UNIVERSIDAD PERUANA DE CIENCIAS APLICADAS

TÍTULO:

“TRABAJO FINAL: ANÁLISIS DE DATOS DE LAS TENDENCIAS DE VIDEOS EN YOUTUBE”

CURSO:

GESTIÓN DE DATOS

INTEGRANTES:

Caparachin Villaverde, Nanto Gustavo (E202110812)
García Godos Villavicencio, Jorge Daniel (E202110981)
Huamani Franco, Ismael (E202110991)
Poma Ludeña, Rodrigo Renato (E202111018)

Lima, abril del 2022

Índice de Contenido

Índice de Ilustraciones

Índice de Tablas

Introducción

El presente trabajo utiliza información pública recolectada de una de las plataformas más grandes de STREAMING de videos del mundo: YOUTUBE. Posteriormente, es usada para un análisis de datos para determinar las características básicas que permiten a un video convertirse en TENDENCIA. Este proceso consta de dos fases, la primera permite almacenar la información en una base de datos, lo que permite centralizar y unificar los datos; el segundo consiste en analizar y representar los datos en información.

La primera etapa se divide en dos grandes procesos. El primer proceso es de **extracción**, el cual utilizo un conjunto de datos del repositorio de datos KAGGLE que contenía los conjuntos de datos YOUTUBE TRENDING VIDEO STADISTICS. Está información muestra los videos en tendencia en la plataforma, con sus respectivas características como: conteo de me gusta, conteo de no me gusta, conteo de comentarios, categoría, enlace de videos, etc. El segundo proceso es de **consolidación y carga**, el cual permitió unificar los conjuntos de datos dispersos (CSV y JSON) y cargarlos a una base da datos NO SQL, denominada MONGO DB; desplegada en la aplicación de la nube ATLAS (servicios alojados en la nube en AWS, AZURE y GOOGLE CLOUD). Está base datos permitió almacenar la información en colecciones, lo cual persiste y da accesibilidad a la data en un solo punto para su posterior análisis. Esta organización de los datos nos permite garantizar la unicidad de fuentes de datos, por lo tanto, es un paso fundamental antes de todo análisis estadístico.

La segunda fase permite el **descubrimiento, el tratamiento, el análisis y la representación** de los datos, el cual nos brinda un panorama de la información más detallada, encontrar y seleccionar las variables con las cuales se puede trabajar, eliminar las que no se utilizarían, convertir y transformar el formato de variables de acuerdo con los requerimientos o necesidades de análisis respecto a otros datos, limpiar la data vacía y proceder con el análisis de la información.

Con todos estos procesos secuenciales se ha podido validar las cualidades de los videos que los convierte en tendencia.

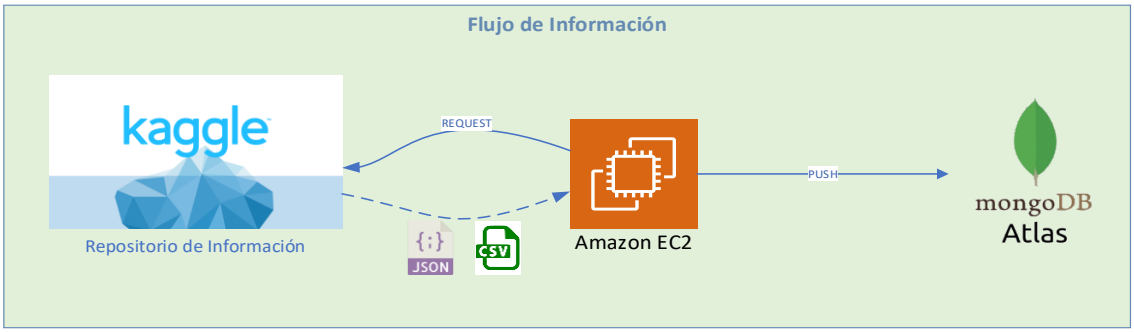

Ilustración 1: Flujo de Información de los Datos para Almacenarlos en una Base de Datos

Ilustración 2: Flujo de Información de los Datos para Analizar y Graficar los Datos

Metodología

A continuación, presentamos la metodología utilizada con los diferentes procedimientos ejecutados, herramientas escogidas y las técnicas utilizadas para la realización de este trabajo.

Procedimiento

El procedimiento realizado nuestro análisis consta de X pasos los cuales describimos a continuación:

1. *Identificación de fuente de datos*

   Identificamos diferentes propuestas de fuentes de datos basándonos en la veracidad, volumen y variedad.

   Este paso tubo como resultado la selección de cuatro propuestas de las cuales se pudo seleccionar el DATASET **"YouTube Trending Video DATASET (updated daily)"** de la plataforma de KAGGLE.

   Dicho DATASET cuenta con información de los videos en tendencia de la plataforma YouTube de cual se puede apreciar información tales como categoría de video, cantidad de visualizaciones, cantidad de me gusta, cantidad de no me gusta, cantidad de comentarios, entre otros.

   Este DATASET contenía los datos en formato JSON y CSV, además esta categorizado por país de los cuales escogimos como foco los países de Estados Unidos, México y Francia.

2. *Dimensionamiento y creación de la base de datos*

   Considerando la cantidad de datos y la estructura de los datos que contenían los tres países previamente escogidos se decidió hacer uso de una base de datos no relacional que permita el escalamiento tanto de transacciones como de espacio para los datos. Para lo cual decidimos utilizar Atlas Mongo DB su versión de plataforma como servicio.

   Para la creación de la base de datos solo fue necesario un correo electrónico y se creó un clúster de Mongo DB con un nodo principal y dos nodos secundarios los cuales distribuyen la carga de transacciones.

