Fall 2018

BUAN 6356: Business Analytics with R

A Report On

YouTube Trending Video Metadata Analysis

Submitted by

Drishya Pillai – DXP180003

Mohit Jain- MXJ170006

Sayali Paseband- SSP180006

Sushant Wavhal- SXW180022

Vaibhav Shrivastava- VXS180016

Under the Guidance of

Prof. Sourav Chatterjee

TABLE OF CONTENTS

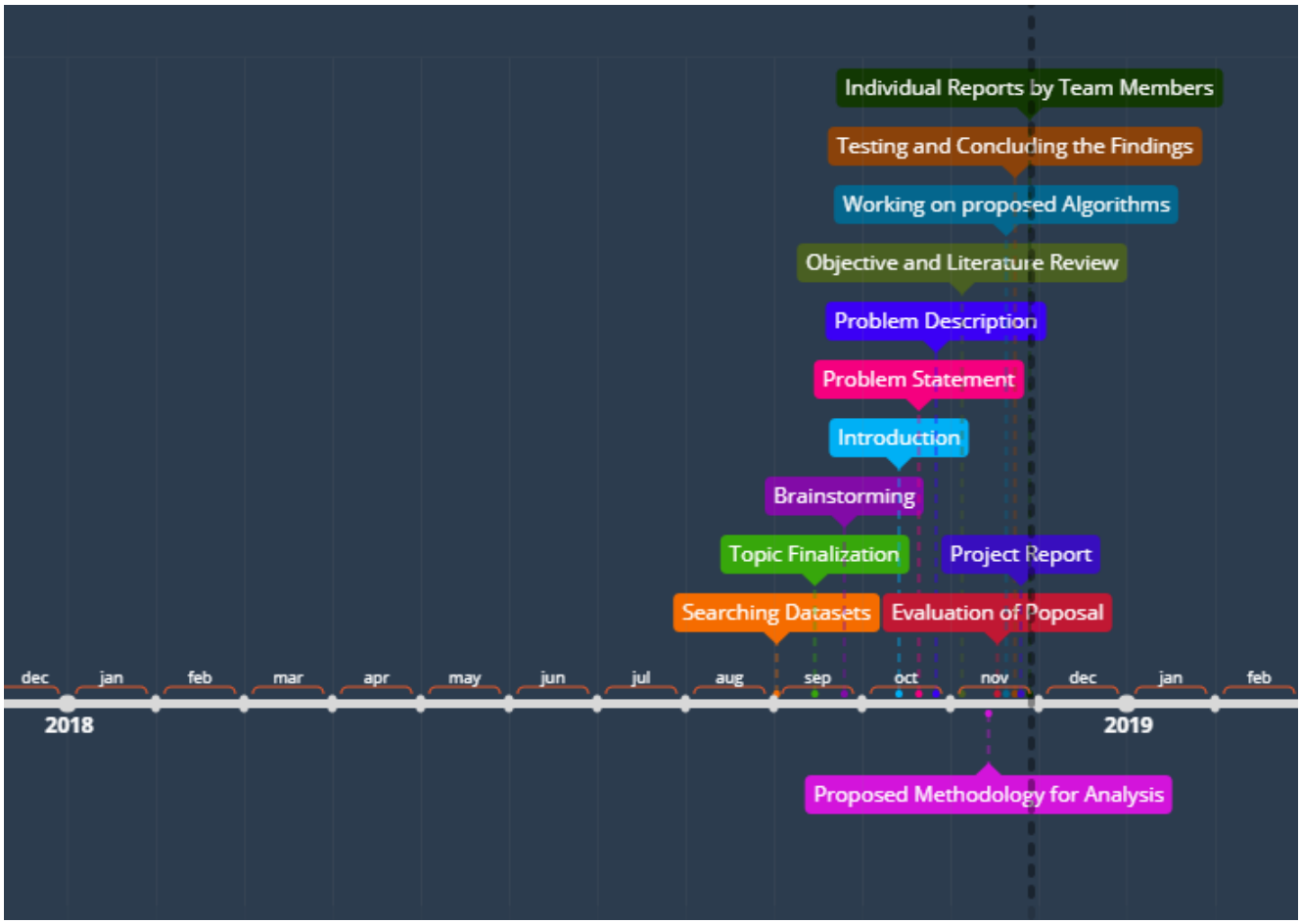
INTRODUCTION

YouTube, the free video sharing website, has emerged as one of the most versatile platforms for sharing knowledge and information. It has become one of the most primary source of entertainments in the present arena and is being browsed by millions of people every second across the globe. On a larger and more useful scale, it has catered to the limpid expression of thoughts and creative ideas by people in a myriad of fields. Although YouTube serves as an excellent platform for sharing a wide variety of videos, one important factor that determines the impact a video makes is its popularity. Since there are almost uncountable videos shared here every day, the popularity of a video depends on a large number of factors. One such important aspect is the 'Trending' section which brings together number of videos from different countries under a single heading. Through this project we have tried to focus on this parameter and what factors contribute towards bringing a video into this very section. During our brainstorming sessions, we tried to analyze the existing algorithms that might be governing the functioning of the 'Trending' section. We also tried to come up with some factors which we felt might be vital based on our thought process and the prior knowhow of the subject matter. Based on a schematic list of all possible factors that the team could come up with, we followed a systematic approach for data analysis utilizing all the classroom techniques that we have imbibed.

MOTIVATION

Considering the time frame for the project, the team explored the 16 most vital factors that we believe shall be analyzed thoroughly for understanding the existing functioning of the 'Trending' section of YouTube. We have paid attention to the utilization of these factors for running different predictive algorithms. All the results of running these algorithms have been properly documented. The results have been verified and appropriate charts have been drawn so as to facilitate a visual perception of the findings. Instead of conducting our analysis taking the globe as a whole, we have focused on a large dataset for one single country which is 'Canada'. We have utilized a diverse dataset of more than forty thousand videos launched on YouTube in Canada and analyzed each of these videos based on multiple factors. We have also tried to come up with some additional parameters that are derived from the existing ones, in order to map any correlation that we may see between them. Holistically, we sum up all our findings to conclude how a video gains popularity on YouTube.

TIMELINE



The project started off with the team meeting at the beginning of the semester to decide on the dataset that we shall use for our project. We browsed through several websites on the internet for the same. Post the finalization of the dataset we conducted brainstorming sessions to decide a methodology that we will follow for our project. Following the same we first came up with a brief introduction of the project, its problem statement and problem description. We presented the same ideas during the summary session we had in the classroom after the 7th week. After this we decided on the objective of the project and analyzed what algorithms could be useful for helping us evaluate the same. We made a systematic list of the steps that we shall follow and algorithms we shall follow for creation of the model for the project. We tried utilizing the maximum knowledge that we got during our classroom sessions and came up with some significant findings. This was followed by the creation of the project report for which we first decided on an outline and then we elaborated each topic that the outline covered. The final step was the creation of individual reports based on the experience of team members in the project.

LITERATURE REVIEW

The 'Trending' section on YouTube has always been a topic of discussion among everyone due to the variety of videos it exhibits, some of those even gaining popularity to the extent of a controversy. There has not been one single factor affecting the popularity of a video that makes into the 'Trending' section that could be distinctly seen unanimously. For instance, at times a highly likes video has made it into this section and at the others a highly disliked one. Sometimes you see a video in a language widely spoken while at times a video made in the least popular language across the world makes it to this sensational section.

In a recent incident, controversies and criticism rose to the peaks when one of the top trending videos, displayed for hundreds of millions of viewers in the United States, alleged that a Marjory Stoneman Douglas High School student was a crisis actor. An another example could be of the thousands of videos posted recently about the school shooting that occurred last week in Parkland, Florida, many of which feature students advocating for gun control legislation.

Hence, quite obviously there have been discussions and various thought processes about what algorithm YouTube follows for managing this section. YouTube has not addressed clearly what algorithm it follows that determines what's authoritative and what's not when it is put into its 'Trending' section. This has been seen as one of the biggest challenges that YouTube has faced as one of the most loved platforms by all sections of the society. It has often battled questions over its lack of oversight when some controversial video has become popular by making it into this section. YouTube has been asked more information and clarity about the working of its 'Trending' section.

This was a reason why we were particularly interested in taking up this as the topic for our project because it would actually mean trying to discover the undiscovered. We would not just analyze the existing factors that are commonly being analyzed but also try to figure out the unseen factors which we could perceive via extensive brainstorming.

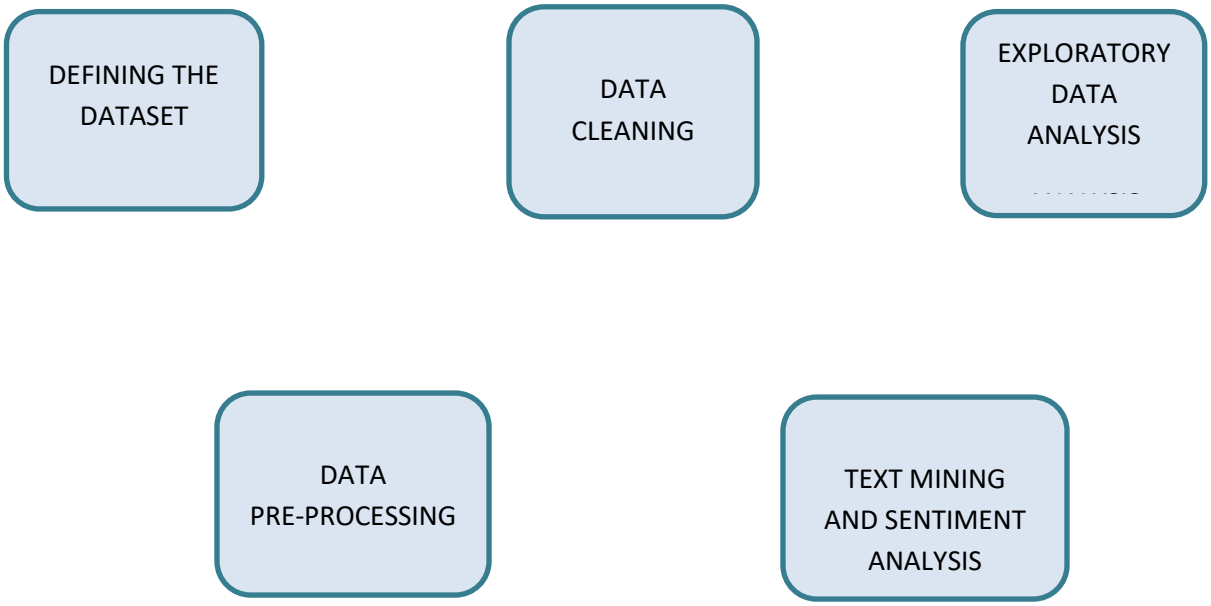
PROBLEM STATEMENT

The 'Trending' section of YouTube has always been popular because it is one of the highly followed sections of this video-sharing platform owing to the variety of videos it displays from various genres. There is supposedly some algorithm which determines whether a video hosted on YouTube makes it to this section or not. There can be a variety of factors that this algorithm takes into consideration before deciding upon the final video. There have been several methods employed for finding out how the algorithm works. Even YouTube has never addressed a single approach towards it's trending algorithm. We have taken up the challenging problem of combating with this issue and analyzing several attributes of over 40,000 YouTube videos in Canada that have made their ways into the 'Trending' section not just once but sometimes several times over a span of few hours to couple of days. Our goal is to find out which attributes contribute the most to how fast a video rises and makes it into the 'Trending' section.

PROBLEM DESCRIPTION

- The dataset chosen contains YouTube video metadata for over 40,000 YouTube videos across Canada.
- We have refrained from the use of data that is difficult to process such as video, image, audio and large text documents. Before building theories from the data we have tried to understand the key attributes of the data.
- Based on our analysis, we are trying to find out how videos on YouTube grow, trend and gain popularity.
- We can sum up our motive with exploiting this dataset as follows-
    - ➢ Clean the data to focus only on key parameters and relevant entries.
    - ➢ Assess the impact of different parameters on the 'trending' of a video
    - ➢ Run the algorithms and develop predictive models
    - ➢ Conclude the findings

DATA COLLECTION AND ANALYSIS



1) ABOUT THE DATA SET

- The dataset we use in our project is entitled 'CAvideos.csv'. The source for this dataset is - https://www.kaggle.com/yanpapadakis/trending-youtube-video-metadata-analysis/data
- This is a collection of the top trending videos on YouTube for the country of Canada.
- It maintains a daily record of the top trending YouTube videos along with a variety of factors including user interactions such as number of views, shares, comments and likes.
- This dataset includes several months (and counting) of data on daily trending YouTube videos in Canada with up to 200 listed trending videos per day.
- There are total 40881 observations with 16 different variables present in the dataset. Each row corresponds to a video identified by a video ID. There may be multiple rows with the same video ID because the dataset tries to capture the number of times a video has trended in a given number of days.
- Below is a brief description of the variables present in the dataset-
  - **video_id** – Uniquely identifies a video
  - **trending_date** – The date when a video makes it to the 'Trending' section
  - **title** – Title of video
  - **channel_title** – Title of channel that publishes the video
  - **category_id** – Uniquely identifies the genre of the video
  - **publish_time** – The time the video is published by a channel
  - **tags** – String of the predominant words related to the video
  - **views** – Number of views for a video
  - **likes** – Count of the number of user likes for a video
  - **dislikes** – Count of the number of user dislikes for a video
  - **comment_count** – Count of user comments on a video
  - **thumbnail_link** – Link to the video
  - **comments_disabled** – Logical variable that denotes if comments have been enabled/disabled for a particular video
  - **ratings_disabled** – Logical variable that denotes if ratings have been enabled/disabled for a particular video
  - **video_error_or_removed** – Logical variable that indicates if the video has been removed or has some error
  - **description** – Description of the video

- As we go further with the analysis, we shall utilize a combination of few of the above variables to process some more variables to get an idea of the correlation between different variables.

2) DATA CLEANING

Before we began our actual analysis with the dataset, we made sure that the data is clean one, meaning it contains all useful values that would help us carry on the analysis smoothly rather than creating any sort of glitches.

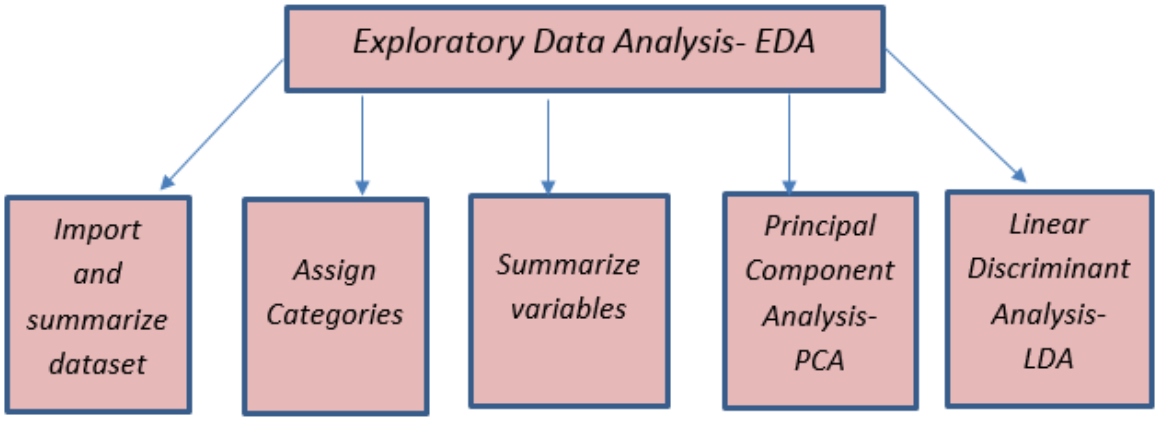
For this we paid attention on the following-

- Understanding the key attributes so that we may recognize the importance and unimportant ones
- Analyzing unique counts. Since there are multiple records for a single video in the dataset, we made sure that there are no junk values and multiple records for a single video make sense in terms of giving meaningful information.
- Looking for missing values. We made sure to omit any missing or null values prior to any sort of analysis.
- Identifying outliers to evaluate any extreme values in the data

3) EXPLORATORY DATA ANALYSIS (EDA)

As a part of EDA, we tried to assess the quality of the data by spotting any possible errors which may have risen due to data collection, data handling or data transfer during the initial stages of the data analysis. We try to spot variables that might require any additional processing that would be carried out in the eventual stages of the analysis post EDA. Apart from this, we evaluate if data requires binning or sampling while we conduct visualizations to depict all our findings as the dataset we utilize is a large one.

