***Introduction**:* Growing up, we watched countless hours of YouTube and always wanted to be like one of our idols. Therefore, the three of us wanted to create a successful YouTube channel called "Platypus Studios" and use data analytics to maximize its success. By analyzing video titles, tags, genres, and engagement metrics, we aim to provide insights and recommendations that can guide data-driven decisions to increase viewership, engagement, and popularity. Our ultimate goal is to develop a framework to optimize our channel and cater to the YouTube algorithm.

***Data Identification Process**:* We searched for a dataset on Kaggle and found a daily trending YouTube dataset containing youtube video data from 2020-08-12 to 2023-04-22. This dataset will help us understand how to optimize our YouTube channel, Platypus Studios, to maximize success.

***Data Ingesting**:* The Kaggle site included a CSV file for United States Trending videos. We downloaded and analyzed potential data-wrangling tasks, which will be described later. It also includes a CSV file for categories that describe the genres of YouTube videos. Afterward, we uploaded the data in SQL to perform mentioned tasks and are using a MySQL connector to bring the data into Python for data analysis. For the project, we only used the given datasets; we did not utilize APIs or web scraping.

***Data Wrangling Process**:* To process the raw data into usable data, we followed three steps. Step one was to create the relational database schema. Step two was to create the tables needed for the schema and import the data into those tables. Finally, step three was to alter those tables to create additional variables valuable for drawing insights.

Our schema includes three tables: a YouTube video table (YouTube_vid), a video category table (YouTube_cats), and a third table that combines data from both those tables (YouTube_final) into one (Exhibit 1). Each entry in the YouTube video table contains data from a YouTube video from a specific date, including # likes, # dislikes, # comments, as well as a “category id (cat_id).” This category ID represents the genre the YouTube video relates to. The category table includes the category id, the title of the category, and whether or not the category is assignable to the video. The primary key of the YouTube video table is simply the ID of each video entry at a specific point in time. Therefore, the primary key of

the video category table is cat_id, which can be referenced to the YouTube video table. Finally, we joined both YouTube_vid and YouTube_cats in a table called YouTube_final.

When importing the data into each table, we encountered a challenge with the boolean variables. Specifically, variables “comments_disabled” and “ratings_disabled” indicate whether the comments or ratings were disabled for a video. In the original dataset, the values under these columns were “TRUE” or “FALSE”. However, SQL could not convert these values into boolean variables. Therefore, we first converted those values into either “1” or “0”, then converted that into boolean variables.

After importing the data into the tables, we generated additional variables such as like-dislike ratio, like-view ratio, and comments-view ratio to gain valuable insights. These ratios provide perspective on the original variables (#likes, #comments, #views, etc.). Each additional variable requires two sets of codes. One code to ADD the variable and set the number of DECIMALs. Another is to SET the variable to an equation such as “like_view_ratio = likes / NULLIF(viewz, 0)” (the equation for the like-view ratio). Once this is done for each new variable, the data is reliable and complete for analysis in Python.

Data Analytics: We started by writing code that calculates the correlation matrix for our YouTube data frame and then plotting it as a heatmap using the sns.heatmap() function. The correlation matrix shows the correlation coefficients between all pairs of variables in the dataframe, and the heatmap makes it easier to visualize the strength and direction of these correlations (Exhibit 2). For example, we found that views, likes, dislikes, and comment_count strongly correlate. These are all strong indicators of success for a YouTube channel.

Next, we wrote code that performs a linear regression analysis on the YouTube dataset. First, it defines the features as 'likes' and 'dislikes,' and the target variable as whether a video is viral (viewed more than 10 million times). It then splits the data into training and testing sets and trains a Linear Regression model on the training data. Next, it makes predictions on the testing set using the trained model and evaluates the model's performance using the R-squared score. We found that the R-squared score for this model is 0.31, which indicates a weak approximation of the actual data. However, we then split the data into training and testing sets again, this time using three features - 'likes,' 'dislikes,’ and

'comment_count' - to predict the target variable 'viewz' (not virality). We found that the R-squared score was 0.77, which indicates a strong approximation of the actual data. Therefore, this implies that likes, dislikes, and comments are a good indicator of the # of views for a YouTube video but less of an indicator of whether or not a video goes "viral." To confirm this hypothesis, we wrote code that performs multiple linear regression using the Ordinary Least Squares (OLS) method to fit a model that predicts the number of views (viewz) of YouTube videos based on the number of likes, dislikes, and comments. The results indicate that likes, dislikes, and comment count are significant, with a p-value of <0.5 (Exhibit 4).

