Twitter Sentiment Analysis

Eli (Ilya) Bolotin

11/18/2018

Introduction

This file is a walkthrough of a statistical experiment described in the project summary ('Project_Summary.pdf').

This experiment aims to answer the question:

- Are people that are vocal about their physical activity happier than people who are vocal about their media consumption? The same question phrased differently: are people that talk about fitness happier than people who talk about media (tv, movies, internet videos, etc.)?
  - **Null hypothesis**: the happiness of people that talk about fitness is not different from the happiness of people that talk about media.
  - **Alternate hypothesis**: the happiness of people that talk about fitness is (statistically) different from the happiness of people that talk about media.

To answer this question, we have performed sentiment analysis on Twitter data. Two sample groups have been collected (using Python and Twitter's API), cleaned, and sentiment calculated using the methdology described in the project summary. These sample groups of tweets are stored in distinct CSV files. They are ready to be read-into R. Let's begin the statistical analysis.

Load libraries

First, load necessary libraries for analysis.

```r
knitr::opts_chunk$set()
library(dplyr)
library(ggplot2)
```

This data has been procured and cleaned in Python and is ready for analysis. We are only concerned with the sentiment scores produces by NLTK's VADER algorithm (and not those from TextBlob, which have also been computed in this data).

1. Read in data

Read in data for both groups: tweets of people who talk about fitness and people who talk about media.

```r
# IMPORTANT: change working dir
setwd('Samples_Round_1/')

# Read in csvs of both sample groups
group_1 <- read.csv("streamed_tweets_fitness_clean_full_analysis.csv", header=TRUE,
↪  sep=",")
group_2 <- read.csv("streamed_tweets_media_clean_full_analysis.csv", header=TRUE,
↪  sep=",")
```

2. Label groups & exclude users

Next, we need to two things. The first is to label each group. The second is to exclude users that appear in both groups.

```r
# Add column to both datasets to label each group and convert group to factor
group_1 = mutate(group_1, group = "fitness")
group_1 = mutate(group_1, group = as.factor(group))
group_2 = mutate(group_2, group = "media")
group_2 = mutate(group_2, group = as.factor(group))

# Create list to exclude users that appear in both lists
excluded_users <- list()

i = 1
for(user_id in group_1$user_id) {
  if(user_id %in% group_2$user_id) {
    excluded_users[[i]] <- user_id
    i = i + 1
  }
}

excluded_users <- unique(excluded_users)

# Exclude excluded users from samples
group_1 <- group_1[!(group_1$user_id %in% excluded_users),]
group_2 <- group_2[!(group_2$user_id %in% excluded_users),]
```

3. Select sample size

Now, we need to ensure that our sample sizes are the same for both groups. The problem is that these groups have different numbers of tweets, and thus the samples vary. But we need the sample size to be the same so that we can accurately gauge sentiment. To do this, we will generate random numbers specific to the tweet indices of each group, and then use those random `N` indices of each group. These `N` indices (randomly generated for each group) will constitute our sample size of `N` tweets. Below, we first set the sample size.

```r
# Select sample sizes and possible ranges to generate indices
n <- 9700
group_1_index_range <- nrow(group_1)
group_2_index_range <- nrow(group_2)

# For group 1: generate indices of random sample
set.seed(1)
sample_indices_group_1 <- sample(1:group_1_index_range, n, replace = FALSE)
sample_indices_group_1 <- sort(sample_indices_group_1)

# For group 2: generate indices of random sample
sample_indices_group_2 <- sample(1:group_2_index_range, n, replace = FALSE)
sample_indices_group_2 <- sort(sample_indices_group_2)

# Select tweets by sample index
group_1 <- group_1[sample_indices_group_1, c("group","sentiment_vader")]
group_2 <- group_2[sample_indices_group_2, c("group","sentiment_vader")]
```

4. Combine groups into dataframe

Next, combine both groups into a single dataframe for analysis.

```
# Combine groups into one dataframe
combined_df = rbind(group_1, group_2)
```

