**Capstone Project 2**

**What is the Project?**

The Project I wanted to figure out is people are starting to grow up on Youtube while companies are starting to get more invested into youtube for their companies (EX: Ryan Reynolds doing ads for Mint Mobile). Therefore, I want to understand, what does it take to have successful Youtube Videos.

**How did I accomplish this?**

I wanted to start smaller because youtube is a big platform and has numerous videos. I searched for videos launched between 1/29 and ⅖. I limited the pool to videos labelled as gaming videos and tried to stick within the parameters of the US. The success metric I set up is how to get a video to over 100k views, which for channels whom upload 3-4 times a week, is enough to make a living off of. Here is the code I used:

[**Part 1: Finding Videos**](https://colab.research.google.com/drive/1N-4L7BF2bmGyOBoY-KUBmKZnpA57iv2w)

* **Goal:** Find gaming videos.
* **Description:** I use a multitude of search queries to find a lot of videos. I had to do a multitude of search queries because I am limited to 550 videos per search. Here are all the search parameters I looped through
  + Additional Queries: These are words that are popular through Google Trends.
  + Video Durations: I used Short, Medium, and Long to find a variety of length of videos
  + Published After/Published Before: This was used to find a multitude of videos per day during the time period of 1/29 to ⅖.
  + These are the parameters of the search that never change
  + videoCategoryId=20: This never changed and this is the Category for Gaming Videos
  + regionCode=US: This is for videos found in the US
  + relevanceLanguage=en: This is for English Videos
  + maxResults=50: This is the most videos you can find per page
* **Output:** Here is the [csv it leads to](https://drive.google.com/file/d/1hCP06fC4PFFZjzsy1_MNrhHc4s3-wLaK/view?usp=drive_link)

[**Part 2: Finding Video Information**](https://colab.research.google.com/drive/1LvE02ZYSfBU4pEZnCSHDQ6Cu6-bRkoiO)

* **Goal:** Find information on each video
* **Description:** I use this to find information points on Youtube Videos. Here are some examples of what I collected:
  + What Channel the video is from
  + What tags are used on the video
  + What is the description of the video
  + What is the title of the video
* I am trying to collect features for a video to understand what works and what doesn’t for a high viewing youtube video
* **Output:** Here is the [csv it leads to](https://drive.google.com/file/d/1kJ3E83pcbLmw8nn2__r3J_JbnLG4LEkL/view?usp=drive_link)

[**Part 3: Figuring out if a video qualifies as a Short**](https://colab.research.google.com/drive/1ch81_SNXuOLl8jggtuKcWk6kxC972bt-)

* **Goal:** Convert duration to seconds and remove Shorts
* **Description:** Convert duration into seconds and remove anything that is less than 60 seconds. Anything less than 60 seconds counts as a short and has a different advertising model. Therefore, I am only concerned with videos over 60 seconds.
* **Output:** Here is the [csv it leads to](https://drive.google.com/file/d/1d5qPg3Fu974CQlknLQb_kGrWJKORj2nH/view?usp=drive_link)

[**Part 4: Collecting Channel Information for every Video**](https://colab.research.google.com/drive/12msE_P9-N8ycJGNNpQHq8v6TXpF2gyID)

* **Goal:** Find information on the channel for each video
* **Description:** Trying to find info on the channel that should help understand a successful Youtube video. Here are some examples of the info being collected:
  + Country Origin of the Channel
  + Date Channel it is created
  + Is the Channel Made for Kids
  + Channel Subscriber Count
  + Channel Views
* **Output:** Here is the [csv](https://drive.google.com/file/d/10MrSXtNPSAYpOs3BDFdqXVcegw42Lr7_/view?usp=drive_link) it leads to

[**Part 5: Finding the Game Played and the Game Year**](https://colab.research.google.com/drive/1JYgNTDSfrR433hPUAB0iIKr99Ovf0rTc)