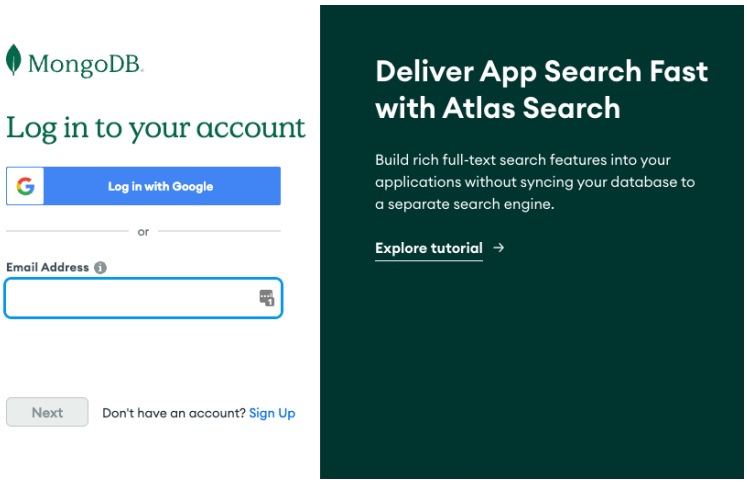

Ilustración 3: Creación de Credenciales para Atlas

Recuerda elegir el proveedor CLOUD de tu clúster e identificar el tamaño más acorde a las necesidades actuales.

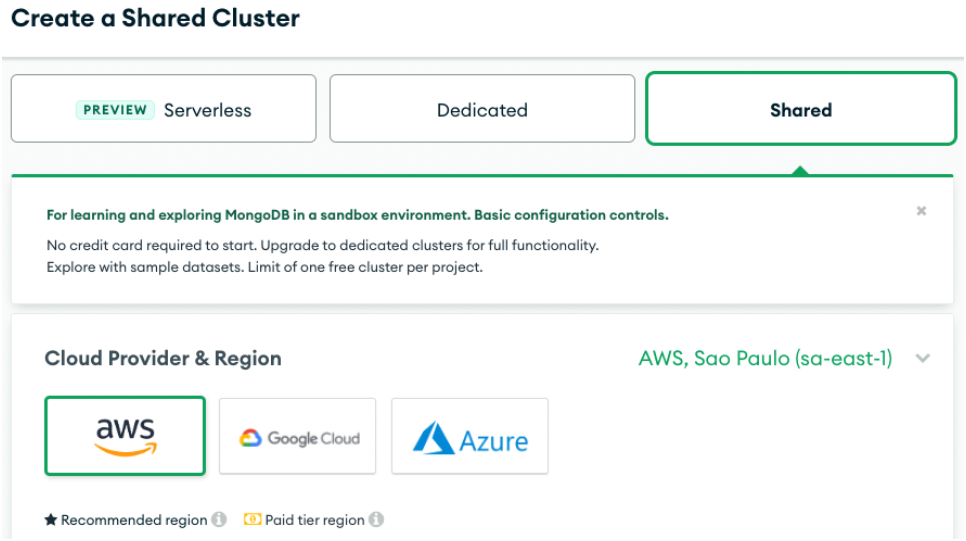

Ilustración 4: Creación de la Base de Datos en MONGODB

Como buena práctica se recomienda utilizar permisos granulares y si son de tipo temporales establecer en ellos una fecha de expiración.

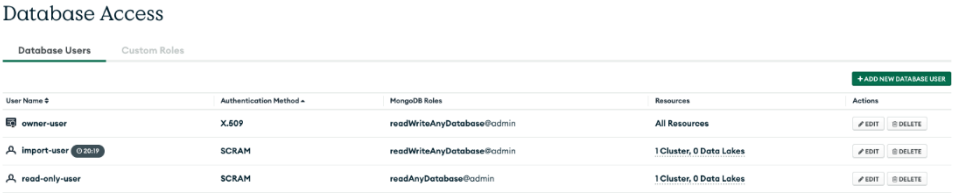

Ilustración 5: Creación de Credenciales de Solo Lectura

Por último, recuerda habilitar el acceso a la base de datos permitiendo que se puedan realizar consultas desde cualquier segmento de red fuera del dominio de la plataforma.

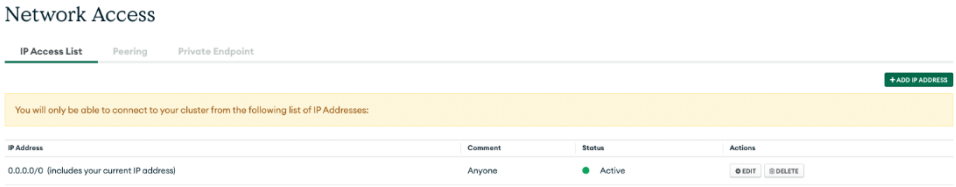

Ilustración 6: Habilitar el Acceso a la Base de Datos

3. *Obtención de datos e importación a base de datos*

La obtención de los datos se hizo uso una instancia EC2 en el proveedor AWS por medio del servicio de CLOUD9 donde instalamos la herramienta "DataBase Tools" el cual permite la subida de archivos de datos y conversión a formatos JSON.

La importación de los diferentes archivos de datos se realizó con mongoimport una función específica de la herramienta "DataBase Tools" en la cual especificamos el formato de los archivos de datos JSON o CSV además de utilizar credenciales específicas que permitan la inserción masiva de datos.

```
mongoimport "mongodb+srv://mycluster.zal73.mongodb.net" --db=dbYouTube --collection=FR_category_id --file=FR_category_id.json --jsonArray --username=import-user --password=<>

mongoimport "mongodb+srv://mycluster.zal73.mongodb.net" --db=dbYouTube --collection=US_YouTube_trending_data --type=csv --headerline --file=US_YouTube_trending_data.csv --username=import-user --password=<>
```

Se estructuró la data en colecciones por países y por tipos de categorías como en la siguiente imagen.

