Youtube Video Trending Analysis

By Simon,Myles,Deborah,Elisabeth

Introduction

- Our initial questions
- How we decided to split up the work
- Review code
- Show graphs to highlight conclusions
- How we can improve
- What purpose can this information serve?

Initial Questions

1. What were the top ten most viewed **video categories** within each country?
2. What were the top five most viewed, liked, and disliked **YouTube videos** within each country (US, GB, CA)?
3. What were the most viewed, liked, and disliked **YouTube channels** within each country?

Using these evaluations, we will compare the most viewed, liked, and disliked videos, categories, and channels between each country to see similarities/differences between cultural preferences

In order to answer these questions...

- We found the data for daily trending YouTube videos from 2017 to 2018 for multiple countries
- Narrowed it down for english speaking countries (US, Great Britain, Canada)
- Then imported CSV files and narrowed columns for data we wanted
- Used Google API to search for category names - merged this information with each csv file
- Split country data between teammates, but used the same code structure since data was formatted the same way

Snippets Code

- Shows how we took category list from api

- An example of Creating one of the many graphs

Analyzed video views, likes, and dislikes

within each category

In [3]:
```
# get api data to add category names
params = {"part":"snippet","regionCode":"US",
          "key":api_key}
base_url= "https://www.googleapis.com/youtube/v3/videoCategories"
response = requests.get(base_url, params=params).json()

# collect category names to add to a dataframe
id_list = []
title_list = []
for i in response['items']:
    #print(i)
    id_list.append(i['id'])
    title_list.append(i['snippet']['title'])

# Create DF
category_df=pd.DataFrame({"category_id":id_list,"title_list":title_list})
category_df["category_id"]=category_df["category_id"].astype("int64")
category_df.dtypes
```

In [10]:
```
# Create a Pie Chart to show top five viewed categories
Top_5_CategoryID_Views_GB=Views_Category_GB_df.sort_values("Views", ascending=False)
Top5Viewed_GB=Top_5_CategoryID_Views_GB.head(5)

# create pie plot for the top viewed categories in YouTube
# Tells matplotlib to seperate the #1 category section from the others
explode = (0.1, 0, 0, 0, 0)
colors = ["springgreen", "lightseagreen", "dodgerblue", "mediumorchid", "hotpink"]
plt.pie(Top5Viewed_GB["Views"], explode=explode, labels=Top5Viewed_GB["Category ID"], colors = colors, aut
opct="%1.1f%%", shadow=True)
plt.title("Top 5 Viewed YouTube Categories in GB")
```

Code Errors

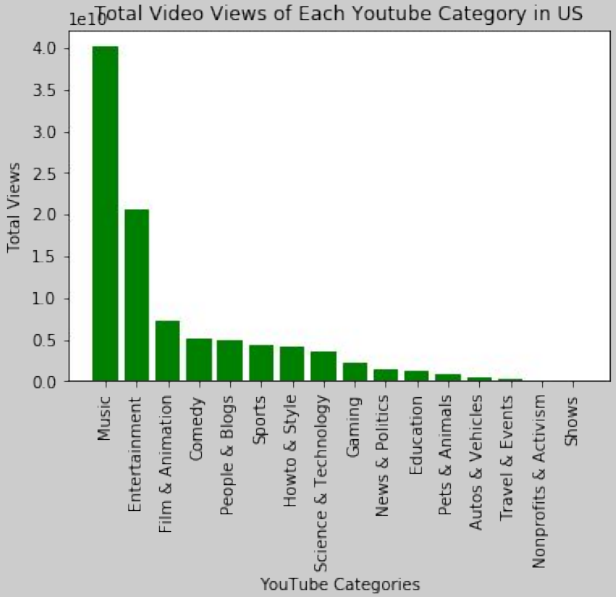
Cleaning Code:

