Milestone Report I

Title:

YouTube video views from the US and UK: Exploratory data analysis, inferential statistics, and classification

Objectives:

The objective of the project is to compare behavioral patterns of youtube viewers in the US and UK. Details of the trending YouTube videos from the US and the UK, for example, views, comments, likes, and dislikes will be analyzed. The project aims to find answers to the following questions: how are the top trending videos different in terms of views, categories, likes and dislikes; whether there exists any seasonal pattern in some specific categories; how are videos categories different in the US and UK.

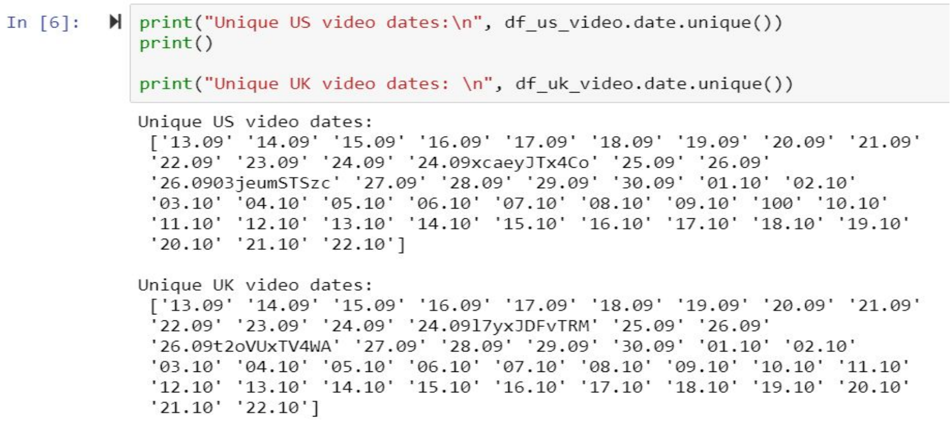
Data description:

The data source is https://www.kaggle.com/datasnaek/youtube. The dataset contains 6 files -

- 3 files on US viewers: UScomments.csv, USvideos.csv, US_category_id.json and
- 3 files on UK data: UKcomments.csv, UKvideos.csv, UK_category_id.json.
- USvidoes.csv has following 11 columns: ['video_id', 'title', 'channel_title', 'category_id', 'tags', 'views', 'likes', 'dislikes', 'comment_total', thumbnail_link', 'date'] where unique 'video_id's are 2364.
- UScomments.csv has 4 columns: ['video_id', 'comment_text', 'likes', 'replies'], where unique 'video_id's are 2266.
- US_category_id.json has [category_id, kind, etag, item_snippet], where unique category_id are 16.

Data cleaning:

1. Check data types, number of null and non-null values: df.info()
2. Changing format of some unusual dates, for example, "26.0903jeumSTSzc" to "26.03".
3. Changing date format from "string" object to "datatime64" object.

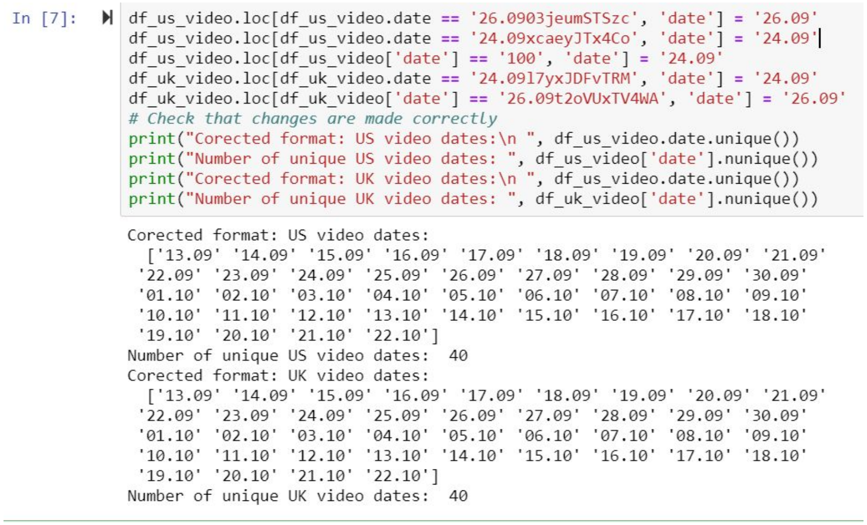


```
In [6]:  print("Unique US video dates:\n", df_us_video.date.unique())
         print()

         print("Unique UK video dates: \n", df_uk_video.date.unique())
```