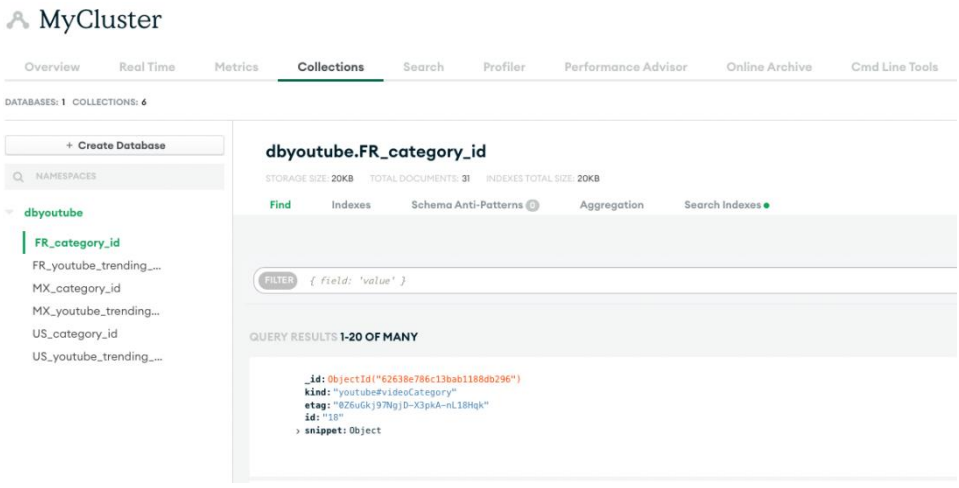

Ilustración 7: Colecciones en la Base de Datos de VIDEOS EN TENDENCIA DE YOUTUBE

Por último, automatizamos este proceso mediante un script BATCH para que cada semana se pueda extraer la información de KAGGLE y se haga una importación a la base de datos.

4. *Conexión a base de datos y extracción de data*

Para establecer la conexión a la base de datos utilizamos la librería 'mongolite' la cual permite realizar diferentes operaciones en la base de datos creada. Se configuró la conexión con un usuario de solo lectura.

```
connection_string = 'mongodb+srv://read-only-
user:U5PN6pQ1eirrSu7e@mycluster.zal73.mongodb.net/dbYouTube'
```

Para la extracción de la información se segmento por colecciones permitiendo tener los datos por países en diferentes variables al igual que las categorías para posteriormente hacer una unificación de los DATAFRAMES.

```
MX_trending = mongo(collection="MX_YouTube_trending_data", db="dbYouTube",
url=connection_string)
US_trending = mongo(collection="US_YouTube_trending_data", db="dbYouTube",
url=connection_string)
FR_trending = mongo(collection="FR_YouTube_trending_data", db="dbYouTube",
url=connection_string)
MX_category = mongo(collection="MX_category_id", db="dbYouTube", url=connection_string)
US_category = mongo(collection="US_category_id", db="dbYouTube", url=connection_string)
FR_category = mongo(collection="FR_category_id", db="dbYouTube", url=connection_string)
```

5. *Comprensión y limpieza de datos*

Preparamos las tablas creando un nuevo campo country que nos permita identificar el origen de los datos México, USA y Francia.

```
MX$country = MX$country = 'Mexico'
US$country = US$country = 'USA'
FR$country = FR$country = 'Francia'
```

Posteriormente eliminamos las columnas que no agregaran valor a nuestro análisis tales como:

- thumbnail_link: Contiene la URL del video
- description: Contiene la descripción del video
- title: Contiene el título del video
- channelId: Contiene el id del canal o dueño del video

Finalmente unificamos los DATASETS para poder realizar nuestro análisis.

```
bind_rows(MX,US,FR)
```

Realizamos una búsqueda de nulos en los diferentes campos, al no contar con nulos no es necesario realizar alguna acción, por el contrario, encontramos datos tipo fecha con diferentes formatos para lo cual estandarizamos el formato de hora y fecha; y por último asignamos factor al campo de país para mejorar la eficiencia de categorización por este campo.

Ahora trabajamos con los datos de categorías los cuales vinieron en colecciones distintas, contamos con muy pocos datos, pero contienen información valiosa

que aporta a nuestro DATASET principal, tampoco hay datos nulos y los formatos están correctos procedemos a desarrollar una función para convertirlos en DATAFRAME antes de unificarlos.

```
for (i in 1:length(cat_MX$id)){
                vector1 = c(vector1, cat_MX$id[i])
                    vector2 = c(vector2, cat_MX$snippet$title[i])
            }

DFX = data.frame("id"=vector1, "title"=vector2)
```

Unificamos los DATASET de categorías separados por país:

```
DFCAT = bind_rows(DFX, DFUS,DFFR)
```

También eliminamos los duplicados que existen de categorías entre los países.

```
DFCATEGORIA = DFCAT[!duplicated(DFCAT), ] #unique(DFCATEGORIA[c("title")])
```

Antes de unificar el DATAFRAME de categorías con nuestro DATAFRAME principal convertimos el ID de correlación en tipo de dato STRING.

```
YTB$categoryId = as.integer(YTB$categoryId)
```

Unificamos las fuentes de datos y aplicamos factor a las categorías en el nuevo DATASET

```
YOUTUBE = merge(x = YTB, y = DFCATEGORIA, all.x = TRUE)

levels(YTB$category) <- c( "Film&Animation", "Autos&Vehicles", "Music", "Pets&Animals",
"Sports", "Travel&Events", "Gaming", "People&Blogs", "Comedy", "Entertainment",
"News&Politics", "Howto&Style", "Education", "Science&Technology", "Nonprofits&Activism",
Movies", "Shows", "Trailers")
```

Finalmente verificamos el DATASET a trabajar.

```
summary(YOUTUBE)
anyNA(YOUTUBE)
colSums(is.na(YOUTUBE))
head(YOUTUBE)
```

Verificamos que los datos tienen el formato correcto, no existen nulos y la data esta correctamente organizada en el DATAFRAME.