5. Verify sample sizes

With the data prepared - verify the group sizes for both samples.

```
group_size <- combined_df %>%
    group_by(group) %>%
    dplyr::summarize(
      group_size = n()
      )

group_size
```

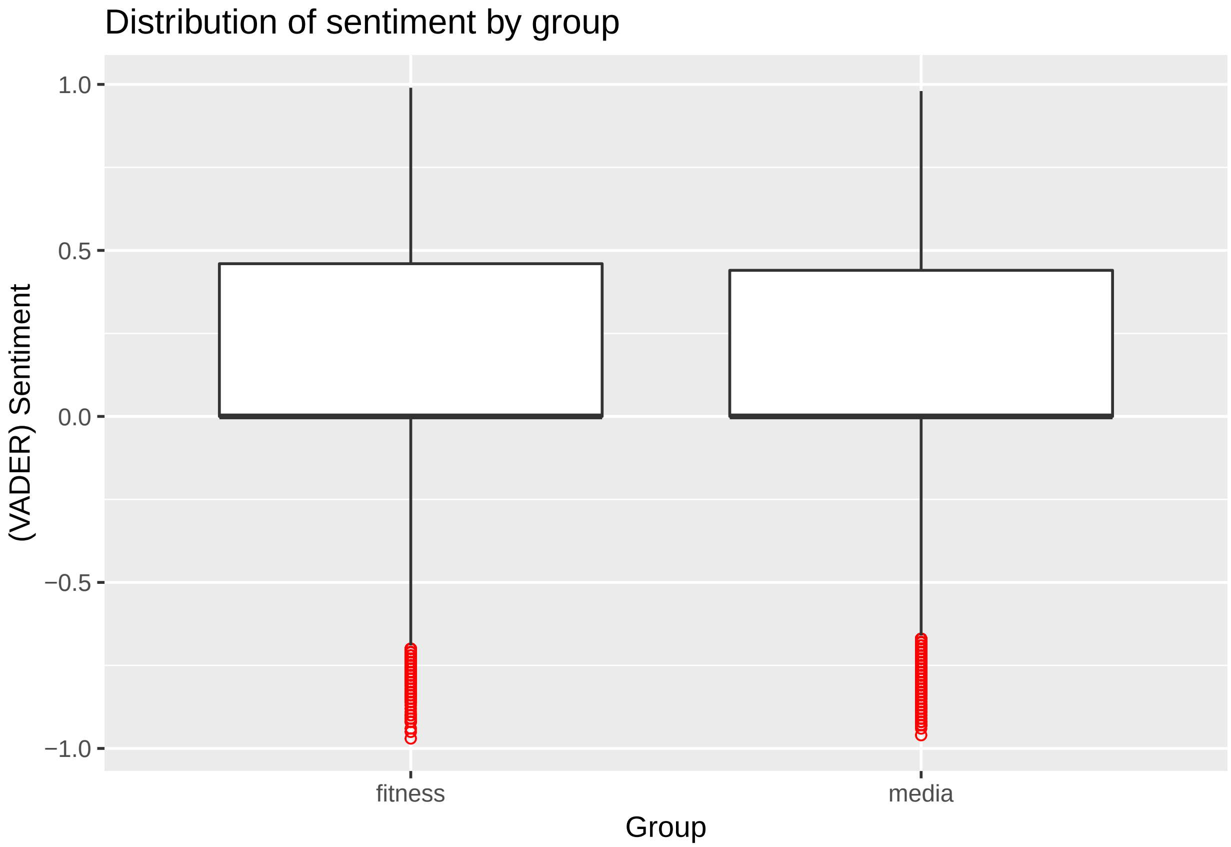
```
## # A tibble: 2 x 2
##   group   group_size
##   <fct>        <int>
## 1 fitness       9700
## 2 media         9700
```

We have verified that our samples match in size. Next, we need to review the sample distributions.

