YouTube Trending Video Analysis

by

Wenhui Fang wf2282

Jianxun Guo jg4427

Yixin Ji yj2666

Jingyou Jiang jj3192

Zhanbei Liu zl3070

Anran Yang ay2535

Department Of Statistics, Columbia University

GR 5291: Advanced Data Analysis

Prof. David Rios

Spring 2022

Content

I. Introduction

YouTube is one of the most popular video-sharing websites that allows uploading videos and subscribing to others around the world. Users can share their lives and view content uploaded by YouTubers around the world. It offers a wide variety of user-generated and corporate media videos. Available content includes entertainment videos, TV show clips, music videos, educational videos… and many genres you can think of. YouTube is the largest online video sharing and social media platform in the world. According to Statista 2022, over 2.6 billion people worldwide use YouTube once a month. YouTube's global advertising revenues in 2021 were estimated to be around $28.84 billion. The United States has over 240 million users with 62% of users accessing the platform on a daily basis.[1]

There are always some videos that catch massive attention in a short period of time and become trending. The dataset records top trending videos on YouTube and is updated daily. The rule of determining whether a video is trending not only depends on the view counts but also on the speed of increasing views, uploaded time of the video, user feedback, et cetera. Trending videos are important information for both the Youtube company and its users, the Youtubers. Youtube can monitor user engagement and usage through the view of trending videos to visualize its company development trend. When firms increase users' engagement and build an environment that helps to foster engagement, they can significantly increase the chances of success in their business (Kim et al. 361)[2]. Thus, the total view of trending videos daily will reflect the fluctuation of user engagement and the future development of Youtube in the industry.

[1] "YouTube Statistics 2022." Official GMI Blog, https://www.globalmediainsight.com/blog/youtube-users-statistics/.
[2] Kim, Young Hoon, Dan J. Kim, and Kathy Wachter. "A study of mobile user engagement (MoEN): Engagement motivations, perceived value, satisfaction, and continued engagement intention." Decision support systems 56 (2013): 361-370.

Users who upload videos can refer to the list of trending videos to improve their content and make their videos go viral. Many YouTubers upload their gaming videos, edited vlogs, and self-created shows to build their unique channels. They utilize Youtube as one of their social media platforms to spread their interesting and insightful content to their audiences and subscribers in the form of all kinds of videos. Sharing everyday life experiences on social media enables feelings of belonging and creates a sense of online community (McCay-Peet)[3]. To acquire a sense of belonging on the Youtube platform, video creators would love to discover the secret techniques to make their videos go trending so that they can have more subscribers and audiences to watch their high-quality content.

Data Preprocessing

The Youtube Trending dataset originates from the Kaggle dataset, which is composed of detailed information on daily trending videos from August 2020 to March 2022. In this project, we mainly analyze the trending video in the United States region. Since some videos are trending for more than one day, they may appear more than one time in this dataset with a unique video ID. Hence, we change the date to the date-time format so that it is more convenient to analyze and visualize data. After that, we eliminate the missing values, delete unusual cases that have more likes than view counts, and prepare two versions of data as validation to proceed with our analysis.

[3] McCay-Peet, Lori, and Anabel Quan-Haase. "A model of social media engagement: User profiles, gratifications, and experiences." Why engagement matters. Springer, Cham, 2016. 199-217.

II. Objectives

Trending videos varied every day on different factors including category, view counts, likes, dislikes, and comments. Therefore, in this project, by using Python and R with relevant packages on this dataset, we are able to perform numerous statistical models, methods, and other analysis techniques including Time Series, Multi-Linear models, and Machine Learning to analyze and predict different trending video features. The analysis of this project will suggest possible future development of both the Youtube company and Youtubers to create a constructive video-making and video-sharing platform.

For Youtube:

1. Predict total trending video views daily to monitor company development trends.
2. Simulate the classification algorithm of the Youtube videos category to target merchants' potential clients on Youtube and advertise relevant products.

For Youtubers:

3. Visualize the most popular category of trending video to give guidance to new users on what video to watch/create.
4. Analyze the influencing factors of video views to visualize content feedback and suggest the direction of improvement for their content.
5. Forecast the life cycle of trending videos to expect how long Youtubers' video trends will last.

III. Statistical Models & Methods

3.1 Exploratory Data Analysis

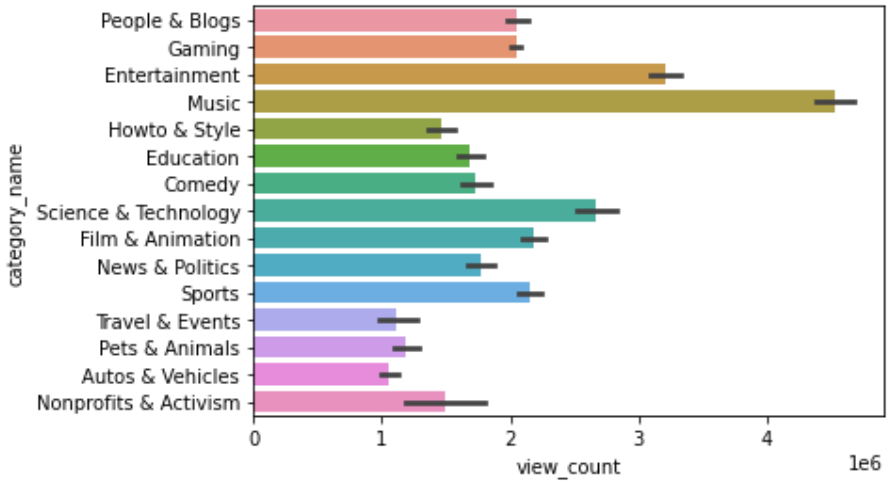
Exploratory data analysis is a method of exploring and understanding the relationship and trend between variables to manipulate data and summarize their main characteristics. We will explore data distribution and nature using visualization packages in Python.

This is the entire dataset information by using the *info( )* function, the report denotes the number of samples in each column and the data type of each column.

