

TIPOLOGIA I CICLE DE VIDA DE LES DADES

PRÀCTICA 2

Ferran Gurri Mancera

24/05/2018

Índex

1.	Descripció del dataset	3
2.	Integració i selecció de les dades d'interès a analitzar	3
3.	Neteja de les dades	6
3.1	Les dades contenen zeros o elements buits? Com gestionaries aquests casos?	6
	Dies en els quals disposem de menys de 200 videos	6
	Likes, dislikes i comment_count a 0	7
	Videos marcats com erronis o trets de la xarxa	9
3.2	Identificació i tractament de valors extrems	10
4.	Anàlisi de les dades	13
4.1	Selecció dels grups de dades que es volen analitzar/comparar (planificació dels anàlisis a aplicar)	13
4.2	Comprovació de la normalitat i homogeneïtat de la variància	14
	Vistes	14
	Likes	15
	Dislikes	16
	Comment count	17
4.3	Aplicació de proves estadístiques per comparar els grups de dades. En funció de les dades i de l'objectiu de l'estudi, aplicar proves de contrast d'hipòtesis, correlacions, regressions, etc.	18
	Correlació Visites – Likes	19
	Correlació Visites – dislikes	19
	Correlació visites – numero de comentaris	20
	Regressió lineal per predir el nombre de visites	20
5.	Representació dels resultats a partir de taules i grafiques	24
6.	Resolució del problema. A partir dels resultats obtinguts, quines són les conclusions? Els resultats permeten respondre al problema?	26
7.	Codi: Cal adjuntar el codi, preferiblement en R, amb el que s'ha realitzat la neteja, anàlisi i representació de les dades. Si ho preferiu, també podeu treballar en Python	27

1. Descripció del dataset

Per l'elaboració d'aquesta pràctica he triat un dels datasets públics disponibles al portal <https://www.kaggle.com> suggerit en l'enunciat, concretament el que fa referència als vídeos de Youtube que són trending tòpic a Canadà (<https://www.kaggle.com/datasnaek/youtube-new>).

He descartat el dataset elaborat en la pràctica 1 perquè he considerat que no tenia prou dades numèriques per poder realitzar-ne una neteja de dades completa ni tampoc un estudi estadístic significatiu.

El dataset triat consta dels 200 vídeos trending tòpic per dia de cada i ens permetrà fer un estudi de quines són les categories que més impacte tenen en la xarxa social basada en vídeos més utilitzada a internet.

L'objectiu d'aquest projecte és analitzar els vídeos que arriben a ser trending tòpic (de més interès per la comunicació basat en el nombre de visualitzacions, comentaris, likes, etc) en funció de la seva categoria. Busquem doncs donar resposta a preguntes com ¿És Youtube una plataforma principalment de música? ¿En quina categoria ens hauríem d'especialitzar si volguéssim dedicar-nos professionalment a la pujada de vídeos? ¿Existeix temporalitat en l'èxit dels vídeos (depen el nombre de visites del mes en que es puja)? I, adicionalment, ¿Existeix una relació entre el nombre de visites i el nombre de likes, dislikes i nombre de comentaris?

2. Integració i selecció de les dades d'interès a analitzar

El dataset ve amb un conjunt considerable de columnes incloses i no totes ens aporten informació útil al problema que volem analitzar. Per fer una primera ullada a les dades descarregades les obrirem en format Excel i n'analitzarem el contingut. Adjunto aquest Excel intermig en el repositori de l'entrega. Complementarem aquesta informació amb les metadades de les columnes que disposem en el portal que conté el dataset original.

Esquematitzaré aquesta informació en la següent taula a mode de guia inicial en el procés de neteja:

Camp	Format	Descripció	Ens interessa?
video_id	alfanumeric	Codi identificador del video	Sí
trending_date	Data: YY.DD.MM	Dia en que va ser trending topic el video.	Sí
title	alfanumeric	Títol del video publicat	No
channel_title	Alfanumeric	Nom del canal que conté el video	No
category_id	Numeric	Codi identificador de la categoria	Sí
publish_time	Alfanumeric	Dia i hora en que es va publicar el video	No
Tags	Alfanumeric	Llista de tags definits per l'usuari. Venen separats per " "	No
Views	Numeric	Nombre de visites que ha rebut el	Sí

		video	
likes	Numeric	Nombre de “m’agrada” que ha rebut el video	Sí
dislikes	Numeric	Nombre de “No m’agrada” que ha rebut el video	Sí
comment_count	Numeric	Nombre de comentaris que conté el video	Sí
thumbnail_link	Alfanumeric	Enllaç a la vista prèvia del video	No
comments_disabled	Alfanumeric	“False” o “True” indicant si el video permet comentaris	Sí en cas de que volguem fer anàlisis sobre el nombre de comentaris del video
ratings_disabled	Alfanumeric	“False” o “True” indicant si el video permet valoracions	Sí, en cas de que volguem fer anàlisi sobre el nombre de likes/dislikes
video_error_or_removed	Alfanumeric	“False” o “True” indicant si el video ha estat eliminat (en aquest cas es conserven les dades que tenia en el moment de ser eliminat)	Sí, ja que pot esbiaixar les dades
description	Alfanumeric	Text obert on la persona que ha pujat el video n’escrui una descripció	No

Per tant el nostre dataset constarà de les columnes: video_id, trending_date, category_id, views, likes, dislikes, comment_count, comments_disabled, ratings_disabled , video_error_or_removed.

Algunes consideracions prèvies a destacar son:

- “trending_date” pren valors entre el 16 de novembre del 2017 (17.16.11) i el 7 d’abril del 2018
- el camp “category_id” és un identificador numèric intern, ens donen la seva equivalència en un fitxer CA_category_id.json a part.
- El camp de text obert “description” conté caràcters que trenquen l’estructura en línees del fitxer i caldrà revisar que es carreguen correctament
- Es probable que existeixi relació entre les columnes que enregistren el nombre de visites amb el nombre de “likes”, “dislikes” o “comment_count” ja que cal pensar que a més visites més probabilitat de tenir-ne activitat.
- Els camps “comments_disabled”, “ratings_disabled” i “video_error_or_removed” ens poden aportar valors zero o incomplets i requeriran un tractament a part si no volem que ens esbiaixin les dades. És a dir, si per exemple volem analitzar la distribució per nombre de comentaris realitzats i no tenim en compte si en el video els comentaris estan permesos estarem incloent moltes dades a 0 incorrectament. Un cas més problemàtic és el camp “video_error_or_removed”, ja que si ens indica que el video va

ser retirat llavors totes les dades fan referència al temps que va estar pujat i no sabem quines serien les seves dades totals en cas de no haver estat retirat.

- Per simplicitat de l'estudi, no tindrem en compte el camp "publish_time", però podria ser interessant intentar relacionar els dies que porta publicat el video amb l'activitat rebuda.
- Tampoc analitzarem el camp "tags" perquè volem fer l'anàlisi en funció de les categories. Una possible ampliació del estudi de les dades seria tallar la llista de tags rebudes en aquest camp i repetir-ne l'anàlisi prenent-les com una categorització alternativa.
- Amb unes transformacions bàsiques en Excel comprovem el nombre de videos per dia inclosos en les dades. En la descripció del dataset se'ns informa que s'han seleccionat els 200 videos más trending topic per dia, però observem que hi ha dies incomplets:

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
1																		
2	18.01.02	190	17.16.12	199	18.22.02	199	17.15.11	200	17.26.12	200	18.03.05	200	18.09.01	200	18.18.01	200	18.25.01	200
3	18.31.01	191	17.19.11	199	18.23.04	199	17.15.12	200	17.27.11	200	18.04.01	200	18.09.02	200	18.18.03	200	18.25.02	200
4	18.02.02	195	17.21.12	199	18.26.02	199	17.16.11	200	17.27.12	200	18.04.02	200	18.10.03	200	18.18.04	200	18.25.03	200
5	18.13.02	195	17.26.11	199	18.27.01	199	17.17.11	200	17.28.11	200	18.04.04	200	18.11.03	200	18.19.02	200	18.26.01	200
6	18.29.01	196	17.29.12	199	18.27.03	199	17.17.12	200	17.28.12	200	18.04.05	200	18.12.01	200	18.19.04	200	18.26.03	200
7	18.09.03	197	18.03.02	199	18.28.03	199	17.18.11	200	17.29.11	200	18.05.01	200	18.12.02	200	18.20.01	200	18.26.04	200
8	18.19.03	197	18.03.03	199	18.29.04	199	17.18.12	200	17.30.11	200	18.05.03	200	18.12.03	200	18.20.02	200	18.27.04	200
9	18.25.04	197	18.04.03	199	17.01.12	200	17.19.12	200	17.30.12	200	18.05.04	200	18.13.01	200	18.20.03	200	18.28.01	200
10	18.28.02	197	18.05.02	199	17.02.12	200	17.20.11	200	17.31.12	200	18.05.05	200	18.13.03	200	18.20.04	200	18.28.04	200
11	17.07.12	198	18.07.03	199	17.03.12	200	17.20.12	200	18.01.01	200	18.06.01	200	18.14.01	200	18.21.01	200	18.29.03	200
12	18.06.02	198	18.14.02	199	17.05.12	200	17.21.11	200	18.01.03	200	18.06.03	200	18.15.01	200	18.21.03	200	18.30.01	200
13	18.10.02	198	18.14.03	199	17.06.12	200	17.22.11	200	18.01.04	200	18.06.04	200	18.15.03	200	18.21.04	200	18.30.03	200
14	18.11.02	198	18.14.04	199	17.08.12	200	17.22.12	200	18.01.05	200	18.06.05	200	18.15.04	200	18.22.03	200	18.30.04	200
15	18.15.02	198	18.16.02	199	17.09.12	200	17.23.11	200	18.02.01	200	18.07.01	200	18.16.01	200	18.22.04	200	18.31.03	200
16	18.23.02	198	18.17.02	199	17.10.12	200	17.23.12	200	18.02.03	200	18.07.02	200	18.16.03	200	18.23.01	200		
17	18.24.02	198	18.18.02	199	17.11.12	200	17.24.11	200	18.02.04	200	18.07.04	200	18.16.04	200	18.23.03	200		
18	18.27.02	198	18.19.01	199	17.13.12	200	17.24.12	200	18.02.05	200	18.08.01	200	18.17.01	200	18.24.01	200		
19	17.04.12	199	18.21.02	199	17.14.11	200	17.25.11	200	18.03.01	200	18.08.02	200	18.17.03	200	18.24.03	200		
20	17.12.12	199	18.22.01	199	17.14.12	200	17.25.12	200	18.03.04	200	18.08.03	200	18.17.04	200	18.24.04	200		