6. View distributions

Plot boxplot of sentiment scores to see distribution stats.

```
ggplot(combined_df, aes(x = group, y = sentiment_vader)) +
geom_boxplot(outlier.colour = "red", outlier.shape = 1) +
  ggtitle("Distribution of sentiment by group") + xlab("Group") + ylab("(VADER)
↪   Sentiment")
```



The distributions of both groups are also quite similar. However, the fitness group has a higher interquartile range. This means that the fitness group has tweets with a higher sentiment than the media group at the 3rd quartile. Another observation is that the first quartile (for both groups) ends nearly on the median. This tells us that a majority of the sentiments less than zero are very close to zero. Meaning, the spread is skewed towards a sentiment of zero.

7. Group sentiment distributions

However, the boxplot above does not give us much more information because the median for both groups appears to be the same. Let's review summary statistics of both groups to get a sense of scale.

```
summary_stats <- combined_df %>%
    group_by(group) %>%
    dplyr::summarize(
      group_size = n(),
      equal_to_0 = length(sentiment_vader[sentiment_vader == 0]),
      above_0 = length(sentiment_vader[sentiment_vader > 0]),
      below_0 = length(sentiment_vader[sentiment_vader < 0])
      )

summary_stats
```

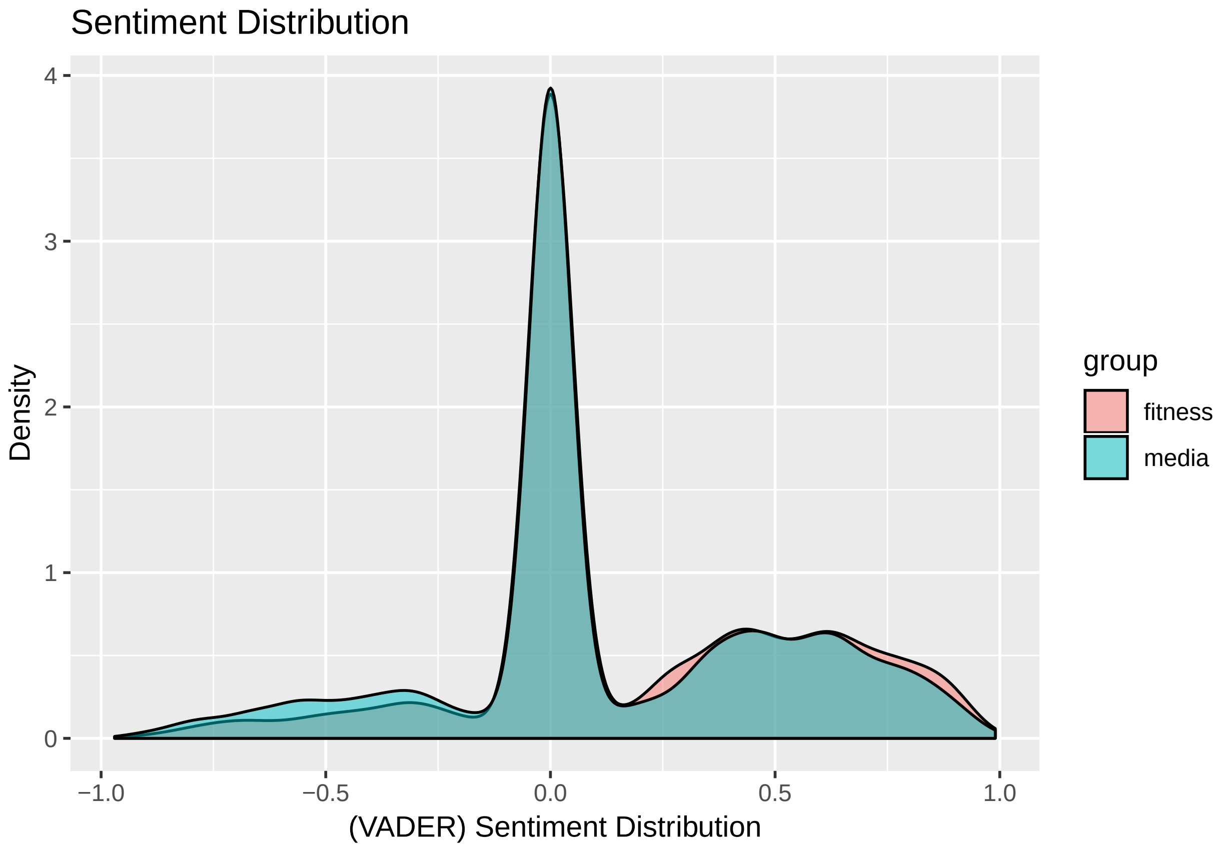
```
## # A tibble: 2 x 5
##   group   group_size equal_to_0 above_0 below_0
##   <fct>        <int>      <int>   <int>   <int>
## 1 fitness       9700       4566    4058    1076
## 2 media          9700        4407     3734     1559
```

