primaria y el nombre del escritor.

TABLA12: AUX_MOVSOU		
Tipo de Clave	Campo	Tipo de Campo
PK,FK	movie_ID	nvarchar(100)
PK,FK	soundmix_ID	int

*Esta tabla, así como todas cuyo nombre comienza en 'Aux' serán desarrolladas para evitar una relación muchos a muchos. En este caso, es una tabla auxiliar para vincular películas a mezclas de sonido (de ahí el nombre Aux\_MovSou, y como tal, ofrece como clave primaria (y foránea) los campos ID las mencionadas tablas, para evitar un índice o concatenar un campo que luego no se usará.*

TABLA13: SOUNDMIX		
Tipo de Clave	Campo	Tipo de Campo
PK	soundmix_ID	int
-	soundmix_name	nvarchar(100)

La tabla Soundmix incluye a cada tipo de mezcla de sonico con un id único como clave primaria y el nombre del de la mezcla.

TABLA14: RELEASEDATES		
Tipo de Clave	Campo	Tipo de Campo
PK	movie_ID	nvarchar(100)
-	theater_date	datetime
-	streaming_date	datetime

Esta última tabla, ReleaseDates posee información del lanzamiento cada película (por eso lleva también movie\_id como PK y FK para relacionarla con la tabla principal). Lleva como datos propios la fecha de estreno en cines y en streaming según corresponda a cada film. *Nota: la división entre esta tabla y la principal tiene como único fin utilizarla como tabla de tiempo por ser los únicos dos campos con formato datetime. Puede devolverse en otro punto del desarrollo del trabajo a la tabla principal.*

#### IV. MAQUETA DE DASHBOARD INICIAL

A continuación, se deja un boceto simple de dashboard, sin estilos aplicados, sino únicamente como estructura inicial

Portada: Una imagen de fondo apuntada a vincular el trabajo al cine, con el logo de la compañía. Posee la fecha de ultima actualización (*aunque actualmente funciona desde una función que devuelve la fecha actual*) con una botonera para navegar entre las páginas del tablero



Pagina 1: Glosario. Lleva esta posición en el tablero para mostrar en principio que la paleta podría percibirse como incorrecta, pero en realidad refleja la identidad de

marca de la fuente.



## Glosario



Rotten Tomatoes es uno de los medios líderes para recibir recomendaciones para quienes disfrutan de consumir cine o shows de TV. El sitio toma como base información acerca de las críticas que hayan emitido los profesionales del rubro en diversos medios, generando a partir de ellas promedios y puntuaciones brindando al usuario una interfaz sencilla y resumida, categorizada de diversas maneras, para ser utilizada, por ejemplo, como recomendación. Cuenta también con las reseñas de usuarios, que a la vez que los críticos especializados, generan un puntaje de audiencia, otro indicador útil a la hora de recibir una recomendación.



Es importante aclarar que Rotten Tomatoes utiliza una paleta de colores que resulta indicativa sobre si misma, pero funciona al revés de la señalización habitual en el resto de nuestras vidas

