# Notes

* Add one model including ratios, and one model only ratios for discussion

# Brainstorming Questions

* Discuss change in engagement over viewership of life of video
* Start with 10 groups and go from there
  + Do this with
* Think about what you want to classify to include for silhouette
* Dig into weirdos on kclass
* The anatomy of a YouTube Music Video
  + GB / LR for: NLP, base model, binary classification
  + Sub categories and their features
  + Groups with highest engagement

# Model Story

Phase 1: Learning API

Phase 2: Tuning

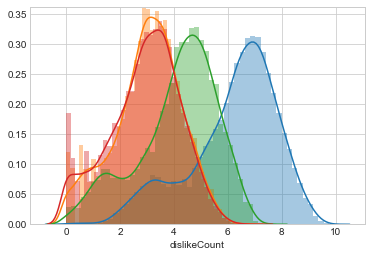
Phase 3: EDA

Phase 4: First Features & Models

Phase 5: New Structure

# Descriptive

## Like / Dislike / Comment / View Count Interaction



Key:

* Blue - views
* Green - Likes
* Red - Comments
* Orange - Dislikes

Table 1: Distribution Values (log)

|  |  |  |  |
| --- | --- | --- | --- |
| **Log** | Mode | Bump | Mode / Bump |
| Views | 6.96 | 3.25 | 2.14 |
| Likes | 4.40 | 1.60 | 2.75 |
| Dislikes | 2.94 |  |  |
| Comments | 3.13 |  |  |

Table 2: Distribution Values (actual)

|  |  |  |  |
| --- | --- | --- | --- |
| **Actual** | Mode | Bump | Mode / Bump |
| Views | 9.08E+06 | 1.78E+03 | 5105 |
| Likes | 2.53E+04 | 3.98E+01 | 635 |
| Dislikes | 8.71E+02 |  |  |
| Comments | 1.34E+03 |  |  |

Table 3: Ratio of Log of Counts at fit Mode (row divided by column)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Views | Likes | Dislikes | Comments |
| Views |  | 1.58 | 2.37 | 2.22 |
| Likes | 0.63 |  | 1.50 | 1.41 |
| Dislikes | 0.42 | 0.67 |  | 0.94 |
| Comments | 0.45 | 0.71 | 1.06 |  |

Dislikes and comments are almost perfectly matched, within 5% in the log (50% actual) of eachother. Dislikes and comments would seem to reflect similar levels of engagement with a video. If you’re wondering about whether a full YouTube comment really takes as much involvement as pressing dislike, I invite you to scroll through the comments of a random video (Sample: ‘Who is watching in 2018??? lol :P).

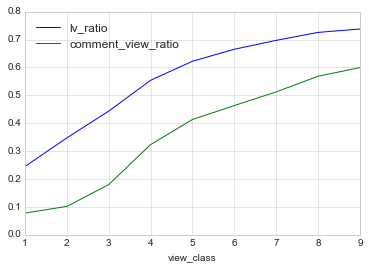
One thing to note about this shape similarity is that it shows in log-space, meaning there is a diminishing degree of engagement for high view count videos. This is going to make analysis very difficult. Interestingly, the relationship falls off even in log space, although not more than as a peculiarity of the math (6/4 is a bigger number than 6+2 / 4+2).

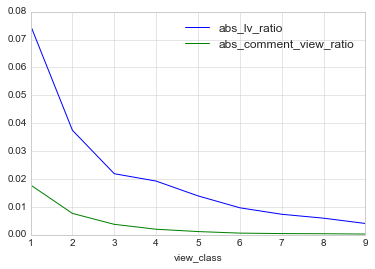
Another peculiarity is the left-most bump in each of the curves. The bump ratio holds to the same description as the mode ratio. I strongly suspect that this is a feature of the YouTube search algorithm, which touches on how this data was collected. I searched for each individual letter of the alphabet, then returned results by relevance. Nothing is relevant to the letter ‘d’, for example, so my guess is that view count is about 5,000 times more important than everything else is the search algorithm (ratio of mode views to bump height views).

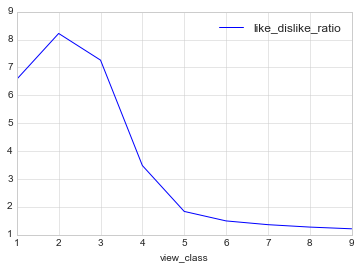
I also suspect that what this means is there are actually two different sets of data in this set which could possibly throw off the analysis. This strongly suggests setting some threshold - e.g. log of views greater than 4.1 - at which to perform the analysis.

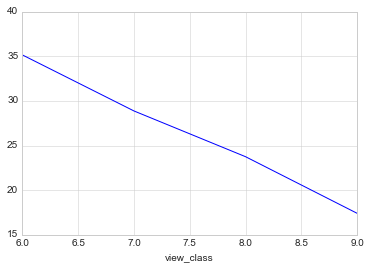
I would love to know if the engagement features described here are time dependent, but for now I’m just going to have to assume that people are disinterested in wildly popular videos from the outset.