We notice that for the fitness group, ~89% of tweets are equal to or above 0 (neutral) sentiment. For the media group, this number is ~84%. The conclusion for both groups is that tweets are overwhelmingly neutral and/or positive in sentiment.

8. Density plot and histogram

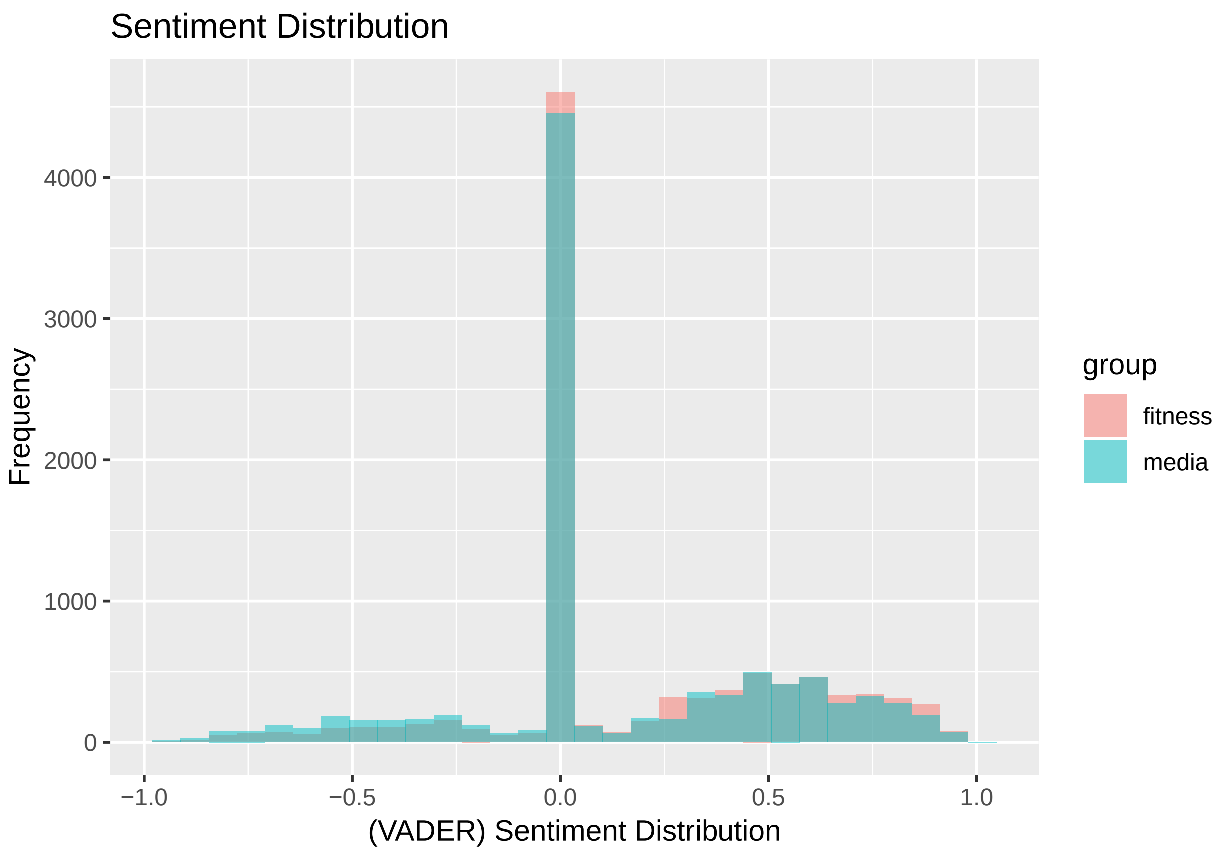
Next, let's visualize the distribution above. We can do this by plotting a density plot and histogram of the (VADER) sentiment scores to see the frequency of sentiments.

```
# Density plots
qplot(sentiment_vader, data=combined_df, geom="density", fill=group, alpha=I(.5),
   main="Sentiment Distribution", xlab="(VADER) Sentiment Distribution",