So we can sum up EDA into the following 3 tasks-



a) Importing the dataset and creating a quick summary to assess the variables present

	video_id	trending_date	title	channel_title	category_id	publish_time
1	n1WpP7iowLc	17.14.11	Eminem - Walk On Water (Audio) ft. BeyoncÃ©	EminemVEVO	10	2017-11-10T17:00:03.000Z
2	0dBIkQ4Mz1M	17.14.11	PLUSH - Bad Unboxing Fan Mail	iDubbbzTV	23	2017-11-13T17:00:00.000Z
3	5qpjK5DgCt4	17.14.11	Racist Superman | Rudy Mancuso, King Bach & Lele Pons	Rudy Mancuso	23	2017-11-12T19:05:24.000Z
4	d380meD0W0M	17.14.11	I Dare You: GOING BALD!?	nigahiga	24	2017-11-12T18:01:41.000Z
5	2Vv-BfVoq4g	17.14.11	Ed Sheeran - Perfect (Official Music Video)	Ed Sheeran	10	2017-11-09T11:04:14.000Z

tags	views	likes	dislikes	comment_count
Eminem|"Walk"|"On"|"Water"|"Aftermath/Shady/Interscop...	17158579	787425	43420	125882
plush|"bad unboxing"|"unboxing"|"fan mail"|"idubbbztv"...	1014651	127794	1688	13030
racist superman|"rudy"|"mancuso"|"king"|"bach"|"racist"|"...	3191434	146035	5339	8181
ryan|"higa"|"higatv"|"nigahiga"|"i dare you"|"idy"|"rhpc"|"...	2095828	132239	1989	17518
edsheeran|"ed sheeran"|"acoustic"|"live"|"cover"|"official"|...	33523622	1634130	21082	85067

thumbnail_link	comments_disabled	ratings_disabled	video_error_or_removed	description
https://i.ytimg.com/vi/n1WpP7iowLc/default.jpg	FALSE	FALSE	FALSE	Eminem's new track Walk on Water ft. BeyoncÃ© is avail...
https://i.ytimg.com/vi/0dBIkQ4Mz1M/default.jpg	FALSE	FALSE	FALSE	STill got a lot of packages. Probably will last for another ...
https://i.ytimg.com/vi/5qpjK5DgCt4/default.jpg	FALSE	FALSE	FALSE	WATCH MY PREVIOUS VIDEO â–¶ \n\nSUBSCRIBE â–º htt...
https://i.ytimg.com/vi/d380meD0W0M/default.jpg	FALSE	FALSE	FALSE	I know it's been a while since we did this show, but we're...
https://i.ytimg.com/vi/2Vv-BfVoq4g/default.jpg	FALSE	FALSE	FALSE	ðŸŽ§: https://ad.gt/yt-perfect\nðŸ’°: https://atlanti.cr/yt-al...

The above figures depict the parts of a single dataframe that we obtain when we import our dataset in R Studio. Clearly, we can see a large variety of variables and the values they hold for different videos. We shall gradually assess the impact of each of these variables in prediction and modelling.

b) Assigning Categories to the videos

In the dataset, we observe that the categories are depicted by their unique IDs and not their names. Instead, we were given a JSON file with all the categories as values in the JSON variable.

When converted to a table, the JSON would appear like below-

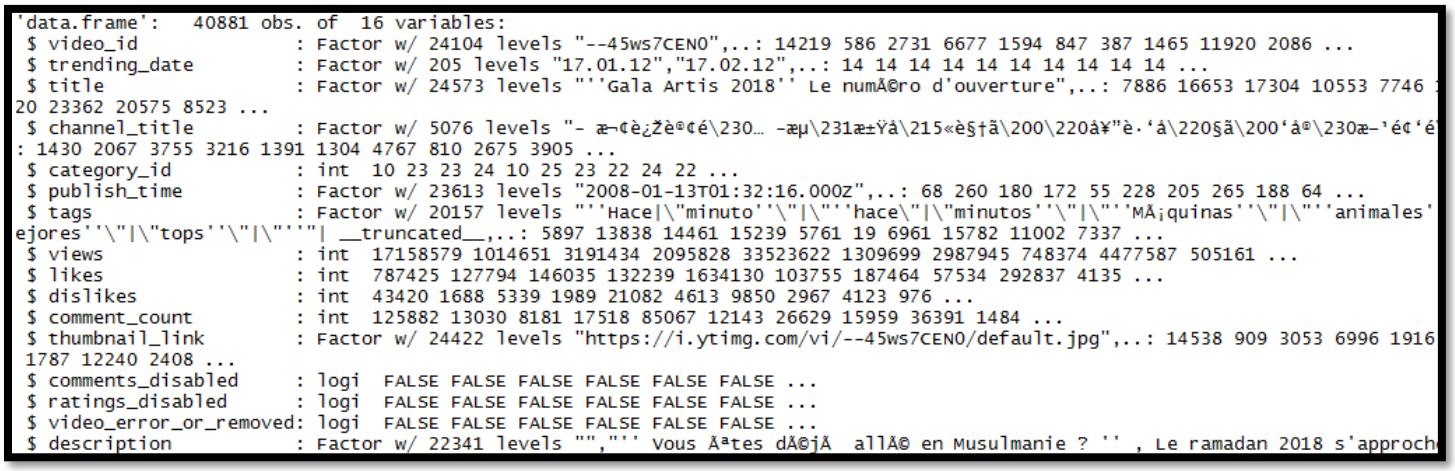
kind	etag	id	snippet.channelId	snippet.title	snippet.assignable
NA	*NA*	*NA*	*NA*	*NA*	*NA*
NA	*NA*	*NA*	*NA*	*NA*	*NA*
youtube#videoCategory	"ld9biNPKjAjgjV7EZ4EKeEGrhao...	1	UCBR8-60-B28hp2BmDPdntcQ	Film & Animation	TRUE

The category names associated with each ID were stored in 'snippet.title' value of the JSON variable. For example, the above table depicts that a video with category ID 1 belongs to the 'Film & Animation' genre.

So, we extracted each of the values from the category and assigned it as levels for the category IDs present in the category_id column of our dataset.

c) Summary of Variables

The next step in EDA was to analyze what is the type of the variables present in the dataset, whether numerical or categorical. For this, we create a summary of all the variables as below-



```
'data.frame':   40881 obs. of  16 variables:
 $ video_id              : Factor w/ 24104 levels "--45ws7CEN0",..: 14219 586 2731 6677 1594 847 387 1465 11920 2086 ...
 $ trending_date         : Factor w/ 205 levels "17.01.12","17.02.12",..: 14 14 14 14 14 14 14 14 14 14 ...
 $ title                 : Factor w/ 24573 levels "''Gala Artis 2018'' Le numÃ©ro d'ouverture",..: 7886 16653 17304 10553 7746 1
20 23362 20575 8523 ...
 $ channel_title         : Factor w/ 5076 levels "- æ¬¢è¿Žè®¢é\230… -æµ\231æ±Ÿå\215«è§†ã\200\220å¥"è·'å\220§ã\200'å®\230æ–'é¢'é
: 1430 2067 3755 3216 1391 1304 4767 810 2675 3905 ...
 $ category_id           : int  10 23 23 24 10 25 23 22 24 22 ...
 $ publish_time          : Factor w/ 23613 levels "2008-01-13T01:32:16.000Z",..: 68 260 180 172 55 228 205 265 188 64 ...
 $ tags                  : Factor w/ 20157 levels "''Hace|\"minuto''\"|\"''hace\"|\"minutos''\"|\"''MÃ¡quinas''\"|\"''animales'
ejores''\"|\"tops''\"|\"''"| __truncated__,..: 5897 13838 14461 15239 5761 19 6961 15782 11002 7337 ...
 $ views                 : int  17158579 1014651 3191434 2095828 33523622 1309699 2987945 748374 4477587 505161 ...
 $ likes                 : int  787425 127794 146035 132239 1634130 103755 187464 57534 292837 4135 ...
 $ dislikes              : int  43420 1688 5339 1989 21082 4613 9850 2967 4123 976 ...
 $ comment_count         : int  125882 13030 8181 17518 85067 12143 26629 15959 36391 1484 ...
 $ thumbnail_link        : Factor w/ 24422 levels "https://i.ytimg.com/vi/--45ws7CEN0/default.jpg",..: 14538 909 3053 6996 1916
 1787 12240 2408 ...
 $ comments_disabled     : logi  FALSE FALSE FALSE FALSE FALSE FALSE ...
 $ ratings_disabled      : logi  FALSE FALSE FALSE FALSE FALSE FALSE ...
 $ video_error_or_removed: logi  FALSE FALSE FALSE FALSE FALSE FALSE ...
 $ description           : Factor w/ 22341 levels "","''' Vous Ãªtes dÃ©jÃ  allÃ© en Musulmanie ? '' , Le ramadan 2018 s'approch
```

d) Principal Component Analysis- PCA

The dimensionality of a model is the number of predictors or input variables used by the model. The curse of dimensionality is the affliction caused by adding variables to multivariate data models.Removing variables that are strongly correlated to others is useful for avoiding multicollinearity problems that can arise in various models.Multicollinearity is the presence of two or more predictors sharing the same linear relationship within the outcome variable.

Principal components analysis (PCA) is a useful method for dimension reduction,especially when the number of variables is large. PCA is especially valuable when we have subsets of measurements that are measured on the same scale and are highly correlated.

Principal component analysis (PCA) is a statistical procedure that uses an orthogonal transformation to convert a set of observations of possibly correlated variables (entities each of which takes on various

numerical values) into a set of values of linearly uncorrelated variables called principal components. If there are n observations with p variables, then the number of distinct principal components is {min(n-1,p)} { min(n-1,p)}. This transformation is defined in such a way that the first principal component has the largest possible variance.

```
Importance of components:
                          PC1    PC2    PC3    PC4     PC5     PC6     PC7
Standard deviation     1.7731 1.0252 1.0004 0.9281 0.72567 0.56475 0.31167
Proportion of Variance 0.4491 0.1502 0.1430 0.1231 0.07523 0.04556 0.01388
Cumulative Proportion  0.4491 0.5993 0.7423 0.8653 0.94056 0.98612 1.00000
> pcs$rot
                         PC1          PC2          PC3         PC4         PC5          PC6          PC7
views           0.505920990 -0.01286294 -0.002557336  0.06420079  0.12124326  0.7149023514 -0.4625634175
likes           0.512652520 -0.10928878  0.026809972 -0.06531745  0.44900835  0.0306058800  0.7195227519
dislikes        0.399164540  0.30719229 -0.013712556 -0.29073425 -0.78174818  0.0260450621  0.2231088711
comment_count   0.493428240  0.04025346  0.023114556 -0.24045515  0.20175862 -0.6624337242 -0.4658650961
frequency       0.279574922 -0.23262637 -0.065462844  0.86179309 -0.26870812 -0.2191124154  0.0231476537
dislike_percent -0.016133358  0.90815507 -0.134215568  0.30988387  0.24393482 -0.0216917170  0.0312654872
days_to_trend   0.002272926 -0.11420338 -0.988055244 -0.10272028  0.01210632  0.0005853715  0.0009453222
```

The idea in PCA is to find a linear combination of the two variables that contains most, even if not all, of the information, so that this new variable can replace the two original variables.

We have removed non numeric variables for PCA as only numeric variables are required for regression model. We ignored column number 1,2,3,4,5,10,11,12,13,14,17,18 from PCA.

Interpretation of PCA output:

The first principal component is dominated by likes and second component is dominated by dislike percentage. To capture 94.06% variation in the data, we need 5 principal components.

e) Linear Discriminant Analysis- LDA

The Linear Discriminant Analysis (LDA) is a well-established machine learning technique and classification method for predicting categories. LDA's main advantages, compared to other classification algorithms such as neural networks and random forests, are that the model is interpretable and that prediction is not difficult. LDA is frequently used as a dimensionality reduction technique for pattern recognition or classification and machine learning.

The LDA algorithm starts by finding directions that maximize the separation between classes, then use these directions to predict the class of individuals. These directions, called linear discriminants, are a linear combinations of predictor variables.

LDA assumes that predictors are normally distributed (Gaussian distribution) and that the different classes have class-specific means and equal variance/covariance.

Before performing LDA, consider:

- Inspecting the univariate distributions of each variable and make sure that they are normally distribute. If not, you can transform them using log and root for exponential distributions and Box-Cox for skewed distributions.
- Removing outliers from your data and standardize the variables to make their scale comparable.

The linear discriminant analysis can be easily computed using the function lda() [MASS package].

```
Coefficients of linear discriminants:
                        LD1           LD2           LD3           LD4           LD5           LD6
views            3.156820e-08  5.302291e-07 -2.603883e-07  4.170850e-07 -2.745626e-07  7.903367e-08
likes            1.908450e-07 -3.816786e-07  1.731247e-06 -1.605011e-05  1.748616e-05 -8.529036e-06
dislikes        -3.655146e-06 -7.179047e-06  7.668266e-05 -2.517705e-05 -1.666751e-05 -7.292271e-05
comment_count   -3.326777e-07 -1.721562e-05  1.888409e-05  2.916024e-05 -4.937770e-05  1.146310e-04
frequency        1.898358e+00 -5.166767e-01  2.243222e-03  3.399721e-02 -7.324223e-02 -8.993064e-02
dislike_percent  8.445806e-01  6.144261e-01 -4.977466e+00 -8.616016e+00 -5.120979e+00  4.436130e-01

Proportion of trace:
   LD1    LD2    LD3    LD4    LD5    LD6
0.9573 0.0346 0.0033 0.0022 0.0022 0.0004
>
```

This means that the first discriminant function is a linear combination of the variables: 3.156820e-08∗views+1.908450e-07∗likes……+8.445806e-01∗dislike_percent. For convenience, the value for each discriminant function (eg. the first discriminant function) are scaled so that their mean value is zero and its variance is one.

The “proportion of trace” (the variable returned by the lda() function) is the percentage separation achieved by each discriminant function. For example, for this data we get the same values as just calculated (95.73%, 3.46%, 3.3%,0.22%, 0.04%).

LDA determines group means and computes, for each individual, the probability of belonging to the different groups. The individual is then affected to the group with the highest probability score.

The lda() outputs contain the following elements:

- Prior probabilities of groups: the proportion of training observations in each group.
- Group means: group center of gravity. Shows the mean of each variable in each group.
- Coefficients of linear discriminants: Shows the linear combination of predictor variables that are used to form the LDA decision rule.

The predict() function returns the following elements:

- class: predicted classes of observations.
- posterior: is a matrix whose columns are the groups, rows are the individuals and values are the posterior probability that the corresponding observation belongs to the groups.
- x: contains the linear discriminants

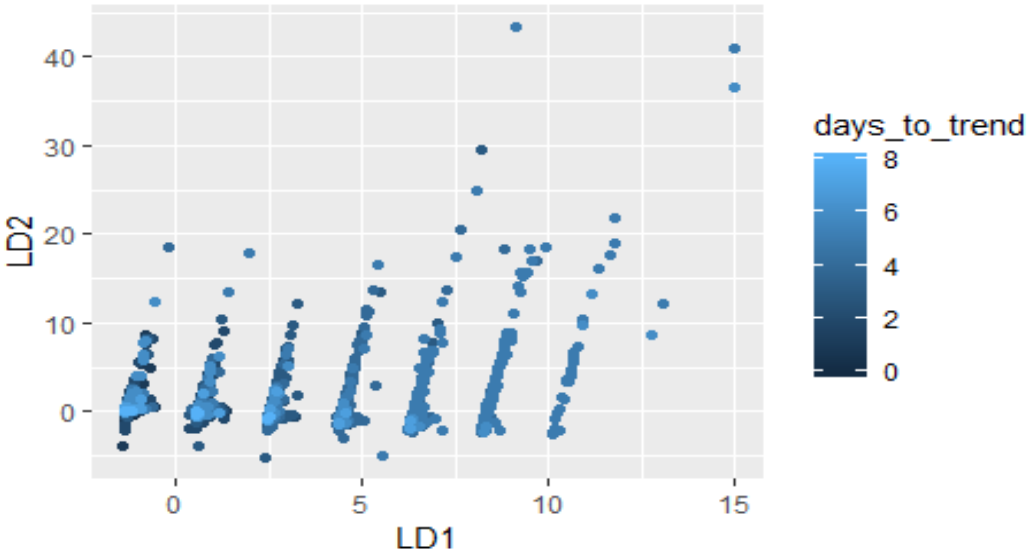
The model accuracy can be calculated for the model:

```
> mean(predict_1$class==video1$days_to_trend)
[1] 0.7506857
```

As you can see, the model accuracy is 75.06%. Which means that our model classified 75% of the data correctly.

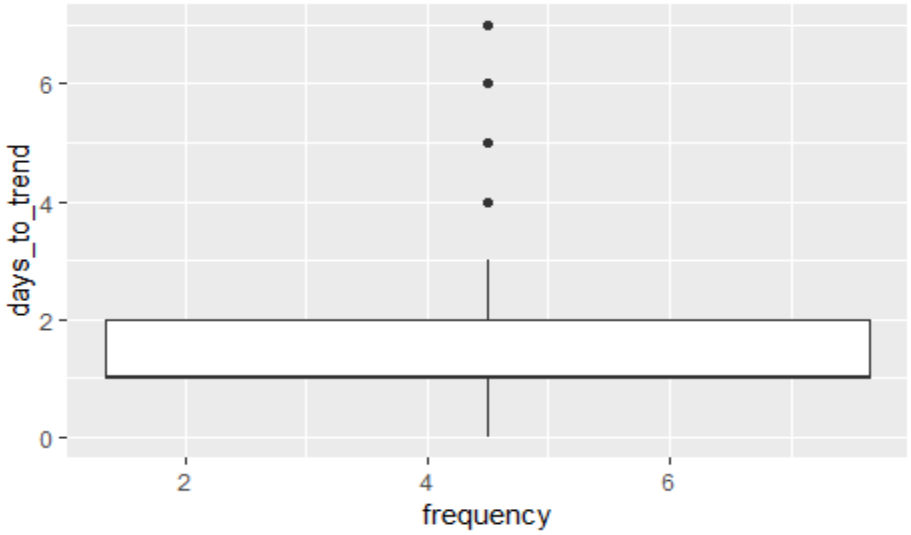
A nice way of displaying the results of a linear discriminant analysis (LDA) is to make a ggplot of the values of the discriminant function for the samples from different groups (different variables in our group).

We can do this using the ggplot function from the ggplot2 library.



The above scatter-plot gives us an idea of how the days_to_trend variable is distributed with respect to the first and second linear discriminant functions(views and likes). This plot shows that most videos have a time span of not more than 0 to 5 days to get into the trending list, but as the plot scatters a lot which represents the outliers.

