

A Project Report On

YOUTUBE TRENDING VIDEOS

Submitted in partial fulfillment of the requirement for the
award of the degree

MASTER OF COMPUTER APPLICATIONS
(INTEGRATED)

from

Gujarat Technological University



Academic Year 2019 - 20

Krunal Bhandari(165220686004)
Yash Dhameliya (165220686009)

Internal Guide

Pro. Sunil Bajaja



Marwadi Education Foundation's Group of Institutions
(MEFGI)

Rajkot-Morbi Road, At & PO :Gauridad, Rajkot 360 003. Gujarat. India.



Marwadi
education foundation

Faculty of Computer Applications (FCA)

Certificate

This is to certify that the project work entitled
YouTube Trending Videos
submitted in partial fulfillment of the requirement for
the award of the degree of
Master of Computer Applications
(Integrated)
of the
Gujarat Technological University
is a result of the Bonafede work carried out by
Krunal Bhandari(165220686004)
Yash Dhameliya (165220686009)
during the academic year 2019 – 2020

Faculty Guide

HOD

External Viva

Name of the Examiners

Signature with Date

CERTIFICATE

DECLARATION

We hereby declare that this project work entitled **YouTube Trending Videos** is a record done by us.

We also declare that the matter embodied in this project is genuine work done by me and has not been submitted whether to this University or to any other University / Institute for the fulfillment of the requirement of any course of study.

Place: _____

Date: _____

Krunal Bhanderi (165220686004) Signature: _____

Yash Dhameliya (165220686009) Signature: _____

ACKNOWLEDGEMENT

It is indeed a great pleasure to express our thanks and gratitude to all those who helped us. No serious and lasting achievement or success one can ever achieve without the help of friendly guidance and co-operation of so many people involved in the work.

We are very thankful to our guide **Dr. Sunil Bajaja**, the person who makes us to follow the right steps during a project work. We express our deep sense of gratitude to for her guidance, suggestions and expertise at every stage. A part from that her valuable and expertise suggestion during documentation of our report indeed help us a lot.

Thanks to our friend and colleague who have been a source of inspiration and motivation that helped to us during our project work.

We would heartily thankful to Head and Dean of our department **Dr. Sunil Bajaja and Dr. R. Sridaran** to give us an opportunity to work over this report and for their end-less and great support. And to all other people who directly or indirectly supported and help us to fulfil our task.

Krunal Bhanderi (165220686004) Signature: _____

Yash Dhameliya(165220686009) Signature: _____

CONTENTS

Chapters	Particulars	Page No.
1	INTRODUCTION	1
2	PROBLEM STATEMENT	2
3	ORGANIZATION OF DATA	3
4	DATA VISULIZATION THROUGH SUMMARY	4
5	METHOD/TECHNIQUES	
5.1	Identification of technique and tool used	5
5.2	Application of techniques and tools used	6
5.3	Discuss the assumptions that inference procedure has	6
5.4	Discuss how you will check those assumptions	6
6	PRILIMINARY ANALYSIS	7-11
7	GRAPHICAL DESCRIPTION OF DATA	12-49
8	CONCLUSION	50
9	BIBLIOGRAPHY	51

1. INTRODUCTION

YouTube (the world-famous video sharing website) maintains a list of the top trending video on the platform. **According to Variety magazine**, “To determine the year’s top-trending videos, YouTube uses a combination of factors including measuring users’ interactions (number of views, comments and likes). Note that they’re not the most-viewed videos overall for the calendar year”.

This dataset is a daily record of the top trending YouTube videos.

Data includes the video id, video title, channel title, publish time, tags, views, likes and dislikes, thumbnail, comment disable, rating disable, video error or removed, description, and comment count.

2. PROBLEM STATEMENT

1. What is relation between video likes, dislike and ranking?
2. What is relation between comment count and ranking?
3. What is relation between video view and ranking?
4. Do tags of video affect the video ranking?
5. Effects of video publish time on video ranking.
6. Percentage of videos that being deleted or face some error.
7. Percentage of videos that comment and ratings were disable.

3. ORGANIZATION OF DATA

This dataset contains information of athlete events from 1998 to 2016, Which amounts to 12826 athlete events total. It contains 12826 observation of 14 variables, which

This database contain information of trending videos on YouTube. It contain 40,000 observation on 15 different values which include the video id, video title, channel title, publish time, tags, views, likes and dislikes, thumbnail, comment disable, rating disable, video error or removed, description, and comment count.

1. video_id:

This field contains unique video id.

2. trending_date:

This field contains the date on which date specific video was in trending.

3. Title:

This field contains the video title.

4. channel_title:

This field contains the channel name related to video.

5. publish_time:

This field contains the date of publish including time.

6. Tags:

This field contains the video tags that helps to search related videos.

7. Views:

This field contains total number of views of videos.

8. Likes:

This field contains total number of likes given to video.

9. Dislikes:

This field contains total number of dislikes given to video.

10. comment_count:

This field contains total count of comments given to the video.

11. thumbnail_link:

This field contains the link of cover image of video uploaded on YouTube.

12. comments_disabled:

It shows the comment section is enabled or disabled.

13. ratings_disabled:

It shows the ratings are enabled or disabled for paid content.

14. video_error_or_removed:

It shows the video status for check the video exists or not and also check whether the video has some errors or not.

15. Description:

This field shows the description of video.

16. Category_id:

Specify the category of video like Music, Movies, Sports etc.

4. DATA VISUALISATION THROUGH SUMMARY

1. How did I obtain the data set?

- I got the dataset from kaggle website. It contain information about trending videos on YouTube.

2. How the data originally collected?

- Data is collected from YouTube website that maintains the record of all trending videos on YouTube.

3. How many rows and columns are in the dataset?

- There are 40882 rows and 15 columns.

4. What are the columns in dataset?

```
> View(CAvideos)
> names(CAvideos)
[1] "video_id"          "trending_date"      "title"              "channel_title"
[5] "publish_time"      "tags"               "views"              "likes"
[9] "dislikes"          "comment_count"      "thumbnail_link"      "comments_disabled"
[13] "ratings_disabled"  "video_error_or_removed" "description"
> |
```

1. METHODS / TECHNIQUES

i. Identification of techniques and tool used

R is a programming language and software environment for statistical analysis, graphics representation and reporting. R was created by Ross Ihaka and Robert Gentleman at the University of Auckland, and is currently developed by the R Development Core Team. R is freely available under the GNU General Public License, and pre-compiled binary versions are provided for various operating systems like Linux, Windows and Mac. This programming language was named R, based on the first letter of first name of the two R authors (Robert Gentleman and Ross Ihaka), and partly a play on the name of the Bell Labs Language S.

As stated earlier, R is a programming language and software environment for statistical analysis, graphics representation and reporting. The following are the important features of R

- R is a well-developed, simple and effective programming language which includes condition, loops, user defined recursive functions and input and output facilities.
- R has an effective data handling and storage facility.
- R provides a suite of operators for calculations on array, lists, vectors and matrices.
- R provides a large, coherent and integrated collection of tools for data analysis.
- R provides graphical facilities for data analysis and display either directly at the computer or printing at the papers.

As a conclusion, R is world's most widely used statistics programming language. It's the #1 choice of data scientists and supported by a vibrant and talented community of contributors. R is taught in universities and deployed in mission critical business applications. This tutorial will teach you R programming along with suitable examples in simple and easy steps.

ii. Application of techniques and tools used

- R will provide numerical or graphical summaries of data
- R has extensive graphical abilities
- R will handle a variety of specific analyses : Correlation, Straight Line Regression

iii. Assumption

According to the Research and Innovative Technology Administration Bureau of Sports Statistics, this athlete dataset includes information about every games that departed from New York City in 2016. Therefore, due to the size and amount of data collected here, we can make the assumption that this dataset exhibits randomization without any bias.

Randomize : What is the Randomization Scheme?

While our original assumption claimed that the entire sports dataset exhibits randomization, our analysis needed to ensure that we developed a completely randomization design. In meeting this objective, a new dataset is created (“athlete”) that randomly selects 12826 observations from “data_raw”. In creating this new dataset, we’re ensuring that our analysis considers a large sample of an even larger population and randomizes the order that the runs of the data are placed in the dataset (since they were originally listed chronologically by departure date.) After creating this new dataset, we can now assume that our randomization scheme represents a completely randomized design.

```
> views1<-data_clean[a1.views,]  
> data_clean<-na.omit(a1)  
> a1.views=sample(1:nrow(data_clean),5000,replace=FALSE)  
> views1<-data_clean[a1.views,]  
> |
```

2. PRELIMINARY ANALYSIS

- The number of rows and number of columns in the dataset:

```
> nrow(data)
[1] 40881
> ncol(data)
[1] 15
> |
```

- The first six value in dataset

	video_id	trending_date	channel_title	publish_time	views
1	n1WpP7iowLc	17.14.11	EminemVEVO	2017-11-10T17:00:03.000Z	17158579
2	OdBIkQ4Mz1M	17.14.11	iDubbbzTV	2017-11-13T17:00:00.000Z	1014651
3	5qpjK5DgCt4	17.14.11	Rudy Mancuso	2017-11-12T19:05:24.000Z	3191434
4	d380meDOWOM	17.14.11	nigahiga	2017-11-12T18:01:41.000Z	2095828
5	2Vv-BfVoq4g	17.14.11	Ed Sheeran	2017-11-09T11:04:14.000Z	33523622
6	0yIWz1XEeyc	17.14.11	DramaAlert	2017-11-13T07:37:51.000Z	1309699

	likes	dislikes	comment_count	comments_disabled	ratings_disabled	video_error_or_remove
	787425	43420	125882	FALSE	FALSE	FALS
	127794	1688	13030	FALSE	FALSE	FALS
	146035	5339	8181	FALSE	FALSE	FALS
	132239	1989	17518	FALSE	FALSE	FALS
	1634130	21082	85067	FALSE	FALSE	FALS
	103755	4613	12143	FALSE	FALSE	FALS

YOUTUBE TRENDING VIDEOS

- The last six values in dataset

video_id	trending_date	channel_title	publish_time	views
40876 7E1np354AEc	18.14.06	Ozuna	2018-06-13T23:53:29.000Z	201847
40877 sGo1xsMSGfQ	18.14.06	Annoying Orange	2018-06-13T18:00:07.000Z	80685
40878 8HNuRNi8t70	18.14.06	Elhiwar Ettounsi	2018-06-13T19:01:18.000Z	103339
40879 GWlKEM3m2EE	18.14.06	Kingdom Hearts	2018-06-11T17:30:53.000Z	773347
40880 1bMKLzQ4cNQ	18.14.06	The Young Turks	2018-06-13T04:00:05.000Z	115225
40881 POTgw38-m58	18.14.06	VH1	2018-06-13T16:00:03.000Z	107392

likes	dislikes	comment_count	comments_disabled	ratings_disabled	video_error_or_removed
1568	407	537	FALSE	FALSE	FALSE
1701	99	1312	FALSE	FALSE	FALSE
460	66	51	FALSE	FALSE	FALSE
25900	224	3881	FALSE	FALSE	FALSE
2115	182	1672	FALSE	FALSE	FALSE
300	62	251	FALSE	FALSE	FALSE

- Names of variables:

```
names(a1)
[1] "video_id"      "trending_date"  "title"          "channel_title"
[5] "publish_time"  "tags"           "views"          "likes"
[9] "dislikes"      "comment_count"  "thumbnail_link" "comments_disabled"
[13] "ratings_disabled" "video_error_or_removed" "description"
```

- Factors and levels of variable “comment_disabled”:

```
g<-as.factor(a1$comments_disabled)
levels(g)
[1] "FALSE" "TRUE"
```

- Factors and levels of variable “trending_date”:

```
> levels(trending)
[1] "17.01.12" "17.02.12" "17.03.12" "17.04.12" "17.05.12" "17.06.12" "17.07.12"
[17] "17.15.12" "17.16.11" "17.16.12" "17.17.11" "17.17.12" "17.18.11" "17.18.12"
[33] "17.23.12" "17.24.11" "17.24.12" "17.25.11" "17.25.12" "17.26.11" "17.26.12"
[49] "18.01.01" "18.01.02" "18.01.03" "18.01.04" "18.01.05" "18.01.06" "18.02.01"
[65] "18.03.05" "18.03.06" "18.04.01" "18.04.02" "18.04.03" "18.04.04" "18.04.05"
[81] "18.06.03" "18.06.04" "18.06.05" "18.06.06" "18.07.01" "18.07.02" "18.07.03"
[97] "18.09.02" "18.09.03" "18.09.05" "18.09.06" "18.10.02" "18.10.03" "18.10.05"
[113] "18.12.06" "18.13.01" "18.13.02" "18.13.03" "18.13.05" "18.13.06" "18.14.01"
[129] "18.15.05" "18.16.01" "18.16.02" "18.16.03" "18.16.04" "18.16.05" "18.17.01"
[145] "18.19.01" "18.19.02" "18.19.03" "18.19.04" "18.19.05" "18.20.01" "18.20.02"
[161] "18.22.02" "18.22.03" "18.22.04" "18.22.05" "18.23.01" "18.23.02" "18.23.03"
[177] "18.25.03" "18.25.04" "18.25.05" "18.26.01" "18.26.02" "18.26.03" "18.26.04"
[193] "18.28.04" "18.28.05" "18.29.01" "18.29.03" "18.29.04" "18.29.05" "18.30.01"
```