Crec que és poc probable que aquests dies incomplets es tracti de dies que no s'han arribat a publicar 200 videos en un dia. Estem per tant davant de un cas de dades buides perquè manquen registres que sabem que existeixen però no tenim.

Tractaré aquests casos en apartats següents de la pràctica.

Carregarem doncs les nostres dades mitjançant les següents instruccions en R:

Carreguem el csv:

```
> youtubes_raw <- read.csv("C:/uoc/Semestre 1/Tipologia i cicle de vida de les dades/Practica2/CAvideos.csv");
```

Seleccionem les columnes que volem fer servir en el nostre anàlisi:

```
> youtubes <- youtubes_raw[, c(1,2,5,8,9,10,11,13,14,15)];
```

Filtrem les columnes sense video_id informat per assegurar-nos que la càrrega no ha tingut problemes amb els registres on el camp "description" trencava l'estructura de linees:

```
> youtubes_clean <- youtubes[is.na(youtubes$video_id) == 0,];
```

Treiem un primer resum del estat de les dades després d'aquesta primera càrrega inicial:

```
> summary(youtubes_clean)
video_id      trending_date  category_id  views      likes      dislikes  comment_count  comments_disabled  rating
6ZfuNTqbHE8:  8  17.01.12: 200  Min. : 1.00  Min. : 1000  Min. : 0  Min. : 0.0  Min. : 0  False:32628  False:
1_lblj8Cq0o:  8  17.02.12: 200  1st Qu.:20.00  1st Qu.: 137261  1st Qu.: 2058  1st Qu.: 92.0  1st Qu.: 390  True : 486  True :
UceaB4D0jpo:  8  17.03.12: 200  Median :24.00  Median : 354587  Median : 8637  Median : 289.0  Median : 1272
9v_rtaye2yY:  7  17.05.12: 200  Mean :20.75  Mean : 1113774  Mean : 38533  Mean : 2000.9  Mean : 4839
Cmg2BnAv4WQ:  7  17.06.12: 200  3rd Qu.:24.00  3rd Qu.: 937967  3rd Qu.: 28194  3rd Qu.: 933.8  3rd Qu.: 3623
doF7xKdGOKs:  7  17.08.12: 200  Max. :43.00  Max. :137843120  Max. :3014479  Max. :1602383.0  Max. :827755
(Other)      :33069  (Other) :31914
video_error_or_removed
False:33088
True : 26
```

Observem que:

- Tenim 33114 registres
- No hi ha cap NA en cap dels camps
- Crida l'atenció que el nombre mínim de visites sigui un número rodó 1000. No he trobat confirmació sobre si només es tenen en compte videos que tinguin com a mínim 1000 visites a l'hora de considerar-los candidats a ser trending tòpic. Només se'ns diu que la API de Youtube té en compte diferents factors a l'hora de marcar un video com a Trending Tòpic. Que existís un nombre mínim de visites podria ser una explicació al fet que alguns dies no s'arribin a 200 videos.
- Existeixen videos amb 0 likes, dislikes o comment_count. Tal i com hem indicat abans, això és degut a que es poden deshabilitar a l'hora de publicar el video.

3. Neteja de les dades

3.1 Les dades contenen zeros o elements buits? Com gestionaries aquests casos?

En el tractament previ de les dades ja hem detectat diferents casos de zeros i elements buits. Considerarem cada cas per separat.

Dies en els quals disposem de menys de 200 videos

En la descripció del dataset se'ns diu que les dades s'han generat a partir de recollir els 200 videos més trending topic per dia al Canadà. En les dades en canvi observem que hi ha dies que no tenim 200 registres. Aquest és un cas d'elements buits on no ens falta el valor de un camp sino el registre sencer que sabem que sí existeix (costa imaginar-se que en un dia no hi hagi 200 videos amb prou activitat en tot un país).

No trobo cap aclariment en la documentació sobre aquest punt en concret. La interpretació que jo faig és que la API de Youtube fa un filtre previ dels videos que té que considerar i que els videos que falten no compleixen algun d'aquests mínims. Una possibilitat que ja he comentat és que és probable que hi hagi un mínim de 1000 vistes per considerar un video com a candidat a ser trending tòpic.

Devant la impossibilitat de reproduir les dades que sabem que falten i tenint en compte que l'impacte d'aquesta manca de informació en l'estudi que volem realitzar és mínim prenem la mesura de ignorar aquestes dades incompletes. Ho assumirem com una limitació del sistema i informariem a l'origen de dades perquè ho tinguin en compte.

L'altra possibilitat seria un error puntual en la recollida de dades, però tampoc tenim informació sobre el procés que captura les dades. En el cas en que les dades de un dia en concret només s'intentessin recollir una única vegada llavors podríem pensar en un error del procés i en suggeriríem que es repetís la càrrega n-vegades per solventar errors puntuals.

Likes, dislikes i comment_count a 0

Les columnes "likes", "dislikes" i "comment_count" ens permetran relacionar l'activitat realitzada sobre el video amb el nombre de visites i, per tant, són significatives pel nostre estudi. En el cas dels videos on no es registren aquestes dades perquè està deshabilitada la opció de premer el "like", el "dislike" o la possibilitat d'introduir algún comentari tindrem aquestes columnes a 0 i ens esbiaixaran els resultats. Així doncs no podem deixar aquestes dades a zero.

Les opcions disponibles davant d'aquest cas són:

- Opció 1: Eliminar aquests registres del nostre conjunt de dades
- Opció 2: Informar-los amb el valor promig dels registres que sí tenim informats
- Opció 3: Fer una estimació del valor més probable en funció del nombre de visites. És a dir, calcular la relació entre el nombre de visites i el número de likes (per exemple) i crear el ratio likes_per_visita i el mateix per cada columna

Opció 1: no tractar aquestes dades

Les dades amb valors a zero suposen un percentatge molt baix respecte al total de registres. L'estimació més pesimista seria que cada comentari o valoració bloquejada pertany a videos diferents. És a dir, que no hi ha cap cas on en un mateix video s'hi han bloquejat els comentaris i també els "m'agrada"/"no m'agrada".

Així doncs el màxim de videos afectats son: 486 (comentaris bloquejats) + 245 (valoracions bloquejades) = 731 sobre un total de 33114 registres, lo que suposa un $(731/33114)*100 = 2,2\%$ del total.

Aquesta opció no seria dolenta ja que ens permetria quedar-nos només amb dades netes sense perdre gran quantitat d'informació.

Opció 2: informar les dades a zero amb el valor promig

Separariem les dades no zero de les zero i en calcularíem el valor promig per després actualitzar els registres amb dades zero.

Aquesta opció és bona perquè no ens modificarà els estimadors estadístics i podrem disposar de la resta de informació de les columnes ben informades.

Opció 3: demostrar la correlació entre les variables a zero i les vistes i aplicar els ratios als registres a zero per deduir-ne el resultat

És la opció més ambiciosa en el sentit en que intentem deduir el valor que ens hagués arribat en cas de no venir a zero. No obstant en aquesta fase del projecte no tenim demostrada estadísticament la correlació entre variables. Descartarem doncs aquesta opció.