To get a more clear idea of, how many days a video takes to get in the trending list a box plot is being plotted which could be more informative rather than the scatter-plot. While plotting the box plot a filter is applied on the dataset, the reason behind it is when observed, almost 98% of data have a time span of not more than 8 days to get in the trending videos list. Thus, plotting a box plot would give more accurate knowledge of the days a video takes to get in the trending videos list.



As you can see in the above box plot most of the videos take a span of 1 to 2 days to get in the trending videos list, with the mean being 1 day. Some outliers are persistent which take days which are beyond the scale.

4) DATA PRE-PROCESSING

While conducting EDA, we figured out few variables which could be processed further for our analysis. So via Data Pre-processing, we focused on these variables and deduced a new data frame by adding them to the existing dataset.

We sum up Data Pre-processing with the following 3 tasks-

a) Variable Transformations

We have created 3 additional variables with the existing variables in the dataset. These are-

- **days_to_trend** – This refers to the number of days in between the publishing and trending date of a video.

  days_to_trend = trending_date – publish_date

- **dislike_percent** – This is the percentage of dislikes for a video as a fraction of the total likes and dislikes

  dislike_percent = (dislikes) / (likes + dislikes)

- **frequency** – This denotes the frequency of occurrence of a video in the dataset, in other words number of times it has trended.

  count(df$video_id)

	video_id	freq
1	--45ws7CEN0	1
2	--7vNbh4UNA	3
3	-0CMnp02rNY	1
4	-0DjA_r32uQ	1
5	-0F7AFzWXik	1
6	-0NhqVYR4UY	1
7	-0Nuw8wX3tE	1
8	-0qe75Q5vOg	1
9	-0QvjiG4sYM	2
10	-0Rj7qcLu1A	1
11	-0XP8UXesHg	1
12	-10X8LZxvOE	1

	video_id	dislike_percent
1	--45ws7CEN0	0.118538939
4	--7vNbh4UNA	0.025309138
5	-0CMnp02rNY	0.015116279
6	-0DjA_r32uQ	0.027531083
7	-0F7AFzWXik	0.415384615
8	-0NhqVYR4UY	0.059870550
9	-0Nuw8wX3tE	0.011824990
10	-0qe75Q5vOg	0.013068182
12	-0QvjiG4sYM	0.237125749
13	-0Rj7qcLu1A	0.182856249
14	-0XP8UXesHg	0.040816327
15	-10X8LZxvOE	0.012121212

	video_id	date_of_publishing	date_of_trending	days_to_trend
1	--45ws7CEN0	2018-06-12	2018-06-12	0
4	--7vNbh4UNA	2018-04-13	2018-04-16	3
5	-0CMnp02rNY	2018-06-04	2018-06-05	1
6	-0DjA_r32uQ	2018-02-12	2018-02-14	2
7	-0F7AFzWXik	2018-02-18	2018-02-19	1
8	-0NhqVYR4UY	2018-03-02	2018-03-03	1
9	-0Nuw8wX3tE	2018-04-21	2018-04-22	1
10	-0qe75Q5vOg	2017-12-24	2017-12-25	1
12	-0QvjiG4sYM	2018-04-05	2018-04-07	2
13	-0Rj7qcLu1A	2018-04-24	2018-04-25	1
14	-0XP8UXesHg	2018-01-23	2018-01-24	1
15	-10X8LZxvOE	2018-01-05	2018-01-06	1

The figure above shows the transformed variables that have been created by us using the video IDs for the videos in the dataset. Each variable has been shown against the variable video_id in the above table. They will now be a part of all the analysis that we follow from here including the algorithms and visualizations.

b) New Data Frame

The next step in our analysis is to add the transformed variables derived above into the dataframe that we obtained from the dataset for the project. After the addition of the variables and proper sorting as per distinct video IDs, we obtain the dataframe as below-

	video_id	trending_date	title	channel_title	category
1	--45ws7CEN0	18.12.06	PlayStation E3 2018 Showcase | English	PlayStation Europe	Gaming
4	--7vNbh4UNA	18.16.04	Responding to ALL The Outrage, Ridiculou...	Philip DeFranco	News & Politics
5	-0CMnp02rNY	18.05.06	Mindy Kaling's Daughter Had the Perfect R...	TheEllenShow	Entertainment
6	-0DjA_r32uQ	18.14.02	2.12 - Q & Current Event Analysis from Feb ...	Destroying the Illusion	Education
7	-0F7AFzWXik	18.19.02	Is Russia funding the NRA?	CBC News	News & Politics
8	-0NhqVYR4UY	18.03.03	President Donald Trump Governing Like 'M...	MSNBC	News & Politics
9	-0Nuw8wX3tE	18.22.04	INFINITY WAR: Sebastian Stan continues to ...	ODE	Entertainment
10	-0qe75Q5vOg	17.25.12	Irish People Taste Test Christmas Cocktails	Facts.	Entertainment

publish_time	tags	views	likes	dislikes	comment_count
2018-06-12T03:11:18.000Z	playstation|"playstation 4"|...	309197	3837	516	278
2018-04-13T19:00:00.000Z	Elizabeth Hurley|"Instagra...	1335225	60694	1576	10150
2018-06-04T13:00:00.000Z	ellen|"ellen degeneres"|"th...	219401	3388	52	128
2018-02-12T23:44:59.000Z	jordan sather|"destroying ...	57494	2190	62	768
2018-02-18T19:04:23.000Z	donald trump|"russia"|"put...	6070	152	108	205
2018-03-02T06:03:54.000Z	Last Word|"The Last Word"...	147706	1162	74	591
2018-04-21T15:49:33.000Z	entertainment|"avengers"|"...	85782	2507	30	131
2017-12-24T21:30:00.000Z	Facts|"Cocktails"|"Mistletoe...	98160	3474	46	259
2018-04-05T23:03:59.000Z	latest News|"Happening N...	383196	1911	594	2612
2018-04-24T05:06:55.000Z	2018|"Jordan peterson"|"jo...	507781	5224	1169	6879

thumbnail_link	comments_disabled	ratings_disabled	video_error_or_removed
https://i.ytimg.com/vi/--45ws7CEN0/default.jpg	FALSE	FALSE	FALSE
https://i.ytimg.com/vi/--7vNbh4UNA/default.jpg	FALSE	FALSE	FALSE
https://i.ytimg.com/vi/-0CMnp02rNY/default.jpg	FALSE	FALSE	FALSE
https://i.ytimg.com/vi/-0DjA_r32uQ/default.jpg	FALSE	FALSE	FALSE
https://i.ytimg.com/vi/-0F7AFzWXik/default.jpg	FALSE	FALSE	FALSE
https://i.ytimg.com/vi/-0NhqVYR4UY/default.jpg	FALSE	FALSE	FALSE
https://i.ytimg.com/vi/-0Nuw8wX3tE/default.jpg	FALSE	FALSE	FALSE
https://i.ytimg.com/vi/-0qe75Q5vOg/default.jpg	FALSE	FALSE	FALSE
https://i.ytimg.com/vi/-0QvjiG4sYM/default.jpg	FALSE	FALSE	FALSE
https://i.ytimg.com/vi/-0Rj7qcLu1A/default.jpg	FALSE	FALSE	FALSE

description	dislike_percent	date_of_publishing	date_of_trending	frequency
Show starts at 1:13:20.E3 2018 has arrived. Join us for th...	0.118538939	2018-06-12	2018-06-12	1
Thanks for tuning in this week, ya Beautiful Bastards. B...	0.025309138	2018-04-13	2018-04-16	3
Ocean's 8 star Mindy Kaling dished on bringing her bab...	0.015116279	2018-06-04	2018-06-05	1
Find me on:\nhttp://www.destroyingtheillusion.com\nT...	0.027531083	2018-02-12	2018-02-14	1
Is Russia funding the NRA? The FBI is looking into whet...	0.415384615	2018-02-18	2018-02-19	1
John Kelly told an audience God punished me with a Tru...	0.059870550	2018-03-02	2018-03-03	1
As Letitia Wright and Sebastian Stan continue the Aven...	0.011824990	2018-04-21	2018-04-22	1
It's everything about Christmas in a drink!\n\nSubscribe ...	0.013068182	2017-12-24	2017-12-25	1
President Donald Trump said he did not know about a $...	0.237125749	2018-04-05	2018-04-07	2
NEW MERCH UP (Peterson Sorting Hat) Spread the philo...	0.182856249	2018-04-24	2018-04-25	1

The above figures are part of the final dataframe that we shall utilize in our project for all the analysis.

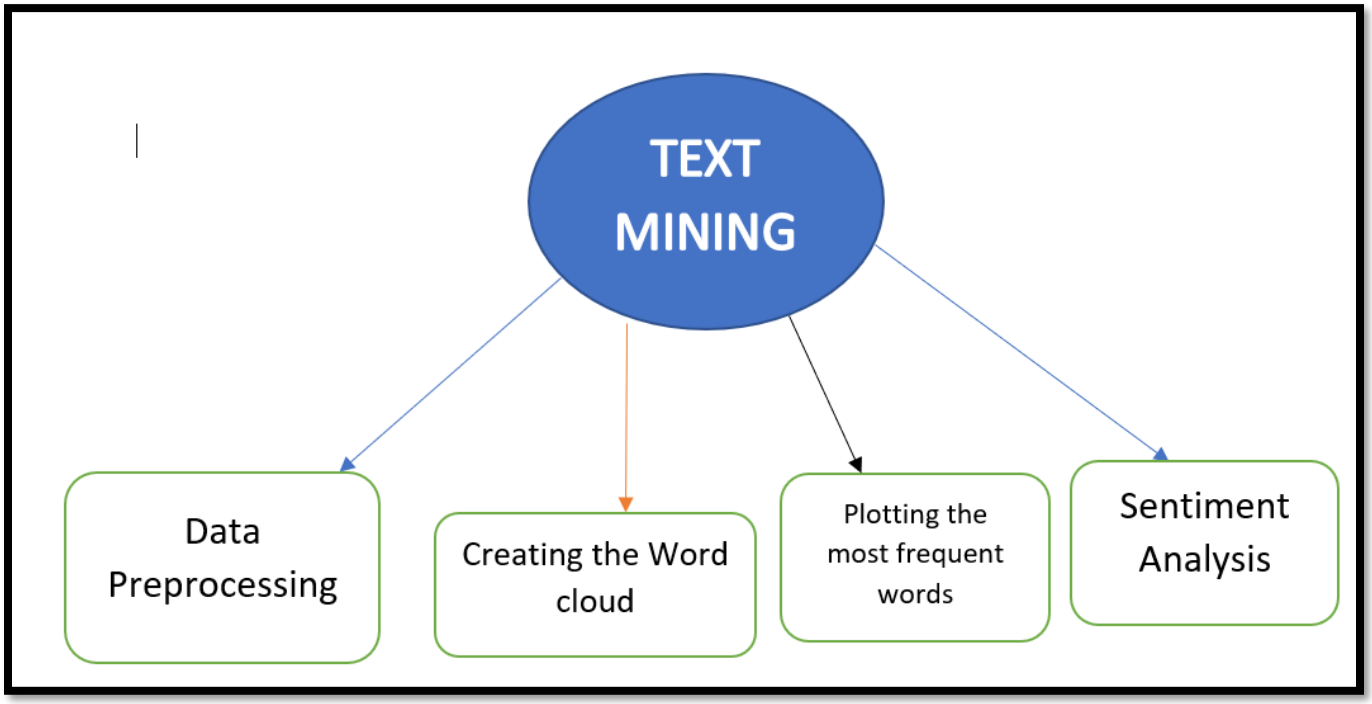
c) Quick Summary

We create a summary of the new dataframe to check all our variables at the go before we jump into prediction and further work.

```
'data.frame':   24104 obs. of  18 variables:
 $ video_id              : Factor w/ 24104 levels "--45ws7CEN0",..: 1 2 3 4 5 6 7 8 9 10 ...
 $ title                 : Factor w/ 24573 levels "''Gala Artis 2018'' Le numÃ©ro d'ouverture",..: 1
0 14700 835 11159 16820 10953 11095 22010 20620 ...
 $ channel_title         : Factor w/ 5076 levels "- æ¬¢è¿Žè®¢é\230… -æµ\231æ±Ÿå\215«è§†ã\200\220å¥"
\200‘å®\230æ–'é¢‘é\201“"",..: 3464 3424 4405 1202 825 3042 3286 1522 974 2922 ...
 $ category              : Ord.factor w/ 30 levels "Film & Animation"<..: 8 13 12 15 13 13 12 12 13
 $ tags                  : Factor w/ 20157 levels "''Hace|\"minuto''\"|\"''hace\"|\"minutos''\"|\"'
''\"|\"''animales''\"|\"''mejores''\"|\"tops''\"|\"''"| __truncated__,..: 13819 5818 5855 9486 5329
3 6259 10450 338 ...
 $ views                 : int  309197 1335225 219401 57494 6070 147706 85782 98160 383196 507781 .
 $ likes                 : int  3837 60694 3388 2190 152 1162 2507 3474 1911 5224 ...
 $ dislikes              : int  516 1576 52 62 108 74 30 46 594 1169 ...
 $ comment_count         : int  278 10150 128 768 205 591 131 259 2612 6879 ...
 $ thumbnail_link        : Factor w/ 24422 levels "https://i.ytimg.com/vi/--45ws7CEN0/default.jpg",
 14 15 16 17 18 19 20 ...
 $ comments_disabled     : logi  FALSE FALSE FALSE FALSE FALSE FALSE ...
 $ ratings_disabled      : logi  FALSE FALSE FALSE FALSE FALSE FALSE ...
 $ video_error_or_removed: logi  FALSE FALSE FALSE FALSE FALSE FALSE ...
 $ description           : Factor w/ 22341 levels "","''' Vous Ãªtes dÃ©jÃ  allÃ© en Musulmanie ? ''
dan 2018 s'approche et nous allons poser quelques questi"| __truncated__,..: 15838 17621 13394 6799
4 2526 9910 14504 12942 ...
 $ dislike_percent       : num  0.1185 0.0253 0.0151 0.0275 0.4154 ...
 $ date_of_publishing    : Date, format: "2018-06-12" "2018-04-13" "2018-06-04" ...
 $ date_of_trending      : Date, format: "2018-06-12" "2018-04-16" "2018-06-05" ...
 $ frequency             : int  1 3 1 1 1 1 1 2 1 ...
```

5) TEXT MINING

Text analytics is the process of deriving high-quality information from text. High-quality information is typically derived through the devising of patterns and trends through means such as statistical pattern learning. Text mining usually involves the process of structuring the input text (usually parsing, along with the addition of some derived linguistic features and the removal of others, and subsequent insertion into a database, deriving patterns within the structured data, and finally evaluation and interpretation of the output. 'High quality' in text mining usually refers to some combination of relevance, novelty and interest. Typical text mining tasks include text categorization, text clustering, concept/entity extraction, production of granular taxonomies.



In Text mining, we tried to assess the strength of the data by taking the crucial part of youtube metadata that is tags which were given to unique video Id. First, we plucked the top 5000 rows of tags from the unique dataset of 24424 observations. After that we vacuumed the data by cleaning the part of tags with functions like string replace, iconvert and corpus which made this task easy. So we started with the data preprocessing of tags with conversion of Non-ASCII words to ASCII and then making a corpus of the data under which following steps were performed before mining the text:

- Convert the uppercase text to lowercase text (A-Z to a-z)
- Remove all the punctuations from the tags like (‘,;.[]/@!#$%^&*())

- Remove numbers from tags of the data (0-9)
- Remove english Stop words like (a, the, from, have etcetera which have no significance)
- Remove URL and foreign language URLs
- Eliminated extra white spaces
- Remove additional stop words which have no significance by looking the data

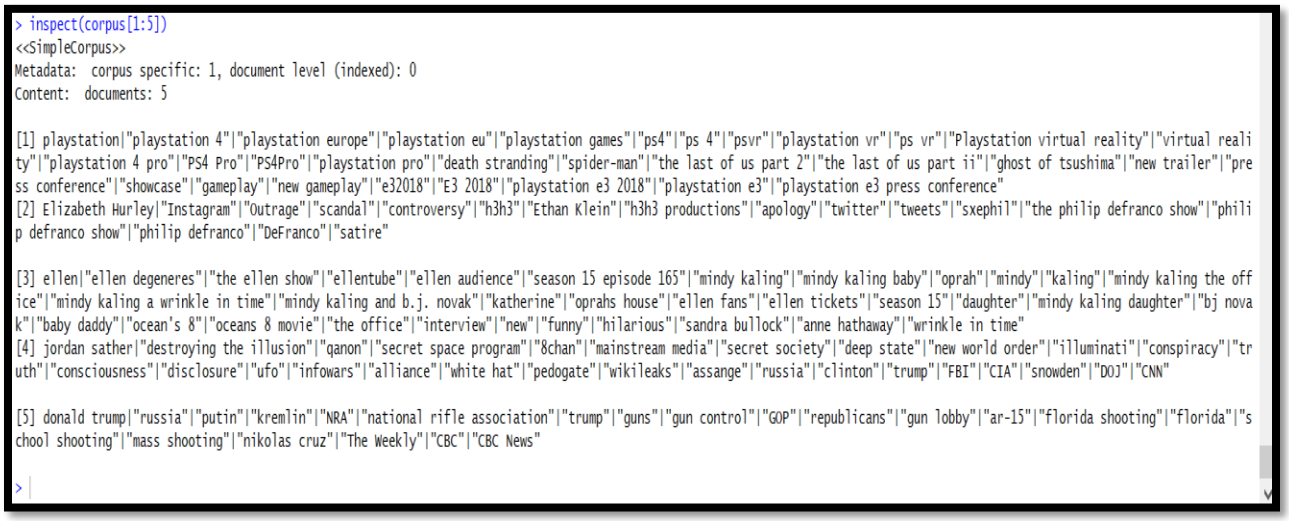
1. Text stemming (reducing words to their root form), that is, making the tenses of the same word as of equal significance and meaning.
2. Before anything was removed our data looks somehow like this which included punctuations, white spaces, stopwords, foreign languages and everything possible:

Filter

1	playstation|"playstation 4"|"playstation europe"|"playstation ...
2	Elizabeth Hurley|"Instagram"|"Outrage"|"scandal"|"controvers...
3	ellen|"ellen degeneres"|"the ellen show"|"ellentube"|"ellen au...
4	jordan sather|"destroying the illusion"|"qanon"|"secret space...
5	donald trump|"russia"|"putin"|"kremlin"|"NRA"|"national rifle ...
6	Last Word|"The Last Word"|"The Last Word with Lawrence O...
7	entertainment|"avengers"|"infinity war"|"sebastian stan"|"letiti...
8	Facts|"Cocktails"|"Mistletoe"|"Grinch"|"Nutcracker"|"Gingerbr...
9	latest News|"Happening Now"|"CNN"|"Politics"|"Donald Tru...
10	2018|"Jordan peterson"|"jordan peterson cathy newman"|"jo...
11	the bachelor|"the bachelorette"|"bachelor nation"|"abc"|"abc ...
12	Heartland|"#iloveheartland"|"Amber Marshall"|"Christ Potter"...
13	Youtube Removed My Logan Paul Video|"Youtube video re...
14	jimmy|"jimmy kimmel"|"jimmy kimmel live"|"late night"|"talk s...
15	awesomenessTV|"awesomenessTV Network"|"DreamWorks"|...
16	Bollywood updates|"bollywood latest news"|"bollywood"
17	Power 106|"Power106"|"The Cruz Show"|"Hip Hop"|"Hip-Hop...
18	benedict cumberbatch|"tom holland"|"doctor strange"|"spid...
19	death wish|"1974"|"deathwish"|"charles bronson"|"cinemasin...
20	omargoshtv|"HAUNTED PRISON TUNNEL AT 3AM"|"EXPLORI...
21	bá°¡n muá»'n há°¹n hÃ²|"ban muon hen ho"|"ban muon hen ...
22	Pandakillç¬¬å»å£|"ç‹¼ä°°æ€"|"ç‹¼ä°°æ®°"|"Pandakillç‹¼ä...
23	lil wayne|"d6"|"dedication 6"|"drake"|"lil wayne drake"|"lil way...
24	marie-mai|"mariemai"|"pop"|"empire"|"electro"|"spectra"|"sp...
25	gordon ramsay|"gordon ramsey"|"ramsay"|"ramsey"|"chef r...
26	thaw|"eli kintisch"|"arctic"|"Vox.com"|"vox"|"explain"|"explaine...
27	fail compilation|"fails of the month"|"epic fail compilation"|"f...

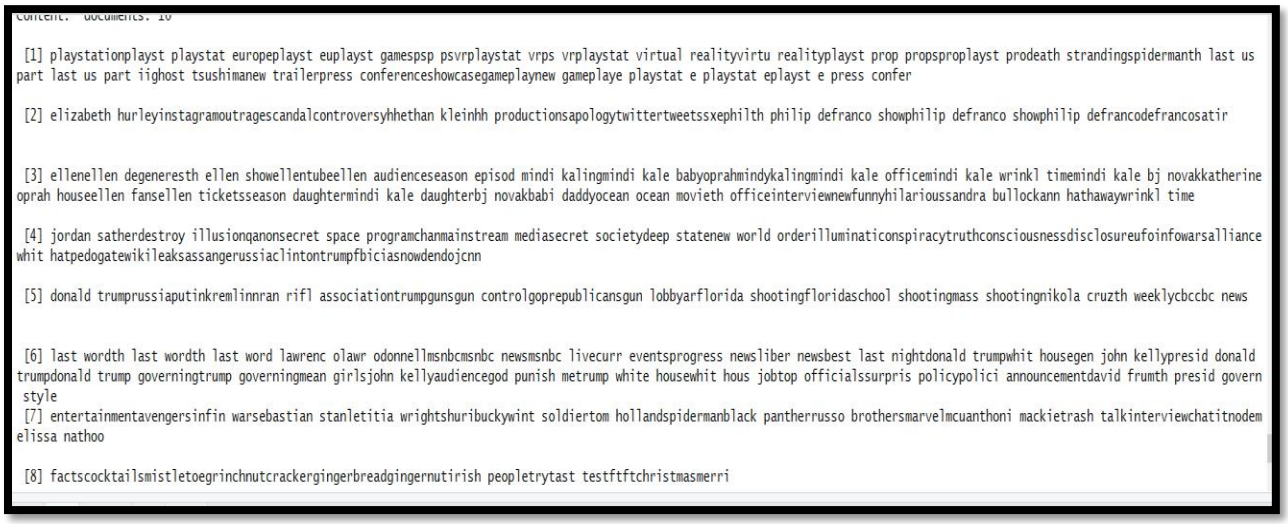
Showing 1 to 28 of 5,000 entries

3. Now first we converted the foreign language that is Non-ASCII to ASCII words which resulted the data to look somehow like this:



```
> inspect(corpus[1:5])
<<SimpleCorpus>>
Metadata:  corpus specific: 1, document level (indexed): 0
Content:  documents: 5

[1] playstation|"playstation 4"|"playstation europe"|"playstation eu"|"playstation games"|"ps4"|"ps 4"|"psvr"|"playstation vr"|"ps vr"|"Playstation virtual reality"|"virtual reali
ty"|"playstation 4 pro"|"PS4 Pro"|"PS4Pro"|"playstation pro"|"death stranding"|"spider-man"|"the last of us part 2"|"the last of us part ii"|"ghost of tsushima"|"new trailer"|"pre
ss conference"|"showcase"|"gameplay"|"new gameplay"|"e32018"|"E3 2018"|"playstation e3 2018"|"playstation e3"|"playstation e3 press conference"
[2] Elizabeth Hurley|"Instagram"|"Outrage"|"scandal"|"controversy"|"h3h3"|"Ethan Klein"|"h3h3 productions"|"apology"|"twitter"|"tweets"|"sxephil"|"the philip defranco show"|"phili
p defranco show"|"philip defranco"|"DeFranco"|"satire"

[3] ellen|"ellen degeneres"|"the ellen show"|"ellentube"|"ellen audience"|"season 15 episode 165"|"mindy kaling"|"mindy kaling baby"|"oprah"|"mindy"|"kaling"|"mindy kaling the off
ice"|"mindy kaling a wrinkle in time"|"mindy kaling and b.j. novak"|"katherine"|"oprahs house"|"ellen fans"|"ellen tickets"|"season 15"|"daughter"|"mindy kaling daughter"|"bj nova
k"|"baby daddy"|"ocean's 8"|"oceans 8 movie"|"the office"|"interview"|"new"|"funny"|"hilarious"|"sandra bullock"|"anne hathaway"|"wrinkle in time"
[4] jordan sather|"destroying the illusion"|"qanon"|"secret space program"|"8chan"|"mainstream media"|"secret society"|"deep state"|"new world order"|"illuminati"|"conspiracy"|"tr
uth"|"consciousness"|"disclosure"|"ufo"|"infowars"|"alliance"|"white hat"|"pedogate"|"wikileaks"|"assange"|"russia"|"clinton"|"trump"|"FBI"|"CIA"|"snowden"|"DOJ"|"CNN"

[5] donald trump|"russia"|"putin"|"kremlin"|"NRA"|"national rifle association"|"trump"|"guns"|"gun control"|"GOP"|"republicans"|"gun lobby"|"ar-15"|"florida shooting"|"florida"|"s
chool shooting"|"mass shooting"|"nikolas cruz"|"The Weekly"|"CBC"|"CBC News"

>
```

4. Now proceeding to the last step of data preprocessing we removed everything except the significant English words of the data which resulted in the following kind of a text:



```
Content:  documents: 10

 [1] playstationplayst playstat europeplayst euplayst gamespsp psvrplaystat vrps vrplaystat virtual realityvirtu realityplayst prop propsproplayst prodeath strandingspidermanth last us
part last us part iighost tsushimanew trailerpress conferenceshowcasegameplaynew gameplaye playstat e playstat eplayst e press confer

 [2] elizabeth hurleyinstagramoutragescandalcontroversyhhethan kleinhh productionsapologytwittertweetssxephilth philip defranco showphilip defranco showphilip defrancodefrancosatir


 [3] ellenellen degeneresth ellen showellentubeellen audienceseason episod mindi kalingmindi kale babyoprahmindykalingmindi kale officemindi kale wrinkl timemindi kale bj novakkatherine
oprah houseellen fansellen ticketsseason daughtermindi kale daughterbj novakbabi daddyocean ocean movieth officeinterviewnewfunnyhilarioussandra bullockann hathawaywrinkl time

 [4] jordan satherdestroy illusionqanonsecret space programchanmainstream mediasecret societydeep statenew world orderilluminaticonspiracytruthconsciousnessdisclosureufoinfowarsalliance
whit hatpedogatewikileaksassangerussiaclintontrumpfbiciasnowdendojcnn

 [5] donald trumprussiaputinkremlinnran rifl associationtrumpgunsgun controlgoprepublicansgun lobbyarflorida shootingfloridaschool shootingmass shootingnikola cruzth weeklycbccbc news


 [6] last wordth last wordth last word lawrenc olawr odonnellmsnbcmsnbc newsmsnbc livecurr eventsprogress newsliber newsbest last nightdonald trumpwhit housegen john kellypresid donald
trumpdonald trump governingtrump governingmean girlsjohn kellyaudiencegod punish metrump white housewhit hous jobtop officialssurpris policypolici announcementdavid frumth presid govern
 style
 [7] entertainmentavengersinfin warsebastian stanletitia wrightshuribuckywint soldiertom hollandspidermanblack pantherrusso brothersmarvelmcuanthoni mackietrash talkinterviewchatitnodem
elissa nathoo

 [8] factscocktailsmistletoegrinchnutcrackergingerbreadgingernutirish peopletrytast testftftchristmasmerri
```

After the data pre-processing completion, we moved forward to make the term document matrix which was necessary to make the "wordcloud" which is a necessary part of text mining presenting the most occurred words in the data with highest frequency words with the larger text and lower frequency words in smaller text forming a cloud. Our matrix of 5000 observations came out to be of 1.2 gigabytes and formed a word cloud like this:

Word Cloud obtained by the Text Mining of YouTube Metadata

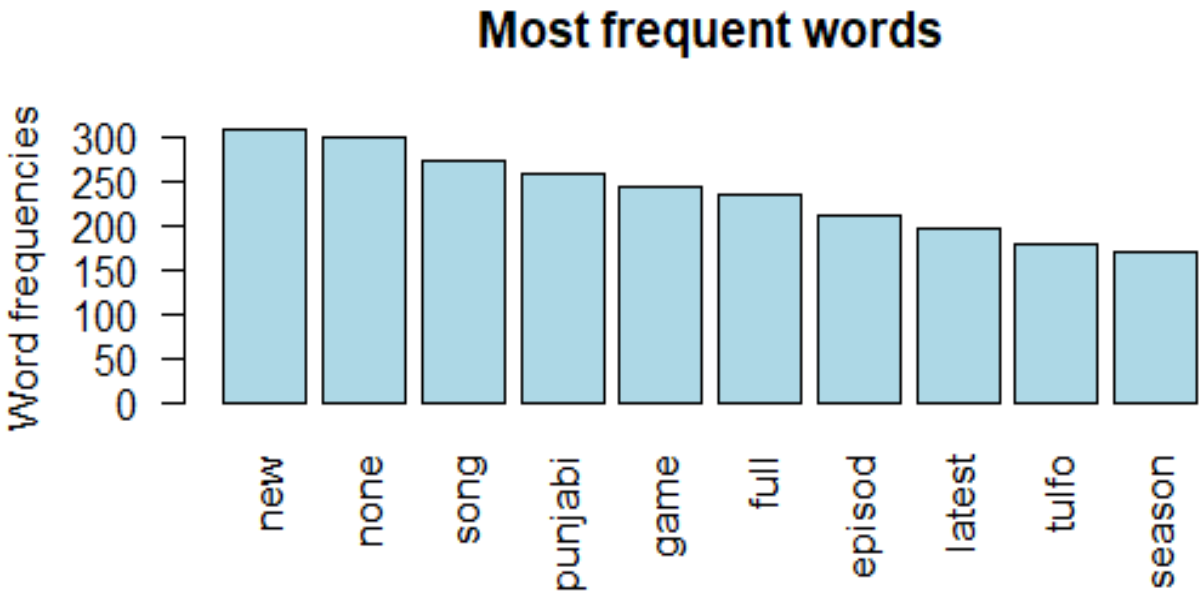
Next Steps:

- We found out the frequency of words

- Then finding the association of a particular word (example- Canada) from higher frequency words with other words resulting in:

```
> findAssocs(dtm, terms = "canada", corlimit = 0.3)
$`canada`
                          countolymp
                                0.69
                           finalsusa
                                0.69
                               fumbl
                                0.69
                         hockeycanada
                                0.69
                          hockeysport
                                0.69
                          hockeywomen
                                0.69
                            koreawint
                                0.69
                          medalcanada
                                0.69
                        olympicssouth
                                0.69
                           scoreteam
                                0.69
                        usapyeongchang
                                0.69
                                women
                                0.68
                            lowryjona
                                0.61
                       basketballdemar
                                0.56
                    nbatorontotoronto
                                0.56
raptorsraptorsbasketballcanadacanada
                                0.56
                       valanciunasair
                                0.56
                           derozankyl
                                0.52
                               hockey
                                0.50
```

- Now finally plotting the barplot for top 10 words with the highest frequency:

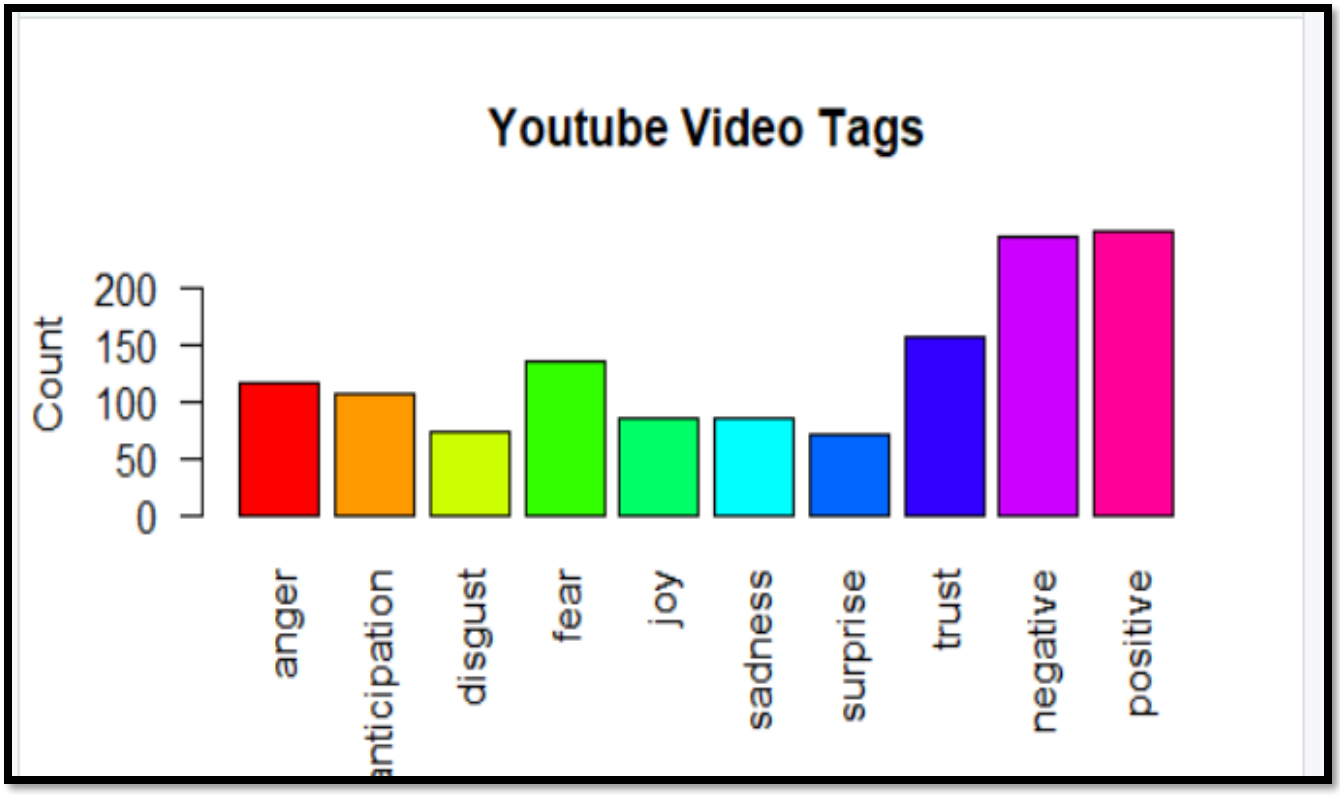


We observe that a number of words are repeated a lot more than the the others which gives us an idea on the kind of words that the most trended words contained in their tags or descriptions that might have pulled the attention of the audience towards watching it or YouTube to be sending it to its 'Trending' section.

Sentiment Analysis

Sentiment analysis is a type of text mining which aims to determine the opinion and subjectivity of its content. When applied to lyrics, the results can be representative of not only the artist's attitudes, but can also reveal pervasive, cultural influences. There are different methods used for sentiment analysis, including training a known dataset, creating your own classifiers with rules, and using predefined lexical dictionaries (lexicons).