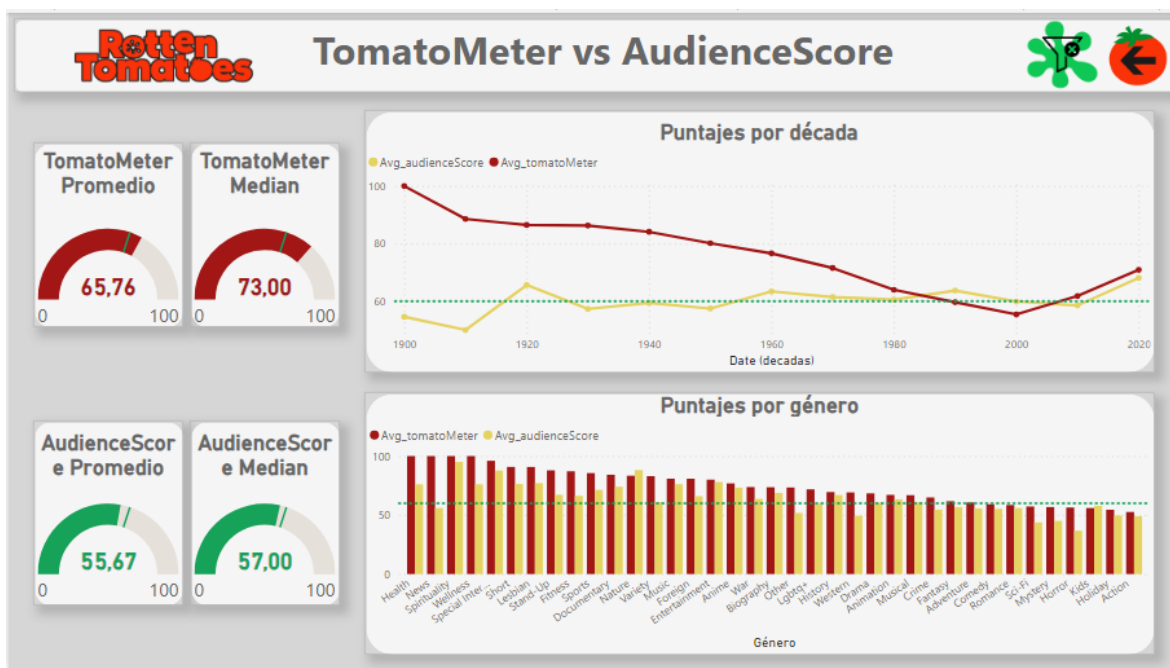




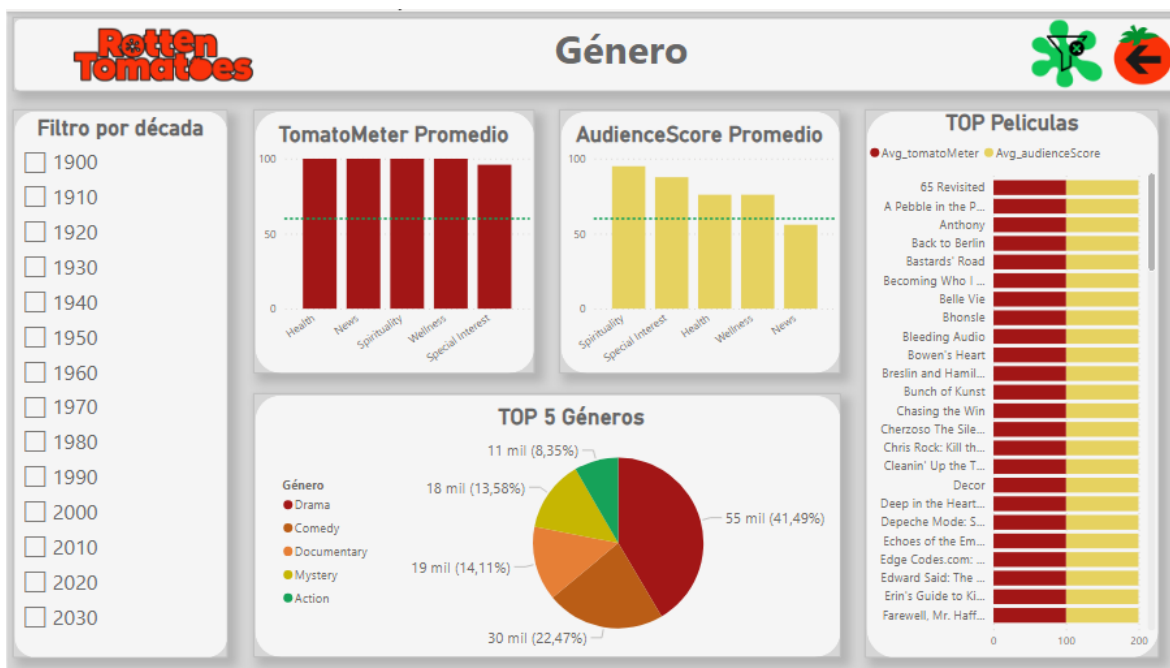
Siendo que la paleta va a respetarse a lo largo del informe, podés pasar el mouse por cada logo y ver qué significa cada color en el caso del TomatoMeter y el AudienceScore



Página 2: Relación entre puntajes. El objeto es apuntar a la hipótesis inicial del trabajo, dando como indicadores principales los promedios de ambos valores (y las medianas, para luego compararlos por década y género, siempre teniendo marcado como marca accesoria el parámetro que el sitio mismo pone para recibir una u otra clasificación (utilizando el 60% como límite duro)



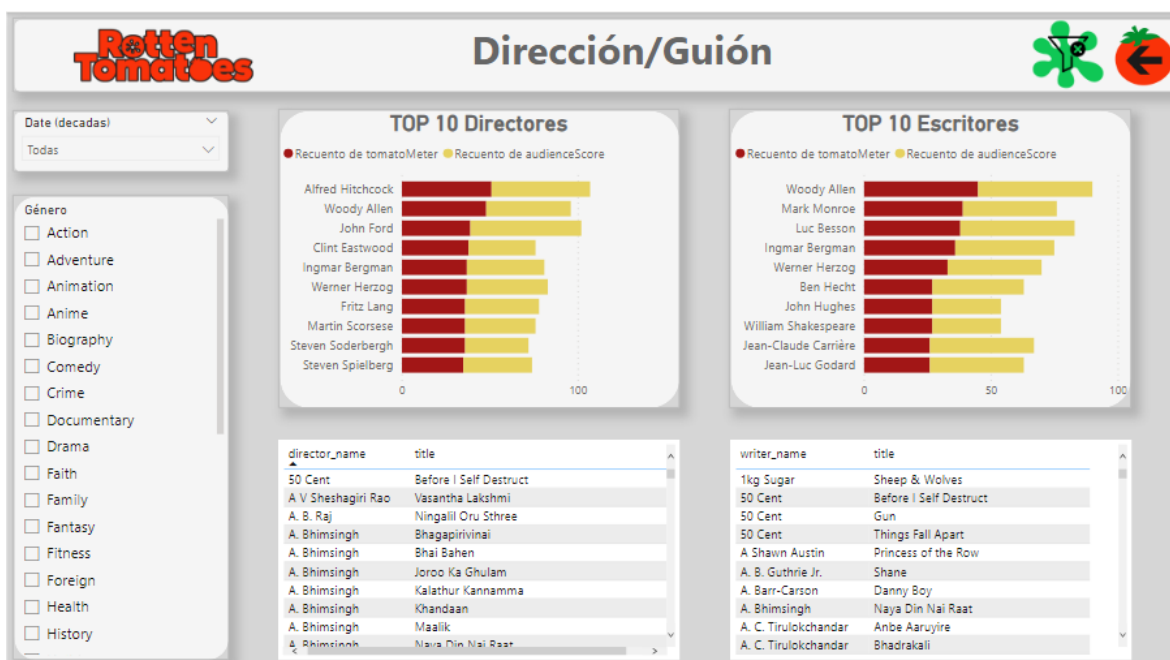
Página 3: Apertura de género. Se presenta para, pudiendo filtrar por décadas, encontrar por cada período los géneros dominantes, los promedios de recepción de crítica y audiencia en cada uno, y en última instancia, un destaque de películas individuales también filtrables bajo los mismos parámetros.



