Metodi Informatici per la Gestione Aziendale

Progetto di Sanvito Simone

Sommario

Data acquisition

Dataset

Il dataset che è stato scelto è il seguente:

https://www.kaggle.com/datasnaek/youtube-new?select=CAvideos.csv

Questo dataset è una registrazione giornaliera dei video di YouTube di tendenza in Canada.

Questo set di dati include i dati di diversi mesi di video su YouTube di tendenza quotidiana. I dati sono riferiti ad una regione geografica (CA, Canada), con un massimo di 200 video di tendenza elencati al giorno.

Composizione

La composizione del dataset è la seguente: le righe rappresentano i vari video, mentre come colonne si hanno:

1. id del video: codice univoco che identifica un video;
2. data di tendenza: quando il video è andato in tendenza;
3. il titolo del video;
4. il nome del canale che ha pubblicato quel video;
5. category_id: codice univoco che identifica la categoria a cui appartiene il video in questione (si veda sotto);
6. data e ora di pubblicazione del video;
7. i tags presenti, associati al video;
8. numero di views;
9. numero di likes ricevuti dal video;
10. numero di dislikes ricevuti dal video;
11. numero dei commenti ricevuti dal video;
12. il link associato al video;
13. una variabile booleana che indica se i commenti sono stati disabilitati per quel video o meno;
14. una seconda variabile booleana che indica se il rating è stato disabilitato per quel video o meno;
15. una terza variabile booleana che indica se il video è stato rimosso o presenta qualche errore;
16. la descrizione del video.

Categorie dei video

Le categorie sono le seguenti (con il loro id associato):

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

```
DFvideo <- read.csv("CAvideos.csv")
View(DFvideo)
dim(DFvideo)
str(DFvideo)
head(DFvideo)
tail(DFvideo)
summary(DFvideo)
```

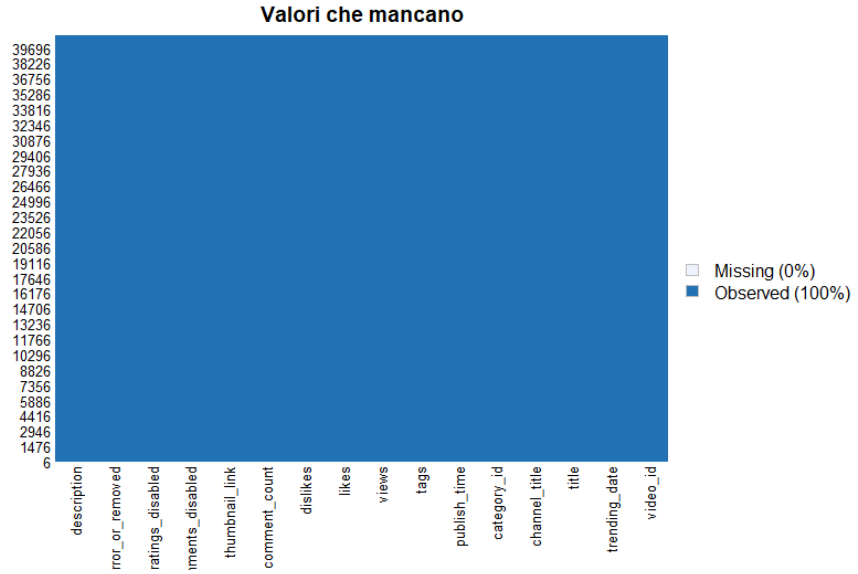
Per prima cosa viene importato il dataset e viene quindi creato il dataframe che verrà utilizzato nel progetto, poi si visualizza lo stesso dataframe e si dà una prima occhiata ad esso. Non si riportano i risultati in quanto sono ancora presenti delle colonne testuali, le quali portano a una cattiva visualizzazione del dataframe. Viene fatta una descrizione generale, per far capire come è costituito il dataset.

Descrizione del dataset:

	vars	n	mean	sd	median	trimmed	mad	min	max	range	skew	kurtosis	se
video_id*	1	40881	12223.62	7040.86	12264	12228.73	9015.69	1	24427	24426	-0.01	-1.20	34.82
trending_date*	2	40881	102.94	59.19	103	102.93	75.61	1	205	204	0.00	-1.20	0.29
title*	3	40881	12308.07	7024.45	12222	12302.53	8876.33	1	24573	24572	0.01	-1.17	34.74
channel_title*	4	40881	2594.89	1478.52	2651	2609.05	1997.06	1	5076	5075	-0.06	-1.27	7.31
category_id	5	40881	20.80	6.78	24	21.97	1.48	1	43	42	-1.58	2.24	0.03
publish_time*	6	40881	11551.39	6805.50	11412	11507.43	8797.75	1	23613	23612	0.04	-1.20	33.66
tags*	7	40881	9561.70	6127.04	9451	9530.48	8060.90	1	20157	20156	0.03	-1.25	30.30
views	8	40881	1147035.91	3390913.02	371204	564872.95	417989.42	733	137843120	137842387	13.49	306.42	16770.88
likes	9	40881	39582.69	132689.53	8780	16182.45	11730.33	0	5053338	5053338	13.66	302.68	656.26
dislikes	10	40881	2009.20	19008.37	303	558.08	378.06	0	1602383	1602383	58.66	4267.74	94.01
comment_count	11	40881	5042.97	21579.02	1301	2143.10	1630.86	0	1114800	1114800	25.15	921.04	106.73
thumbnail_link*	12	40881	12221.18	7039.08	12263	12226.40	9014.21	1	24422	24421	-0.01	-1.20	34.81
comments_disabled*	13	40881	1.01	0.12	1	1.00	0.00	1	2	1	8.19	65.13	0.00
ratings_disabled*	14	40881	1.01	0.08	1	1.00	0.00	1	2	1	11.98	141.53	0.00
video_error_or_removed*	15	40881	1.00	0.03	1	1.00	0.00	1	2	1	38.87	1509.04	0.00
description*	16	40881	10802.79	6664.30	10797	10788.78	8591.67	1	22346	22345	0.00	-1.21	32.96

Cleaning dataset

Poi si passa alla parte di pulizia del dataset: viene fatta la missmap, per individuare i valori mancanti (non ce ne sono).



In seguito, vengono eliminati, in primis, gli eventuali valori mancanti (che non sono presenti in questo caso), gli Na o Nan (tramite la na.omit), poi alcune colonne testuali, in quanto non servono per l'analisi da fare successivamente (tags, descrizione del video e link associato al video). Infine, vengono tolti i video con 0 visualizzazioni e i video che valgono true nella colonna "video error or removed" in quanto non sono considerabili nell'analisi.
I video che nella colonna "ratings disabled" hanno valore true, nonostante abbiano 0 likes e 0 dislikes, vengono tenuti in considerazione perché hanno comunque registrati sia il numero di views che il numero di commenti (che verranno utilizzati nelle analisi successive).

Il dataset così ottenuto è così composto:

```
> summary(DFvideo)
        video_id       trending_date
 6ZfuNTqbHE8:    8   17.01.12:  200
 l_lblj8Cq0o:    8   17.02.12:  200
 UceaB4D0jpo:    8   17.03.12:  200
 VYOjWnS4cMY:    8   17.06.12:  200
 7X_WvGAhMlQ:    7   17.08.12:  200
 9v_rtaye2yY:    7   17.09.12:  200
 (Other)    :40808   (Other) :39654
                                                                                                      title
 Drake - God's Plan (Official Audio)                                                                 :   15
 Most Popular Violin Covers of Popular Songs 2018 || Best Instrumental Violin Covers 2018            :   15
 Bruno Mars,Charlie Puth,Ed Sheeran Best Christmas Songs,Greatest Hits Pop Playlist Christmas 2018:   13
 Merry Christmas 2018 - Top Christmas Songs Playlist 2018 - Best Christmas Songs Ever                :   10
 Maroon 5 - Wait                                                                                     :    9
 Mission: Impossible - Fallout (2018) - Official Trailer - Paramount Pictures                        :    9
 (Other)                                                                                             :40783
          channel_title     category_id                      publish_time        views
 SET India       :  191   Min.   : 1.0   2017-12-20T23:00:00.000Z:   11   Min.   :      733
 MSNBC           :  189   1st Qu.:20.0   2017-11-18T17:00:00.000Z:   10   1st Qu.:   143894
 FBE             :  188   Median :24.0   2018-01-29T04:00:00.000Z:   10   Median :   371176
 The Young Turks:  186   Mean   :20.8   2018-02-11T15:00:01.000Z:   10   Mean   :  1146868
 REACT           :  183   3rd Qu.:24.0   2018-03-11T16:00:00.000Z:   10   3rd Qu.:   963184
 CNN             :  182   Max.   :43.0   2017-12-22T05:00:00.000Z:    9   Max.   :137843120
 (Other)         :39735                  (Other)                 :40794
     likes             dislikes         comment_count      comments_disabled ratings_disabled
 Min.   :      0   Min.   :      0   Min.   :      0   False:40271       False:40575
 1st Qu.:   2192   1st Qu.:     99   1st Qu.:    417   True :  583       True :  279
 Median :   8780   Median :    303   Median :   1301
 Mean   :  39576   Mean   :   2009   Mean   :   5041
 3rd Qu.:  28704   3rd Qu.:    950   3rd Qu.:   3711
 Max.   :5053338   Max.   :1602383   Max.   :1114800

 video_error_or_removed
 False:40854
 True :    0
```

Ora si stampano la prima parte del dataset (head) e l'ultima (tail).

```
> head(DFvideo)
     video_id trending_date                                                                     title
1 n1WpP7iowLc      17.14.11                                  Eminem - Walk On Water (Audio) ft. BeyoncÃ©
2 0dBIkQ4Mz1M      17.14.11                                               PLUSH - Bad Unboxing Fan Mail
3 5qpjK5DgCt4      17.14.11                          Racist Superman | Rudy Mancuso, King Bach & Lele Pons
4 d380meD0W0M      17.14.11                                                   I Dare You: GOING BALD!?
5 2Vv-BfVoq4g      17.14.11                                   Ed Sheeran - Perfect (Official Music Video)
6 0yIWz1XEeyc      17.14.11 Jake Paul Says Alissa Violet CHEATED with LOGAN PAUL! #DramaAlert Team 10 vs  Martinez Twins!
  channel_title category_id             publish_time     views   likes dislikes comment_count comments_disabled
1    EminemVEVO          10 2017-11-10T17:00:03.000Z 17158579  787425    43420        125882             False
2      iDubbbzTV          23 2017-11-13T17:00:00.000Z  1014651  127794     1688         13030             False
3  Rudy Mancuso          23 2017-11-12T19:05:24.000Z  3191434  146035     5339          8181             False
4       nigahiga          24 2017-11-12T18:01:41.000Z  2095828  132239     1989         17518             False
5    Ed Sheeran          10 2017-11-09T11:04:14.000Z 33523622 1634130    21082         85067             False
6    DramaAlert          25 2017-11-13T07:37:51.000Z  1309699  103755     4613         12143             False
  ratings_disabled video_error_or_removed
1            False                  False
2            False                  False
3            False                  False
4            False                  False
5            False                  False
6            False                  False
```

```
> tail(DFvideo)
         video_id trending_date                                                                                title
40876 7E1np354AEc      18.14.06 Ð'ÐµÑ‡ÐµÑ€ Ñ\u0081 Ð'Ð»Ð°Ð´Ð¸Ð¼Ð¸Ñ€Ð¾Ð¼ Ð¡Ð¾Ð»Ð¾Ð²ÑŒÐµÐ²Ñ‹Ð¼ Ð¾Ñ, 13.06.2018
40877 sGolxsMSGfQ      18.14.06                                                             HOW2: How to Solve a Mystery
40878 8HNuRNi8t70      18.14.06                                                         Eli Lik Lik Episode 13 Partie 01
40879 GWlKEM3m2EE      18.14.06                                   KINGDOM HEARTS III â€“ SQUARE ENIX E3 SHOWCASE 2018 Trailer
40880 lbMKLzQ4cNQ      18.14.06                                                         Trump Advisor Grovels To Trudeau
40881 POTgw38-m58      18.14.06 ã€\u0090å®Œæ•´ç‰ˆã€'é\u0081‡å˜°æ\u0081\u0090æ€–æƒ…äººèØ²æ€Žé°¼è¾¦ï¼Ÿ2018.06.13å°\u008fæ˜Žæ˜Ÿå¤§è·Ÿç\u008f
                channel_title category_id             publish_time  views likes dislikes comment_count comments_disabled
40876 Ð Ð¾Ñ\u0081Ñ\u0081Ð¸Ñ\u008f 24          24 2018-06-13T23:53:29.000Z 201847  1568      407           537             False
40877          Annoying Orange          24 2018-06-13T18:00:07.000Z  80685  1701       99          1312             False
40878         Elhiwar Ettounsi          24 2018-06-13T19:01:18.000Z 103339   460       66            51             False
40879           Kingdom Hearts          20 2018-06-11T17:30:53.000Z 773347 25900      224          3881             False
40880          The Young Turks          25 2018-06-13T04:00:05.000Z 115225  2115      182          1672             False
40881 æˆ'æ„›å°\u008fæ˜Žæ˜Ÿå¤§è·Ÿç\u008f          24 2018-06-13T16:00:03.000Z 107392   300       62           251             False
      ratings_disabled video_error_or_removed
40876            False                  False
40877            False                  False
40878            False                  False
40879            False                  False
40880            False                  False
40881            False                  False
```

Questo dataset è composto da 40854 righe e 13 colonne.