* **Goal:** Finding the Game Played and Game Year for the Video
* **Description:** Wanted to understand if the Game Played and the Game Year matters when having a video over 100k
* **Note:** This has a big delay. Ran multiple processes for this to go faster but it did eventually work. Because it took a while, I added a part where it double checks a CSV to see what else needs to be added.
* **Output:** Here is the [csv](https://drive.google.com/file/d/1yD3hD2rcFz27lKkCl00LyL2_UrXZbiub/view?usp=drive_link) it leads to

[**Part 6: Collecting Videos for all channels for Past 3 Months**](https://colab.research.google.com/drive/1Sm_DdXXsKGuiIZvcRBuvZ9NsCbNhGSF0)

* **Goal:** Collecting past videos for each channel to better understand if a channel creator is consistent and see if past views matter
* **Description:** For these past views, I collected numerous fields. The main fields I eventually focus on are the following
  + Likes: Trying to understand if past likes factor into understanding a good video
  + Views: Trying to understand if past
  + Videos: Trying to understand if you post videos consistently
* **Note:** This also took a while but it definitely paid dividends in the end. Also, the file ends up being over 1 GB. I ran the code multiple times because sometimes it was someones first video ever posted. So had to run multiple times to confirm that was the case.
* **Output:** Here is the [csv](https://drive.google.com/file/d/1Rafp0nQd6jGXO-VstrqAgqGMSRviW92a/view?usp=drive_link) it leads to

**Part 7:** [**Dropping duplicates for the activity + Joining the channel info, video info, and activity together**](https://colab.research.google.com/drive/1VatRko0oaJUMT_J_GZ-42VHBHps_eLbW)

* **Goal:** Merging Video Info Captured, Channel Info Captured, Channel Stats Captured, and the Video Information
* **Description:** I wanted to merge all the info into one place so I can then have a CSV where I can add new features where necessary
* **Note:** Nov\_Jan\_Channel\_Stats.csv is a shortened version of past\_channel\_videos. I did this because past\_channel\_videos was 1 gb.
* **Output:** This will be the outputted [csv](https://drive.google.com/file/d/1yD3hD2rcFz27lKkCl00LyL2_UrXZbiub/view?usp=drive_link). It is from another csv because I overwrote it

**Part 8:** [**Creating new Features**](https://colab.research.google.com/drive/1LgEjI2Dg1tPkkPKd35TVGbKxMnyTmdsZ)

* **Goal:** Create new features to better understand how a good youtube video is made. Also filled in null values
* **Description:** Created a variety of new features while also filling in null values. For filling null values, this is what I did:
  + If Channel Views were null, I filled it in with the View Count of the current video
  + If the Channel Country was null, I filled it in with US
  + If Channel Published Date was null, I filled it in with Published at date due to the amount of nulls being low
  + If any past channel stats were null, I filled it in as 0 because they had no videos
  + For Game Played, if it was null, I tried to look at title and tags to fill in the correct game. If it was null, I kept it that way because it usually was a rare game or it actually ended up not being a game. I also made games the same name where necessary
  + For Game Year, I filled in for any games that I already had or that I knew
  + Used Google Trends to find where a trending word was located in the video details. Separated it into Top Terms and Rising Words because Google trends has each
  + Found top games in (<https://steamdb.info>/). I looked at top games sold from Jan 15th to Feb 15th because those were the games that would potentially be popular
  + Found Top Selling Games for 2024 (<https://www.gamespot.com/gallery/2024s-best-selling-games-in-the-us/2900-5106/>) to understand if people played these games,would they do better
  + Found the Avg Views, Likes and Videos for last 3 months to understand how popular the channel was in the past. Also found the percentage of likes for past videos to better understand if the videos have been consistently popular over time
  + Found the hour that the video got launched using EST to see if that matters. Also put hours into different blocks to see if that mattered
  + Put Day of Week to better understand if launching on a certain day mattered
  + I found channel age to see if older channels would do better because they are around longer
  + I wanted to see how a video was launched, so I double checked to know if it used the premier feature or if it was a stream
  + I checked what kind of Social Media links used in the description to see if Promoting their own videos mattered
  + For tags, I checked if they existed and how many tags you would have to see if adding more tags would mean more views and if tags mattered at all
  + For a description, I was trying to understand if that mattered and found if a description mattered or not
  + Finally, I put a number code to everything so it would work for a linear regression model
* **Output:** Here is the [CSV](https://drive.google.com/file/d/1g0sPqeXa5xKzoU-ZPi4EGwlZnIRBpBKh/view?usp=drive_link) it gets outputted to