- Structure of dataset

```
> str(a1)
'data.frame': 40881 obs. of 15 variables:
 $ video_id      : Factor w/ 24103 levels "--45ws7CEN0",...: 14218 585 2730 6676 1593 846 386 1464 1191
 $ trending_date : Factor w/ 205 levels "17.01.12","17.02.12",...: 14 14 14 14 14 14 14 14 14 ...
 $ title         : Factor w/ 24573 levels "'Gala Artis 2018' Le numÃero d'ouverture",...: 7885 16652
 $ channel_title : Factor w/ 5076 levels "-Wen Zhao Officialæ-þæ\230è«þå\217ð«-ä»$",...: 1430 2067 375
 $ publish_time  : Factor w/ 23613 levels "2008-01-13T01:32:16.000Z",...: 68 260 180 172 55 228 205 269
 $ tags          : Factor w/ 20157 levels "'Hace|minuto'|"hace|minutos'|"MÃquinas'|"animales'":
14460 15238 5760 19 6960 15781 11001 7336 ...
 $ views         : Factor w/ 40169 levels "1000","1000086",...: 40169 NA 20445 13047 21272 4707 19261 3
 $ 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 9
 $ 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/ 22345 levels "", "' vous Ãates dÃÃjÃ allÃÃ en Musulmanie ? '", Le ramad
520 20875 8951 6118 734 20404 15743 10586 15785 ...
```

- Data Cleansing : omit NA cases using the na.omit function and recode the “views” column using indexing and reassignment

```
a2<-na.omit(a1)
log<-a1$views=="0"
a2$views[log]<- "No"
```


- What was the mean, median and standard deviation of views?

```
> mean(a1.views)
[1] 20455.67
> median(a1.views)
[1] 20467.5
> sd(a1.views)
[1] 11721.28
```

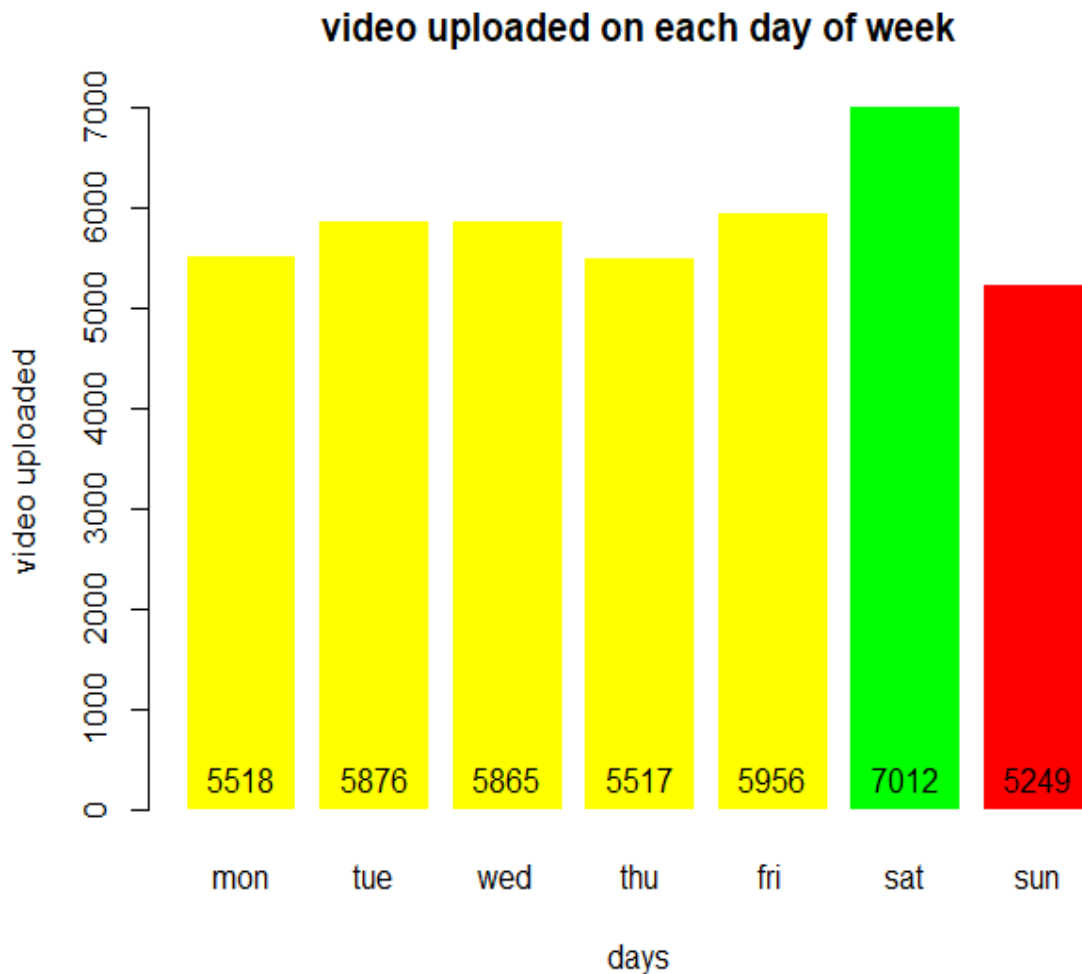
The mean views is 20455.67 views with a standard deviation of 11721.28 views. The median is at 20467.5 views.

3. GRAPHICAL DESCRIPTION OF DATA

- Reading CSV

```
a1=read.csv(file = "c:/krunal/CAvideos.csv")
```

- Bar graph of video uploads on each day of week



```
install.packages("lubridate")
sun=0
mon=0
tue=0
wed=0
thu=0
fri=0
sat=0
library(lubridate)
for (i in a1["publish_time"])
{
```

```

for (b in i)
{
  c=strsplit(b,"T")[[1]]
  date=ymd(c[1])
  a=wday(date)
#
switch(a,mon=mon+1,tue=tue+1,wed=wed+1,thu=thu+1,fri=fri+1,sat=sat+1,sun=sun+1)
  if(a==1)
  {
    mon=mon+1
    print(mon)
  }else if(a==2){
    tue=tue+1
  }else if(a==3){
    wed=wed+1
  }else if(a==4){
    thu=thu+1
  }else if(a==5){
    fri=fri+1
  }else if(a==6){
    sat=sat+1
  }else if(a==7){
    sun=sun+1
  }
  print(count)
  count=count+1

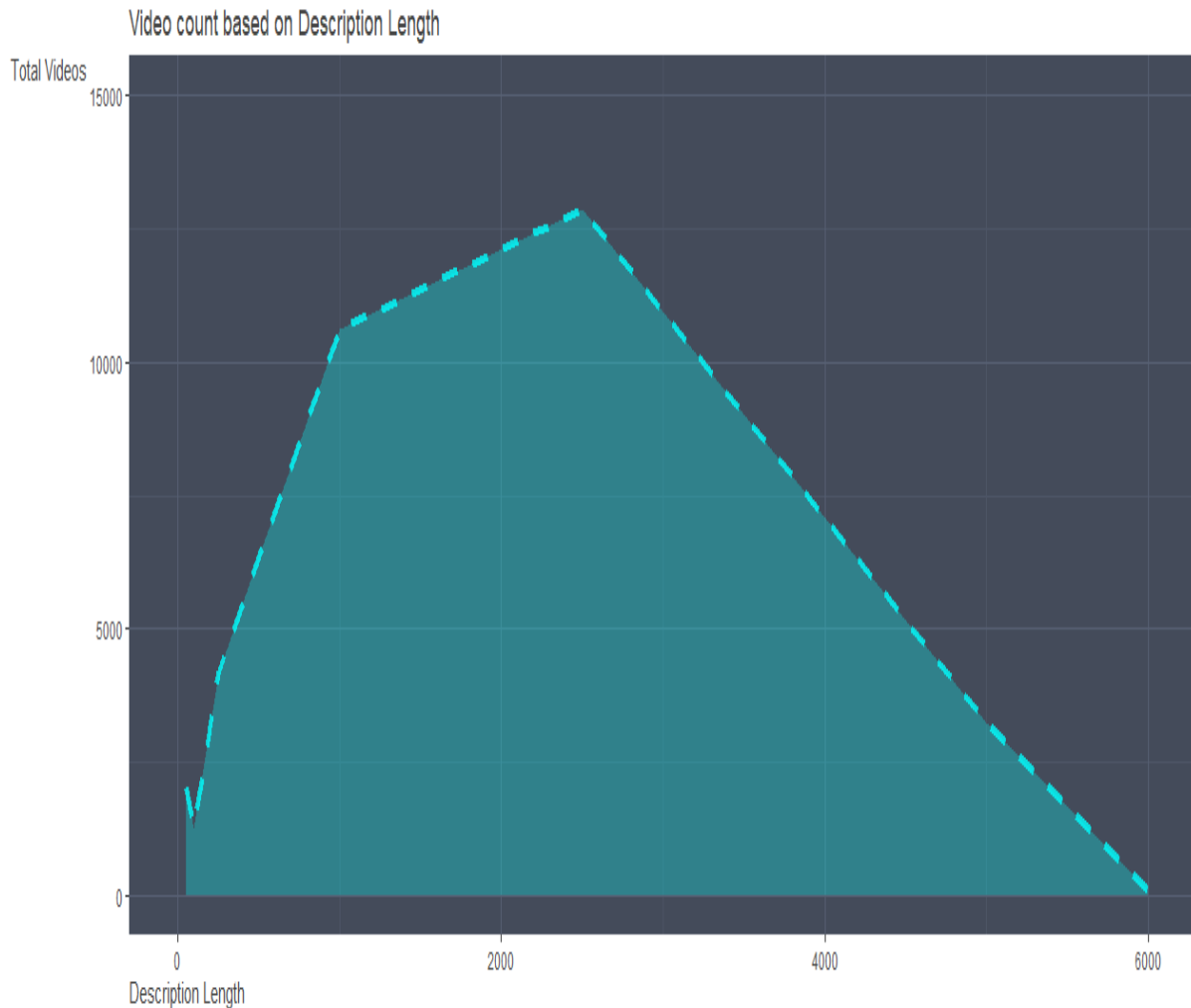
}
}

install.packages("lubridate")
date<-ymd("2019-01-12")
print(wday(date))

da=c(mon,tue,wed,thu,fri,sat,sun)
da1=c("mon","tue","wed","thu","fri","sat","sun")
f<-barplot(da,main="video uploaded on each day of week",xlab = "days",ylab = "video
uploaded",args.legend=list(bty="n",horiz=TRUE)      ,      border="white",names.arg      =
da1,col="yellow")
f

```

- **Area graph of video count based on description length**



```
install.packages("gganimate")
install.packages("babynames")
install.packages("grbrthemes")
library(ggplot2)
library(gganimate)
library(babynames)
library(hrbrthemes)

count=110=120=130=140=150=160=170=180=190=1100=0
for (i in a1$description)
{
  for (c in i)
  {
    b=nchar(c)

    print(b)
    if(b<50)
    {
      110=110+1
    }
  }
}
```

```

    }else if(50<b & b<100){
        l20=l20+1
    }else if(100<b & b<250){
        l30=l30+1
    }else if(250<b & b<500){
        l40=l40+1
    }else if(500<b & b<1000){
        l50=l50+1
    }else if(1000<b & b<2500){
        l60=l60+1
    }else if(2500<b & b<5000){
        l70=l70+1
    }else if(5000>b){
        l80=l80+1
    }

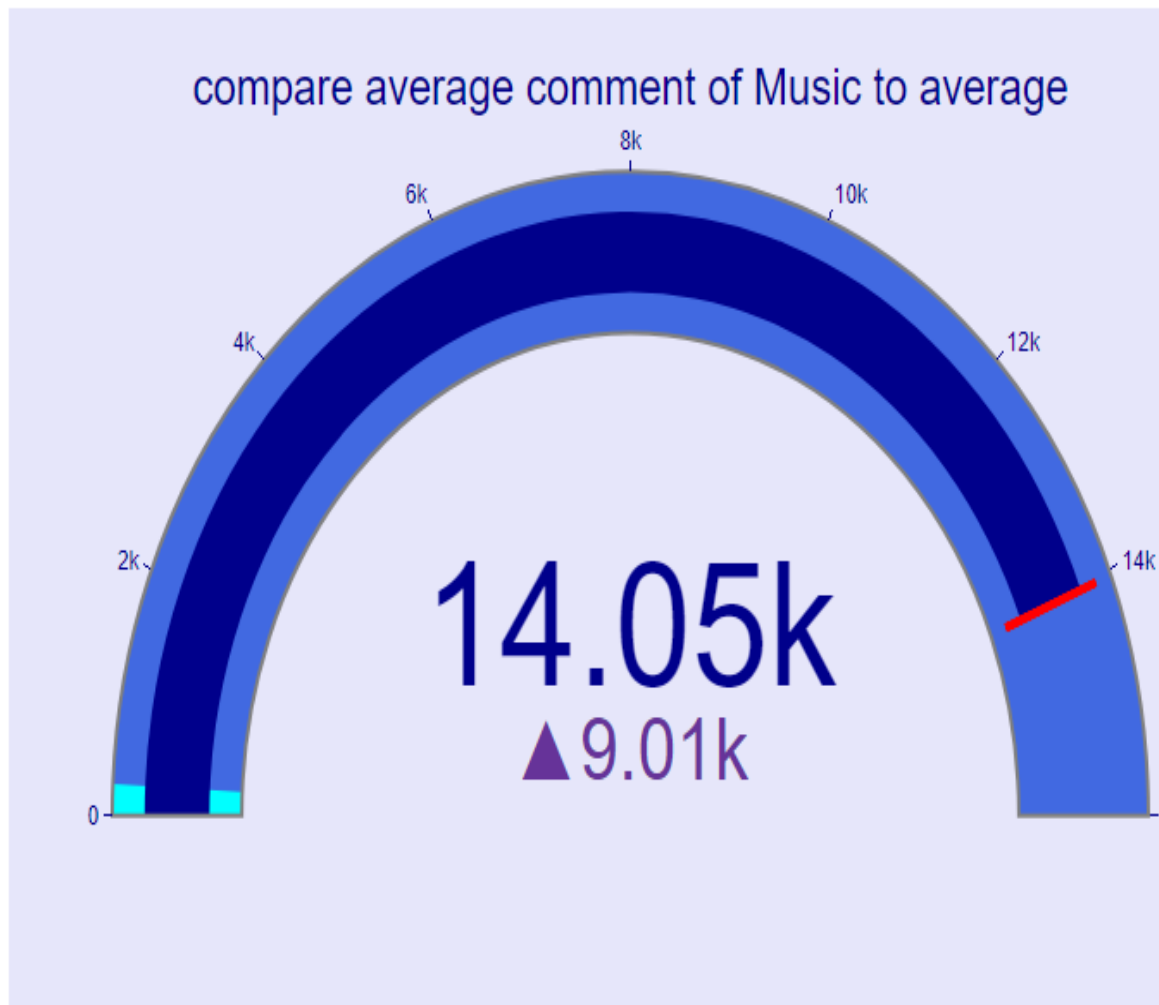
    print(count)
    count=count+1
}
}

y=c(110,l20,l30,l40,l50,l60,l70,l80)
x=c(50,100,250,500,1000,2500,5000,6000)

c="hello"
data=data.frame(x,y)
data
x<-ggplot(data,aes(x=x,y=y,group=1))+ geom_line(color="cyan", size=1.5, alpha=.8,
linetype=8)+labs(x="Description Length",y="Total Videos", title = "Video count based on
Description Length")+expand_limits(x=c(0,6000),y=c(0,15000))
+geom_area(fill="cyan",alpha=0.3)+theme(text = element_text(family = 'Gill Sans',colour =
"#444444"),panel.background = element_rect(fill = '#444B5A'),panel.grid.minor =
element_line(color='#4d5566'),panel.grid.major = element_line(color = '#586174'),axis.title.y
= element_text(vjust = 1,angle = 0),axis.title.x = element_text(hjust = 0))
x