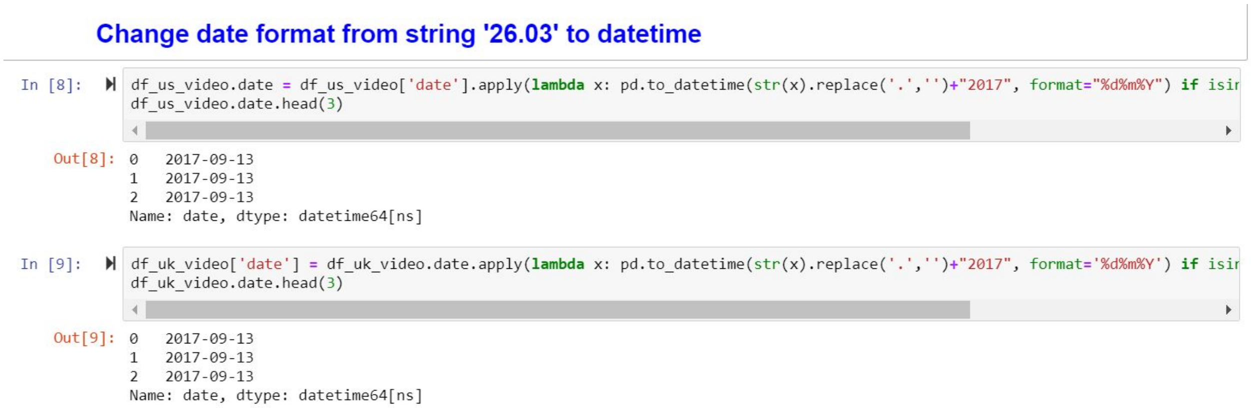
```
Unique US video dates:
 ['13.09' '14.09' '15.09' '16.09' '17.09' '18.09' '19.09' '20.09' '21.09'
 '22.09' '23.09' '24.09' '24.09xcaeyJTx4Co' '25.09' '26.09'
 '26.0903jeumSTSzc' '27.09' '28.09' '29.09' '30.09' '01.10' '02.10'
 '03.10' '04.10' '05.10' '06.10' '07.10' '08.10' '09.10' '100' '10.10'
 '11.10' '12.10' '13.10' '14.10' '15.10' '16.10' '17.10' '18.10' '19.10'
 '20.10' '21.10' '22.10']

Unique UK video dates:
 ['13.09' '14.09' '15.09' '16.09' '17.09' '18.09' '19.09' '20.09' '21.09'
 '22.09' '23.09' '24.09' '24.0917yxJDFvTRM' '25.09' '26.09'
 '26.09t2oVUxTV4WA' '27.09' '28.09' '29.09' '30.09' '01.10' '02.10'
 '03.10' '04.10' '05.10' '06.10' '07.10' '08.10' '09.10' '10.10' '11.10'
 '12.10' '13.10' '14.10' '15.10' '16.10' '17.10' '18.10' '19.10' '20.10'
 '21.10' '22.10']
```



```
In [7]:  df_us_video.loc[df_us_video.date == '26.0903jeumSTSzc', 'date'] = '26.09'
         df_us_video.loc[df_us_video.date == '24.09xcaeyJTx4Co', 'date'] = '24.09'
         df_us_video.loc[df_us_video['date'] == '100', 'date'] = '24.09'
         df_uk_video.loc[df_uk_video.date == '24.09l7yxJDFvTRM', 'date'] = '24.09'
         df_uk_video.loc[df_uk_video['date'] == '26.09t2oVUxTV4WA', 'date'] = '26.09'
         # Check that changes are made correctly
         print("Corected format: US video dates:\n ", df_us_video.date.unique())
         print("Number of unique US video dates: ", df_us_video['date'].nunique())
         print("Corected format: UK video dates:\n ", df_us_video.date.unique())
         print("Number of unique UK video dates: ", df_uk_video['date'].nunique())

         Corected format: US video dates:
           ['13.09' '14.09' '15.09' '16.09' '17.09' '18.09' '19.09' '20.09' '21.09'
          '22.09' '23.09' '24.09' '25.09' '26.09' '27.09' '28.09' '29.09' '30.09'
          '01.10' '02.10' '03.10' '04.10' '05.10' '06.10' '07.10' '08.10' '09.10'
          '10.10' '11.10' '12.10' '13.10' '14.10' '15.10' '16.10' '17.10' '18.10'
          '19.10' '20.10' '21.10' '22.10']
         Number of unique US video dates:  40
         Corected format: UK video dates:
           ['13.09' '14.09' '15.09' '16.09' '17.09' '18.09' '19.09' '20.09' '21.09'
          '22.09' '23.09' '24.09' '25.09' '26.09' '27.09' '28.09' '29.09' '30.09'
          '01.10' '02.10' '03.10' '04.10' '05.10' '06.10' '07.10' '08.10' '09.10'
          '10.10' '11.10' '12.10' '13.10' '14.10' '15.10' '16.10' '17.10' '18.10'
          '19.10' '20.10' '21.10' '22.10']
         Number of unique UK video dates:  40
```