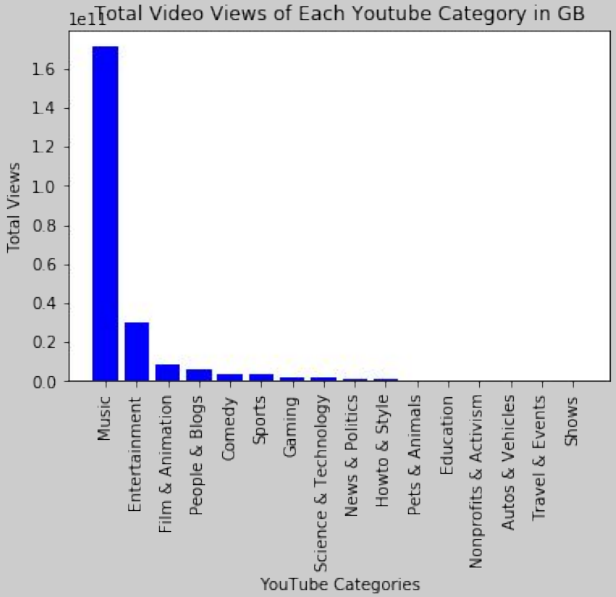
- Drop vs Loc
- Groupby - altogether or dictionary

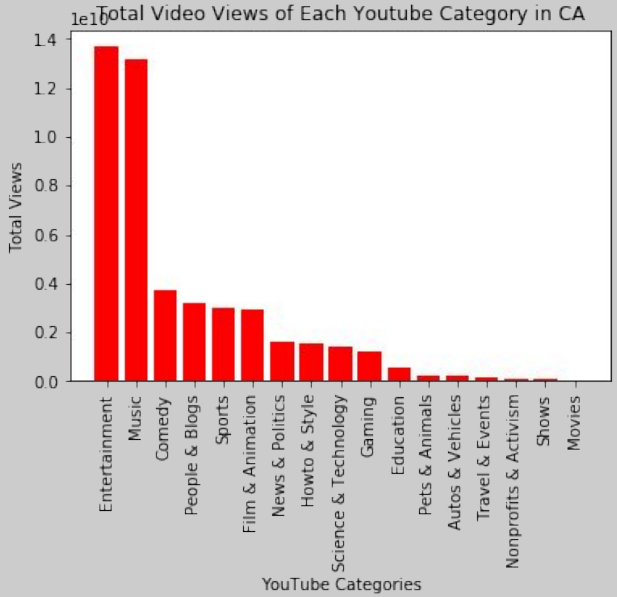
Dataset included top trending videos EACH DAY - videos appeared more than once