Considero que la millor opció és la 2. Aquestes són les comandes en R per actualitzar les dades a zero amb el valor promig:

Primer comprovem que el subconjunt amb “ratings_disabled” a “True” té efectivament els valors “likes” i “dislikes” a 0:

```
> youtube_clean_no_likes <- youtube_clean[youtube_clean$ratings_disabled == "True",]
> summary(youtube_clean_no_likes)
      video_id      trending_date      category_id      views      likes      dislikes      comment_count      comments_disabled      ratings_disabled
Bh1EIO0vaBE: 6      17.22.12: 6      Min. : 1.00      Min. : 2307      Min. : 0      Min. : 0      Min. : 0.0      False:186      False: 0
nx1R-eHSkFM: 6      18.05.02: 5      1st Qu.:24.00      1st Qu.: 54876      1st Qu.:0      1st Qu.:0      1st Qu.: 3.0      True : 59      True :245
z3XJdMwAvZI: 5      18.19.03: 5      Median :24.00      Median : 139664      Median :0      Median :0      Median : 202.0
-H90GFnHq8: 4      18.04.02: 4      Mean :21.99      Mean : 1529910      Mean :0      Mean :0      Mean : 745.3
gQnBX6e3RJc: 4      18.24.03: 4      3rd Qu.:25.00      3rd Qu.: 709490      3rd Qu.:0      3rd Qu.:0      3rd Qu.:1031.0
mRgqSaePz0w: 4      18.27.02: 4      Max. :29.00      Max. :51243149      Max. :0      Max. :0      Max. :5953.0
(Other) :216      (Other) :217
video_error_or_removed
False:245
True : 0
```

Calculem la mitjana del valor sense aquests registres:

```
> youtube_clean_likes <- youtube_clean[youtube_clean$ratings_disabled == "False",]
> like_mean = mean(youtube_clean_likes$likes);
> like_mean
[1] 38820.15
```

Actualitzem els valors:

```
> youtube_clean <- within(youtube_clean, likes[ratings_disabled == "True"] <- like_mean);
```

Comprovem la mitjana després d’haver actualitzat:

```
> mean(youtube_clean$likes);
[1] 38820.15
```

Repetim el procés però amb el camp “dislike”:

```
> dislike_mean = mean(youtube_clean_likes$dislikes);
> dislike_mean
[1] 2015.779
> youtube_clean <- within(youtube_clean, dislikes[ratings_disabled == "True"] <- dislike_mean);
> mean(youtube_clean$dislikes);
[1] 2015.779
```


I de forma anàloga tractem el camp “comment_count”:

```
> youtubes_clean_no_comments <- youtubes_clean[youtubes_clean$comments_disabled == "False",];
> comments_mean <- mean(youtubes_clean_no_comments$comment_count);
> youtubes_clean <- within(youtubes_clean, comment_count[comments_disabled == "True"] <- comments_m
> mean (youtubes_clean_no_comments$comment_count);
[1] 4910.925
> mean (youtubes_clean$comment_count);
[1] 4910.925
```

Surgeix ara una darrera pregunta. Hem modificat tots els valors a zero? Existeix encara algun registre amb “likes”, “dislikes” o “comment_count” amb valor 0? Comprovem-ho:

```
> youtubes_still_zeros <- youtubes_clean[youtubes_clean$likes == 0 | youtubes_clean$dislikes == 0 | youtubes_clean$comment_count == 0,];

> summary(youtubes_still_zeros);
      video_id      trending_date      category_id      views      likes      dislikes      comment_count      comments_disabled ratings_disabled
BhIEIO0vaBE: 6 17.16.11: 8 Min. : 1.00 Min. : 1141 Min. : 0.00 Min. : 0.00 Min. : 0.0 False:338 False:155
nxIR-eISkRf: 6 18.01.02: 7 1st Qu.:17.00 1st Qu.: 6588 1st Qu.: 98.75 1st Qu.: 0.00 1st Qu.: 8.0 True: 62 True:245
z3XJdMwAvZl: 5 18.05.02: 7 Median :24.00 Median : 54914 Median :38820.00 Median : 0.00 Median :137.5
-H9qSPnHlq8: 4 18.05.03: 7 Mean :21.21 Mean : 961266 Mean :24083.64 Mean : 20.39 Mean :1222.4
8XiqERZq_8: 4 18.27.02: 7 3rd Qu.:25.00 3rd Qu.: 240759 3rd Qu.:38820.00 3rd Qu.: 0.00 3rd Qu.:1642.5
gGnBX6e3RJC: 4 18.30.03: 7 Max. :29.00 Max. :51243149 Max. :38820.00 Max. :1550.00 Max. :5953.0
(Other) :371 (Other) :357
video_error_or_removed
False:397
True : 3
```

Observem que encara hi ha registres amb dades a zero en els camps “likes”, “dislikes” o “comment_count”. ¿Què hauríem de fer amb aquests registres? ¿Són un nou cas de valor zero que ha sorgit després d’arreglar-ne el primer?

Sense més informació sobre el dataset considero que tinc que tractar aquests valors com a reals malgrat que se’m fa difícil de creure que un video seleccionat entre els 200 de més trending topic en un país no tingui activitat en aquests camps.

Videos marcats com erronis o trets de la xarxa

Ens queda encara per tractar un altre cas de dades que no són ni buïdes ni a zero pero sí incompletes: els videos que ens venen marcats amb el camp “video_error_or_removed” a cert.

No disposem de informació detallada sobre la casuística d’aquest camp. Hem de suposar que es tracta de videos que contenen algun error de visualització o que han estat trets potser per incomplir les condicions d’us de la plataforma o per decisió del propietari del canal.

Anem a comprovar de quants casos estem parlant:

```
> summary(youtubes_clean);
      video_id      trending_date      category_id      views      likes      dislikes      comment_count      comments_disabled ratings_disabled
6ZfuUTqbHE8: 8 17.01.12: 200 Min. : 1.00 Min. : 1000 Min. : 0 Min. : 0.0 Min. : 0 False:32628 False:32869
1_lblj8CqQo: 8 17.02.12: 200 1st Qu.:20.00 1st Qu.: 137261 1st Qu.: 2156 1st Qu.: 96.0 1st Qu.: 426 True: 486 True: 245
UCeaB4D0jpo: 8 17.03.12: 200 Median :24.00 Median : 354587 Median : 8926 Median : 298.5 Median : 1350
Sv_rtaey2yY: 7 17.05.12: 200 Mean :20.75 Mean : 1113774 Mean : 38820 Mean : 2015.8 Mean : 4911
Cmg2BnAv4WQ: 7 17.06.12: 200 3rd Qu.:24.00 3rd Qu.: 937967 3rd Qu.: 29280 3rd Qu.: 972.8 3rd Qu.: 3890
doP7xRdGOK: 7 17.08.12: 200 Max. :43.00 Max. :137843120 Max. :3014479 Max. :1602383.0 Max. :827755
(Other) :33069 (Other) :31914
video_error_or_removed
False:33088
True : 26
```

Veiem que hi ha 26 registres amb el camp “video_error_or_removed” a True. Com que són molt pocs casos i tenim poc coneixement de funcionament d’aquest camp decideixo treure’ls de la mostra.

```
> youtube_clean_no_error <- youtube_clean[youtube_clean$video_error_or_removed == "False",]
> summary(youtube_clean_no_error);
```

video_id	trending_date	category_id	views	likes	dislikes	comment_count	comments_disabled	ratings_disabled
6ZfuNTqbHE8	17.01.12	200	Min. : 1.00	Min. : 1000	Min. : 0	Min. : 0	Min. : 0	False:32602
1_lblj8CqDo	17.02.12	200	1st Qu.:20.00	1st Qu.: 137248	1st Qu.: 2157	1st Qu.: 96	1st Qu.: 426	False:32843
UGeaB4D0jpo	17.03.12	200	Median :24.00	Median : 354462	Median : 8926	Median : 298	Median : 1350	True : 486
9v_rtaYe2yY	17.06.12	200	Mean :20.75	Mean : 1113513	Mean : 38811	Mean : 2015	Mean : 4909	
Cmg2BnAv4WQ	17.08.12	200	3rd Qu.:24.00	3rd Qu.: 937059	3rd Qu.: 29274	3rd Qu.: 972	3rd Qu.: 3889	
doP7xKdGOKs	17.09.12	200	Max. :43.00	Max. :137843120	Max. :3014479	Max. :1602383	Max. :827755	
(Other)								

```
video_error_or_removed
False:33088
True : 0
```

Ara la columna “video_error_or_removed” no aporta informació al conjunt de dades, així que la trec del dataset:

```
> youtube_clean_no_error <- youtube_clean_no_error[, c(1,2,3,4,5,6,7,8,9)];
```

No he detectat cap cas més de dades a zero o buïdes.

3.2 Identificació i tractament de valors extrems

Fent una ullada a les dades ja es veu que tenim unes dades molt disperses tot i tractar-se de videos amb les mateixes característiques: tots ells han sigut trending topic.