Exploratory Analysis

Statistiche descrittive

Media: media aritmetica di una collezione di dati.
Varianza: varianza di una collezione di dati, esprime di quanto oscilla il valore rispetto alla magnitudo della collezione di dati.
Deviazione standard: deviazione standard di una collezione di dati (come la varianza, ma in un intervallo più stretto).
Skewness: misura la simmetria di una distribuzione attorno alla sua media (= 0 simmetrica, distribuzione normale; > 0 coda destra più lunga della distribuzione normale; <0 coda sinistra più lunga della distribuzione normale).
Kurtosis: misura lo spessore della coda di distribuzione (> 0 coda più larga della coda di distribuzione normale; <0 coda più sottile della coda di distribuzione normale; = 0 coda come coda di distribuzione normale)
Quantili: sono punti che dividono un insieme di osservazioni in gruppi di uguale dimensione.

Ora si calcolano i valori delle statistiche descrittive per gli indici numerici del dataset, ovvero: numero di views, numero di likes, numero di dislikes e numero di commenti per i video.

```
views_kurt                309.396139795523
views_mean                1146867.81233172
views_q                   Named num [1:5] 7.33e+02 1.44e+05 3.71e+05 9.63e+05 1.38e+08
views_sd                  3391576.34689375
views_skew                13.4873831258535
views_var                 11502790116809.2
```

```
summary(DFvideo$views)
   Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
    733    143894    371176   1146868    963184 137843120
```

```
likes_kurt                305.662394232955
likes_mean                39576.0255054585
likes_q                   Named num [1:5] 0 2192 8780 28705 5053338
likes_sd                  132714.93612034
likes_skew                13.6576851991423
likes_var                 17613254269.4258
```

```
summary(DFvideo$likes)
 Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
    0    2192    8780   39576   28704 5053338
```

```
dis_q                     Named num [1:5] 0 99 303 950 1602383
dislikes_kurt             4268.491546163
dislikes_mean             2008.88461350174
dislikes_sd               19014.2532894469
dislikes_skew             58.6508598450986
dislikes_var              361541828.155241
```

```
summary(DFvideo$dislikes)
 Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
    0      99     303    2009     950 1602383
```

com_q	Named num [1:5] 0 417 1301 3711 1114800
comment_kurt	923.917152635206
comment_mean	5041.24550839575
comment_sd	21583.6060610348
comment_skew	25.15627316703
comment_var	465852050.597938

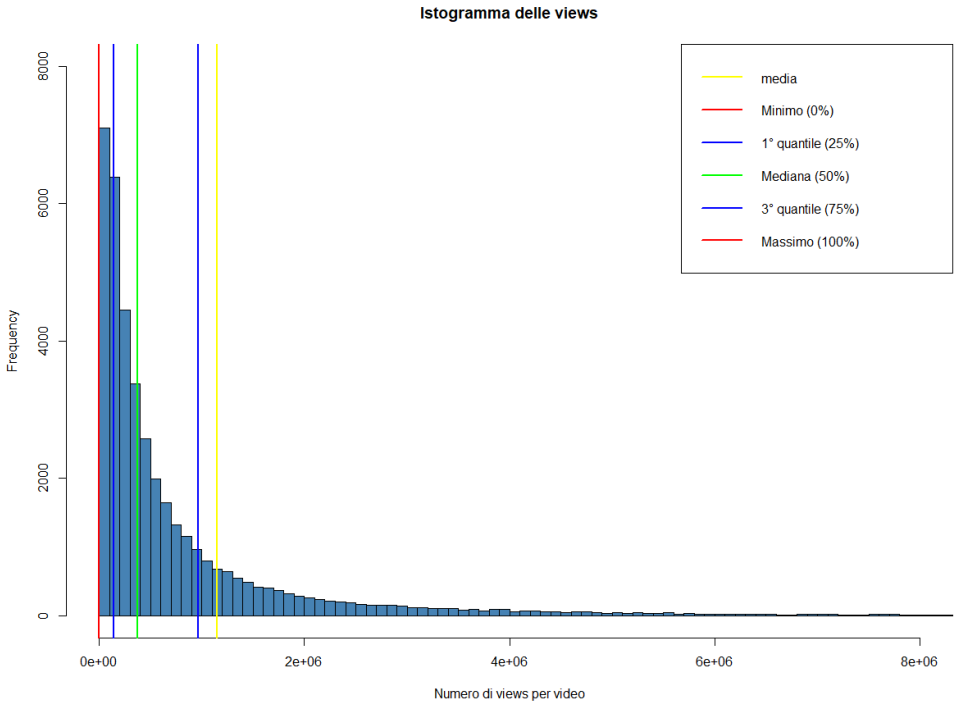
```
summary(DFvideo$comment_count)
 Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
    0     417    1301    5041    3711 1114800
```

Si vede che per tutti gli indici considerati i valori sia di skewness che di kurtosis sono maggiori di 0, dunque tutte le distribuzioni dei vari indici avranno coda destra più lunga della distribuzione normale e anche coda più larga della coda di distribuzione normale.

Istogrammi

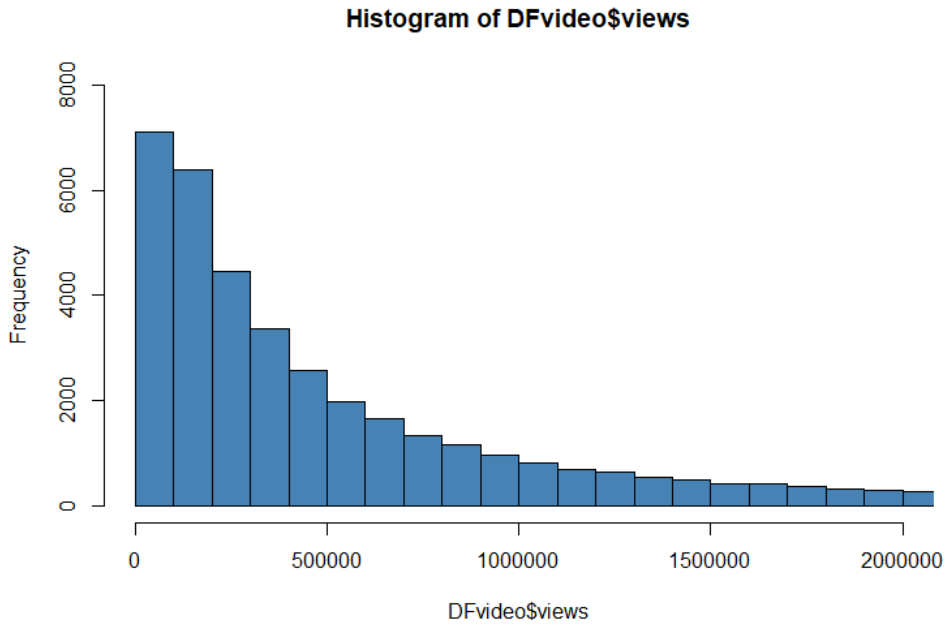
A questo punto si vanno ad analizzare gli istogrammi creati dai valori di numero di views, numero di likes, numero di dislikes e numero di commenti per i video.

Istogramma delle views:

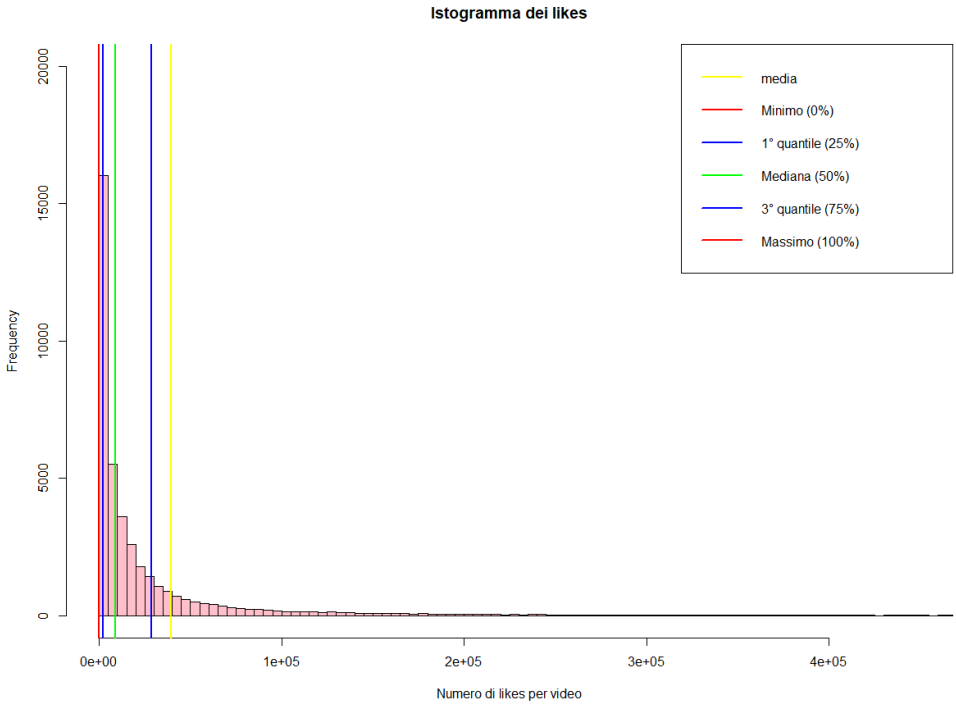


Da questo grafico si nota come la maggioranza del numero di views è distribuito tra 0 e 2mln, il che può essere sensato, perché non sono "all'ordine del giorno" i video con più di 2mln di views. Nel grafico sono state disegnate anche le rette verticali che, come scritto nella legenda del grafico, rappresentano i quantili e la media del numero di visualizzazioni. Il massimo (4° quartile) sta fuori dal grafico, come si vede dal quantile vale 137843120, quindi in questo intervallo di valori per l'asse delle ascisse il massimo è al di fuori del grafico.

Si fa uno zoom per i valori da 0 a 2mln di visualizzazioni per capire meglio la distribuzione, visto che si vede che i valori sono maggiormente concentrati in quell'intervallo: si vede che il numero delle views è maggiormente concentrato tra 0 e 500 mila, con la maggior parte compresa nell'intervallo (0, 100000).



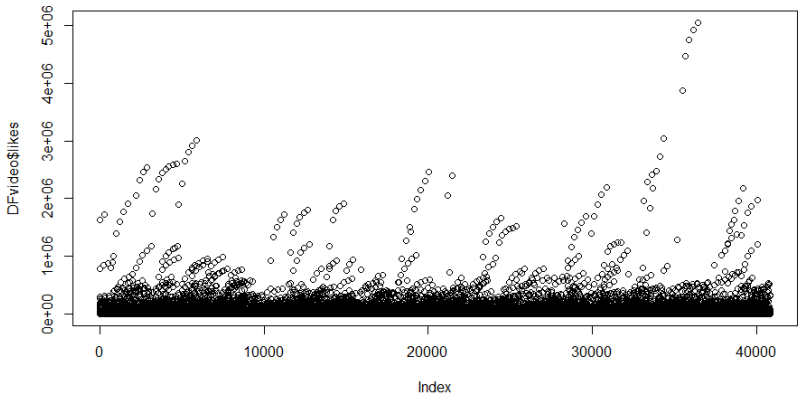
Istogramma dei likes:



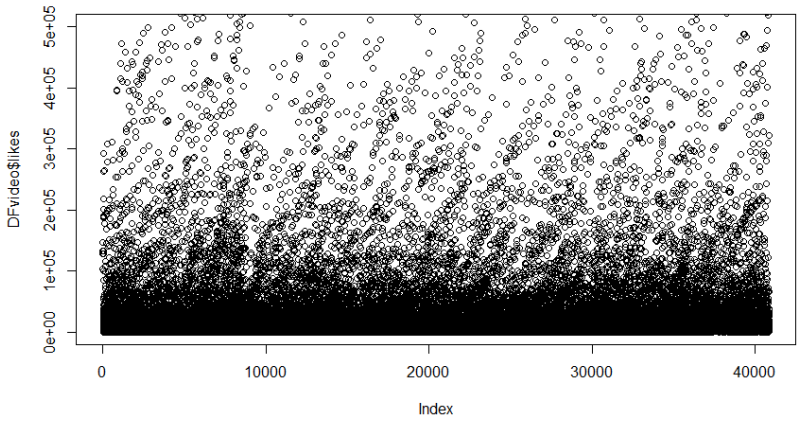
Da questo grafico si nota come la maggioranza del numero di likes è distribuito tra 0 e 100 mila. Anche in questo grafico sono state disegnate le rette verticali che, come scritto nella legenda del grafico, rappresentano i quantili e la media del numero di visualizzazioni. Il

massimo (4° quartile) sta fuori dal grafico e, come si vede dal valore del quantile, vale 5053338, quindi in questo intervallo di valori per l'asse delle ascisse il massimo è al di fuori del grafico.