Category Analysis

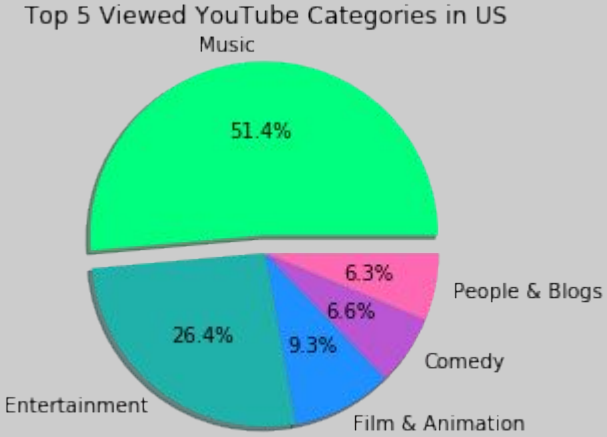
Total Views of Each YouTube Category

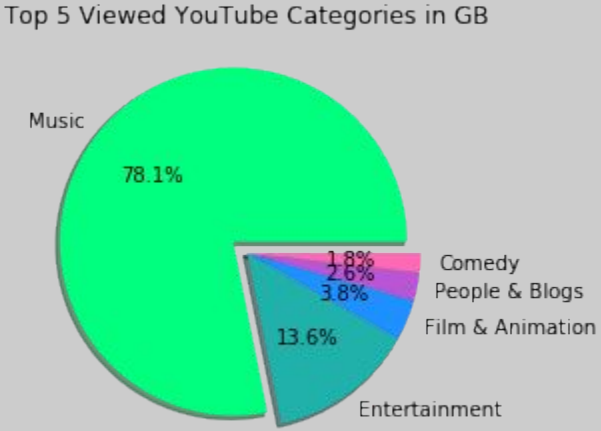


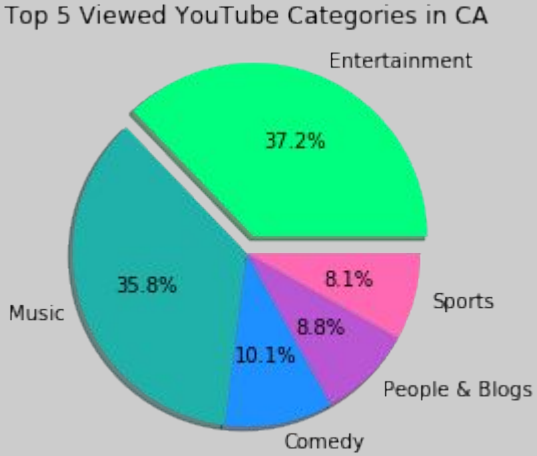