```

- Gauge chart comparing average comment of music to the average comment



```
library(plotly)
library(sqldf)
```

```
a=sqldf("select sum(comment_count),count(video_id),category_id from a1 group by category_id")
```

```
a[,1]
```

```
a[,2]
```

```
b=52435252/3731
```

```
b
```

```
a1[,1]/a[,2]
```

```
b1=sum(a[,1])/sum(a[,2])
```

```
b1
```

```
b1=sum(a[,1])
```

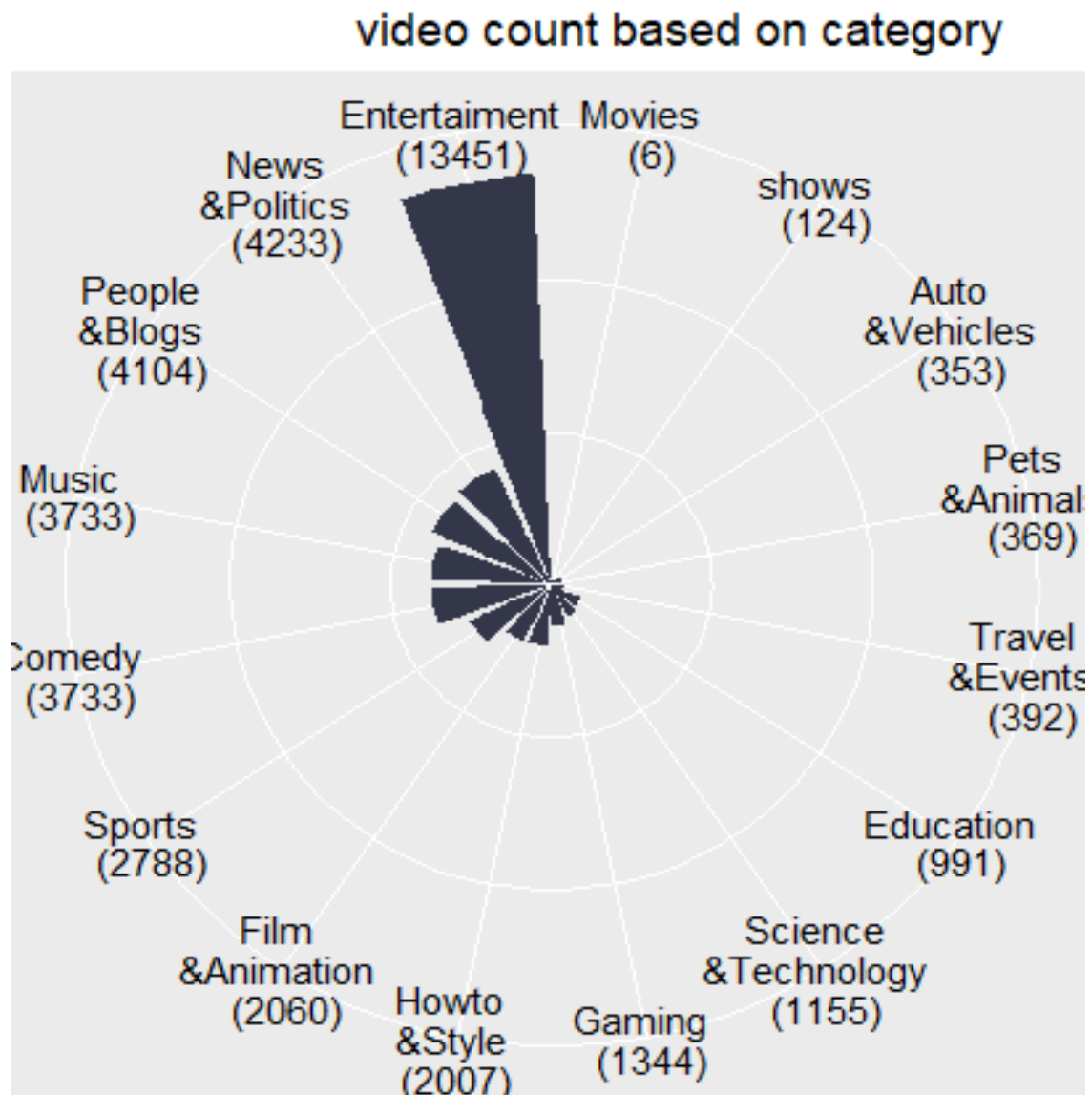
```
c <- plot_ly(
```

```
  type = "indicator",
```

```
  mode = "gauge+number+delta",
```

```
value = b,  
title = list(text = "compare average comment of Music to average", font = list(size = 24)),  
delta = list(reference = b1, increasing = list(color = "RebeccaPurple")),  
gauge = list(  
  axis = list(range = list(NULL, 16000), tickwidth = 1, tickcolor = "darkblue"),  
  bar = list(color = "darkblue"),  
  bgcolor = "white",  
  borderwidth = 2,  
  bordercolor = "gray",  
  steps = list(  
    list(range = c(0, 8000), color = "cyan"),  
    list(range = c(250, 16000), color = "royalblue")),  
  threshold = list(  
    line = list(color = "red", width = 4),  
    thickness = 0.75,  
    value = 14050)))  
c <- c %>%  
  layout(  
    paper_bgcolor = "lavender",  
    font = list(color = "darkblue", family = "Arial"))
```

- Bar Chart of video count based on category



```
library(sqldf)
library(ggplot2)
a2<-sqldf("select category_id from a1")
a3=data.frame(table(a2))
a3
a3[,2][1]
a3[,2][2]
a3[,1][1]
a3[,2][14]
a3[,1]

g=v/sum(v)*100
p1=c(formatC(g,digits = 1,format = "f"))
p2=paste0(p1,"%")
```


p1

p2

v=sort(v)

v=c(a3[,2][1],a3[,2][2],a3[,2][3],a3[,2][4],a3[,2][5],a3[,2][6],a3[,2][7],a3[,2][8],a3[,2][9],a3[,2][10],a3[,2][11],a3[,2][12],a3[,2][13],a3[,2][14],a3[,2][15],a3[,2][16])

f=c("Movies \n (6)","shows \n (124)","Auto \n&Vehicles \n (353)","Pets \n&Animals \n (369)","Travel \n&Events \n (392)","Education \n (991)","Science \n&Technology \n (1155)","Gaming \n (1344)","Howto \n&Style \n(2007)","Film \n&Animation \n (2060)","Sports \n (2788)","Comedy \n (3733)","Music \n (3733)","People \n&Blogs \n (4104)","News \n&Politics \n (4233)","Entertainment \n (13451)")

f

gh=data.frame(v,f)

ggplot(gh, aes(x=reorder(f,v), y=v)) +

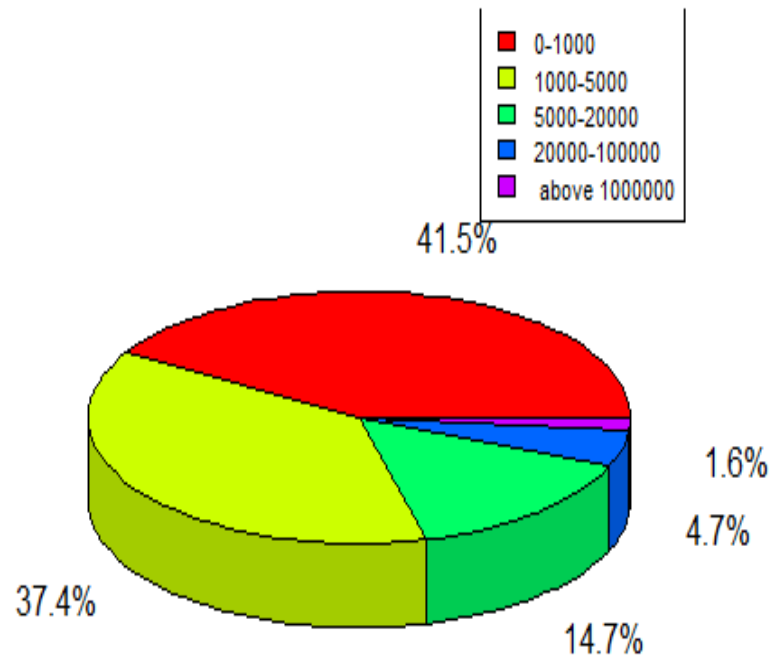
geom_bar(stat="identity",fill="#34374a",width = 0.8) +

theme_grey()+ggtitle(" video count based on category")+

theme(axis.title.y = element_blank(), axis.title.x = element_blank(),axis.text = element_blank(), axis.ticks = element_blank()+theme(axis.text.x = element_text(angle=0))+coord_polar()

• PieChart Of Comment Counts

Pie Chart of total comments



```
comment1k=nrow(table(subset(a1$comment_count,a1$comment_count<1000)))
comment5k=nrow(table(subset(a1$comment_count,a1$comment_count>1000
a1$comment_count<5000)))
comment20k=nrow(table(subset(a1$comment_count,a1$comment_count>5000
a1$comment_count<20000)))
comment50k=nrow(table(subset(a1$comment_count,a1$comment_count>20000
a1$comment_count<100000)))
comment100k=nrow(table(subset(a1$comment_count,a1$comment_count>100000
a1$comment_count<500000)))
comment500k=nrow(table(subset(a1$comment_count,a1$comment_count>500000 )))
```

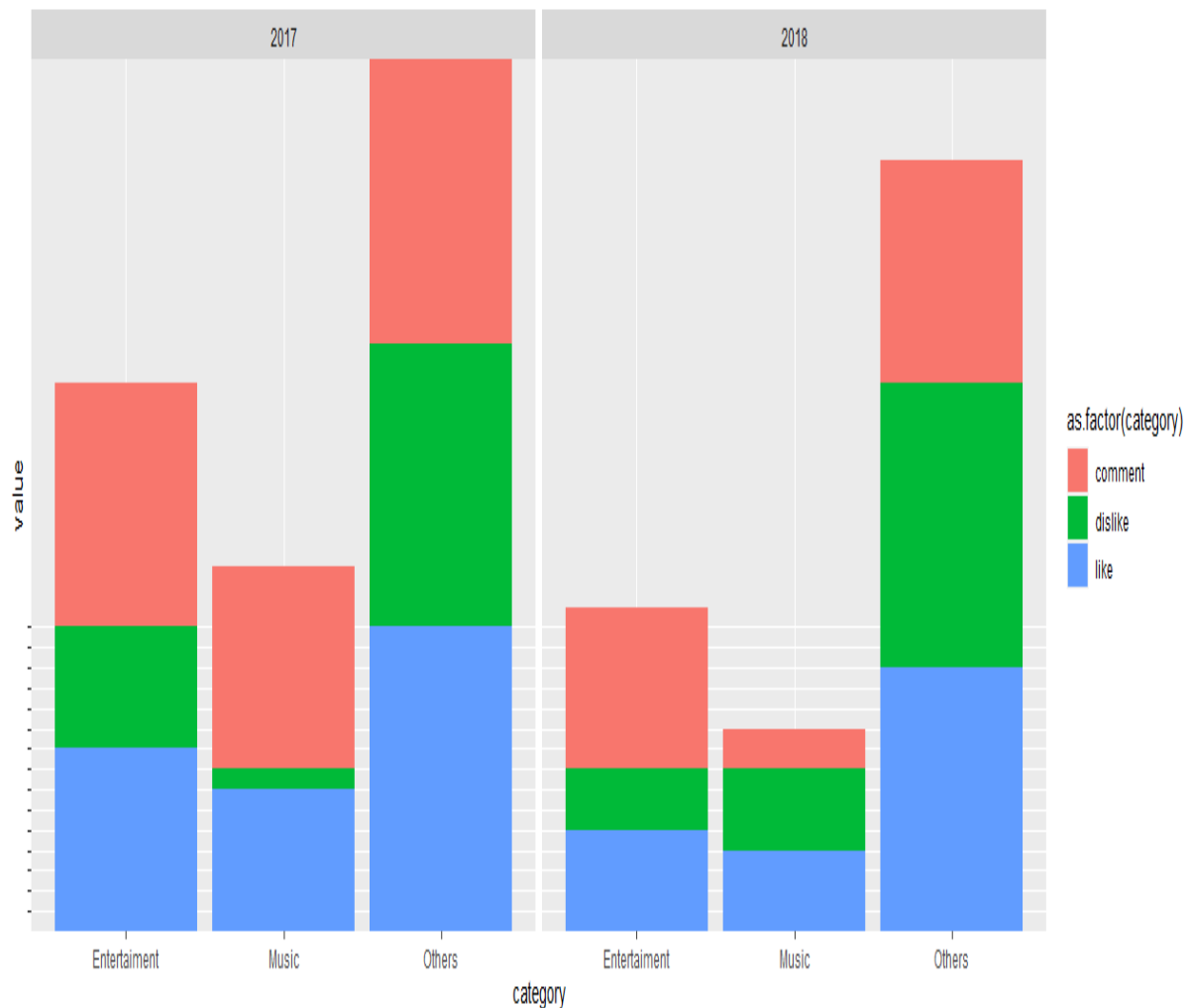
```
comment1k1=nrow(table(subset(a1$comment_count,a1$comment_count<1000)))
```

```
comment1k1
print(comment1k)
print(comment5k)
print(comment20k)
print(comment50k)
print(comment100k)
print(comment500k)

p=c(comment1k,comment5k,comment20k,comment50k,comment100k,comment500k)
p1=(p/sum(p))*100
p1
p2=formatC(p1,digits = 1,format = "f")
t=pie(p,labels = paste0(p2,"%"),explode = 0,radius = .9,labelcex = 1, main = "Pie Chart of
total comments")
t
g=c("0-1000","1000-5000","5000-20000","20000-100000"," above 1000000")
legend("topright",g,cex=0.8, fill=rainbow(length(t)))
```