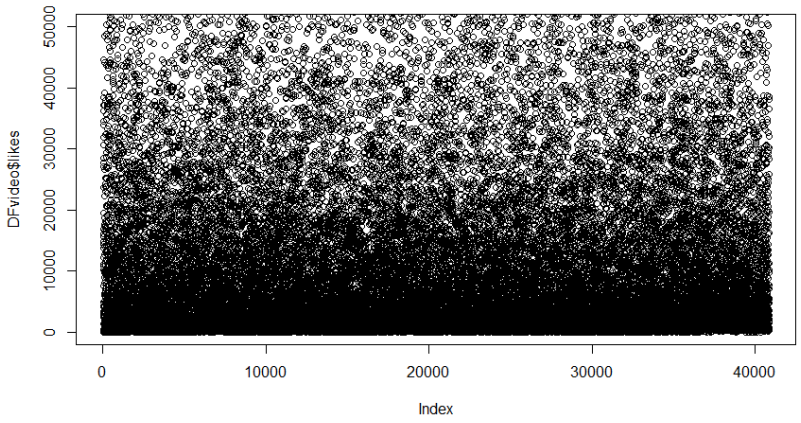
Ora si fa un grafico cercando di capire meglio la distribuzione del numero di likes per video:



Si ottiene che il numero di likes, come ben visibile già nell'istogramma, è denso, molto concentrato tra 0 e mezzo milione; per questo si fa lo zoom su valori compresi tra 0 e 500 mila.

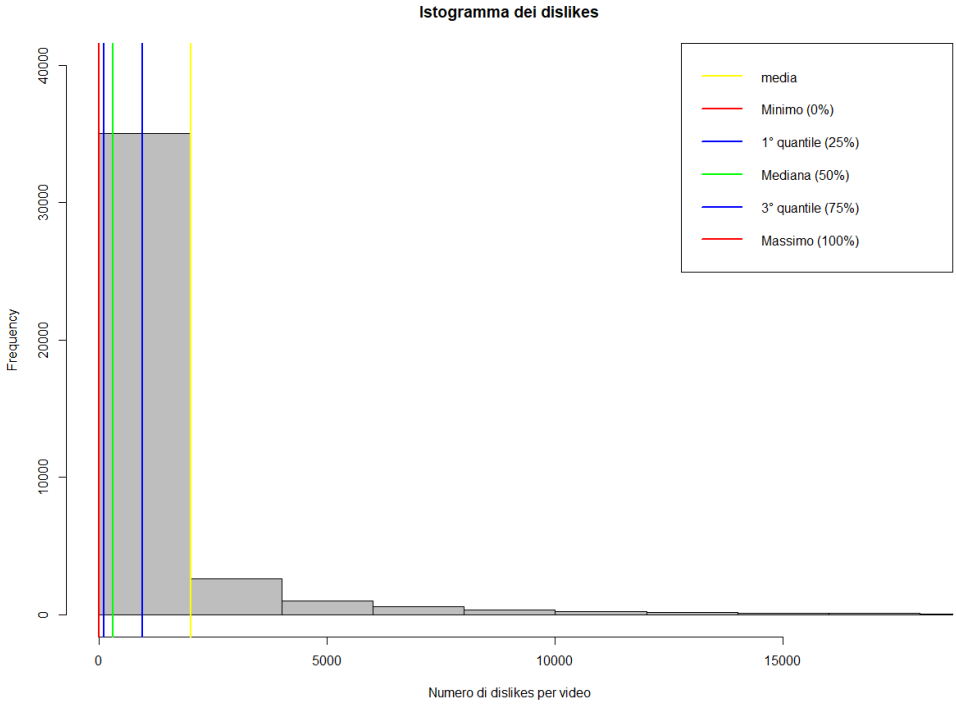


Si ottiene quanto visibile qui sopra; si può vedere ancora che i valori sono concentrati tra 0 e 50 mila circa, dunque si procede con lo zoom tra 0 e 50 mila.



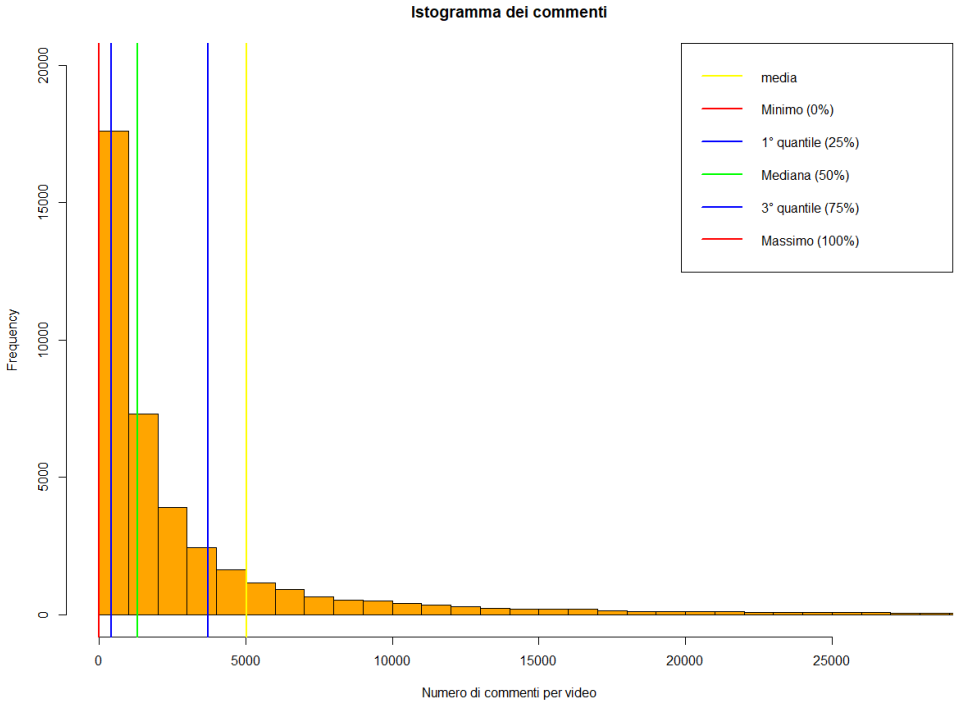
Si vede che in questo intervallo ci sono molti valori, il grafico è nettamente riempito, ci sono sostanzialmente tutti i valori; si nota, ancora una volta, che la maggior concentrazione dei valori è nell'intervallo di valori con le ordinate più prossime allo zero, in particolare c'è una grande concentrazione di valori tra 0 e 10 mila.

Istogramma dei dislikes:



Anche in questo grafico si possono fare sostanzialmente le stesse osservazioni fatte in precedenza: il massimo quartile è al di fuori del grafico, dato che il suo valore (1602383) non sta nell'intervallo mappato sull'asse delle ascisse. Si vede che il numero di dislikes è praticamente distribuito completamente tra 0 e 4000, in particolare si vede bene che la maggior parte dei valori stanno tra 0 e 2000.

Istogramma dei commenti:



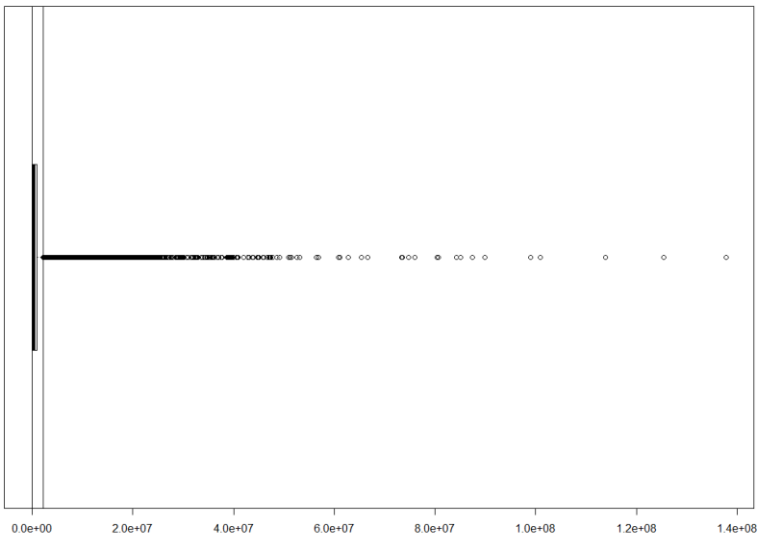
Infine, per quanto riguarda il numero dei commenti, il quartile massimo vale 1114800 che, ancora una volta, sta fuori dall'intervallo considerato sull'asse delle x quindi non è rappresentato nell'istogramma; anche in questo caso si nota che c'è un'ampia densità nei valori più "bassi", si può dire che la maggior parte dei valori siano concentrati tra il valore minimo e il 3° quartile.

Boxplot

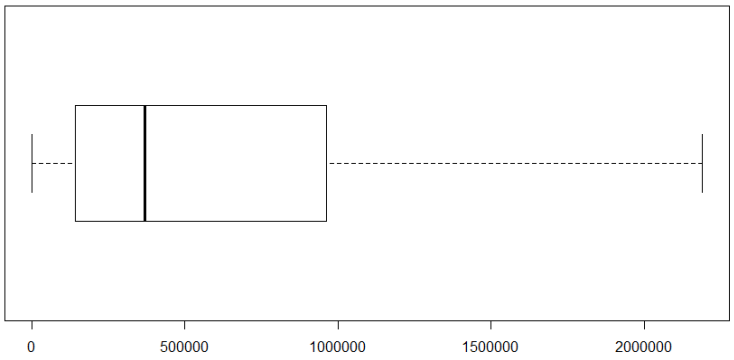
In seguito, si passa a considerare i boxplot relativi a questi "indici".

Nel boxplot viene rappresentata una box, all'interno della quale vi è un segmento che indica la mediana dei valori; inoltre vi è un range, delimitato dai segmenti verticali, all'interno del quale è contenuta una parte dei valori (massimo e minimo della distribuzione) posti alla fine del segmento orizzontale tratteggiato. I punti che non sono all'interno di questo intervallo (delimitato dal segmento tratteggiato) sono chiamati outlier, ovvero dei punti che si discostano fortemente dai valori visti nella distribuzione. Se la mediana si trova sulla linea centrale, allora si avrà una distribuzione normale.

Boxplot per il numero di views:



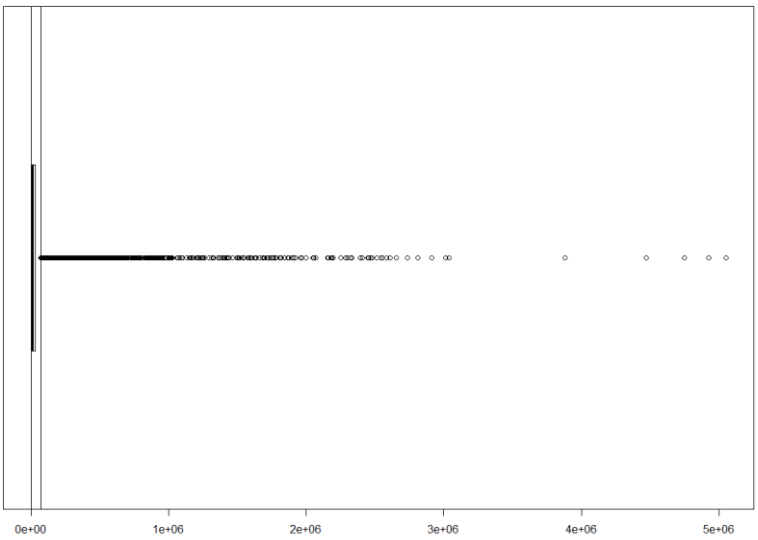
Da questo grafico si può capire poco perché il dataset ha molti valori, quindi ha anche diversi outlier. Le linee verticali all'interno del grafico indicano la grandezza della box, per capire meglio si procede con uno zoom che mostra il boxplot senza gli outlier.



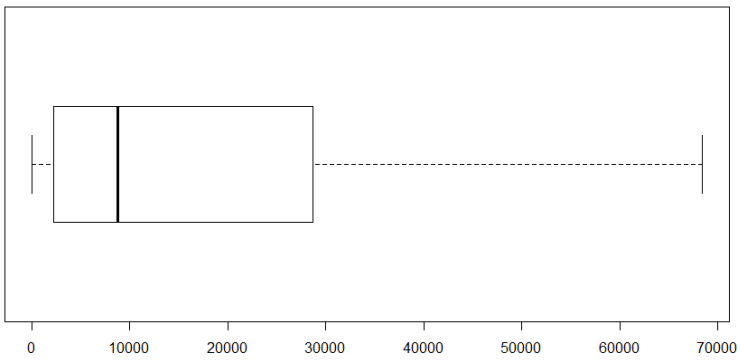
Da questo grafico si può capire che, per i valori superiori a 2000000, i punti rappresentano degli outlier; si vede che la mediana ha un valore molto alto, non al centro della box, quindi la distribuzione non sarà normale, ma la coda di destra sarà più allungata di quella di sinistra.

Boxplot per il numero di likes:

Anche in questo caso, come in quello precedente, i valori sono molti, quindi si deve effettuare uno zoom per capire meglio il boxplot (le rette verticali rappresentano ancora una volta l'ampiezza della box, che viene vista in dettaglio nel grafico successivo).