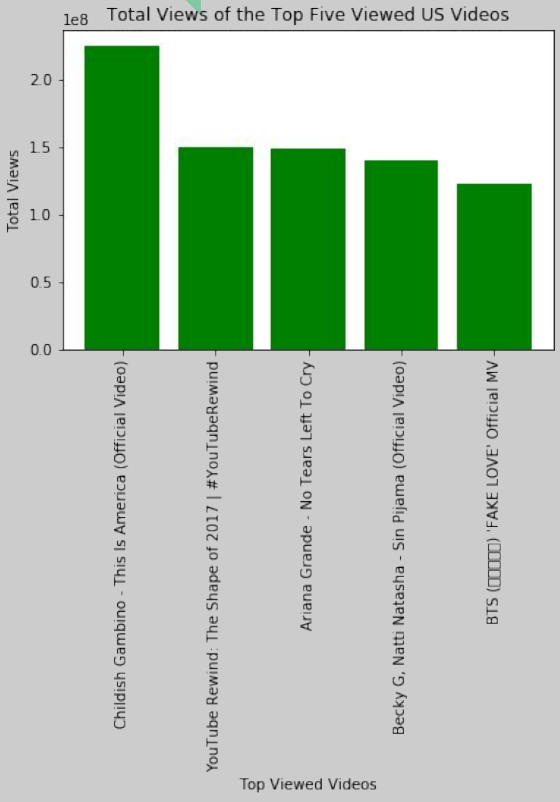
Category Analysis - Top Viewed

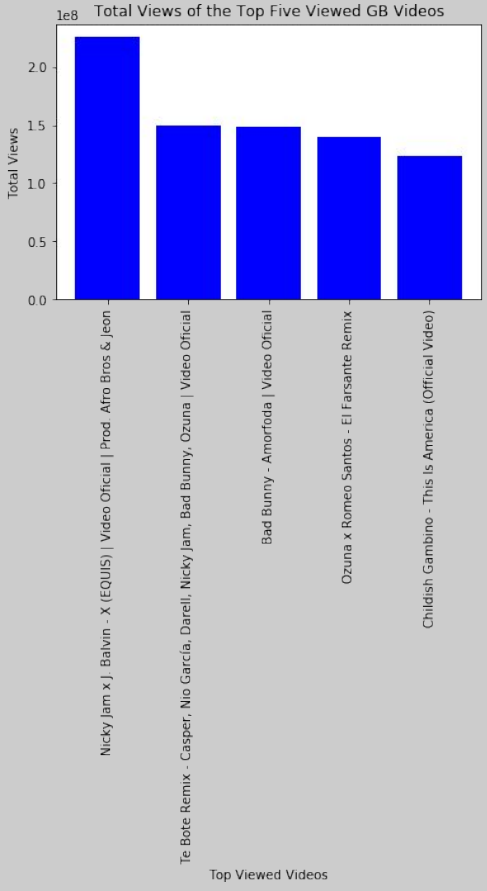


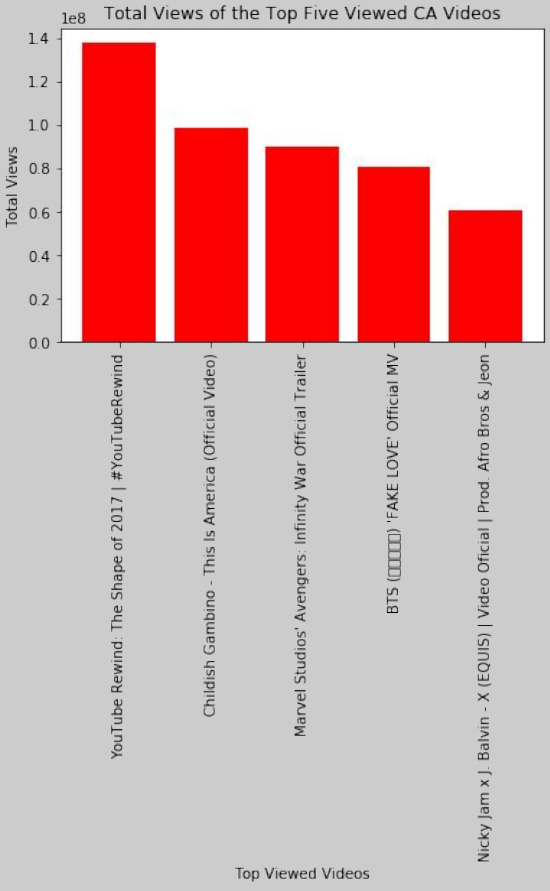




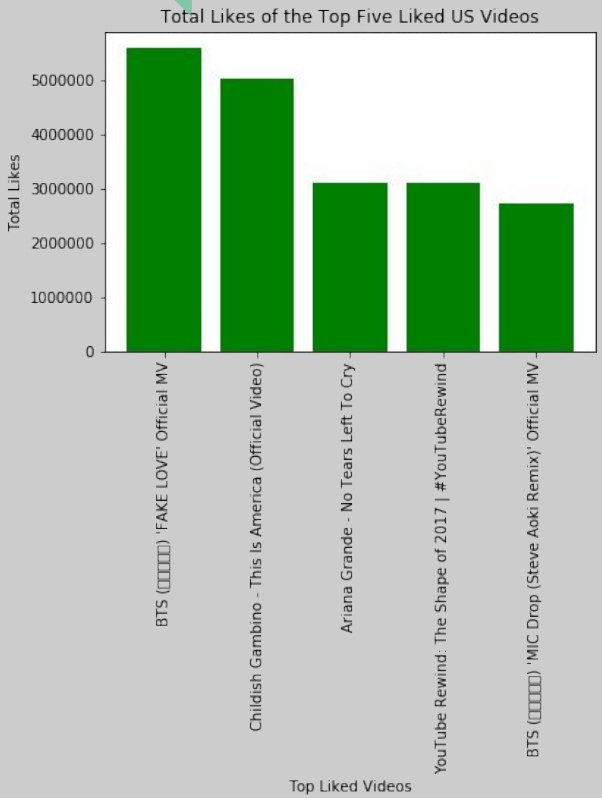