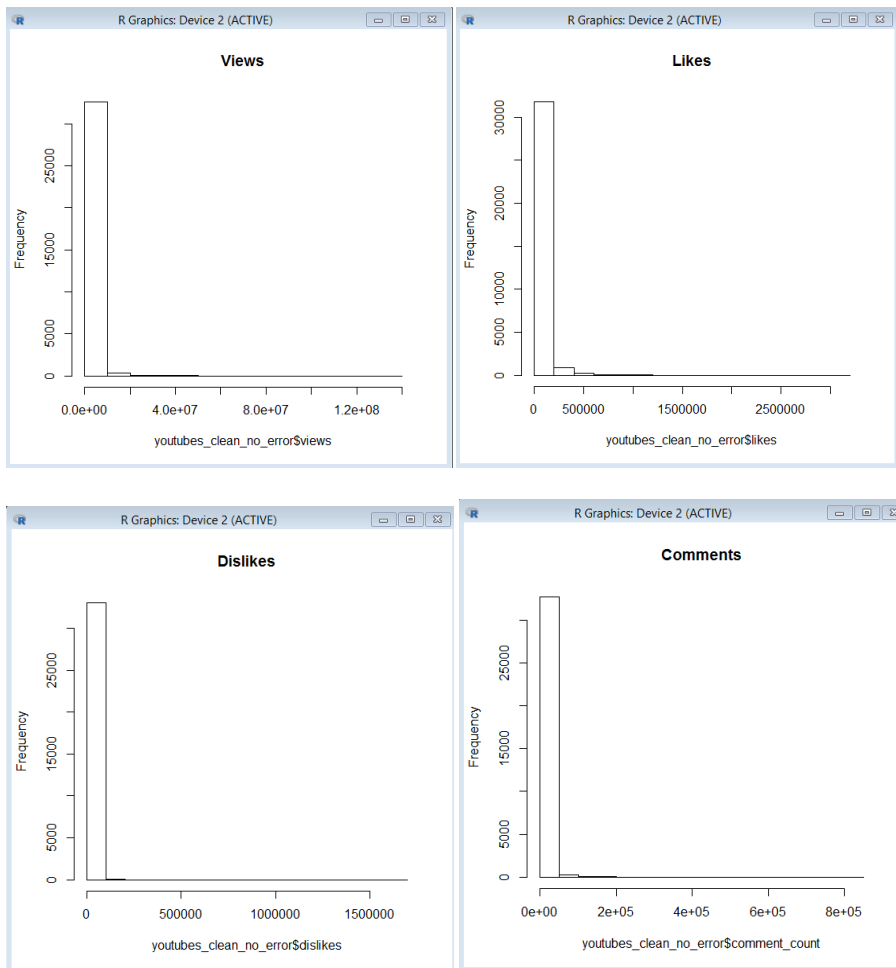
views		likes		dislikes		comment_count	
Min. :	1000	Min. :	0	Min. :	0	Min. :	0
1st Qu.:	137248	1st Qu.:	2157	1st Qu.:	92	1st Qu.:	426
Median :	354462	Median :	8926	Median :	288	Median :	1350
Mean :	1113513	Mean :	38811	Mean :	2000	Mean :	4909
3rd Qu.:	937059	3rd Qu.:	29274	3rd Qu.:	933	3rd Qu.:	3889
Max. :	137843120	Max. :	3014479	Max. :	1602383	Max. :	827755

Tenim el mateix comportament en qualsevol de les quatre variables: valors màxims molt per sobre de la mitjana i encara més per sobre del 3er Quadrant.

Agafant d’exemple l’atribut “views” veiem que el tercer quadrant ve marcat pel valor 937 mil mentre que el màxim és de més de 137 milions de visites. Per la part inferior tenim també un mínim de 1000 que és mil vegades inferior que la mitjana (Mean) del conjunt de la mostra. Podem afirmar doncs que tenim valors extrems.

Treuré grafiques per cada atribut per comprovar visualment l’existència de valors extrems i l’impacte que tenen:

```
> hist(youtubes_clean_no_error$views, main="Vistes");
> hist(youtubes_clean_no_error$likes, main="Likes");
> hist(youtubes_clean_no_error$dislikes, main="Dislikes");
> hist(youtubes_clean_no_error$comment_count, main="Comments");
```



Podem observar en les gràfiques com l'existència de valors extrems serà un inconvenient en qualsevol estudi estadístic que intentem realitzar. Cap dels histogrames aporta massa informació degut a que els valors extrems provoquen que quasi tota la mostra quedi en una única columna.

Abans de prendre cap mesura pel tractament d'aquests valors extrems hem de intentar esbrinar de quin tipus de outliers es tracta. ¿Són errors en la recollida de dades o són valors reals?

Considero que són valors legítims perquè:

1. Ja hem eliminat de la mostra els registres amb "video_error_or_removed" a cert i, per tant, hem de confiar en que la mostra ve lliure d'errors
2. El propi comportament viral de les xarxes socials afavoreix l'aparició de diferències exponencials entre videos
3. En l'estudi de les dades realitzat fins ara, considero que caldria revisar la metodologia de la captura de dades. Crec que agafar una quantitat fixe per dia pot no ser la millor opció ja que hi ha dies de menys activitat que d'altres. Un mètode millor de captura de

dades potser seria analitzar-ho per períodes de temps més amplis (setmana, mes, etc) o bé establir uns mínims d'activitat per considerar un video com a trending topic i agafar tots els que compleixin aquest mínim encara que en un dia en poguessim recollir 350 i en un altre 50

Per desenvolupar més el que exposo en el punt 3, diré que crec que l'estudi té un esbiaixament degut a la manera com les dades han estat recollides pels motius que ja he exposat. Asumiré aquest esbiaixament com a inevitable i segueixo endavant amb l'estudi.

Per tractar valors extrems legítims tenim diferents opcions¹:

- **Opció 1:** truncar les dades definint un valor màxim i fixar-lo per totes les dades que el superin
- **Opció 2:** transformacions matemàtiques de les dades. Podem, per exemple, aplicar l'arrel quadrada a tots els valors de manera que els més extrems perderan més magnitud que la resta durant el procés.
- **Opció 3:** calcular els estadístics sense tenir en compte els valors extrems
- **Opció 4:** discretitzar els valors i classificar-los segons un conjunt finit de valors com, per exemple: [Molt Baix, Baix, Mitja, Alt, Molt Alt]. Aquesta opció afavoreix aplicar models de classificació
- **Opció 5:** normalitzar tots els valors al interval [0,1]. Aquesta opció afavoreix l'estudi gràfic de les dades

No existeix una opció correcta universal, hem de tenir en compte l'objectiu de cada anàlisi. En el nostre cas l'objectiu és fer un estudi de l'activitat que reben els videos més vistos en funció de la categoria per identificar-ne les més interessants a l'hora de penjar-hi contingut professionalment.

No ens interessen les dades concretes, sino analitzar conjunts d'èxit. Per una banda, la quantitat de videos que conté cada categoria ja és un indicador perquè recordem que el conjunt de dades ja conté videos trending topic. A part d'això, ens interessa tenir mesures de videos de molt èxit, d'aquesta manera podrem analitzar cada categoria des de dues perspectives: volum de videos i qualitat del volum (si conté molts videos de molt èxit o no).

Per tant considero que ens interessa aplicar transformacions que discretitzin els valors en {Baix, Mitja, Alt }.

Les comandes R per discretitzar aquests valors són:

Explicaré al detall cada pas per la columna "Views" i després inclouré les comandes per la resta d'atributs. He creat un alias "yf" al dataset per reduir el tamany de les comandes.

Afegeixo el camp discretitzat per les vistes amb el seu valor per defecte:

```
> yf$d_views="Mitja";
```

¹ Jason W. Osborne (2010). Data Cleaning Basics: Best Practices in Dealing with Extreme Scores. Newborn and Infant Nursing Reviews; 10 (1): pp. 1527-3369

Actualitzo els valors per sota del primer quadrant amb el valor “Baix”:

```
> # lower quad  
> yf <- within(yf, d_views[yf$views <= 137248 ] <- "Baix");
```

El valor 137248 està tret de fer un `summary(yf)` i prenent el valor proporcionat per “1st Qu.”

Actualitzo els valors per sobre del tercer quadrant amb el valor “Alt”:

```
> # upper quad  
> yf <- within(yf, d_views[yf$views >= 937059 ] <- "Alt");
```

Aconseguint així una distribució dels valors en {Baix, Mitja, Alt} amb significació estadística.

Repetim el procés per la resta de variables:

```
> yf$d_likes="Mitja";  
> # lower quad  
> yf <- within(yf, d_likes[yf$likes <= 2059 ] <- "Baix");  
> # upper quad  
> yf <- within(yf, d_likes[yf$likes >= 28189 ] <- "Alt");  
> yf$d_dislikes="Mitja";  
> # lower quad  
> yf <- within(yf, d_dislikes[yf$dislikes <= 92 ] <- "Baix");  
> # upper quad  
> yf <- within(yf, d_dislikes[yf$dislikes >= 933 ] <- "Alt");  
> yf$d_comment_count="Mitja";  
> # lower quad  
> yf <- within(yf, d_comment_count[yf$comment_count <= 426 ] <- "Baix");  
> # upper quad  
> yf <- within(yf, d_comment_count[yf$comment_count >= 3889 ] <- "Alt");
```

4. Anàlisi de les dades

4.1 Selecció dels grups de dades que es volen analitzar/comparar (planificació dels anàlisis a aplicar)

Tenim les nostres dades discretitzades però diferents mesures a triar. Hauríem d’analitzar per nombre de visites? Per “likes”? Per nombre de comentaris? Sospito que el nombre de visites ja és un indicador suficient ja que per lògica a més visites més probabilitats d’aconseguir algun “like” o de que algú hi faci algun comentari. Comprovarem aquesta hipòtesi buscant correlació entre vistes i la resta de mesures. Un cop demostrada aquesta correlació podrem extreure les nostres conclusions analitzant la relació entre el nombre de visites i la categoria a la que pertany el video.

Aprofitarem també que disposem de prou mesos de dades per buscar correlació entre el mes on es publica el video i el nombre de vistes. Per aconseguir afegiré al dataset una variable “month” calculada a partir de la data “trending_date” mitjançant les següents comandes en R:

```
> yf$month=paste(substr(yf[, "trending_date"],1,2),substr(yf[, "trending_date"],7,9), sep="");
```

La nova columna “month” té el següent format: YYYY on YY són les dues darreres xifres de l’any i MM indica el mes.

Cal destacar que tenim dades des del novembre del 2017 fins al maig del 2018 però els primer i el darrer mes estan incomplets.

Adicionalment, aplicarem regressió lineal per buscar correlacions entre diferents variables del nostre conjunt de dades. Ordenarem les dades aleatòriament i dividirem el conjunt en un 80% per construir el model i un 20% per testejarlo mitjançant prediccions.

```
> #calc sample sizes  
> yf_random <- yf[sample( nrow( yf ) ),,];  
> # first 80%  
> train <- yf_random[1:26470,];  
> # last 20%  
> test <- yf_random[26471:33088,];
```

26470 representen el 80% del total de la mostra (33088 valors).

