INTRODUCTION TO DATA-SCIENCE PROJECT

Topic: - TRENDING YOUTUBE VIDEOS STATISTICS

Data set choosen: Trending Youtube videos

The data set chosen is the set of youtube videos that was trending in the US.

- The dataset contains over 36000 rows and 16 columns.
- The columns of the dataset are:

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

video_id	trending_date	title	channel_title	category_id	publish_time	tags	views	likes	dislikes	...
2kyS6SvSYSE	17.14.11	WE WANT TO TALK ABOUT OUR MARRIAGE	CaseyNeistat	22	2017-11-13T17:13:01.000Z	SHANtell martin	748374	57527	2966	...
1ZAPwfrtAFY	17.14.11	The Trump Presidency: Last Week Tonight with J...	LastWeekTonight	24	2017-11-13T07:30:00.000Z	last week tonight trump presidency"|"last week...	2418783	97185	6146	...
5qpjK5DgCt4	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	146033	5339	...
puqaWrEC7tY	17.14.11	Nickelback Lyrics: Real or Fake?	Good Mythical Morning	24	2017-11-13T11:00:04.000Z	rhett and link"|"gmm"|"good mythical morning"|...	343168	10172	666	...
d380meD0W0M	17.14.11	I Dare You: GOING BALD!?	nigahiga	24	2017-11-12T18:01:41.000Z	ryan"|"higa"|"higatv"|"nigahiga"|"i dare you"|...	2095731	132235	1989	...

ɔws × 29 columns

Cleaning the data-set

```
In [107]: print('NaN CELLS {}:\n{}\n'.format('BEFORE DIRTYING', my_df.isna().sum()))

NaN CELLS BEFORE DIRTYING:
video_id                      0
trending_date                 0
title                         0
channel_title                 0
category_id                   0
publish_time                  0
tags                          0
views                         0
likes                         0
dislikes                      0
comment_count                 0
thumbnail_link                0
comments_disabled             0
ratings_disabled              0
video_error_or_removed        0
description                 570
Unnamed: 16               40767
Unnamed: 17               40903
Unnamed: 18               40928
```

Since the dataset does not contain any dirty(NaN) values,we make 3% data of two columns NaN.

```
In [108]: for i in range(1, 36000, 30):
              my_df.loc[[i], my_df.columns[4]] = np.nan
              my_df.loc[[i], my_df.columns[10]] = np.nan

In [109]: print('NaN CELLS {}:\n{}\n'.format('AFTER DIRTYING', my_df.isna().sum()))

NaN CELLS AFTER DIRTYING:
video_id                      0
trending_date                 0
title                         0
channel_title                 0
category_id                1200
publish_time                  0
tags                          0
views                         0
likes                         0
dislikes                      0
comment_count              1200
thumbnail_link                0
comments_disabled             0
ratings_disabled              0
video_error_or_removed        0
description                 570
Unnamed: 16               40767
Unnamed: 17               40903
Unnamed: 18               40928
```

And now, All the NaN cells for categorical columns will be replaced with their previous row's value. All the NaN cells for numerical columns will be replaced with their column's average.

```
In [110]: avg = my_df[my_df.columns[10]].mean()
          for i in range(1, 36000, 30):
              my_df.loc[[i], my_df.columns[10]] = avg
          my_df.fillna(method='ffill', inplace=True)

In [111]: print('NaN CELLS {}:\n{}\n'.format('AFTER CLEANING', my_df.isna().sum()))

NaN CELLS AFTER CLEANING:
video_id                  0
trending_date             0
title                     0
channel_title             0
category_id               0
publish_time              0
tags                      0
views                     0
likes                     0
dislikes                  0
comment_count             0
thumbnail_link            0
comments_disabled         0
ratings_disabled          0
video_error_or_removed    0
description               0
Unnamed: 16               0
```