Change date format from string '26.03' to datetime

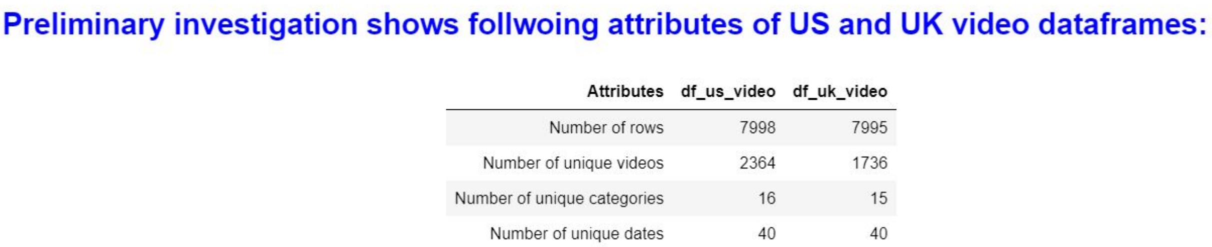
```
In [8]:  df_us_video.date = df_us_video['date'].apply(lambda x: pd.to_datetime(str(x).replace('.','')+"2017", format="%d%m%Y") if isir
         df_us_video.date.head(3)

Out[8]:  0    2017-09-13
         1    2017-09-13
         2    2017-09-13
         Name: date, dtype: datetime64[ns]

In [9]:  df_uk_video['date'] = df_uk_video.date.apply(lambda x: pd.to_datetime(str(x).replace('.','')+"2017", format='%d%m%Y') if isir
         df_uk_video.date.head(3)

Out[9]:  0    2017-09-13
         1    2017-09-13
         2    2017-09-13
         Name: date, dtype: datetime64[ns]
```

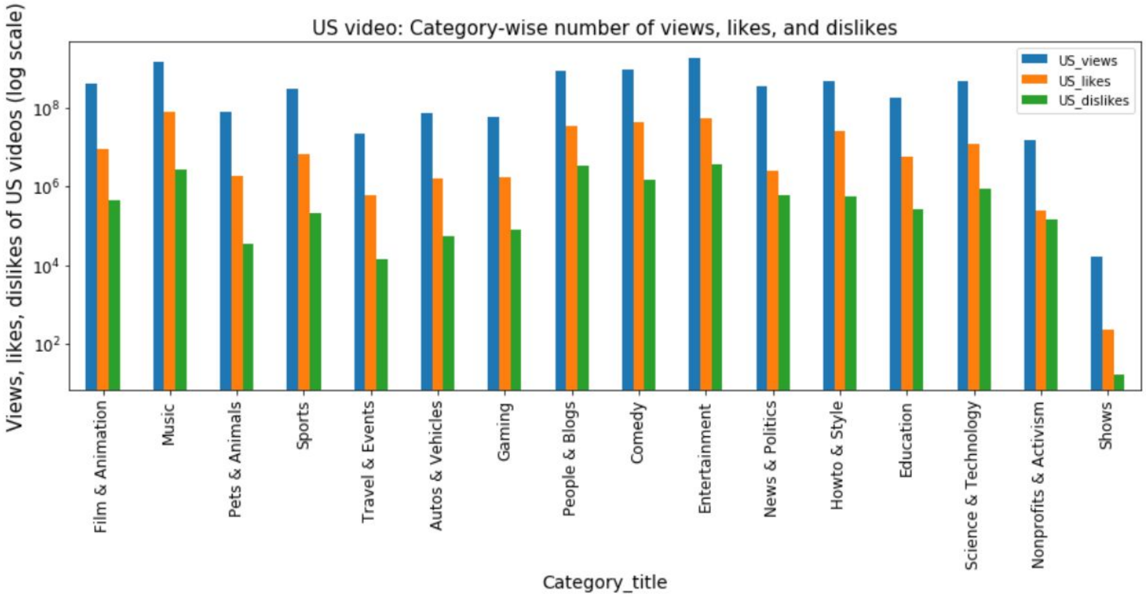
Data Exploration and analysis:



