



UNIVERSIDAD PERUANA DE CIENCIAS APLICADAS

SECCIÓN: CC51

GRUPO: 1

CURSO: Fundamentos de Data Science

PROFESOR(A): Nérida Isabel Manrique Tunque

TÍTULO: Trabajo Final

El presente trabajo ha sido realizado por:

Francesca Nicole Bances Torres (U202214090)

Marsi Valeria Figueroa Larragán (U202220990)

Mauricio Eduardo Vera Castellón (U20181H114)

Índice

1. INTRODUCCIÓN	2
2. INTEGRANTES Y ROLES	2
3. METODOLOGÍA CRISP-DM	3
4. MODELAR Y EVALUAR LOS RESULTADOS	17
5. CONCLUSIONES	19
6. BIBLIOGRAFÍA	20

1. INTRODUCCIÓN

En la era digital, el análisis de datos se ha convertido en una herramienta esencial para tomar decisiones informadas. Este proyecto tiene como objetivo desarrollar una propuesta de análisis y analítica a partir de un conjunto de datos sobre Tendencias de estadísticas de YouTube en Canadá, aplicando la metodología CRISP-DM. Con información de la consultora internacional con sede en Lima, contratada por una importante empresa de marketing digital, buscamos entender las tendencias de los videos de YouTube para optimizar sus campañas de marketing y tomar decisiones estratégicas basadas en datos concretos.

El conjunto de datos analizado, conocido como "Tendencias de las estadísticas de videos de YouTube," incluye registros diarios de los videos de mayor tendencia. A través de la metodología CRISP-DM, que abarca la comprensión de los datos, la preparación de los datos, el modelado, la evaluación, se busca crear conocimiento y valor, identificando patrones y tendencias que proporcionen información valiosa y fundamenten decisiones estratégicas.

2. INTEGRANTES Y ROLES

Alumno	Rol	Desempeño
Francesca Nicole Bances Torres (U202214090)	Data Analytics	En mi rol como analista de datos, recojo y limpio datos de diversas fuentes para asegurar su calidad. Realizó análisis exploratorios para identificar patrones y tendencias, y desarrolló modelos predictivos utilizando técnicas estadísticas. Interpreto los resultados y creo visualizaciones para comunicar mis hallazgos de manera efectiva. Además, elaboro informes y presentaciones para recomendar acciones basadas en los datos analizados.
Marsi Valeria Figueroa Larragán (U202220990)	Data Science	Como Data Scientist, me encargo de analizar y extraer valor de grandes volúmenes de datos utilizando técnicas de estadística, aprendizaje automático y análisis de datos. Mi labor incluye recolectar, limpiar y preprocesar datos, desarrollar modelos predictivos y algoritmos, y visualizar los resultados para facilitar la toma de decisiones estratégicas. Colaboro con equipos multidisciplinarios para identificar problemas y oportunidades de negocio, traduciendo las necesidades en soluciones basadas en datos que optimizan procesos y mejoran los resultados empresariales. Además, interpreto y comunico hallazgos complejos de manera clara y accesible a diferentes audiencias

		dentro de la organización.
Mauricio Eduardo Vera Castellón(U20181H114)	Data Engineer	Como data engineer en este proyecto, mi rol es esencial para asegurar la correcta gestión y procesamiento de los datos de YouTube en Canadá, siguiendo la metodología CRISP-DM. Me encargo de la adquisición y consolidación de datos, integrando múltiples fuentes para formar un conjunto de datos unificado y confiable. Diseño y construyó procesos eficientes para la limpieza, transformación y almacenamiento de los datos. Implemento y gestiono infraestructuras de datos seguros, garantizando la calidad y accesibilidad de los datos para su análisis. Colaboró estrechamente con los data scientists para proporcionar los datos necesarios y optimizados, facilitando así el desarrollo de modelos predictivos y análisis descriptivos precisos y efectivos.

3. METODOLOGÍA CRISP-DM

Consiste en la comprensión y descripción de los objetivos y requisitos del proyecto.

❖ Objetivos del proyecto

El objetivo general del proyecto es realizar un análisis de las tendencias de videos en Youtube de Canadá, con el propósito de cumplir los siguientes objetivos propuestos por una empresa de marketing digital:

- Análisis de tendencias por categoría de videos:
 - Identificar y clasificar las categorías de videos más populares y tendencias en el mercado.
 - Evaluar las preferencias de los usuarios mediante el análisis de las categorías con mayor y menor cantidad de interacciones positivas (me gusta) y negativas (no me gusta).
 - Calcular las proporciones de me gusta/no me gusta y vistas/comentarios para cada categoría de videos, proporcionando insights sobre la recepción del contenido por parte del público.
- Análisis temporal de tendencias:
 - Comprender cómo ha variado el volumen de los videos en tendencia a lo largo del tiempo, identificando patrones o tendencias significativas.
- Análisis por canales de Youtube:
 - Identificar los canales de YouTube que presentan más frecuentemente videos en tendencia y aquellos que tienen menor frecuencia de aparición en este tipo de listas.
- Análisis geográfico:
 - Determinar los estados o regiones del país donde se concentran las mayores vistas, me gusta y no me gusta de los videos, ofreciendo una visión geográfica detallada del comportamiento del público.
- Predicción:

- Desarrollar modelos predictivos que permitan predecir el número de vistas, me gusta y no me gusta de los videos.

Finalmente, como objetivo específico buscamos mejorar las campañas de marketing digital, aumentando la interacción y la participación de la audiencia. Asimismo, con las predicciones métricas deseamos facilitar la planificación estratégica y la creación de iniciativas más efectivas que se adapten a las tendencias cambiantes de YouTube.

❖ Objetivos de Data Science

Para este análisis de tendencias de videos en YouTube en Canadá, estructuramos los objetivos de Data Science de la siguiente manera para cumplir con las metas propuestas por la empresa de marketing digital:

1. Identificación de variables dependientes e independientes:
 - a. Identificar las variables independientes que influyen en las tendencias de videos en YouTube, como la categoría del video, el canal, la fecha de publicación, entre otras.
 - b. Determinar la variable dependiente, que podría ser el número de vistas, me gusta, no me gusta o comentarios, dependiendo del objetivo específico del análisis.
2. Limpieza y Preprocesamiento de Datos:
 - a. Realizar una limpieza de los datos obtenidos de los conjuntos de datos de YouTube para garantizar la calidad y consistencia de la información.
 - b. Tratar los datos faltantes, valores atípicos y errores para asegurar la integridad de los análisis y modelos predictivos.
3. Explorar Datos:
 - a. Utilizar técnicas de visualización adecuadas para representar las variables relevantes del conjunto de datos de YouTube.
4. Modelado Predictivo:
 - a. Desarrollar modelos predictivos que utilicen las variables identificadas para predecir las métricas clave de interacción de los videos.
 - b. Evaluar los resultados de los modelos para asegurar su utilidad en la planificación estratégica y la toma de decisiones.
5. Resultados:
 - a. Presentar hallazgos y recomendaciones derivadas del análisis a través de informes detallados y visualizaciones comprensibles.
 - b. Facilitar la implementación de insights en estrategias de marketing digital para mejorar la interacción y participación del público en YouTube.

❖ Comprensión de los datos

El dataset utilizado para el trabajo ha sido proporcionado por la Universidad Peruana de Ciencias Aplicadas. Este contiene 20 variables y x observaciones

Dataset original: [Trending YouTube Video Statistics](#)

Diccionario de datos:

Nombre de variable	Descripción
video_id	El identificador único del video en YouTube.
trending_date	La fecha en la que el video fue tendencia
title	El título del video.
channel_title	El nombre del canal que subió el video.
category_id	El identificador de la categoría del video.
publish_time	La fecha y hora en que el video fue publicado.
tags	La fecha y hora en que el video fue publicado.
views	El número de vistas del video.
likes	El número de "me gusta" del video.
dislikes	El número de "no me gusta" del video.
comment_count	El número de comentarios en el video.
thumbnail_link	El enlace a la miniatura del video.
comments_disabled	Indica si los comentarios están deshabilitados para el video.
ratings_disabled	Indica si las calificaciones están deshabilitadas para el video.
video_error_or_removed	Indica si el video tiene un error o ha sido removido.
description	La descripción del video.
state	Nombre del Estado perteneciente al país
lat	La latitud geográfica de ubicación del Estado.
lon	Longitud geográfica de ubicación del Estado.
geometry	Registra las coordenadas de las geometrías donde su ubica el Estado dentro del planeta. Es de utilidad si se decide utilizar la librería GeoPandas para la elaboración de mapas.

