YouTubeCA Report

Ricky Rahardja

01/09/2019

Introduction

This assignment is one of the final project in HarvardX Data Science Professional Program on edX. In this project, author is required to obtain public data, conduct necessary analysis and build up machine learning algorithm based on data acquired and in interest of author.

Background of this assignment is driven by author's interest in learning more on YouTube and its contents. Author has selected Canada channels because he has observed that author's kids who are 11 and 9 years old are regularly watching YouTube Channel from Canada. From this point, author found the relevant dataset from kaggle. Author mainpoint of interest is where youtuber always note to viewers to: watch full videos, like, subscribe,and comment.

Author is then preparing the necessary for the dataset and named as dataCA which is also provided in this assignmnet submission. Author starts with data analysis and describing the data on hand. From the descriptive analysis on the data onhand, then author starts to study the connection between interested variables. Further author will build up machine learning algorith based on regression models and try to find the best model by assessing models with RMSE.

Loading and Describing Data

```
library(tidyverse)
```

```
## -- Attaching packages ---------------------------------------------------- tidyverse 1.2.1 --
## v ggplot2 3.2.1     v purrr   0.3.2
## v tibble  2.1.3     v dplyr   0.8.3
## v tidyr   0.8.3     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.4.0
## Warning: package 'ggplot2' was built under R version 3.5.2
## Warning: package 'tibble' was built under R version 3.5.2
## Warning: package 'tidyr' was built under R version 3.5.2
## Warning: package 'purrr' was built under R version 3.5.2
## Warning: package 'dplyr' was built under R version 3.5.2
## Warning: package 'stringr' was built under R version 3.5.2
## Warning: package 'forcats' was built under R version 3.5.2
## -- Conflicts ------------------------------------------------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
load("~/LearnR/YouTubeCA/dataCA.RData")

glimpse(dataCA)
```

```
## Observations: 40,881
## Variables: 5
## $ views         <int> 17158579, 1014651, 3191434, 2095828, 33523622, 1...
## $ likes         <int> 787425, 127794, 146035, 132239, 1634130, 103755,...
## $ dislikes      <int> 43420, 1688, 5339, 1989, 21082, 4613, 9850, 2967...
## $ comment_count <int> 125882, 13030, 8181, 17518, 85067, 12143, 26629,...
## $ category_id   <int> 10, 23, 23, 24, 10, 25, 23, 22, 24, 22, 10, 26, ...
```