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 116033 entries, 0 to 119390
Data columns (total 22 columns):
 #   Column              Non-Null Count   Dtype
---  ------              --------------   -----
 0   video_id            116033 non-null  object
 1   title               116033 non-null  object
 2   publishedAt         116033 non-null  object
 3   channelId           116033 non-null  object
 4   channelTitle        116033 non-null  object
 5   categoryId          116033 non-null  int64
 6   trending_date       116033 non-null  object
 7   tags                116033 non-null  object
 8   view_count          116033 non-null  int64
 9   likes               116033 non-null  int64
 10  dislikes            116033 non-null  int64
 11  comment_count       116033 non-null  int64
 12  thumbnail_link      116033 non-null  object
 13  comments_disabled   116033 non-null  bool
 14  ratings_disabled    116033 non-null  bool
 15  description         116033 non-null  object
 16  kind                116033 non-null  object
 17  etag                116033 non-null  object
 18  id                  116033 non-null  int64
 19  category_name       116033 non-null  object
 20  snippet.assignable  116033 non-null  bool
 21  snippet.channelId   116033 non-null  object
dtypes: bool(3), int64(6), object(13)
memory usage: 18.0+ MB
```

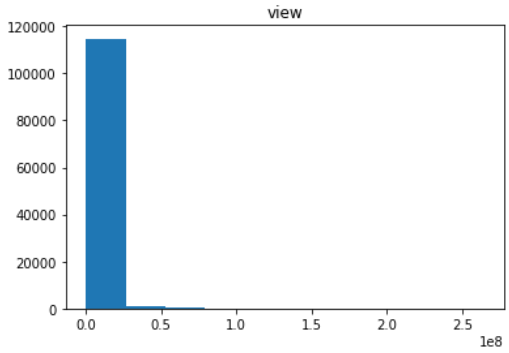
We can see from the table above that there are 116033 cases and 22 features (columns) with their appropriate data types, including *video ID*, *view counts*, *date of trending*, *likes*, *dislikes*, *number of comments*, and other relevant feedback.

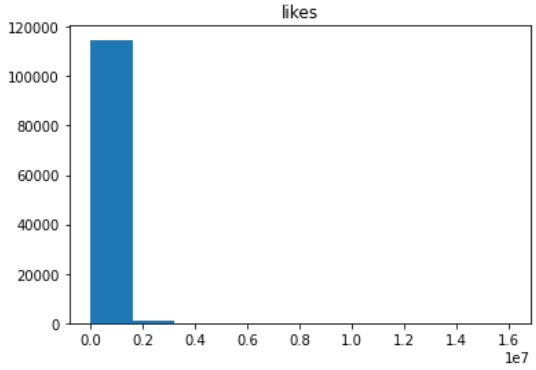
If you want to be a video creator and look for a head start on Youtube, it would be better to check which category is most popular among all with respect to the view count.

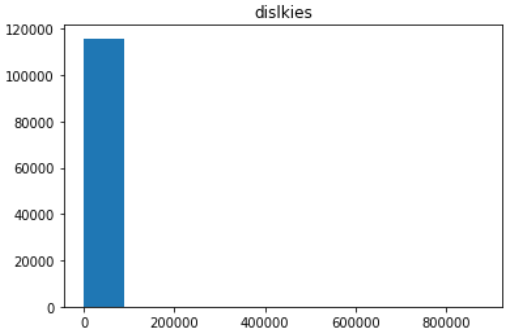


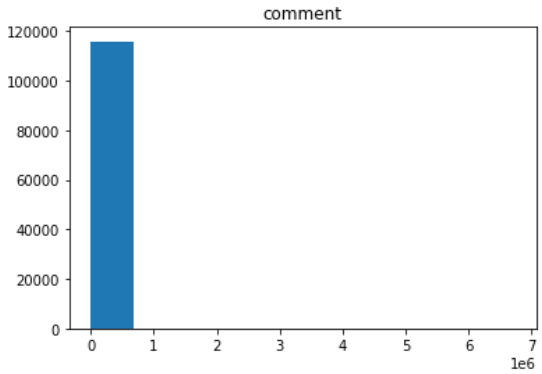
The graph shows the total view counts for each category. It can be seen easily that the *Music* category has the highest number of views, and the *Entertainment* is the second highest.