❖ CARGAR DATOS

```
data = pd.read_csv("CAvideos_cc50_202101.csv")
data.head()
```

	video_id	trending_date	title	channel_title	category_id	publish_time	tags	views	likes	dislikes	comment_count	thumbnail_link
0	n1WpP7lowLc	17.14.11	Eminem - Walk On Water (Audio) ft. Beyoncé	EminentVEVO	10	2017-11-10T17:00:03.000Z	Eminem "Walk On Water" "Aftermath/Shady/In...	17158579	787425	43420	125882	https://i.ytimg.com/vi/n1WpP7lowLc/h
1	0dBkQ4Mz1M	17.14.11	PLUSH - Bad Unboxing Fan Mail	iDubbzbTV	23	2017-11-13T17:00:00.000Z	plush "bad unboxing" "unboxing" "fan mail" "id...	1014651	127794	1688	13030	https://i.ytimg.com/vi/0dBkQ4Mz1M/h
2	5qpjK5DgC14	17.14.11	Racist Superman Rudy Mancuso, King Bach & Le...	Rudy Mancuso	23	2017-11-12T19:05:24.000Z	racist superman "rudy" "mancuso" "king" "bach"...	3191434	146035	5339	8181	https://i.ytimg.com/vi/5qpjK5DgC14/h
3	d380meDOW0M	17.14.11	I Dare You: GOING BALDI?	nigahiga	24	2017-11-12T18:01:41.000Z	ryan "higa" "higatv" "nigahiga" "i dare you" "...	2095828	132239	1989	17518	https://i.ytimg.com/vi/d380meDOW0M/h
4	2Vv-BfVoq4g	17.14.11	Ed Sheeran - Perfect (Official Music Video)	Ed Sheeran	10	2017-11-09T11:04:14.000Z	edsheeran "ed sheeran" "acoustic" "live" "cove...	33523622	1634130	21082	85067	https://i.ytimg.com/vi/2Vv-BfVoq4g/h

Gracias a esta función podemos cargar el csv para poder visualizar la información necesaria

❖ INSPECCIONAR DATOS

```
[ ] type(data)
```

→	pandas.core.frame.DataFrame																																																															
type(category_data)																																																																
dict																																																																
data.describe()																																																																
→	<table border="1"> <thead> <tr> <th></th><th>views</th><th>likes</th><th>dislikes</th><th>comment_count</th><th>lat</th><th>lon</th></tr> </thead> <tbody> <tr> <td>count</td><td>4.088100e+04</td><td>4.088100e+04</td><td>4.088100e+04</td><td>4.088100e+04</td><td>40881.000000</td><td>40881.000000</td></tr> <tr> <td>mean</td><td>1.147036e+06</td><td>3.958269e+04</td><td>2.009195e+03</td><td>5.042975e+03</td><td>52.025876</td><td>-88.817702</td></tr> <tr> <td>std</td><td>3.390913e+06</td><td>1.326895e+05</td><td>1.900837e+04</td><td>2.157902e+04</td><td>7.213076</td><td>25.119498</td></tr> <tr> <td>min</td><td>7.330000e+02</td><td>0.000000e+00</td><td>0.000000e+00</td><td>0.000000e+00</td><td>44.566645</td><td>-139.000002</td></tr> <tr> <td>25%</td><td>1.439020e+05</td><td>2.191000e+03</td><td>9.900000e+01</td><td>4.170000e+02</td><td>46.249282</td><td>-110.733329</td></tr> <tr> <td>50%</td><td>3.712040e+05</td><td>8.780000e+03</td><td>3.030000e+02</td><td>1.301000e+03</td><td>49.822578</td><td>-81.236083</td></tr> <tr> <td>75%</td><td>9.633020e+05</td><td>2.871700e+04</td><td>9.500000e+02</td><td>3.713000e+03</td><td>53.016698</td><td>-64.347995</td></tr> <tr> <td>max</td><td>1.378431e+08</td><td>5.053338e+06</td><td>1.602383e+06</td><td>1.114800e+06</td><td>68.767467</td><td>-57.426919</td></tr> </tbody> </table>		views	likes	dislikes	comment_count	lat	lon	count	4.088100e+04	4.088100e+04	4.088100e+04	4.088100e+04	40881.000000	40881.000000	mean	1.147036e+06	3.958269e+04	2.009195e+03	5.042975e+03	52.025876	-88.817702	std	3.390913e+06	1.326895e+05	1.900837e+04	2.157902e+04	7.213076	25.119498	min	7.330000e+02	0.000000e+00	0.000000e+00	0.000000e+00	44.566645	-139.000002	25%	1.439020e+05	2.191000e+03	9.900000e+01	4.170000e+02	46.249282	-110.733329	50%	3.712040e+05	8.780000e+03	3.030000e+02	1.301000e+03	49.822578	-81.236083	75%	9.633020e+05	2.871700e+04	9.500000e+02	3.713000e+03	53.016698	-64.347995	max	1.378431e+08	5.053338e+06	1.602383e+06	1.114800e+06	68.767467	-57.426919
	views	likes	dislikes	comment_count	lat	lon																																																										
count	4.088100e+04	4.088100e+04	4.088100e+04	4.088100e+04	40881.000000	40881.000000																																																										
mean	1.147036e+06	3.958269e+04	2.009195e+03	5.042975e+03	52.025876	-88.817702																																																										
std	3.390913e+06	1.326895e+05	1.900837e+04	2.157902e+04	7.213076	25.119498																																																										
min	7.330000e+02	0.000000e+00	0.000000e+00	0.000000e+00	44.566645	-139.000002																																																										
25%	1.439020e+05	2.191000e+03	9.900000e+01	4.170000e+02	46.249282	-110.733329																																																										
50%	3.712040e+05	8.780000e+03	3.030000e+02	1.301000e+03	49.822578	-81.236083																																																										
75%	9.633020e+05	2.871700e+04	9.500000e+02	3.713000e+03	53.016698	-64.347995																																																										
max	1.378431e+08	5.053338e+06	1.602383e+06	1.114800e+06	68.767467	-57.426919																																																										

```
[ ] data.info()

→ <class 'pandas.core.frame.DataFrame'>
RangeIndex: 40881 entries, 0 to 40880
Data columns (total 20 columns):
 #   Column           Non-Null Count  Dtype  
--- 
 0   video_id        40881 non-null   object  
 1   trending_date   40881 non-null   object  
 2   title            40881 non-null   object  
 3   channel_title   40881 non-null   object  
 4   category_id     40881 non-null   object  
 5   publish_time    40881 non-null   object  
 6   tags              40881 non-null   object  
 7   views             40881 non-null   int64  
 8   likes             40881 non-null   int64  
 9   dislikes          40881 non-null   int64  
 10  comment_count   40881 non-null   int64  
 11  thumbnail_link  40881 non-null   object  
 12  comments_disabled 40881 non-null   bool   
 13  ratings_disabled 40881 non-null   bool   
 14  video_error_or_removed 40881 non-null   bool  
 15  description      39585 non-null   object  
 16  state             40881 non-null   object  
 17  lat               40881 non-null   float64 
 18  lon               40881 non-null   float64 
 19  geometry          40881 non-null   object  
dtypes: bool(3), float64(2), int64(4), object(11)
memory usage: 5.4+ MB
```

```
[ ] len(data.video_id.unique())
```

```
→ 24427
```

Observamos que el total de datos son 40881 y solo 24427 tiene video_id unico

```
[ ] data.columns.values

→ array(['video_id', 'trending_date', 'title', 'channel_title',
       'category_id', 'publish_time', 'tags', 'views', 'likes',
       'dislikes', 'comment_count', 'thumbnail_link', 'comments_disabled',
       'ratings_disabled', 'video_error_or_removed', 'description',
       'state', 'lat', 'lon', 'geometry'], dtype=object)
```

aquí vemos que el total de variables son 20

❖ PRE-PROCESAR DATOS

[]	data[data.duplicated()]
[] data[data['video_id'].duplicated()]	
0	video_id trending_date title channel_title category_id publish_time tags views likes dislikes comment_count thumbnail_link comments_disabled ratings_disabled video_error_or_removed description
200	0dBikQ4Mz1M 17.15.11 PLUSH - Bad Unboxing Fan Mail iDubbzbTV 23 2017-11-13T17:00:00.000Z plush["bad unboxing"] "unboxing" "fan mail" "id..." 2649977 193479 3496 17846 https://i.ytimg.com/vi/0dBik
214	n1WpP7lowLc 17.15.11 Eminem - Walk On Water (Audio) ft. Beyoncé EminemVEVO 10 2017-11-10T17:00:03.000Z Eminem["Walk" "On" "Water" "Aftermath/Shady/In..." 20539417 840642 47715 124236 https://i.ytimg.com/vi/n1Wp
216	teXaL6GdQRk 17.15.11 STRANGER JOKES Jokes de Papa avec les teens ... Le Jeu, C'est Sérieux 23 2017-11-13T15:48:57.000Z Stranger Jokes["Jokes de Papa"] "Stranger Thing..." 443131 27026 491 746 https://i.ytimg.com/vi/teXaL
218	2kyS6SvSYSE 17.15.11 WE WANT TO TALK ABOUT OUR MARRIAGE CaseyNeistat 22 2017-11-13T17:13:01.000Z SHANtell martin 2188675 88100 7150 24225 https://i.ytimg.com/vi/2kyS6
219	5qjK5DgCt4 17.15.11 Racist Superman Rudy Mancuso, King Bach & Le... Rudy Mancuso 23 2017-11-12T19:05:24.000Z racist superman["rudy"] "mancuso" "king" "bach"..." 4326684 167696 6730 9265 https://i.ytimg.com/vi/5qj
...	...
	HTC U12+