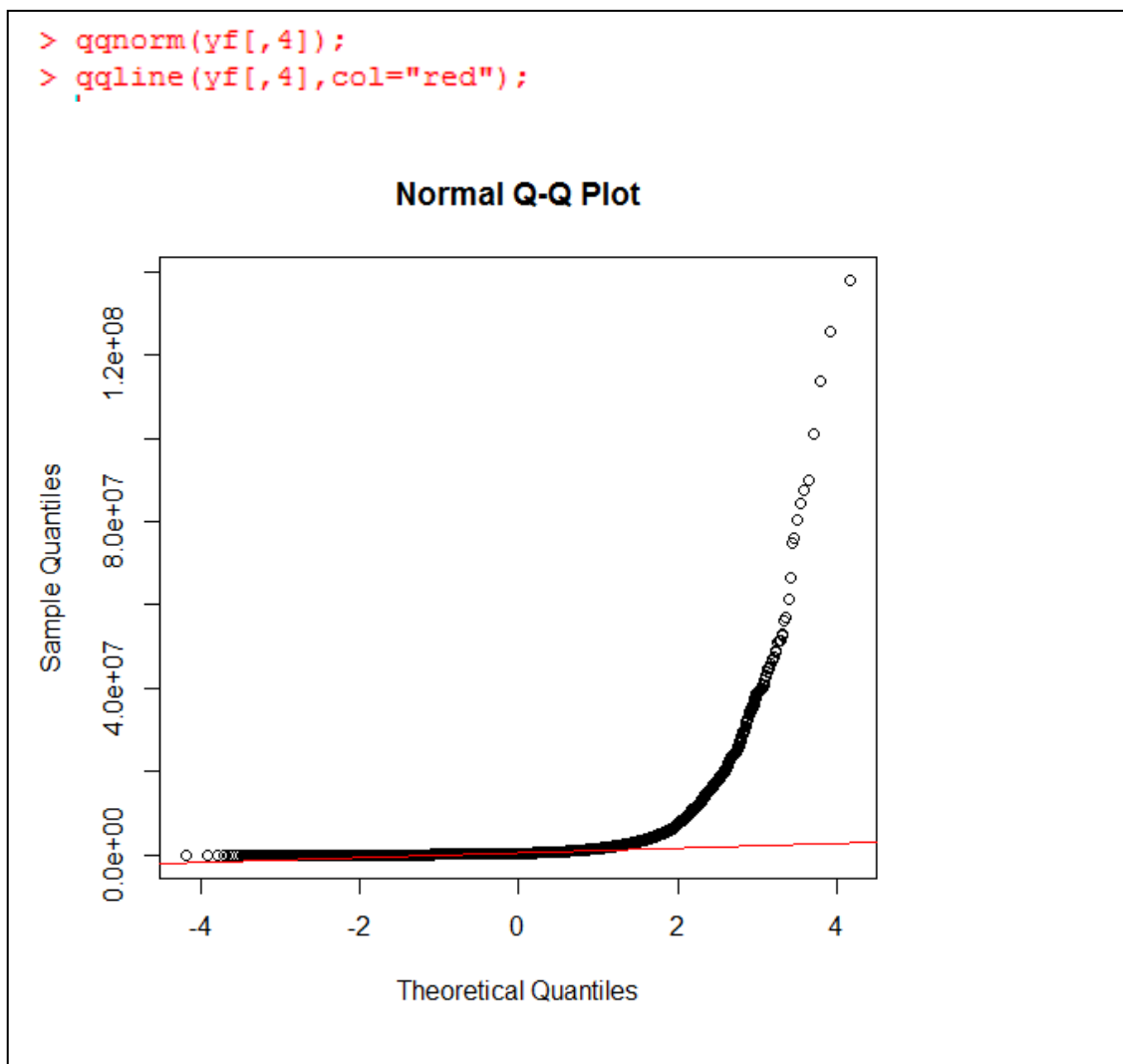
Aplicarem diferents fórmules de regressió lineal entre variables en l’apartat 4.3

4.2 Comprovació de la normalitat i homogeneïtat de la variancia

Comprovarem la normalitat de cadascuna de les variables numèriques implicades en el dataset mitjançant la representació visual per Q-Q plot. En dibuixarem la distribució dels punts i la linea teòrica de normalització. Si la majoria de punts segueixen la linea llavors podem afirmar que segueixen una distribució normal. Afegirem també un test de shapiro en cada variable per comprovar si les dades ja segueixen una distribució normal. Aquest test per defecte accepta mostres de fins a 5000 registres, per tant n’escollirem aquesta quantitat al atzar.

Recordem que no hem aplicat mesures pels outliers perquè treballem amb una variable discretitzada nova. Això ens afectarà als estudis estadístics que realitzem sobretot en els valors més alts.

Vistes



Podem veure l'efecte de no haver tret els outliers de les dades originals. Tot i això considero que està provat que la variable es podria normalitzar si fos necessari.

Ens falta comprovar si les dades actuals ja segueixen una distribució normal.

```

> shapiro.test(yf[sample(nrow(yf), 5000), "views"]);

```

Shapiro-Wilk normality test

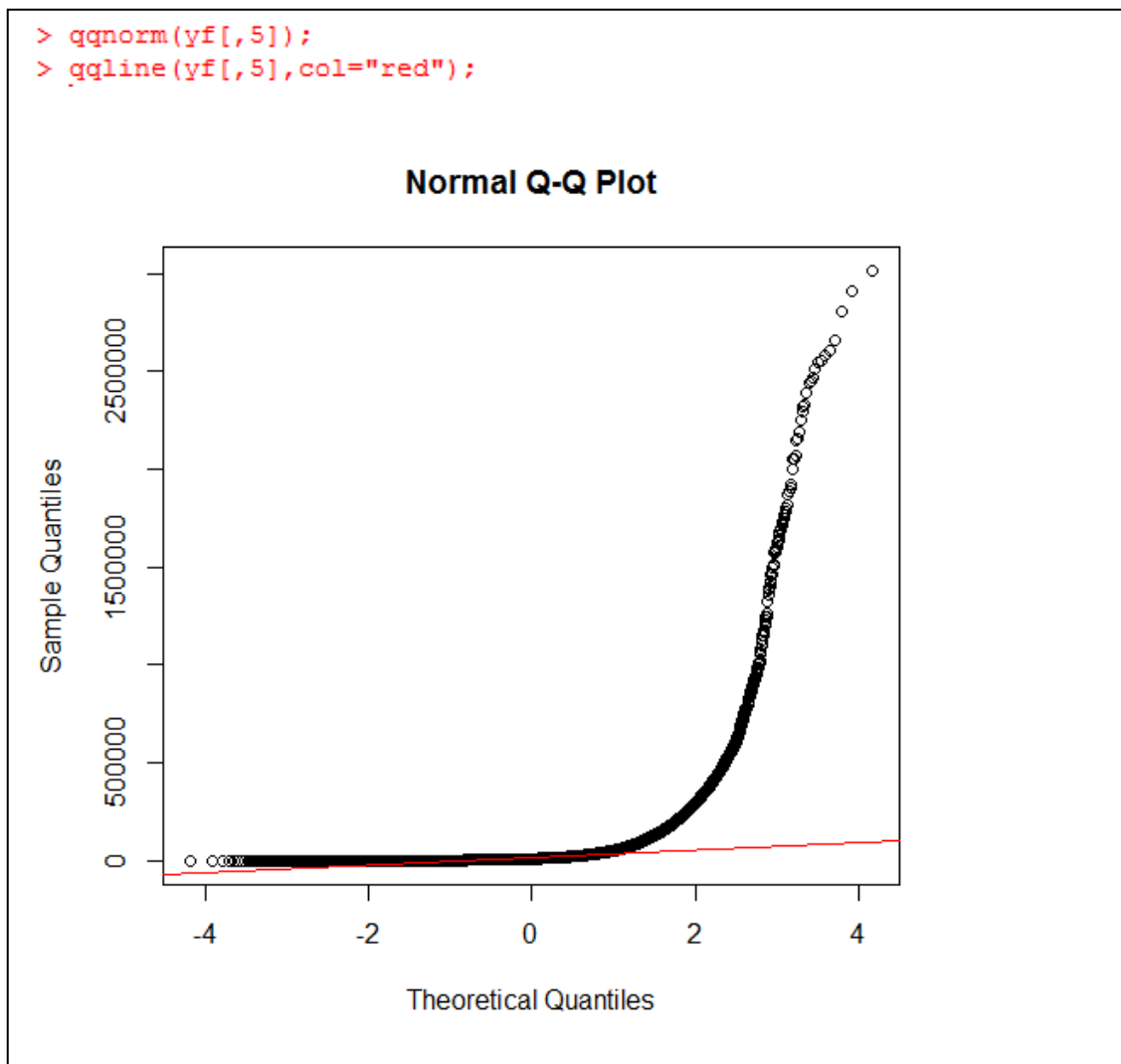
```

data: yf[sample(nrow(yf), 5000), "views"]
W = 0.24931, p-value < 2.2e-16

```

El p-valor es menor a 0.05 i per tant es rebutja la hipòtesis de que la variable segueixi una distribució normal.

[Likes](#)



Observem el mateix comportament que amb les vistes.

Ens falta comprovar si les dades actuals ja segueixen una distribució normal.

