Classification

Nikilas John

September 25th, 2022

Overview of Logistic Regression

Logistic Regression (better known as classification) is the way we can classify and predict qualititative values. Most of the time they will be of binary nature.

Load the data

Import the USvideos.csv data set.This data set has 15424 observations of 16 variables This data set comes from "https://www.kaggle.com/datasets/datasnaek/youtube-new"

```
df <- read.csv("USvideos.csv")
```

Train/Test Split

Split the data into Train and Test

```
i <- sample(1:nrow(df), nrow(df)*0.8, replace=FALSE)
train <- df[i,]
test <- df[-i,]
```

Data Exploration

Since we are using the same data set as the Regression notebook, we will use different exploration functions

This command below shows the last ten rows of the train$video_id column

```
tail(train$video_id, n=10)
```

```
##  [1] "XkVwqEdYlNA" "QwZT7T-TXT0" "iHcfeuGYuzU" "aPrpWFl2eik" "NHrwcQQ38bA"
##  [6] "l_lblj8Cq0o" "B450jhJ5vHg" "mB2UTIkgY1I" "8HTdbCohTXY" "HJqALBKE4r4"
```

Running head()

Running head() will displayu the first 6 rows of the train$title column

```
head(train$title)
```

```
## [1] "Sergei Eisenstein the Father of Montage"
## [2] "Andre The Giant Official Trailer (2018) | HBO"
## [3] "Stranger Things Cast Answer the Web's Most Searched Questions |
WIRED"
## [4] "How To Enlarge And Correct Lip Shape | John Maclean"
```

```
## [5] "Pie - Cyanide & Happiness Shorts"
## [6] "BEST OF BEAUTY 2017 | Jaclyn Hill"
```

Finding the number of NAs using a differnt command

This time we will use a combination of the colSum() and is.na() functions to display the number of NAs per column

```
colSums(is.na(train))
```

```
##                video_id          trending_date                  title
##                       0                      0                      0
##          channel_title            category_id           publish_time
##                       0                      0                      0
##                    tags                  views                  likes
##                       0                      0                      0
##                dislikes          comment_count         thumbnail_link
##                       0                      0                      0
##      comments_disabled       ratings_disabled video_error_or_removed
##                       0                      0                      0
##            description
##                       0
```

Finding the average length of a description

Lets find out the average length of a YouTube trending video using the train data set

The average length should be 976 characters (rounded up since you cannot have 3/4 of a character)

```
mean(nchar(train$description))
```

```
## [1] 960.2439
```

Whats the median of tags

Lets see what the median value of the tags column is in the training data

Based on these tags, the video is most likely a country music video that came out in 2017

```
median(train$tags)
```

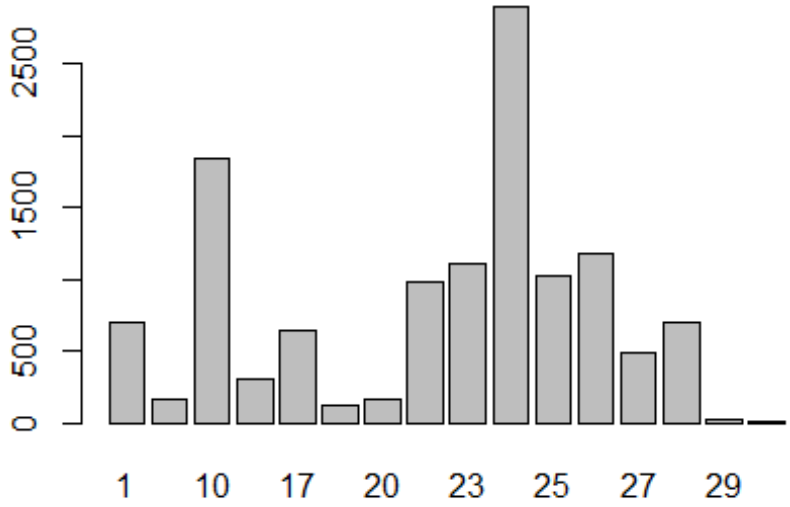
```
## [1] "keaton jones|\"anti bullying\"|\"keaton jones bullying video\"|\"us
news\"|\"us\"|\"usa news\"|\"tennessee\"|\"tennessee news\"|\"keaton jones
chris evans video\"|\"keaton\"|\"keaton bullying\"|\"keaton bullying
video\"|\"bullied boy
video\"|\"bullies\"|\"bully\"|\"bullying\"|\"athletes\"|\"ugly\"|\"nose\"|\"f
riends\"|\"milk\"|\"keaton jones video\"|\"keaton jones facebook\"|\"keaton
jones bullying\"|\"usa\"|\"america\"|\"united
states\"|\"school\"|\"playground
bullying\"|\"kids\"|\"children\"|\"child\"|\"childhood\"|\"guardian\"|\"2017\
""
```