Aquí verificamos la duplicación de los datos

Se eliminarán las filas con video_id que se han duplicado

```
duplicated_ids = data[data['video_id'].duplicated()]['video_id'].unique()
data = data[~data['video_id'].isin(duplicated_ids)]
```

```
data.info()

<class 'pandas.core.frame.DataFrame'>
Index: 14516 entries, 5 to 40880
Data columns (total 20 columns):
 #   Column      Non-Null Count  Dtype  
--- 
 0   video_id    14516 non-null   object 
 1   trending_date 14516 non-null   object 
 2   title        14516 non-null   object 
 3   channel_title 14516 non-null   object 
 4   category_id  14516 non-null   category
 5   publish_time 14516 non-null   object 
 6   tags          14516 non-null   object 
 7   views         14516 non-null   int64  
 8   likes         14516 non-null   int64  
 9   dislikes       14516 non-null   int64  
 10  comment_count 14516 non-null   int64  
 11  thumbnail_link 14516 non-null   object 
 12  comments_disabled 14516 non-null   bool   
 13  ratings_disabled 14516 non-null   bool   
 14  video_error_or_removed 14516 non-null   bool   
 15  description    13775 non-null   object 
 16  state          14516 non-null   object 
 17  lat            14516 non-null   float64
 18  lon            14516 non-null   float64
 19  geometry       14516 non-null   object 
dtypes: bool(3), category(1), float64(2), int64(4), object(10)
memory usage: 1.9+ MB
```

Se eliminan las filas con video_id duplicadas y se utiliza data.info() para la verificación

❖ VISUALIZAR DATOS

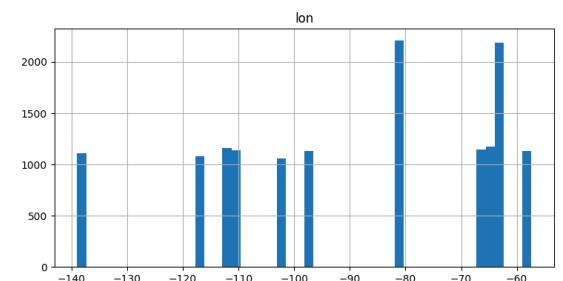
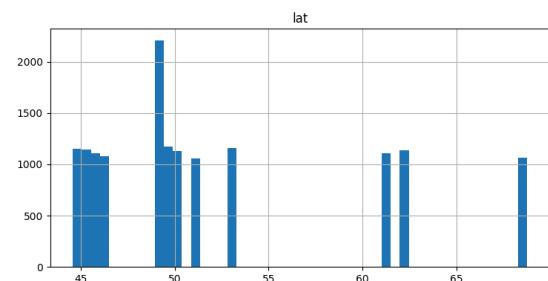
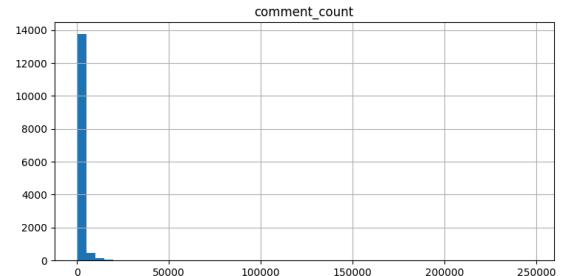
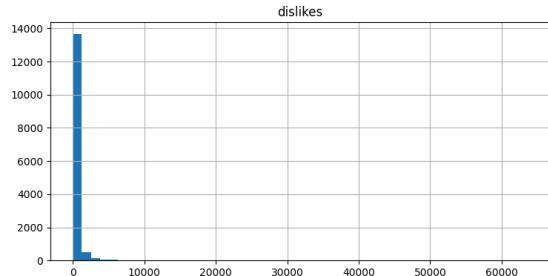
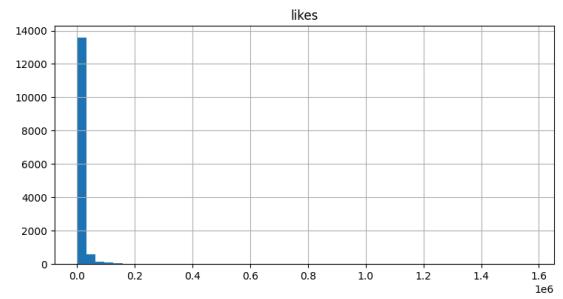
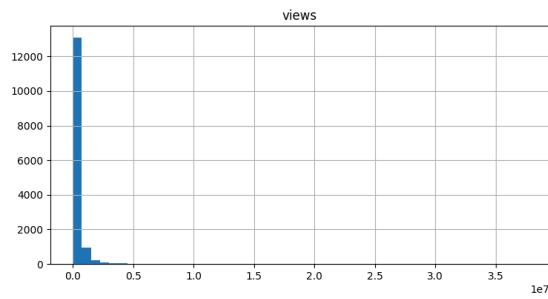
Tabla de frecuencia de la variable objetivo

Ya que piden analizar la tendencia, nuestra variable objetivo serían las views

```
views
108967    3
141897    3
61941     3
9924      3
95771     2
..
60161     1
184655    1
131486    1
5288      1
107392    1
Name: count, Length: 14319, dtype: int64
```

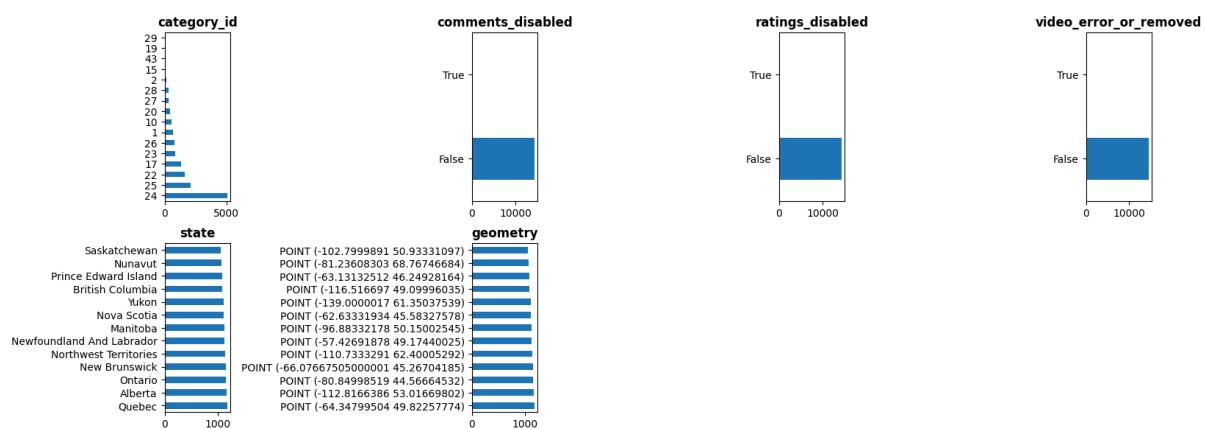
```
numericas= data.select_dtypes(include=['float64', 'int'])
numericas.hist(bins=50,figsize=(20,15))
plt.show()
```