Next, we plotted histograms for certain variables to understand the distribution of data. From the graph, it is easy to find out the right-skew nature of the distribution. Some of the data lie far right compared to the rest of the samples, this illustrates that there are some hot videos way more popular than others based on their views, likes, dislikes, and the number of comments.

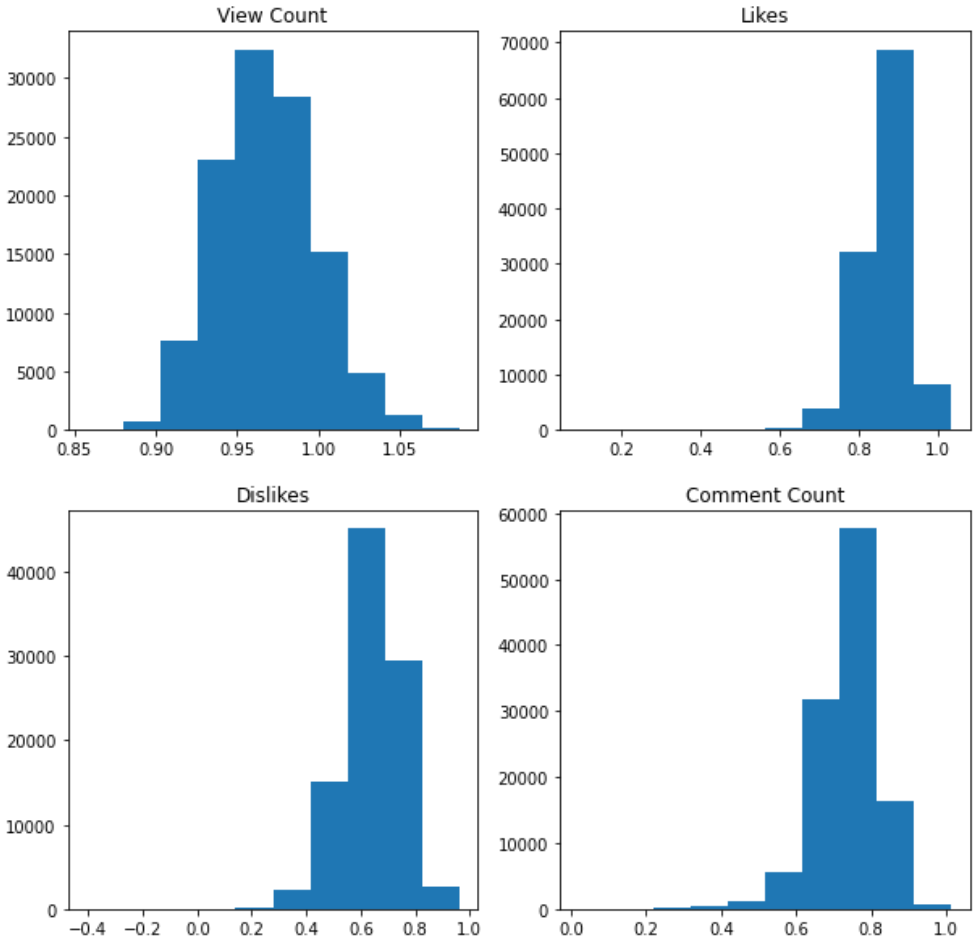








Since some statistical techniques such as ANOVA, ANCOVA, and linear regression require the normality of data, we will then transform the data by applying logarithmic functions. The plots after the logarithmic function are shown below:



From the plots above, important features such as view counts, likes, dislikes, and comment counts behave normally after applying the logarithmic function. There are still some outliers in these graphs.

3.2 ANOVA

ANOVA is a very useful tool to test if there are significant differences between groups. In this dataset, every trending video is marked with its category, we will first try One-Way ANOVA on categories with *likes*, *dislikes*, and *view count* and all F observed values are very large with p-values less than 0.05. Therefore, we can conclude that different groups marked with different categories have significant differences among them.

index	sum_sq	df	F	PR(>F)
view_count~category	3.4607216564772982	14.0	265.8106390206175	0.0
Residual	88.08900809837485	94723.0	NaN	NaN
likes~category	87.63139676319422	14.0	1572.5610740231316	0.0
Residual	377.03322364290415	94723.0	NaN	NaN
dislikes~category	153.06010365163078	14.0	341.09693148239154	0.0
Residual	3036.0687325498493	94723.0	NaN	NaN

Checking assumptions for ANOVA:

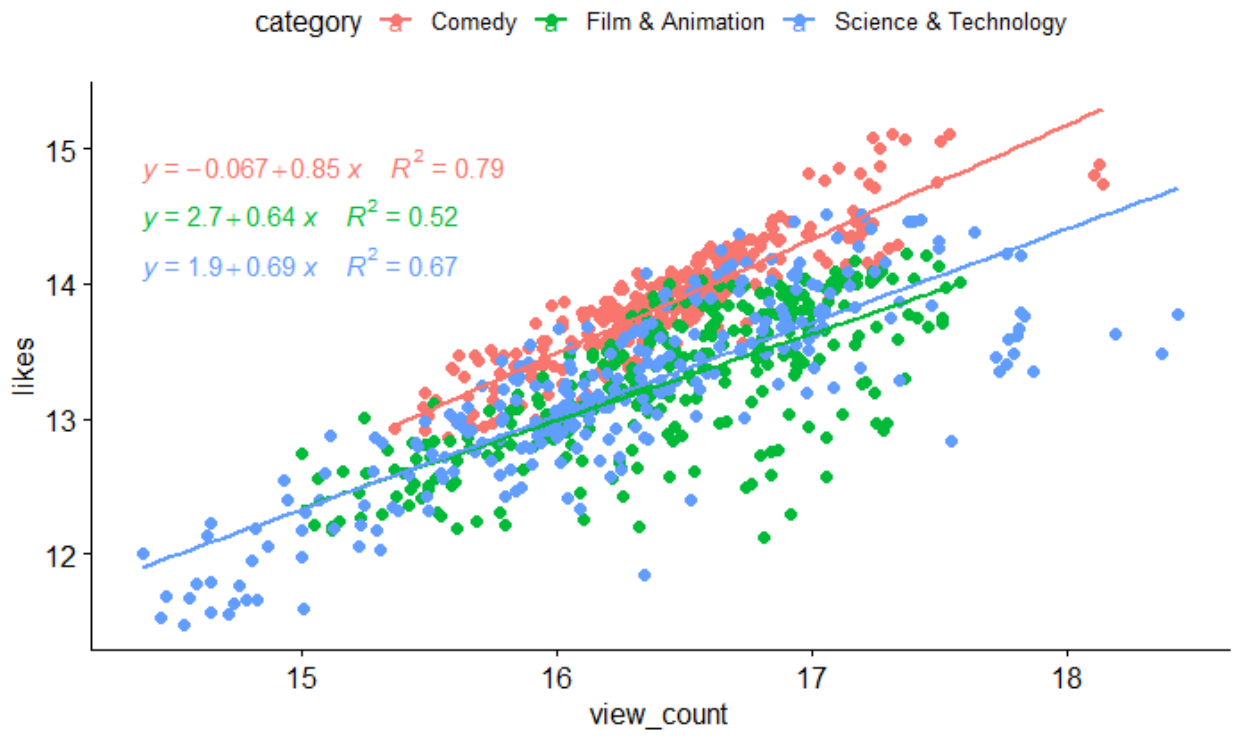
- Normality of data has been checked in the previous part.
- Since we have a large dataset, we can apply CLT to this dataset and we treat the condition satisfied as n is large enough

3.3 ANCOVA

ANCOVA is a strategy that combines linear regression and ANOVA. We have proved that there are significant differences between each category with different features including *view counts*, *likes*, and *dislikes*. Therefore, to quantify the differences between them, we use ANCOVA to adjust the mean for every feature with respect to their categories.