Category Bar Graph

There is also a numerical vector for the category_id, so lets see if the trending page has a bias towards one category or another.

As seen by the bar plot, there are a lot of videos in category 10 and 24 that have ended up on the trending page. These categories are “Comedy” and “Family” respectively (these can be found in the US_category_id.json file)

```
barplot(table(train$category_id))
```

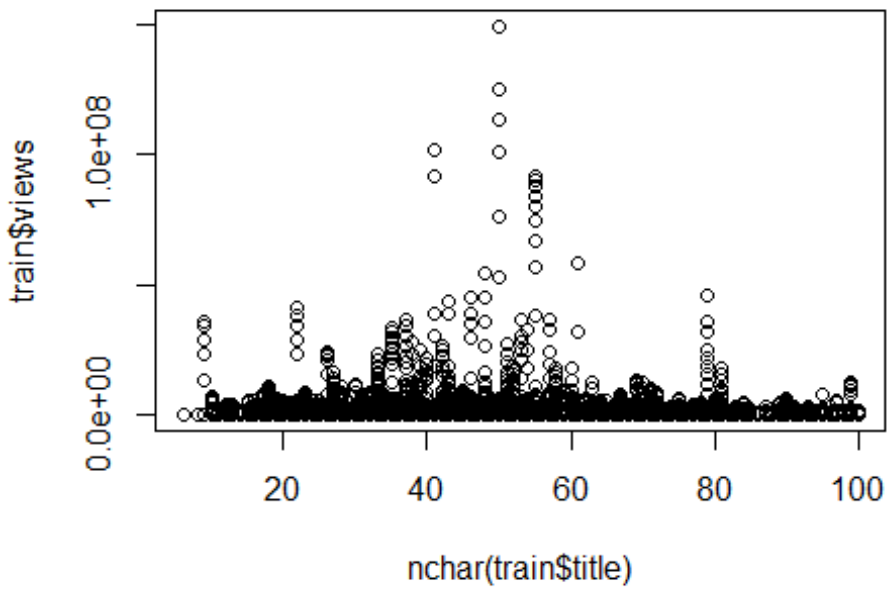


Length of Title vs Views

Lets plot the length of the title against the views to see if there is a correlation between them

Oddly enough, it seems like there is almost a bell curve with the outlying cases, but overall, the length of the title almost has no bearing on how many views it gets. Almost all of the trending videos have the same number of views

```
plot(nchar(train$title), train$views)
```



Logistic Regression Model

In this section, we will use logistic regression to see if we can predict when a video has comments disabled using dislikes as a basis

What we can summarize through this output very simply is that dislikes is NOT a good indicator on if the comments are disabled or not.

I was able to draw this conclusion through many factors, some of them including: 1. There were no stars given to the dislikes predictor by R 2. The null and residual deviance are almost the same value 3. The AIC is very high

```
glm1 <- glm(comments_disabled~dislikes, data=train, family=binomial)
summary(glm1)
```

```
##
## Call:
## glm(formula = comments_disabled ~ dislikes, family = binomial,
##     data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.1833  -0.1833  -0.1833  -0.1832   2.8836
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -4.078e+00  7.080e-02 -57.600   <2e-16 ***
```

```
## dislikes    -1.086e-06  3.254e-06  -0.334    0.738
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2086.6  on 12338  degrees of freedom
## Residual deviance: 2086.4  on 12337  degrees of freedom
## AIC: 2090.4
##
## Number of Fisher Scoring iterations: 7
```

Naive Bayes Model

Here we apply the Naive Bayes algorithm to the same column (comments_disabled) with dislikes.