Our next goal was to find the best time to post videos to maximize our chances of success. Therefore, we wrote our next section of code to analyze the relationship between the publishing month, trending month, and the average number of views, likes, dislikes, and comment_count of YouTube videos. (Exhibits 5-9). By grouping the data by the publishing month and trending month and plotting the average number of views against these months, the code allows for the visualization of trends and patterns that indicate the most optimal months to post YouTube videos to maximize views and engagement. The code also provides insights into the average number of likes and dislikes by month and the like/view ratio, which can inform us about how video content resonates audiences during different months of the year. Overall, we found that the best month to post was around May or June, as the videos that were published in May or June had the most views. Furthermore, videos that went trending in June had the most views.

To improve our video titles and tags, we generated word clouds using code to highlight the top words used in titles (Exhibit 10), excluding music-related words (Exhibit 11), and the top tags used (Exhibit 12). The exclude_words list in our code eliminated common words like prepositions and conjunctions, while the STOPWORDS set updated with words to exclude. Then, a WordCloud object with these stopwords and a white background was created, and the resulting visualization showed the frequently used words in video titles. However, we recognized these insights' limitations and opted to determine the ideal category for our YouTube channel's success before we dive into word analysis.

To find the best YouTube channel category for Platypus Studios, we wrote code that reads data from our SQL database table YouTube_final, calculates the average number of views per video for each

video category, and returns the top 3 categories with the highest average views per video. We found that the top 3 were (respectively): Music, Entertainment, and Nonprofits & Activism (Exhibit 13). Since none of us are great musicians, and none of us are currently invested in non-profits, we decided that entertainment would maximize our YouTube channel success.

After deciding to focus on Entertainment, our next objective was to identify the most effective video titles and tags that would increase our viewership within this category. We developed a code to produce the top 10 tags that generated the highest number of views in the entertainment segment. Our findings showed that the top two tags were "funny" and "marvel" (refer to Exhibit 14). Furthermore, we conducted a similar analysis to determine the top 10 words in titles that generated the most views in the entertainment category. Our results highlighted the significance of keywords such as "trailer", "season", "vs", "official", among others (refer to Exhibit 15). Considering that the words "trailer" and "season" are reliant on having a product to promote, we concluded that "vs" would be the optimal title word to use.

Now that we have found the best month, best category, best tags, and best title words to maximize our viewership, which is a measurement of success, we can summarize our insights into the following: Best month: May/June, Best Category: Entertainment, Best tag(s): “funny”, “marvel”, and best title words: “vs”. Therefore, the first YouTube video for Platypus Studios will be a funny video about two marvel characters “vs”-ing each other. We will assign them to the “entertainment” category, and release the video around May/June.

To forecast the potential views for our upcoming video, we developed an ML linear regression model widely utilized in Big Data due to its versatility. Our predicted views for the new video amounted to 5,095,687 views.

At Platypus Studios, we think our data analysis was well worth our time, energy, and efforts. We look forward to creating videos related to entertainment, releasing around May/June, and estimating about 5.1M views per video.