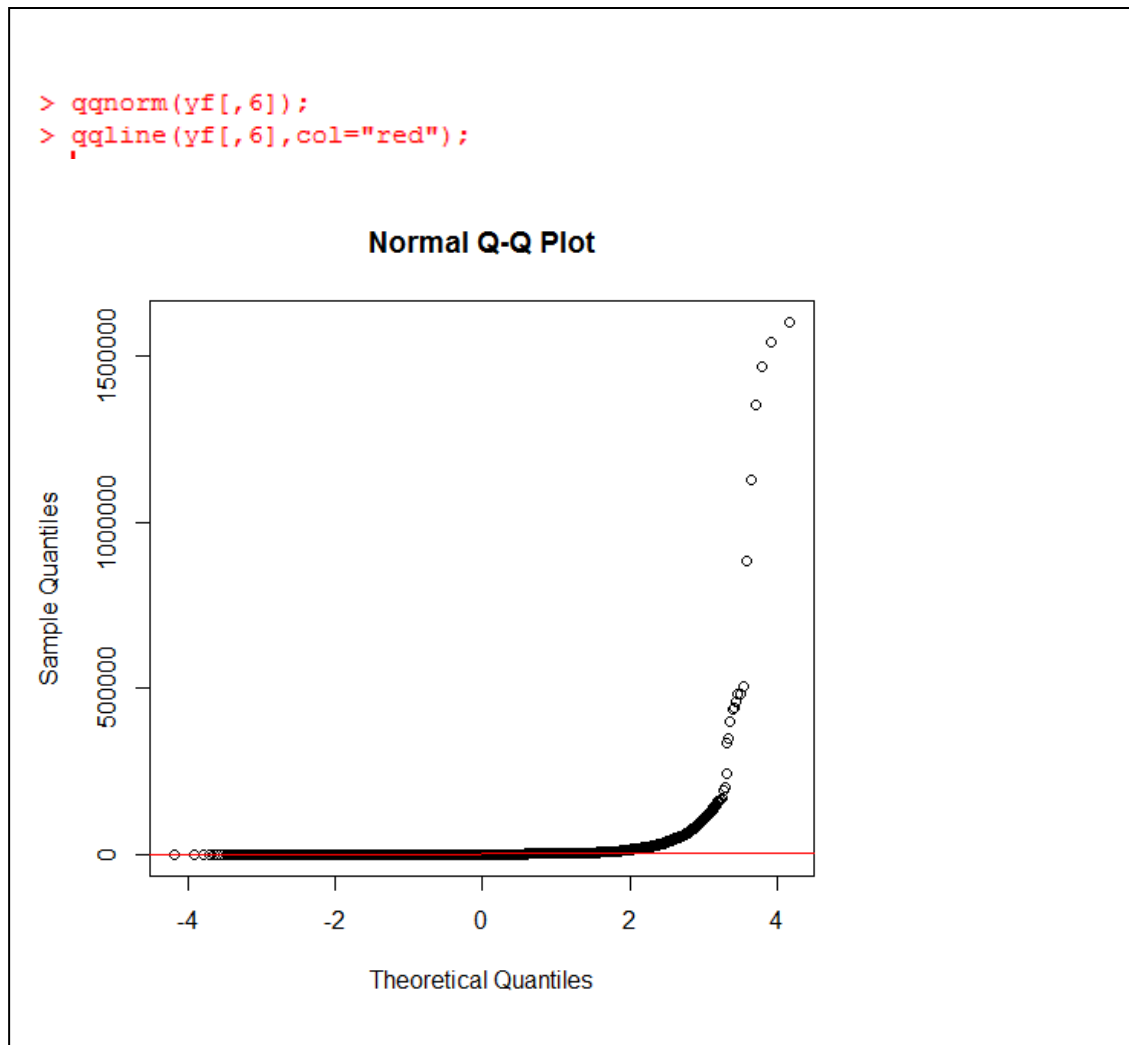
```
> shapiro.test(yf[sample(nrow(yf), 5000), "likes"]);
```

Shapiro-Wilk normality test

```
data: yf[sample(nrow(yf), 5000), "likes"]  
W = 0.32808, p-value < 2.2e-16
```

El p-valor es menor a 0.05 i per tant es rebutja la hipòtesis de que la variable segueixi una distribució normal.

Dislikes



Observem el mateix comportament que amb les vistes.

Passem ara el test de Shapiro per comprovar si les dades ja estan normalitzades:

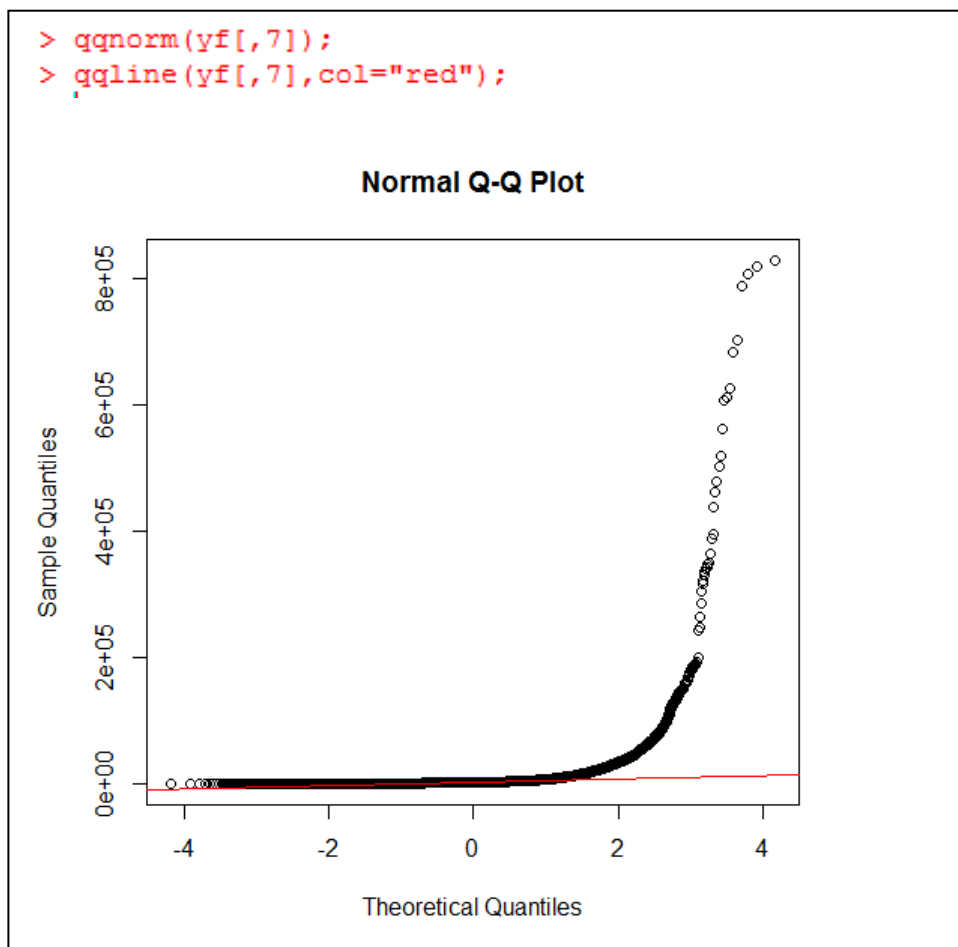
```
> shapiro.test(yf[sample(nrow(yf), 5000), "dislikes"]);
```

```
Shapiro-Wilk normality test
```

```
data:  yf[sample(nrow(yf), 5000), "dislikes"]
W = 0.034489, p-value < 2.2e-16
```

El p-valor es menor a 0.05 i per tant es rebutja la hipòtesis de que la variable segueixi una distribució normal.

[Comment count](#)



Observem el mateix comportament que amb les vistes.

Passem el test de Shapiro per comprovar si les dades ja estan normalitzades:

```

> shapiro.test(yf[sample(nrow(yf), 5000), "comment_count"]);

      Shapiro-Wilk normality test

data:  yf[sample(nrow(yf), 5000), "comment_count"]
W = 0.24577, p-value < 2.2e-16

```

El p-valor es menor a 0.05 i per tant es rebutja la hipòtesis de que la variable segueixi una distribució normal.

4.3 Aplicació de proves estadístiques per comparar els grups de dades.
En funció de les dades i de l'objectiu de l'estudi, aplicar proves de contrast d'hipòtesis, correlacions, regressions, etc.

Les gràfiques anteriors quasi idèntiques ja ens indiquien que hi haurà una correlació entre variables. Anem a verificar-ho amb proves estadístiques.

Utilitzarem el test de correlació de Pearson el qual ens dona un valor entre 1 i -1 que indica la relació entre dues variables i on 1 significa una correlació positiva absoluta i -1 negativa. El valor 0 indicaria que les dues variables no tenen cap correlació.

Correlació Visites – Likes

```
> cor.test(yf$views,yf$likes,method="pearson");

Pearson's product-moment correlation

data:  yf$views and yf$likes
t = 269.26, df = 33086, p-value < 2.2e-16
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.8252370 0.8319902
sample estimates:
      cor 
0.8286438
```

El valor de correlació és de 0.8286438 i indica una correlació positiva molt alta. Podem afirmar que a més visites, més “likes”.