• Bar Graph of comparing 2017 and 2018 like, dislike and comments

comaprison of comments,likes,dislikes between 2017 and 2018



```
library(sqldf)
library(ggplot2)
```

```
s=sqldf("select sum(comment_count),category_id from a1 group by category_id")
```

```
s1=s[,1][4]
```

```
s11=s[,1][4]/sum(s[,1])
```

```
s11=s11*100
```

```
s11=formatC(s11,digits = 0,format = "f")
```

```
s11
```

```
s2=s[,1][9]/sum(s[,1])
```

```
s22=s2*100
```

```
s22=formatC(s22,digits = 0,format = "f")
```

```
s22
```

```
s3=sum(s[,1])-s1-s2
```

```
s33=s3/sum(s[,1])
```

```
s33=s33*100
```

```
s33=formatC(s33,digits = 0,format = "f")
s33
```

```
m=sqldf("select sum(likes),category_id from a1 group by category_id")
m1=m[,1][4]
m11=m[,1][4]/sum(m[,1])
m11=m11*100
m11=formatC(m11,digits = 0,format = "f")
m11
```

```
m2=m[,1][9]
m22=m2/sum(m[,1])
m2
m22=s2*100
m22=formatC(m22,digits = 0,format = "f")
m22
m33=100-24-25
m33=formatC(m33,digits = 0,format = "f")
m33
```

```
n=sqldf("select sum(dislikes),category_id from a1 group by category_id")
n1=n[,1][4]
n11=n[,1][4]/sum(n[,1])
n11=n11*100
n11=formatC(n11,digits = 0,format = "f")
n11
```

```
n2=n[,1][9]
n22=n2/sum(n[,1])
n2
n22=s2*100
n22=formatC(n22,digits = 0,format = "f")
n22
```

```
n3=sum(n[,1])-n1-n2
n33=n3/sum(n[,1])
n33=n33*100
n33=formatC(n33,digits = 0,format = "f")
n33
```

```
o=sqldf("select sum(comment_count),category_id from a1 where trending_date<='17.31.12'
group by category_id")
o1=o[,1][4]
o11=o[,1][4]/sum(o[,1])
o11=o11*100
o11=formatC(o11,digits = 0,format = "f")
o11
```

```
o2=o[,1][9]/sum(o[,1])
o22=o2*100
```

```
o22=formatC(o22,digits = 0,format = "f")
o22
```

```
o33=100-34-3
o33=formatC(o33,digits = 0,format = "f")
o33
```

```
p=sqldf("select sum(likes),category_id from a1 where trending_date<='17.31.12' group by
category_id")
p1=p[,1][4]
p11=p[,1][4]/sum(p[,1])
p11=p11*100
p11=formatC(p11,digits = 0,format = "f")
p11
```

```
p
p2=p[,1][9]
p22=p2/sum(p[,1])
p22
p22=p22*100
p22=formatC(p22,digits = 0,format = "f")
p22
```

```
p33=100-24-25
p33=formatC(p33,digits = 0,format = "f")
p33
```

```
q=sqldf("select sum(dislikes),category_id from a1 where trending_date<='17.31.12' group by
category_id")
q1=q[,1][4]
q11=q[,1][4]/sum(q[,1])
q11=q11*100
q11=formatC(q11,digits = 0,format = "f")
q11
```

```
q2=q[,1][9]
q22=q2/sum(q[,1])
q22
q22=q22*100
q22=formatC(q22,digits = 0,format = "f")
q22
```

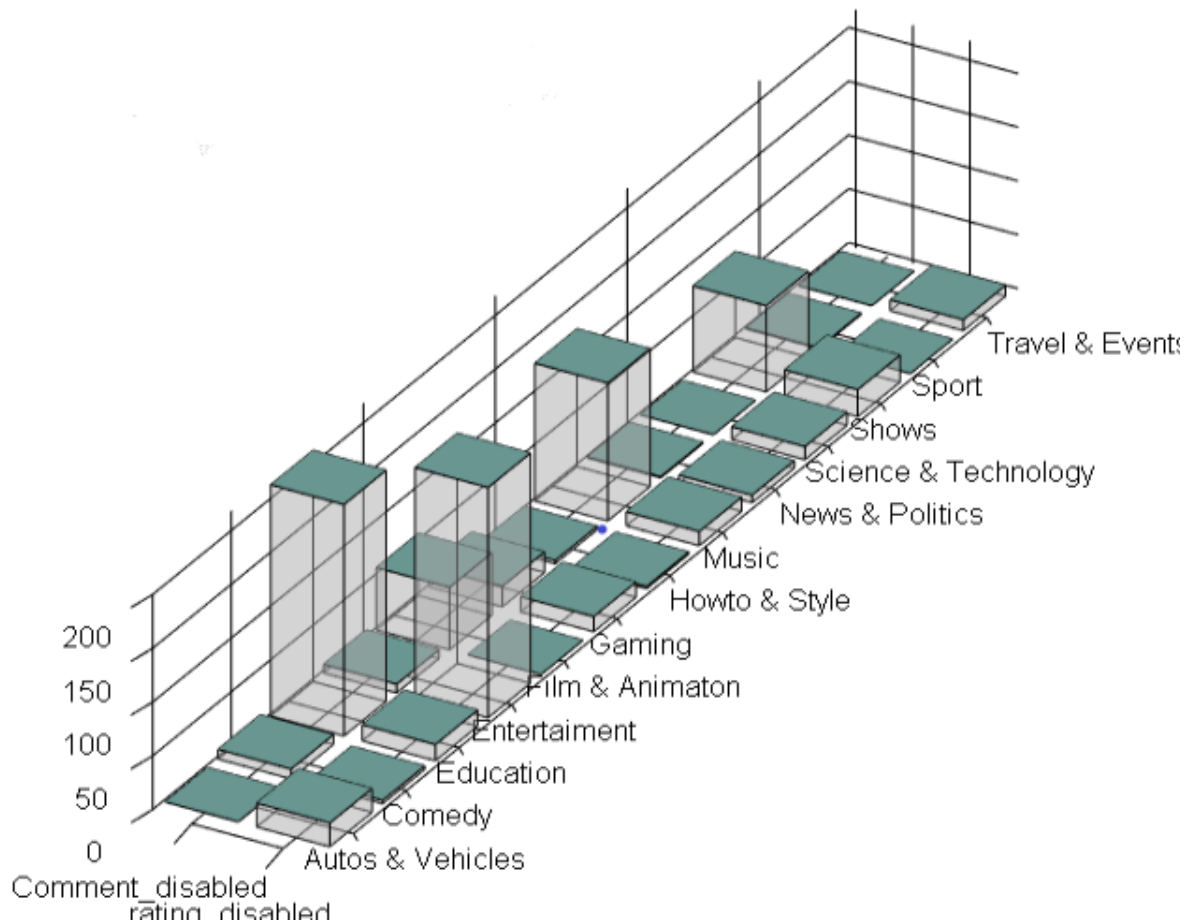
```
q33=100-45-4
q33=formatC(q33,digits = 0,format = "f")
q33
```

```
df <- data.frame("category" = c("like", "dislike", "comment"),
```

YOUTUBE TRENDING VIDEOS

```
"reason"      =      c("Entertainment",      "Entertainment",      "Entertainment",  
"Music","Music","Music",      "Others","Others",      "Others","Entertainment",      "Entertainment",  
"Entertainment", "Music","Music","Music", "Others","Others", "Others"),  
  "time" = c("2017", "2017", "2017","2017","2017","2017","2017","2017","2017",  
"2018","2018","2018","2018","2018","2018", "2018", "2018","2018"),  
  "value"                                           =  
c(o11,p11,q11,o22,p22,q22,o33,p33,q33,s11,m11,n11,s22,m22,n2,s33,m33,n33))  
value  
ggplot(data = df, aes(x = as.factor(reason), y = value)) +  
  geom_bar(aes(fill = as.factor(category)), stat = "identity") +  
  facet_grid(~time)+theme(axis.text.y                                           =  
element_blank())+xlab("category")+ggtitle("comaprison of comments,likes,dislikes between  
2017 and 2018")
```

• Bar Graph of Comment & Rating disabled:



```
d1=sqldf("select      count(comments_disabled),category_id      from      a1      where
comments_disabled==TRUE group by category_id")
```

```
d1
```

```
d2=sqldf("select      count(ratings_disabled),category_id      from      a1      where
ratings_disabled==TRUE group by category_id")
```

```
d2
```

```
d3=sqldf("select      count(video_error_or_removed),category_id      from      a1      where
video_error_or_removed==TRUE group by category_id")
```

```
d3
```

```
x<-c("      Autos & Vehicles","      Comedy","      Education","      Entertainment","
Film & Animation","      Gaming","      Howto & Style","      Music","
News & Politics", "      Science & Technology","Shows","Sport","
Travel & Events")
```

```
b<-c(0,22,9,3,215,15,8,214,58,0,19,14,3,2,128,13,1,7,0,12,80,25,1,0,0,10)
```

```
barplot3d(rows=13,cols=2,z=b,scalexy=80,alpha=0.3,theta=30,phi=30,
```

```
      topcolors=c("#699691"),xlabels
```

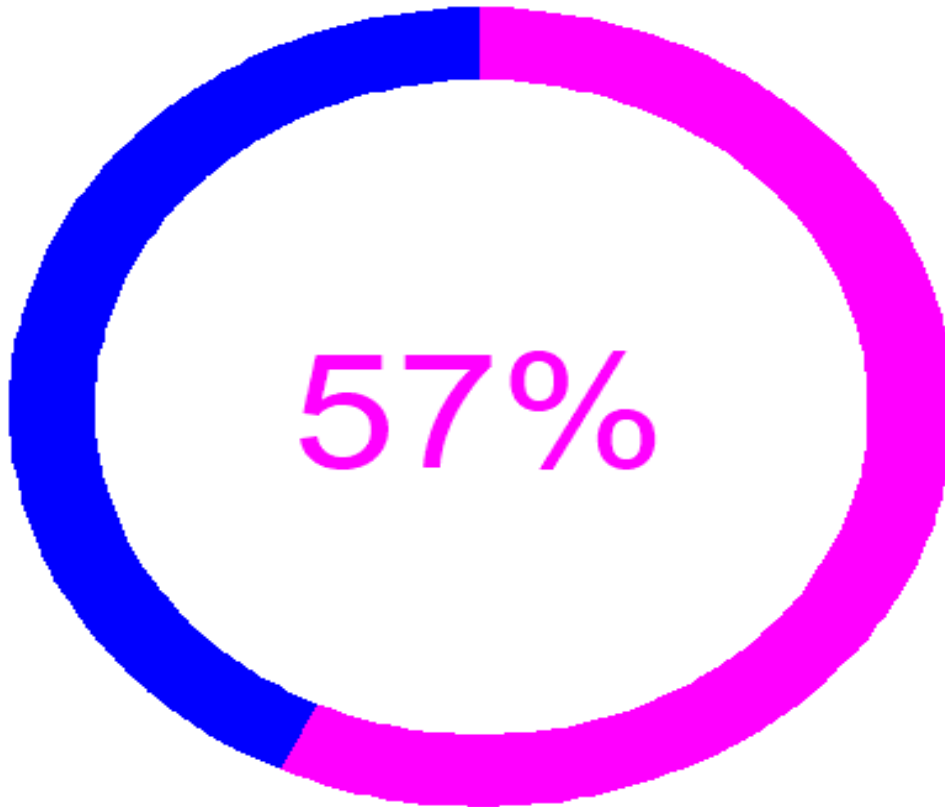
```
=
```

```
c("Comment_disabled","rating_disabled"),ylabels=x,
```

```
)
```