6. *Análisis de OUTLIERS*

Creamos diferentes resúmenes de datos que utilizaremos posteriormente

- Resumen del campo de categoría por frecuencia
- Resumen de los canales que más se frecuentan
- Correlación entre ‘me gusta’ y ‘vistas’

Para el análisis de OUTLIERS seleccionamos los campos a trabajar:

- view_count: Contador de vistas de cada video
- likes: Cantidad de me gustas de cada video
- dislikes: Cantidad de no me gusta de cada video
- comment_count: Cantidad de comentarios de cada video

Con los campos a trabajar realizamos una validación de nulos y validación de ceros.

```
sapply(df_YOUTUBE, function(x) sum(is.na(x)))


colSums(df_YOUTUBE == 0)
```

Antes de empezar con nuestro análisis realizamos una normalización y usaremos el criterio de normalización de máximos y mínimos tomando como máximo el valor uno y mínimo el valor cero.

```
minmax <- function(x, vmin, vmax){
                    xmin <- min(x)
                  xmax <- max(x)
                    y <- ((x-xmin)*(vmax-vmin))/(xmax-xmin) + vmin
              return(y)
          }

min_val <- 0
max_val <- 1


df_YOUTUBE$view_countmm <- minmax(df_YOUTUBE$view_count, min_val, max_val)
df_YOUTUBE$likesmm <- minmax(df_YOUTUBE$likes, min_val, max_val)
df_YOUTUBE$dislikesmm <- minmax(df_YOUTUBE$dislikes, min_val, max_val)
df_YOUTUBE$comment_countmm <- minmax(df_YOUTUBE$comment_count, min_val,
max_val)
```

Aplicamos la técnica SHAPIRO con 5000 registros su límite para identificar si existe una distribución normal.

```
distr_shapiro = list()
for (d_col in distr_cols){
  distr_shapiro[[d_col]] <- shapiro.test(df_YOUTUBE[, d_col][0:5000])$p.value
}

distr_shapiro_val <- distr_shapiro > 0.05
```

Aplicamos la técnica Anderson-Darling para validar más de 5000 registros para identificar si existe una distribución normal

```
distr_anderson = list()
for (d_col in distr_cols){
  distr_anderson[[d_col]] <- ad.test(df_YOUTUBE[, d_col])$p.value
}
distr_anderson <- distr_anderson > 0.05
```

Posteriormente identificamos si los valores atípicos para lo cual aplicamos la técnica de distancia de MAHALANOBIS para realizar un análisis multivariable.

```
df_YOUTUBE <-
df_YOUTUBE[c("view_countmm","likesmm","dislikesmm","comment_countmm")]

mu <- colMeans(df_YOUTUBE)
S <- cov(df_YOUTUBE)

dm2 <- MAHALANOBIS(df_YOUTUBE, mu, S)
```

Por último, realizamos la distribución Chi-Cuadrado al 90% y 99.9% identificando en los índices de los valores atípicos.

```
p1 <- 1-0.1
k1 <- qchisq(p1, dof)
idx_outliers1 <- which(dm2>k1)
idx_outliers1
dm2[idx_outliers1]


p2 <- 1-0.001
k2 <- qchisq(p2, dof)
idx_outliers2 <- which(dm2>k2)
idx_outliers2
abline (h = k2, col = 'green', lwd = 1)
```

7. *Exploración de datos en SHINY*

Para iniciar la exploración de datos en SHINY iniciamos creando un proyecto tipo SHINY como mono archivo para no tener inconvenientes al subirlo a SHINY APPS.

Añadimos los siguientes campos en el componente UI:

- Campo para el análisis de visualización de las tendencias
- Campo para el análisis de visualización de los canales más vistos
- Creación del campo para el análisis de correlación entre LIKES y vistas

Finalmente creamos el componente servidor y añadimos los resultados al graficarlos:

- Gráfico de barras horizontales de frecuencia de categoría, interactivo con su frecuencia
- Gráfico de barras verticales de la frecuencia de canales, interactivo con su frecuencia
- Gráfico de correlación interactivo en función a la categoría
- Gráfico de barras de MAHALANOBIS
- Gráfico de ojiva para la probabilidad acumulada
- Gráfico cuantil-cuantil

Por último, ejecutamos la función SHINYAPP para lanzar la ejecución de la web.

Herramientas

Tabla 1: Herramientas Utilizadas en el Flujo de Datos

HERRAMIENTA	DETALLE	UTILIZACIÓN
KAGGLE	KAGGLE es un repositorio de DATASETS abiertos.	Obtención de nuestro DATASET actualizado de tendencias de videos.
Amazon CLOUD9	IDE de desarrollo de software ejecutado en EC2.	Manipulación del comprimido del DATASET e importación a la base de datos.
Atlas MongoDB	Base de datos no relacional administrada por Atlas.	Base de datos de consulta de DATASET.

DataBase Tools	Librería de manipulación de datos	Uso de la función mongo-import para la importación del DATASET.
RSTUDIO CLOUD	IDE CLOUD de desarrollo de proyectos en lenguaje R.	Desarrollo de análisis de datos.
Amazon EC2	Máquina virtual multipropósito.	Ejecución de consultas con alto consumo de datos y procesamiento.

Técnicas Utilizadas

Tabla 2: Técnicas Utilizadas en el Flujo de Datos

TÉCNICA	ETAPA DE UTILIZACIÓN
Eliminación de duplicados	Comprensión y limpieza de datos
Aplicación de Factor o LEVELS	Comprensión y limpieza de datos
Validación de Nulos	Comprensión y limpieza de datos, Análisis de OUTLIERS
Validación de Ceros	Análisis de OUTLIERS
Normalización usando criterio Min-Max	Análisis de OUTLIERS
Shapiro	Análisis de OUTLIERS
Anderson-Darling	Análisis de OUTLIERS
Distancia de MAHALANOBIS	Análisis de OUTLIERS, Exploración de datos en SHINY
Chi cuadrado	Análisis de OUTLIERS, Exploración de datos en SHINY
Análisis de frecuencia	Exploración de datos en SHINY
Análisis de dispersión	Exploración de datos en SHINY

Resultados

Resultados de análisis de dispersión:

Resultado 1: Inicialmente utilizamos 4 variables para el análisis: LIKES, VIEWS y COMENTS y DISLIKES hemos encontrado que la distribución de las variables de LIKES, VIEW y COMENTS, tienen una distribución claramente asimétrica positiva.