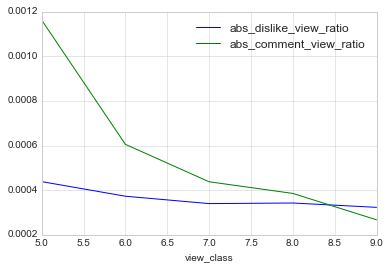
## Engagement









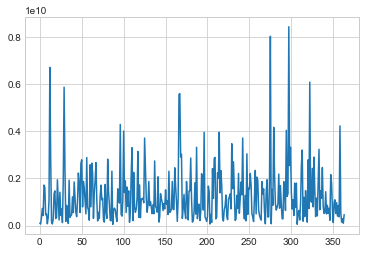


## Publish Date Features

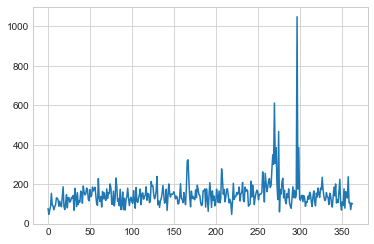
There is no reason there should be any signal here, and yet…

The importance of examining the right variable:

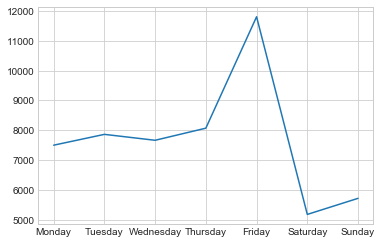
Noise:



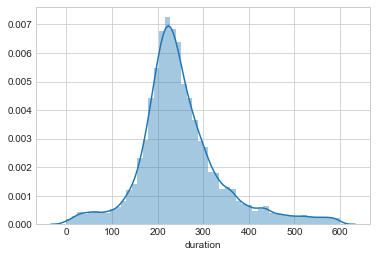
Signal:



Release on a Friday:



## Duration features, or lack thereof…



## Text Analysis

Sentiment analysis and wordcount done for each of these.

### Title



Sentiment score:

Discussion:

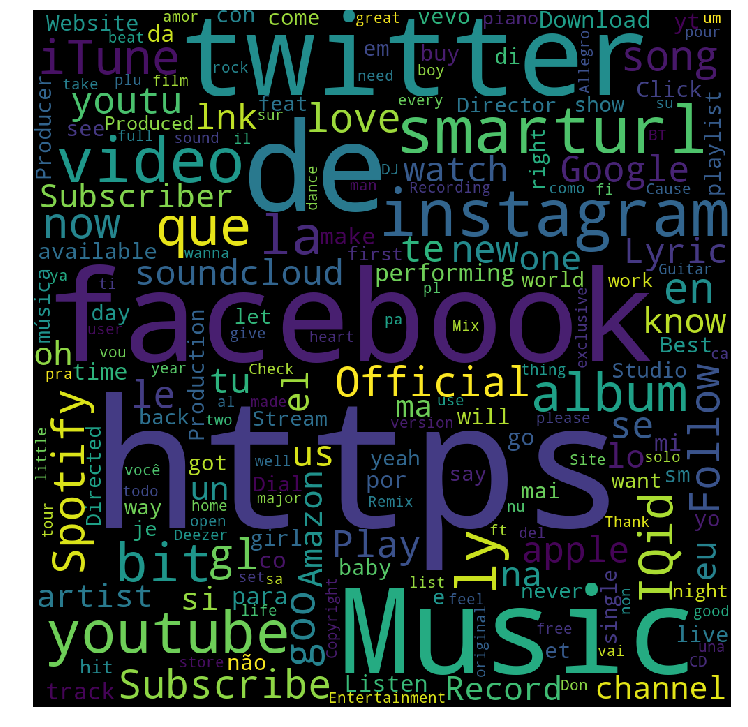
### Tags



Sentiment score:

Discussion:

### Description



Sentiment score:

Discussion:

## Feature by Feature Analysis:

Date Time

General: publishedAt

Friday (is\_friday)

Is weekend (is\_weekend)

week number (week)

Month day (month\_day)

Month (month)

2 digit year number (year)

Ignoring overall day count - possible overfit to search algorithm

Description

Sentiment (description\_sentiment)

Overall word count (description\_wordcount)

CVEC (./data/engineered\_data/description\_wordvec.csv)

Tags

Sentiment (tags\_sentiment)

Num tags (tags\_wordcount)

CVEC (./data/engineered\_data/tag\_wordvec.csv)

Title

Sentiment (title\_sentiment)

Word count (title\_wordcount)

feat / ft. (title\_featuring)

Letter Count (intitle\_ + a,b,c,...)

String length (title\_length)

Which letters

Length

CVEC (./data/engineered\_data/title\_wordvec.csv)

Ratios

likes / views (lv\_ratio)

comments / view (comment\_view\_ratio)

likes / dislikes (like\_dislike\_ratio)

Measures (includes log)

commentCount

dislikeCount

favoriteCount

likeCount:

viewCount: view\_log

Other

Has content rating restriction (contentRating)

Has any region Restriction (regionRestriction)

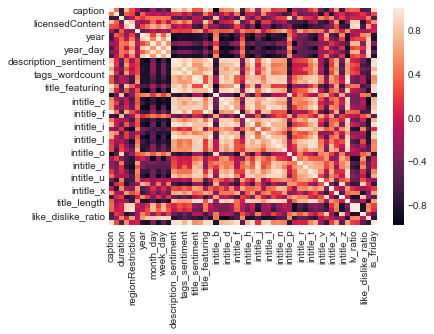
Has captioned text (caption)

Content has approved license (licensedContent)

Duration of video (duration)

High definition 1 or std (0) (definition)

## After feature engineering



# Predictive

## Issues

A key

## Classification

## Regression

Vectorizer alone was .25 regression, but adding to existing models generally made them worse (except for GradientBoost)