Página 4: Recaudación. Es un resumen que vincula la recaudación según el género, mostrando los 5 géneros que más recaudaron, mientras que debajo se presenta el vínculo entre promedio de recaudación (con línea de mediana) y puntaje de crítica. También trae un slider para modificar el total recaudado por cada género y el promedio.

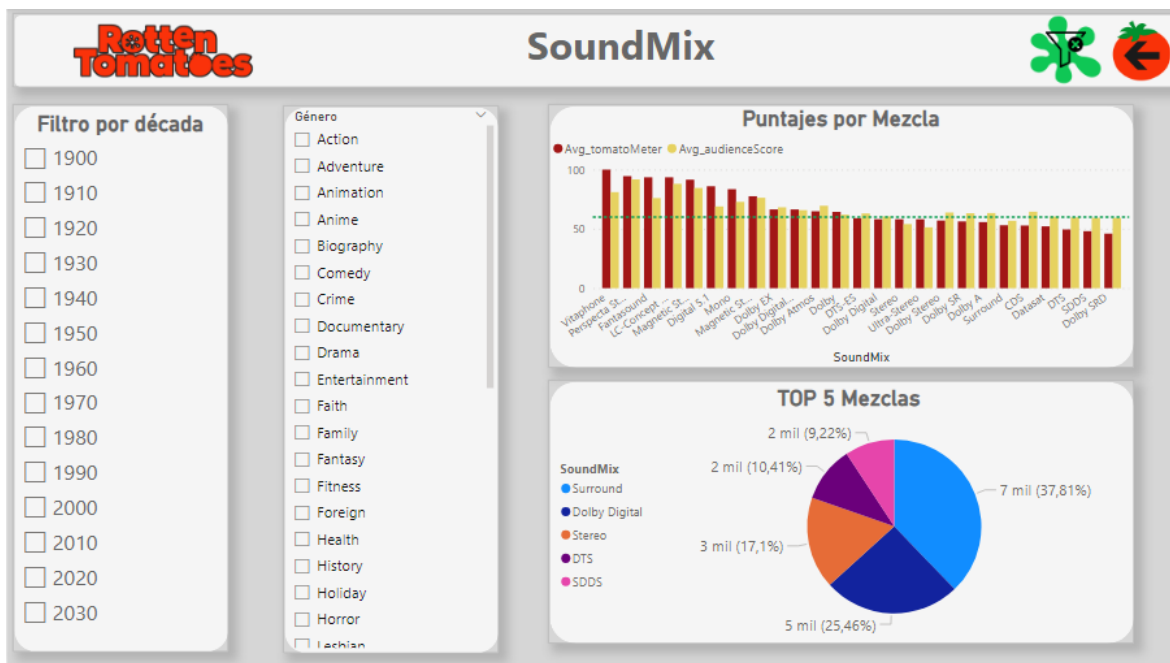


Página 5: Dirección/Guión. Expande el conocimiento de género presentando directores y escritores destacados, pudiendo segregar nuevamente por década. Ofrece también tablas para facilitar recomendaciones a simple vista.

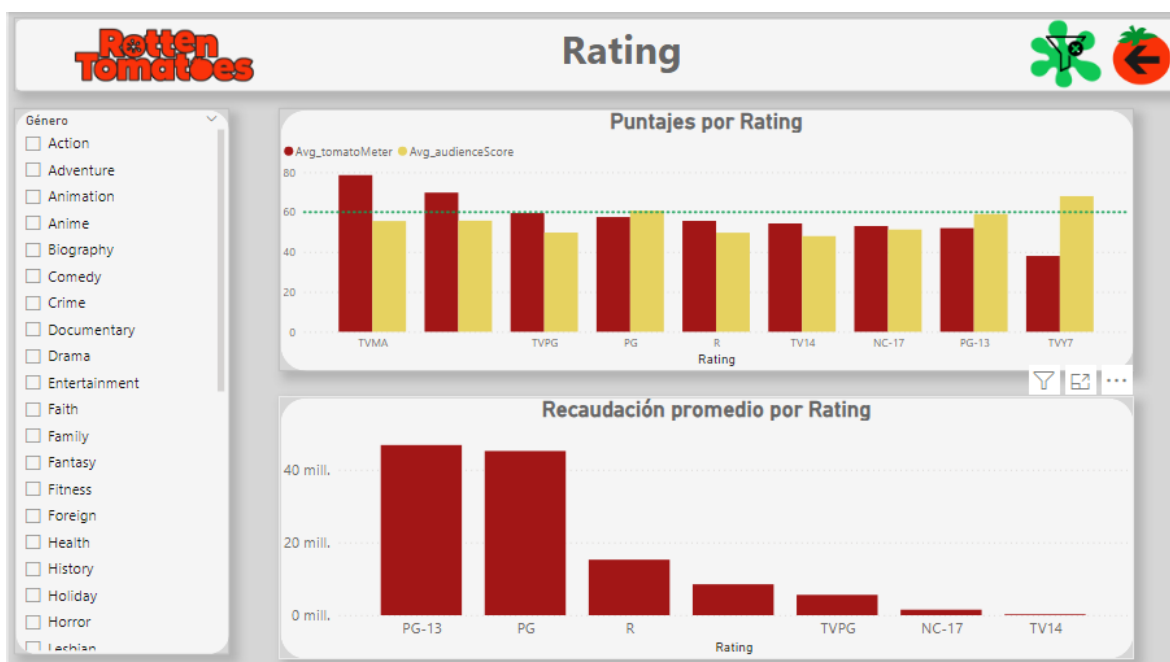


Página 6: SoundMix. Abre el análisis sobre la mezcla de sonido para mostrar la influencia que podría tener en los puntajes de recepción y los divide nuevamente por género y década.