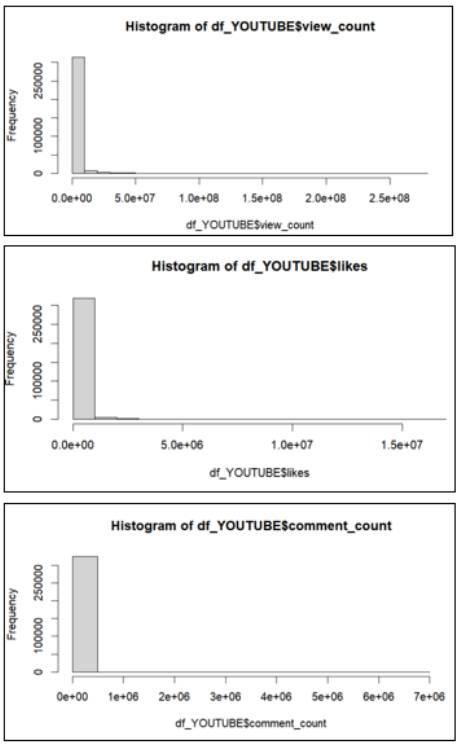

Ilustración 8: Histograma de las Variables LIKES, VIEWS y COMENTS de la Fuente de Información

Resultado 2: Al ser los rangos en los cuales se distribuye cada variable son muy diferentes por lo que se procedió a normalizarlos, tanto por el método de Puntaje Z, para que conserve la distribución de los resultados, como MINMAX.

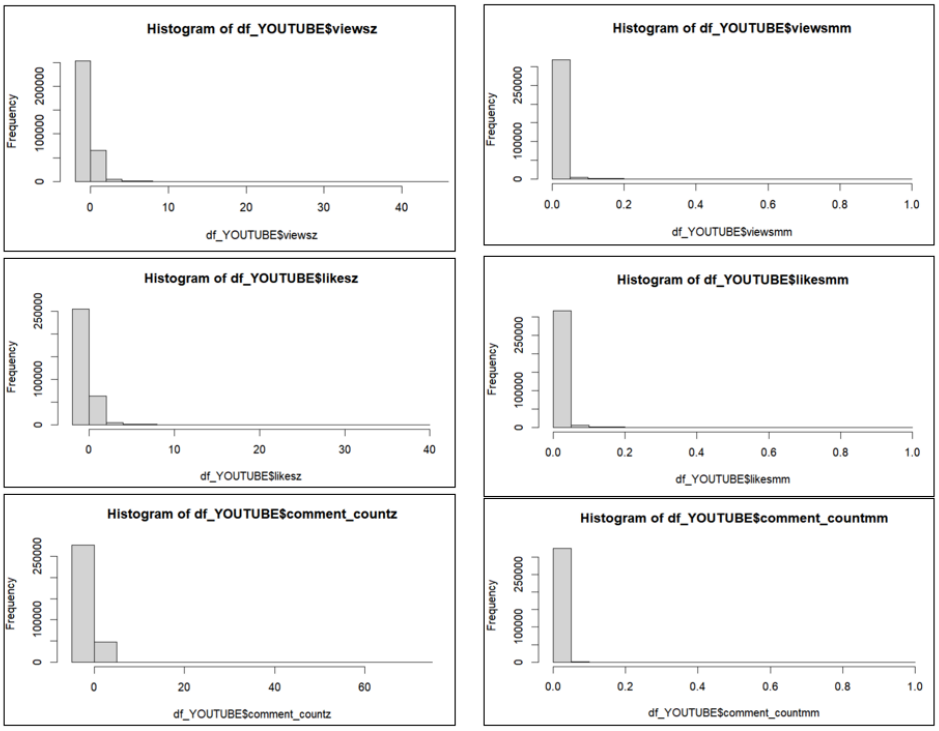

Ilustración 9: Histogramas de las Variables Normalizadas con Puntaje Z

Resultado 3: Hemos logrado identificar aquellos valores que en su combinación de sus variables views, likes y coments presentan un comportamiento muy diferenciado al resto, logrando valores lejos del centro alcanzando un valor de 6,241.