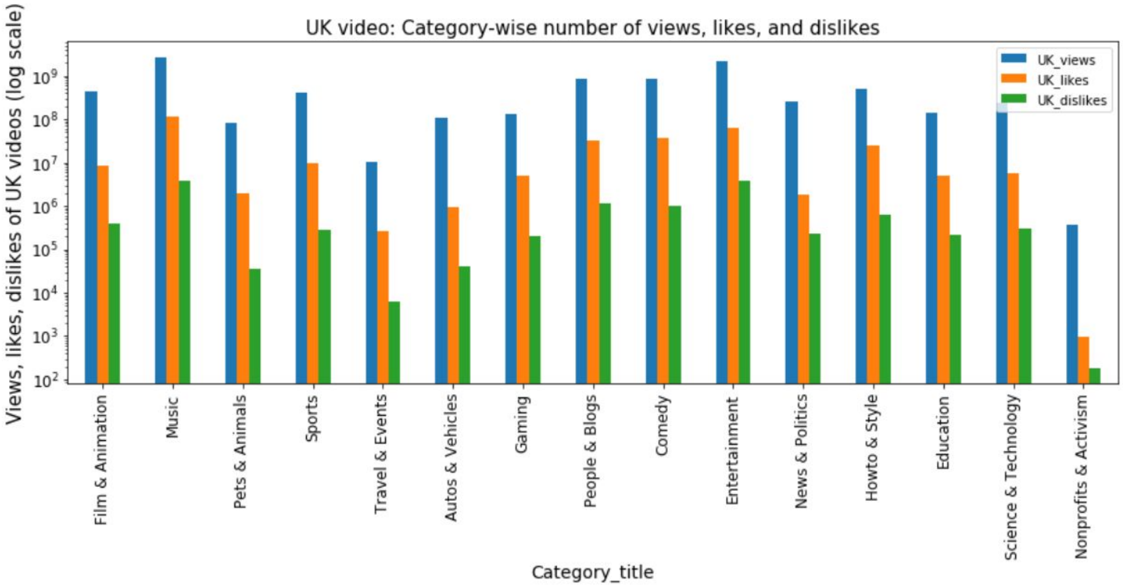
Preliminary investigation shows follwoing attributes of US and UK video dataframes:

Attributes	df_us_video	df_uk_video
Number of rows	7998	7995
Number of unique videos	2364	1736
Number of unique categories	16	15
Number of unique dates	40	40

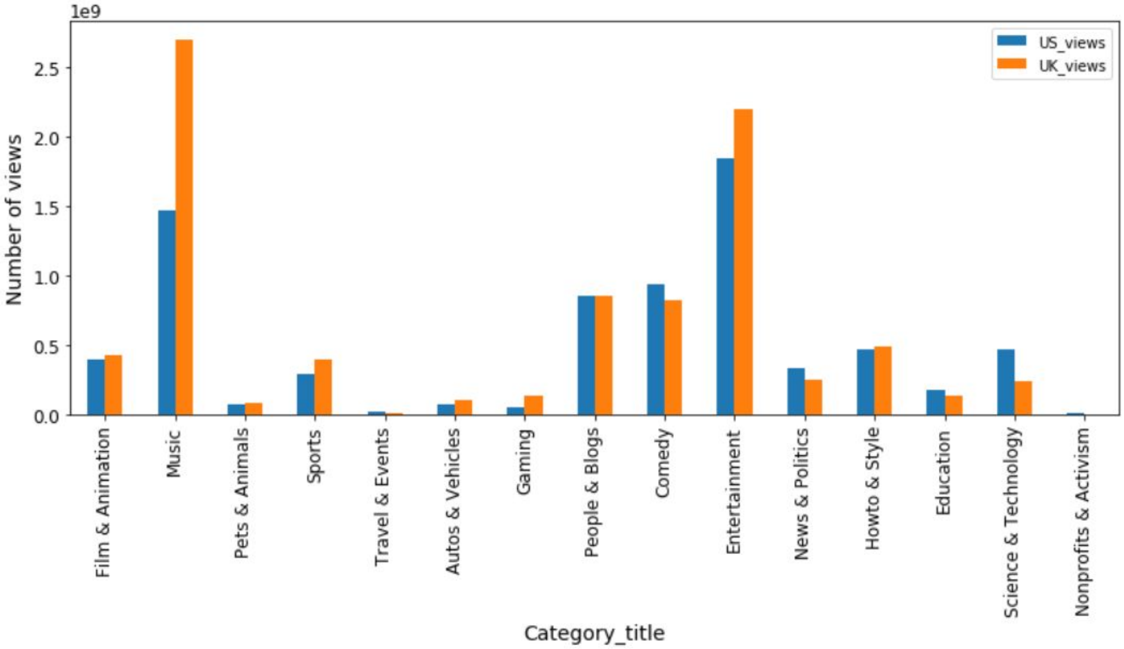
a) Video category-wise number of US views:



b) Video category wise UK views: Top views are Music.



c) Category wise comparison of number youtube videos in the US and UK:



To do next:

a) Explore more data analysis
b) Apply statistical inferences
c) Apply machine learning classification algorithms