- Donut Graph of share of views of Entertainment & Music from all views

Share of views of entertainment & Music
from total views



```
install.packages("ggiraph")
library(dplyr)
library(ggplot2)
library(ggiraph)
library(sqldf)
```

```
a=sqldf("select sum(views),category_id from a1 group by category_id")
c=sum(a[,1])
b=a[,1][4]+a[,1][9]
d=b/c
d
```

```
donut_data <- data.frame(type = c("Blue", "Pink"), value = c(43, 57)) %>%
  mutate(
    percentage = value / sum(value),
    hover_text = paste0(type, ": ", value)
```

```

) %>%
mutate(percentage_label = paste0(round(100 * percentage, 1), "% "))

donut_plot <- ggplot(donut_data, aes(y = value, fill = type)) +
  geom_bar_interactive(
    aes(x = 1, tooltip = hover_text),
    width = 0.2,
    stat = "identity",
    show.legend = FALSE
  ) +
  annotate(
    geom = "text",
    x = 0,
    y = 0,
    label = donut_data[["percentage_label"]][donut_data[["type"]] == "Pink"],
    size = 20,
    color = "magenta"
  ) +

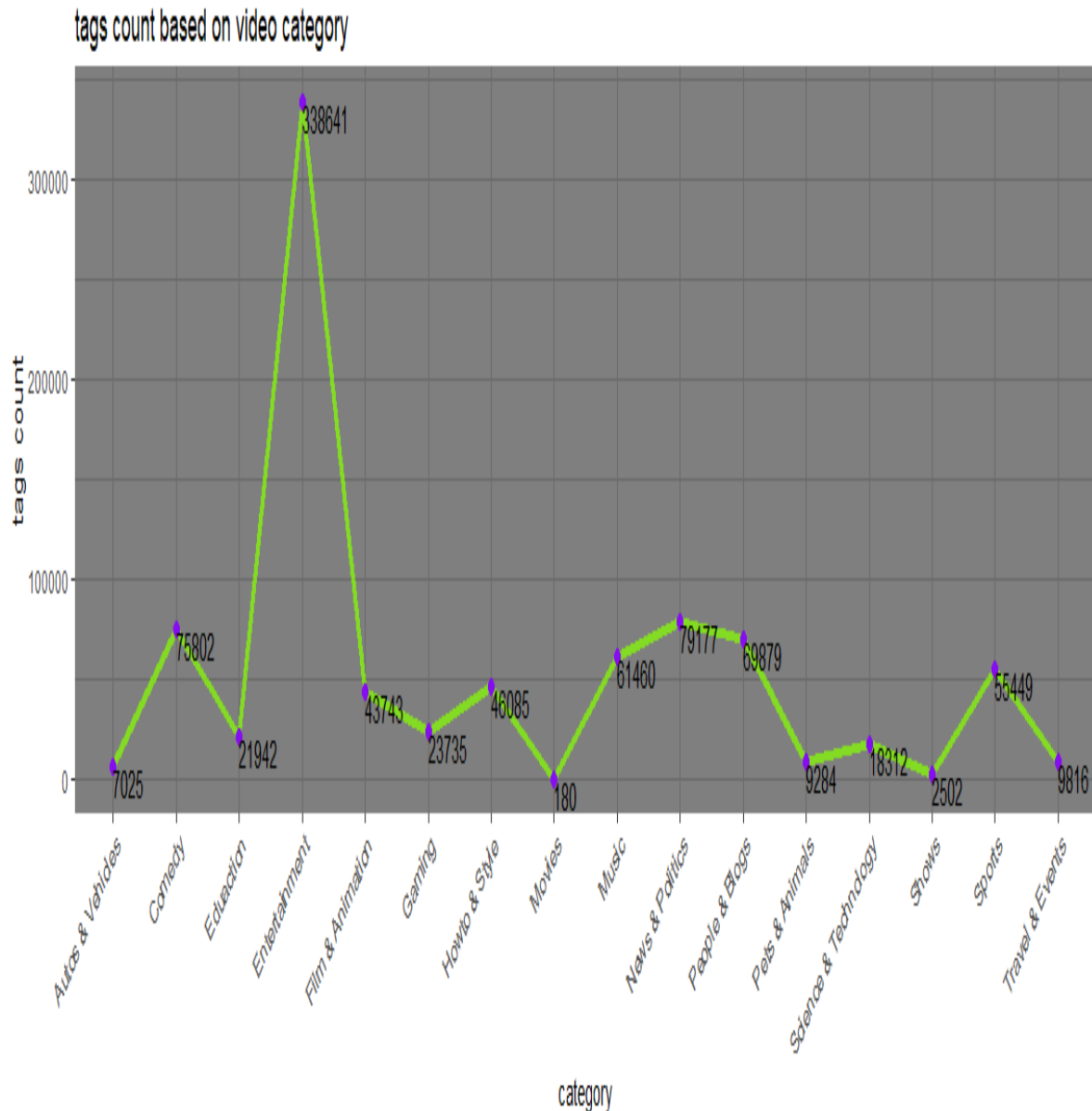
  scale_fill_manual(values = c(Blue = "blue", Pink = "magenta")) +
  coord_polar(theta = "y") +
  theme_void()
donut_plot+ggtitle("Share of views of entertainment & Music \n
total views")
ggtitle("hello")

ggiraph(ggobj=donut_plot)

```

YOUTUBE TRENDING VIDEOS

- Line Graph of tags count based on video category

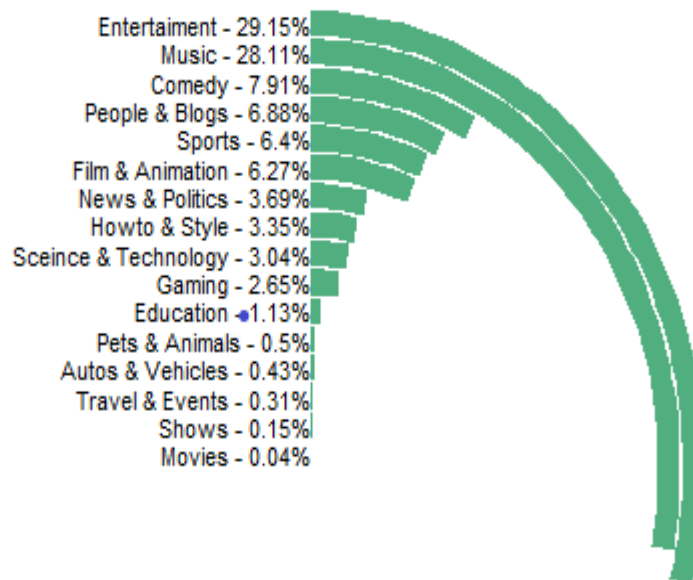


```
library(ggplot2)
library(sqldf)
library(stringr)
c=str_count(a1$tags,"[[:punct:]]")
d=c/2
aa1=formatC(d,digits = 0,format = "f")
da=data.frame(video_id=a1$video_id,category_id=a1$category_id,aa=aa1)
f=sqldf("select category_id,sum(aa) from da ASC group by category_id")
x=c("Autos & Vehicles","Comedy","Education","Entertainment","Film & Animation","Gaming","Howto & Style","Movies","Music","News & Politics","People & Blogs","Pets & Animals","Science & Technology","Shows","Sports","Travel & Events")
y=c(7025,75802,21942,338641,43743,23735,46085,180,61460,79177,69879,9284,18312,2502,55449,9816)
dat=data.frame(x,y)
```

```
options(scipen=999)
ggplot(dat,aes(x=x,y=y,group=1))+  geom_line(color="#84f20d",  size=1.5,  alpha=.8,
linetype=1)+theme_dark()  +theme(axis.text.x  =  element_text(angle  =  45,hjust  =
1))+geom_point(color="#850ef1",size=2)+geom_text(label=y,vjust=1,hjust=0)+xlab("category")+ylab("tags count")+ggtitle("tags count based on video category")
```

- Bar Graph of video views percentage based on category

Video views percentage



```
library(sqldf)
```

```
a3<-sqldf("select category_id ,sum(views) from a1 group by category_id")
```

```
Category <- c("Movies","Shows","Travel & Events","Autos & Vehicles","Pets & Animals","Education","Gaming","Sceince & Technology","Howto & Style","News & Politics",  
              "Film & Animation","Sports","People & Blogs","Comedy","Music","Entertainment")
```

```
#Percent  
c(a3[,2][1],a3[,2][2],a3[,2][3],a3[,2][4],a3[,2][5],a3[,2][6],a3[,2][7],a3[,2][8],a3[,2][9],a3[,2][10],a3[,2][11],a3[,2][12],a3[,2][13],a3[,2][14],a3[,2][15],a3[,2][16])
```

```
Percent<-
```

```
c(0.04,0.15,0.31,0.43,0.50,1.13,2.65,3.04,3.35,3.69,6.27,6.40,6.88,7.91,28.11,29.15)
```

```
Percent1=Percent/sum(Percent)*100
```

```
p2=formatC(Percent1,digits = 2,format = "f")
```

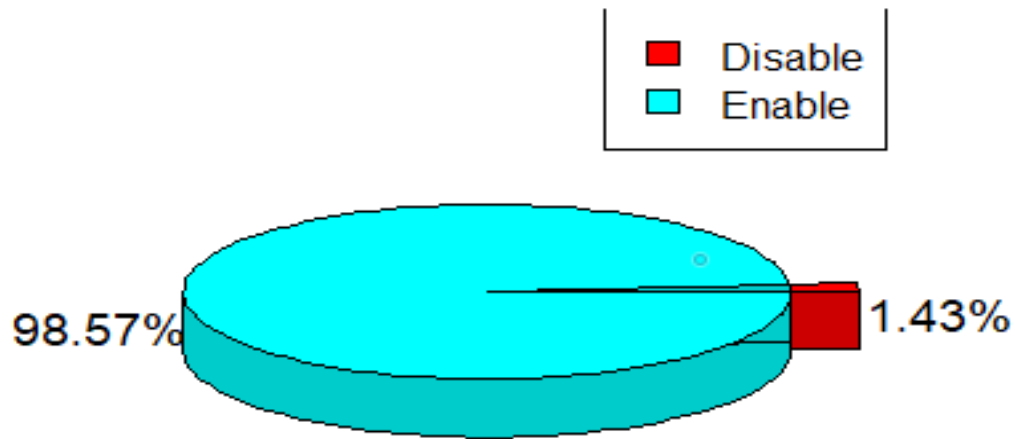
```
#install.packages("hrbrthemes")
library(hrbrthemes)

i1<-data.frame(Category,Percent)

ggplot(i1, aes(x =reorder(Category,Percent), y = Percent,
               fill = Category)) +
  geom_bar(width = 0.9, stat="identity",fill="#51ae7e") +
  coord_polar(theta = "y") +
  xlab("") + ylab("") +
  ylim(c(0,100)) +
  ggtitle("      Video views percentage") +
  geom_text(data =i1, hjust = 1, size = 3,
            aes(x = Category, y = 0, label = paste0(Category," - ", Percent,"%")))+
  theme_ipsum() +
  theme(legend.position = "none",
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        axis.line = element_blank(),
        axis.text.y = element_blank(),
        axis.text.x = element_blank(),
        axis.ticks = element_blank())
```

- Pie chart of Comment Enable/Disable

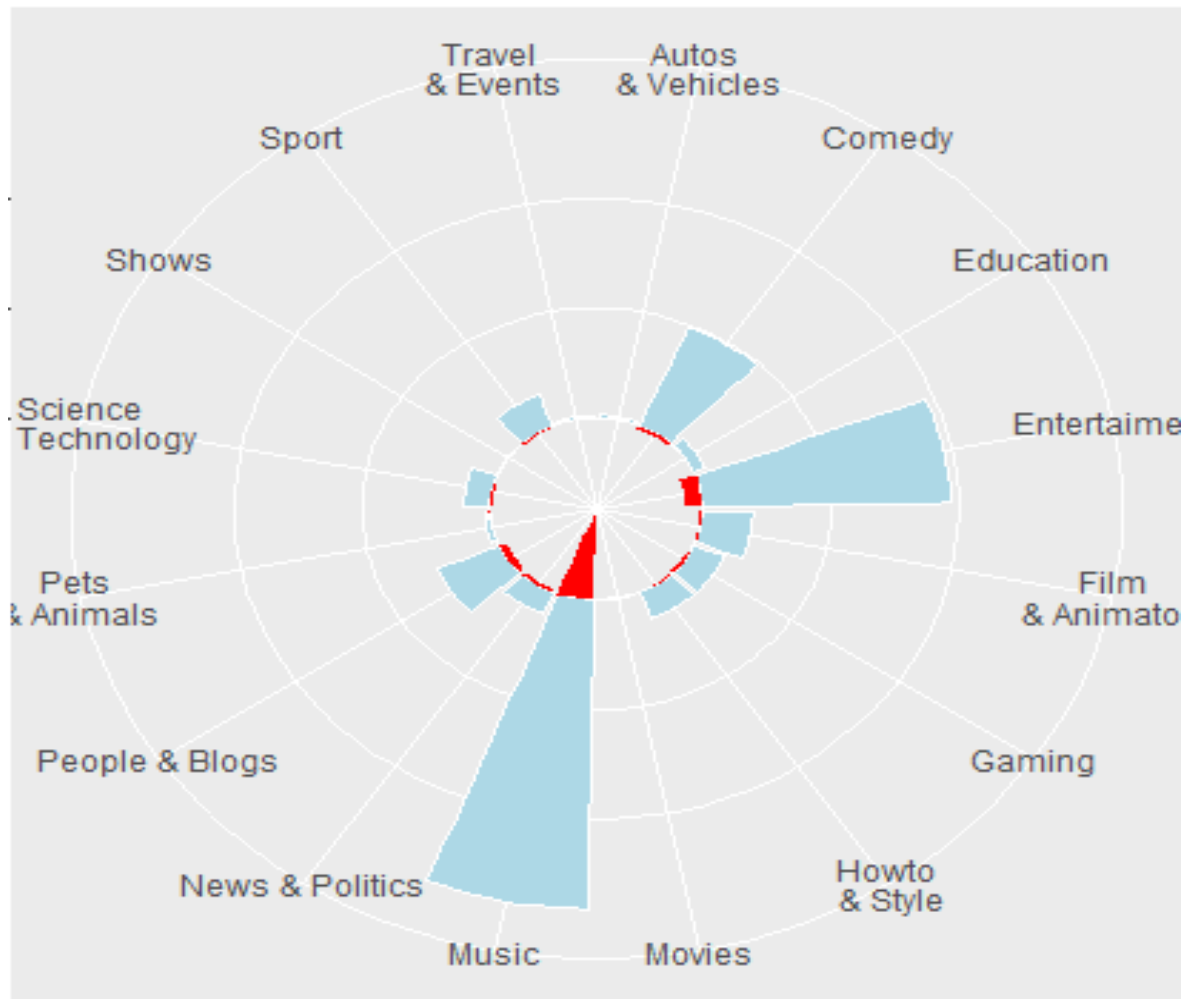
Pie Chart of comment Enable/Disbaled



```
install.packages("plotrix")
library(plotrix)
p=c(tr,fa)
p1=(p/sum(p))*100
p1
p2=formatC(p1,digits = 2,format = "f")
t=pie3D(p,labels = paste0(p2,"%"),explode = 0.1,radius = .9,labelcex = 1.2, main = "Pie
Chart of comment Enable/Disbaled ")
t
legend("topright",c("Disable","Enable"),cex=1, fill=rainbow(length(t)))
```