En los siguientes recuadros mostramos las variables numéricas

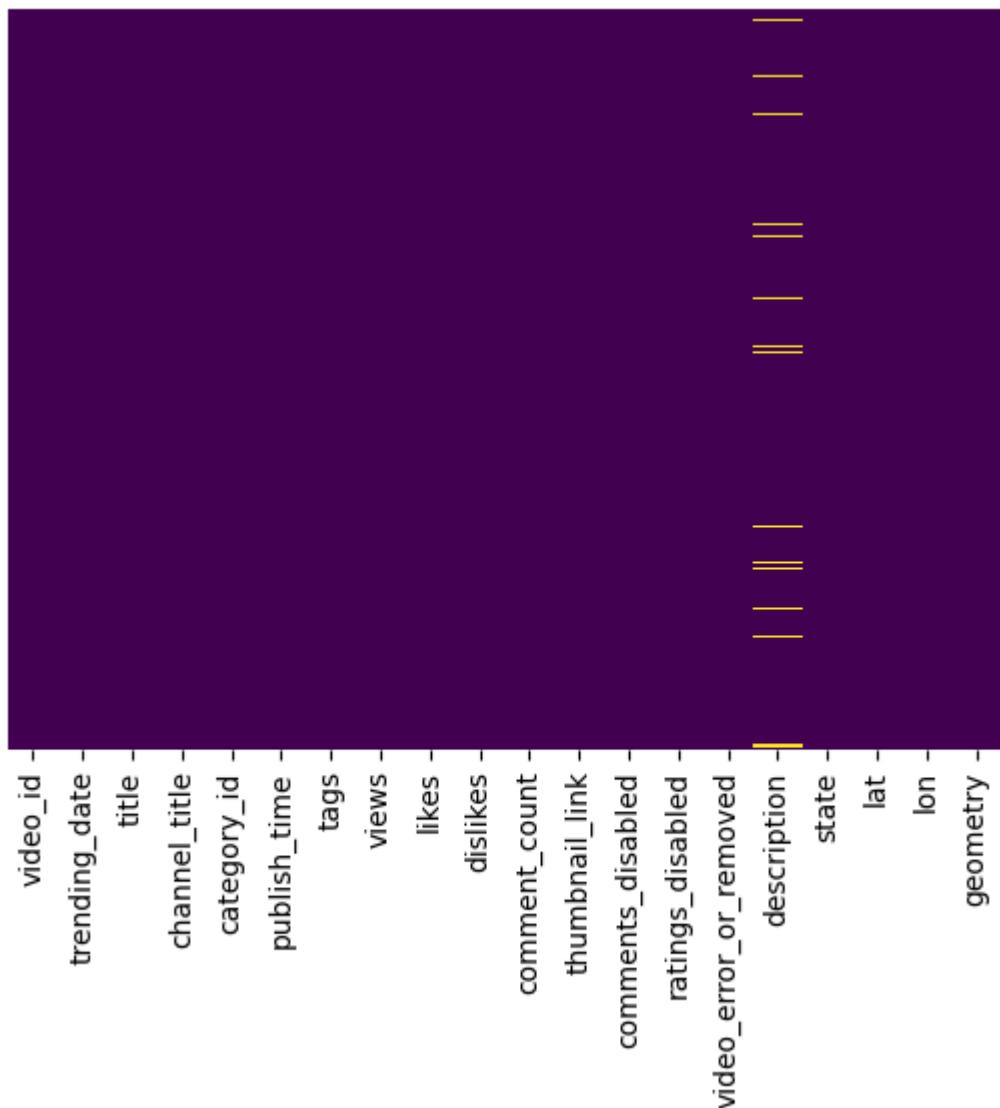


Distribución de variables Cualitativas

Distribución variables cualitativas



Verificar cantidad de nulos



```
[ ] data['description'].nunique()
```

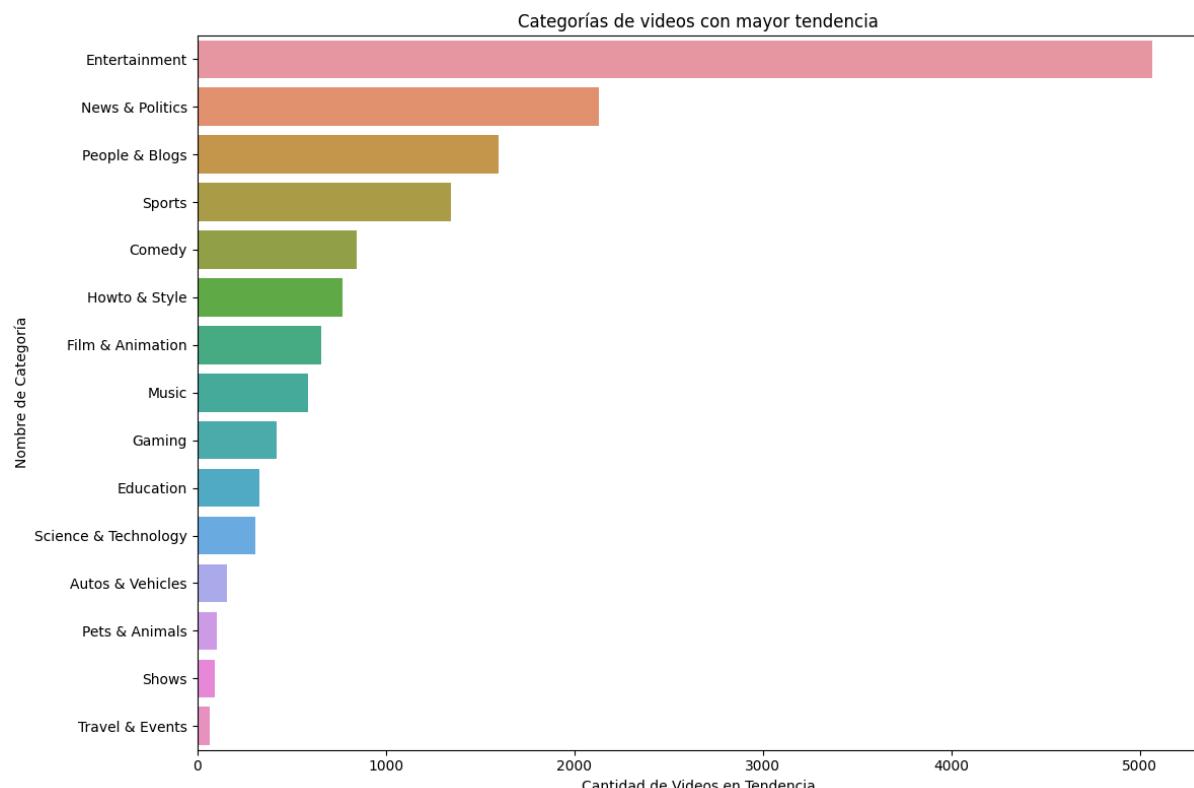
```
→ 12385
```

Entonces, se observa que la única columna con datos faltantes de la de "description". En total, esta columna tiene 1296 valores nulos, lo cual representa el 3.17% del total.

Además de ello, más de la mitad de sus valores son únicos, por lo que reemplazarlos no es una opción conveniente.

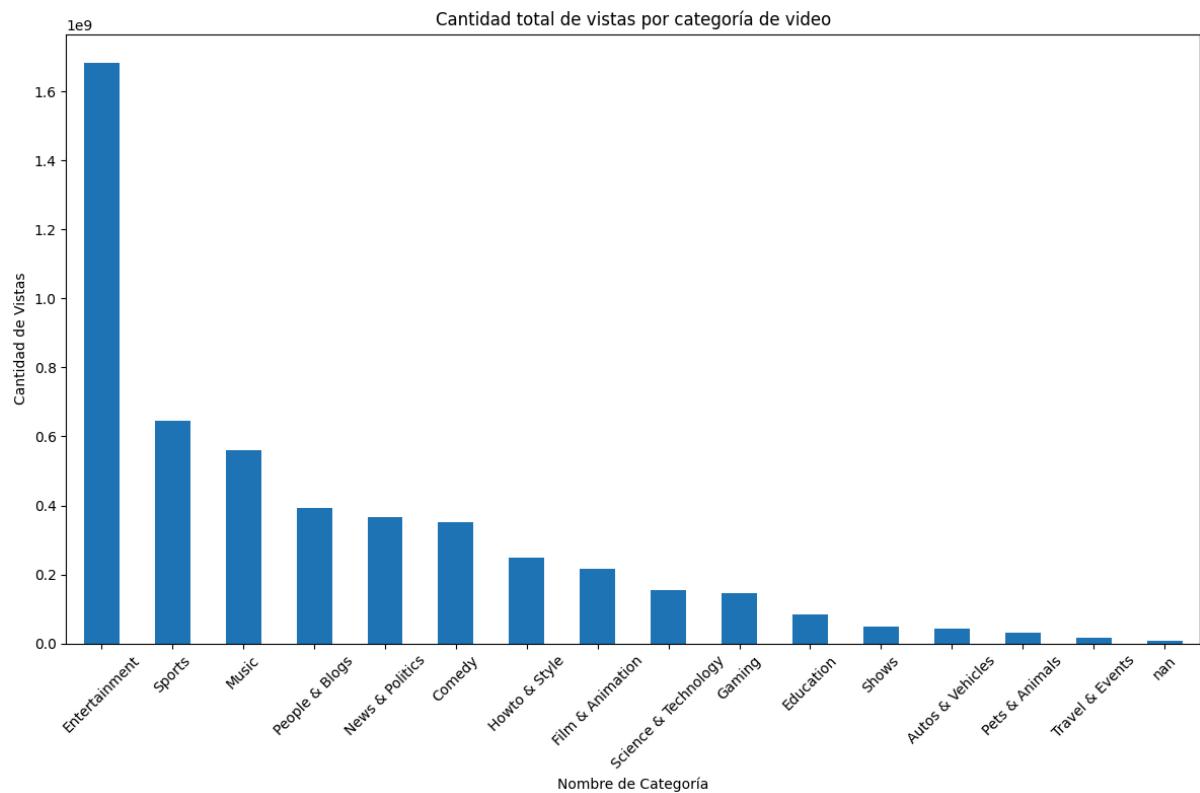
Por ello se eliminarán las filas que tengas datos nulos

¿Qué categorías de videos son las de mayor tendencia?