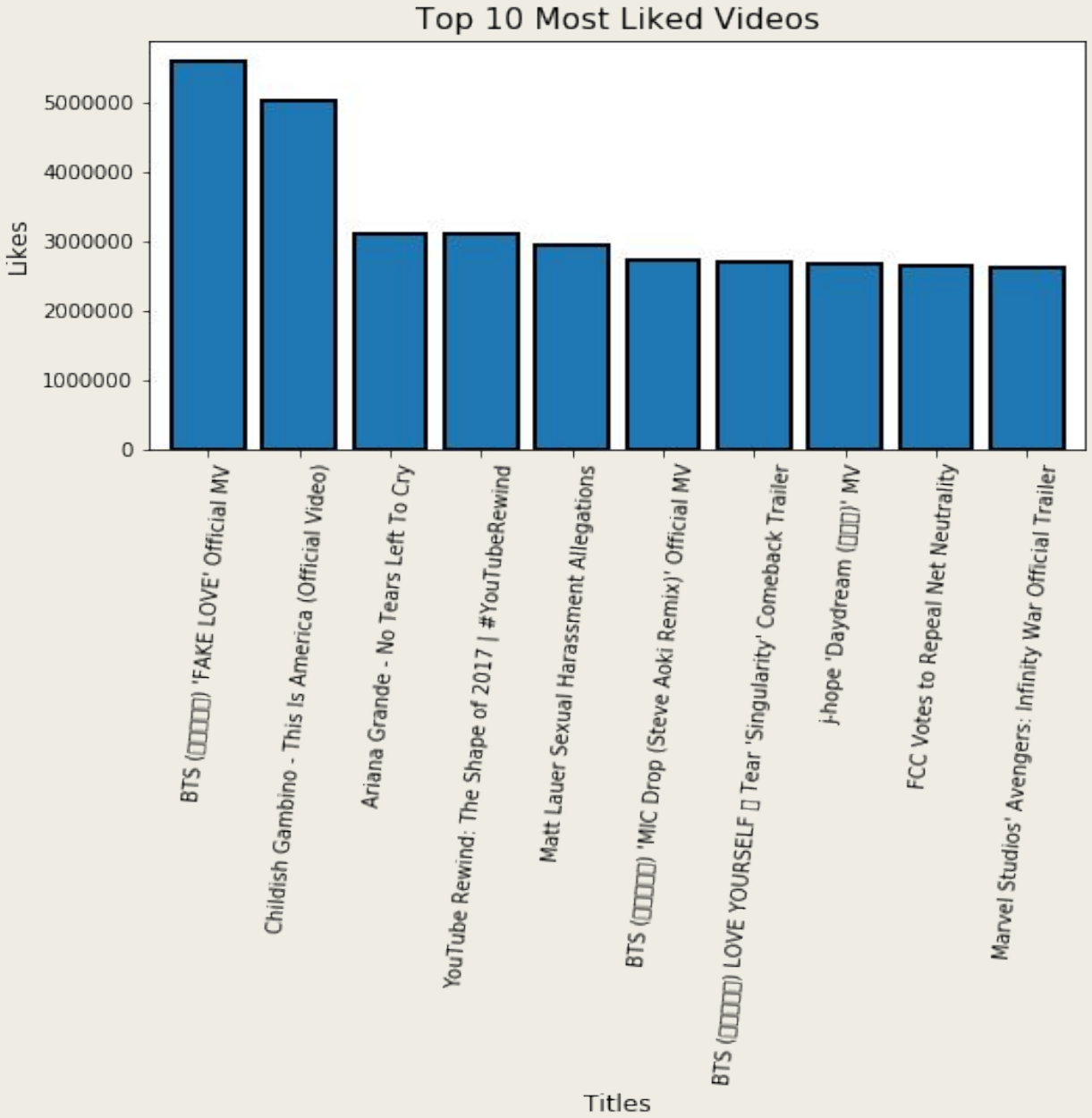
Hypothesis Testing

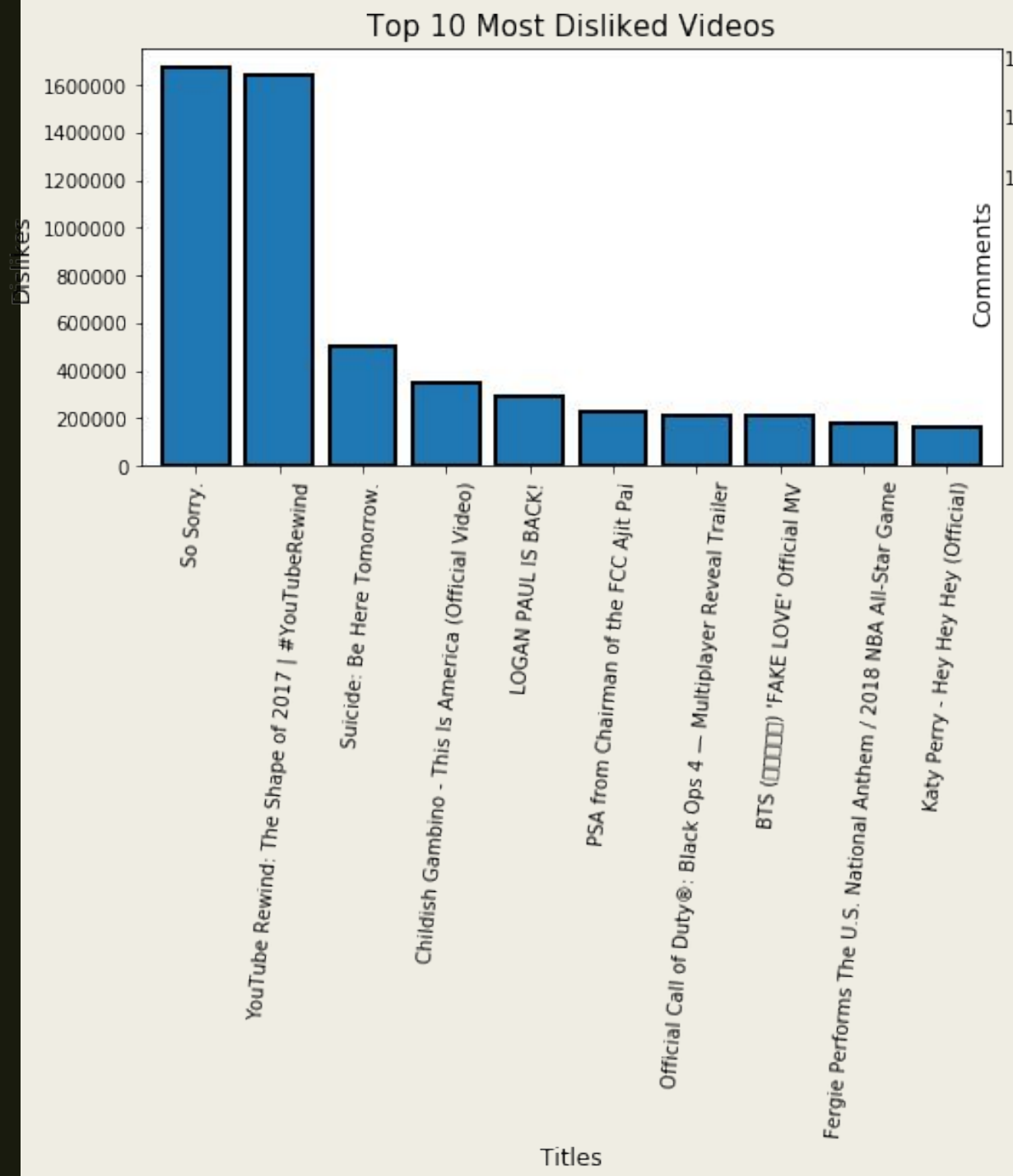
Null hypothesis: Trending videos receive an average of over 3.5k dislikes each
Alternate hypothesis: Trending videos receive less than or equal to 3.5k dislikes each