- Bar Chart of Like & Dislike based on Category

Like & dislike based on video category



```
library(sqldf)
library(ggplot2)
a3<-sqldf("select category_id ,sum(likes),sum(dislikes) from a1 group by category_id")
x<-c("Autos \n& Vehicles","Comedy","Education","Entertainment","Film\n&
Animaton","Gaming","Howto\n & Style","Movies","Music","News & Politics", "People &
Blogs","Pets \n& Animals","Science\n & Technology","Shows","Sport","Travel \n&
Events")
a<-
c(a3[,2][1],a3[,2][2],a3[,2][3],a3[,2][4],a3[,2][5],a3[,2][6],a3[,2][7],a3[,2][8],a3[,2][9],a3[,2][
10],a3[,2][11],a3[,2][12],a3[,2][13],a3[,2][14],a3[,2][15],a3[,2][16])
b<-
c(a3[,3][1],a3[,3][2],a3[,3][3],a3[,3][4],a3[,3][5],a3[,3][6],a3[,3][7],a3[,3][8],1666666666,a3[,
3][10],a3[,3][11],a3[,3][12],a3[,3][13],a3[,3][14],a3[,3][15],a3[,3][16])

y=append(a,b*-1)
data<-data.frame(s,y)
```

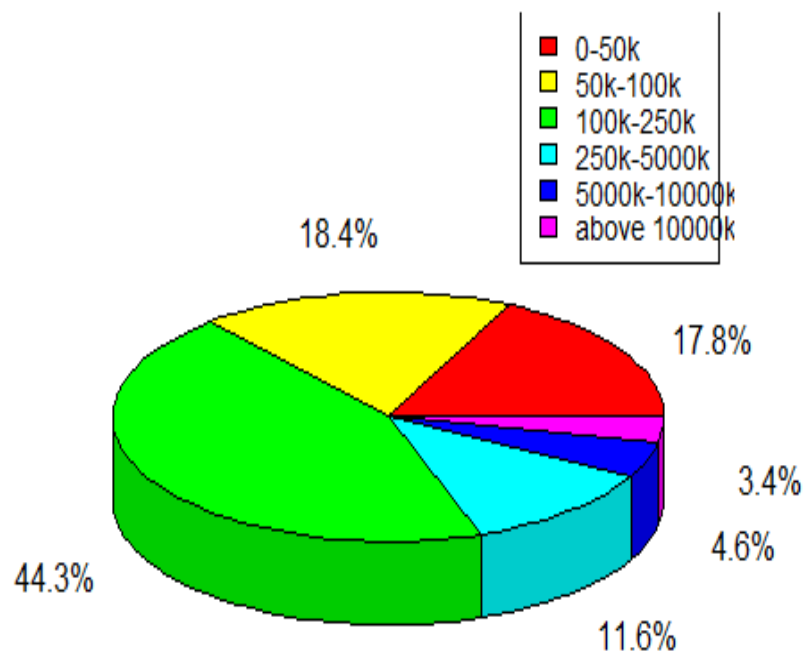

data

options(scipen=999)

```
ggplot(data, aes(x = x, y = y)) +
  geom_bar(
    aes(x,y),
    stat = "identity", position = position_stack(),
    color = "white", fill = "lightblue",
    subset(data,subset = y>0)
  ) +
  geom_bar(
    aes(x,y),
    stat = "identity",
    fill="red",
    subset(data,subset = y<0)
  )+coord_polar()+theme(axis.title.x = element_blank(),axis.text.y
=element_blank(),axis.title.y = element_blank() )+
  ggtitle("Like & dislike based on video category")
```

- **Pie Chat of total Views**

Pie Chart of total views



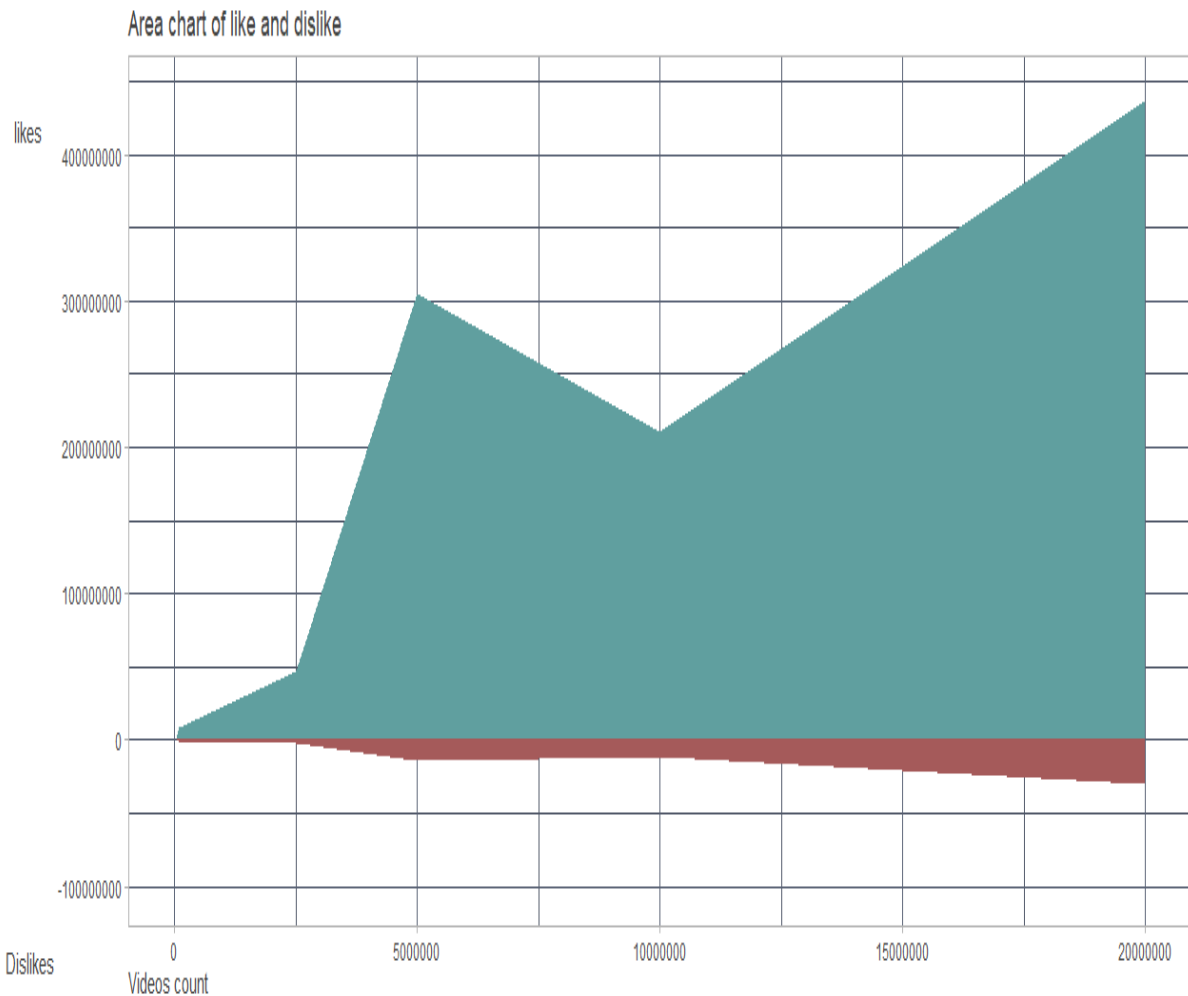
```
library("plotrix")
views50k=nrow(table(subset(a1$views, subset = a1$views<50000)))
views100k=nrow(table(subset(a1$views, subset= a1$views>50000 &
a1$views<100000)))
views250k=nrow(table(subset(a1$views,subset = a1$views >100000 &
a1$views<250000)))
views500k=nrow(table(subset(a1$views,subset = a1$views >2500000 &
a1$views<5000000 )))
views1000k=nrow(table(subset(a1$views,subset= a1$views>5000000 &
a1$views<10000000)))
```

YOUTUBE TRENDING VIDEOS

```
views2000k=nrow(table(subset(a1$views,subset =a1$views>10000000)))

p=c(views50k,views100k,views250k,views500k,views1000k,views2000k)
p1=(p/sum(p))*100
p1
p2=formatC(p1,digits = 1,format = "f")
g=c("0-50k","50k-100k","100k-250k","250k-5000k","5000k-10000k","above
10000k")
t=pie3D(p,labels = paste0(p2,"%"),explode =0 ,radius = .9,labelcex = 1, main = "Pie
Chart of total views")
t
legend("topright",g,cex=0.9, fill=rainbow(length(t)))
```

- Area Graph like, dislike comparison



```
library(ggplot2)
likes50k=sum(subset(a1$likes, subset=(a1$views<50000)))
likes100k=sum(subset(a1$likes, subset=(a1$views>50000 & a1$views<100000)))
likes250k=sum(subset(a1$likes, subset=(a1$views>100000 & a1$views<250000)))
likes500k=sum(subset(a1$likes, subset=(a1$views>250000 & a1$views<500000 )))
likes1000k=sum(subset(a1$likes, subset=(a1$views>500000 & a1$views<1000000)))
likes2000k=sum(subset(a1$likes, subset=(a1$views>1000000)))

dislikes50k=sum(subset(a1$dislikes, subset=(a1$views<50000)))
dislikes100k=sum(subset(a1$dislikes, subset=(a1$views>50000 & a1$views<100000)))
dislikes250k=sum(subset(a1$dislikes, subset=(a1$views>100000 & a1$views<250000)))
dislikes500k=sum(subset(a1$dislikes, subset=(a1$views>250000 & a1$views<500000 )))
dislikes1000k=sum(subset(a1$dislikes, subset=(a1$views>500000 & a1$views<1000000)))
dislikes2000k=sum(subset(a1$dislikes, subset=(a1$views>1000000)))

sum(dislikes50k)
like=c()
```

YOUTUBE TRENDING VIDEOS

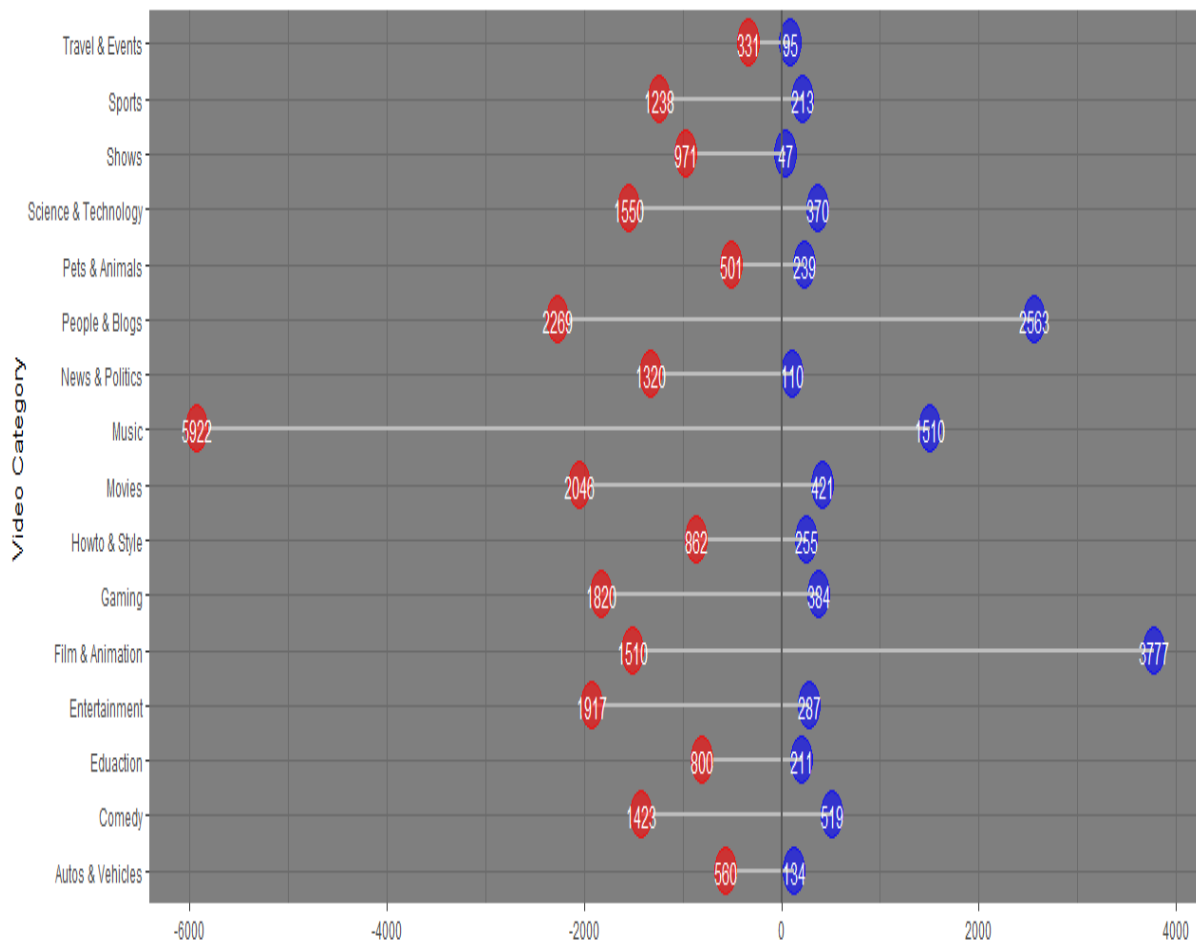
```
like=c(likes50k,likes100k,likes250k,likes500k,likes1000k,likes2000k)
dislike=c(dislikes50k,dislikes100k,dislikes250k,dislikes500k,dislikes1000k,dislikes2000k)
y=append(like,dislike*-1)
y
options(scipen=999)
x=c(50000,100000,2500000,5000000,10000000,20000000)
length(y)
length(x)
li=data.frame(x,y)

ggplot(li,aes(x=x,y=y))+ylim(-100000000,440000000)+geom_area(fill="#a55a5a",aes(x=x,y=y),subset(li,subset=y<0))+geom_area(fill="#609f9f",aes(x,y),subset(li,subset=y>0)))+theme_light()+theme(legend.position = c(0.95,0.95),legend.justification = c("vbfh","fdn"))+labs(x="Videos count",y="\n\nlikes\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n Dislikes",title = "Area chart of like and dislike")+theme(text = element_text(family = 'Gill Sans',colour = "#444444"),panel.grid.minor = element_line(color='#4d5566'),panel.grid.major = element_line(color = '#586174'),axis.title.y = element_text(vjust = 1,angle = 0),axis.title.x = element_text(hjust = 0))
```