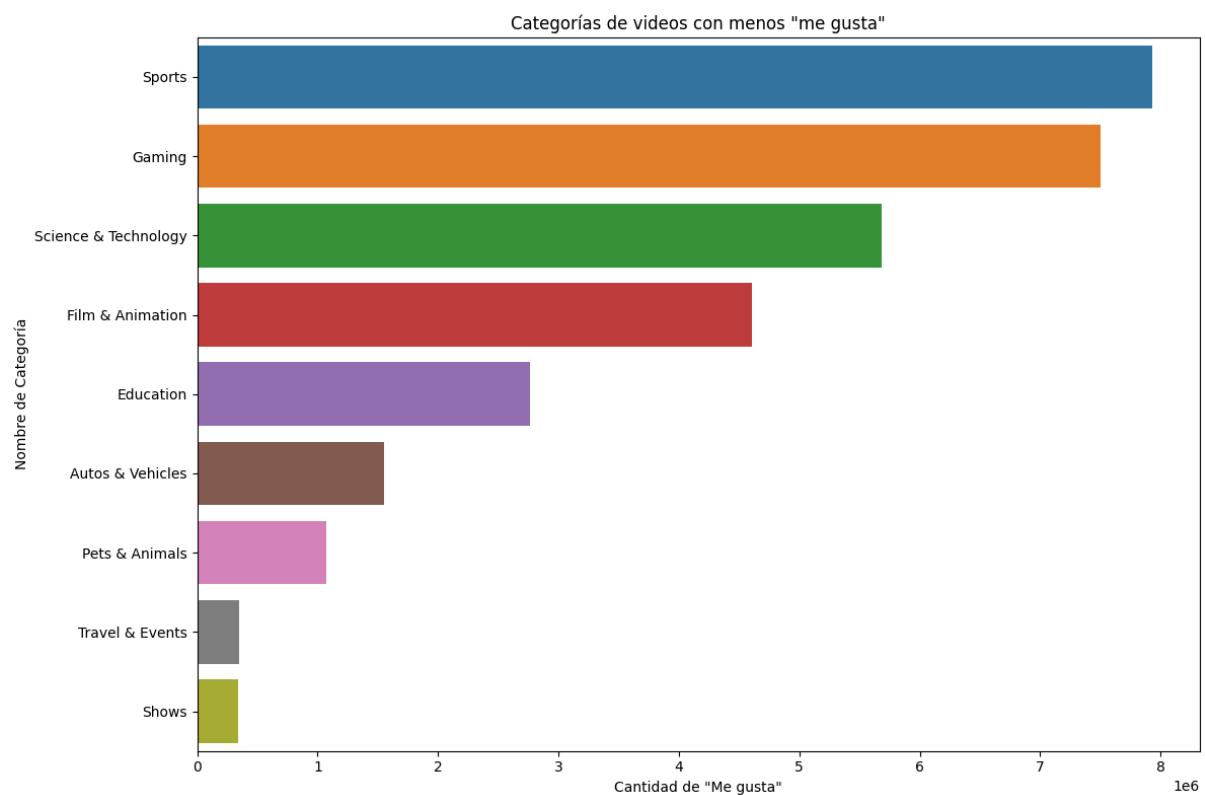
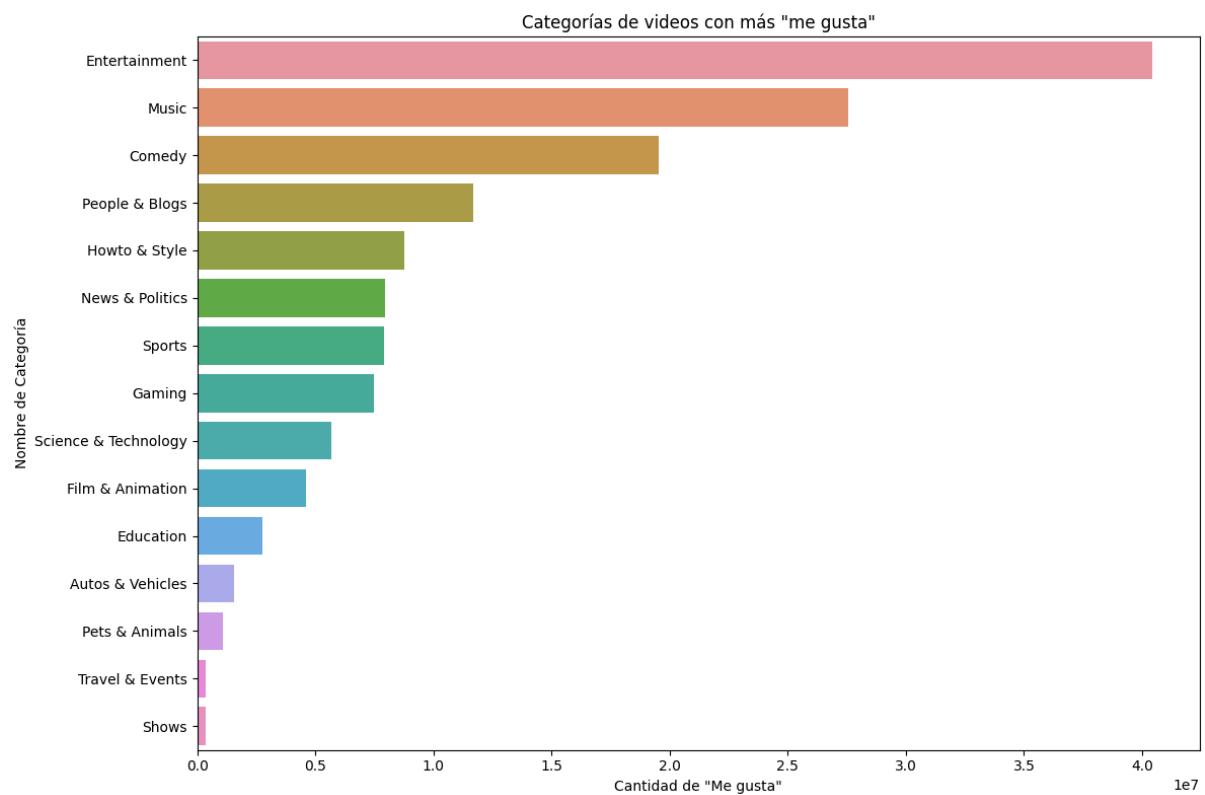
Se muestra la categoría con mayor tendencia (Entertainment) y la cantidad de videos en tendencia.

¿Qué categorías de video tienen más vistas?



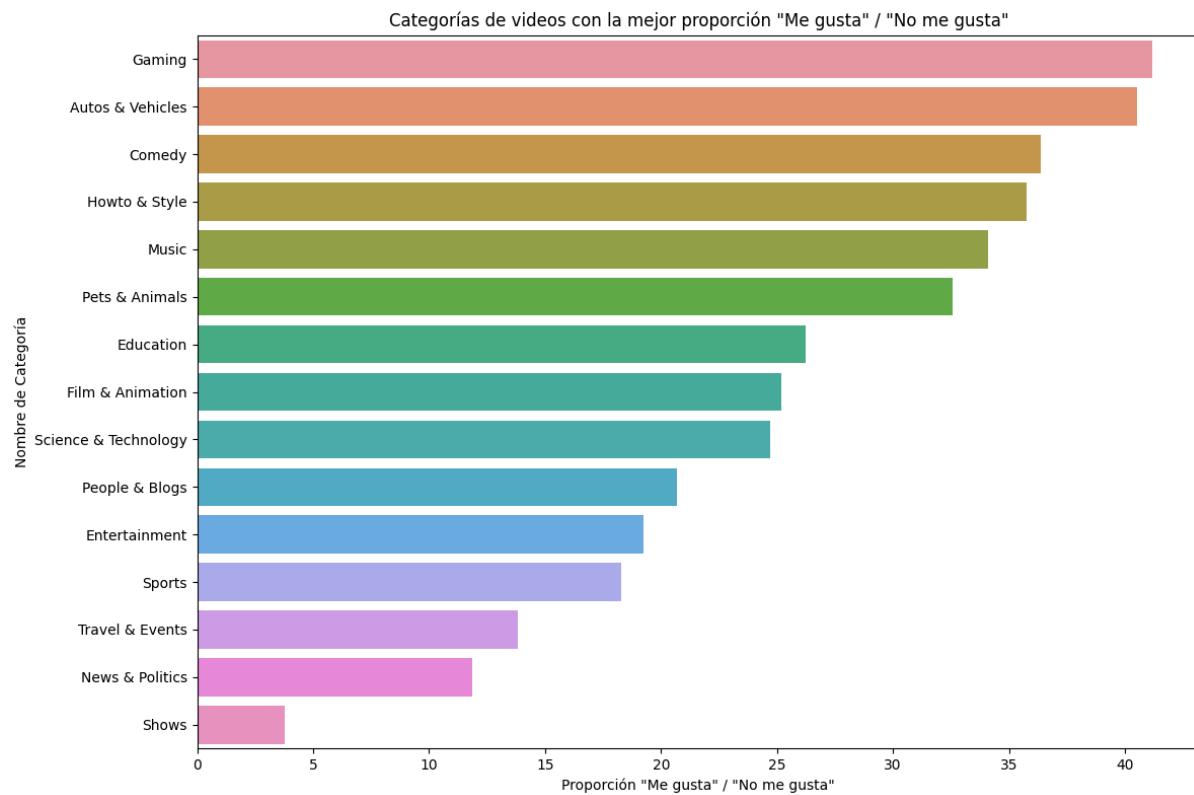
La categoría con mayor visitas sería Entertainment con un valor de 1.6, seguido por los sports y music, con un aproximado de 0.8 las dos categorías cuentan con valores bastante cercanos a comparación de entertainment.