**Part 9:** [**Seeing which Model is more accurate**](https://colab.research.google.com/drive/1626Og70VpXMB3u1qoLDFWLFfUISNfOcM)

* **Goal:** Trying to see which model is better
* **Description:** I printed out all the models and imported them into the following [Google Sheet](https://docs.google.com/spreadsheets/d/14jlshDILuqAhJBXq2VUmHmBKPk5Fhhhka_UIY8qREiI/edit). Here are the Important Tabs:
  + [**Fields**](https://docs.google.com/spreadsheets/d/14jlshDILuqAhJBXq2VUmHmBKPk5Fhhhka_UIY8qREiI/edit#gid=1323183346)**:** Explaining the name of each field and then categorizing them into why they are important. I use this to ensure I am ideally using all the categories to come up with a good model
  + **Feature Importance (**[**EX**](https://docs.google.com/spreadsheets/d/14jlshDILuqAhJBXq2VUmHmBKPk5Fhhhka_UIY8qREiI/edit#gid=1882613437)**):** Understanding which features were important so I know what to tweak. For “All Features” Model, this was my way to understand which features were important and knowing what to keep. For each version I tried to keep the top features per categories. In my model names, it explains what it was doing
  + [**Compare 3 (Classification)**](https://docs.google.com/spreadsheets/d/14jlshDILuqAhJBXq2VUmHmBKPk5Fhhhka_UIY8qREiI/edit#gid=201532733)**:** For each model, I captured which ones had the best metrics. For Overall Score, it is a weighted average of Precision, Recall, F1 Score, and ROC AUC. I weighted Precision and ROC AUC higher because, if I received a FP, it would be worse than receiving an FN. I want to be conservative and not predict a video will hit 100k and it won’t
* **Note:** I kept Regression metrics as well because I thought I had a regression problem. But, because I want to know if a video hits 100k or not, it is a Classification problem. Hence, Classification metrics were used. I also tweaked the models to better understand the features to better understand the relationship between features.
* **What Model did I choose and Why:** I ended up choosing a model with 10 features. It wasn’t the best performing but it was only worse than a model with 16 features by .11%. Because it would take less computational power, I stuck with a model with 10 features. I chose XGBoost because it performed the best overall. Decision Tree was a close second but XGBoost performed better overall
* **Output:** I put all coefficients/feature importance into CSVs, then would put them in this [Sheet](https://docs.google.com/spreadsheets/d/14jlshDILuqAhJBXq2VUmHmBKPk5Fhhhka_UIY8qREiI/edit#gid=201532733) to better understand what models were better and why

**Part 10:** [**Figuring out what hyperparameters to Choose**](https://colab.research.google.com/drive/1tcEK17Yaj1XuAFqriF-2cwLytVHfBvEr)

* **Goal:** Trying to see which parameters to use. Then based on best parameters, show the final model I will use
* **Description:** The script would loop through about 6300 permutations to see the best parameters. All parameters would be stored into a CSV which I would then look into and calculate what the best parameters would be to use. Similar to what model I chose, I weighted Precision, and ROC AUC as 30% and Recall + F1 score as 20%. Based on the weightage I chose what parameters I would be using
* **Output:** Here is the way I decided what parameters to use. It improved every metric with ROC AUC seeing the biggest jump. The parameters found in [Row 2](https://docs.google.com/spreadsheets/d/191bXjNVnoC3doJX52vNUTWv-h73TRw9uXsFzsy0HjqM/edit#gid=758268270) are the ones I ended up using