Video Analysis - Most Viewed Videos

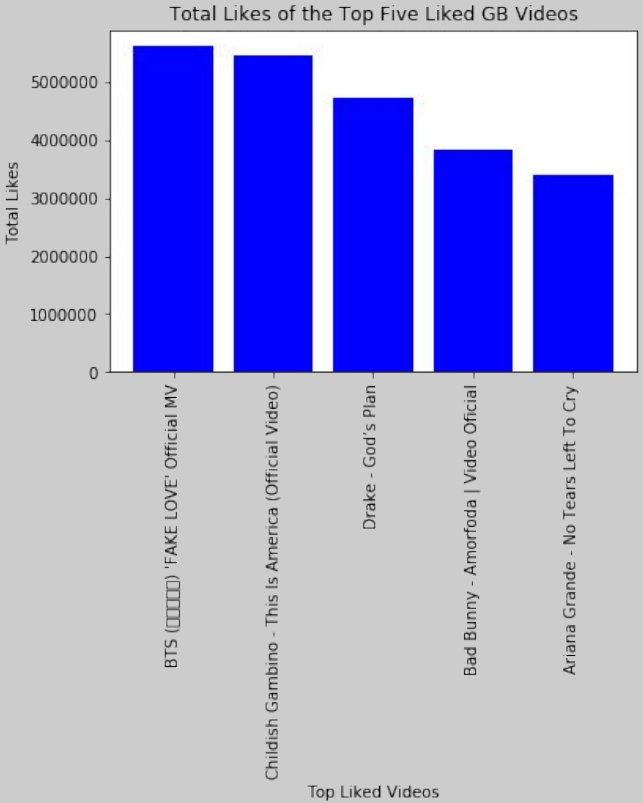


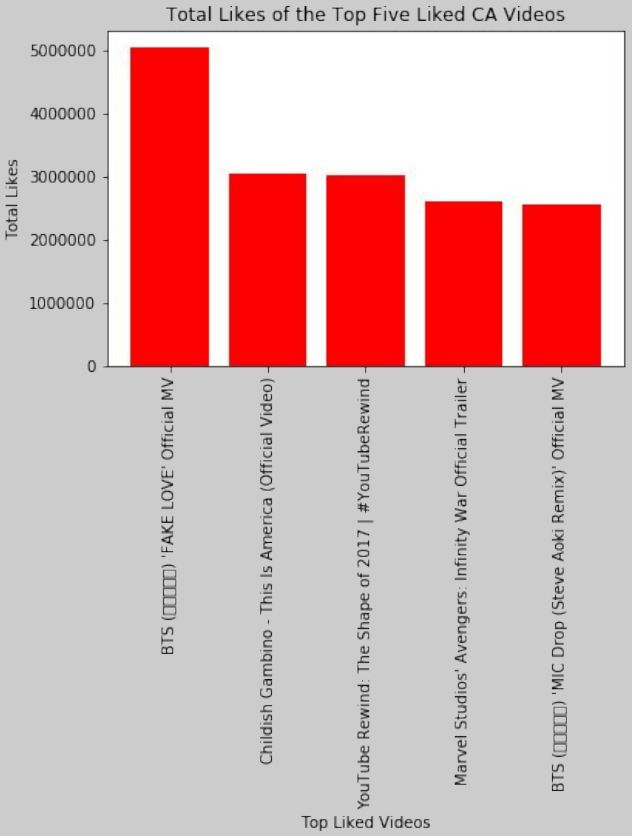




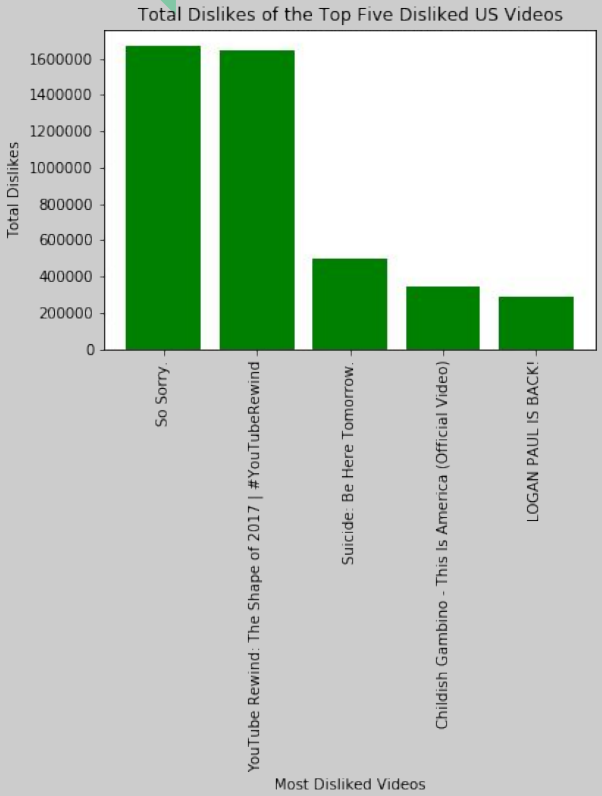