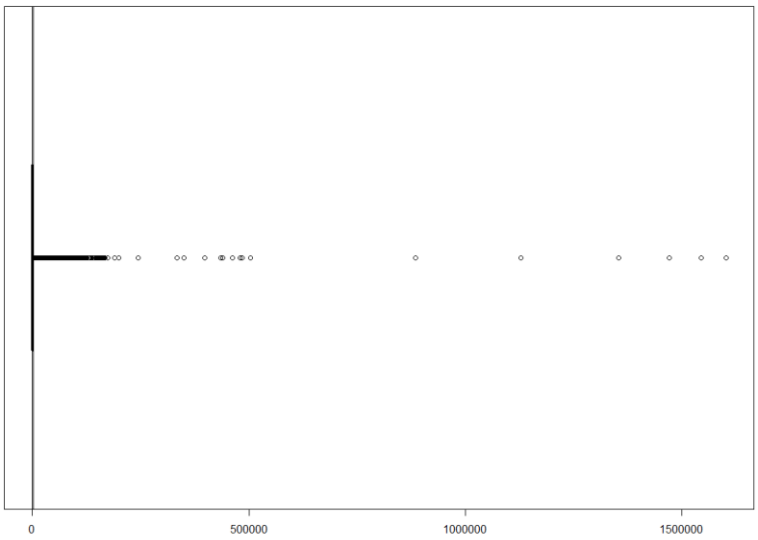


In questo grafico viene fatto lo zoom escludendo gli outlier; si può affermare che gli outlier in questo caso saranno tutti i pallini che rappresentano punti con valori maggiori di 70 mila; anche in questo caso la mediana ha un valore spostato dal centro della box, quindi la distribuzione non sarà normale.

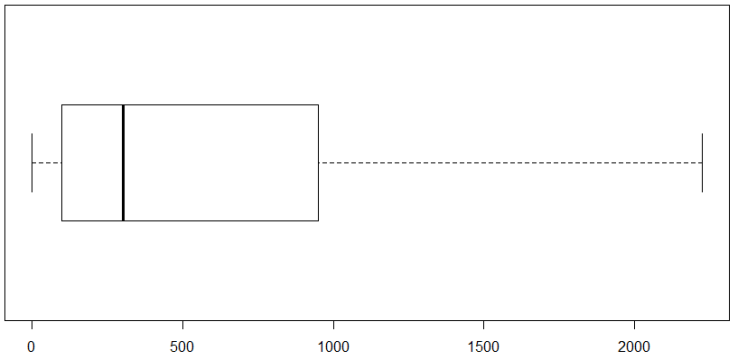


Boxplot per il numero di dislikes:

La situazione descritta precedentemente si ripropone anche in questo caso: l'ampiezza della box in questo caso è molto ridotta poiché ci sono outlier molto grandi che fanno diminuire la dimensione del grafico e quindi anche quella della box.

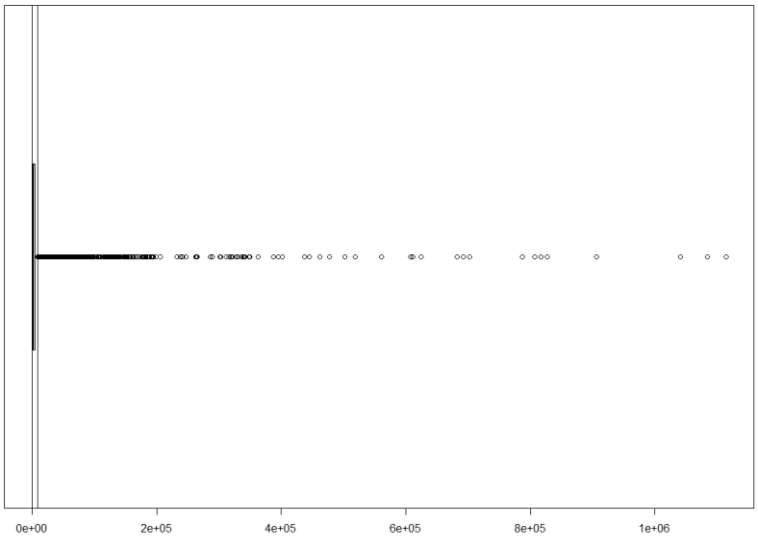


Come fatto in precedenza si procede con uno zoom: gli outlier saranno tutti i valori che superano 2000, la mediana è più vicina allo zero rispetto ai casi precedenti, ma è comunque lontana dal centro della box, quindi la distribuzione non sarà normale.

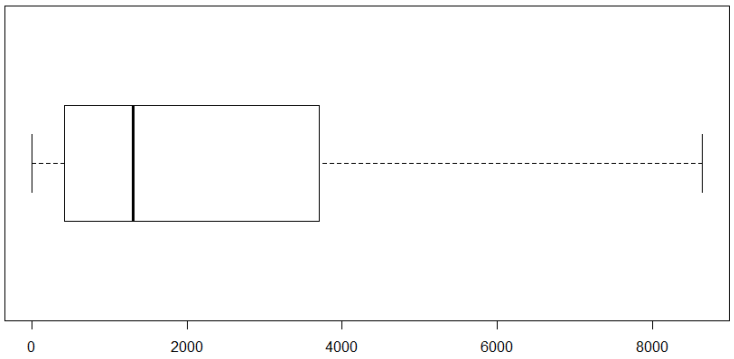


Boxplot per il numero di commenti:

Anche in questo caso si può vedere approssimativamente il boxplot, con la box sempre contraddistinta dalle due rette verticali e con un gran numero di outlier.



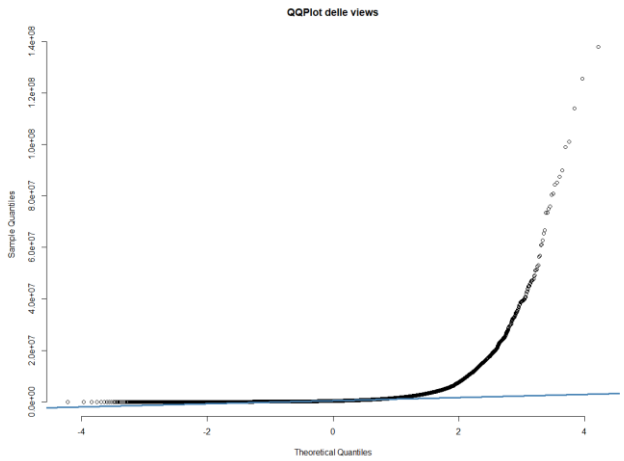
Facendo lo zoom si nota che i valori saranno distribuiti tra 0 e 8000, se superiori a 8000 saranno degli outlier. La mediana vale circa 1000, quindi anche questa volta la distribuzione non sarà normale.



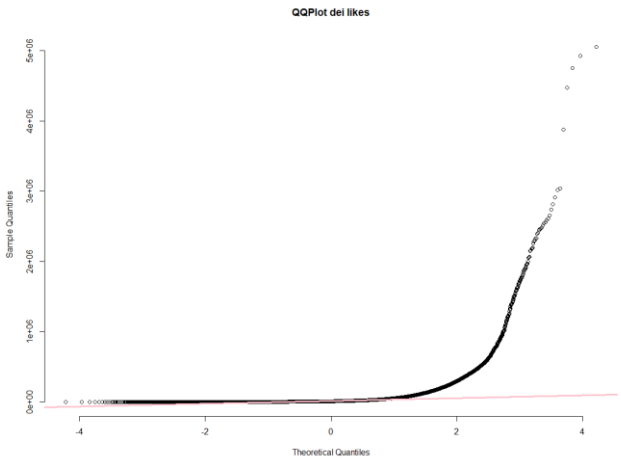
QQPlot

Nel QQplot si può vedere che vi sono diversi punti e una retta: se tutti i punti fossero su quella retta, allora avremmo che la variabile presa in considerazione ha una distribuzione normale.

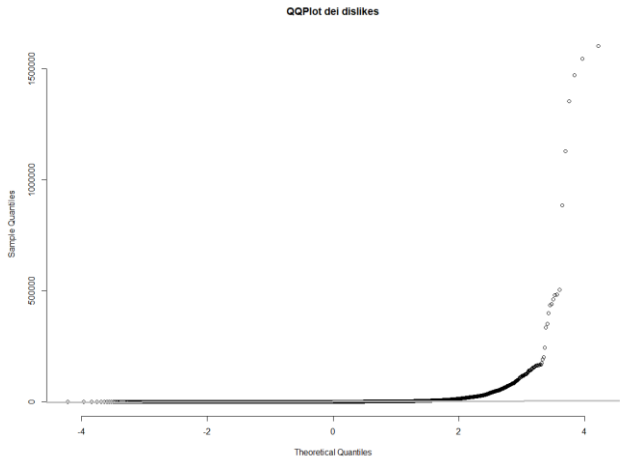
Views



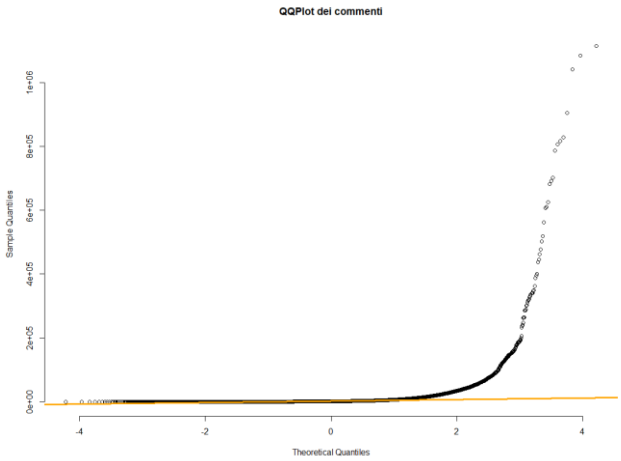
Likes



Dislikes



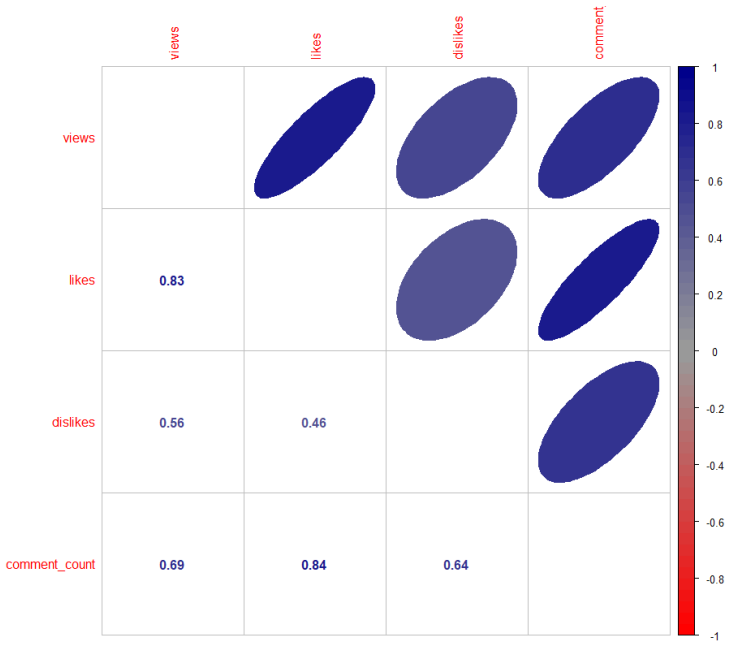
Commenti



Dai grafici qui sopra si nota che in quasi tutti i grafici, i punti stanno in parte sulla qqline, ma, soprattutto alla fine, si discostano molto dalla linea, quindi si avrà che nessuno degli indici considerati ha una distribuzione normale (e si avranno molti outlier per ogni indice).

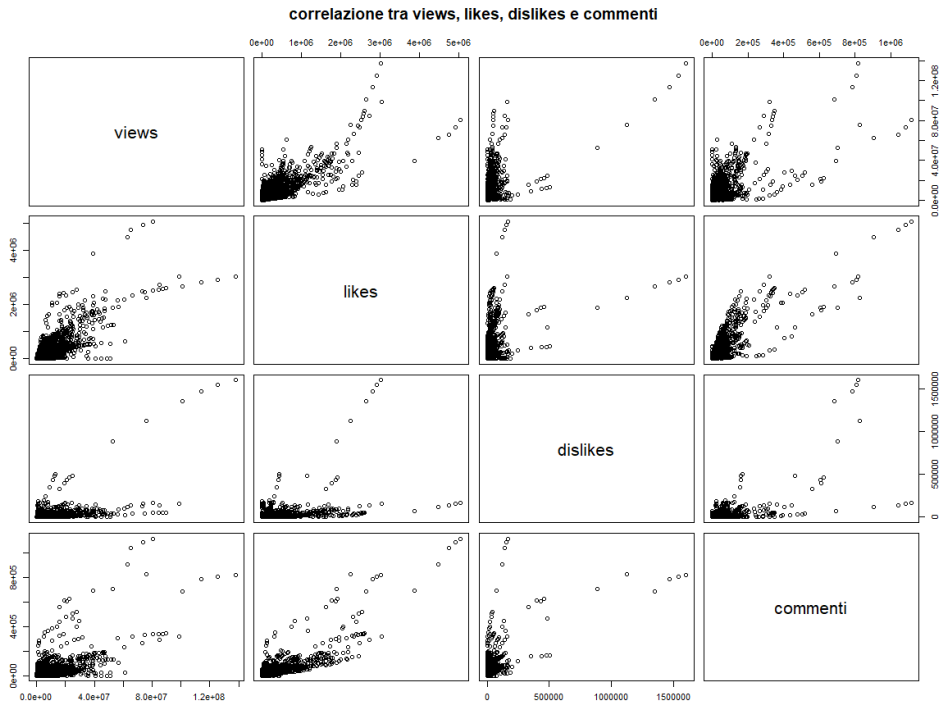
Correlazione

Ora si va ad analizzare la correlazione tra i 4 indici presi in considerazione (n°views, likes, dislikes e commenti).



Le correlazioni che hanno indice di correlazione vicine a zero sono plottate come un cerchio.