Página 7: Rating. Para enriquecer la hipótesis inicial, se suma una apertura por rating (recordando que es el indicador que muestra la restricción de edad que puede o no tener una película)



## **V. Documentación en Power BI**

### **V.I. Estructura inicial de datos**

Se conectó Power BI a la base de datos que será utilizada para el proyecto. Esto se hizo desde Power BI Desktop usando “web” como conector, con la URL. Este es el link del archivo en Google Drive:

<https://docs.google.com/spreadsheets/d/1IbiSMg10mEFUpR2rKhcZJ4R483zXPS2HLLdLynyJg8Q/edit?usp=sharing>

Y el siguiente el enlace con el que se hizo la conexión a Power BI a través de la opción de “Publicar en la Web” de Google:

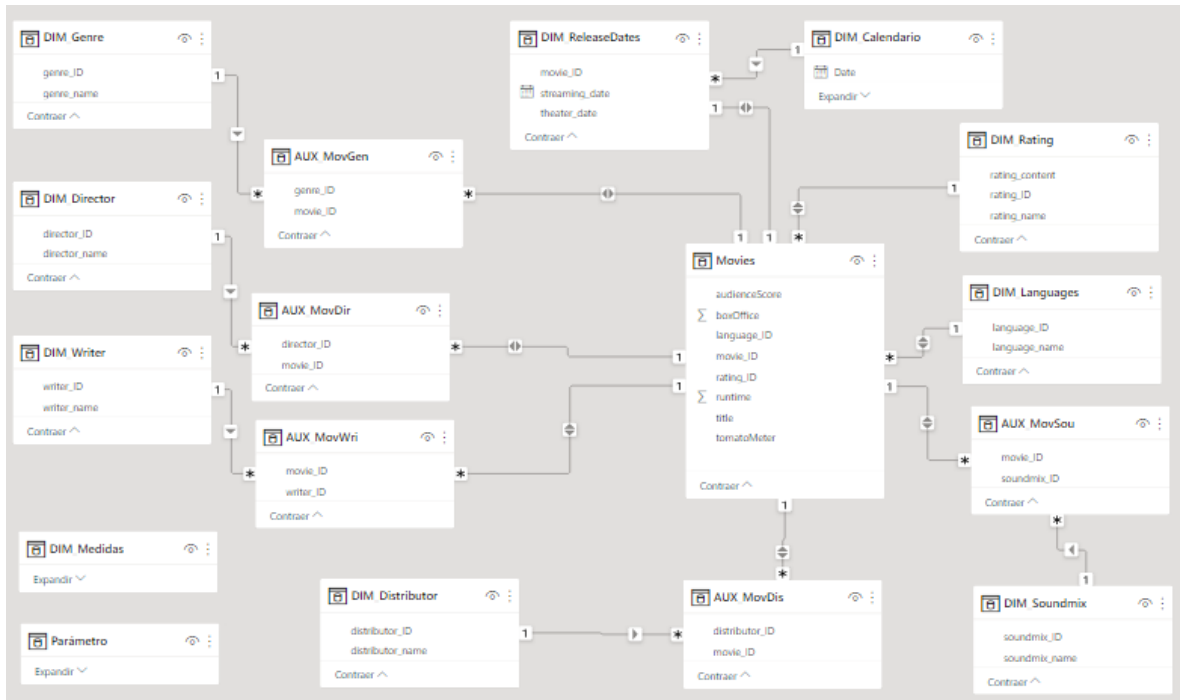
<https://docs.google.com/spreadsheets/d/e/2PACX-1vSfktKT8kQzCxQvzudntNINo2hoOhdG0c9kP113d2azGQq6KIARKi-Q0X-91odSaWuJaevDg4KHT6vR/pubhtml>

Esto facilita la visualización del proyecto evitando la necesidad de descargar la base, así como también su actualización posible.

### **V.II. Modelo de Datos Relacional**

Como se especificó anteriormente, este dataset necesitaba de varias tablas puente/auxiliares, y se detallará con cada una de sus transformaciones correspondientes a continuación:

[Imagen del modelo](#)



### V.III. Transformaciones aplicadas

#### Tabla Movies:

- = Table.TransformColumnTypes("#Encabezados promovidos",{{"id", type text}, {"title", type text}, {"audienceScore", Int64.Type}, {"tomatoMeter", Int64.Type}, {"rating", type text}, {"ratingContents", type text}, {"releaseDateTheaters", type date}, {"releaseDateStreaming", type date}, {"runtimeMinutes", Int64.Type}, {"genre", type text}, {"originalLanguage", type text}, {"director", type text}, {"writer", type text}, {"boxOffice", type text}, {"distributor", type text}, {"soundMix", type text}})
- = Table.NestedJoin("#Tipo cambiado", {"originalLanguage"}, DIM\_Languages, {"language\_name"}, "DIM\_Languages", JoinKind.LeftOuter)
- = Table.ExpandTableColumn("#Consultas combinadas(DIM\_Languages)", "DIM\_Languages", {"language\_ID"}, {"language\_ID"})
- = Table.ReorderColumns("#Se expandió DIM\_Languages",{ "id", "title", "audienceScore", "tomatoMeter", "rating", "ratingContents", "releaseDateTheaters", "releaseDateStreaming", "runtimeMinutes", "genre", "language\_ID", "originalLanguage", "director", "writer", "boxOffice",

"distributor", "soundMix"))