So we have performed the sentiment analysis on the tags column of the cleanset of the data which is showing that our data has a little higher positivity than negativity:



Through this plot we can conclude that the sample of 5000 observations considered is leading that in the population data of 24424 observations consists most words of “Entertainment” category showing that the entertainment category videos have an inclining scope of trending videos faster and having more positivity than negativity

ALGORITHMS

LINEAR REGRESSION

RANDOM FOREST

CLASSIFICATION TREE

ASSOCIATION RULE MINING

1) LINEAR REGRESSION

To analyze the relationship and interactions between different variables of our dataset, we built a Linear Regression Model considering the days between the publishing and trending of the video (days_to_trend) as the response variable. Since we conduct data pre-processing prior the beginning our analysis, we decided to conduct regression in two steps -

a) By considering the original dataset keeping all the variables
b) By considering the reduced dataset where we consider only unique values and add couple of transformed variables which we thought might serve useful

The results we obtained by each of the above are given below-

a) All the original variables against days_to_trend

```
Call:
lm(formula = days_to_trend ~ ., data = videoReg.train.df)

Residuals:
   Min     1Q Median     3Q    Max
  -7.5   -2.4   -2.0   -1.1 3652.9

Coefficients:
                Estimate    Std. Error t value       Pr(>|t|)
(Intercept)    6.3076075465  1.0310545468   6.118 0.000000000959 ***
category_id   -0.1334905868  0.0462543346  -2.886         0.0039 **
views          0.0000001709  0.0000001794   0.953         0.3408
likes         -0.0000065904  0.0000061681  -1.068         0.2853
dislikes      -0.0000026722  0.0000248190  -0.108         0.9143
comment_count  0.0000053835  0.0000333995   0.161         0.8719
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 59.36 on 36789 degrees of freedom
Multiple R-squared:  0.0002703, Adjusted R-squared:  0.0001344
F-statistic: 1.989 on 5 and 36789 DF,  p-value: 0.0768
```

In this case we observed that we get one significant variable i.e. 'category_id', indicating the significance of the genre of a video as an important parameter that determines whether it makes in to the 'Trending' section of YouTube or not.

b) The transformed variables against days_to_trend

```
Call:
lm(formula = days_to_trend ~ ., data = videoReg.train.df)

Residuals:
   Min     1Q Median     3Q    Max
  -5.3   -2.0   -1.9   -1.7 3653.9

Coefficients:
                     Estimate     Std. Error t value Pr(>|t|)
(Intercept)      1.80910698826  0.83786159655   2.159  0.03085 *
views            0.00000007711  0.00000029558   0.261  0.79418
likes           -0.00000723679  0.00000996502  -0.726  0.46771
dislikes         0.00000247208  0.00004181090   0.059  0.95285
comment_count    0.00000003523  0.00005179897   0.001  0.99946
frequency        1.17948613648  0.42589439083   2.769  0.00562 **
dislike_percent -1.65529445617  4.35578279449  -0.380  0.70393
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 57.8 on 21795 degrees of freedom
  (181 observations deleted due to missingness)
Multiple R-squared:  0.0004081, Adjusted R-squared:  0.0001329
F-statistic: 1.483 on 6 and 21795 DF,  p-value: 0.1795
```

With the reduced dataset, we observed that the ‘frequency’ of a video was a significant role which is nothing but how many times a video successfully makes it to YouTube’s ‘Trending’ section.

INFERENCE -

With the Regression Model we infer two main factors affecting the trending of a video the most-

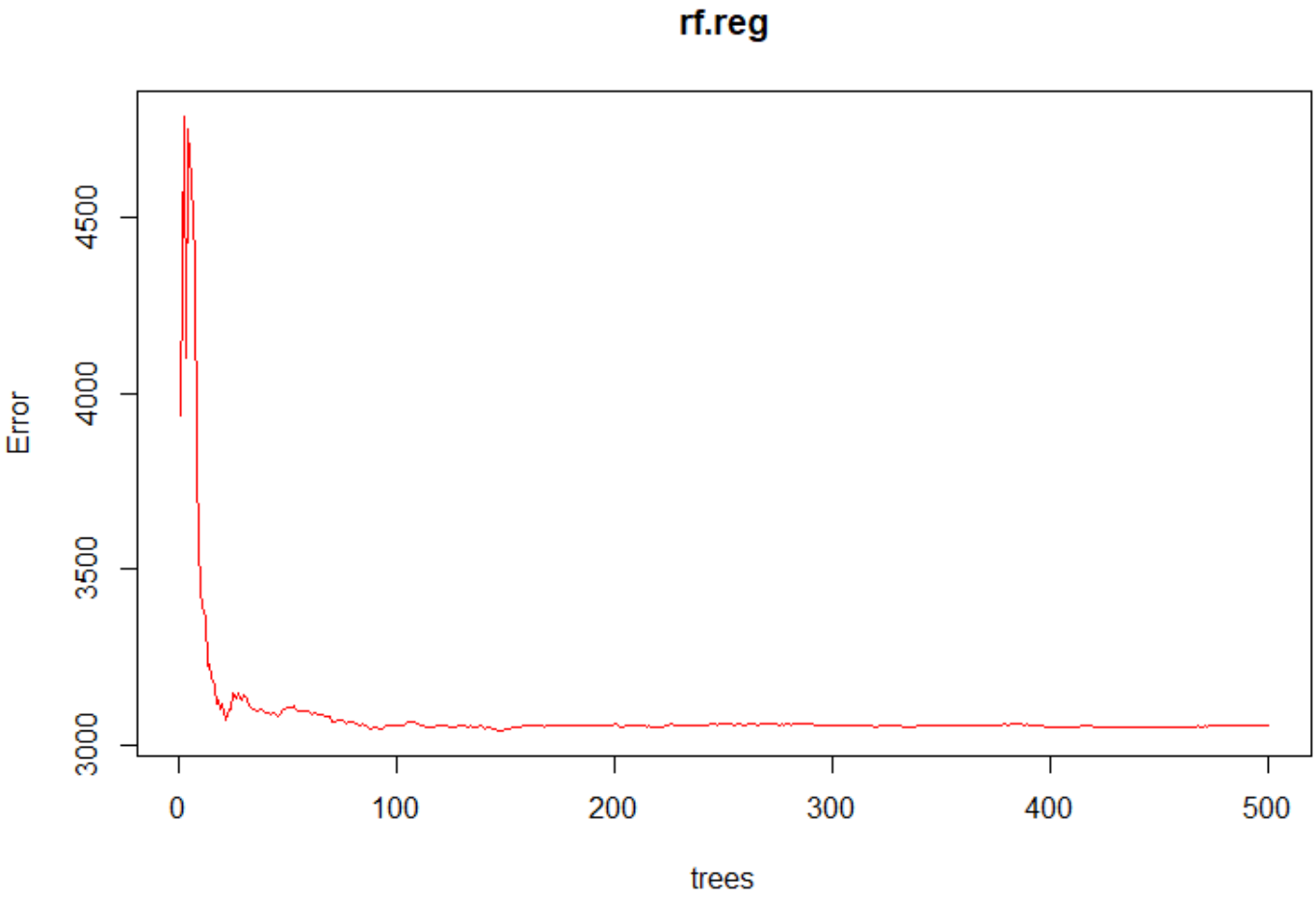
1- Category or genre of a video

2- The number of times a video makes in to the trending list

We observe that the adjusted R Square value of the model upon processing and transformation increases slightly, suggesting that our model is doing well with respect to fitness and the errors are decreasing to a significant extent.

2) RANDOM FOREST

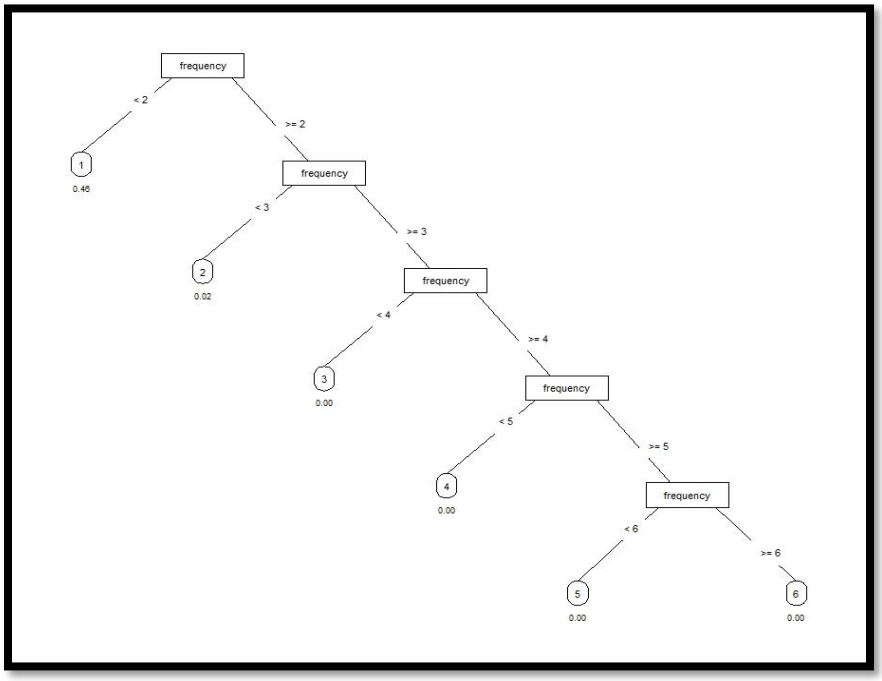
To evaluate the predictive performance of our model, we built a Random forest from our dataset. Via the random forest, we try to analyze an estimate of the information gain at each step of our analysis. As, we know random forests combine the output of various decision trees, making it an excellent ensembling technique. By making a random selection of variables, the random forest tends to remove any kind of bias that might be introduced by a normal decision tree. We have focused on the reduction of errors as we progress with our model and the random forest plot is a way we judge the same.



The random forest plot above has been made considering views, likes, dislikes, comment_count, comments_disabled, ratings_disabled, video_error_or_removed, frequency, dislike_percent and days_to_trend as the variables. This plot shows the Error and the Number of Trees. We can easily notice that how the Error is dropping as we keep on adding more and more trees and average them. This clearly shows an estimate of the predictive power of our model which is quite significant.

3) CLASSIFICATION TREE

Tree models where the target variable can take a discrete set of values are called classification trees; in these tree structures, leaves represent class labels and branches represent conjunctions of features that lead to those class labels. Classification tree analysis is used when the predicted outcome is the class to which the data belongs.




```
Classification tree:
rpart(formula = days_to_trend ~ category + frequency, data = bank.df,
    method = "class", control = rpart.control(cp = 0.001))

Variables actually used in tree construction:
[1] frequency

Root node error: 12601/24427 = 0.51586

n= 24427

         CP nsplit rel error  xerror      xstd
1 0.3431474      0   1.00000 1.00000 0.0061984
2 0.1404650      1   0.65685 0.65685 0.0058706
3 0.0594397      2   0.51639 0.51639 0.0054830
4 0.0282517      3   0.45695 0.45695 0.0052645
5 0.0046028      4   0.42870 0.42870 0.0051475
6 0.0010000      5   0.42409 0.42409 0.0051276
```

Output of classification tree shows that nsplit is 5 and it is the maximum branch available in the model. Tree doesn't classify as per category, which is important variable to study. Considering the output of classification analysis, it is not considered for concluding results.

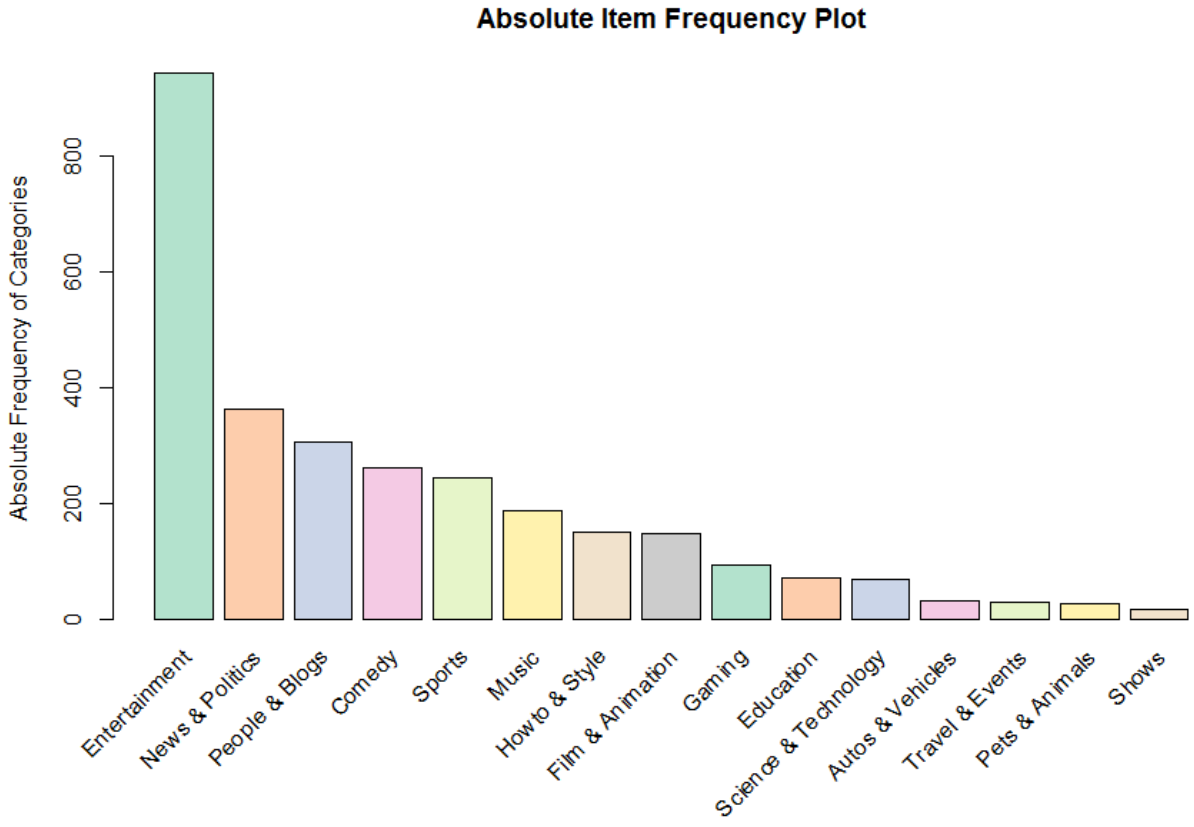
4) ASSOCIATION RULE MINING

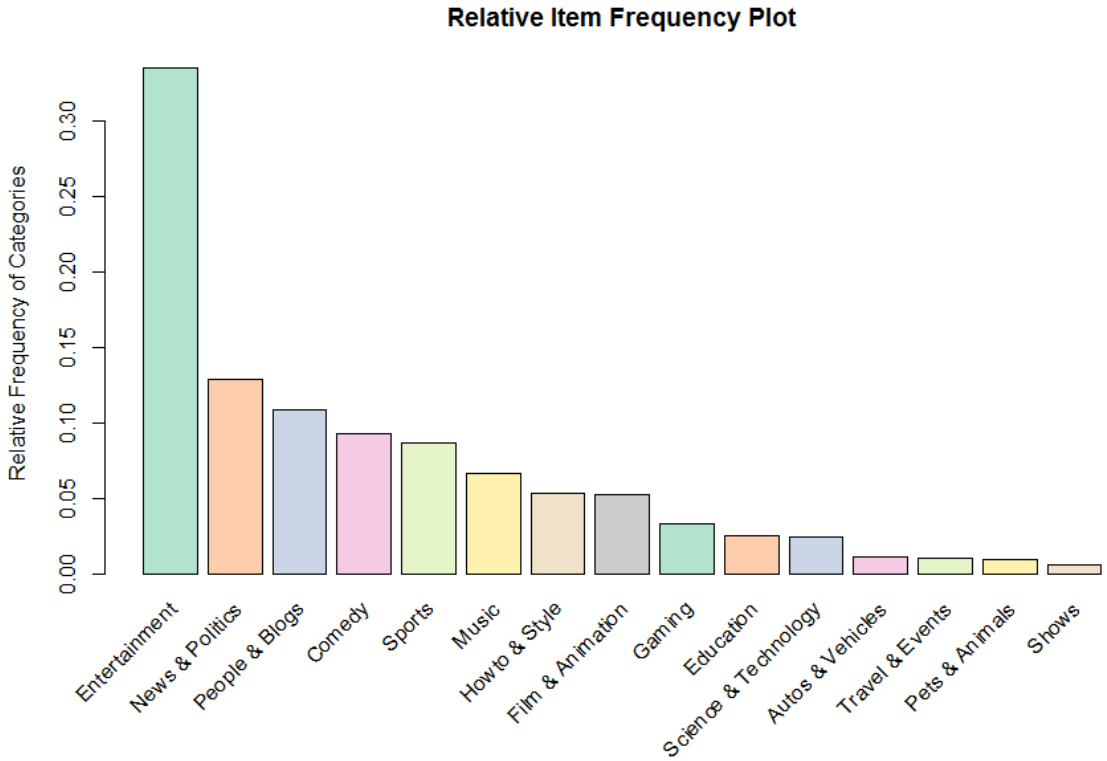
We conducted Association mining on our dataset to determine if we could establish some relation between the views of videos from different categories. We did this in order to understand if the category of a video and its relationship with the previously trended video affects its entry into the 'Trending' section. For this we first obtained a transaction database from our dataset following which we attained the following output –

```
transactions as itemMatrix in sparse format with
 2812 rows (elements/itemsets/transactions) and
 2827 columns (items) and a density of 0.0007229364

most frequent items:
  Entertainment News & Politics  People & Blogs          Comedy          Sports         (Other)
            943             362             306             260             245            3631

element (itemset/transaction) length distribution:
sizes
   1    2    3    4    5    6
   1 2726   58   17    8    2

   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  1.000   2.000   2.000   2.044   2.000   6.000

includes extended item information - examples:
  labels
1      1
2     10
3    100
```

The first plot shows us how frequently a video from a specific category has occurred in the trending videos list. We observed that the videos from the 'Entertainment' category made it to the 'Trending' section the maximum number of times in YouTube in Canada. This was followed by 'News and Politics', 'People and Blogs', 'Comedy' and 'Sports'.