YOUTUBE TRENDING VIDEOS

- Lollipop Graph of average like and dislike based on catgory**

Average likes&dislikes on video
(likes in 100)



```
library(ggplot2)
library(sqldf)
```

```
f=sqldf("select category_id,sum(likes),sum(dislikes) from a1 group by category_id")
```

```
d=sqldf("select count(video_id),category_id from a1 group by category_id")
```

```
d
```

```
f
```

```
x=c("Autos & Vehicles","Comedy","Eduaction","Entertainment","Film & Animation",
"Gaming","Howto & Style","Movies","Music","News & Politics","People & Blogs",
"Pets & Animals","Science & Technology","Shows","Sports","Travel & Events")
#a=c(13405,51960,21162,28789,377767,38487,25502,42106,151042,10145,256333,23931,37068,4752,21390,9568)
```

```
b=c(-560,-1423,-800,-1917,-1510,-1820,-862,-2046,-5922,-1320,-2269,-501,-1550,-971,-1238,-331)
```

```
z=c(13,51,21,28,377,38,25,42,151,10,256,23,37,4,21,9,-560,-1423,-800,-1917,-1510,-1820,-862,-2046,-5922,-1320,-2269,-501,-1550,-971,-1238,-331)
```

```
a=c(134,519,211,287,3777,384,255,421,1510,110,2563,239,370,47,213,95)
```

```
a=a/100
```

```

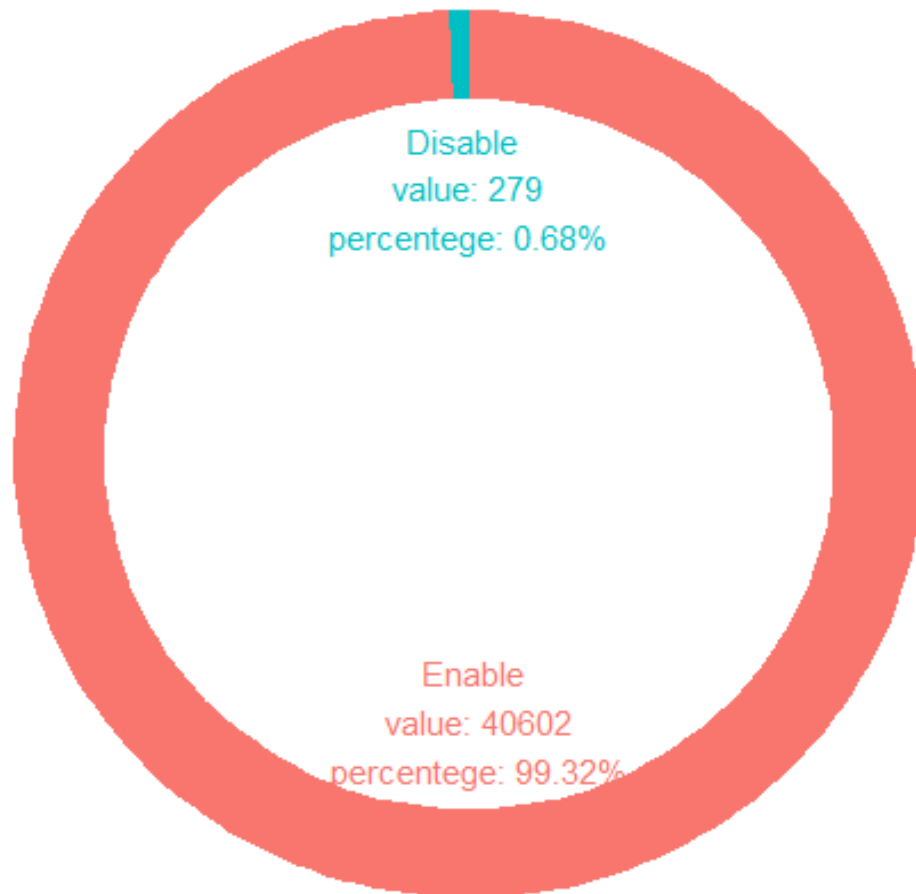
length(z1)
length(z2)
a=formatC(a,digits = 0,format = "f")
z1
z=append(a,b)
z
a=a
da=data.frame(x,z,a,b)
length(z)
z

a
b
z
ggplot(da,aes(x,z))+
  geom_segment(aes(x=x, y = 0, xend = x, yend =z ), color = "grey50")+
  geom_point(color="blue",size=8,alpha=0.6,aes(x,a), subset(da,subset = z>0))+
  geom_segment(aes(x=x,xend=x,y=1,yend=b),color="grey",size=1)+
  geom_segment(aes(x=x,xend=x,y=0,yend=a),color="grey",size=1)+
  geom_point(color="red",size=8,alpha=0.6, aes(x,z), subset(da,subset = z<0))+
  geom_hline(aes(yintercept = 0), color = "grey35", size = 1) +
  geom_text(label=a,aes(x,a),color="white", subset(da,subset = z<0))+
  geom_text(label=b*-1,aes(x,b),color="white", subset(da,subset = z>0))+
  theme(panel.grid.major = element_blank(),
        panel.border = element_blank(),
        axis.ticks.y = element_blank(),
        axis.text.x = element_text(angle = 45,hjust=1))+ylab("")+
  coord_flip()+
  theme_dark()+
  xlab("Video Category")+
  ggtitle("Average likes&dislikes on video \n (likes in 100) ")

```

- Donut Graph of Rating Enabled/Disables

Dount chart of Rating Enable/Disable



```
data<-file("C:/krunal/CAvideos.csv")
da=data.frame(data)
a2<-sqldf("select ratings_disabled from a1")
a3=data.frame(table(a2))
a3[,2][1]
a3[,2][2]
```

```
library(ggplot2)
library(plotrix)
p=c(a3[,2][1],a3[,2][2])
p1=(p/sum(p))*100
p1
p2=formatC(p1,digits = 2,format = "f")
```



```

data <- data.frame(
  category=c("Enable","Disable"),
  count=p
)

data$fraction <- data$count / sum(data$count)

data$ymax <- cumsum(data$fraction)

data$ymin <- c(0, head(data$ymax, n=-1))

data$labelPosition <- (data$ymax + data$ymin) / 2

data$label <- paste0(data$category, "\n value: ", data$count, "\n percentege: ",p2,"%")

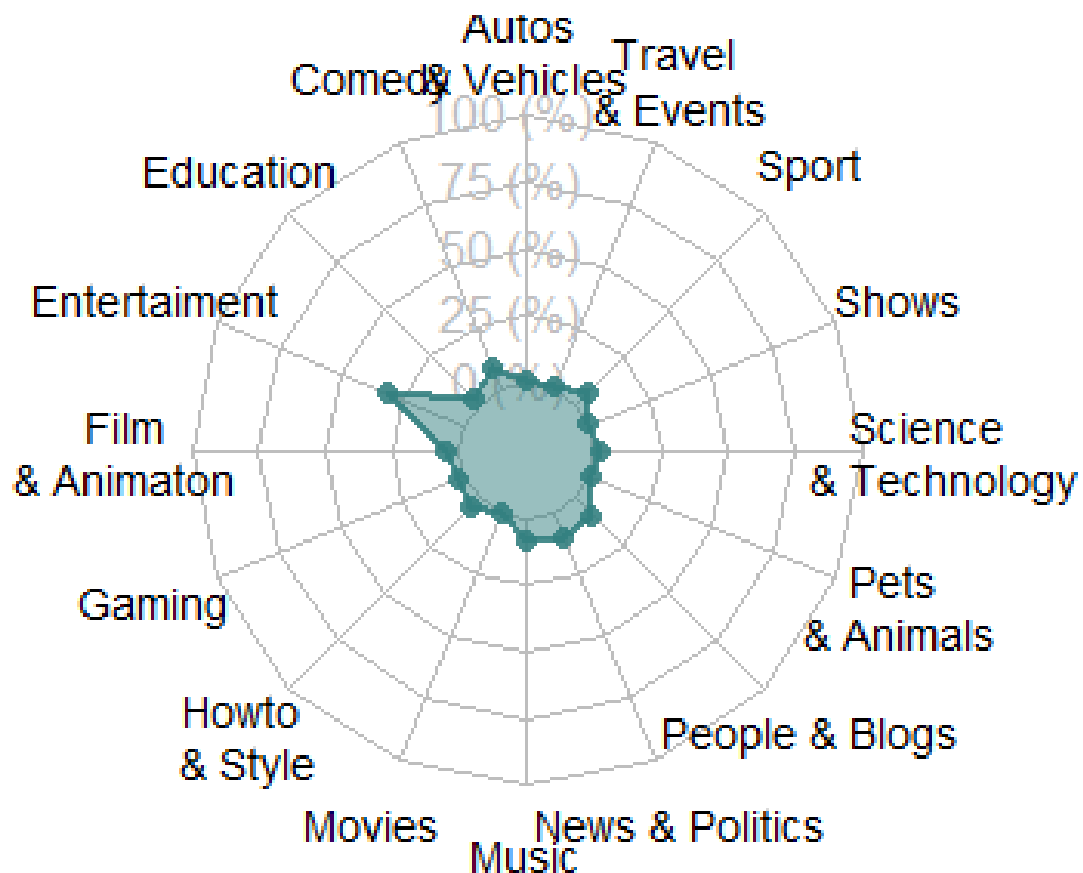
ggplot(data, aes(ymax=ymax, ymin=ymin, xmax=4, xmin=3, fill=c("blue","red"))) +
  geom_rect() +
  geom_text( x=2, aes(y=labelPosition, label=label, color=c("blue","red")), size=4) +
  coord_polar(theta="y") +
  xlim(c(-1, 4)) +
  theme_void() +
  theme(legend.position = "none")+labs(x="",y="",title = "
Enable/Disable")

```

Dount chart of Rating

- Radar Graph of video percentage based on category

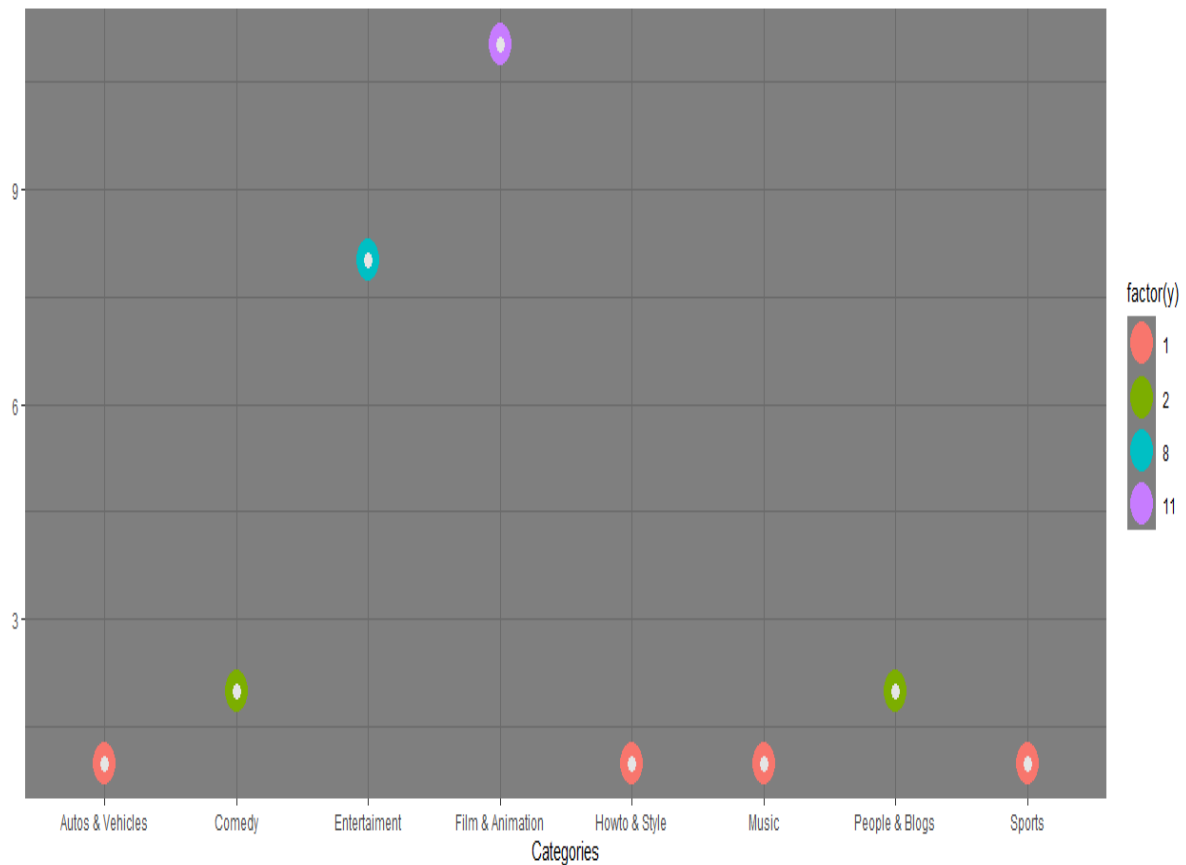
Radar chart on video percentage



```
library(fmsb)
data
as.data.frame(matrix(c(a3[,2][1],a3[,2][2],a3[,2][3],a3[,2][4],a3[,2][5],a3[,2][6],a3[,2][7],a3[,2][8],a3[,2][9],a3[,2][10],a3[,2][11],a3[,2][12],a3[,2][13],a3[,2][14],a3[,2][15],a3[,2][16]),
ncol=16))
colnames(data) <- c("Autos \n& Vehicles","Comedy","Education","Entertainment","Film\n& Animaton","Gaming","Howto\n & Style","Movies","Music","News & Politics", "People & Blogs","Pets \n& Animals","Science\n      & Technology","Shows","Sport","Travel \n& Events")
data <- rbind(rep(43000,43000) , rep(0,43000) , data)
data
radarchart( data , axistype=1 ,
            pcol=rgb(0.2,0.5,0.5,0.9) , pfc=rgb(0.2,0.5,0.5,0.5) , plwd=2 ,
            cglcol="grey", cglty=1, axislabcol="grey", cglwd=0.8,
            vlce=0.9
)
title("Radar chart on video percentage")
```