- = Table.RemoveColumns("#Columnas reordenadas(DIM\_Languages)","{"originalLanguage"})
- = Table.NestedJoin("#Columnas quitadas(DIM\_Languages)", {"rating"}, DIM\_Rating, {"rating\_name"}, "DIM\_Rating", JoinKind.LeftOuter)
- = Table.ExpandTableColumn("#Consultas combinadas1(DIM\_Rating)", "DIM\_Rating", {"rating\_ID"}, {"rating\_ID"})
- = Table.ReorderColumns("#Se expandió DIM\_Rating",{"id", "title", "audienceScore", "tomatoMeter", "rating\_ID", "rating", "ratingContents", "releaseDateTheaters", "releaseDateStreaming", "runtimeMinutes", "genre", "language\_ID", "director", "writer", "boxOffice", "distributor", "soundMix"})
- = Table.RemoveColumns("#Columnas reordenadas1(DIM\_Rating)","{"rating", "ratingContents"})
- = Table.RenameColumns("#Columnas quitadas1(DIM\_Rating)","{"id", "movie\_ID"})
- = Table.Distinct("#Columnas con nombre cambiado", {"movie\_ID"})
- = Table.RemoveColumns("#Duplicados quitados(movie\_IDs vacios)","{"genre", "director", "writer", "distributor", "soundMix", "releaseDateTheaters", "releaseDateStreaming"})
- = Table.RenameColumns("#Columnas quitadas(DIMS con AUX)","{"runtimeMinutes", "runtime"})
- = Table.TransformColumns("#Columnas con nombre cambiado1",{"boxOffice", Text.Clean, type text})
- = Table.ReplaceValue("#Texto limpio(boxOffice)","\$"," ",Replacer.ReplaceText,{"boxOffice"})
- = Table.ReplaceValue("#Valor reemplazado(boxOffice)","K","00",Replacer.ReplaceText,{"boxOffice"})

- = Table.ReplaceValue("#Valor  
reemplazado1(boxOffice)","M","00000",Replacer.ReplaceText,{"boxOffice"})
- = Table.ReplaceValue("#Valor  
reemplazado2(boxOffice)",".",",",Replacer.ReplaceText,{"boxOffice"})
- = Table.TransformColumnTypes("#Valor  
reemplazado3(boxOffice)",{{"boxOffice", Int64.Type}})

**Nota:** Para evitar la redundancia de registros al separar tablas de dimensiones, se indicará con un separador que las transformaciones previas en el historial provienen de la tabla “Movies”

#### Tabla DIM\_Languages:

- [heredado de Movies]
- = Table.SelectColumns("#Tipo cambiado",{"originalLanguage"})
- = Table.Distinct("#Otras columnas quitadas")
- = Table.Sort("#Duplicados quitados",{{"originalLanguage",  
Order.Ascending}})
- = Table.AddIndexColumn("#Filas ordenadas", "Índice", 1, 1, Int64.Type)
- = Table.ReorderColumns("#Índice agregado",{"Índice", "originalLanguage"})
- = Table.RenameColumns("#Columnas reordenadas",{{"originalLanguage",  
"language\_name"}, {"Índice", "language\_ID"}})

#### Tabla DIM\_Rating:

- [heredado de Movies]
- = Table.SelectColumns("#Columnas quitadas",{"rating", "ratingContents"})
- = Table.Distinct("#Otras columnas quitadas", {"rating"})
- = Table.Sort("#Duplicados quitados",{{"rating", Order.Ascending}})

- = Table.AddIndexColumn("#Filas ordenadas", "Índice", 1, 1, Int64.Type)
- = Table.ReorderColumns("#Índice agregado",{ "Índice", "rating", "ratingContents"})
- = Table.RenameColumns("#Columnas reordenadas1",{ "Índice", "rating\_ID", {"rating", "rating\_name"}, {"ratingContents", "rating\_content"}})

#### Tabla DIM\_Genre:

- [*heredado de Movies*]
- = Table.SelectColumns("#Columnas quitadas1(DIM\_Rating)","genre")
- = Table.ExpandListColumn(Table.TransformColumns("#Otras columnas quitadas", {{"genre", Splitter.SplitTextByDelimiter(",", QuoteStyle.None), let itemType = (type nullable text) meta [Serialized.Text = true] in type {itemType}}}), "genre")
- = Table.TransformColumnTypes("#Dividir columna por delimitador",{ "genre", type text})
- = Table.TransformColumns("#Tipo cambiado1",{ "genre", Text.Trim, type text})
- = Table.ExpandListColumn(Table.TransformColumns("#Texto recortado", {{"genre", Splitter.SplitTextByDelimiter("&", QuoteStyle.None), let itemType = (type nullable text) meta [Serialized.Text = true] in type {itemType}}}), "genre")
- = Table.TransformColumnTypes("#Dividir columna por delimitador1",{ "genre", type text})
- = Table.TransformColumns("#Tipo cambiado2",{ "genre", Text.Trim, type text})
- = Table.TransformColumns("#Texto recortado1",{ "genre", Text.Proper, type text})

- = Table.Distinct("#Poner En Mayúsculas Cada Palabra")
- = Table.Sort("#Duplicados quitados",{{"genre", Order.Ascending}})
- = Table.AddIndexColumn("#Filas ordenadas", "Índice", 1, 1, Int64.Type)
- = Table.ReorderColumns("#Índice agregado",{{"Índice", "genre"}})
- = Table.RenameColumns("#Columnas reordenadas",{{"Índice", "genre\_ID"}, {"genre", "genre\_name"}})

#### Tabla AUX MovGen:

- [*heredado de Movies*]
- = Table.SelectColumns("#Columnas quitadas1(DIM\_Rating)","{id", "genre"}")
- = Table.ExpandListColumn(Table.TransformColumns("#Otras columnas quitadas", {{"genre", Splitter.SplitTextByDelimiter(",", QuoteStyle.None), let itemType = (type nullable text) meta [Serialized.Text = true] in type {itemType}}}), "genre")
- = Table.TransformColumnTypes("#Dividir columna por delimitador",{{"genre", type text}})
- = Table.ExpandListColumn(Table.TransformColumns("#Tipo cambiado1", {{"genre", Splitter.SplitTextByDelimiter("&", QuoteStyle.Csv), let itemType = (type nullable text) meta [Serialized.Text = true] in type {itemType}}}), "genre")
- = Table.TransformColumnTypes("#Dividir columna por delimitador1",{{"genre", type text}})
- = Table.TransformColumns("#Tipo cambiado2",{{"genre", Text.Trim, type text}})
- = Table.TransformColumns("#Texto recortado",{{"genre", Text.Proper, type text}})