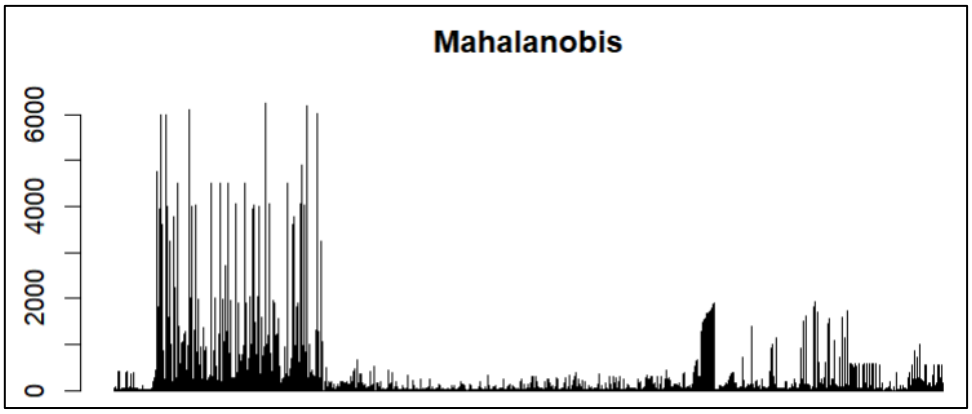

Ilustración 10: Gráfico de MAHALANOBIS de las Variables

```
dmX[order(dmX, decreasing=TRUE)]      # Valores ordenados
  [1] 6241.9481 6174.6837 6087.5568 6026.6682 5974.7298 5974.6639 4904.8350 4755.1686 4501.4315
 [10] 4501.3477 4501.2585 4491.3957 4491.3852 4491.3624 4052.6728 4052.6044 4052.5759 4025.7232
 [19] 4025.5937 4025.4940 3992.9173 3992.8885 3992.8532 3941.1282 3941.1132 3941.0153 3762.3705
 [28] 3762.3158 3762.1394 3609.6299 3609.5634 3609.4146 3248.5931 3248.5884 2706.3610 2467.6685
 [37] 2223.4181 2037.2953 2037.2560 2014.1696 2014.1562 1978.9732 1978.9282 1978.9125 1943.1308
 [46] 1942.9957 1942.9774 1914.9581 1904.5193 1891.6663 1891.4509 1891.4359 1890.0737 1866.0360
 [55] 1847.0950 1826.4574 1826.4227 1826.1002 1824.4694 1798.9475 1779.3315 1763.5663 1745.0841
 [64] 1726.4912 1711.3655 1697.1365 1681.4905 1664.1336 1648.6631 1631.2750 1614.1504 1598.1326
 [73] 1591.0377 1587.9573 1587.6974 1576.8235 1562.7157 1562.6217 1562.5767 1551.9625 1528.3684
 [82] 1509.1910 1486.9190 1469.3362 1445.6227 1441.4363 1413.5958 1385.7964 1382.8505 1357.9724
 [91] 1325.1695 1325.1692 1325.1678 1285.3492 1283.6251 1283.5290 1283.4736 1239.2959 1239.1103
[100] 1239.1102 1193.2691 1189.7019 1189.4728 1189.4653 1147.2802 1143.1372 1110.1379 1110.0581
[109] 1109.9607 1091.7923 1061.2180 1059.8396 1059.8387 1048.7490 1048.7276 1048.6966 1040.8313
[118] 1018.6890 1018.6779 1017.6260 1011.6857 1011.6604 1011.5700 1007.6972 1007.6944 1002.5069
[127]  999.2607  999.2480  999.2472  999.2675  999.2003  999.6182  979.0240  979.0205  978.9775
```

Ilustración 11: Listado de Valores Después de Aplicar MAHALANOBIS

Resultado 3: El umbral encontrado a través de chi-cuadro es de 16.26. Los datos que superan ese valor de dispersión son 5,532; lo cual representa el 1,70%.

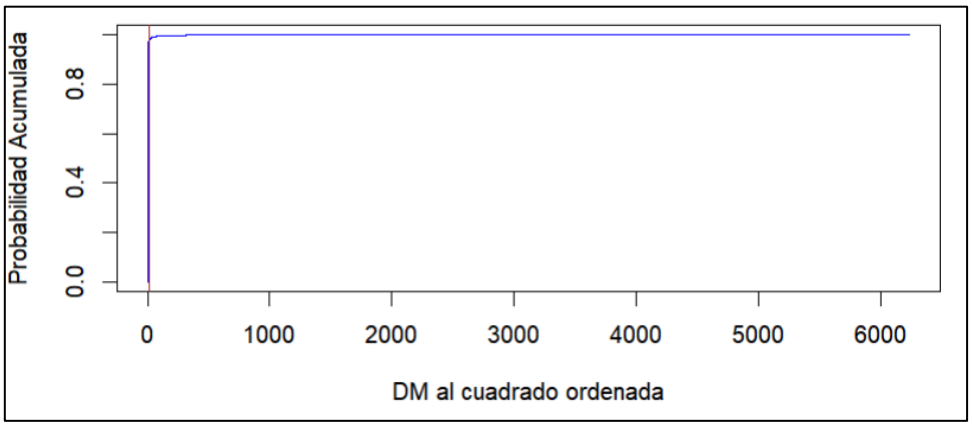

Ilustración 12: Gráfico de la Probabilidad Acumulada al 99.9%

Resultado 4: Sin embargo, si la evaluación es mediante la gráfica de cuantil – cuantil, observamos una mayor dispersión en relación al valor de chi-cuadro . Los resultados se presentan a continuación:

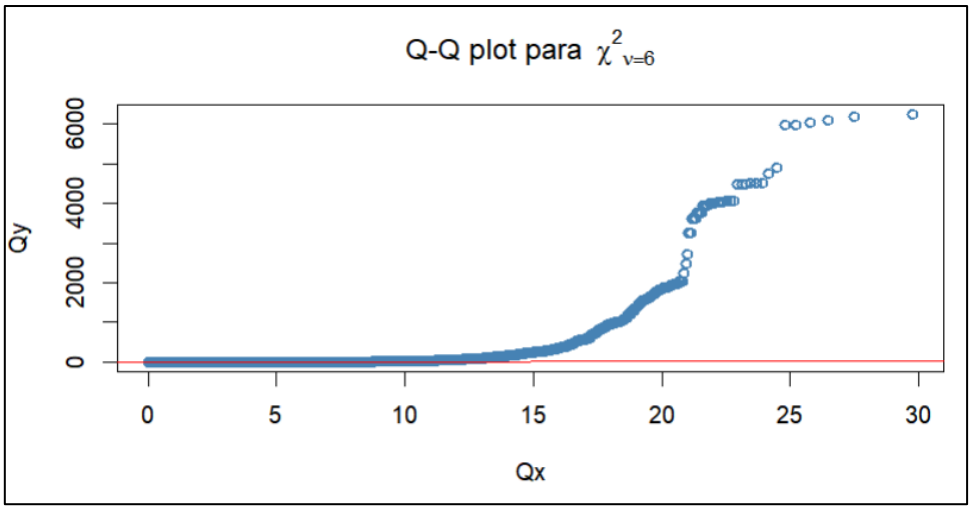

Ilustración 13: Gráfico de QQ-PLOT para el Análisis de Cuartiles

Resultado 5: Encontramos en el análisis una mayor inclinación por la generación de videos en las categorías de Entretenimiento y Música, del otro lado del análisis se encuentran Vehículos, Educación, y Ciencia y tecnología.