Magnitudini lontane dallo zero producono ellissi che sono sempre più strette, blu per la correlazione positiva e rosse per quella negativa.
In questo grafico si osserva che i due indici più correlati sono numero di likes e numero di commenti, con indice di correlazione di Pearson pari a 0.84; segue la correlazione pari a 0.83 tra numero di likes e numero di views, gli altri valori sono visibili nel grafico.



Successivamente è stato fatto uno scatterplot per visualizzare in un'altra maniera la correlazione tra gli indici descritti prima. In questo grafico si sa che, nel caso in cui i punti (delle variabili che si stanno considerando) si trovano sulla bisettrice del quadrante considerato, allora le variabili saranno correlate con un coefficiente di correlazione alto (1 al massimo).

In questo grafico si può confermare quanto detto in precedenza, ovvero che i due indici più correlati sono le coppie numero di likes-commenti e numero di likes-views, infatti nei quadranti corrispondenti si vede come ci sia una buona densità di punti sulla bisettrice del quadrante considerato. Nel caso peggiore (numero di likes-dislikes, coefficiente pari a 0.46) si vede che i punti sulla bisettrice dei quadranti sono pochi, quindi la correlazione c'è, ma ha un valore basso.

Considerazioni più generiche

Video con più views

	video_id	trending_date	channel_title	category_id	publish_time	views	likes	dislikes	comment_count
5901	FlsCjmMhFmw	17.13.12	YouTube Spotlight	24	2017-12-06T17:58:51.000Z	137843120	3014479	1602383	817582
5624	FlsCjmMhFmw	17.12.12	YouTube Spotlight	24	2017-12-06T17:58:51.000Z	125431369	2912715	1545018	807558
5399	FlsCjmMhFmw	17.11.12	YouTube Spotlight	24	2017-12-06T17:58:51.000Z	113876217	2811217	1470387	787174
5198	FlsCjmMhFmw	17.10.12	YouTube Spotlight	24	2017-12-06T17:58:51.000Z	100911567	2656678	1353655	682890
34362	VYOjWnS4cMY	18.13.05	ChildishGambinoVEVO	10	2018-05-06T04:00:07.000Z	98938809	3037318	161813	319502
4700	6ZfuNTqbHE8	17.07.12	Marvel Entertainment	24	2017-11-29T13:26:24.000Z	89930713	2606665	53011	347982
4452	6ZfuNTqbHE8	17.06.12	Marvel Entertainment	24	2017-11-29T13:26:24.000Z	87450245	2584675	52176	341571
34132	VYOjWnS4cMY	18.12.05	ChildishGambinoVEVO	10	2018-05-06T04:00:07.000Z	85092067	2735961	140711	289682
4203	6ZfuNTqbHE8	17.05.12	Marvel Entertainment	24	2017-11-29T13:26:24.000Z	84281319	2555414	51008	339708
36454	7C2z4GqqS5E	18.23.05	ibighit	10	2018-05-18T09:00:02.000Z	80738011	5053338	165854	1114800

Ora si considerano i video che hanno registrato il maggior numero di visualizzazioni:

1. YouTube Rewind: The Shape of 2017 | #YouTubeRewind
2. Childish Gambino - This Is America (Official Video)
3. Marvel Studios' Avengers: Infinity War Official Trailer
4. BTS (방탄소년단) 'FAKE LOVE' Official MV

I video sono ripetuti perché possono essere andati in tendenza per più giorni, quindi c'è un'occorrenza per ogni giorno in tendenza.

Video con più likes

video_id	trending_date	channel_title	category_id	publish_time	views	likes	dislikes	comment_count
7C2z4GqqS5E	18.23.05	ibighit	10	2018-05-18T09:00:02.000Z	80738011	5053338	165854	1114800
7C2z4GqqS5E	18.22.05	ibighit	10	2018-05-18T09:00:02.000Z	73463137	4924056	156026	1084435
7C2z4GqqS5E	18.21.05	ibighit	10	2018-05-18T09:00:02.000Z	65396157	4750254	141966	1040912
7C2z4GqqS5E	18.20.05	ibighit	10	2018-05-18T09:00:02.000Z	62796390	4470888	119046	905912
7C2z4GqqS5E	18.19.05	ibighit	10	2018-05-18T09:00:02.000Z	39349927	3880074	72707	692311
VYOjWnS4cMY	18.13.05	ChildishGambinoVEVO	10	2018-05-06T04:00:07.000Z	98938809	3037318	161813	319502
FlsCjmMhFmw	17.13.12	YouTube Spotlight	24	2017-12-06T17:58:51.000Z	137843120	3014479	1602383	817582
FlsCjmMhFmw	17.12.12	YouTube Spotlight	24	2017-12-06T17:58:51.000Z	125431369	2912715	1545018	807558
FlsCjmMhFmw	17.11.12	YouTube Spotlight	24	2017-12-06T17:58:51.000Z	113876217	2811217	1470387	787174
VYOjWnS4cMY	18.12.05	ChildishGambinoVEVO	10	2018-05-06T04:00:07.000Z	85092067	2735961	140711	289682

Per quanto riguarda i video con più likes, si è ottenuto che i video che più sono piaciuti sono stati i seguenti:

1. BTS (방탄소년단) 'FAKE LOVE' Official MV
2. Childish Gambino - This Is America (Official Video)
3. YouTube Rewind: The Shape of 2017 | #YouTubeRewind

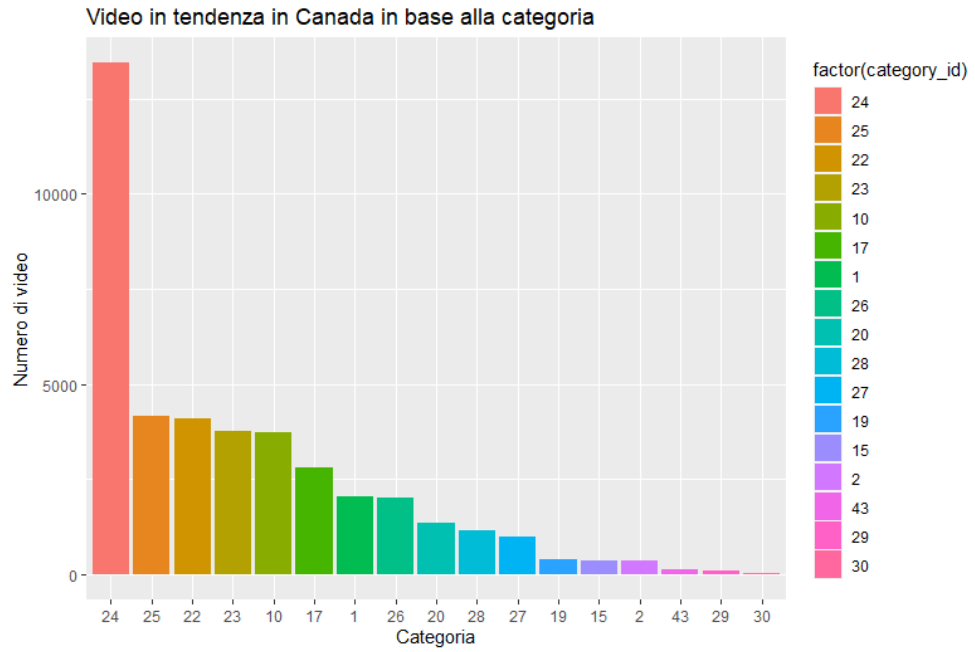
Video con più dislikes

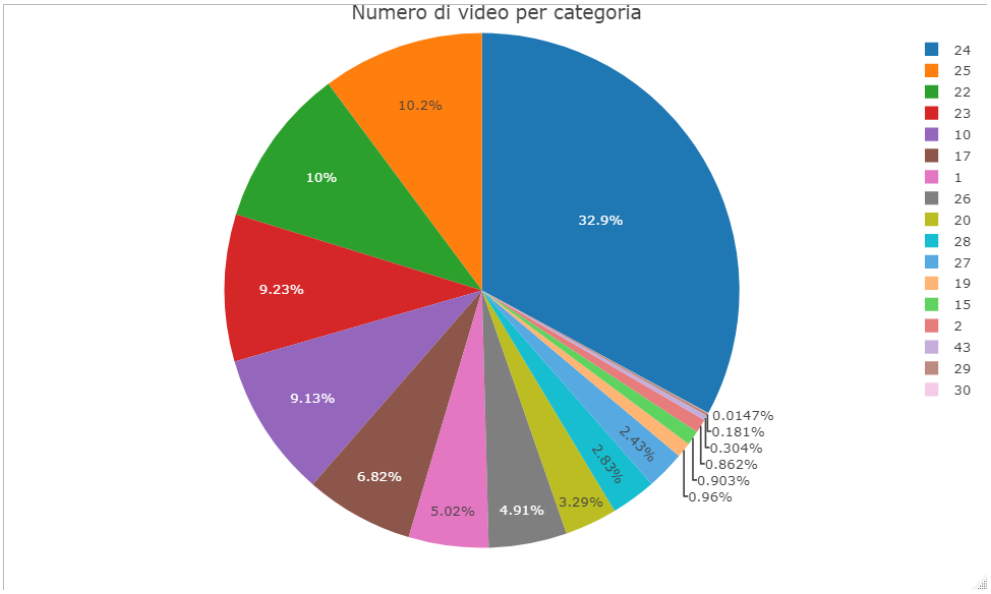
video_id	trending_date	channel_title	category_id	publish_time	views	likes	dislikes	comment_count
FIsCjmMhFmw	17.13.12	YouTube Spotlight	24	2017-12-06T17:58:51.000Z	137843120	3014479	1602383	817582
FIsCjmMhFmw	17.12.12	YouTube Spotlight	24	2017-12-06T17:58:51.000Z	125431369	2912715	1545018	807558
FIsCjmMhFmw	17.11.12	YouTube Spotlight	24	2017-12-06T17:58:51.000Z	113876217	2811217	1470387	787174
FIsCjmMhFmw	17.10.12	YouTube Spotlight	24	2017-12-06T17:58:51.000Z	100911567	2656678	1353655	682890
FIsCjmMhFmw	17.09.12	YouTube Spotlight	24	2017-12-06T17:58:51.000Z	75969469	2251826	1127811	827755
FIsCjmMhFmw	17.08.12	YouTube Spotlight	24	2017-12-06T17:58:51.000Z	52611730	1891822	884963	702784
pOHQdIDds6s	17.28.11	Jake Paul	22	2017-11-23T00:00:51.000Z	12921578	448453	504340	168477
FIsCjmMhFmw	17.07.12	YouTube Spotlight	24	2017-12-06T17:58:51.000Z	24784863	1149214	483943	461970
pOHQdIDds6s	17.27.11	Jake Paul	22	2017-11-23T00:00:51.000Z	12214051	439508	480359	161892
oWjxSkJpxFU	18.29.01	Logan Paul Vlogs	29	2018-01-24T18:30:01.000Z	22387656	1919980	461660	625010

Infine, considerando il numero di dislikes, i video che hanno ricevuto il maggior numero di "non mi piace" sono:

1. YouTube Rewind: The Shape of 2017 | #YouTubeRewind
2. Jake Paul - It's Everyday Bro (Remix) [feat. Gucci Mane]
3. Suicide: Be Here Tomorrow.

Numero di video in tendenza in base alla categoria



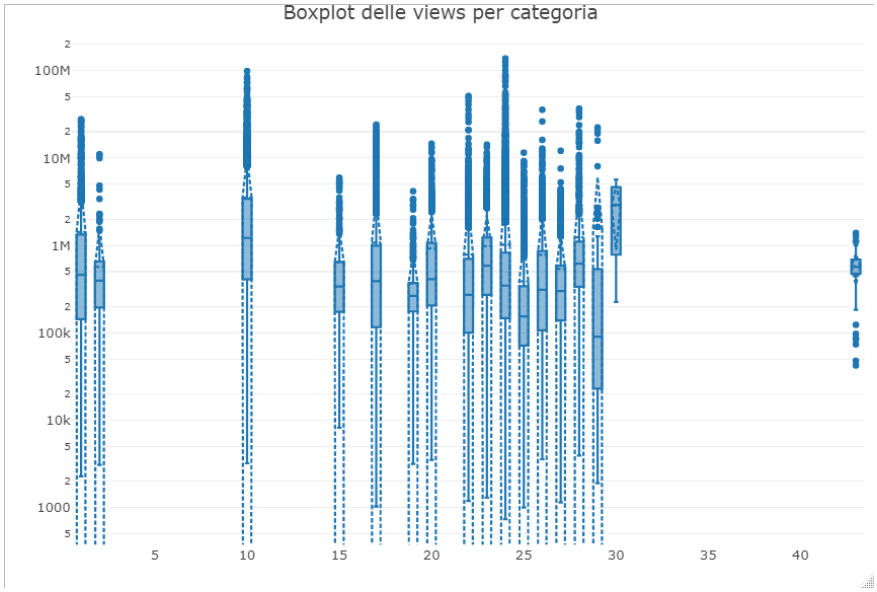


In questi due grafici si può vedere sostanzialmente la stessa cosa, ovvero il numero di video che sono andati in tendenza divisi per categoria di appartenenza.