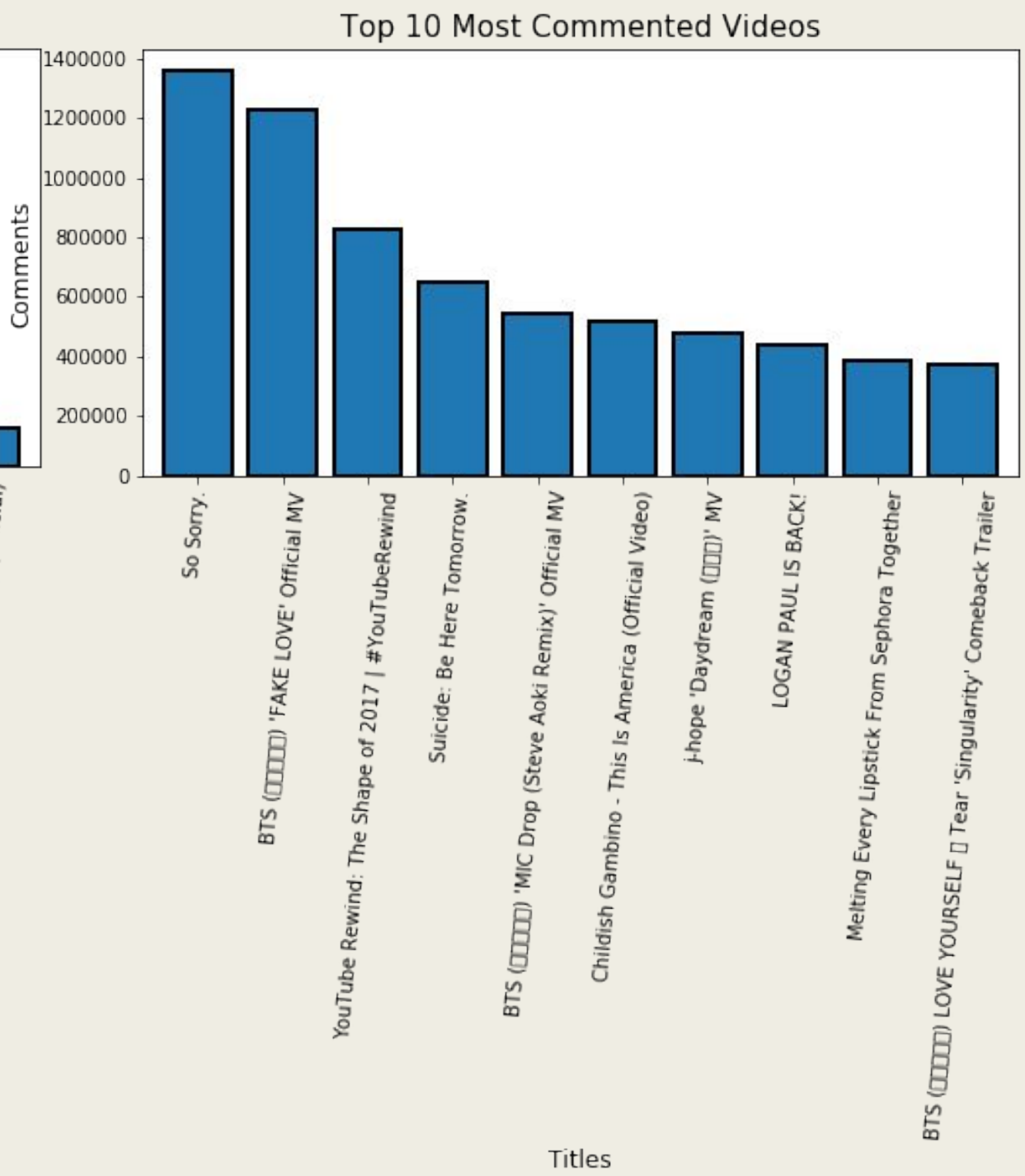
In [116]:
```
print('Null hypothesis: Trending videos receive an average of over 3.5k dislikes each')
print('Alternate hypothesis: Trending videos receive less than or equal to  3.5k dislikes each')
alpha = 0.05
x =my_df['dislikes'].mean()
mu = 3500
std = my_df['dislikes'].std()
root_n =my_df['dislikes'].count() ** (1/2)
z = (x-mu)/(std/root_n)
print('alpha: {}\nmu: {}\nx: {}\nstd: {}\nroot_n: {}\nz: {}'.format(alpha, mu, x, std, root_n, z))
print('Putting this value in the z-table we get:-')
print('p value:', 0.96)
print('Therefore, we reject the null hypothesis.')
```

```
Null hypothesis: Trending videos receive an average of over 3.5k dislikes each
Alternate hypothesis: Trending videos receive less than or equal to  3.5k dislikes each
alpha: 0.05
mu: 3500
x: 3744.4168111553395
std: 29049.336049107784
root_n: 202.35859260234045
z: 1.7026152277675077
Putting this value in the z-table we get:-
p value: 0.96
Therefore, we reject the null hypothesis.
```