- **Dot Graph of video error or removed based on category**

Error chart based on category



```
library(ggplot2)
```

```
library(sqldf)
```

```
s1=sqldf("select count(video_error_or_removed),category_id from a1 where video_error_or_removed=TRUE group by category_id")
```

```
s1
```

```
x=c("Autos & Vehicles","Comedy","Entertainment","Film & Animation","Howto & Style","Music","People & Blogs","Sports")
```

```
y=c(1,2,8,11,1,1,2,1)
```

```
y
```

```
h=data.frame(x,y)
```

```
ggplot(h,aes(x=x,y=y))+
```

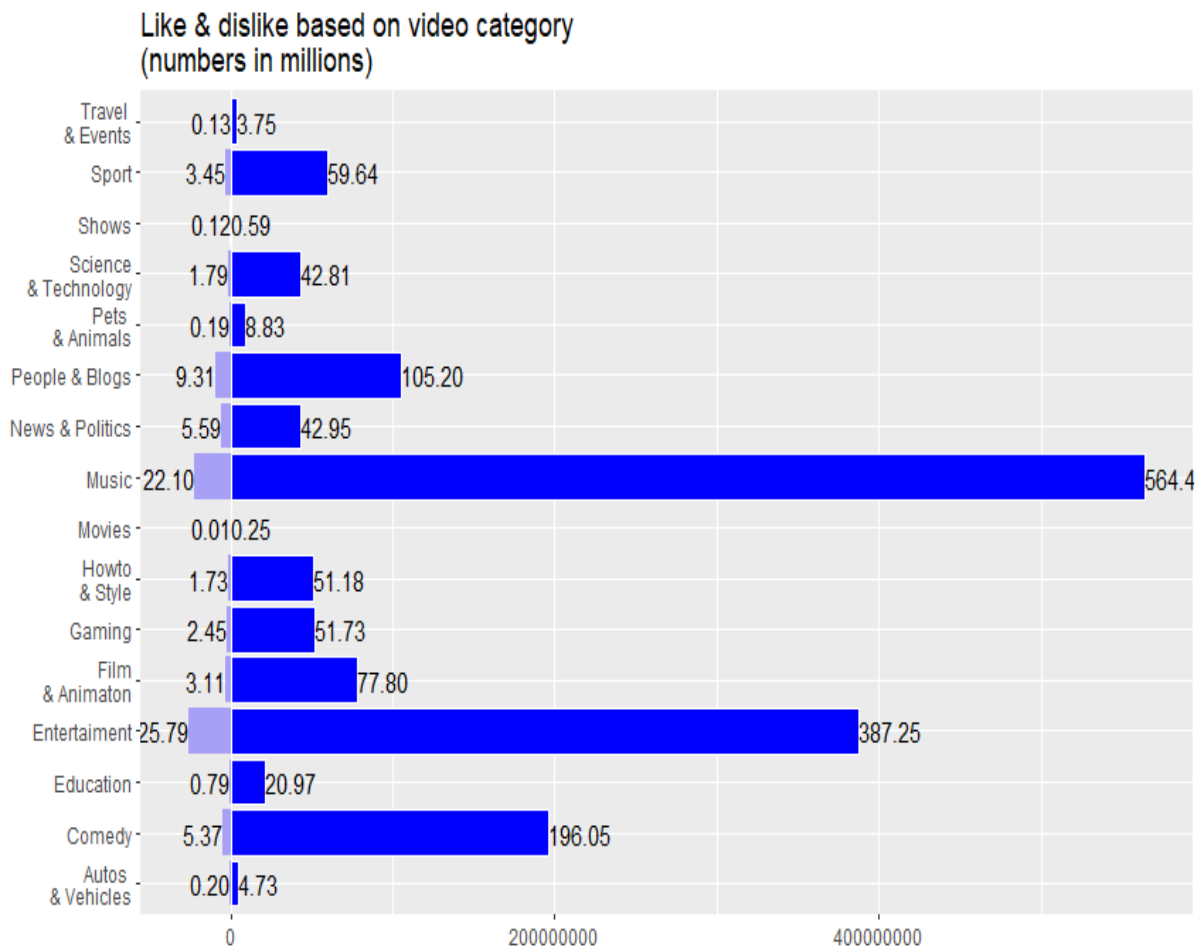
```
geom_point(aes(colour=factor(y)),size=8)+
```

```
geom_point(colour="grey90",size=3)+
```

```
theme_dark()+xlab("Categories")+ggtitle("Error chart based on category")+ylab("")
```

YOUTUBE TRENDING VIDEOS

- Bar Graph of like and dislike based on category



```
library(sqldf)
```

```
library(ggplot2)
```

```
a3<-sqldf("select category_id ,sum(likes),sum(dislikes) from a1 group by category_id")
```

```
x<-c("Autos \n& Vehicles","Comedy","Education","Entertainment","Film\n& Animation","Gaming","Howto\n & Style","Movies","Music","News & Politics", "People & Blogs","Pets \n& Animals","Science\n & Technology","Shows","Sport","Travel \n& Events")
```

```
a<-
```

```
c(a3[,2][1],a3[,2][2],a3[,2][3],a3[,2][4],a3[,2][5],a3[,2][6],a3[,2][7],a3[,2][8],a3[,2][9],a3[,2][10],a3[,2][11],a3[,2][12],a3[,2][13],a3[,2][14],a3[,2][15],a3[,2][16])
```

```
b<-
```

```
c(a3[,3][1],a3[,3][2],a3[,3][3],a3[,3][4],a3[,3][5],a3[,3][6],a3[,3][7],a3[,3][8],a3[,3][9],a3[,3][10],a3[,3][11],a3[,3][12],a3[,3][13],a3[,3][14],a3[,3][15],a3[,3][16])
```

```
aa=a/1000000
```

```
bb=b/1000000
```

```
aa1=formatC(aa,digits = 2,format = "f")
```

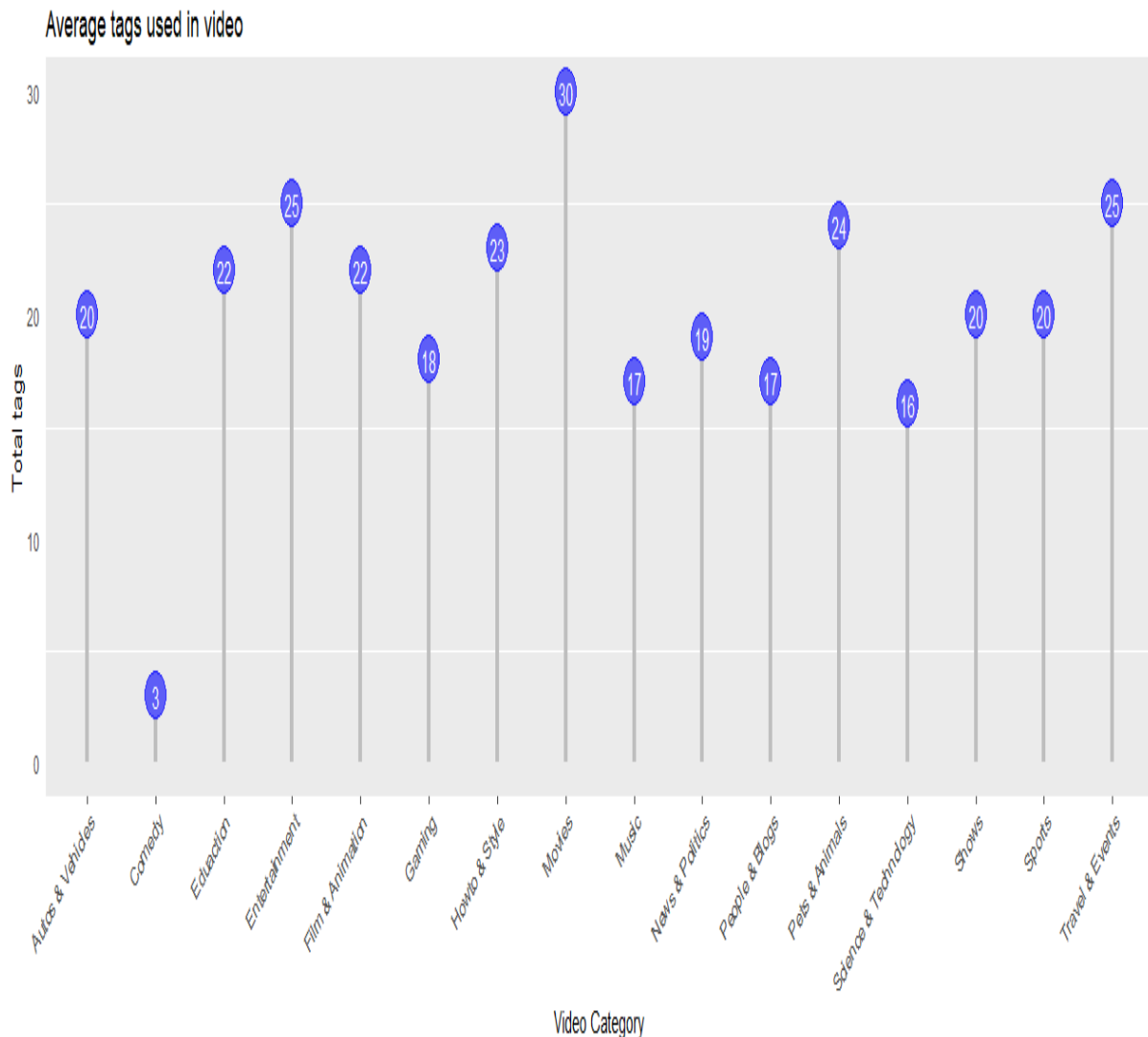
```
bb1=formatC(bb,digits = 2,format = "f")
```

```
y=append(a,b*-1)
data<-data.frame(s,y)
data
```

```
options(scipen=999)
```

```
ggplot(data, aes(x = x, y = y,fill=y)) +
  geom_bar(
    aes(x,y),
    stat = "identity", position = position_stack(),
    color = "white", fill = "blue",
    subset(data,subset = y>0)
  ) +
  geom_bar(
    aes(x,y,fill=y),
    stat = "identity",
    fill="#a7a0f6",
    subset(data,subset = y<0)
  )+geom_text(aes(x,y),label=aa1, subset(data,subset = y>0),hjust=0)+
  geom_text(aes(x,y),label=bb1,subset(data,subset = y<0),hjust=1)+
  coord_flip()+
  ggtitle("Like & dislike based on video category \n(numbers in millions)") + theme(axis.title.x
= element_blank(),axis.title.y = element_blank())
```

• Lollipop Graph of Average tags used by videos based on category



```
library(ggplot2)
library(sqldf)
library(stringr)
```

```
c=str_count(a1$tags,"[[:punct:]]")
d=c/2
aa1=formatC(d,digits = 0,format = "f")
aa1
da=data.frame(video_id=a1$video_id,category_id=a1$category_id,aa=aa1)
da
f=sqldf("select category_id,sum(aa) from da ASC group by category_id")
d=sqldf("select count(video_id),category_id from a1 group by category_id")
d
f
```

```
x=c("Autos & Vehicles","Comedy","Education","Entertainment","Film & Animation",
"Games","Howto & Style","Movies","Music","News & Politics","People & Blogs",
"Science & Technology","Shows","Sports","Travel & Events")
```

```
Blogs","Pets & Animals","Science & Technology","Shows","Sports","Travel & Events")
y=c(7025,75802,21942,338641,43743,23735,46085,180,61460,79177,69879,9284,18312,250
2,55449,9816)
```

```
z=c(20,3,22,25,22,18,23,30,17,19,17,24,16,20,20,25)
```

```
da=data.frame(x,z)
da
```

```
ggplot(da,aes(x,z))+
  geom_segment(aes(x=x,xend=x,y=0,yend=z-1.5),color="grey",size=1)+
  geom_point(color="blue",size=8,alpha=0.6)+
  geom_text(label=z,color="white")+

  theme(panel.grid.major = element_blank(),
        panel.border = element_blank(),
        axis.ticks.y = element_blank(),
        axis.text.x = element_text(angle = 45,hjust=1))+ylab("Total tags")+xlab("Video
Category")+ggtitle("Average tags used in video")
```

8. CONCLUSION

This analysis is created for better understanding. It working fine in all situation and ready to get implemented in real world problems.

As perfection has no limit in the same way there are many future scopes where we try to implement more functionality to this analysis such as dynamic data like online data and do analysis etc.

9. BIBLIOGRAPHY

➤ References

- www.tutorialspoint.com
- www.r-tutor.com
- www.W3schools.in
- www.r-graph-gallery.com
- <https://rpubs.com/CosHoef/159665>