   ylab="Density")
```



```
# Histogram
ggplot(combined_df, aes(sentiment_vader, fill = group)) +
  geom_histogram(position="identity", alpha = .5) + ggtitle("Sentiment Distribution") +
↪  xlab("(VADER) Sentiment Distribution") + ylab("Frequency")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



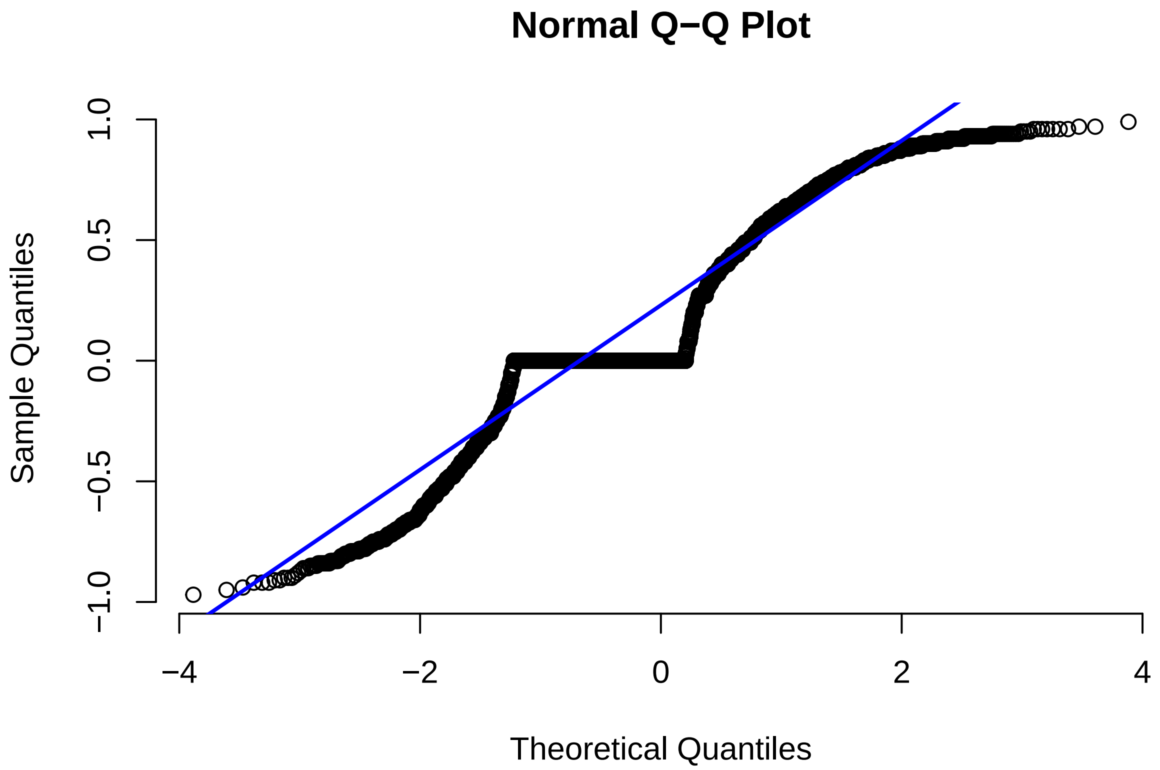
The density and histogram plots tell us a few things clearly: 1. The sentiments for both groups do not quite follow a normal distribution **at this sample size**. 2. Both groups have a very similar distribution of sentiments. Meaning, people in both groups tend to express themselves similarly in sentiment. 