Let’s plot some data before we normalise it.




Top 10 Most Disliked Videos
Dislikes
1600000
1400000
1200000
1000000
800000
600000
400000
200000
0
So Sorry.
YouTube Rewind: The Shape of 2017 | #YouTubeRewind
Suicide: Be Here Tomorrow.
Childish Gambino - This Is America (Official Video)
LOGAN PAUL IS BACK!
PSA from Chairman of the FCC Ajit Pai
Official Call of Duty®: Black Ops 4 — Multiplayer Reveal Trailer
BTS (□□□□) 'FAKE LOVE' Official MV
Fergie Performs The U.S. National Anthem / 2018 NBA All-Star Game
Katy Perry - Hey Hey Hey (Official)
Titles


Top 10 Most Commented Videos
Comments
1400000
1200000
1000000
800000
600000
400000
200000
0
So Sorry.
BTS (□□□□) 'FAKE LOVE' Official MV
YouTube Rewind: The Shape of 2017 | #YouTubeRewind
Suicide: Be Here Tomorrow.
BTS (□□□□) 'MIC Drop (Steve Aoki Remix)' Official MV
Childish Gambino - This Is America (Official Video)
j-hope 'Daydream (□□□)' MV
LOGAN PAUL IS BACK!
Melting Every Lipstick From Sephora Together
BTS (□□□□) LOVE YOURSELF □ Tear 'Singularity' Comeback Trailer
Titles

Notice that YouTube Rewind 2017 is present in the top 10 of most liked,disliked and commented videos.This shows that it is not necessary for a video that has a lot of dislikes to not to go to the Trending Page of YouTube.All that matters is how much the viewers interact (like,dislike,comment) with the video.

Some more interesting compilation.

```
In [122]: my_df[['channel_title','comment_count']].groupby('channel_title',axis=0)
          |.sum().sort_values(by='comment_count',ascending=False).head(10)
```

Out[122]:

channel_title	comment_count
ibighit	31372065
Logan Paul Vlogs	13593022
ChildishGambinoVEVO	10151289
jypentertainment	7537073
Marvel Entertainment	6444740
YouTube Spotlight	4998569
Safiya Nygaard	4373417
ArianaGrandeVevo	4295333
Call of Duty	4109126
jacksfilms	3964916

jacksfilms 3964916

```
In [123]: my_df[['channel_title','likes']].groupby('channel_title',axis=0)
          |.sum().sort_values(by='likes',ascending=False).head(10)
```

Out[123]:

channel_title	likes
ibighit	199247121
ChildishGambinoVEVO	96700818
Dude Perfect	60275557
Marvel Entertainment	55873344
ArianaGrandeVevo	52170970
jypentertainment	44900910
TaylorSwiftVEVO	39292840
Ed Sheeran	39279211
ZaynVEVO	31695245
Logan Paul Vlogs	31545290

Normalization and Standardization
We'll normalize the numerical columns in order to make the mean 0, and the variance 1.

Why is normalization important?
Normalization is important because it brings all the values of numerical columns to a common scale.

How does normalization affect the dataset?
It affects the dataset by making all the elements lie between 0 and 1.

Normalisation

Downloads/ | idsprojectfinal

localhost:8888/notebooks/Downloads/idsprojectfinal.ipynb

For quick access, place your bookmarks here on the bookmarks bar. Import bookmarks now...

jupyter idsprojectfinal Last Checkpoint: an hour ago (autosaved)

Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

Run Code

In [126]:

```
print("Before normalisation:")
for col in type_int_list:
        print('{0} mean = {1}\n{0} variance = {2}'.format(col, my_df[col].mean(), my_df[col].var()))
print()
for col in type_int_list:
    values = my_df[[col]].values.astype(float)
    my_df[col] = pd.DataFrame(MinMaxScaler().fit_transform(values))
print("After normalisation:")
for col in type_int_list:
        print('{0} mean = {1}\n{0} variance = {2}'.format(col, my_df[col].mean(), my_df[col].var()))
print()
```

```
Before normalisation:
likes mean = 76281.4717819727
likes variance = 55786866586.375244
dislikes mean = 3744.4168111553395
dislikes variance = 843863924.893993
comment_count mean = 8441.19131114313
comment_count variance = 1344853933.5722234

After normalisation:
likes mean = 0.013588140814095798
likes variance = 0.0017701656903189567
dislikes mean = 0.0022362470653452308
dislikes variance = 0.0003009841428674703
comment_count mean = 0.006199555891789784
comment_count variance = 0.0007254187732034617
```