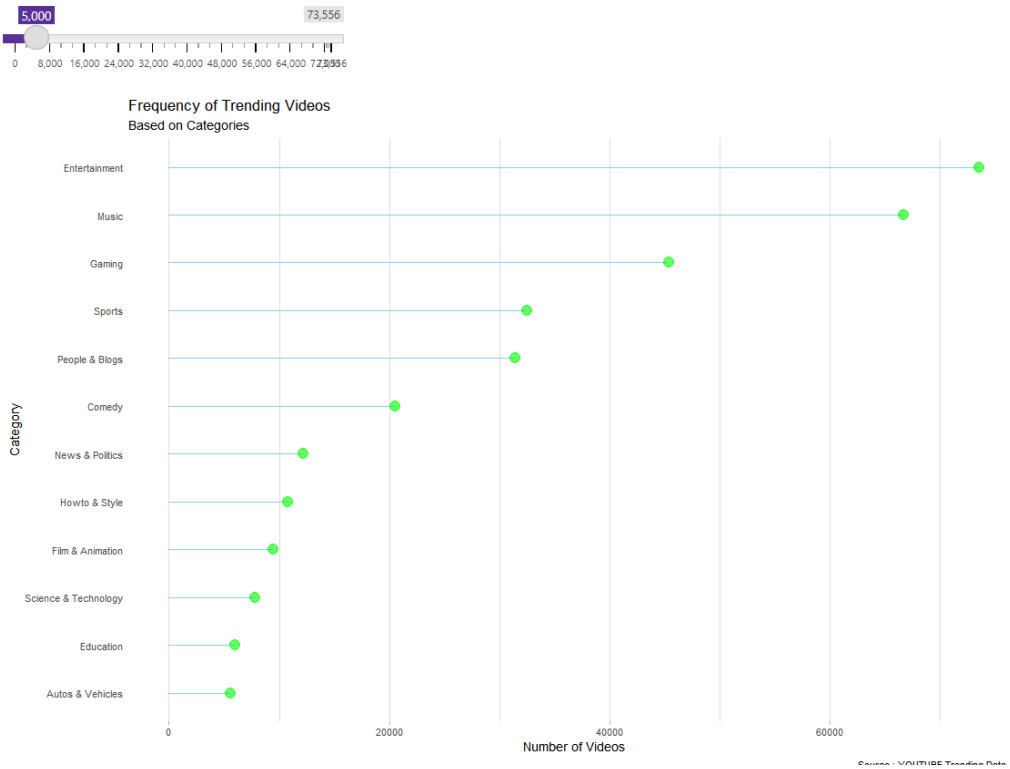

Ilustración 14: Frecuencia de Videos en Tendencia según Categoría

Resultado 6: Es interesante, encontrar que entre los canales que más contenido sube, está en primer lugar “Pongámoslo a prueba cortos”. Lo que sorprende es un canal chino que hace contenido para cada uno de los países, posicionándose como uno de los que más generen en cada uno de ello.

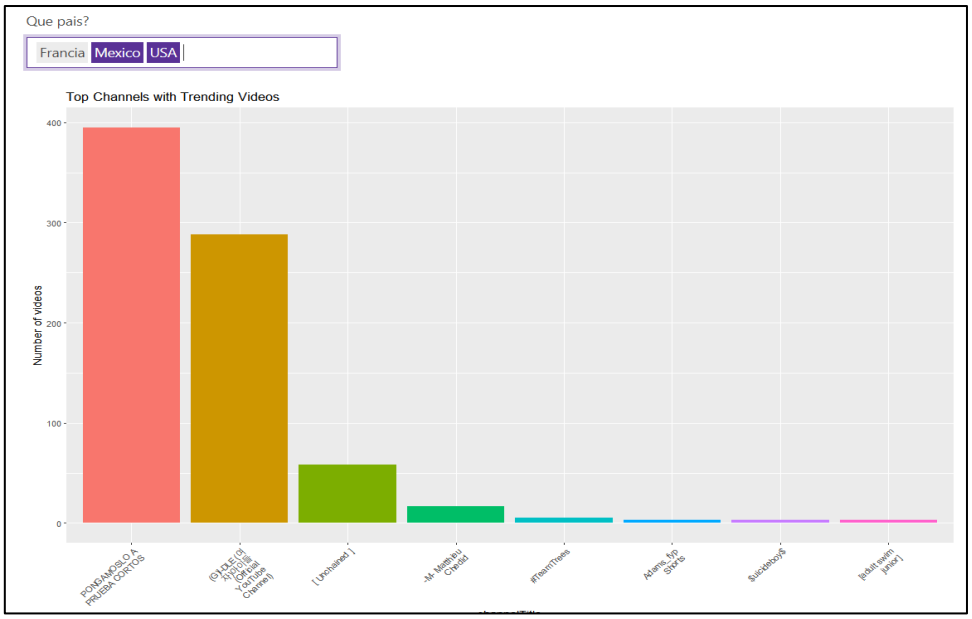

Ilustración 15: Frecuencia de Canales con los Videos en Tendencia

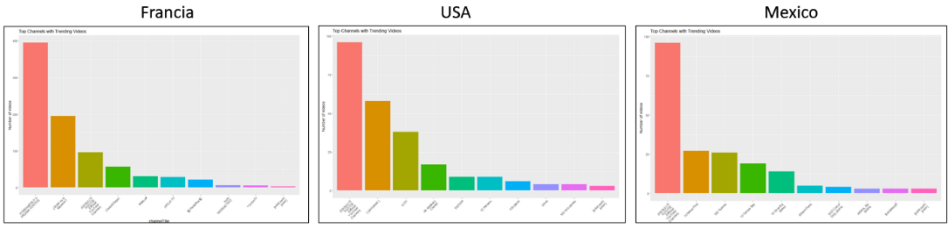

Ilustración 16: Frecuencia de Canales con los Videos en Tendencia según País de Procedencia

Resultado 7: Por último, hicimos el ejercicio para evaluar la correlación entre el número de reproducciones y LIKES. De primera vista, vemos que las categorías que podríamos llamar divertidas, como “Música” y “Comedy”, tienen una distribución casi homogénea en estas dos variables. No pasa de forma tan notoria en “Educación” y “News Politycs”. Menos aún en “Ciencias y Tecnología”, y “Sin fines de lucro y activismo”. Probablemente, estos resultados pueden ayudar a ahondar en la forma psicología del comportamiento de las personas. Que si bien en general, todos tienen una misma dirección en cosas no serias, en temas que son más serios, existe ligera diferencia en la opinión de cada persona