Video Analysis - Most Liked Videos

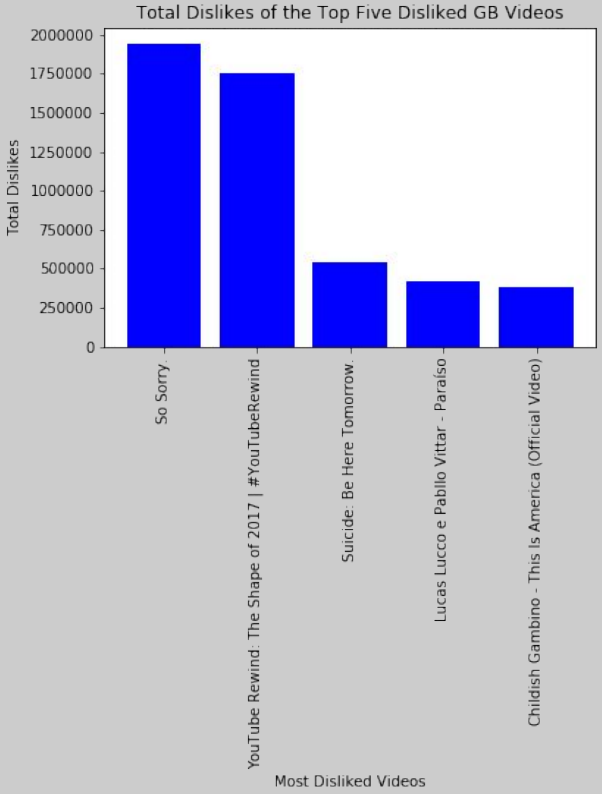


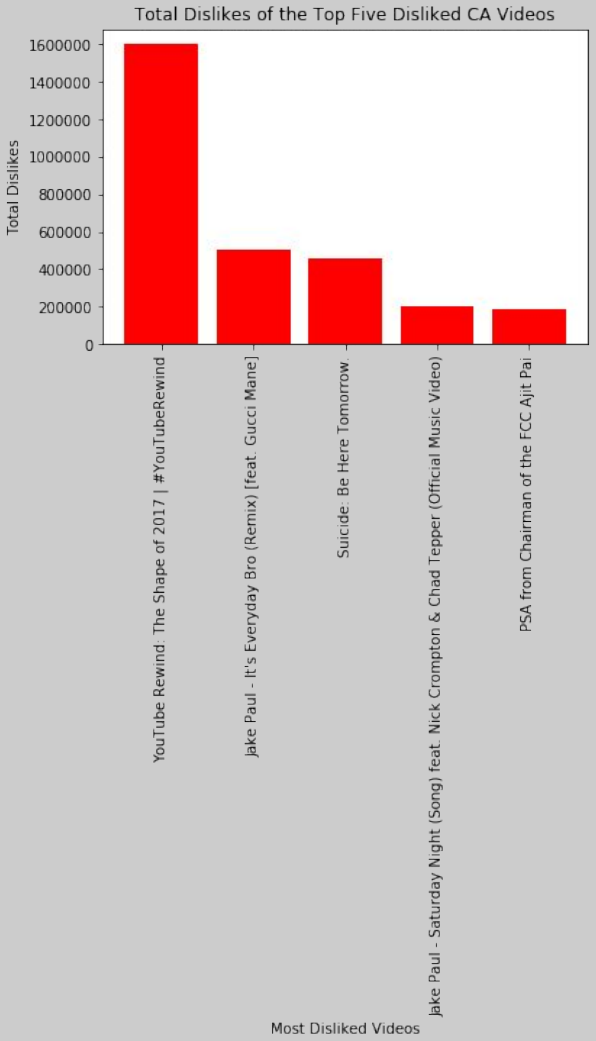




Video Analysis - Most Disliked Videos

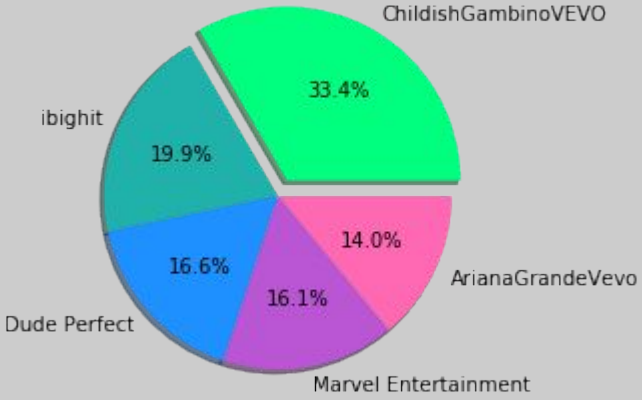