- = Table.NestedJoin("#Poner En Mayúsculas Cada Palabra", {"genre"}, DIM\_Genre, {"genre\_name"}, "DIM\_Genre", JoinKind.LeftOuter)
- = Table.ExpandTableColumn("#Consultas combinadas", "DIM\_Genre", {"genre\_ID"}, {"genre\_ID"})
- = Table.RemoveColumns("#Se expandió DIM\_Genre", {"genre"})
- = Table.RenameColumns("#Columnas quitadas", {"id", "movie\_ID"})

#### Tabla DIM\_Director:

- [*heredado de Movies*]
- = Table.SelectColumns("#Duplicados quitados", {"director"})
- = Table.ExpandListColumn(Table.TransformColumns("#Otras columnas quitadas", {"director", Splitter.SplitTextByDelimiter(",", QuoteStyle.None), let itemType = (type nullable text) meta [Serialized.Text = true] in type {itemType}}}, "director")
- = Table.TransformColumnTypes("#Dividir columna por delimitador", {"director", type text})
- = Table.TransformColumns("#Tipo cambiado1", {"director", Text.Trim, type text})
- = Table.Distinct("#Texto recortado")
- = Table.Sort("#Duplicados quitados1", {"director", Order.Ascending})
- = Table.AddIndexColumn("#Filas ordenadas", "Índice", 1, 1, Int64.Type)
- = Table.ReorderColumns("#Índice agregado", {"Índice", "director"})
- = Table.RenameColumns("#Columnas reordenadas", {"Índice", "director\_ID"}, {"director", "director\_name"})



- = Table.Sort("#Columnas con nombre cambiado1",{{"director\_ID", Order.Ascending}})

#### Tabla AUX\_MovDir:

- [*heredado de Movies*]
- = Table.SelectColumns("#Columnas quitadas(AUX Y DIM Genre)","{movie\_ID", "director"}")
- = Table.ExpandListColumn(Table.TransformColumns("#Otras columnas quitadas", {{"director", Splitter.SplitTextByDelimiter(",", QuoteStyle.None), let itemType = (type nullable text) meta [Serialized.Text = true] in type {itemType}}}), "director")
- = Table.TransformColumnTypes("#Dividir columna por delimitador",{{"director", type text}})
- = Table.TransformColumns("#Tipo cambiado1",{{"director", Text.Trim, type text}})
- = Table.NestedJoin("#Texto recortado", {"director"}, DIM\_Director, {"director\_name"}, "DIM\_Director", JoinKind.LeftOuter)
- = Table.ExpandTableColumn("#Consultas combinadas", "DIM\_Director", {"director\_ID"}, {"director\_ID"})
- = Table.RemoveColumns("#Se expandió DIM\_Director",{{"director"}})

#### Tabla DIM\_Writer:

- [*heredado de Movies*]
- = Table.SelectColumns("#Columnas quitadas(AUX Y DIM Genre)","{writer}")
- = Table.ExpandListColumn(Table.TransformColumns("#Otras columnas quitadas", {{"writer", Splitter.SplitTextByDelimiter(",", QuoteStyle.None), let itemType = (type nullable text) meta [Serialized.Text = true] in type {itemType}}}), "writer")

- = Table.TransformColumnTypes("#Dividir columna por delimitador",{{"writer", type text}})
- = Table.TransformColumns("#Tipo cambiado1",{{"writer", Text.Trim, type text}})
- = Table.Distinct("#Texto recortado")
- = Table.Sort("#Duplicados quitados",{{"writer", Order.Ascending}})
- = Table.AddIndexColumn("#Filas ordenadas", "Índice", 1, 1, Int64.Type)
- = Table.ReorderColumns("#Índice agregado",{ "Índice", "writer"})
- = Table.RenameColumns("#Columnas reordenadas",{{"Índice", "writer\_ID"}, {"writer", "writer\_name"}})

#### Tabla AUX MovWri:

- [*heredado de Movies*]
- = Table.SelectColumns("#Columnas quitadas(AUX Y DIM Genre)",{"movie\_ID", "writer"})
- = Table.ExpandListColumn(Table.TransformColumns("#Otras columnas quitadas", {{"writer", Splitter.SplitTextByDelimiter(",", QuoteStyle.None), let itemType = (type nullable text) meta [Serialized.Text = true] in type {itemType}}}), "writer")
- = Table.TransformColumnTypes("#Dividir columna por delimitador",{{"writer", type text}})
- = Table.TransformColumns("#Tipo cambiado1",{{"writer", Text.Trim, type text}})
- = Table.NestedJoin("#Texto recortado", {"writer"}, DIM\_Writer, {"writer\_name"}, "DIM\_Writer", JoinKind.LeftOuter)

- = Table.ExpandTableColumn("#Consultas combinadas", "DIM\_Writer", {"writer\_ID"}, {"writer\_ID"})
- = Table.RemoveColumns("#Se expandió DIM\_Writer", {"writer"})