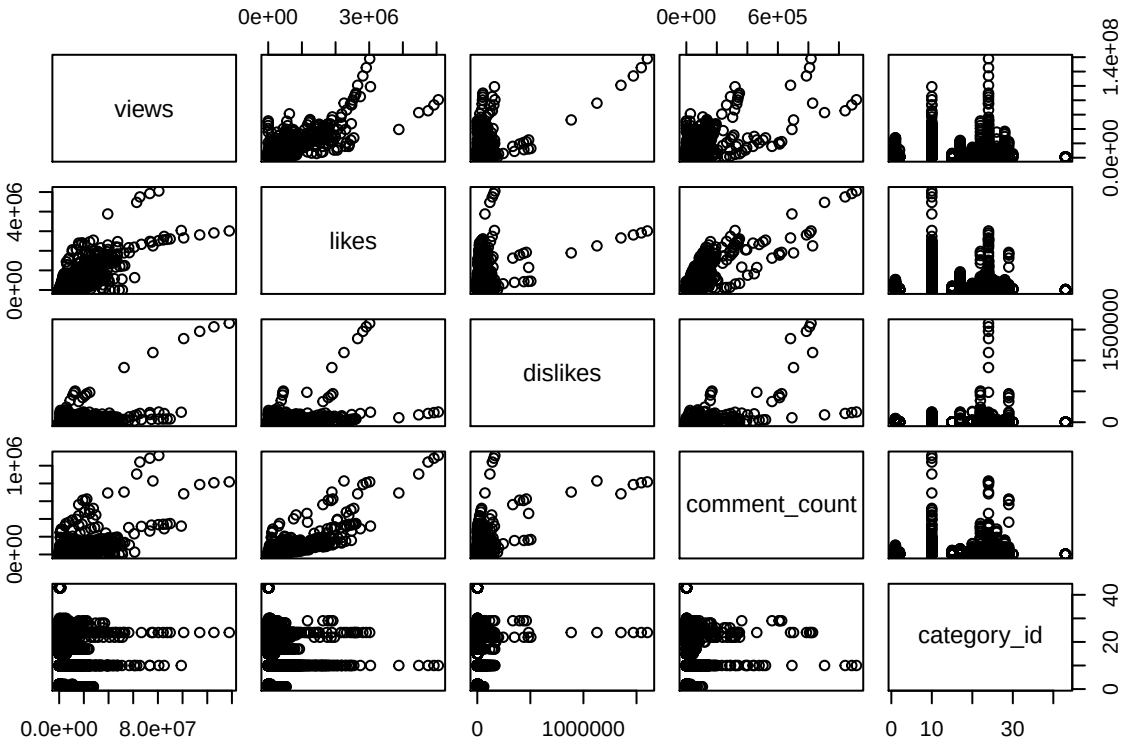
```
summary(dataCA)
```

```
##      views               likes            dislikes       comment_count
##  Min.   :      733   Min.   :      0   Min.   :      0   Min.   :      0
##  1st Qu.:   143902   1st Qu.:   2191   1st Qu.:     99   1st Qu.:    417
##  Median :   371204   Median :   8780   Median :    303   Median :   1301
##  Mean   :  1147036   Mean   :  39583   Mean   :   2009   Mean   :   5043
##  3rd Qu.:   963302   3rd Qu.:  28717   3rd Qu.:    950   3rd Qu.:   3713
##  Max.   :137843120   Max.   :5053338   Max.   :1602383   Max.   :1114800
##   category_id
##  Min.   : 1.0
##  1st Qu.:20.0
##  Median :24.0
##  Mean   :20.8
##  3rd Qu.:24.0
##  Max.   :43.0
```

There are 43 category ID with 40,881 observation and four continuous discrete variables. The four continuous discrete variables are number of views, likes, dislikes and comment counts.

Now I want to get a general ideas on the correlation between variables in the dataset.

```
pairs(dataCA)
```



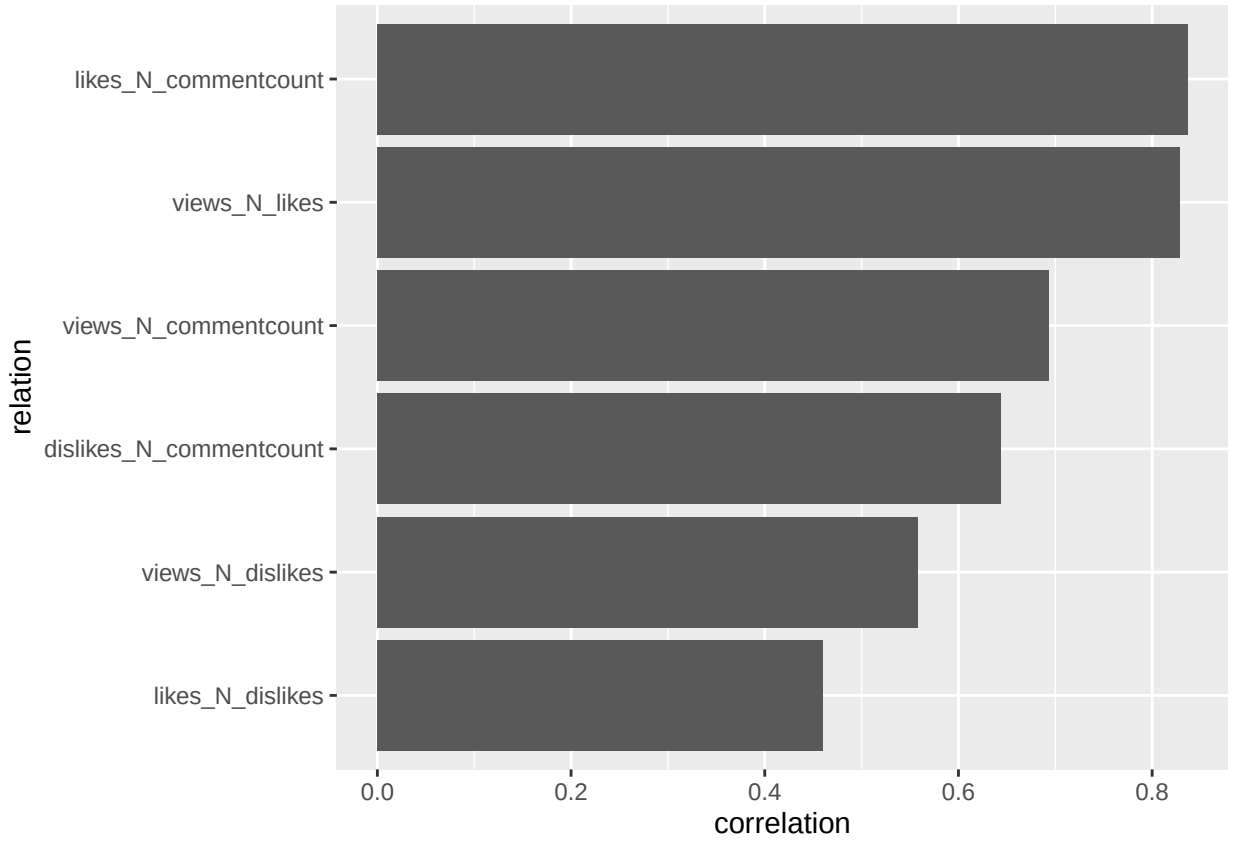
From the overall visualization of correlation between variables, it seems that number of views has strongest