The second plot compares the categories with each other indicating which category videos have made it to the 'Trending' section more times with respect each of their other counterparts. This plot again confirms that the videos from 'Entertainment' and 'News and Politics' are the winners when it comes to making their way into the 'Trending' section of YouTube in Canada.

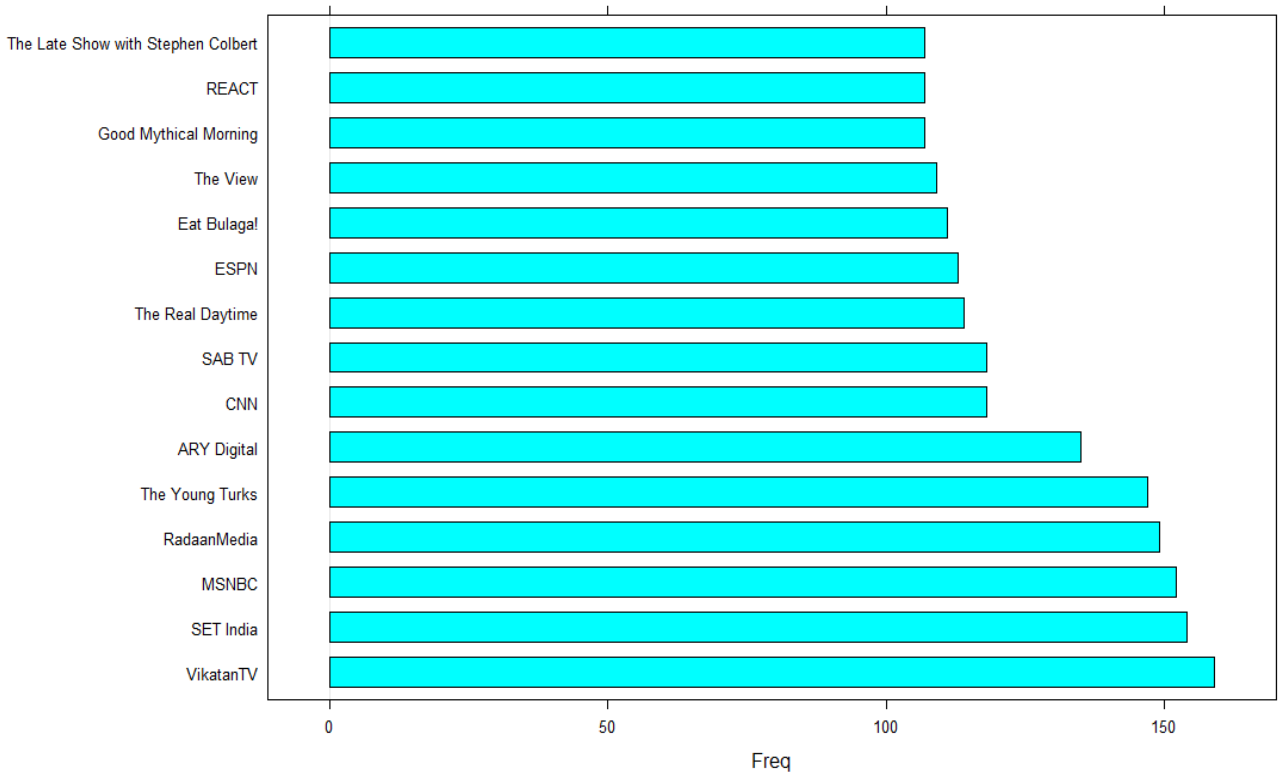

Absolute Item Frequency Plot
Absolute Frequency of Categories
0
200
400
600
800
Entertainment
News & Politics
People & Blogs
Comedy
Sports
Music
How to & Style
Film & Animation
Gaming
Education
Science & Technology
Autos & Vehicles
Travel & Events
Pets & Animals
Shows


Relative Item Frequency Plot
Relative Frequency of Categories
0.00
0.05
0.10
0.15
0.20
0.25
0.30
Entertainment
News & Politics
People & Blogs
Comedy
Sports
Music
How to & Style
Film & Animation
Gaming
Education
Science & Technology
Autos & Vehicles
Travel & Events
Pets & Animals
Shows

DATA VISUALIZATIONS

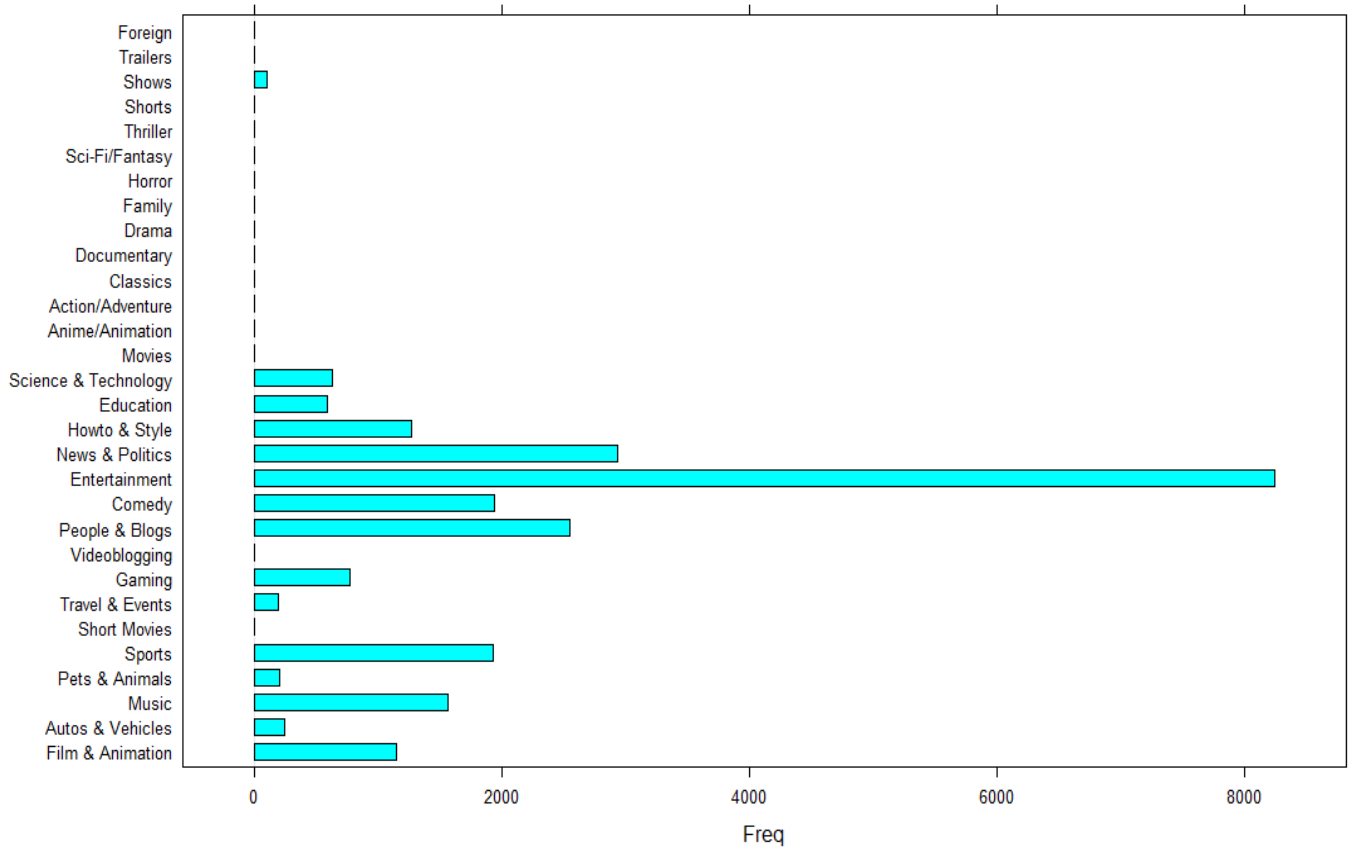
Data visualization is the fastest and most useful way to summarize, analyze and learn more about the data. We have created several charts and plots in our project to better understand the YouTube dataset. the visualizations incorporated helped us spot the outliers in the data and gave us you an idea of possible data transformations that could be applied into the project. The plots of the relationships between attributes gave us an idea of attributes that were redundant, and the resampling methods that might be needed. The visualizations incorporated in our project are described below.

a) Most Influential Creators



The bar chart shows the most influential creators in our dataset. Youtube Creators are the people who create the content and upload the videos on Youtube. The bar chart shows the top 15 most influential creators. the Y-axis depicts the names of channels and X-axis has frequency (number of videos) uploaded by each channel. It can be seen from the graph that VikatanTV has the most number of videos (over 150) which makes it the top most influential creator, followed by SET India MSNBC.

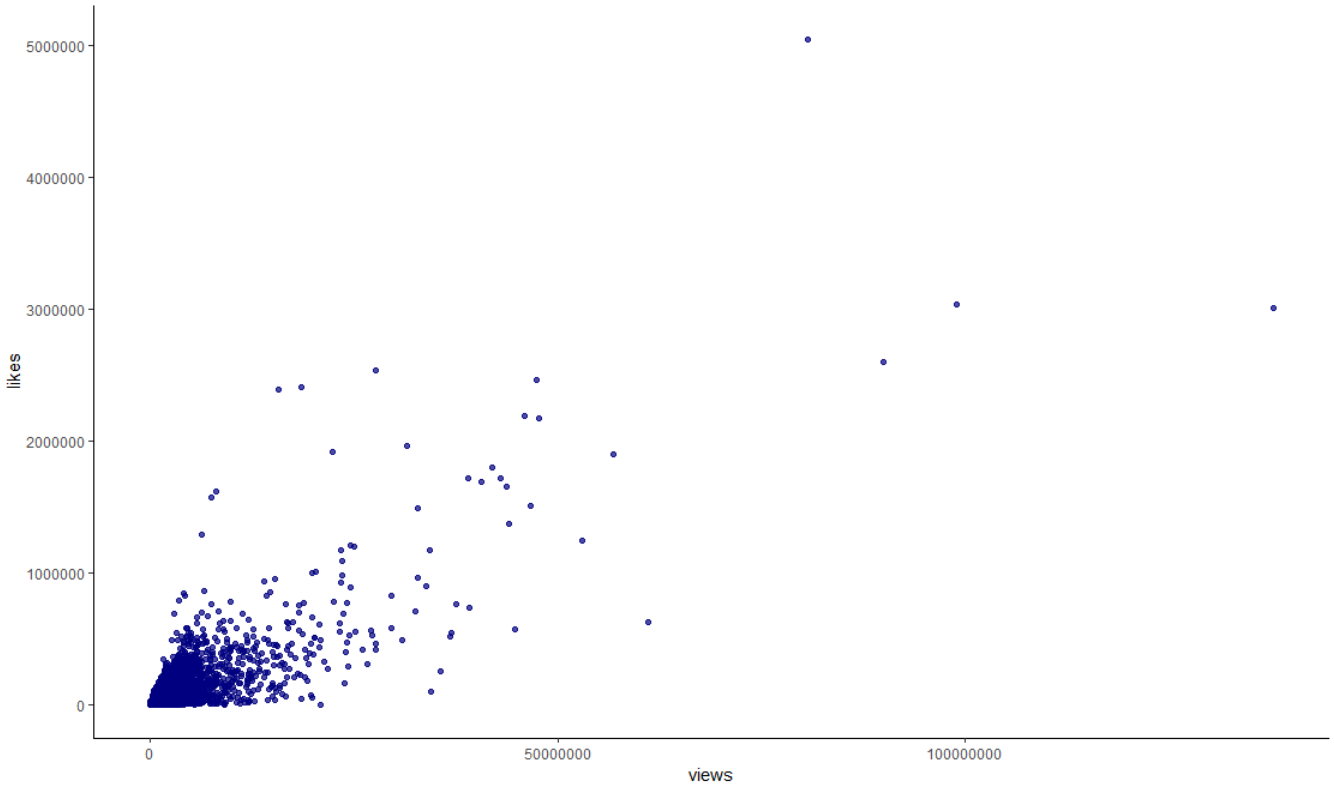
b) Video Category Distribution



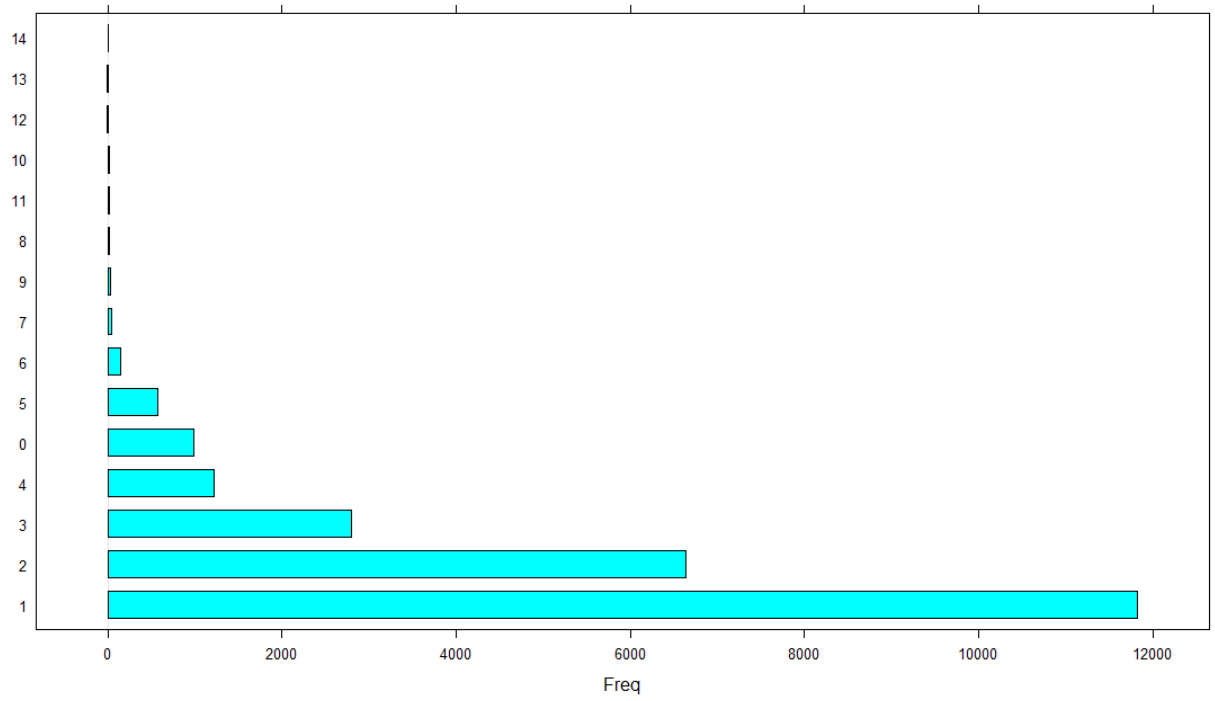
As Bar charts are useful for comparing a single statistic (e.g., average, count, percentage) across groups, we have used it to display the Video Category Distribution. The x-axis (or length in a horizontal display) represents the number of videos, and different bars as shown on the y-axis, correspond to the different categories. It is evident from the graph that there are maximum number of videos in Entertainment category (over 8000) whereas the number of videos in other categories are less than 4000. This supports the predictive task: the numerical outcome is on the x-axis and the y-axis is used for a potential categorical predictor.

c) Scatter Plot (Views vs. Likes)

We have plotted a scatter plot to find out correlation between likes and views for videos. Scatter plot is an important plot in prediction task. Because both variables (likes and views) used in a scatter plot are numerical, it cannot be used to display the relation between days_to_trending and potential predictors for the classification task. This particular scatter plot helps study the association between the two numerical variables (likes and views) in terms of information overlap as well as identifying clusters of observations. The scatter plot shows a positive correlation with some outliers. If the scale of the graph is reduced, we can see a strong positive correlation. It can be inferred that as the number of views on a particular video increase, the number of likes increases too.