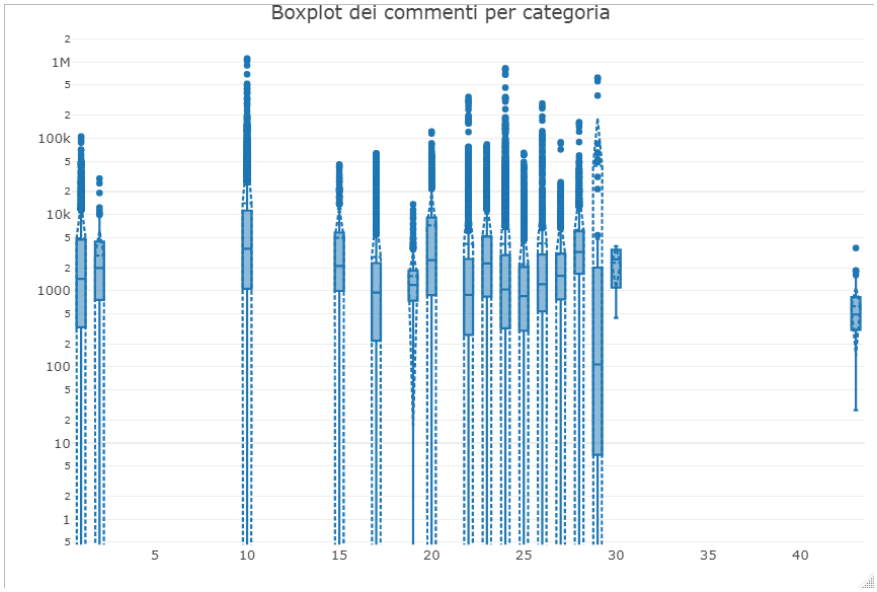
Nel primo grafico viene fatto un istogramma per rappresentare quanto detto sopra: si ottiene che la categoria che ha il maggior numero di video in tendenza è la 24, ovvero Entertainment (ogni video può essere contato più di una volta, una volta per ogni giorno in cui è andato in tendenza). Seguono, in misura simile tra di loro, le categorie 25, 22, 23, 10 (rispettivamente News&Politics, People&Blogs, Comedy e Music) e poi a mano a mano le altre con un numero di video decrescente.

Non tutte le categorie sono presenti in questo grafico, poiché vengono rappresentate solo quelle categorie i cui video sono andati in tendenza.

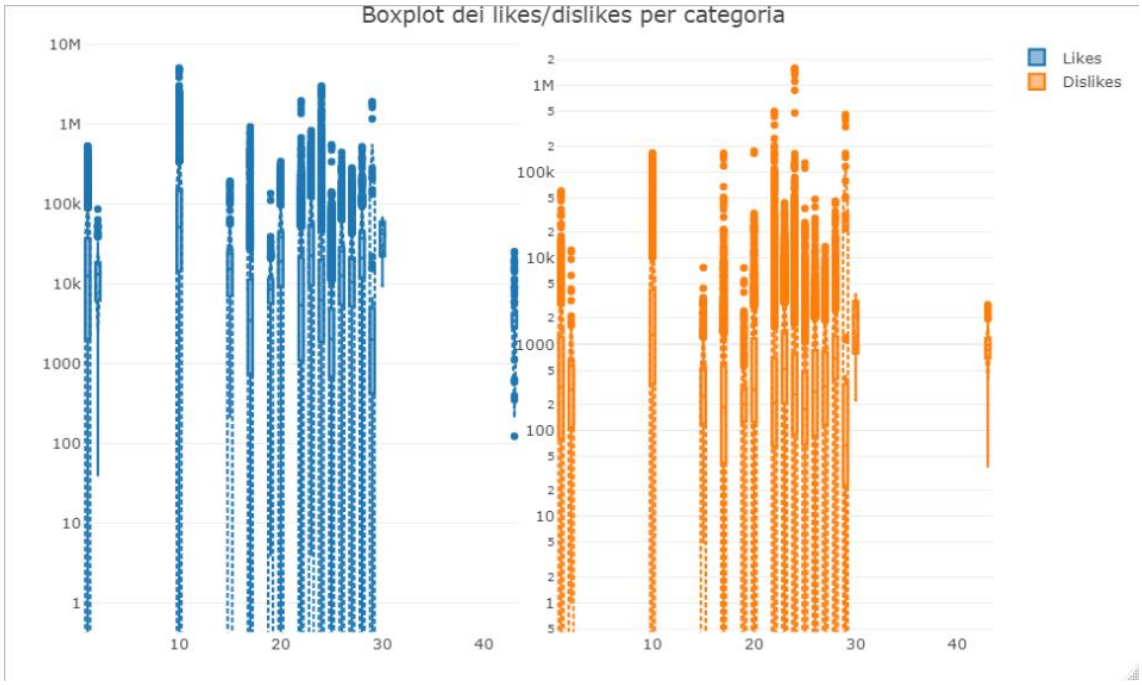
Boxplot dinamici



Questo boxplot, come i seguenti, è un boxplot dinamico: rappresenta una serie di boxplot, uno per ogni categoria di video, e di ogni boxplot si va a vedere il numero di visualizzazioni dei video di quella categoria. Si vede che non tutte le categorie sono state mappate, per lo stesso motivo descritto poco fa; le distribuzioni delle views delle varie categorie sono tutte abbatastanza simili, presentano tutte degli outlier per valori alti (da 1mln circa in su) ad eccezione dell'ultima categoria, la 43, che ha una box molto piccola e outlier sia al di sopra della box che al di sotto di essa.



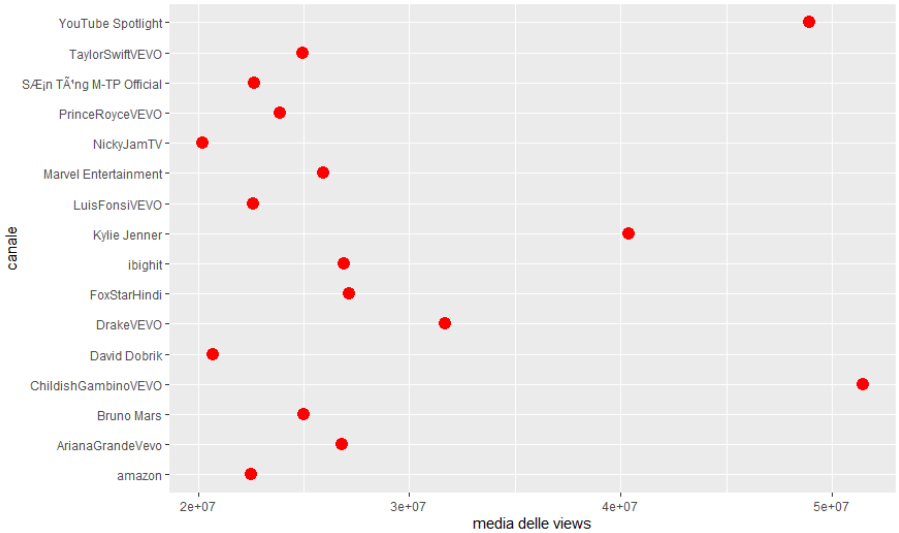
In questo grafico, invece, vengono rappresentati i boxplot delle categorie che mappano la distribuzione del numero di commenti per i video della categoria in esame. Rispetto al grafico precedente si nota che la scala dell'asse delle ordinate comprende un range di valori più bassi; come prima si può dire che le box sono circa tutte nello stesso intervallo, a parte quella della categoria 29 che è nettamente più ampia. Gli outlier anche in questo caso si trovano per i valori più alti (10 mila circa in su).



In questo grafico invece vengono affiancati i boxplot del numero di likes (a sx) e il boxplot del numero dei dislikes (a dx); sull'asse delle ascisse ci sono, come in precedenza, le categorie di appartenenza dei video.

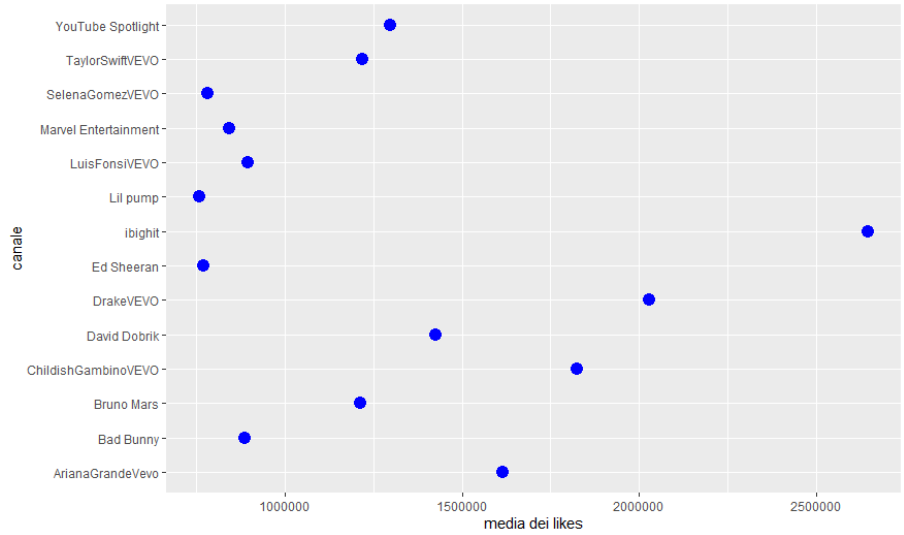
La differenza principale che si nota è la diversa scala di valori sull'asse delle ordinate: il numero di likes è nettamente maggiore del numero di dislikes. Questo è sensato, infatti i video hanno, in genere, soprattutto quelli più popolari, un maggior numero di likes rispetto a quello di dislikes.

Canali con maggior numero di views in media (> 20mln)



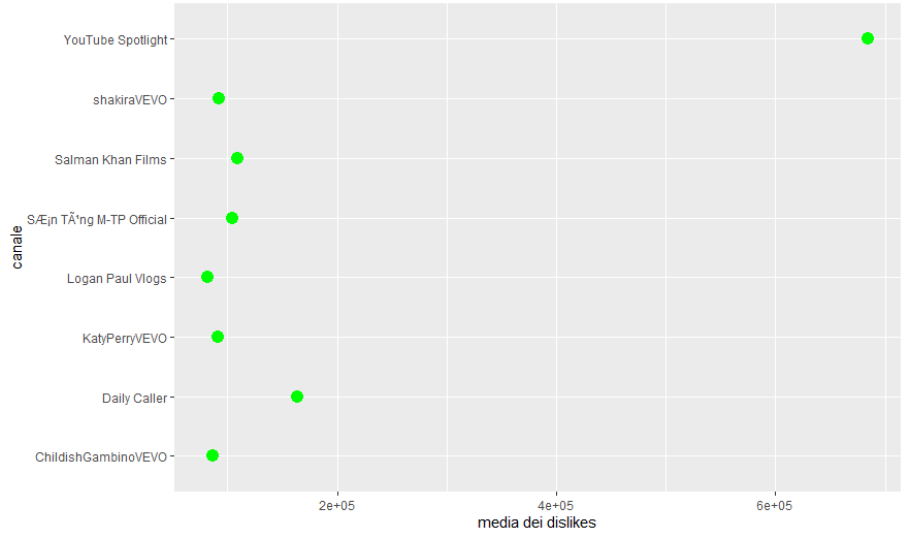
In questo grafico si vanno a visualizzare i canali che hanno fatto registrare il maggior numero di views in media; in questo grafico si sono considerati solo i canali che, in media, avessero più di 20 mln di views: il canale con più views risulta essere "ChildishGambinoVEVO" (più di 50mln di views) seguito da "YouTube Spotlight (poco meno di 50 mln di views in media).

Canali con maggior numero di likes in media (> 700 mila)



In questo grafico si considerano i canali che hanno pubblicato i video che hanno avuto, in media, il maggior numero di likes, considerando solo i canali che hanno ricevuto almeno 700 mila like (in media). Il canale che ha ricevuto più likes è "ibighit", ovvero "Big Hit Label", con più di 2.5 mln di likes.

Canali con maggior numero di dislikes in media (> 70 mila)



Infine, in questo grafico, sono stati individuati i canali che hanno ricevuto, in media, il maggior numero di dislikes, considerando solo i canali con più di 70 mila dislikes in media. Si ottiene come risultato che il canale con più dislikes in media è "YouTube Spotlight", con più di 500 mila dislikes.

ML algorithm: Reccomendation

Fase preliminare

Innanzitutto, viene creato un dataset di appoggio che è identico a quello utilizzato finora, ma con solo valori unici, ovvero considerando un'unica occorrenza per ogni video (non si avranno più multipli valori per un video); successivamente si procede togliendo le colonne che non servono (trending_date, channel_title, publish_time, comments_disabled, ratings_disabled, video_error_or_removed).

Non avendo giudizi, ratings espressi da ogni user nei confronti di un item si è optato per la costruzione di una matrice di similarità (in questo caso sono stati creati vettori di similarità per ogni video) item-item in cui sia sulle righe che sulle colonne sono posti i video e, nella cella ij-esima, c'è il valore dell'indice di similarità tra i due video.

Per calcolare la similarità sono stati presi in considerazione (a gruppi) il numero di commenti (poiché hanno la massima correlazione con il numero di likes), il numero delle views, il numero di likes, il numero di dislikes e l'intero associato alla categoria del video. Sono stati presi in considerazione anche id del video e titolo del video per una maggiore chiarezza nella stampa dei video simili.