Correlation

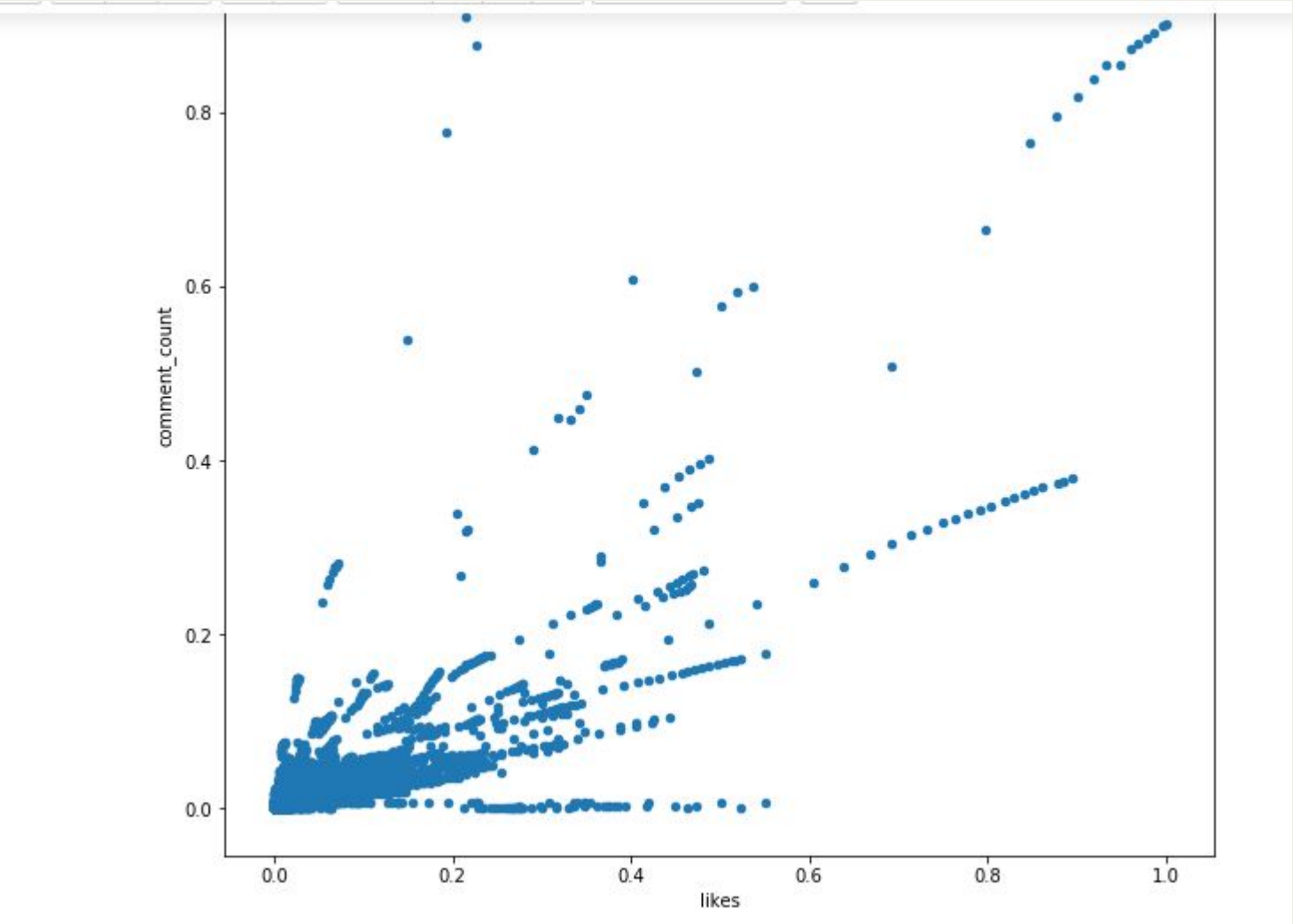
In [129]:
```
keep_columns = ['likes', 'dislikes', 'comment_count'] # only looking at correlations between these variables
corr_matrix = my_df[keep_columns].corr()
corr_matrix
```

Out[129]:

	likes	dislikes	comment_count
likes	1.000000	0.440239	0.767594
dislikes	0.440239	1.000000	0.653605
comment_count	0.767594	0.653605	1.000000

In [130]:
```
my_df.plot.scatter(x='likes', y='comment_count')
```

The correlation coefficient is 0.77 between the number of comments
and the number of likes.