Correlació Visites – dislikes

```
> cor.test(yf$views,yf$dislikes,method="pearson");

Pearson's product-moment correlation

data:  yf$views and yf$dislikes
t = 124.47, df = 33086, p-value < 2.2e-16
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.5573462 0.5720240
sample estimates:
      cor 
0.5647297
```

Curiosament observem un valor molt diferent quan analitzem la correlació entre les visites i el número de “dislikes”. Tot i que continua sent un valor positiu (a més vistes, més “dislikes”), aquest marca una relació molt menys marcada.

Una possible explicació a aquest fenomen seria que costa més als usuaris penalitzar negativament un video. La reacció més habitual quan un video no ens agrada és veure'n un altre o tancar el navegador i, per tant, aquesta mesura deu tenir molts menys "dislikes" dels que realment són i d'aquí que no augmenti tant positivament en funció del nombre de vistes.

Correlació vistes – quantitat de comentaris

```
> cor.test(yf$views,yf$comment_count,method="pearson");

Pearson's product-moment correlation

data: yf$views and yf$comment_count
t = 172.11, df = 33086, p-value < 2.2e-16
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.6815605 0.6929316
sample estimates:
      cor 
0.6872882
```

Observem un valor positiu prou alt (proper a 1) i, per tant, podem afirmar que existeix relació entre les visites i el número de comentaris.

Podem concloure doncs que per analitzar l'activitat generada en els videos la mesura de les vistes es un indicador fiable del nostre conjunt de dades.

Regressió lineal per predir el nombre de visites

Intentarem extreure més informació sobre el nostre conjunt de dades mitjançant un model de regressió lineal que ens permeti predir el nombre de visites en funció de la categoria o del dia de publicació del video.

L'atribut category_id no aporta informació numérica, és una simple codificació. No tenim informació sobre aquesta codificació i per tant no podem intentar donar-li significat a cada codi. És a dir, si tinguessim, per exemple, el valor d'edat promitja dels usuaris que consulten cada categoria podríem ordenar-les per aquest camp de manera que la mitjana d'edat de la categoria 1 fos inferior que la de la categoria 2 i així successivament. Com no és el cas, aplico el model sobre el camp tot i no esperar obtenir cap model significatiu.

```

> model<-lm(formula = views ~ category_id, data=train);
> summary(model);

Call:
lm(formula = views ~ category_id, data = train)

Residuals:
    Min       1Q   Median       3Q      Max
-2444075  -845159  -634303  -103839  136934621

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  2513212     66555   37.76  <2e-16 ***
category_id  -66863      3049  -21.93  <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3365000 on 26468 degrees of freedom
Multiple R-squared:  0.01784,    Adjusted R-squared:  0.01781
F-statistic: 480.9 on 1 and 26468 DF,  p-value: < 2.2e-16

```

Si ens fixem en el valor de “Multiple R-squared” observem un valor molt baix: 0,01784. Això vol dir que el camp category_id només influeix un 1,7% sobre la predicció de les vistes.

Tal i com esperava no hem aconseguit un bon model de predicció.

Anem a utilitzar el conjunt de test per comprovar-ne la predicció i analitzar els resultats:

```

> prob<-predict(model, test, type="response");
> check<-data.frame(
+   real=test$views,
+   predicted= prob,
+   dif=ifelse(test$views>prob, -prob*100/test$views,prob*100/test$views),
+ );
> colnames(check)<-c("Real","Prediction","Dif%");
> summary(check);

```

Real		Prediction		Dif%	
Min.	: 1508	Min.	:-361899	Min.	: -99.94
1st Qu.:	135922	1st Qu.:	908499	1st Qu.:	111.39
Median :	356116	Median :	908499	Median :	290.37
Mean :	1063501	Mean :	1124071	Mean :	1263.72
3rd Qu.:	928100	3rd Qu.:	1175951	3rd Qu.:	752.46
Max.	:87450245	Max.	:2446349	Max.	:91282.50

Hem calculat la predicció del model sobre el nostre conjunt de test i a partir del valor real i l'esperat hem calculat la diferència en percentatge. Mitjançant estadístics bàsics com pot ser la mitjana i la distribució en quartils observem que la predicció és massa a l'alça. Per exemple el valor promig de la diferència és del 1263%.

Anem a intentar-ho pel camp "trending_date", però abans intentarem donar-ne significat convertint-lo a numeric mitjançant la formula $YY*10000+MM*100+DD$:

```
> train$n_date = as.numeric(substr(train[, "trending_date"], 1, 2)) * 10000 +  
+ as.numeric(substr(train[, "trending_date"], 7, 9)) * 100 +  
+ as.numeric(substr(train[, "trending_date"], 4, 5));  
> test$n_date = as.numeric(substr(test[, "trending_date"], 1, 2)) * 10000 +  
+ as.numeric(substr(test[, "trending_date"], 7, 9)) * 100 +  
+ as.numeric(substr(test[, "trending_date"], 4, 5));
```

Creem el model de regressió lineal a partir d'aquesta nova variable:

```
> model<-lm(formula = views ~ n_date, data=train);  
> summary(model);  
  
Call:  
lm(formula = views ~ n_date, data = train)  
  
Residuals:  
      Min       1Q   Median       3Q      Max   
-1271220  -978310  -758225  -180906 136572940  
  
Coefficients:  
              Estimate Std. Error t value Pr(>|t|)      
(Intercept)  5.120e+06   8.969e+05   5.709 1.15e-08 ***  
n_date       -2.249e+01   5.048e+00  -4.455 8.43e-06 ***  
---  
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 3394000 on 26468 degrees of freedom  
Multiple R-squared:  0.0007492, Adjusted R-squared:  0.0007115  
F-statistic: 19.85 on 1 and 26468 DF, p-value: 8.428e-06
```

Els valors són encara menys bons que pel camp de categoria i només expliquen un 0,07% de les vistes a partir del valor de la data. Aquest valor sí que ens dona però una dada interpretable: podem deduir que el nombre de visites no va en augment, ja que si a major data hi hagués major nombre de visites el model ho reflexaria.

Anem a utilitzar el conjunt de test per comprovar-ne la predicció i analitzar els resultats:

```

> prob<-predict(model, test, type="response");
> check<-data.frame(
+   real=test$views,
+   predicted= prob,
+   dif=ifelse(test$views>prob, -prob*100/test$views,prob*100/test$views,
+   );
> colnames(check)<-c("Real","Prediction","Dif%");
> summary(check);

```

Real	Prediction	Dif%
Min. : 1508	Min. :1061196	Min. : -100.0
1st Qu.: 135922	1st Qu.:1065424	1st Qu.: 119.7
Median : 356116	Median :1067920	Median : 313.9
Mean : 1063501	Mean :1124390	Mean : 1265.9
3rd Qu.: 928100	3rd Qu.:1269888	3rd Qu.: 830.1
Max. :87450245	Max. :1272406	Max. :81078.0

Hem calculat la predicció del model sobre el nostre conjunt de test i a partir del valor real i l'esperat hem calculat la diferència en percentatge. Mitjançant estadístics bàsics com pot ser la mitjana i la distribució en quartils observem que la predicció és massa a l'alça.

Com a últim intent, anem a calcular regressió lineal a partir de la combinació de les dues variables: categoria i data:

```

> model<-lm(formula = views ~ category_id + n_date, data=train)
> summary(model);

```

Call:
lm(formula = views ~ category_id + n_date, data = train)

Residuals:

Min	1Q	Median	3Q	Max
-2588970	-868236	-622666	-96495	136787689

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	6.591e+06	8.914e+05	7.394	1.47e-13 ***
category_id	-6.692e+04	3.048e+03	-21.956	< 2e-16 ***
n_date	-2.295e+01	5.003e+00	-4.587	4.51e-06 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3364000 on 26467 degrees of freedom
Multiple R-squared: 0.01862, Adjusted R-squared: 0.01855
F-statistic: 251.1 on 2 and 26467 DF, p-value: < 2.2e-16

Cap sorpresa en els resultats, la combinació de les dues variables no millora significativament cap dels models anteriors.

Anem a utilitzar el conjunt de test per comprovar-ne la predicció i analitzar els resultats:

```

> prob<-predict(model, test, type="response");
> check<-data.frame(
+   real=test$views,
+   predicted= prob,
+   dif=ifelse(test$views>prob, -prob*100/test$views,prob*100/test$vie
+   );
> colnames(check)<-c("Real","Prediction","Dif%");
> summary(check);