¿Qué categorías de videos son los que más gustan? ¿Y las que menos gustan?



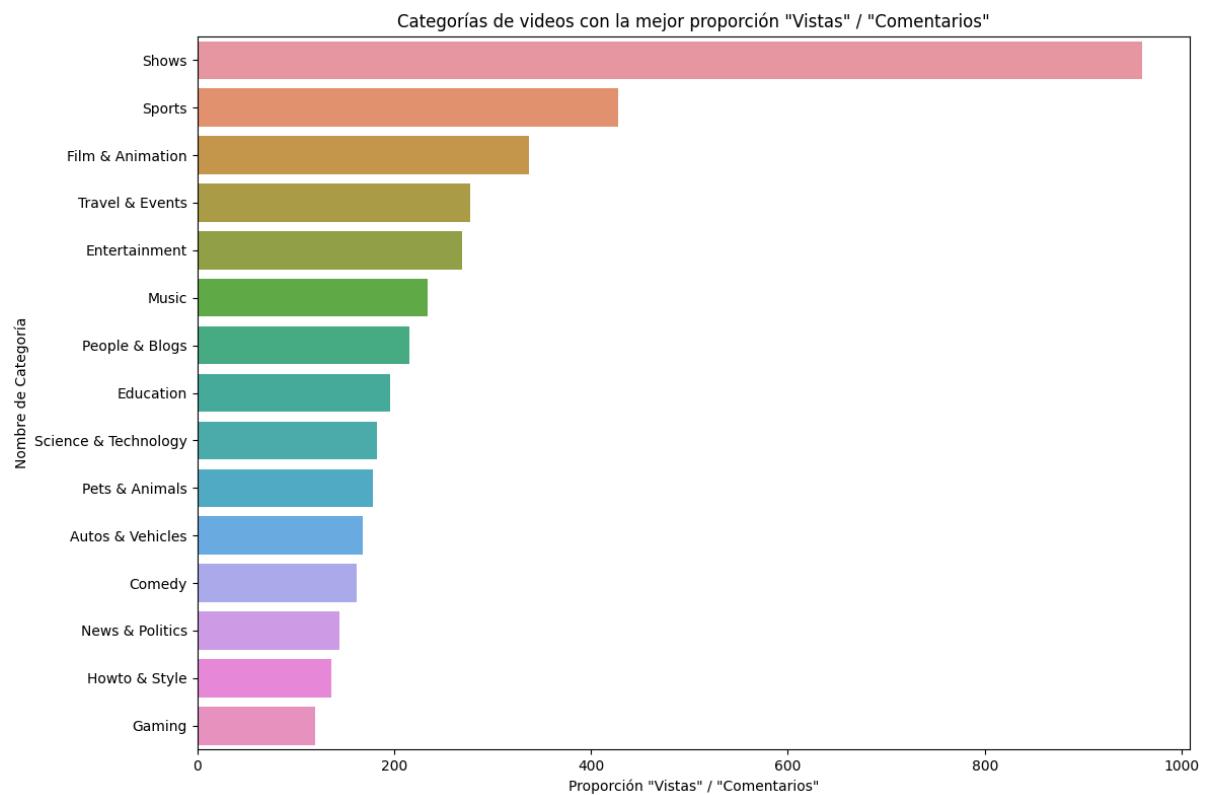
Analizando el gráfico podemos concluir que las tres categorías de videos que más gustan son Entertainment, Music y Comedy. Por otro lado, las tres categorías que menos gustan son las categorías de Travel & Events, Shows y Movies.

¿Qué categorías de videos tienen la mejor proporción (ratio) de “Me gusta” / “No me gusta”?



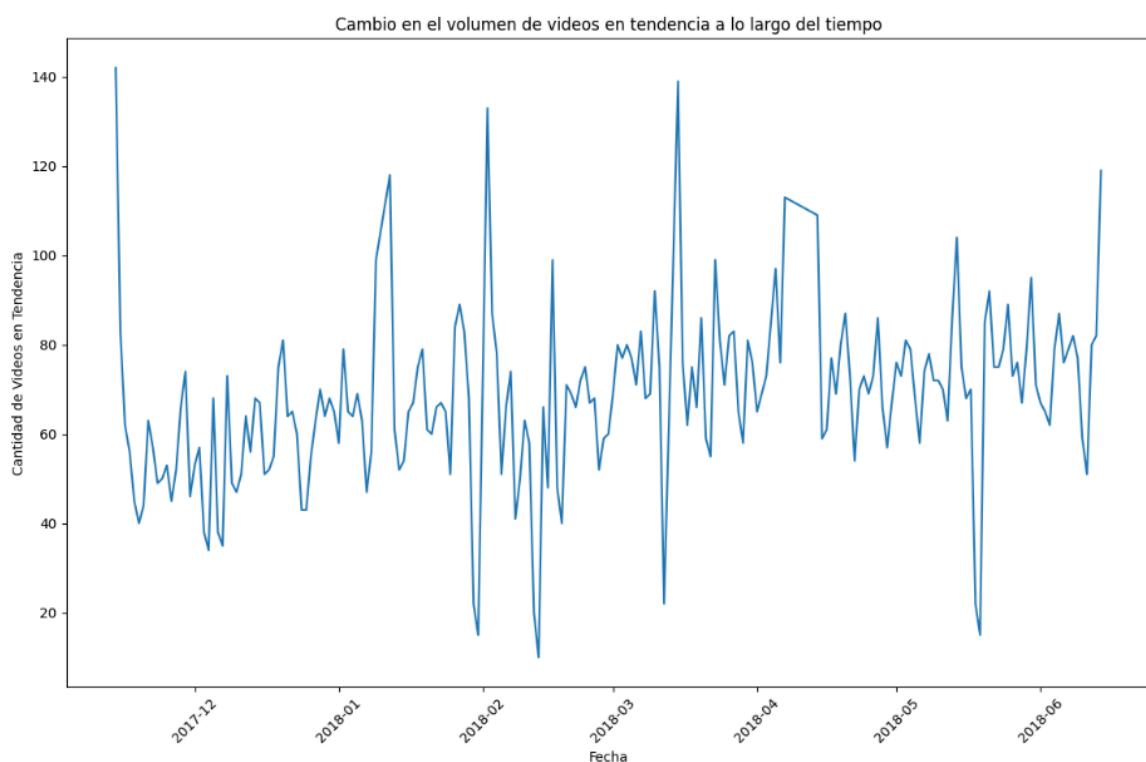
La categoría de “Gaming” tiene la mejor proporción (Mayor a 35) de “Me gusta” y “No Me gusta” en la plataforma, siguiendo las categorías de Autos y Vehículos, y Comedia

¿Qué categorías de videos tienen la mejor proporción (ratio) de “Vistas” / “Comentarios”?



La categoría de “Shows” tiene la mejor proporción (Mayor a 800) de “Vistas” y “Comentarios” en la plataforma, luego viene deportes pero con una menor proporción de aproximadamente 400 y Películas y Animaciones con una proporción menor a 400.

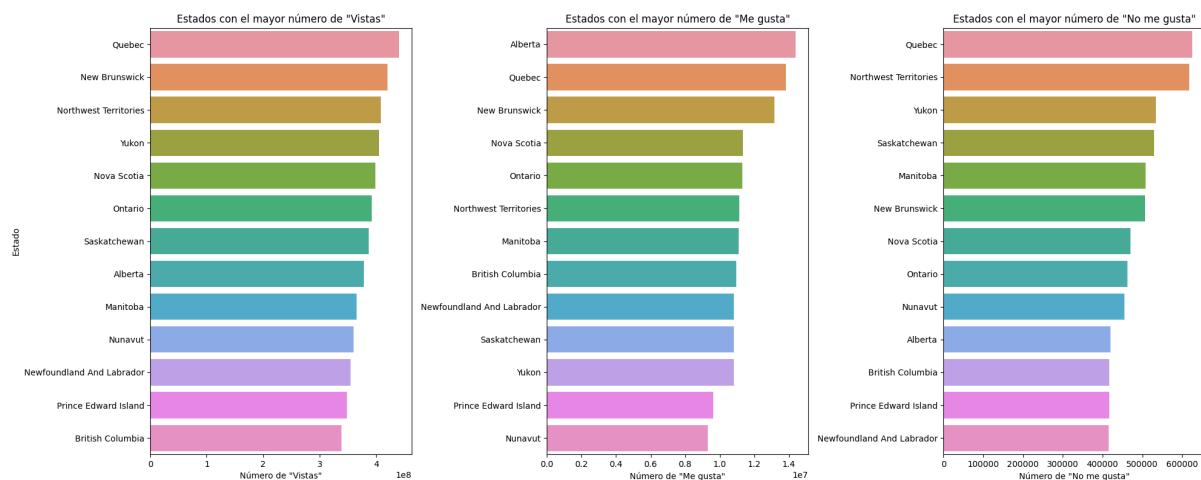
¿Cómo ha cambiado el volumen de los videos en tendencia a lo largo del tiempo?