#### Tabla DIM\_Distributor:

- [*heredado de Movies*]
- = Table.SelectColumns("#Columnas quitadas(DIMS con AUX)", {"distributor"})
- = Table.Distinct("#Otras columnas quitadas")
- = Table.ExpandListColumn(Table.TransformColumns("#Duplicados quitados", {"distributor", Splitter.SplitTextByDelimiter(",", QuoteStyle.None), let itemType = (type nullable text) meta [Serialized.Text = true] in type {itemType}}}), "distributor")
- = Table.TransformColumnTypes("#Dividir columna por delimitador", {"distributor", type text})
- = Table.TransformColumns("#Tipo cambiado1", {"distributor", Text.Trim, type text})
- = Table.Distinct("#Texto recortado")
- = Table.Sort("#Duplicados quitados1", {"distributor", Order.Ascending})
- = Table.AddIndexColumn("#Filas ordenadas", "Índice", 1, 1, Int64.Type)
- = Table.ReorderColumns("#Índice agregado", {"Índice", "distributor"})
- = Table.RenameColumns("#Columnas reordenadas", {"distributor", "distributor\_name"}, {"Índice", "distributor\_ID"})

#### Tabla AUX\_MovDis:

- [*heredado de Movies*]

- = Table.SelectColumns("#Columnas quitadas(DIMS con AUX)","movie\_ID", "distributor"))
- = Table.ExpandListColumn(Table.TransformColumns("#Otras columnas quitadas", {{"distributor", Splitter.SplitTextByDelimiter(",", QuoteStyle.None), let itemType = (type nullable text) meta [Serialized.Text = true] in type {itemType}}}}, "distributor")
- = Table.TransformColumnTypes("#Dividir columna por delimitador",{{"distributor", type text}})
- = Table.TransformColumns("#Tipo cambiado1",{{"distributor", Text.Trim, type text}})
- = Table.NestedJoin("#Texto recortado", {"distributor"}, DIM\_Distributor, {"distributor\_name"}, "DIM\_Distributor", JoinKind.LeftOuter)
- = Table.ExpandTableColumn("#Consultas combinadas", "DIM\_Distributor", {"distributor\_ID"}, {"distributor\_ID"})
- = Table.RemoveColumns("#Se expandió DIM\_Distributor",{"distributor"})

#### Tabla DIM Soundmix:

- [*heredado de Movies*]
- = Table.SelectColumns("#Columnas quitadas(DIMS con AUX)","soundMix"))
- = Table.Distinct("#Otras columnas quitadas")
- = Table.ExpandListColumn(Table.TransformColumns("#Duplicados quitados", {{"soundMix", Splitter.SplitTextByDelimiter(",", QuoteStyle.None), let itemType = (type nullable text) meta [Serialized.Text = true] in type {itemType}}}}, "soundMix")
- = Table.TransformColumnTypes("#Dividir columna por delimitador",{{"soundMix", type text}})

- = Table.TransformColumns("#Tipo cambiado1",{{"soundMix", Text.Trim, type text}})
- = Table.Distinct("#Texto recortado")
- = Table.Sort("#Duplicados quitados1",{{"soundMix", Order.Ascending}})
- = Table.AddIndexColumn("#Filas ordenadas", "Índice", 1, 1, Int64.Type)
- = Table.ReorderColumns("#Índice agregado",{ "Índice", "soundMix"})
- = Table.RenameColumns("#Columnas reordenadas",{{"soundMix", "soundmix\_name"}, {"Índice", "soundmix\_ID"}})

#### Tabla AUX\_MovSou:

- [*heredado de Movies*]
- = Table.SelectColumns("#Columnas quitadas(DIMS con AUX)",{"movie\_ID", "soundMix"})
- = Table.ExpandListColumn(Table.TransformColumns("#Otras columnas quitadas", {{"soundMix", Splitter.SplitTextByDelimiter(",", QuoteStyle.None), let itemType = (type nullable text) meta [Serialized.Text = true] in type {itemType}}}), "soundMix")
- = Table.TransformColumnTypes("#Dividir columna por delimitador",{{"soundMix", type text}})
- = Table.TransformColumns("#Tipo cambiado1",{{"soundMix", Text.Trim, type text}})
- = Table.NestedJoin("#Texto recortado", {"soundMix"}, DIM\_Soundmix, {"soundmix\_name"}, "DIM\_Soundmix", JoinKind.LeftOuter)
- = Table.ExpandTableColumn("#Consultas combinadas", "DIM\_Soundmix", {"soundmix\_ID"}, {"soundmix\_ID"})
- = Table.RemoveColumns("#Se expandió DIM\_Soundmix",{ "soundMix"})

#### Tabla DIM\_ReleaseDates:

- [*heredado de Movies*]
- = Table.SelectColumns("#Columnas quitadas(DIMS con AUX)","{movie\_ID", "releaseDateTheaters", "releaseDateStreaming"}")
- = Table.RenameColumns("#Otras columnas quitadas",{{"releaseDateTheaters", "theater\_date"}, {"releaseDateStreaming", "streaming\_date"}})

### **V.IV. Tablas de medidas y medidas calculadas**

#### 1. Tabla DIM\_Calendario

Se generó a través de DAX la tabla calendario necesaria para inteligencia de tiempo

DIM\_Calendario =  
CALENDAR(MIN(DIM\_ReleaseDates[theater\_date]),MAX(DIM\_ReleaseDates[theater\_date]))

Siendo que este proyecto abarca una extensión de tiempo que supera los 100 años, se generó un grupo desde la interfaz visual para crear una columna con la década

Nombre *	Campo
<input type="text" value="Date (decadas)"/>	<input type="text" value="Date"/>
Tipo de grupo	Tipo de discretización
<input type="text" value="Discretizar"/>	<input type="text" value="Tamaño de las discretizaciones"/>
Valor mínimo	Valor máximo
<input type="text" value="04/10/1902"/>	<input type="text" value="21/04/2032"/>

La discretización divide datos numéricos o de fecha y hora en grupos de igual tamaño. Escriba un tamaño de discretización.