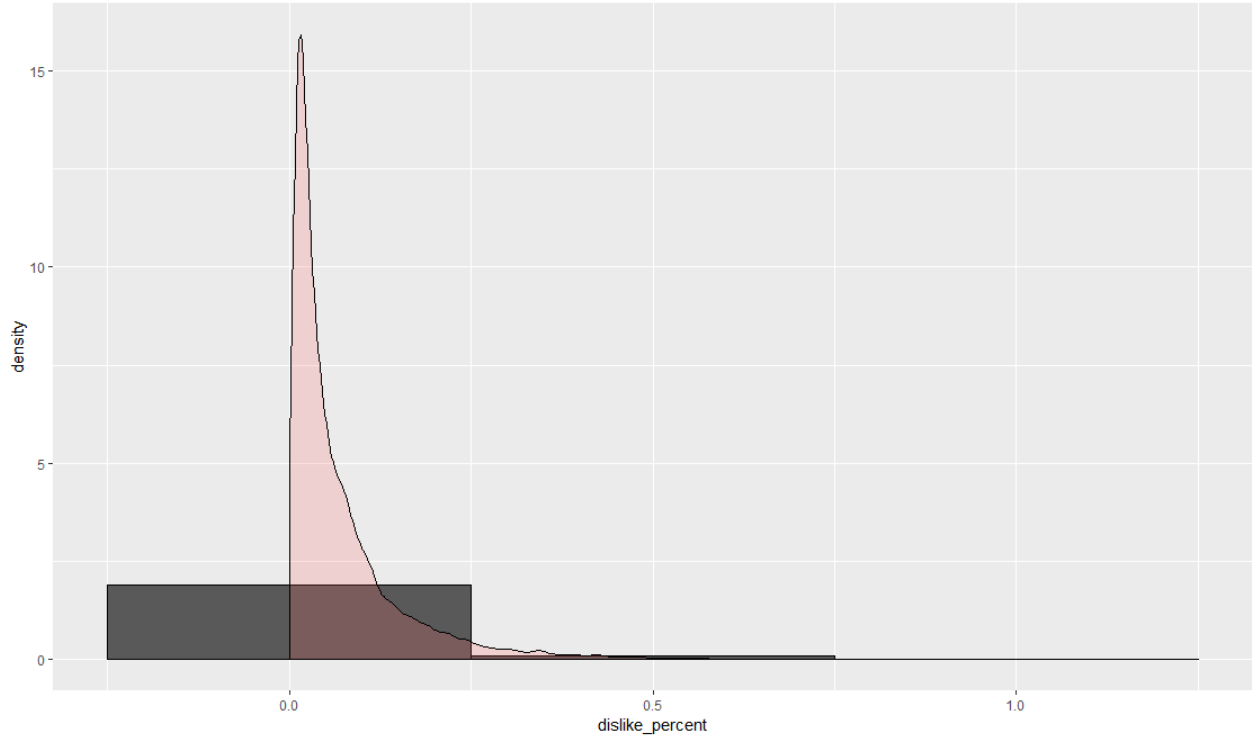
d) Days to trending status



The above bar chart shows the number of days it took for the videos to trend. The Y axis depicts the number of days and the X-axis shows the number of videos. The bar chart shows that approximately 11900 videos took 1 day to trend. It took 2 days for approximately 6700 to get into trending list. The bar chart shows that it takes at the most 6 days for the video to get into the trending list.

e) Density Plot

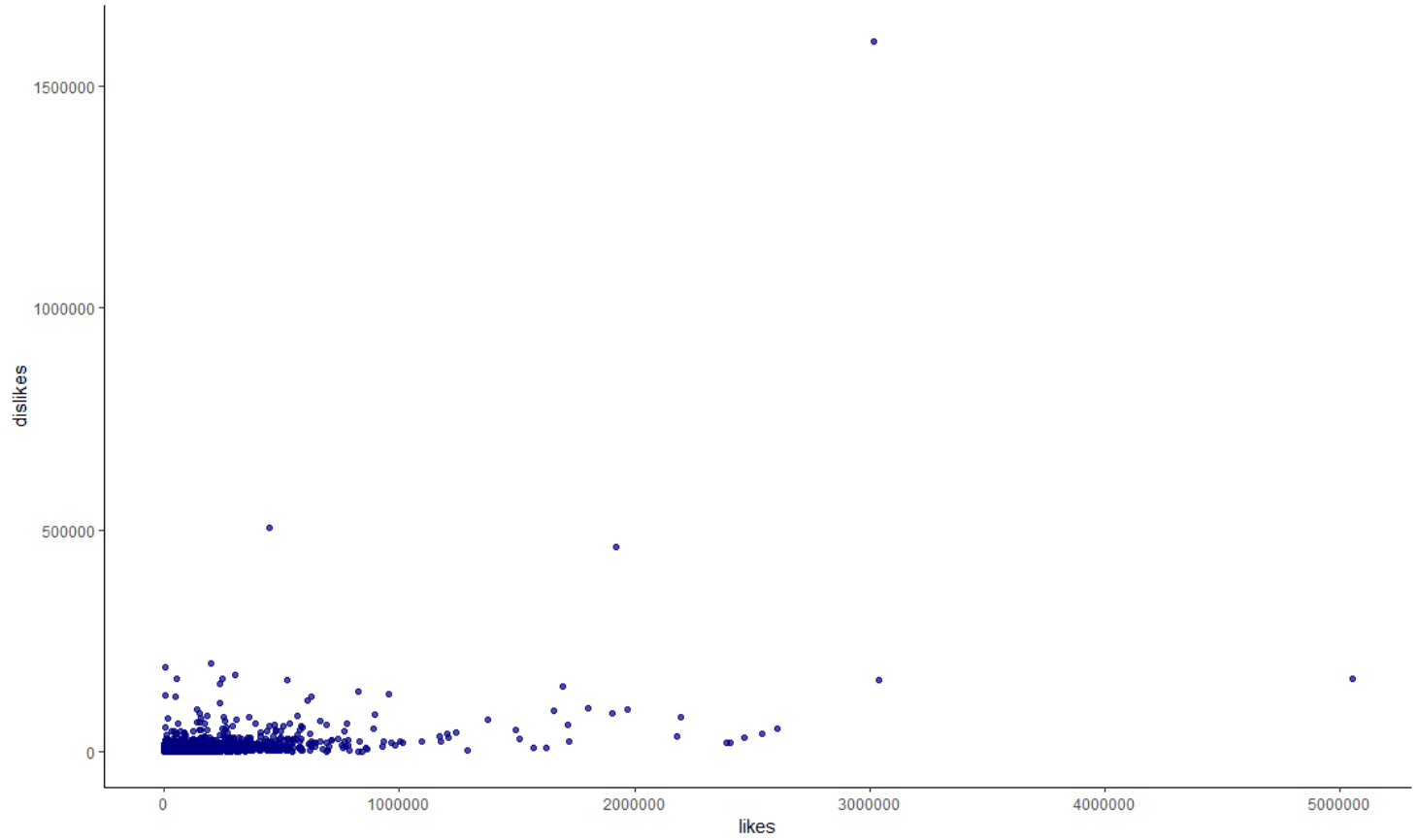
The density plot shows the distribution of dislike percentage in the dataset.



Density is another way to describe the overall connectedness of a graph which focuses on the edges and not the nodes.

f) Scatter Plot (Likes Vs. Dislikes)

The scatter plot depicts success by dislike percentage. The scatter plot shows an extremely weak positive correlations with a number of outliers. This plot shows the videos that were not an instant hit. It can also be inferred from the graph that the number of likes help in determining the days to trending status more than the number of dislikes.

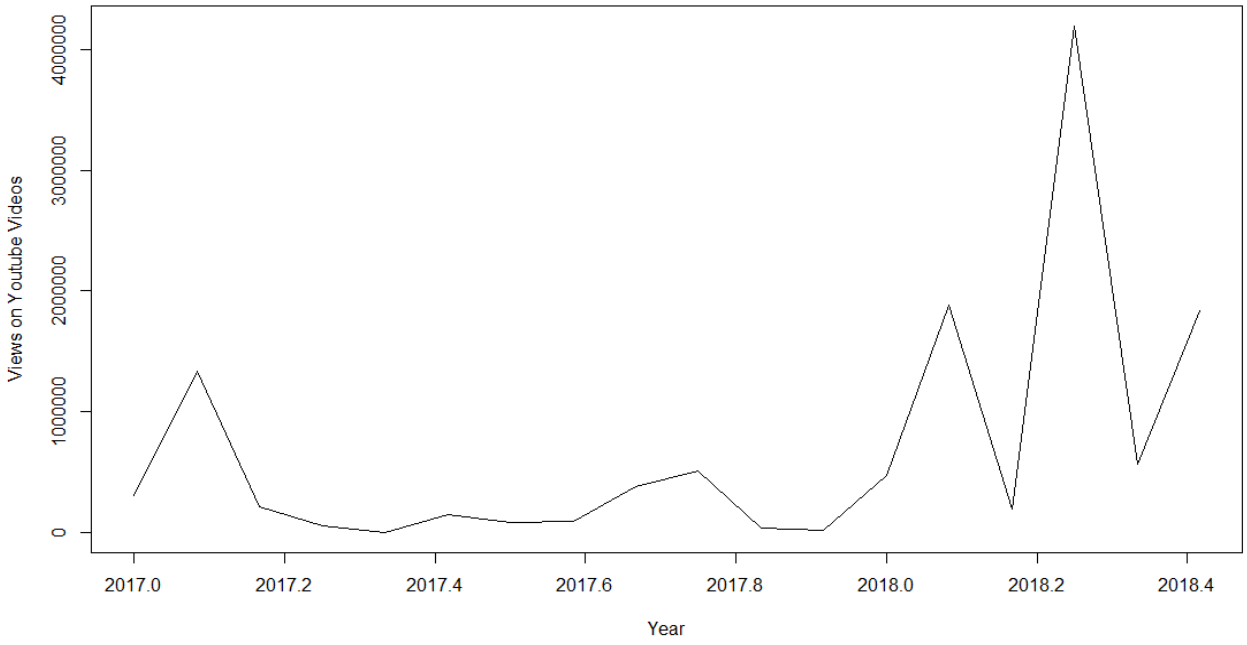


g) Time Series Analysis

Time series analysis accounts for the fact that data points taken over time may have an internal structure that should be accounted for. Time series analysis comprises methods for analyzing time series data in order to extract meaningful statistics and other characteristics of the data. We have performed time series analysis for three variables namely, views, likes and dislikes,

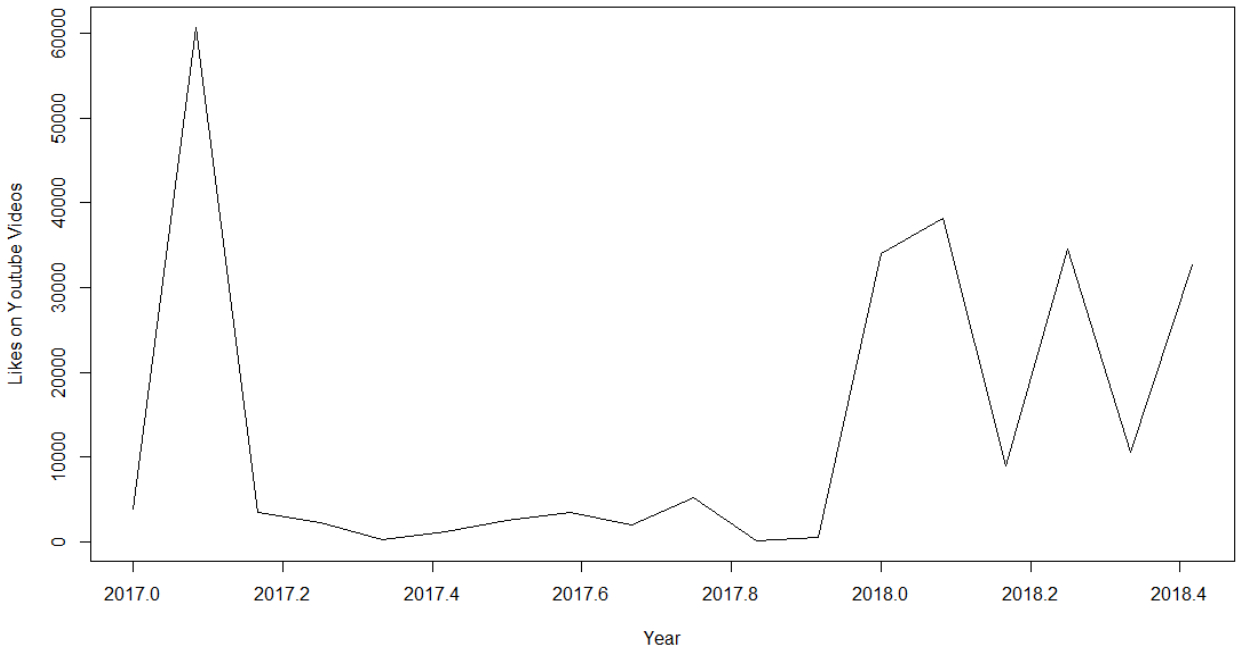
The plot shows time series line graph with zooming in. The ability to zoom in and out of certain areas of the data on a plot is important for revealing patterns and outliers. We are often interested in more detail on areas of dense information or of special interest. Panning refers to the operation of moving the zoom window to other areas.

The views variable on the youtube videos is zoomed in to the two years of the series ranging from 2017 to 2018 is as shown in the graph below.



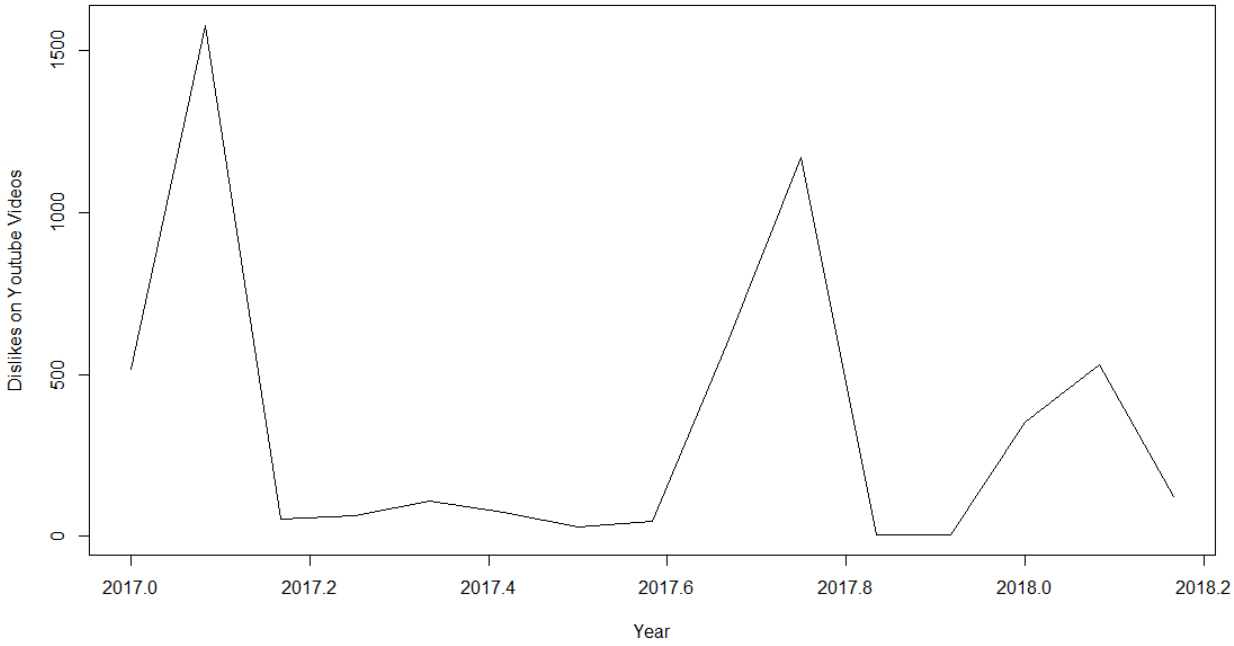
Number of likes over year-

The graph below shows the likes variable which is zoomed in to the two years of the series ranging from 2017 to 2018.



Number of dislikes over year-

The graph below shows the time series analysis for the dislikes on youtube videos that were received by a video over an year.



FINDINGS AND CONCLUSIONS

Assumption:

We have following assumptions while doing analysis and concluding our result:

1. Effect of Number of subscribers with respect to respective channel is not considered. Reason for assuming is that first data is not available for such analysis and second it will bring multicollinearity in the model. Any factor that is not a component of video, but it is contributing for video growth should not be considered.
2. We have decided to do analysis for Canada only. Our conclusion may vary as per different attributes for other countries.
3. "How a video can reach for first time trending" is the out of scope of this project.

Inference:

- The fastest a video makes it to the "trending videos" list the more likely it is for it to increase further in popularity. This means that instant hits are more likely than slow risers to see further growth in views after they enter the "trending videos" list.
- Also, we find that video views tend to drift upwards. This means that trending video views are expected to be higher in 2018 than they were in 2017.

RECOMMENDATIONS

Practical use of Predictive model:

Online channel doesn't know what is required for video to because successful. Our research will help to find this gap understanding. If some YouTube channel wants to make upcoming video more successful, they should plan to make first trending as soon as possible after launch of video. To achieve this, channel can use online marketing, publicity and video promotion events. Once Video make in trending for first time, its probability will increase to grow faster.

Fast growth rate is also dependent on category specific. Fox example, if video is from Entertainment or News & Politics, it's probability of growth will be higher than other less famous category.