Podemos notar que el volumen de videos en tendencia varía con el tiempo, lo más resaltante es lo siguiente:

- Hay picos de videos en tendencia en los meses de marzo y junio del 2018.
- El volumen mínimo se presenta en abril del 2018 aproximadamente.

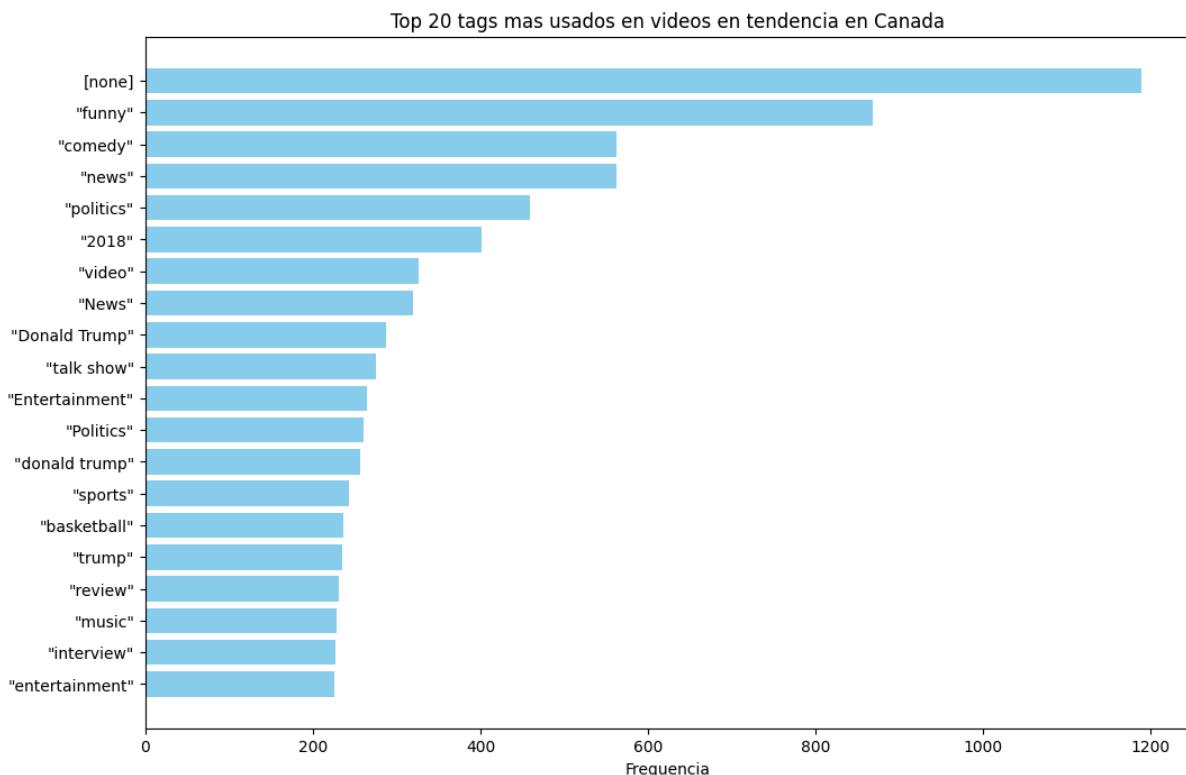
¿En qué Estados se presenta el mayor número de “Vistas”, “Me gusta” y “No me gusta”?



De los gráficos podemos obtener lo siguiente:

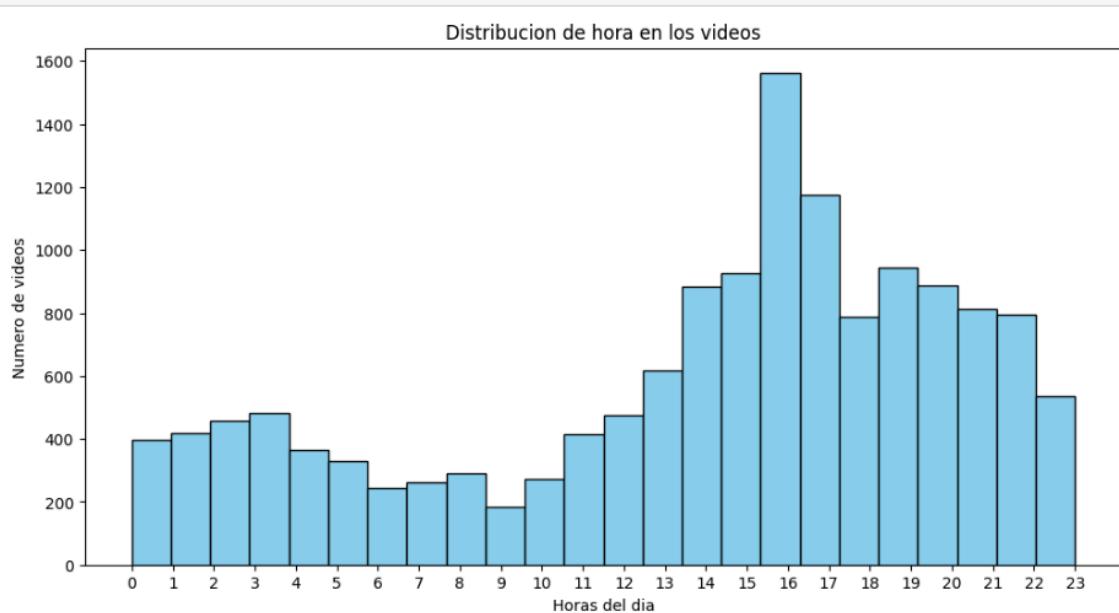
- Los tres estados con mayor número de vistas son Quebec, New Brunswick y Northwest.
- Los tres estados con el mayor número de me gustas son Alberta, Quebec y New Brunswick.
- Los tres estados con el mayor número de no me gusta son Quebec, Northwest Territories y Yukon.

¿Cuáles son los tags más usados en los videos?



De los gráficos podemos obtener lo siguiente:

- Los tres estados con mayor número de vistas son Quebec, New Brunswick y Northwest Territories.
- Los tres estados con el mayor número de me gustas son Alberta, Quebec y New Brunswick
- Los tres estados con el mayor número de no me gusta son Quebec, NortWest Territories y Yukon.

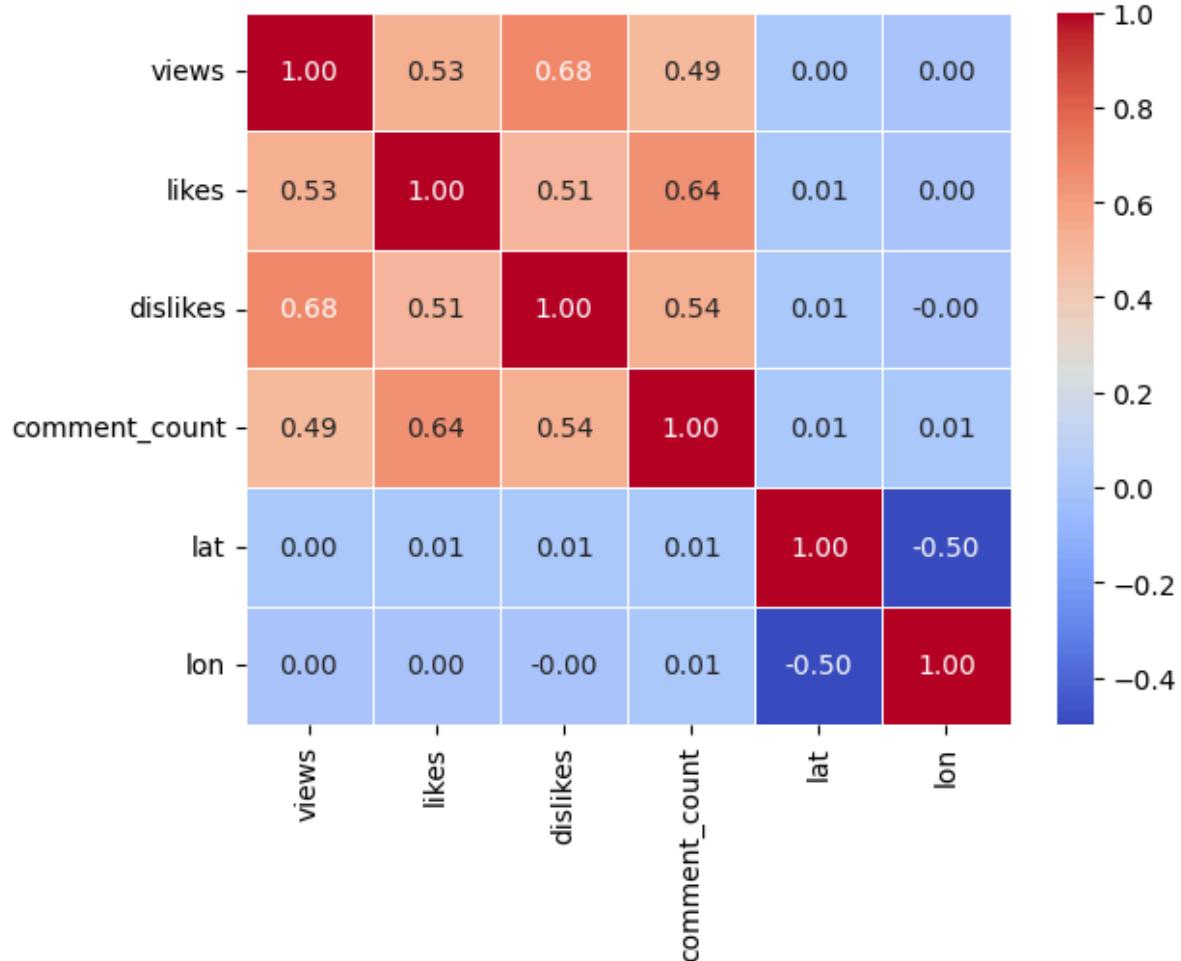


La mayoría de los vídeos en tendencia se publicaron entre las 3 y 4 de la tarde aproximadamente. Por otro lado, en un intervalo de las 0 a 10 am, hubo bajas publicaciones.