Since we have over 10 categories, we choose three categories that have similar view counts but different genres that may have different audiences. Moreover, since the *view count*, *likes*, and *dislikes* are large numbers without normal distribution, we log them to move on to our analysis.

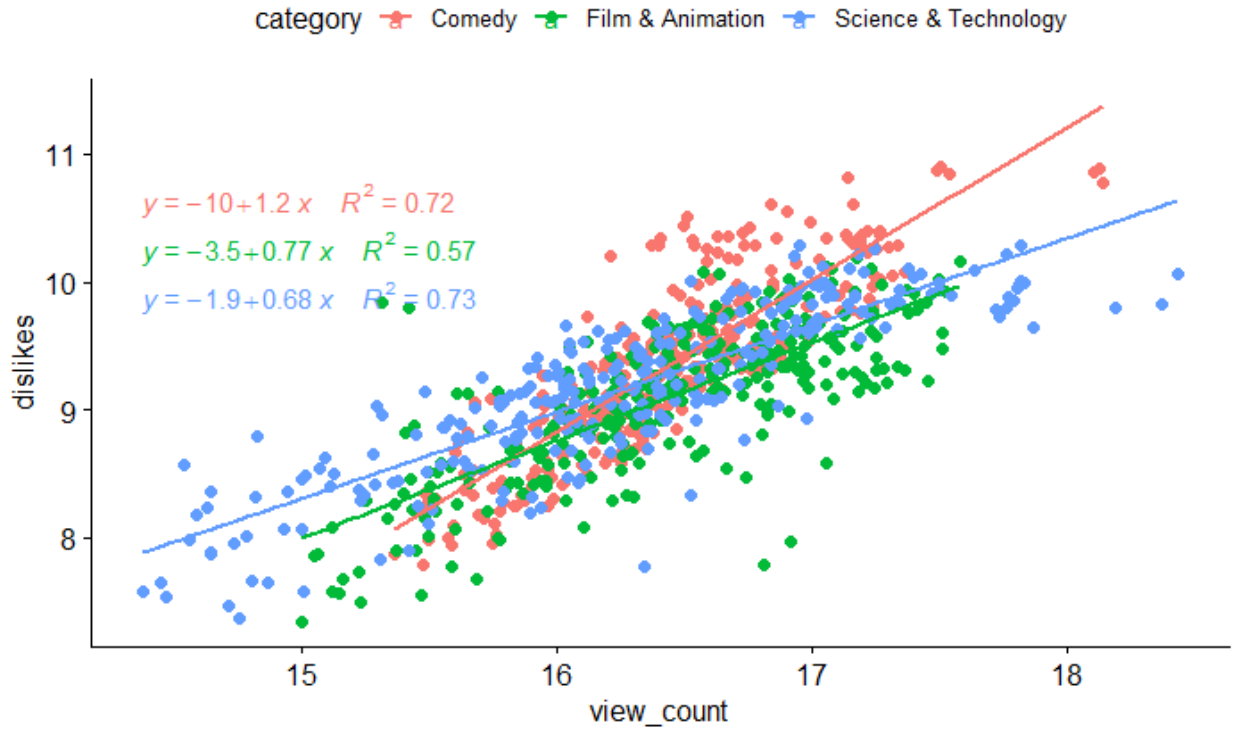
ANCOVA on likes versus view count



From the results, we can see that they basically have similar slopes with different intercepts. The category *Film&Animation* and *Science$Technology* have a very close slope compared with each

other. Therefore, ANCOVA will be a good model for predicting view count by likes in different categories with different means. We can also see that there are positive correlations between likes and view count. From this result, we observe that likes will increase as view count increases, and trending videos generally have very large view counts because more audiences favor the content. *Comedy* with the largest slope shows that more viewers will click likes after watching than the trending videos from the other two categories.

Dislikes vs View Count

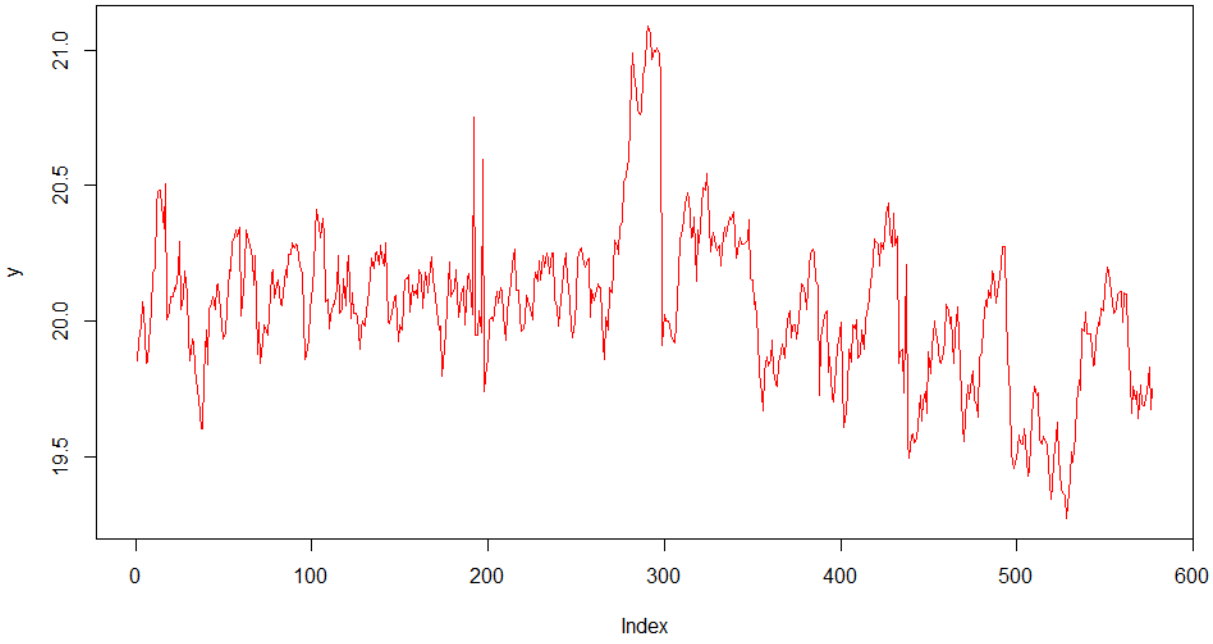


We can see that *Film&Animation* and *Science&Technology* have a close slope shown in the graph above, similar to the previous ANCOVA on *Likes* vs. *View count*. But we observe that *Comedy* has a much larger slope of 1.2, the largest slope among the three categories. Therefore, we conclude that a larger view count will lead to an obvious increase in both *likes* and *dislikes*, and the *Comedy* category has more significant effects on feedback than the other two categories.

IV. Results

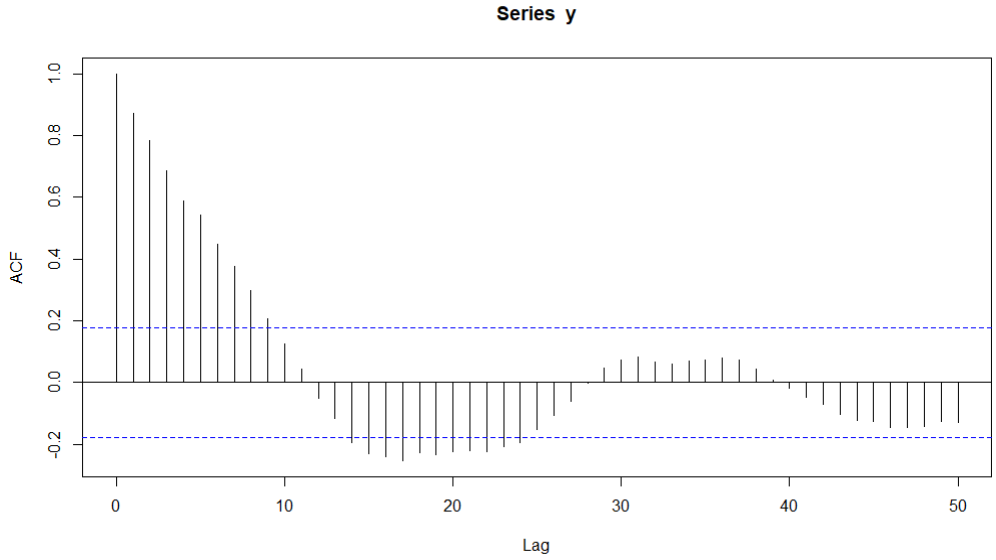
4.1 Time Series

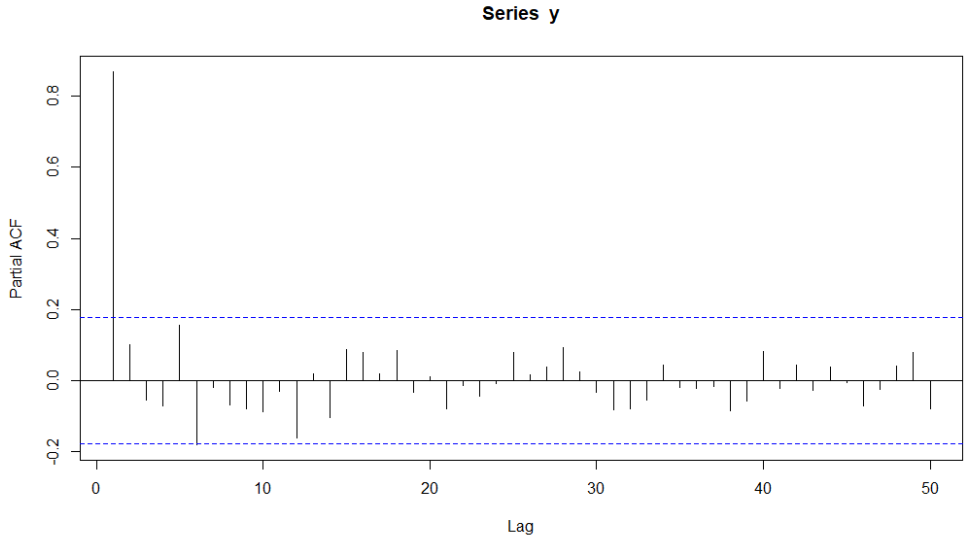
Time Series is a useful tool to predict the future by using the information and correlation in the past days. In this dataset, we will try to predict the view count by applying time series techniques. To analyze time series, view count has been applied to logarithmic function to stabilize the variance of residual.



After preprocessing the view count data, the time series acts differently after 300 hundred days from the beginning. To have a better prediction, we use the latest 120 days to analyze.

To imply the time series model, we plot the ACF plot to speculate parameters for MA, and PACF for AR parameters.