3. Another insight is that the most common sentiment is neutral (equal to 0), or close to it. 4. There are **more** positive sentiments (greater than 0) than there are negative sentiments. 5. The fitness group has a higher frequency of positive sentiments than the media group at nearly every level of sentiment above ~0.2. 6. The media group has a higher frequency of negative sentiment than the fitness group.

9. QQ Plot: Check distribution type

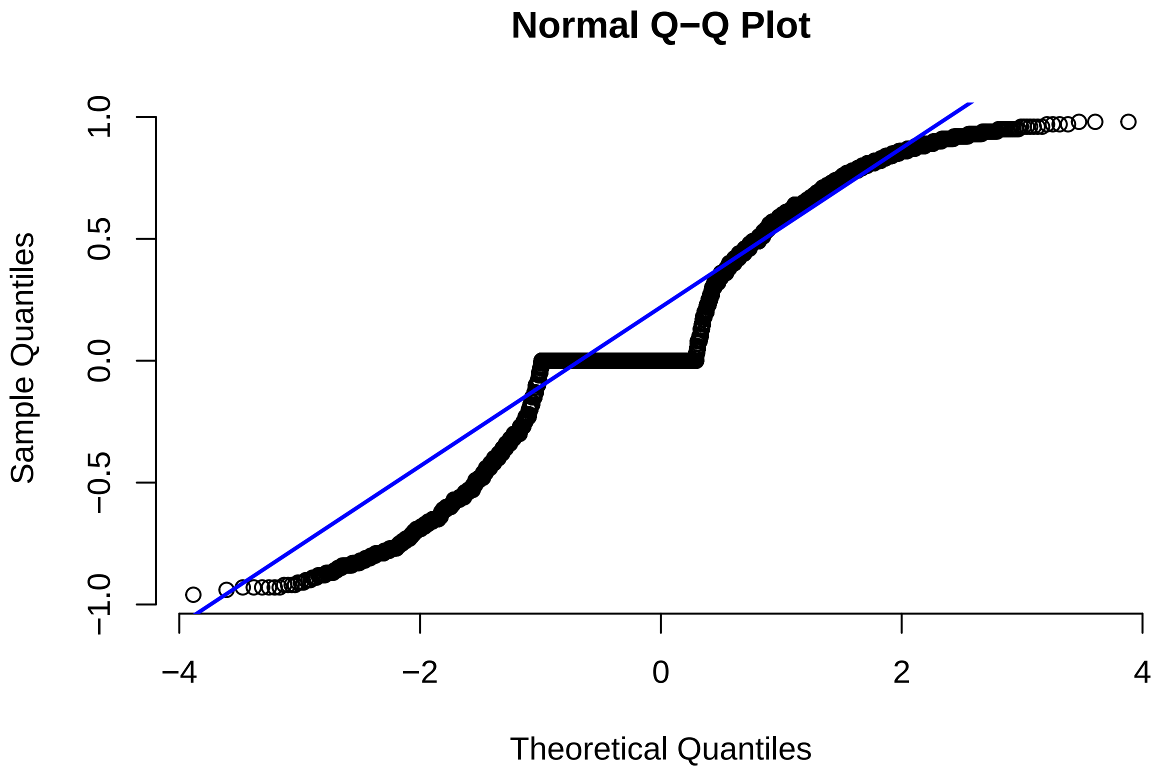
Given the density distribution, let's check if both samples follow a normal distribution:

```r
# Define groups
fitness_group = combined_df$sentiment_vader[combined_df$group == 'fitness']
media_group = combined_df$sentiment_vader[combined_df$group == 'media']

# Q-Q Plot for fitness group
qqnorm(fitness_group, pch = 1, frame = FALSE)
qqline(fitness_group, col = "blue", lwd = 2)
```



```
# Q-Q plot for media group
qqnorm(media_group, pch = 1, frame = FALSE)
qqline(media_group, col = "blue", lwd = 2)
```



The Q-Q plot confirms that the sentiment from our sample groups does not follow a normal distribution. This means that parametric statistics tests such as the T-Test may not be ideally suited to determine if one group is statistically different from another. This is because parametric statistics tests make assumptions about "about the parameters (defining properties) of the population distribution from which one's data are drawn" [1]. Because we cannot assume the data to be symmetrical about the mean (or equal variance, among other differences), it is probably not ideal to depend on these parametric tests.

[1] Parametric vs Non-parametric, http://vassarstats.net/textbook/parametric.html

10. Compute summary statistics

Next we need to compute summary statistics for both groups to determine key metrics.

```
# Compute statistical summary for both groups
stat_summary <- combined_df %>%
    group_by(group) %>%
    dplyr::summarize(
      group_size = n(),
      median_sentiment = median(sentiment_vader),
      mean.sentiment = mean(sentiment_vader),
      sd.sentiment = sd(sentiment_vader),
      se.mean.sentiment = sd.sentiment / sqrt(group_size)
      )

stat_summary
```

```
## # A tibble: 2 x 6
##   group group_size median_sentiment mean.sentiment sd.sentiment
##   <fct>      <int>            <dbl>          <dbl>        <dbl>
## 1 fitn~       9700                0          0.177        0.364
## 2 media       9700                0          0.133        0.382
## # ... with 1 more variable: se.mean.sentiment <dbl>
```

The median sentiment between both groups appears to be the same (0). However, the mean sentiment appears statistically different. SD and SE are also different between both samples.

11. Welch's T-Test

Although Welch's test is a parametric test (and thus not ideally suited for this hypothesis test), let's run this test as a point of reference to determine if the means of both groups are significantly different.

```
t.test <- t.test(sentiment_vader ~ group, var.equal = FALSE, data = combined_df)
t.test
```

```
##
##  Welch Two Sample t-test
##
## data:  sentiment_vader by group
## t = 8.1143, df = 19354, p-value = 5.176e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.03298353 0.05399379
## sample estimates:
## mean in group fitness   mean in group media
##             0.1768639             0.1333753
```

The T-test confirms that the mean's of both groups are statistically significant. This would suggest that we can reject the null hypothesis that people that talk about fitness are equally happy as people that talk about media.

However, can we trust this test given that our small sample sizes and the fact that the data does not follow a standard normal distribution?

12. Non-parametric T-Test: Mann-Whitney U Test

To find out how reliable this conclusion is, let's run the Mann-Whitney U test (also known as the Wilcox test). It is a nonparametric test of the null hypothesis designed to account for non-normal distributions.

```
wilcox.test <- wilcox.test(sentiment_vader ~ group, data = combined_df, conf.int=TRUE,
↪  alternative = "greater")
wilcox.test
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  sentiment_vader by group
## W = 49932000, p-value = 3.169e-15
## alternative hypothesis: true location shift is greater than 0
## 95 percent confidence interval:
##  8.03616e-06         Inf
## sample estimates:
## difference in location
##           8.697598e-05
```

The Wilcox test confirms the T-Test of unequal variances, leading us to conclude that the NULL hypothesis should be rejected and that there is a significant difference in happiness between individuals that discuss fitness and those that discuss media.

13. Kruskal-Wallis test

Another nonparametric test that we can use to check whether our samples originate from the same distribution is the Kruskal-Wallis test [2], which compares/analyzes the medians of two independent samples. Being nonparametric, this test does not assume that our samples have equal variance or are normally distributed. The KW analysis will do well to tell us if the medians of both samples are statistically significantly different, and thus if we can reject the null hypothesis.

[2] Kruskal-Wallis Test, https://en.wikipedia.org/wiki/Kruskal\T1\textendashWallis_one-way_analysis_of_variance

```
k.test <- kruskal.test(sentiment_vader ~ group, data = combined_df)
k.test
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  sentiment_vader by group
## Kruskal-Wallis chi-squared = 60.794, df = 1, p-value = 6.339e-15
```

In this test, the p-value is smaller than our significance level of 0.05. As a result, we must reject our NULL hypothesis that people that discuss fitness are equally happy as people that discuss media.

Conclusion

Reject Null Hypothesis