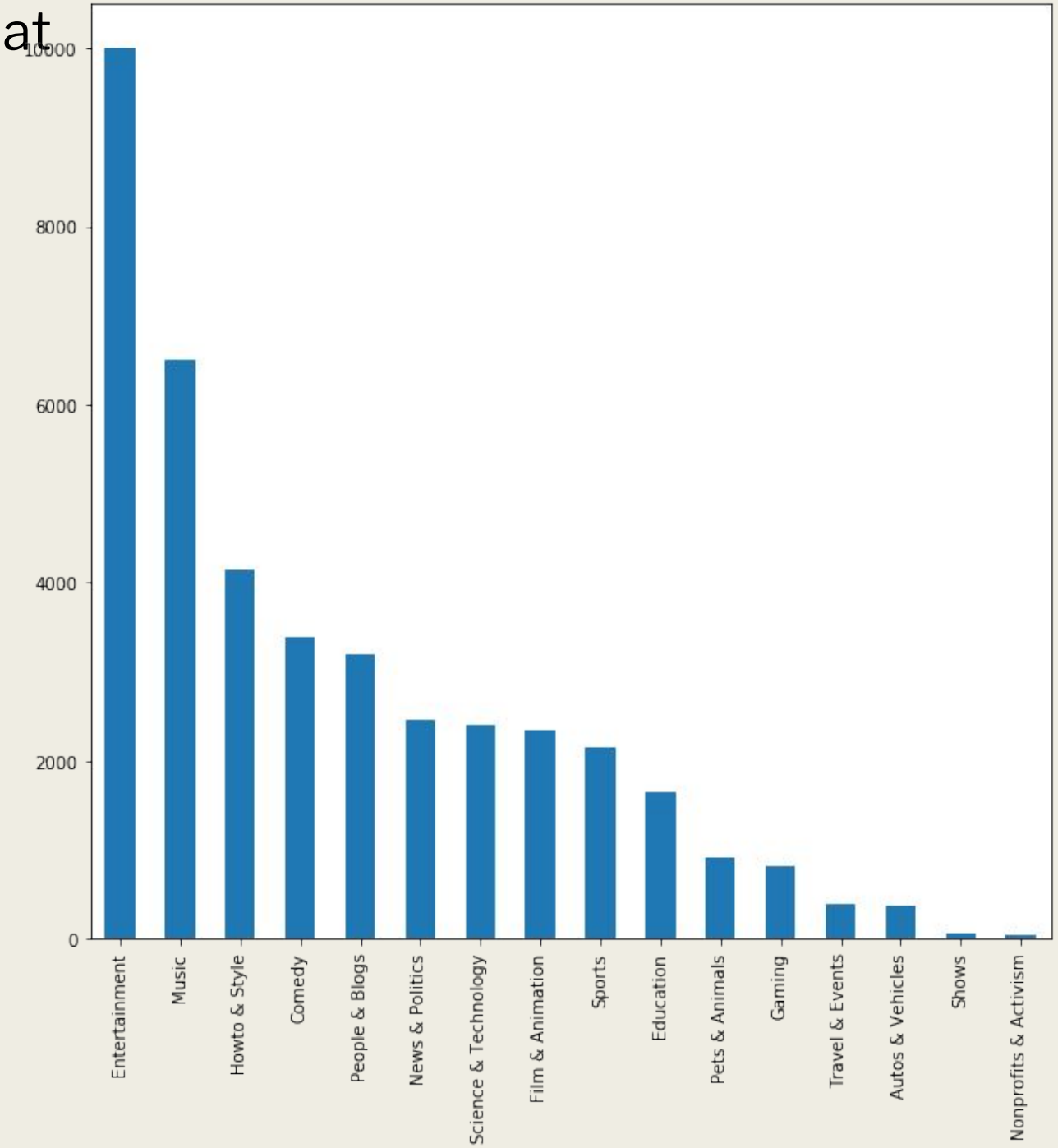


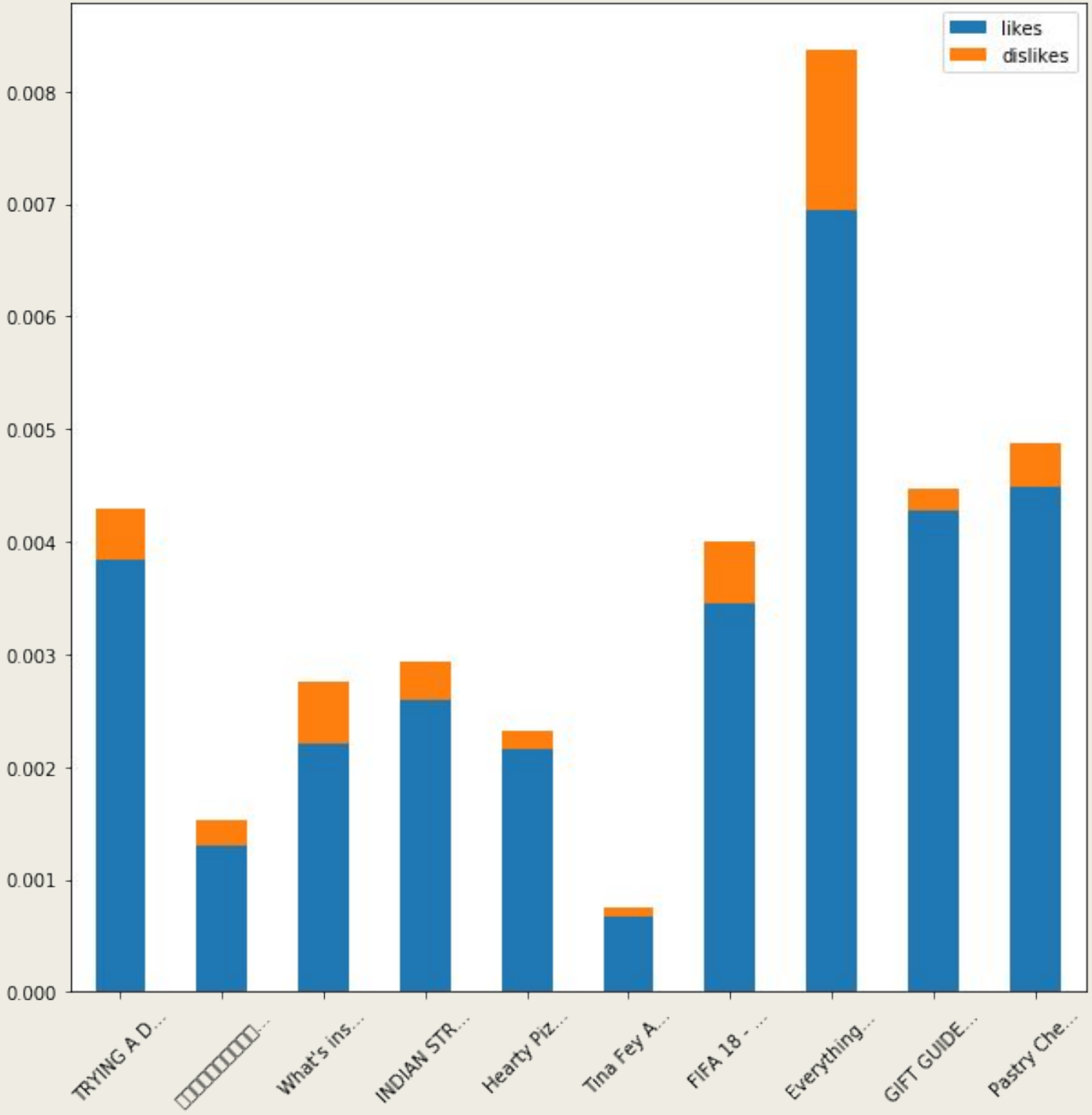
Let’s look at some graphs and what we can infer from them.