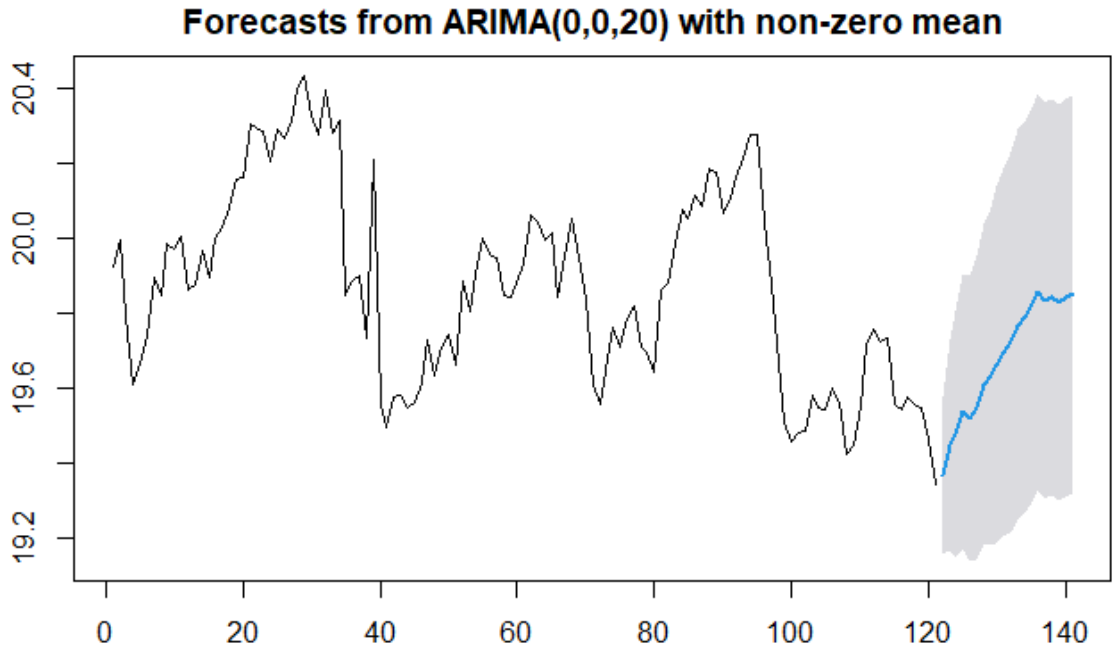
4.1.1. ARIMA Model

From ACF and PACF graphs, it is reasonable to assume that MA is 20 or 21, and AR for 0 or 1.

ARIMA Model order	AIC
(0, 0, 21)	-144.49

(1, 0 ,21)	-142.66
(0, 0, 20)	-146.52
(1, 0, 20)	-144.44

Since the smaller AIC value gets, the better a model fits. For the Time Series of view count, we chose the ARIMA model with order (0, 0, 20) to predict the future.



Since the dataset is updated on a daily basis, it gives us the chance to calculate the difference between our prediction and actual results. Our dataset ended on March 27, 2022, and we have the new dataset ended on May 1, 2022. By comparing the actual view count and the predicting view count for the next 20 days, the average error rate is 19%. We conclude that Time Series with the ARIMA model (0, 0, 20) can successfully predict the view count with a relatively low error rate.

4.1.2. Harmonic Model

In addition to the ARIMA model, we can also try to fit a harmonic model to the view count data since we observe that the data varies up and down in a period of about 30 days.

To imply this time series model, we first use the period T = 30 days to calculate the parameters λ:

$$\lambda = \frac{2\pi}{T}$$

Here we take $\lambda = 0.2$.

To fit the data better since the data has a certain trend, we can also add polynomial terms to the model, Then the model becomes:

$$\textbf{\textit{View count = Intercept + a*sin(}}\lambda t\textbf{\textit{) + b*cos(}}\lambda t\textbf{\textit{) + c*}}t + \textbf{\textit{d*}}t^2$$

By running this regression with view count from December 1, 2021, to April 1, 2022, a total of 120 days using R, we can easily find coefficients in the equation above:

$$\textit{Intercept} = 19.87433$$

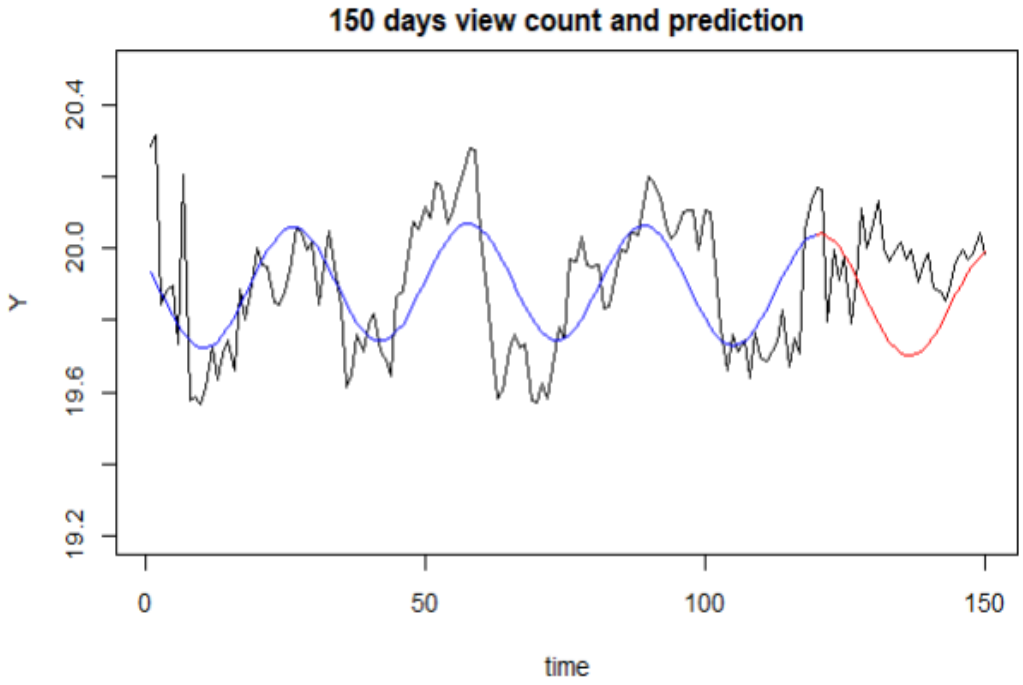
$$a = -0.1356309$$

$$b = 0.08866832$$

$$c = 0.00105505$$

$$d = -8.423769e\text{-}06$$

With this model, we can see if the total view count has a seasonal trend and then make predictions of the next month's view count based on the current data. We also test the validation of our forecasting data by comparing it with the updated data for the month of April 2022.