Il dataset ottenuto è il seguente:

```
> str(DF_appoggio)
'data.frame':   24413 obs. of  7 variables:
 $ video_id     : Factor w/ 24427 levels "--45ws7CEN0",..: 14542 909 3054 7000 1917 1170 710 1788 12243 2409 ...
 $ title        : Factor w/ 24573 levels "'''Gala Artis 2018'' Le numÃ©ro d'ouverture",..: 7886 16653 17304 10553 7746
 11410 22920 23362 20575 8523 ...
 $ category_id  : int  10 23 23 24 10 25 23 22 24 22 ...
 $ views        : int  17158579 1014651 3191434 2095828 33523622 1309699 2987945 748374 4477587 505161 ...
 $ likes        : int  787425 127794 146035 132239 1634130 103755 187464 57534 292837 4135 ...
 $ dislikes     : int  43420 1688 5339 1989 21082 4613 9850 2967 4123 976 ...
 $ comment_count: int  125882 13030 8181 17518 85067 12143 26629 15959 36391 1484 ...
```

A questo punto vengono riordinate le colonne in modo che gli indici corrispondenti ai video che sono stati eliminati vengano riassociati a video presenti nel dataset (e avere una numerazione che arriva a 24 mila circa e non più a 40 mila circa).

Sono stati creati 3 modelli di similarità basati sulla cosine similarity e 3 basati su Pearson, in modo da poterli confrontare e stabilire quale, tra di essi, fosse il modello migliore.

Cosine similarity:

La cosine similarity rappresenta una misura di somiglianza tra 2 vettori; geometricamente corrisponde al coseno dell'angolo tra i due vettori. Nel caso in cui le componenti dei vettori siano non-negative (come spesso nel caso dei ratings), la cosine similarity può variare tra 0 ($\theta=\pi 2$, completa diversità) e 1 ($\theta=0$, massima somiglianza).

$$\text{sim}(\vec{a},\vec{b}) = \frac{\vec{a}\cdot\vec{b}}{\|\vec{a}\|\cdot\|\vec{b}\|} = \frac{\sum_{i=1}^{n} a_i \cdot b_i}{\sqrt{\sum_{i=1}^{n} a_i^2}\cdot\sqrt{\sum_{i=1}^{n} b_i^2}}$$

Pearson correlation:

L'indice di correlazione è dato dal rapporto tra covarianza e prodotto delle deviazioni standard dei 2 campioni a e b può variare tra -1 (perfetta correlazione lineare negativa) a +1 (perfetta correlazione lineare positiva) dove il valore 0 significa che i due vettori non sono correlati linearmente (non che siano indipendenti).

$$\mathrm{sim}(\vec{a},\vec{b}) = \frac{\sum_{i\in I_{ab}}(a_i-\bar{a})\cdot(b_i-\bar{b})}{\sqrt{\sum_{i\in I_{ab}}(a_i-\bar{a})^2}\cdot\sqrt{\sum_{i\in I_{ab}}(b_i-\bar{b})^2}}$$

In seguito, viene creata la matrice associata al dataset di appoggio (che servirà poi nel calcolo della similarità) e viene scelto casualmente un valore tra 1 e l'indice massimo dei video in modo da prendere come target il video che ha come indice quello estratto casualmente, per trovare i video più simili a quel target.

```
i <- floor(runif(1, min=1, max=24414))   #prendo un numero casuale
```

```
> some(DF_matrix)
      video_id title category_id   views likes dislikes comment_count
1207      2739  2823          24   19711    22        6             15
2438     14062 15228          24 1337759 30387     1686           6010
6116      8445 13377          23   48235  4812       36            492
6482     15664 18358          22   29692   704       87            734
10352    22986 10220          24  653580 28426      331           2143
16283     7598 11173          24  525944 13154      508           1561
17283    12259   800          24   78574  5750       55           2016
18468     2017 13178          24  266619 46836      348           3054
20116     5928  7111          24  580146 24820      318           4481
22975    14311  1547           1  754601 20226      462           1340
```

(matrice del dataset)

Indice associato al video

Si è provato a creare la matrice di similarità,

```
similarity_matrix <- matrix(, ncol = nrow(DF_appoggio), nrow = nrow(DF_appoggio))
for (i in 1:nrow(DF_appoggio)) {
  for (j in 1:nrow(DF_appoggio)) {
    if(DF_appoggio$category_id[i] == DF_appoggio$category_id[j]){
     similarity_matrix[i][j] <- cosine(c(DF_matrix[i, 4], DF_matrix[i, 5], DF_matrix[i, 6], DF_matrix[i, 7]),
                                       c(DF_matrix[j, 4], DF_matrix[j, 5], DF_matrix[j, 6], DF_matrix[j, 7]))
    }
    else similarity_matrix[i][j] <- 0
  }
}
```

ma a causa delle grandi dimensioni del dataset, non si è potuto crearla; perciò per ogni video-target verrà calcolato il vettore di similarità (ordinato in ordine decrescente di similarità) e verranno stampati i 5 video più simili al target (il video target + i top 5 video più simili).

Modello #1

Cosine

In questo caso viene costruito un modello che calcola la similarità tra i video basandosi sul numero di views, likes, dislikes e commenti con la similarità del coseno.

```
similarity_vector_vld <- vector(, length = nrow(DF_appoggio))
for (j in 1:nrow(DF_appoggio)) {
   similarity_vector_vld[j] <- cosine(c(DF_matrix[i, 4], DF_matrix[i, 5], DF_matrix[i, 6], DF_matrix[i, 7]),
                          c(DF_matrix[j, 4], DF_matrix[j, 5], DF_matrix[j, 6], DF_matrix[j, 7]))
}
sort_vector_vld <- sort(similarity_vector_vld, decreasing = TRUE, index.return = TRUE)
print(DF_appoggio[sort_vector_vld$ix[1:6], ]) #video-target + migliori 5 video correlati
```

Si procede ora ad illustrare un esempio per il video di indice 11409.

I 5 video più simili saranno i seguenti:

```
> print(DF_appoggio[sort_vector_vld$ix[1:6], ]) #video-target + migliori 5 video correlati
        video_id                                                                 title category_id  views likes dislikes comment_count
11409 44aLHs6uEcs EXOTIC THAI FOOD Tour! SUPER RARE street food of Chiang Mai, Thailand          19  86544  3503       62           696
6366   tW69sJkOwMA                                                  This Rock Is Important          24 365814 14836      200          3031
19825 bKYOe8-r3Mw  Prepare Yourself, Deep State Narrative Falling Apart - Episode 1562b          22  41898  1688       39           354
17155 vDOww9hfTpw                                        How to catch a ball with your face          24 119170  4767      100           970
21367 KwsMBGm1zBc                        TAKING SIDES | Niki and Gabi Take New York S3 EP 3          24 212123  8652      196          1777
20342 9nkiqSX9h1E                                  Furniture FLIP Challenge! (ft. Team Edge)          24  86699  3517       46           656
```

La prima riga sta ad indicare il video-target, le altre righe i video più simili, i quali avranno correlazione pari a 1.0000000 il video-target con il primo più simile, 0.9999999 con il secondo, 0.9999999 con il terzo, 0.9999999 con il quarto e 0.9999999 con il quinto (questi valori sono estratti dal sort_vector utilizzato dal modello preso in considerazione).

Pearson

Ora si procede al calcolo della similarità tramite lo stesso modello, ma con il calcolo della similarità tramite il coefficiente di correlazione di Pearson, sempre per il video 11409.

```
similarity_vector_vld_p <- vector(, length = nrow(DF_appoggio))
for (j in 1:nrow(DF_appoggio)) {
  similarity_vector_vld_p[j] <- cor(c(DF_matrix[i, 4], DF_matrix[i, 5], DF_matrix[i, 6], DF_matrix[i, 7]),
                                    c(DF_matrix[j, 4], DF_matrix[j, 5], DF_matrix[j, 6], DF_matrix[j, 7]))
}
sort_vector_vld_p <- sort(similarity_vector_vld_p, decreasing = TRUE, index.return = TRUE)
print(DF_appoggio[sort_vector_vld_p$ix[1:6], ]) #video-target + migliori 5 video correlati
```

I 5 video più simili sono i seguenti:

```
> print(DF_appoggio[sort_vector_vld_p$ix[1:6], ]) #video-target + migliori 5 video correlati
          video_id                                                                  title category_id  views likes dislikes comment_count
11409 44aLHs6uEcs EXOTIC THAI FOOD Tour! SUPER RARE street food of Chiang Mai, Thailand          19  86544  3503       62           696
5764  It5hauk2cV8                                    FISH HOUSE TOUR of Solid Gold Aquatics      15 132376  5426      166          1132
21367 KwsMBGm1zBc                        TAKING SIDES | Niki and Gabi Take New York S3 EP 3      24 212123  8652      196          1777
3045  rx8FzevZzvU                                          If Undertale was Realistic 15      20 241671  9868      284          2077
6828  nyrIKtFZx3s                                       Weigh Yourself After Every Dump       24 215416  8675       93          1635
6366  tW69sJk0wMA                                                 This Rock Is Important      24 365814 14836      200          3031
```

con similarità pari a 1.0000000 per il video target con sé stesso, 1.0000000 per il video-target con il primo video più simile, 1.0000000 con il secondo, 1.0000000 con il terzo, 1.0000000 con il quarto e 0.9999999 con il quinto.

Si nota che i video che vengono scelti come più simili al video-target cambiano in base al modo in cui si calcola la similarità; alcuni video restano uguali, altri cambiano o cambia l'ordine in cui vengono consigliati (questo perché cambia il valore dell'indice di similarità).

Questo modello viene però bocciato, poiché tiene conto solo delle variabili scelte all'inizio (views, likes, dislikes e commenti), quindi può succedere che il reccomender system consigli video di categorie diverse da quella del video che è appena stato guardato. Questo "comportamento" non è troppo sensato perché si sarà più predisposti a guardare un video simile, appartenente alla stessa categoria del video appena visto (se questo è piaciuto).

Si passa perciò a prendere in considerazione il secondo modello.

Modello #2

Cosine

Il secondo modello si basa sull'intero associato all'id della categoria, numero di views, likes, dislikes e commenti. Vengono calcolati i video più simili con la cosine similarity.

```
similarity_vector_ccvId <- vector(, length = nrow(DF_appoggio))
for (j in 1:nrow(DF_appoggio)) {
   similarity_vector_ccvId[j] <- cosine(c(DF_matrix[i, 3], DF_matrix[i, 4], DF_matrix[i, 5], DF_matrix[i, 6], DF_matrix[i, 7]),
                        c(DF_matrix[j, 3], DF_matrix[j, 4], DF_matrix[j, 5], DF_matrix[j, 6], DF_matrix[j, 7]))
}
sort_vector_ccvId <- sort(similarity_vector_ccvId, decreasing = TRUE, index.return = TRUE)
print(DF_appoggio[sort_vector_ccvId$ix[1:6], ])   #video-target + migliori 5 video correlati
```

Si procede ora ad illustrare un esempio per il video di indice 11409 (con il video + la top 5 dei simili):

```
> print(DF_appoggio[sort_vector_ccvId$ix[1:6], ])   #video-target + migliori 5 video correlati
        video_id                                                                title category_id  views likes dislikes comment_count
11409 44aLHs6uEcs EXOTIC THAI FOOD Tour! SUPER RARE street food of Chiang Mai, Thailand          19  86544  3503       62           696
6366  tW69sJkOwMA                                                  This Rock Is Important          24 365814 14836      200          3031
17155 vDOww9hfTpw                                          How to catch a ball with your face    24 119170  4767      100           970
21367 KwsMBGm1zBc                          TAKING SIDES | Niki and Gabi Take New York S3 EP 3    24 212123  8652      196          1777
20342 9nkiqSX9h1E                                  Furniture FLIP Challenge! (ft. Team Edge)     24  86699  3517       46           656
6828  nyrIKtFZx3s                                            Weigh Yourself After Every Dump     24 215416  8675       93          1635
```

I video più simili avranno similarità pari a 1.0000000 (il video target con sé stesso), 0.9999999 il video-target con il primo video più simile, 0.9999999 con il secondo, 0.9999999 con il terzo, 0.9999999 con il quarto e 0.9999998 con l'ultimo.

Pearson

Il secondo modello si basa sull'intero associato all'id della categoria, numero di views, likes, dislikes e commenti. Vengono ora calcolati i video più simili con la Pearson correlation.

```
similarity_vector_ccvId_p <- vector(, length = nrow(DF_appoggio))
for (j in 1:nrow(DF_appoggio)) {
  similarity_vector_ccvId_p[j] <- cor(c(DF_matrix[i, 3], DF_matrix[i, 4], DF_matrix[i, 5], DF_matrix[i, 6], DF_matrix[i, 7]),
                                      c(DF_matrix[j, 3], DF_matrix[j, 4], DF_matrix[j, 5], DF_matrix[j, 6], DF_matrix[j, 7]))
}
sort_vector_ccvId_p <- sort(similarity_vector_ccvId_p, decreasing = TRUE, index.return = TRUE)
print(DF_appoggio[sort_vector_ccvId_p$ix[1:6], ])   #video-target + migliori 5 video correlati
```

Esempio per il video di indice 11409:

```
> print(DF_appoggio[sort_vector_ccvId_p$ix[1:6], ])   #video-target + migliori 5 video correlati
         video_id                                                                title category_id  views likes dislikes comment_count
11409 44aLHs6uEcs EXOTIC THAI FOOD Tour! SUPER RARE street food of Chiang Mai, Thailand          19  86544  3503       62           696
6828  nyrIKtFZx3s                                        Weigh Yourself After Every Dump          24 215416  8675       93          1635
6366  tW69sJkOwMA                                                  This Rock Is Important          24 365814 14836      200          3031
21367 KwsMBGm1zBc                        TAKING SIDES | Niki and Gabi Take New York S3 EP 3          24 212123  8652      196          1777
19825 bKYOe8-r3Mw  Prepare Yourself, Deep State Narrative Falling Apart - Episode 1562b          22  41898  1688       39           354
20342 9nkiqSX9h1E                                Furniture FLIP Challenge! (ft. Team Edge)          24  86699  3517       46           656
```

I video più simili avranno similarità pari a 1.0000000 (il video target con sé stesso), 1.0000000 il video-target con il primo video più simile, 0.9999999 con il secondo, 0.9999999 con il terzo, 0.9999999 con il quarto e 0.9999999 con l'ultimo.
Anche in questo caso alcuni video coincidono, ma non sono gli stessi tra i due modelli, cambiano in base al modo in cui si calcola la similarità.