Link del GitHub: <https://github.com/mauriciocastellon/FDS-2024-1-CC51>

❖ Transformación de datos

TRANSFORMACIÓN DE



```
[ ] videos_df = data.copy()
columnas_relevantes = ['likes', 'dislikes', 'comment_count', 'views']
videos_df = videos_df[columnas_relevantes]

#para categoricas
#categorica_columns = ['category_id', 'state', 'channel_title']
#videos_df = pd.get_dummies(videos_df, columns=categorica_columns, drop_first=True)
```

```
[ ] X = videos_df.drop('views', axis=1)
y = videos_df['views']
```

```
[ ] #numericas
scaler = StandardScaler()
X = scaler.fit_transform(X)
```

Observamos que las variables que tienen más correlación con las vistas son los likes, dislikes y cantidad de comentarios. Por ello, planteamos que estas variables sean nuestras

variables independientes, mientras que la dependiente es las vistas de cada video. Luego de ello lo estandarizamos.

4. MODELAR Y EVALUAR LOS RESULTADOS

- En esta fase se busca establecer las técnicas de modelado más apropiadas para el proyecto de Data Mining específico de acuerdo con los objetivos planteados.

Escoger la técnica de modelado

En esta fase se busca establecer las técnicas de modelado más apropiadas para el proyecto de Data Mining específico de acuerdo con los objetivos planteados. La selección de la técnica de Data Mining depende en gran medida del tipo de problema que se desea resolver. Para nuestro caso, es un problema de predicción de una variable, por lo que se puede usar técnicas de regresión lineal para problemas continuos o la regresión logística para problemas categóricos.

En nuestro caso, al ser una variable numérica la que se quiere predecir, hemos optado por elaborar un modelo de regresión lineal múltiple con la finalidad de predecir la cantidad de vistas de los videos de Youtube.

Generar el plan de prueba

Se decidió dividir los datos en dos conjuntos: uno de entrenamiento y otro de prueba. El conjunto de entrenamiento se utilizará para ajustar el modelo, mientras que el conjunto de prueba se utilizará para evaluar su rendimiento. En nuestro caso, se optó por hacer que el 80% sea para entrenar el modelo y el 20% sea para su evaluación. Todo esto con la ayuda de la librería en Python de Sklearn.

Construir el modelo

A continuación, se ejecuta la técnica seleccionada sobre los datos previamente preparados para generar uno o más modelos.

```
: X = videos_df.drop('views', axis=1)
y = videos_df['views']

:#numericas
scaler = StandardScaler()
X = scaler.fit_transform(X)

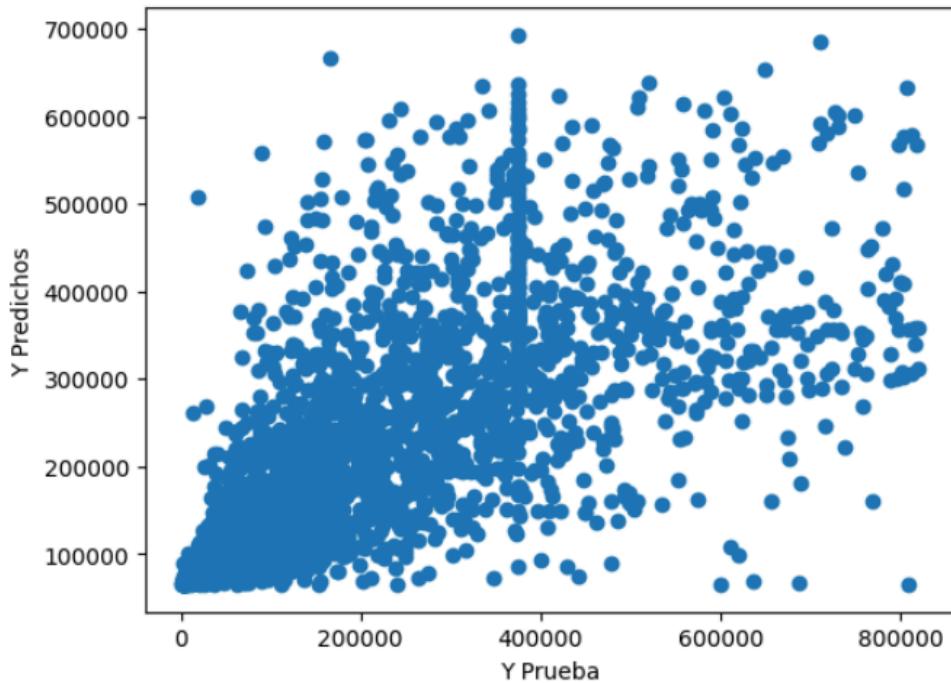
:x_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

: model = LinearRegression()
model.fit(X_train, y_train)

: LinearRegression
LinearRegression()
```

Evaluar el modelo

Finalmente, se aplican métricas que permiten determinar el rendimiento del modelo en cuanto a la precisión y/o exactitud de los resultados obtenidos. En el caso de la regresión, métricas como el error cuadrático medio (MSE).



```
: print('MAE:', metrics.mean_absolute_error(y_test, y_pred))
print('MSE:', metrics.mean_squared_error(y_test, y_pred))
print('RMSE:', np.sqrt(metrics.
```

```
mean_squared_error(y_test, y_pred)))
```

```
MAE: 91395.61501541096
```

```
MSE: 17026799489.509556
```

```
RMSE: 130486.77898357961
```

La evaluación del modelo de regresión lineal muestra que, aunque se ha realizado un esfuerzo considerable en la limpieza de datos y en la eliminación de outliers, las métricas de error (MAE, MSE y RMSE) indican que todavía existen errores significativos en las predicciones. El error cuadrático medio (MSE) es particularmente alto, lo que sugiere que hay grandes desviaciones en algunas predicciones.

5. CONCLUSIONES

Realizamos un análisis de los videos en tendencia en Canadá y aquí resumimos los insights que sacamos: La mayoría de los videos pertenecen a la categoría de Entretenimiento y justamente esta categoría es la que mayor cantidad de vistas tienen y mayor me gusta. Por lo cual sugerimos a la empresa de marketing digital que se centren y le den una mayor importancia a este. Por otro lado, la categoría música le

hace comparación a la categoría Entretenimiento en cuanto a vistas. Además de ello, es la categoría que más me gusta tienen Si hablamos de proporciones, el gaming supera a otras categorías en cuanto a la proporción de Me gusta/ No me gusta. Y en cuanto a la proporción Vistas/ Comentarios, los shows ocupan el primer y segundo puesto respectivamente. Los videos en tendencia se publican más frecuentemente en la tarde, especialmente a las 3pm y 4pm, lo que sugiere que los creadores de contenido pueden beneficiarse al publicar en estos horarios.

Se cumplieron todos los requisitos establecidos para este proyecto. Primero, se preprocesaron y transformaron los datos de videos de YouTube de manera efectiva, aplicando técnicas de normalización y codificación de variables categóricas sin eliminar categorías. Se aseguraron de que las variables relevantes fueran seleccionadas y las irrelevantes fueran eliminadas, garantizando que el conjunto de datos resultante fuera apropiado para el análisis y el modelado.

Posteriormente, se construyó y evaluó un modelo de regresión lineal para predecir el número de vistas (views). El modelo fue entrenado y evaluado utilizando un conjunto de datos dividido en entrenamiento y prueba, y se calculó el error cuadrático medio (MSE) para medir su precisión. Sin embargo, el modelo no fue el esperado, pues tuvo un error muy alto. Consideramos que esto fue debido a los valores atípicos muy altos que hubieron en el dataset y como sugerencia, planteamos también crear nuevas columnas que considerar en las variables independientes.

6. BIBLIOGRAFÍA

- Kaggle. (2019). [Trending YouTube Video Statistics](https://www.kaggle.com/datasets/datasnaek/youtube-new) [Conjunto de datos]. <https://www.kaggle.com/datasets/datasnaek/youtube-new>
- GitHub. (s.f.). Documentation. Recuperado de <https://docs.github.com/en>
- Link del GitHub: <https://github.com/mauriciocastellon/FDS-2024-1-CC51>