correlation wih number of likes, followed by number of comment counts. Author also found that category cannot be included this analysis because it a categorical factor without level.

```
correlation_matrix <- data.frame(relation = c("views_N_likes", "views_N_dislikes", "views_N_commentcoun
correlation_matrix
```

```
##                  relation correlation
## 1    likes_N_commentcount   0.8365847
## 2           views_N_likes   0.8289643
## 3    views_N_commentcount   0.6931066
## 4 dislikes_N_commentcount   0.6434942
## 5        views_N_dislikes   0.5576211
## 6        likes_N_dislikes   0.4604267
```

```
correlation_matrix %>% mutate(relation = fct_reorder(relation, correlation)) %>%
  ggplot(aes(relation, correlation)) +
  geom_bar(stat = "identity") + coord_flip()
```



Author discover that number of likes and comments has the highest correlation, follows by correalation between number of views and likes and correation between number of views and comments. Thus, this shows there are strong relationship between these 3 variables: number of views, likes and comments. number of likes seems to be strongly correlated with number of comments and views.

In this assignment, author wants to build up a predictive model that is useful for users such as youtubers. After some observation on several youtubers, author notice that some youtubers are interested on number of views, some are on number of likes, and some are on number of comments.

Now, if author wants to become a youtuber and producing content regularly is the main objective, the basic response variable is logically the number of viewers earned for the content uploaded. However, this is not necessarily the case as number of views does not guarantee viewer are engage with the content.

Based on the descriptive analysis above, number of likes have the strongest correlation with number of views and number of comments. This does make sense since likes, dislikes, and comment counts are the engaging links between viewers and contents, for those who likes and engage with the contents.

Therefore, building a predictive model on number of likes does make more sense by using number of views and comments as the explanatory variables. Further, author will expand the model also using category ID.

Separating train and test set with validation set.