```

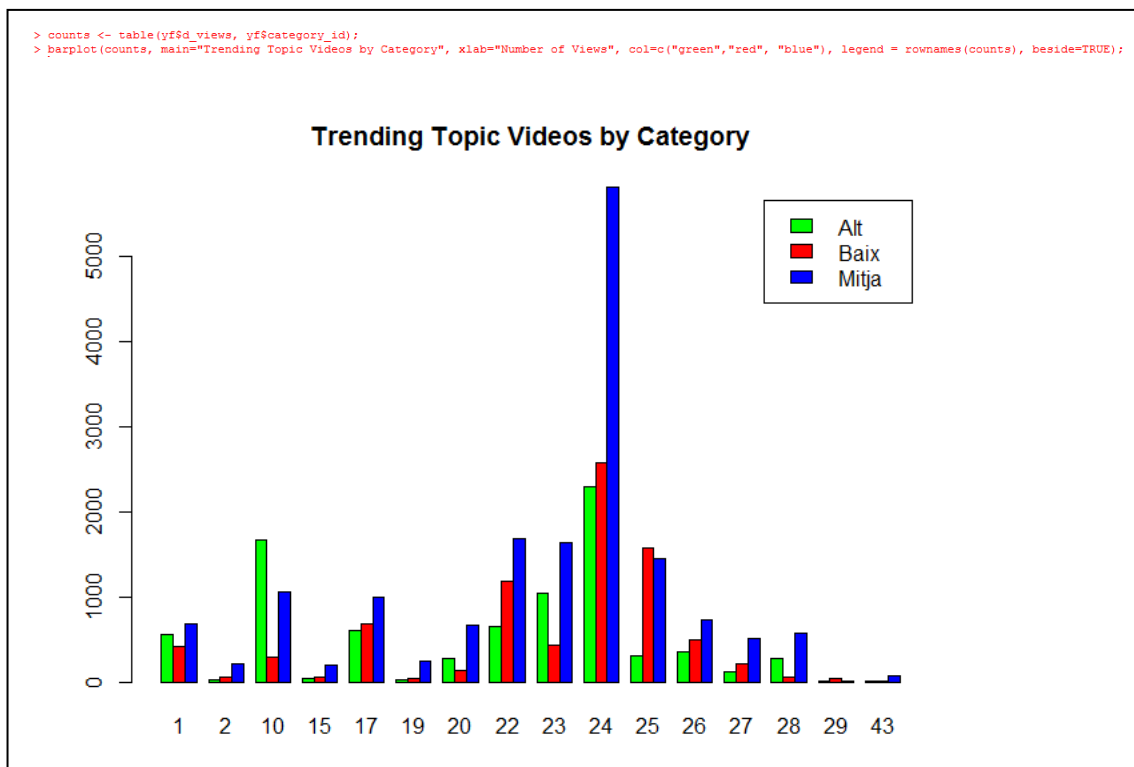
Real		Prediction		Dif%	
Min. :	1508	Min. : -	429359	Min. :	-99.72
1st Qu.:	135922	1st Qu.:	846539	1st Qu.:	111.59
Median :	356116	Median :	980268	Median :	287.73
Mean :	1063501	Mean :	1122409	Mean :	1248.70
3rd Qu.:	928100	3rd Qu.:	1191312	3rd Qu.:	753.07
Max. :	87450245	Max. :	2596905	Max. :	87222.34

Un cop més podem comprovar com el model prediu massa a l'alça.

En els diferents tests de predicció observem que hem tingut un efecte negatiu per no haver tractat els outliers. Una possible opció seria truncar-los tal i com havíem plantejat en l'apartat corresponent i repetir els càlculs estadístics. No ho faré perquè no n'espero obtenir variacions de pes en el resultat dels models obtinguts que variï les interpretacions finals del estudi de les dades.

5. Representació dels resultats a partir de taules i gràfiques

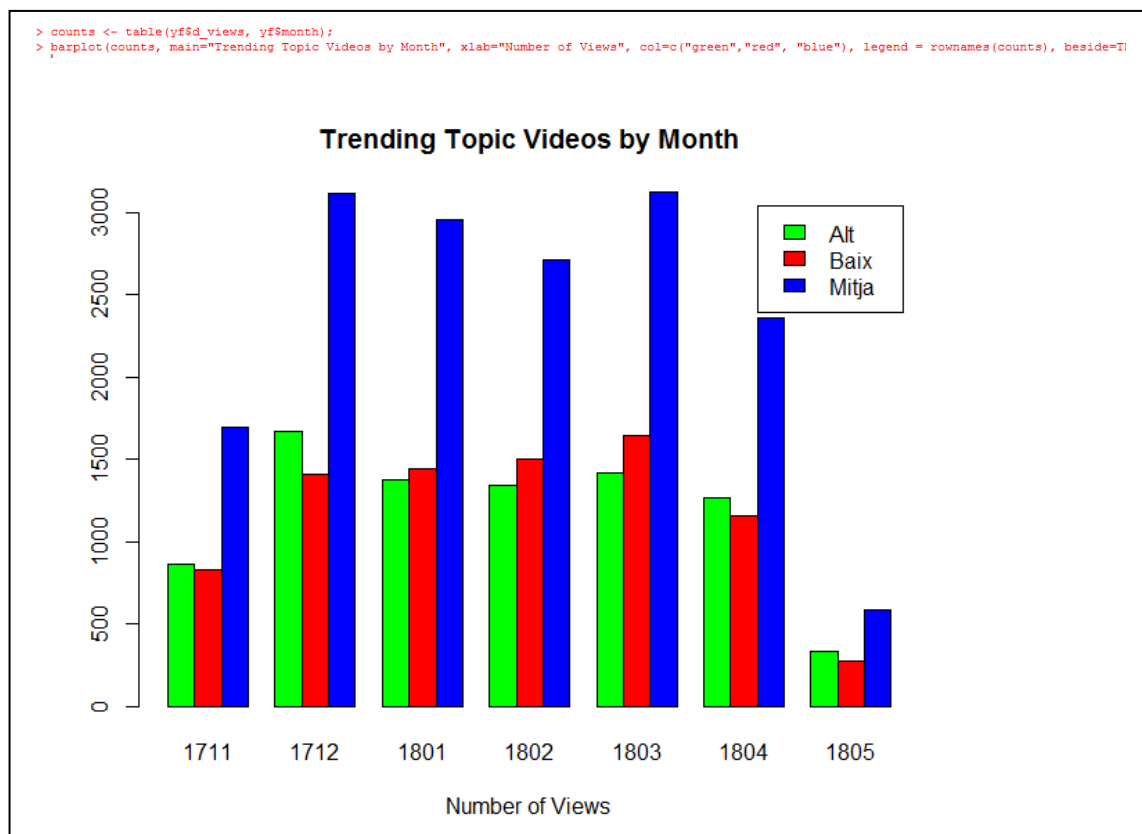
Per analitzar el comportament dels videos trending topic en funció de la categoria utilitzare un barplot:



Recordem que estem fent servir la columna discretitzada “d_views” que ens agrupa els videos segons un numero alt, mitjà o baix de visites. Les categories estan representades per un codi intern present a les dades. La següent taula mostra els noms de les categories presents en el gràfic segons la documentació adjunta amb el dataset:

Id	Nom
1	Film & Animation
2	Autos & Vehicles
10	Music
15	Pets & Animals
17	Sports
19	Travel & Events
20	Gaming
22	People & Blogs
23	Comedy
24	Entertainment
25	News & Politics
26	Howto & Style
27	Education
28	Science & Technology
29	<descripció no disponible>
43	Shows

Anem a analitzar com es comporten el número de visites per mesos:



Recordem que el primer i el darrer mesos estan incomplets. No els he eliminat de la mostra perquè tot i que no es poden comparar a la resta de mesos en valors absoluts ens poden servir per analitzar la distribució en la que els videos es clasifiquen en {Baix, Mitja, Alt}.

6. Resolució del problema. A partir dels resultats obtinguts, quines són les conclusions? Els resultats permeten respondre al problema?

Estem en disposició de donar respostes a les preguntes plantejades.

¿És Youtube una plataforma principalment de música? Si suposem que els videos que arriben a ser trending topic són una representació proporcional del total de videos existents a la xarxa està clar que no. La categoria de música té poca representació en la gràfica.

¿Existeix una relació entre el nombre de visites i el número de likes, dislikes i número de comentaris? Sí, tal i com cabia esperar. Hem descobert que la relació no és tan marcada en quant al número de “dislikes”, fet que sense més dades atribueixo a que ens costa més penalitzar un video amb un “no m’agrada” que no pas deixar-hi una valoració positiva.

¿En quina categoria ens hauríem d’especialitzar si volguéssim dedicar-nos professionalment a la pujada de videos? Aquesta pregunta és més complexa de respondre ja que dependrà de la

nostra estratègia com a youtubers. Si ens basem només en el volum total de vídeos que hi apareixen ens hauríem de decantar per l'entreteniment en concret, o bé en contingut tipus "blog" en general (les categories "People & Blogs", "Comedy", "Entertainment" i "News & Politics" representen la major part de la mostra).

En canvi, si volguéssim apostar per aconseguir vídeos concrets amb un alt nivell de visites hauríem d'arriscar-nos amb la música o l'entreteniment, ja que són les dues categories que més proporció de franja "Alt" tenen de la mostra.

A partir de la distribució de les franges "Alt", "Mitja" i "Baix" intepreto que les categories amb un comportament més normalitzat, és a dir, amb una presència majoritària de franja mitjana i equilibri entre franges alta i baixa, són "Sports", "People & Blogs" i "Entertainment". Crec que això pot ser un indicador de que tenen una presència més sostinguda entre els trending topics a diferència de, per exemple, la categoria de música que al tenir una altíssima proporció de la franja alta i mitjana fa pensar que és deu més a fenòmens virals espontànies.

¿Existeix temporalitat en l'èxit dels vídeos (depen el número de visites del mes en que es puja)?

Per poder respondre a aquesta pregunta caldrien dades de un any complet i només en tenim cinc mesos. D'aquests cinc mesos només notem una baixada en l'abril però considero que no es prou significatiu com per extreure'n conclusions. Si que podem observar que en tots els mesos es manté la mateixa proporció de vídeos en les franges altes, mitjanes i baixes.

7. Codi: Cal adjuntar el codi, preferiblement en R, amb el que s'ha realitzat la neteja, anàlisi i representació de les dades. Si ho preferiu, també podeu treballar en Python

El codi fet servir durant aquesta pràctica està disponible al repository de github creat expressament per la mateixa.

He basat algunes parts del codi en R en la pràctica publicada com a exemple en l'enunciat: <https://github.com/Bengis/nba-gap-cleaning>.