Tamaño de la discretización

<input type="text" value="10"/>	<input type="text" value="Años"/>
---------------------------------	-----------------------------------

## 2. Tabla DIM\_Medidas

Se creó una tabla vacía para albergar todas las medidas calculadas que se detallan a continuación:

- Avg\_audienceScore = AVERAGE(Movies[audienceScore])  
*Brinda el promedio de puntaje de audiencia para indicadores y gráficos donde el recuento no refleja los datos a presentar*
- Avg\_boxOffice = AVERAGE(Movies[boxOffice])  
*Brinda el promedio de recaudación para indicadores y gráficos donde el recuento no refleja los datos a presentar*
- Avg\_General = ([Avg\_audienceScore] + [Avg\_tomatoMeter])/2  
*Promedia entre puntajes de crítica y audiencia para unificarlos en medidas únicas*
- Avg\_tomatoMeter = AVERAGE(Movies[tomatoMeter])  
*Brinda el promedio de puntaje de audiencia para indicadores y gráficos donde el recuento no refleja los datos a presentar*
- Count\_Movie = COUNT(Movies[movie\_ID])  
*Hace un recuento del total de películas para aplicar en gráficos*
- Median\_AS = MEDIAN(Movies[audienceScore])  
*Calcula la media del puntaje de audiencia para comparar con el promedio (y ayuda a mostrar los valores atípicos)*

- Median\_BO = MEDIAN(Movies[boxOffice])  
*Calcula la media de recaudación para comparar con el promedio (y ayuda a mostrar los valores atípicos)*
  - Median\_AS = MEDIAN(Movies[tomatoMeter])  
*Calcula la media del puntaje de crítica para comparar con el promedio (y ayuda a mostrar los valores atípicos)*
  - BoxOffice variable total =  
SUM (Movies[boxOffice])  
+ SUM (Movies[boxOffice]) \* 'Parámetro'[Valor de Parámetro]  
/100  
*Permite la utilización posterior de un parámetro, pensado para ver una variación porcentual en función del valor seleccionado en el slider*
- Parámetro = GENERATESERIES(-100, 100, 10)
- Ultima\_Actualizacion = TODAY()  
*Levanta la fecha del día, para poder medir fácilmente cuando fue la última vez que se actualizó*

## VI. CONCLUSIÓN Y FUTURAS LÍNEAS

De las visualizaciones se desprende que, en principio, sólo para algunos períodos temporales la crítica y la audiencia tienen valores relativamente similares. En particular, de los 80 hacia el presente. Por sí solo, esto podría indicar que, aunque posean valores diferentes, en general, hay una tendencia a no distanciarse mas de 5 puntos porcentuales. También resulta llamativo que el promedio de puntaje de audiencia sea mejor al 60%, mientras que la crítica supera ese indicador. Las medianas soportan esta hipótesis, sin distanciarse en estos indicadores. Revisar también la apertura por género, muestra que, salvo donde los datos hacen picos, donde un caso da una diferencia del 44%, el resto lleva una cercanía mucho más adecuada a la conclusión anterior.

Puede concluirse también en base al análisis de la segunda pestaña, que no necesariamente los géneros más populares coinciden con los mejores promedios de recepción de audiencia ni de críticas, por lo que la moda de género responde a otros parámetros no analizados en tal caso. Según la división en mezclas de sonido, no puede vincularse tampoco el dato a los puntajes de manera conclusiva, sino mostrar el avance mismo de la tecnología utilizada.



Complementando toda la información anterior, la tercer página demuestra también que no hay una vinculación directa entre la crítica especializada y la recaudación de un film. La mediana representada como línea en este caso, demuestra que los casos de éxito subiendo el promedio son realmente datos atípicos por sobre la cantidad de la que se tiene información *(Nota: o probablemente la información no sea exacta, haya faltantes, inconsistencias, etc. Es un dataset más apuntado a puntajes que a recaudación.)*

Como última información para colorear el panorama, la segregación por rating agrega en función de la recaudación, que podría haber una vinculación con el público al que se apunta una película, donde sorprendentemente, se desprende que las categorías más destacadas, son las que sugieren control parental.

La pestaña de directores y escritores no se utiliza para un dato minucioso al respecto, pero puede ser útil también para la funcionalidad secundaria del dashboard, y encontrar a partir de ella, o en conjunto con la de género, una recomendación sencilla para alguien buscando disfrutar de una película. Las tablas al fondo pueden utilizarse para esto último con suma facilidad.

Puede agregarse a este informe abrir el parámetro de idioma original, que no fue tenido en cuenta, así como también comparativas entre streaming y cine clásico *(el análisis se centra sobre este último, por metodología SMART)*. También puede incluirse el contenido de los ratings, para profundizar sobre los motivos de los mismos. Por último, existe un dataset accesorio no procesado con el detalle de las puntuaciones tanto de audiencia como de crítica, que podría utilizarse, por ejemplo, para buscar patrones entre determinadas frases o palabras con puntajes finales, buscar coincidencias en la utilización de las mismas entre crítica y audiencia y demás.

## VII. BIBLIOGRAFÍA

- [https://www.kaggle.com/datasets/andrezaza/clapper-massive-rotten-tomatoes-movies-and-reviews?select=rotten\\_tomatoes\\_movies.csv](https://www.kaggle.com/datasets/andrezaza/clapper-massive-rotten-tomatoes-movies-and-reviews?select=rotten_tomatoes_movies.csv)  
(Consultado el 23/04/2023)
- <https://www.rottentomatoes.com/about> (Consultado el 23/04/2023)
- <https://www.apd.es/niveles-gestion-empresarial/> (Consultado el 06/05/2023)
- <https://support.zendesk.com/hc/es/articles/4408839402906-La-media-vs-la-mediana#:~:text=La%20media%20se%20utiliza%20para,caso%20de%20distribuciones%20num%C3%A9ricas%20sesgadas>. (Consultado el 17/06/2023)