The blue line represents the fitted model and the red line represents the prediction with an average error rate of 13.93%. The average error rate of this model is smaller than the one with the ARIMA model (0, 0, 20). It seemed that the harmonic model is also a good way to predict the view count.

4.2 Multi-Linear Regression

4.2.1. Predictive analysis of the number of views

Linear regression model is the most common part of machine learning used to determine the relationship between target and features. Also, linear regression is helpful for analyzing continuous variables, as the result, in this part, our goal is to predict views based on some factors such as *likes* and *dislikes*.

From the correlation map, it is easier to find that the selected features were correlated with each other, and the most obvious features of a popular video: *likes*, *dislikes*, and *comments* all had significant correlations with each other. Thus, we want to use all of these features to predict the number of views and select the most significant features to avoid making the model redundant.

We start by splitting our data into a 70%-30% split to create a training set and testing set. In addition, we use 10-fold cross-validation to select the best model. We use the training set to train the model and the testing data to test how well the model will predict.

Our predicting variables are *category_ID*, *comment count*, *likes*, *dislikes*, *trending_year*, *trending_month*, *trending_day_number*, and *published_hour*. We used the built-in function for linear regression.

	Multi_Linear	Lasso	Ridge
Score	0.751512288	0.751512291	0.751512405

From the result, we find that the scores of these models are relatively similar. However, Lasso and Ridge are more computationally expensive than multi-linear regression, so in this case, we can choose a multi-linear model to predict views.

4.2.2. Forward selection

As we can see that there are lots of features to predict the views, but not all of them are equally important, and too many features will cause multicollinearity, so it is necessary to reduce dimension. In order to decide how many factors need to be chosen, we use forward selection. A type of stepwise regression that begins with an empty model and adds in variables one by one.