Questo secondo modello viene scartato invece perché, nonostante tenga conto della categoria, il che è un upgrade rispetto al modello precedente, presenta una possibile serie di errori: considerando infatti l'intero associato alla categoria nel calcolo della similarità si può ottenere che due video siano simili poiché sono vicini i numeri delle loro categorie (es 19 e 20, rispettivamente Travel&Events e Gaming), ma in realtà le categorie potrebbero trattare di argomenti decisamente lontani tra di loro e quindi il video simile che viene raccomandato non è in realtà simile al video-target.

Modello #3

Cosine

Il terzo modello si basa sul numero di views, likes, dislikes e commenti, ma considera solo i video appartenenti alla stessa categoria del video in analisi (tramite l'if nel codice sottostante).

```
similarity_vector <- vector(, length = nrow(DF_appoggio))
for (j in 1:nrow(DF_appoggio)) {
  if(DF_appoggio$category_id[i] == DF_appoggio$category_id[j]){
    similarity_vector[j] <- cosine(c(DF_matrix[i, 4], DF_matrix[i, 5], DF_matrix[i, 6]),
                                   c(DF_matrix[j, 4], DF_matrix[j, 5], DF_matrix[j, 6]))
  }
else similarity_vector[j] <- 0
}
sort_vector <- sort(similarity_vector, decreasing = TRUE, index.return = TRUE)
print(DF_appoggio[sort_vector$ix[1:6], ])   #video-target + migliori 5 video correlati
```

Si procede ora ad illustrare un esempio per il video di indice 11409:

```
> print(DF_appoggio[sort_vector$ix[1:6], ])   #video-target + migliori 5 video correlati
         video_id                                                                  title category_id  views likes dislikes comment_count
11409 44aLHs6uEcs     EXOTIC THAI FOOD Tour! SUPER RARE street food of Chiang Mai, Thailand          19  86544  3503       62           696
8561  FUmUoE-n8BU                              NOODLE PARADISE! Central Vietnam Noodle Tour          19 104868  4358       37           847
24341 cXiOMqyDRrU       Spicy KOREAN FOOD Tour! Can Foreigners handle SPICY FOOD in Korea?          19 143515  5625      100          1097
16395 GfUIsHizsis                                 Street Food & Insane SEAFOOD in Mumbai India      19 155112  6317      134          1032
16019 OZo-NYh8F3k                First Time Trying TRADITIONAL Indian Food in Mumbai India!       19 188168  7651      145          1821
20554 sLOJtdXSJ6w Exotic Indian Street Food Tour in Delhi, India! Crazy FLAMING FIRE PAAN!       19 117756  4525      100           902
```

I valori di similarità ottenuti nel vettore associato al video target sono i seguenti: 1.0000000 (il video target con sé stesso), 0.9999994 il video-target con il primo video più simile, 0.9999991 con il secondo, 0.9999990 con il terzo, 0.9999986 con il quarto e 0.9999978 con l'ultimo.

Successivamente si passa allo sviluppo del modello con l'indice di correlazione di Pearson.

Pearson

Infine, l'ultimo modello si basa, come quello appena sopra, sul numero di views, likes, dislikes e commenti (considerando solo i video appartenenti alla stessa categoria del video in analisi), ma, in questo caso, la similarità tra i video viene calcolata attraverso la correlazione di Pearson, non attraverso la cosine similarity.

```
similarity_vector_p <- vector(, length = nrow(DF_appoggio))
for (j in 1:nrow(DF_appoggio)) {
  if(DF_appoggio$category_id[i] == DF_appoggio$category_id[j]){
    similarity_vector_p[j] <- cor(c(DF_matrix[i, 4], DF_matrix[i, 5], DF_matrix[i, 6], DF_matrix[i, 7]),
                                  c(DF_matrix[j, 4], DF_matrix[j, 5], DF_matrix[j, 6], DF_matrix[j, 7]))
  }
else similarity_vector[j] <- 0
}
sort_vector_p <- sort(similarity_vector_p, decreasing = TRUE, index.return = TRUE)
print(DF_appoggio[sort_vector_p$ix[1:6], ])   #video-target + migliori 5 video correlati
```

Si procede ora ad illustrare un esempio per il video di indice 11409:

```
> print(DF_appoggio[sort_vector_p$ix[1:6], ])   #video-target + migliori 5 video correlati
          video_id                                                    title category_id  views likes dislikes comment_count
11409 44aLHs6uEcs EXOTIC THAI FOOD Tour! SUPER RARE street food of Chiang Mai, Thailand          19  86544  3503       62           696
24341 cXiOMqyDRrU    Spicy KOREAN FOOD Tour! Can Foreigners handle SPICY FOOD in Korea?          19 143515  5625      100          1097
8561  FUmUoE-n8BU                          NOODLE PARADISE! Central Vietnam Noodle Tour          19 104868  4358       37           847
16019 0Zo-NYh8F3k           First Time Trying TRADITIONAL Indian Food in Mumbai India!          19 188168  7651      145          1821
16395 GfUIsHizsis                             Street Food & Insane SEAFOOD in Mumbai India          19 155112  6317      134          1032
13905 w46Fslwmujo             NIGHT MARKET FOOD in Manila Philippines: BBQ & BLOOD STEW          19 222803  8682      147          1293
```

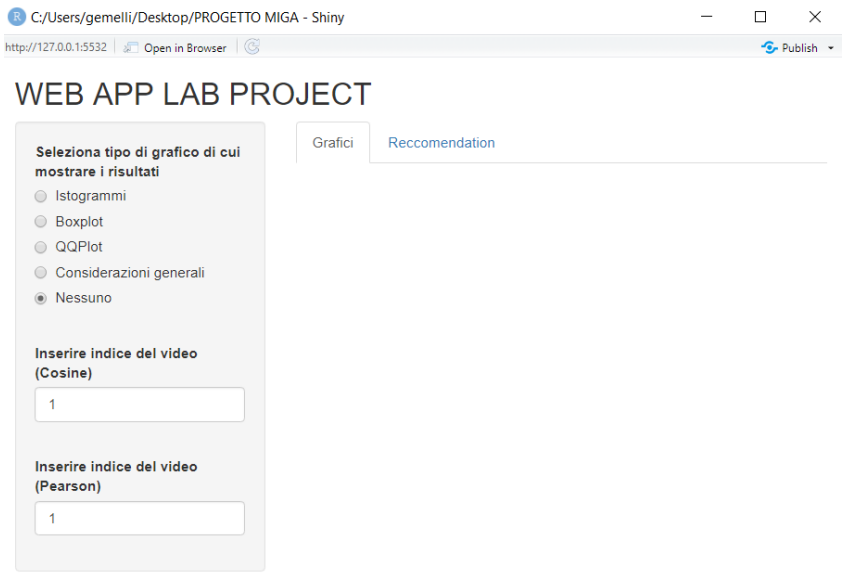
I valori di similarità ottenuti sono 1.0000000 (il video-target con sé stesso), 0.9999994 il video-target con il primo video più simile, 0.9999992 con il secondo, 0.9999989 con il terzo, 0.9999988 con il quarto e 0.9999983 con il quinto.
Anche in questo caso si ha che i top 5 video più simili al video-target sono diversi tra di loro (o in ordine diverso) in base all'indice di similarità che si utilizza per il calcolo del vettore di similarità.

Si può dire che questo è il modello migliore, infatti si vanno a selezionare i video più simili tra di loro, ovvero quelli più simili al video-target che contemporaneamente appartengano alla sua stessa categoria; sicuramente due video della stessa categoria saranno più simili tra di loro piuttosto che due video di due categorie diverse. Il modello è ottimale in questo caso perché va a scegliere, tra i video della stessa categoria, quelli che sono più simili al video-target per quanto riguarda numero di views, likes, dislikes e commenti.

Si può infine concludere che i video che saranno maggiormente simili al video-target saranno quelli trovati nel terzo modello, in particolare quelli che saranno presenti in entrambe le top 5 dei modelli, sia di quello calcolato con la cosine similarity, sia di quello calcolato con la Pearson correlation.

Web App Shiny

Come ultimo punto del progetto è stata implementata una web app, nel caso specifico è stata implementata in parte sulla sezione di **exploratory analysis** e in parte su quella di **ML algorithm**; quest'app è interattiva, infatti mostra diverse osservazioni, diversi grafici in base agli input ricevuti dall'user che utilizza l'applicazione; nella parte del reccomendation suggerisce i top 5 video più simili al video che viene passato in input (tramite l'indice associato al video).



La UI della web applet presenta, all'apertura, un pannello laterale e due finestre dedicate: una per la presentazione dei grafici, l'altra per il reccomendation system.

Nel pannello laterale si ha la possibilità di scegliere quale tipo di grafico si vuole visualizzare: "Istogrammi", ovvero gli istogrammi delle 4 variabili numeriche (numero di views, likes, dislikes e commenti); "Boxplot", che permette di vedere i boxplot delle 4 variabili appena citate; "QQPlot", che permette di visualizzare i QQPlot delle 4 variabili; "Considerazioni generali", che va a mostrare alcuni grafici più generali, come la correlazione tra le 4 variabili, i video in tendenza raggruppati per categoria, numero di views medie per canale, ecc.; cliccando il bottone "Nessuno" si tornerà alla visione di default, ovvero non verrà mostrato nessun grafico. Questi grafici appariranno nel tab "Grafici" (che è quello che si apre di default).

Se si clicca, invece, sul tab "Reccomendation", appariranno due box testuali che mostreranno i risultati del reccomendation system costruito nella parte di ML algorithm. Una box sarà per il video-target più i 5 video più simili che vengono trovati con la cosine similarity e una per il video-target più i 5 video più simili che vengono trovati tramite la Pearson correlation. Nel pannello laterale, oltre alla scelta dei grafici, si può inserire un

valore numerico (compreso tra 1 e 24413, ovvero l'intervallo in cui sono compresi gli indici dei video) che sta ad indicare l'indice del video target, di cui si vogliono trovare i 5 video più simili. Le "caselle di testo" sono 2, una per la selezione del video target di cui si vogliono visualizzare i video simili trovati tramite la cosine similarity e una per la selezione del video target di cui si vogliono visualizzare i video simili trovati tramite la Pearson correlation. I valori nelle caselle di testo possono essere modificati sia incrementando/decrementando di 1 il valore dell'indice del video-target (con le freccette messe a disposizione nell'interfaccia grafica), sia digitando nella casella un valore desiderato; con il cambiamento di valore dell'indice cambieranno il video-target e, di conseguenza, anche i 5 video più simili. Del video target e dei video simili verranno mostrati: id del video, titolo del video, categoria di appartenenza, numero di views, likes, dislikes e commenti. Il modello utilizzato per fare le due raccomandazioni è il terzo modello descritto nella sezione di ML algorithm, ovvero il modello migliore.

Conclusions

In conclusione, si può affermare che l'obiettivo del progetto era quello di fare un'analisi sul dataset scelto e poi applicare un algoritmo di ML su questo dataset.

Si è deciso di procedere mediante diverse fasi: data acquisition, exploratory analysis, ML algorithm e web app.

Nella prima parte di **data acquisition** è stato fatto un lavoro di descrizione della composizione del dataset e successivamente di pulizia del dataset da eventuali valori mancanti o Na o NaN e da valori non rilevanti ai fini dell'analisi successiva.

Nella parte successiva, ovvero quella di **exploratory analysis,** sono state considerate le variabili numeriche del dataset e su quelle è stata condotta un'analisi: sono stati fatti diversi grafici per descrivere meglio le variabili, come istogrammi, boxplot, QQPlot, e altri grafici, più complessi, che mostrassero alcune situazioni particolari e le relazioni tra le variabili all'interno del dataset stesso (come correlazione tra le variabili, video con più likes, video per ogni categoria, ecc.).

Per la parte di **ML algorithm** è stato creato un reccomendation system, o meglio sono stati creati 3 modelli di reccomendation che trovassero, per un video-target rappresentato dal suo indice, i 5 video più simili ad esso, calcolando la similarità prima con la cosine similarity e poi con la correlazione di Pearson.

Nell'ultima parte, quella dedicata alla **web app,** è stata creata una web app interattiva nella quale, in base alle scelte dell'utente, vengono implementate diverse funzionalità che hanno come finalità principale quella di mostrare i grafici e i vari risultati che sono stati raccolti nella parte di exploratory analysis e, su input dell'utente, di stampare i video più simili al video-target dato in input dall'user (tramite l'indice del video).