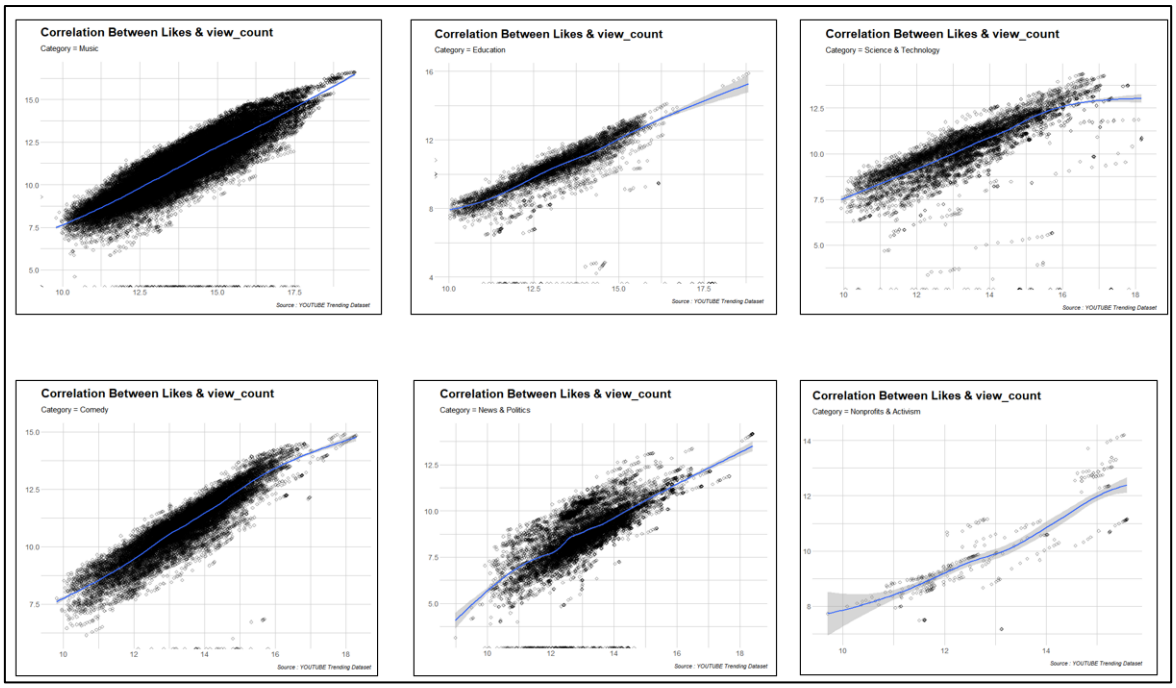

Ilustración 17: Gráfico de Dispersión para Determinar la Correlación de Variables

Conclusiones

- La herramienta de MongoDB nos permitió extraer los datos de manera segura y organizada utilizando usuarios de solo lectura con la finalidad de la inmutabilidad de la base de datos.
- La importación de los datos debe realizarse en una instancia virtualizada que facilite el traslado de altos volúmenes de datos sin interrupción y con la posibilidad.
- Hemos constatado la capacidad de R-Studio sobre su capacidad para la conexión a una base de datos como servicio, como es MongoDB, la misma que después de la exploración y procesamiento de la información también ayudo a su presentación de información mediante Shiny Apps. de programarse como una tarea repetitiva.
- Hemos encontrado también una limitación en la conexión a la base de datos como servicio, cuando se trata de la extracción de datos pesados, ya que en oportunidades la carga exigió bastante recurso computacional. Revisar la cantidad de conexiones máximas hacia la BD, es necesario.
- Hemos encontrado que un canal chino genera una cantidad significativa de videos en diferentes países tales como USA, México y Francia. De forma muy diferente al comportamiento de otros canales que tienen mayor presencia en ciertos países.
- Menos del 0.01% podemos decir que son videos que han logrado una variación representativa al resto. Si lo vemos en números, existe una probabilidad de 0.01%. de que si generas un video alcance un gran éxito.

Bibliografía

- A. N. PETTITT, A two-sample Anderson-Darling rank statistic, Biometrika, Volume 63, Issue 1, 1976, Pages 161–168, https://doi.org/10.1093/biomet/63.1.161
- Emad-Eldin, A. A., & Öztürk, A. (1988). A modified one-sample QQ plot and a test for normality. Journal of Statistical Computation and Simulation, 29(1), 1-15.
- G. Fasano, A. Franceschini, A multidimensional version of the Kolmogorov–Smirnov test, Monthly Notices of the Royal Astronomical Society, Volume 225, Issue 1, March 1987, Pages 155–170, https://doi.org/10.1093/mnras/225.1.155
- Hanusz, Z., & Tarasińska, J. (2015). Normalization of the Kolmogorov–Smirnov and Shapiro–Wilk tests of normality. Biometrical Letters, 52(2), 85-93.
- Lee, R., Qian, M., & Shao, Y. (2014). On rotational robustness of Shapiro-Wilk type tests for multivariate normality. Open Journal of Statistics, 4(11), 964.
- Peter A. W. Lewis. (1961). Distribution of the Anderson-Darling Statistic. The Annals of Mathematical Statistics, 32(4), 1118–1124. http://www.jstor.org/stable/2237910
- Ronald L. Iman (1982) Graphs for use with the Lilliefors Test for Normal and Exponential Distributions, The American Statistician, 36:2, 109-112, DOI: 10.1080/00031305.1982.10482799
- Walfish, S. (2006). A review of statistical outlier methods. Pharmaceutical technology, 30(11), 82.

Anexos

Anexo 1: Repositorio de GITHUB

https://github.com/RodrigoRenatoPomaLudena/TrendingYouTubeVideoStatistics

Anexo 2: Gráficos Públicos en SHINY APPS

https://e202111018.shinyapps.io/TF_GestionDatos/