The most popular categories are entertainment and music.

This is so because viewers usually tend to hear music repeatedly online instead of downloading them.This substantially increases the video’s views ,thereby making it popular.

Youtubers must focus on making entertainment and music videos in order to gain popularity.

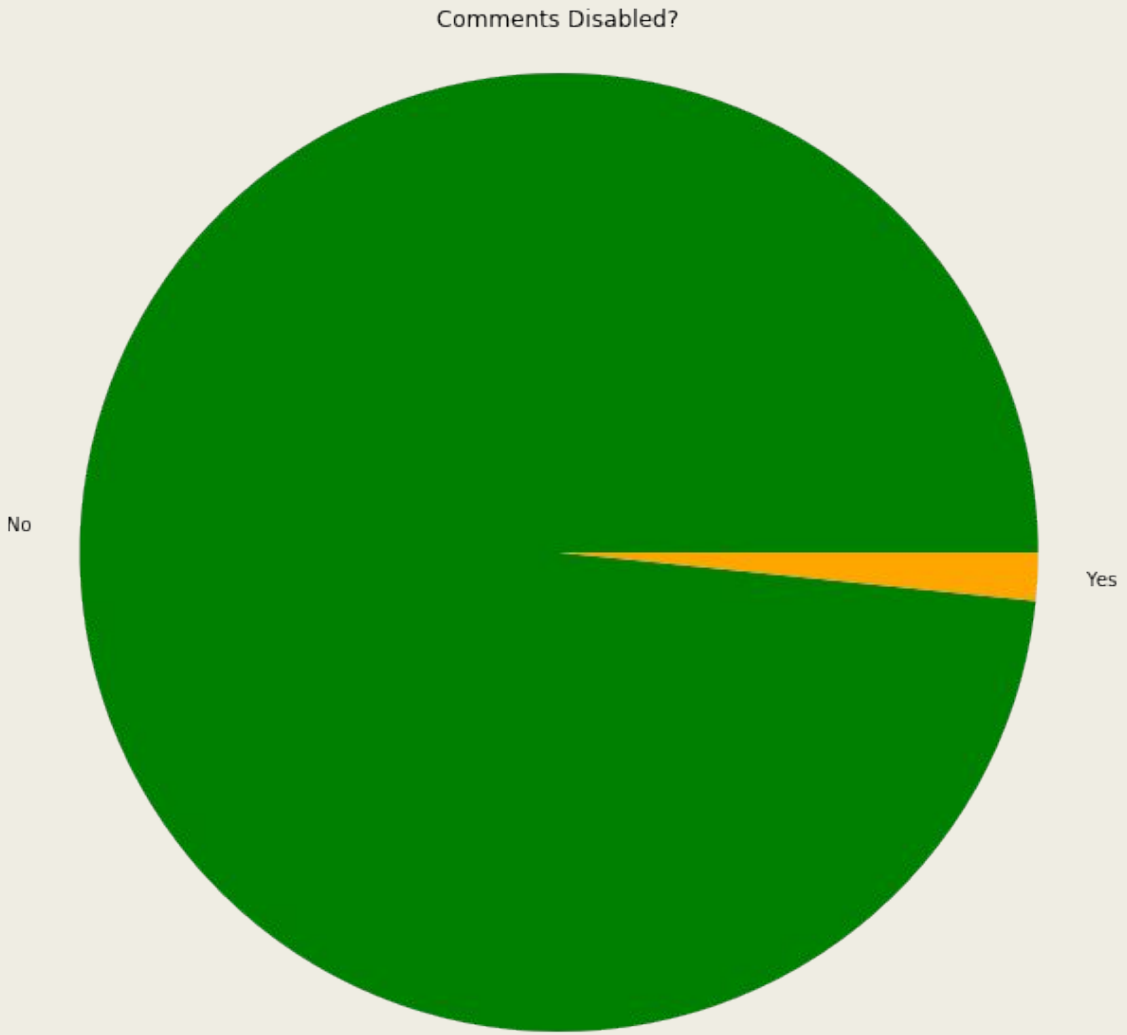


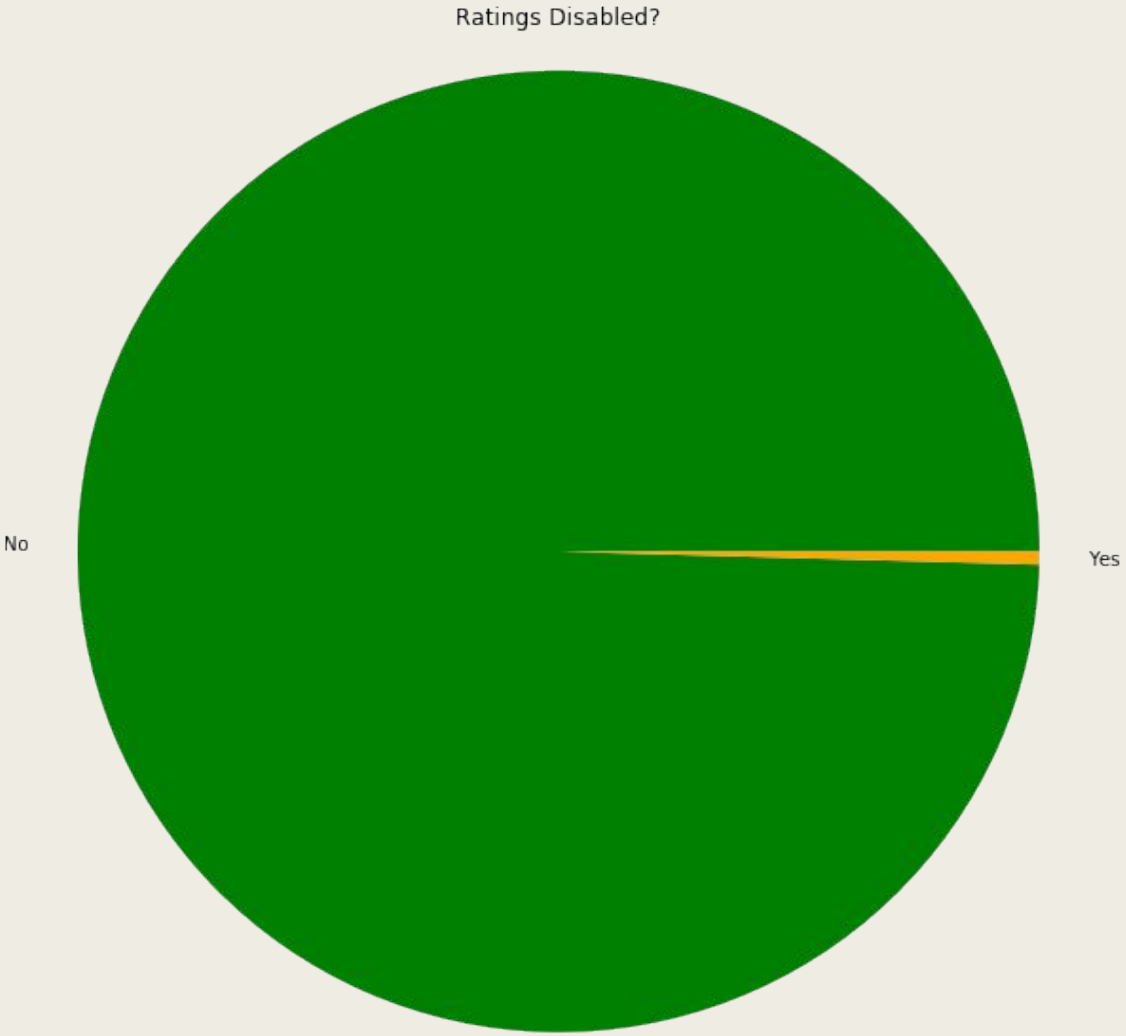

likes
dislikes
0.008
0.007
0.006
0.005
0.004
0.003
0.002
0.001
0.000
TRYING A D...
What's ins...
INDIAN STR...
Hearty Piz...
Tina Fey A...
FIFA 18 - ...
Everything...
GIFT GUIDE...
Pastry Che...

Most of the trending videos have a very small proportion of dislikes when compared to likes.

This is because people recommend videos to others only if they like your video.This increases the popularity of the video tremendously.

Hence Youtubers must create good quality content which viewers like and are likely to recommend to someone.


Comments Disabled?
No
Yes


Ratings Disabled?
No
Yes

Disabling comments or ratings severely impacts the ability of videos to become popular.

Viewers feel that videos with their ratings or comments disabled are a denial of their freedom of speech

Hence Youtubers must not disable the comments or rating.