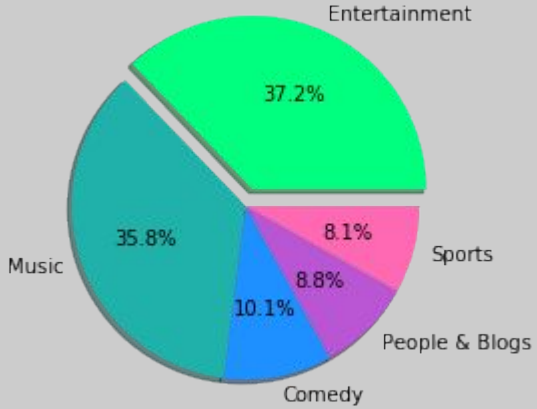


Channel Analysis - Top Viewed Channels

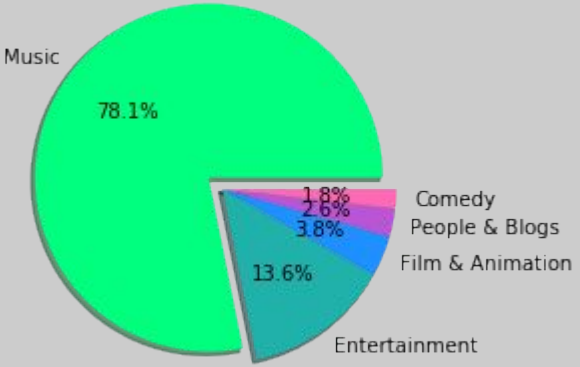
Total top 5 Viewed YouTube Channels in the US