Exhibit 1: Schema Model of Final YT Trending Dataset

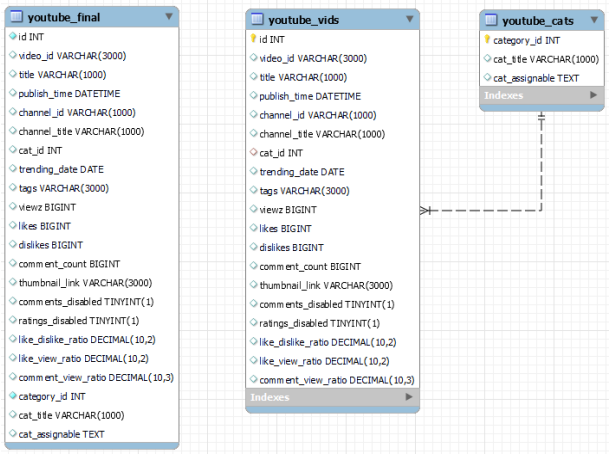

Exhibit 2: Correlation of all variables in YT Trending Dataset

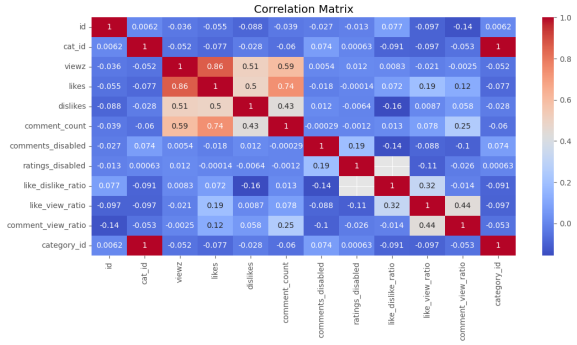

Exhibit 3: Scatterplot of "likes" to "viewz"

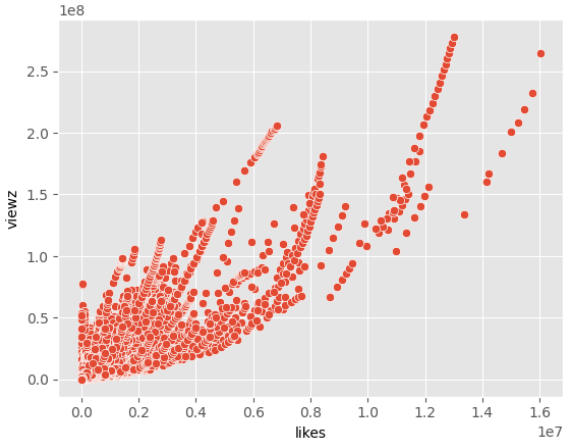

Exhibit 4: OLS Regression Model of "likes", "dislikes", "comment_count" to "viewz"

OLS Regression Results

Dep. Variable:	viewz	R-squared:	0.754
Model:	OLS	Adj. R-squared:	0.754
Method:	Least Squares	F-statistic:	2.022e+05
Date:	Mon, 01 May 2023	Prob (F-statistic):	0.00
Time:	22:31:29	Log-Likelihood:	-3.2599e+06
No. Observations:	197590	AIC:	6.520e+06
Df Residuals:	197586	BIC:	6.520e+06
Df Model:	3		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	4.025e+05	8492.131	47.394	0.000	3.86e+05	4.19e+05
likes	15.3015	0.030	510.561	0.000	15.243	15.360
dislikes	90.6155	0.997	90.901	0.000	88.662	92.569
comment_count	-9.1427	0.145	-63.013	0.000	-9.427	-8.858

Omnibus:	248787.089	Durbin-Watson:	1.977
Prob(Omnibus):	0.000	Jarque-Bera (JB):	197205774.696
Skew:	6.327	Prob(JB):	0.00
Kurtosis:	157.251	Cond. No.	4.64e+05

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 4.64e+05. This might indicate that there are strong multicollinearity or other numerical problems.