```
# forward selection
sfs1=sfs(LinearRegression())
sfs1=sfs1.fit(train_X,train_y)
names=sfs1.get_feature_names_out()
names
```

```
array(['comment_count', 'likes', 'dislikes', 'published_hour'],
      dtype=object)
```

From the above results, we choose *likes*, *dislikes*, *comment count*, and *published_hour* to be our independent variables in our model predicting the views of trending videos. Coincidentally, it is also consistent with our correlation map, where views have the highest correlation with *likes*, *dislikes*, *comment count*, and *published_hour*. The model we get is:

View = -10.63*comment +13.31*likes +137.68*dislikes -22295.9*published_hour +834145.2

4.3 Machine Learning

4.3.1. Predictive analysis of how long a video will trend

After a Youtube video is published, it generally trends for several days and fades out gradually. We know that if a Youtube video performs well on the trending page, this video will be revealed by more people and last longer on the trending list. To understand and predict generally how long a popular video will trend, we will use Machine Learning methods on our dataset. We will predict the total number of days for a video could trend based on the number of views, likes, dislikes, and comments.

To prepare the dataset for Machine Learning, we need to do further data cleaning. We choose to use the variables including *video_id*, *trending_date*, *publishedAt*, *view_count*, *likes*, *dislikes*, and *comment_count*. We first group by *video_id*, *trending_date*, and *publishedAt* variables to extract the maximum values for the other predictor variables. Then we create a new variable *total_trending_days* to evaluate the number of trending days for each video. The last step is to extract the *published year*, *month*, *day*, and *hour* as four predictors. Below is the head and the correlation table of the dataset.

	view_count	likes	dislikes	comment_count	total_trending_days	published_year	published_month	published_day	published_hour
0	2146104	167034	1755	12998	5	2021	6	10	16
1	3963014	218568	2847	15442	5	2021	6	10	16
2	5167987	240113	3414	16241	5	2021	6	10	16
3	6078723	252005	3778	16228	5	2021	6	10	16
4	6823249	262692	4107	16445	5	2021	6	10	16

	total_trending_days	view_count	likes	dislikes	comment_count	published_year	published_month	published_day	published_hour
total_trending_days	1.000000	0.372676	0.253476	0.189485	0.074529	-0.089375	-0.041575	0.016915	-0.026968
view_count	0.372676	1.000000	0.853163	0.656802	0.521632	-0.021390	0.008842	-0.009280	-0.067216
likes	0.253476	0.853163	1.000000	0.627886	0.693067	-0.033963	0.024067	0.005065	-0.049375
dislikes	0.189485	0.656802	0.627886	1.000000	0.529177	-0.092051	0.030803	0.001198	-0.056131
comment_count	0.074529	0.521632	0.693067	0.529177	1.000000	-0.042644	0.009680	0.001896	-0.045231
published_year	-0.089375	-0.021390	-0.033963	-0.092051	-0.042644	1.000000	-0.658048	-0.070302	0.006297
published_month	-0.041575	0.008842	0.024067	0.030803	0.009680	-0.658048	1.000000	0.034471	-0.001839
published_day	0.016915	-0.009280	0.005065	0.001198	0.001896	-0.070302	0.034471	1.000000	-0.001397
published_hour	-0.026968	-0.067216	-0.049375	-0.056131	-0.045231	0.006297	-0.001839	-0.001397	1.000000

From the correlation matrix, we do not find the multicollinearity problem since we do not find a close relationship between the predictors. We will use all the variables as predictors to predict the total number of trending days.

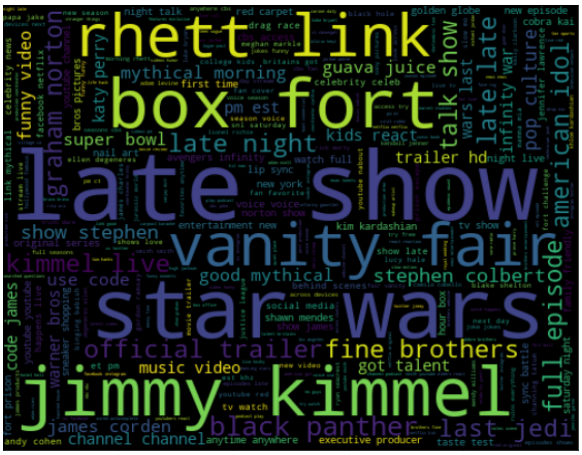
We are still using the 70%-30% split to create the training and testing data. We want to find the best regressor for this model. The four regressors are the decision tree, Lasso, Ridge, and the k-Nearest Neighbors(KNN). For the decision tree regressor, we set the maximum depth of the tree to 5. For KNN, we set the number of neighbors to 3. To evaluate each regressor, we calculate the mean absolute error (MAD), mean squared error(MSE), and accuracy. Below is the table of metrics for the 4 regressors.

	Decision Tree	Lasso	Ridge	KNN
MAD	1.134366058	1.287637300	1.252585144	1.640754848
MSE	2.995504728	4.177827825	4.122993465	6.126500462
Accuracy	0.395937134	0.777816251	0.168571752	0.222183749

From the table, we can find the decision tree regressor has the least MAD and MSE and the Lasso regression has the highest accuracy. As a result, we can use the Lasso regression to predict how long a video will trend and the average trending time is 5.909 days.

4.3.2. Predictive analysis of the video category classification

Before predicting the category of each video, we first visualize the most common words that appear in each category. Here we present *Entertainment* and *Sports* as examples:



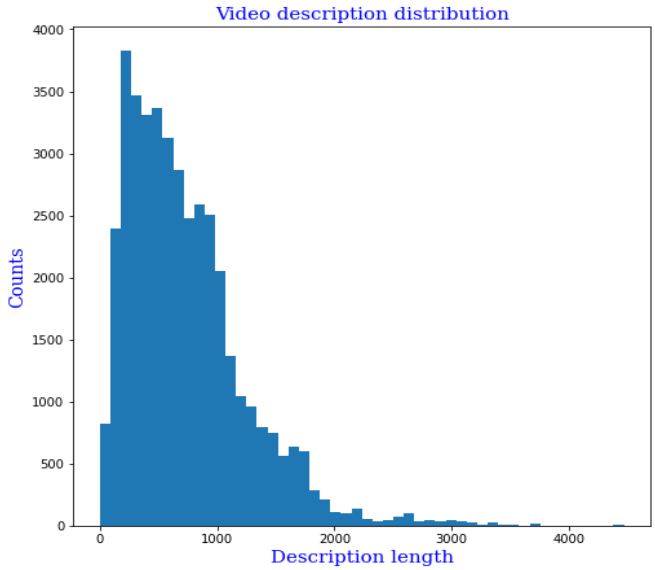
Entertainment

Sports

We aggregated all text information, which is *video titles*, *descriptions*, *tags*, *channel_titles*, and *descriptions*, as the input for predicting the category. Also, some common words such as ‘the’, ‘a’, ‘be’ might negatively affect the accuracy of the results, so we filtered them out of the text.

Since our inputs are text sequence data, we decided to choose RNN (LSTM) as our model. The next step is transforming the text data into numerical data. There are 16 categories, so we simply one-hot encoding each category into a vector with dim=16. For the input, we

tokenized each word into a number, and since each sample has a different length of text information, we padded the text sequence to keep them having equal lengths while feeding the model. We show the text length distribution below.



Theoretically, since the maximum length of text data in our samples is 4486, sequences that are shorter than 4486 are padded with 0 until they are 4486 long. However, we found that the model also works well while using 40 as the maximum length. That is, truncating the part in the sequence that is longer than 40 or padding the sequence with 0 until it is 40.

We then set the RNN architecture: first added an embedding layer that reduces the input dimension from 58818 to 128, an LSTM layer with 128 units, a dense layer with 128 units, and RELU activation function, an output layer with 16 units and Softmax activation function. The model was run for 5 epochs with batch size 128.