Author is using cret package to make partition of train and test set. There are 40,881 observations in this dataset and author is using 20% to test the algorithm.

```
library(caret)
```

```
## Warning: package 'caret' was built under R version 3.5.2
```

```
## Loading required package: lattice
```

```
##
## Attaching package: 'caret'
```

```
## The following object is masked from 'package:purrr':
##
##     lift
```

```
set.seed(2)
test_index <- createDataPartition(y = dataCA$category_id, times = 1, p = 0.2, list = FALSE)
test_set <- dataCA[test_index,]
train_set <- dataCA[-test_index,]
```

Linear Regression Modeling

Model 1 is taking a guessing approach by taking average to make prediction. Model 2 is using number of views to predict number of likes. Model 3 is using number of views and comments to predict number of likes. Model 4 is using number of views, comments and dislikes to predict number of likes. Model 5 is using number of views, comments, dislikes and category ID to predict number of likes.

```
model_1 <- mean(train_set$likes)
y_hat_1 <- mean(train_set$likes)
model_2 <- lm(likes ~ views, data = train_set)
y_hat_2 <- predict(model_2, test_set)
model_3 <- lm(likes ~ views + comment_count, data = train_set)
y_hat_3 <- predict(model_3, test_set)
model_4 <- lm(likes ~ views + comment_count + dislikes, data = train_set)
y_hat_4 <- predict(model_4, test_set)
model_5 <- lm(likes ~ views + comment_count + dislikes + factor(category_id), data = train_set)
y_hat_5 <- predict(model_5, test_set)
```

Model Assessment

Author is using RMSE to assess the model.

```
RMSE <- function(true_likes, predicted_likes){
  sqrt(mean((true_likes - predicted_likes)^2))
}
```

```
RMSE_1 <- RMSE(test_set$likes, y_hat_1)
RMSE_2 <- RMSE(test_set$likes, y_hat_2)
RMSE_3 <- RMSE(test_set$likes, y_hat_3)
RMSE_4 <- RMSE(test_set$likes, y_hat_4)
RMSE_5 <- RMSE(test_set$likes, y_hat_5)

rmse_results <- tibble(Method = c("Method 1",
                                  "Method 2",
                                  "Method 3",
                                  "Method 4",
                                  "Method 5"),
                       RMSE = c(RMSE_1,
                                RMSE_2,
                                RMSE_3,
                                RMSE_4,
                                RMSE_5),
                       Predictor = c("Naive Average",
                                     "views",
                                     "views_N_comments",
                                     "views_N_comments_N_dislikes",
                                     "views_N_comments_N_dislikes_N_categoryID"))
rmse_results
```

```
## # A tibble: 5 x 3
##   Method      RMSE Predictor
##   <chr>      <dbl> <chr>
## 1 Method 1 146091. Naive Average
## 2 Method 2  82123. views
## 3 Method 3  58320. views_N_comments
## 4 Method 4  49888. views_N_comments_N_dislikes
## 5 Method 5  48215. views_N_comments_N_dislikes_N_categoryID
```

It shows from the modelling above that the four variables number of views, comments, dislikes and category ID does provide significant improvements in predicting number of likes.

This shows that by targeting number of views and comments, in a specific category and avoiding dislikes, we could somehow and somewhat predict the number of likes a YouTuber could earned.

Validation

In this part, author is using 10% of the data set to validate the algorith.

```
set.seed(13)
val_index <- createDataPartition(y = dataCA$category_id, times = 1, p = 0.1, list = FALSE)
val_set <- dataCA[test_index,]
train_set_val <- dataCA[-test_index,]

model_val <- lm(likes ~ views + comment_count + dislikes + factor(category_id), data = train_set)
y_hat_val <- predict(model_val, test_set)
RMSE_val <- RMSE(val_set$likes, y_hat_val)
RMSE_val
```

```
## [1] 48215.46
```

The result is the same. This means this algorithm is consistent. Thus, in the context of YouTube videos in

Canada, we could predict the number of likes by controlling variables such as number of views, comments, dislikes and category ID. This approach could also be useful for YouTuber to measure the engagement rate between their contents and the viewers.

However, this approach also have limitations such as the scope of the sample acquired from for this assignment for a particular period and for the context of Canada only. Further, This assignment is using multiple linear regression only and can be further developed to a more sophisticated models such as KNN, Regularization, etc. Also, others approach to develop the data set such as log transformation, etc. And most importantly Author must deepen the knowledge how to construct "the right question" that is useful for business that can be answered through statistical domain and calculated with the right approach using R Language.