The A-priori probabilities displayed by R are FALSE: 0.982 and TRUE: 0.018

The Conditional Probabilities section looks a little different because the dislikes column is a continuous variable. We can see that the mean number of dislikes that were required to have the comments not disabled were 3.5k and for the comments to be disabled, the count was 2.2k. The standard deviation is extremely large for the number of dislikes as well, being 46k for the comments not being disabled.

```
library(e1071)
nb1 <- naiveBayes(comments_disabled~dislikes, data=train)
nb1

##
## Naive Bayes Classifier for Discrete Predictors
##
## Call:
## naiveBayes.default(x = X, y = Y, laplace = laplace)
##
## A-priori probabilities:
## Y
##      FALSE       TRUE
## 0.98338601 0.01661399
##
## Conditional probabilities:
##        dislikes
## Y           [,1]      [,2]
##   FALSE 3223.798 41554.214
##   TRUE  2204.054  8562.023
```

Evaluate on Test Data

Here we will evaluate our algorithms using the test data, then compare their results

Logistic Regression Test

We can see that the accuracy of prediction to see if the comments are disabled are 98.2%, meaning that our algorithm is extremely accurate and can be used as a precise predictor

```
probs <- predict(glm1, newdata=test, type="response")
pred <- ifelse(probs>0.5, TRUE, FALSE)
acc <- mean(pred==test$comments_disabled)
print(paste("Accuracy = ", acc))

## [1] "Accuracy =  0.983468395461912"

table(pred, test$comments_disabled)

##
## pred    FALSE TRUE
##   FALSE  3034   51
```

Naive Bayes Test

With the Naive Bayes algorithm, we can see that the accuracy is worse than the logistical regression one with an accuracy of 98%.

```
p1 <- predict(nb1, newdata=test, type="class")
acc <- mean(p1==test$comments_disabled)
print(paste("Accuracy: ", acc))

## [1] "Accuracy:  0.982495948136143"

table(p1, test$comments_disabled)

##
## p1      FALSE TRUE
##   FALSE  3031   51
##   TRUE      3    0
```

Strength and Weaknesses of Logistic Regression and Naive Bayes

In this section we will detail the strengths and weaknesses of Logical Regression and Naive Bayes

Logistic Regression

Strengths: 1. Will seperate classes well ONLY when they are able to be linearly separable 2. In regards to computational power, it is a very easy algorithm to run 3. Has a nice, probabilistic output

Weaknesses: 1. Tends to under fit patterns because it is not able to show very elaborate non-linear decision boundaries

Naive Bayes

Strengths: 1. Works best on smaller data sets 2. Easy to implement 3. Easy to understand 4. Handles high amounts of variables well

Weaknesses: 1. Does not work as well as other classifiers on large amounts of data 2. The algorithm guesses a value in the test set that were not in the training data 3. The predictors have to be independent or else the algorithm's performance is less than it could be

Classification Metrics

Here we will go over the classification metrics and their benefits/downsides

Accuracy

This metric is the most simple of them all, displaying the percentage of predictions that were correct

Sensitivity and Specificity

Sensitivity will measure the rate of true positives

Specificity will measure the rate of true negatives

Kappa

Kappa is a more specific metric than accuracy because it accounts for the correct prediction by chance, leading to a better metric of how good the classification is

ROC and AUC

ROC shows us how the true positive relates to the false positive, usually ending up in the shape of a curve. The AUC is the area under the curve of the ROC that ranges between [0.5, 1] with 1 being a perfect classifier

MCC

MCC stands for Matthew's correlation coefficient and will only work for binary classification. The range for the values in MCC are [-1, 1]. This is more accurate than the accuracy metric because it accounts for the class distribution