```
model = Sequential()
model.add(Embedding(unique_words, output_dim = 128,input_length = 40))
# Bi-Directional RNN and LSTM
model.add(Bidirectional(LSTM(128)))
# Dense layers
model.add(Dense(128, activation = 'relu'))
model.add(Dense(16,activation= 'softmax'))
model.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['acc'])
```

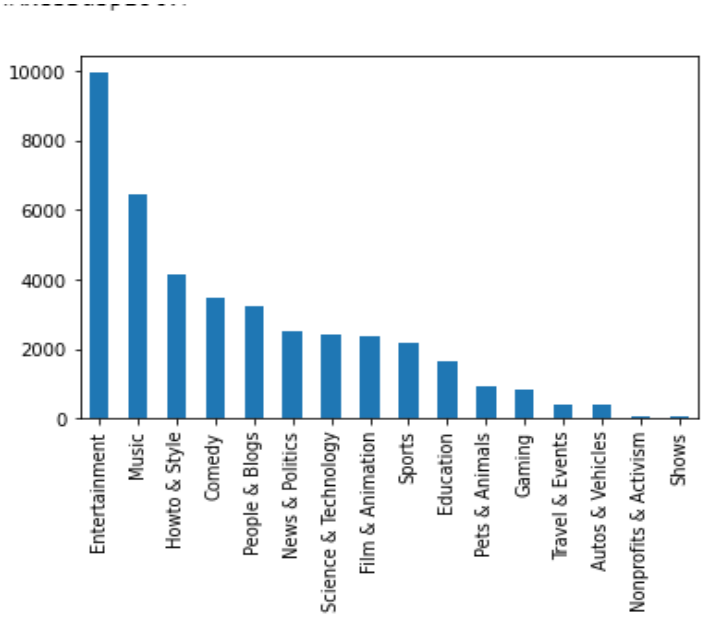
This model achieved a 98% accuracy in classifying the category on test data.

	precision	recall	f1-score	support
Autos & Vehicles	1.00	0.96	0.98	70
Comedy	0.98	0.97	0.97	704
Education	0.98	0.98	0.98	323
Entertainment	0.98	0.99	0.98	2041
Film & Animation	1.00	0.93	0.96	447
Gaming	0.97	0.96	0.97	159
Howto & Style	0.99	0.99	0.99	860
Music	1.00	0.98	0.99	1268
News & Politics	0.94	0.99	0.96	504
Nonprofits & Activism	0.52	1.00	0.69	12
People & Blogs	0.93	0.98	0.96	630
Pets & Animals	0.96	0.95	0.96	191
Science & Technology	0.98	0.98	0.98	470
Shows	0.81	1.00	0.90	13
Sports	0.99	0.93	0.96	428
Travel & Events	0.89	0.89	0.89	70
accuracy			0.98	8190
macro avg	0.93	0.97	0.94	8190
weighted avg	0.98	0.98	0.98	8190

Classification Report

Our model shows a good performance in most categories but we found that the model gets high precision and low recall in the category "*Nonprofits & Activism*". This implies that the model correctly classifies all "*Nonprofits & Activism*" but it also assigns "*Nonprofits & Activism*" to

many other videos that are not involved under this category. The reason for this occurrence might be due to the imbalance of our data: compared to other categories, there is a very small amount of "*Nonprofits & Activism*" videos in the data sample. The figure below is the distribution of video categories.



We can see the top 2 most trending categories *Entertainment* and *Music* have the largest weight among all the categories. Thus, the distribution behaves right-skewed and is biased for the prediction model. As a result, the model prediction performs not very well in the category *Nonprofits & Activism*. But it generally performs good in other categories with more views.

V. Conclusion

Through this project, we acquire the results of analyzing Youtube trending video data from August 2020 to March 2022 and thus propose some conclusions and assumptions in our report:

1. We successfully predict daily total views of trending videos on Youtube through a time series model ARIMA (0, 0, 20) and a harmonic model. Both methods suggest a monthly trend of daily total views with a peak around mid-month. The overall decreasing trend of recent total views signals the possible decline of user engagement and usage on Youtube.
2. We successfully use the multi-linear regression model to imply that variables *likes*, *dislikes*, *comment_count,* and *published_hour* have a significant influence on the views of a trending video. YouTubers can improve their video quality based on these aspects of their video feedback.
3. We successfully utilize the machine learning method to predict that a video will generally trend around 6 days. This would give Youtubers some insights on their trending video lifespan and the creation and upload cycle to get more chances for trending videos.
4. The top 3 popular categories of trending videos are *Music*, *Entertainment*, and *Science&Technology*. Users could refer to this list to follow the trending content, and YouTube could fund to simulate more high-quality videos in other categories.
5. Our classification model illustrates a good performance in predicting the category of trending videos. This would help users quickly find their interested content and assist advertisers efficiently in targeting their business field as well.

Understanding these results will not only help YouTube develop better features and algorithms to improve user engagement and earn profits but also benefit YouTubers to improve their video quality and content in order to compete for opportunities of having more trending videos.

Scope and Limitations

Due to the imbalance of our dataset and a limited number of records on certain video categories, some models are biased toward the majority classes. Since the data is not stable enough, the confidence interval for the ARIMA model prediction is very large. And the period of the harmonic model is not very accurate since the data may not follow a monthly period. If there exist some big events or relevant cases, the view counts usually fluctuate drastically. Since we only use the trending video dataset without other videos' information to predict views and categories, the prediction using machine learning could be more comprehensive and realistic, when more data are included to build robust models.

VI. Reference

jgolani2. “EDA and ML Insights on Youtube Trending Dataset.” Kaggle, Kaggle, 3 Feb. 2022, https://www.kaggle.com/code/jgolani2/eda-and-ml-insights-on-youtube-trending-dataset/notebook.

Kim, Young Hoon, Dan J. Kim, and Kathy Wachter. "A study of mobile user engagement (MoEN): Engagement motivations, perceived value, satisfaction, and continued engagement intention." Decision support systems 56 (2013): 361-370.

McCay-Peet, Lori, and Anabel Quan-Haase. "A model of social media engagement: User profiles, gratifications, and experiences." Why engagement matters. Springer, Cham, 2016. 199-217.

Sharma, Rishav. “YouTube Trending Video Dataset (Updated Daily).” Kaggle, 3 May 2022, https://www.kaggle.com/datasets/rsrishav/youtube-trending-video-dataset?resource=download.

“YouTube Statistics 2022.” Official GMI Blog. https://www.globalmediainsight.com/blog/youtube-users-statistics/.