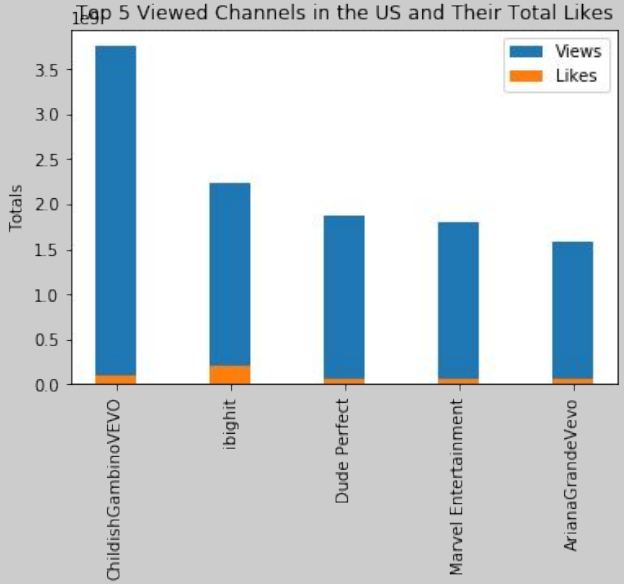
Top 5 Viewed YouTube Categories in CA

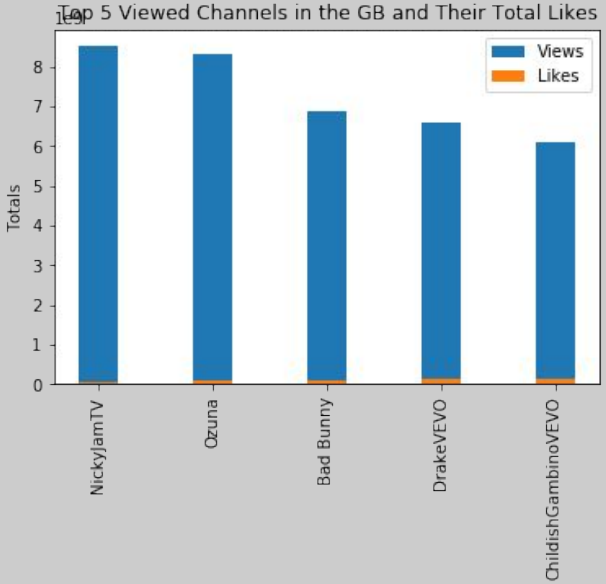


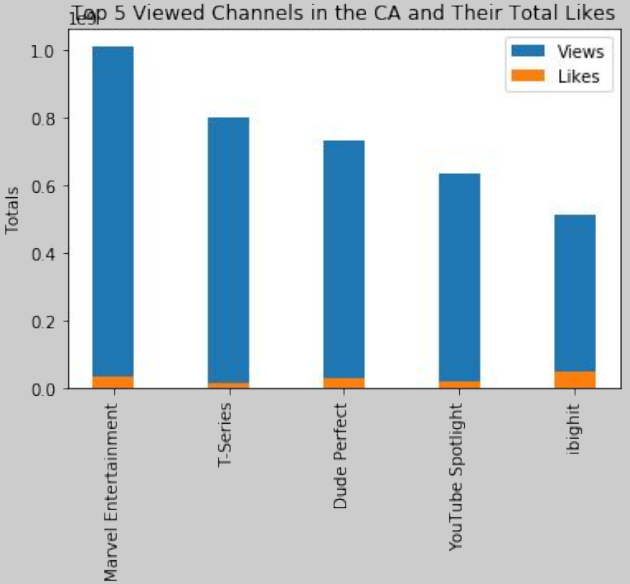
Top 5 Viewed YouTube Categories in GB



Channel Analysis - Top 5 Channels and Their Likes







How can this analysis be used?

- Find trends in videos and categories
- Allows business to see where ads might be beneficial
- Show cultural overlap - if videos are popular in more than one country they reach a wider audience
- Shows the power of Data analytics

What we learned and how we can improve

Specifically with this dataset:

- Improving API calling and application
- Dataset collected top trending videos each day, so videos/channels were repeated within the CSV files
- Likes vs. Dislikes - sorting to see which videos got the least likes did not serve the same purpose as sorting by the most dislikes
- Comparing in pie charts to total amount of views/likes, not just the top 5

Tips to Apply to Next Group Work

In general:

- Spend more time initially with the data
- Begin with a master Jupyter Notebook, then edit individually in each branch
- Communication amongst team members
- Time Management
- Communicate effectively and efficiently
- Work more as a team, not individually - affected by not having class time together
- Be okay with changing your question to match the dataset, or finding a dataset to help answer your question

Sources

- https://github.com/elisabethvirak/Top_Music_Project1
-