Exhibit 5: Average Number of Views by Publish Month

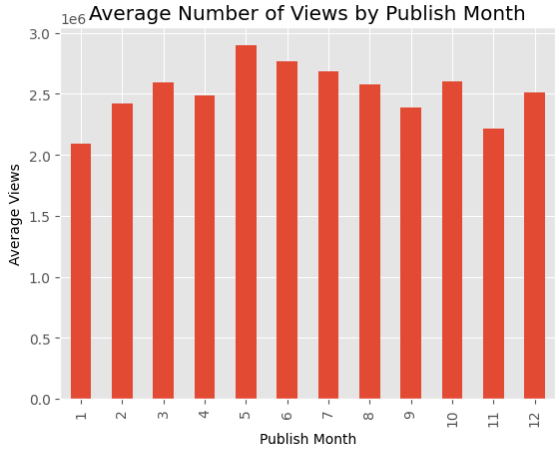

Exhibit 6: Average Number of Views by Trending Month

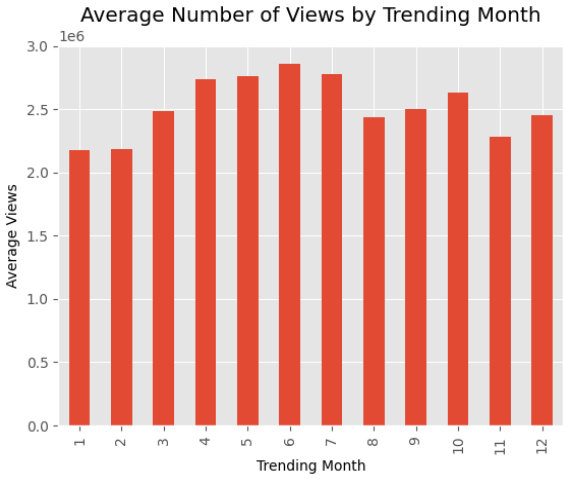

Exhibit 7: Average Number of Likes and Dislikes by Publish Month

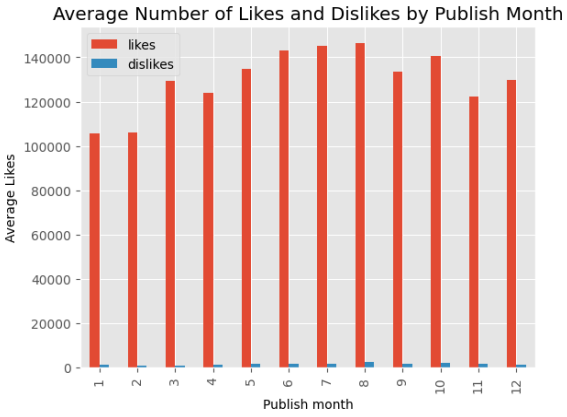

Exhibit 8: Average Number of Likes and Dislikes by Trending Month

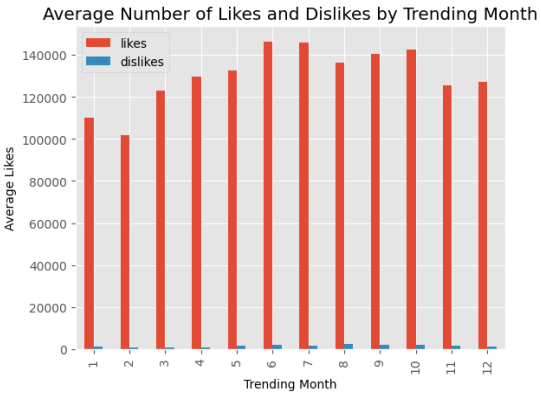

Exhibit 9: Average Like/View Ratio by Trending Month

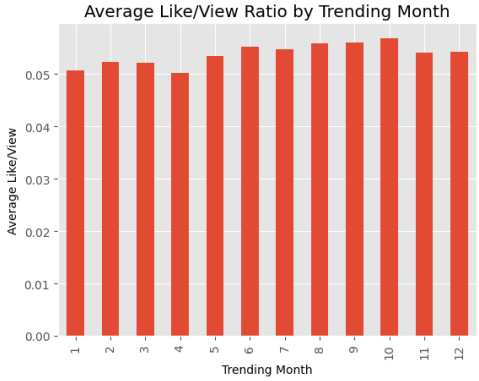

Exhibit 10: WorldCloud for "titles"

Exhibit 11: WordCloud for "titles" without "official", "video," "music," "oficial," "None"

Exhibit 12: WordCloud for "tags"

Exhibit 13: Top 3 Categories based on avg. views per video

```
cat_df = pd.read_sql(query, cnx)
              cat_title  avg_views_per_video
0                 Music             25524923
1         Entertainment             17362974
2  Nonprofits & Activism             15514909
```

Exhibit 14: Top 10 tags that generate the most views in the Entertainment category

```
dfe = pd.read_sql(sql_select_query, cnx)
                tag        views
13948        [None]  45244545351
24025         funny  11414023654
31522        marvel   6248151674
19507        comics   5702809349
18676     challenge   4998628313
19479        comedy   4783138988
29783         laugh   4210683469
23979           fun   3846598508
24111  funny videos   3564294537
33570           new   3458805171
```

Exhibit 15: Top 10 words in "channel_title" that generate the most views in the Entertainment category

```
            title     views
27632     trailer 888218049
24207      season 830175369
29604          vs 750674296
20137    official 730155003
19641         new 661988205
19123       music 572541413
2788            2 542263321
28747          us 525475916
2014   (official 502611917
17116        live 448792790
```