CODE BASE

```
#-------------------------------------------------------------------------------#

### 1- DATA PRE-PROCESSING

library(dplyr)

### Transforming our variables
# 1- Frequency , 2- Dislike Percentage, 3- Trending Days

# The objective is to form a table of unique values based on video_id and append 3 columns with variables
for # the above attributes

### Forming the table with unique values based on video_id such

# Reading the dataset
df <- read.csv('CAvideos.csv')
# Sorting as per 'video_id'
dfsort <- arrange(df,video_id)


# Assigning categories to the videos

dfsort$category_id <- ordered(dfsort$category_id,
                        levels = c(1, 2, 10, 15, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 30, 31, 32, 33, 34,
                                35, 36, 37, 39, 40, 41, 42, 43, 44, 38),
                        labels = c("Film & Animation", "Autos & Vehicles", "Music", "Pets & Animals", "Sports",
"Short Movies",
                                "Travel & Events", "Gaming", "Videoblogging", "People & Blogs", "Comedy",
"Entertainment",
                                "News & Politics", "Howto & Style",  "Education", "Science & Technology",
"Movies",
                                "Anime/Animation", "Action/Adventure", "Classics", "Comedy", "Documentary",
"Drama",
                                "Family","Horror", "Sci-Fi/Fantasy", "Thriller", "Shorts", "Shows", "Trailers",
"Foreign"))
View(dfsort)


#changing the column name of category id to category
colnames(dfsort)[5] <- "category"

# Extracting unique values from sorted dataset
dfnew <- dfsort[order(dfsort$video_id, -abs(dfsort$likes), -abs(dfsort$dislikes), -
abs(dfsort$comment_count)), ]
videoUniq <-  dfnew[ !duplicated(dfnew$video_id), ]
```

```
## 1- Frequency of a video [No. of times it trended]
freq.df = count(df, df$video_id)
videoUniq['frequency'] <- freq.df$n

### 2- Dislike Percentage for each video
videoUniq['dislike_percent'] <-  (videoUniq$dislikes) / (videoUniq$likes + videoUniq$dislikes)

## 3- Days to Trending for each video

# Separating 'trending_date' and 'publish_time' into 2 different columns
y.df <- videoUniq[2]
x.df <- videoUniq[6]

View(y.df)
View(x.df)

# Extracting the date from column 'publish_time'
#x.df %>%
x.df <- mutate(x.df, publish_time = as.Date(substr(publish_time, 1, 10)))

# Converting the date-format of 'trending_date' from yyddmm to yymmdd
#y.df %>%
y.df <-  mutate(y.df, trending_date= as.Date(trending_date, format = "%y.%d.%m"))

View(y.df)
View(x.df)


# Combining the new dates into one single dataframe
z.df <- videoUniq[1]
z.df['date_of_publishing'] <- x.df$publish_time
z.df['date_of_trending'] <- y.df$trending_date

View(z.df)

# Finding the number of days between the 2 dates
z.df <- mutate(z.df, No_of_days= as.numeric(date_of_trending-date_of_publishing, units = "days") )

View(z.df)

# So, 'z.df' is the dataframe with the columns that can be added to our original dataset


videoUniq ['date_of_publishing'] <- z.df$date_of_publishing
videoUniq ['date_of_trending'] <- z.df$date_of_trending
videoUniq ['days_to_trend'] <- z.df$No_of_days

#drpping the original trending date column and publish time

videoUniq <- videoUniq[,-c(2)]
```

```
videoUniq <- videoUniq[,-c(5)]
View(videoUniq)

##-----------------------------------------------------------------------------------------------------------------------##

### 2- PRINCIPAL COMPONENT ANALYSIS - PCA

videoUniq1 <- videoUniq

pcs <- prcomp(na.omit(videoUniq1[,-c(1,2,3,4,5,10,11,12,13,14,17,18)]), scale=T)
summary(pcs)
pcs$rot

##-----------------------------------------------------------------------------------------------------------------------##
### 3- CLASSIFICATION TREE

#Classification tree

library(rpart)
library(rpart.plot)
library(rattle)

videoUniq.df <- videoUniq
videoUniq.df <- videoUniq.df[ , -c(1,2,3,5,10,14,17,18)]

default1.ct <- rpart(days_to_trend ~ category + frequency, data = videoUniq.df, method = "class", control =
rpart.control(cp = 0.001))
prp(default1.ct, type = 5, extra = 10, under = TRUE, split.font = 50, varlen = -100)
printcp(default1.ct)
##-----------------------------------------------------------------------------------------------------------------------##

### 4- LINEAR REGRESSION

library(caret)
library(tidyverse)
library(forecast)
library(leaps)
library(dplyr)


# Selecting variables for regression
videoRegVar.df <- videoUniq[, c(6,7,8,9,15,16,19)]

# Data partition
set.seed(123)
training.index <- createDataPartition(videoRegVar.df$days_to_trend, p = 0.9, list = FALSE)
videoReg.train.df <- videoRegVar.df[training.index, ]
videoReg.valid.df <- videoRegVar.df[-training.index, ]

### Run regression
video.lm <- lm(days_to_trend ~ ., data = videoReg.train.df)
```

```
options(scipen = 999)
summary(video.lm)
##-----------------------------------------------------------------------------------------------------------------------------##

### 5- RANDOM FOREST

library(randomForest)

videoUniq_new<- videoUniq[complete.cases(videoUniq), ]
videoUniq1 <- videoUniq_new[, c(6,7,9,8,11,12,13,15,16,19)]

train.index <- createDataPartition(videoUniq1$days_to_trend, p = 0.8, list = FALSE)
train.df <- videoUniq1[train.index,]
valid.df <- videoUniq1[-train.index,]
train.df$days_to_trend <- as.character(videoUniq$days_to_trend)
train.df$days_to_trend <- as.factor(videoUniq$days_to_trend)
rf.reg <- randomForest(days_to_trend ~., data = train.df)
plot(rf.reg, col = "red")

##-----------------------------------------------------------------------------------------------------------------------------##

### 6- ASSOCIATION RULE MINING

assoc1.df <- head(dfsort, 5000)
assoc.df <- assoc1.df[, c(1,2,14,16,18,5,4)]
assoc.df <- assoc.df[complete.cases(assoc.df), ]
assoc2.df <- assoc.df
assoc.df[] <- lapply(assoc.df, as.character)
assoc.df$days_to_trend<- as.numeric(as.character(assoc.df$days_to_trend))
assoc.df$date_of_publishing <-assoc2.df$date_of_publishing

glimpse(assoc.df)

transactionData <- ddply(assoc.df,c("video_id","date_of_publishing"),
                         function(df1)paste(df1$category,
                                            collapse = ","))

transactionData$video_id <- NULL
transactionData$date_of_publishing <- NULL
colnames(transactionData) <- c("items")

library(csv)
library(arules)
library(RColorBrewer)
dev.off()


write.csv(transactionData,"D:/R-Documents/market_basket_transactions.csv", quote = FALSE, row.names =
TRUE)
trans <- read.transactions('D:/R-Documents/market_basket_transactions.csv', format = 'basket', sep=',')
```

```
summary(trans2)

# Create an item frequency plot for the top 15 items

itemFrequencyPlot(trans,topN=15,type="absolute",col=brewer.pal(8,'Pastel2'),
                  main="Absolute Item Frequency Plot",
                  ylab= 'Absolute Frequency of Categories')

itemFrequencyPlot(trans,topN=15,type="relative",col=brewer.pal(8,'Pastel2'),
                  ylab= 'Relative Frequency of Categories',
                  main="Relative Item Frequency Plot")


rules <- apriori(trans, parameter = list(supp = 0.2, conf = 0.5, target = "rules"))
inspect(head(sort(rules, by = "lift")))

##-------------------------------------------------------------------------------------------------------------------------##

### 7- LINEAR DISCRIMINANT ANALYSIS - LDA

library(MASS)
library(dplyr)
library(ggplot2)

## Fit the model
video1<- videoRegVar.df[order(videoRegVar.df$days_to_trend),]
lda_1 <- lda(days_to_trend~.,data=video1)

##Compute LDA
lda_1

## Make Predictions
predict_1 <- predict(lda_1,data=video1)

## Model Accuracy
mean(predict_1$class==video1$days_to_trend)

## LDA plot using ggplot 2
lda_1 <- cbind(video1[1:24229, ], predict_1$x)

ggplot(lda_1, aes(LD1, LD2)) +
  geom_point(aes(color = days_to_trend))

# videos having less than 8 days_to_trend
fil <- filter(video1, days_to_trend < 8)

ggplot(aes(x=frequency,y=days_to_trend), data = fil)+
  geom_boxplot()

##-------------------------------------------------------------------------------------------------------------------------##
```

```
### 8- VISUALIZATIONS
##Most influential creators
creators.df <- table(videoUniq$channel_title)

barchart(head(sort(creators.df, decreasing = TRUE),15))


##Video Category Distribution
videoUniq$category_id <- ordered(videoUniq$category)

barchart(table(videoUniq$category))


##Time Series Analysis (number of Views over year)
view.ts <- ts(videoUniq$views, start = c(2017, 1), end = c(2018, 6), freq = 12)
plot(view.ts, xlab = "Year", ylab = "Views on Youtube Videos")
##Time Series Analysis (number of likes over year)
like.ts <- ts(videoUniq$likes, start = c(2017, 1), end = c(2018, 6), freq = 12)
plot(like.ts, xlab = "Year", ylab = "Likes on Youtube Videos")
##Time Series Analysis (number of dislikes over year)
dislike.ts <- ts(videoUniq$dislikes, start = c(2017, 1), end = c(2018, 3), freq = 12)
plot(dislike.ts, xlab = "Year", ylab = "Dislikes on Youtube Videos")

## Scatter plot to find out correlation between likes and views for videos
ggplot(videoUniq) +
  geom_point(aes(x = views, y = likes), color = "navy", alpha = .7) +
  theme_classic()


##Days to trending status
barchart(table(videoUniq$days_to_trend))

# Filled Density Plot (Distribution for dislike percentage)
ggplot(videoUniq, aes(x=dislike_percent)) +
  geom_histogram(aes(y=..density..),
                 binwidth=.5,
                 colour="black") +
  geom_density(alpha=.2, fill="#FF6666")

## Not Instant Hits : Success by dislike percentage
ggplot(videoUniq) +
  geom_point(aes(x = likes, y = dislikes), color = "navy", alpha = .7) +
  theme_classic()

##-------------------------------------------------------------------------------------------------------------------------------##

### 9- DATA MINING AND SENTIMENT ANALYSIS

library(tidyverse)
library(tidytext)
library(glue)
```

```
library(stringr)
library(SnowballC)
library(tm)
library(wordcloud)
library(wordcloud2)
library(gridExtra) #viewing multiple plots together
library(RColorBrewer)


video_uniq_mine <- head(videoUniq$tags,5000)

str_replace_all(video_uniq_mine, "[^[:alnum:]]", " ")

video_uniq_mine <- iconv(video_uniq_mine, 'UTF-8', 'ASCII')


corpus <- Corpus(VectorSource(video_uniq_mine))
inspect(corpus[1:5])

# Convert the text to lower case
corpus <- tm_map(corpus, tolower)

# Remove punctuations
corpus <- tm_map(corpus, removePunctuation)

# Remove numbers
corpus <- tm_map(corpus, removeNumbers)

# Remove english stopwords
cleanset <- tm_map(corpus, removeWords, stopwords('english'))

# Remove URL and foreign language urls'
removeURL <- function(x) gsub('http[[:alnum:]]*', '', x)
cleanset <- tm_map(cleanset, content_transformer(removeURL))

# Eliminate extra white spaces
cleanset <- tm_map(cleanset, stripWhitespace)

# Remove additional stopwords
cleanset <- tm_map(cleanset, removeWords, c('ã˜â','ã™â€','ã\u0090â','ã™â€žã˜â','ã™å','ã™ë','â\u0081ã',
                                            'noahã','â€žã','â€˜ã','ã™â€ž','â\u008dã','ùšø','ùˆø','ùƒø','ùšù',
                                            'ùˆù','ñ€ð','ù\u0081ø','ùšùˆø','ñ\u0081ð','ûœø','ù\u0081ùšø',
                                            'noahå','ùƒù','ukur','˜çš','ù\u0081ù','noahé','æžœçˆ','ùˆù\u0081ø',
                                            'espnespn','ñœð','ùšùˆù','è\u0081žå','ñƒð','â€\u009d','â€žÃ','Â\u0081Ã',
                                            'â€šÃ','â€˜Ã','â€\u009dÃ ','â€ž','noahÃ','Â\u008dÃ','ËœÃ','Â\u0090Ã',
                                            'â€žÂ','Â\u009dÃ','Â\u008fÃ','â€\u009dÂ','Â\u009dÂ','â€šÂ','â€œÃ',
                                            'Â\u008fÂ','Â\u0090Â ','Â\u0081Â ','â€™Ã','Â\u008dÂ','â€žâ€',
                                            'Â\u009dÅ','Â\u008dâ€','ËœÂ','Ëœâ€','ËœÅ','â€œÂ','Â\u008dÅ','â€˜Â',
                                            'Â\u0090Å','â€š','Â\u0081â€','â€˜ËœÃ','thÃ','Â\u0081Å','â€™Â','Â\u0081â€œÃ',
                                            'ËœÂ\u0081Ã','Â\u008dÂ\u0081Ã'))
```

```
# Text stemming (reduces words to their root form)
cleanset <- tm_map(cleanset, stemDocument)


View(cleanset)
inspect(cleanset[1:10])


dtm <- TermDocumentMatrix(cleanset)
m <- as.matrix(dtm)
v <- sort(rowSums(m),decreasing=TRUE)
d <- data.frame(word = names(v),freq=v)
head(d, 200)

# Generate the WordCloud
par(bg='mistyrose')
png(file="WordCloud.png",width=1000,height=1000, bg="grey30")
wordcloud(d$word, d$freq, max.words = 200, col=terrain.colors(length(d$word), alpha=0.9),
          random.order=FALSE, rot.per=0.3 )
title(main = "Most words in tags", font.main = 1, col.main = "cornsilk3", cex.main = 1.5)
dev.off()
title(main = "Most words in tags", font.main = 1, col.main = "cornsilk3", cex.main = 1.5)
dev.off()

#finding frequency of words

findFreqTerms(dtm, lowfreq = 4)

#finding association between frequent terms
findAssocs(dtm, terms = "beautiful", corlimit = 0.3)

#barplot of different words with frequency

barplot(d[1:10,]$freq, las = 2, names.arg = d[1:10,]$word,
        col ="lightblue", main ="Most frequent words",
        ylab = "Word frequencies")

#Sentiment Analysis
library(syuzhet)
library(lubridate)
sentiment <- iconv(cleanset)
s. <- get_nrc_sentiment(sentiment)
barplot(colSums(s.), las=2, col= rainbow(10), ylab= 'Count', main = 'Youtube Video Tags')

##-------------------------------------------------------------------------------------------------------------------------##
```

CITATIONS

Website

[1] Kaggle Inc. 2018. Trending YouTube Video Metadata analysis. https://www.kaggle.com/yanpapadakis/trending-youtube-video-metadata-analysis

[2] A. Kassambara. 2017. Text mining and word cloud fundamentals in R : 5 simple steps you should know. http://www.sthda.com/english/wiki/text-mining-and-word-cloud-fundamentals-in-r-5-simple-steps-you-should-know

[3] DataCamp Inc. 2018. Market Basket Analysis using R. https://www.datacamp.com/community/tutorials/market-basket-analysis-r#firsthead

[4] Wikipedia, 2018. Text mining. https://en.wikipedia.org/wiki/Text_mining

[5] Wikipedia, 2018. Text mining https://en.wikipedia.org/wiki/Decision_tree_learning

[6] Machine Learning Mastery. 2018 Classification And Regression Trees for Machine Learning. https://machinelearningmastery.com/classification-and-regression-trees-for-machine-learning/

[7] DataSciencePlus.com. 2018. Advanced Modeling Random Forests in R. https://datascienceplus.com/random-forests-in-r/

[8] Naval Postgraduate School.2018 Top of Form https://faculty.nps.edu/sebuttre/home/R/factors.html

[9] Dalhousie University. 2018. https://www.mscs.dal.ca/Splus/part5.html

[10] Software Carpentry .2018.Programming with R.

https://swcarpentry.github.io/r-novice-inflammation/12-supp-factors/

BOOKS

[1] Julia Silge and David Robinson.2018. Text Mining with R. https://www.tidytextmining.com/

[2] Galit Shmueli, Peter C. Bruce, Inbal Yahav, Nitin R. Patel, Kenneth C. Lichtendahl, Jr. DATA MINING FOR BUSINESS ANALYTICS. 2018 John Wiley & Sons, Inc.

INDIVIDUAL REPORT

Fall 2018
Subject: BUAN 6356: Business Analytics with R
Self-Contribution report on "**YouTube Trending Video Metadata Analysis"**

As a first project of MS in BA, I was very excited to learn and explore how analytics report works. I have learned a lot during project. We have started meeting and invested enough for deciding Scope of project and Problem statement. We have decided to explore YouTube trending data file for Canada. Once we have decided problem statement and planning of project, we divided the task and study topic. I was fortunate to work with such dynamics people. Mohit has a great skill in coding and in planning project. Drishya has contributed in algorithm exploration, deciding output or conclusion and documentation. Sushant and Sayali are great team-mates and contributed their part in text mining, EDA and Algorithm. I have learned many skills from my team-mates during project discussion. We have faced challenges to fix meeting time, deciding algorithm and coordination in thanks-giving week.

Contribution of each group member are as follow:
1. Vaibhav: Scope of Project, Problem statement, Reading JSON file, extracting categories name, PCA, Classification tree
2. Mohit : Scope of project, Exploratory Data Analysis, Text Mining, Sentiment Analysis
3. Sushant: Linear Discriminant Analysis, Data Visualizations.
4. Drishya: Data Pre-processing, Part of LDA, Linear Regression, Random Forest, Association Rule Mining
5. Sayali : Data Cleaning, Data pre-processing, Exploratory Data Analysis, Data Visualizations

We were successful in deciding problem statement in very short time. This help us to lead the project in right direction. We have faced challenge while extracting data from JSON file for channel category. Then we use level and factor function to replace with category name. In Classification tree model, we have made tree, but it was either very complex considering important variables or quite simple with 2 variables. This doesn't lead us to conclude the rule.

In conclusion, we did mining and data exploration. We spend reasonable time to understand data and to do brain-storming. We have concluded that the fastest a video makes it to the "trending videos" list the more likely it is for it to increase further in popularity. This means that instant hits are more likely than slow risers to see further growth in views after they enter the "trending videos" list. So, coming for first time in trending is very important for fast growing of video.