NYC Group TED Talks Review Study

Used kernel : https://www.kaggle.com/mikaelhuss/r-clone-of-ted-data-analys scriptVersionId=1614520

```
month_order = c('Jan', 'Feb', 'Mar', 'Apr', 'May', 'Jun', 'Jul', 'Aug', 'Sep', 'Oct', 'Nov', 'Dec')
day_order = c('Mon', 'Tue', 'Wed', 'Thu', 'Fri', 'Sat', 'Sun')
```

```
df <- read_csv("C:/AT/tedtalks/ted_main.csv")
```

```
## Parsed with column specification:
## cols(
##   comments = col_integer(),
##   description = col_character(),
##   duration = col_integer(),
##   event = col_character(),
##   film_date = col_integer(),
##   languages = col_integer(),
##   main_speaker = col_character(),
##   name = col_character(),
##   num_speaker = col_integer(),
##   published_date = col_integer(),
##   ratings = col_character(),
##   related_talks = col_character(),
##   speaker_occupation = col_character(),
##   tags = col_character(),
##   title = col_character(),
##   url = col_character(),
##   views = col_integer()
## )
```

```
colnames (df)
```

```
##  [1] "comments"           "description"        "duration"
##  [4] "event"              "film_date"          "languages"
##  [7] "main_speaker"       "name"               "num_speaker"
## [10] "published_date"     "ratings"            "related_talks"
## [13] "speaker_occupation" "tags"               "title"
## [16] "url"                "views"
```

```
# For reordering data
df = df[, c('name', 'title', 'description', 'main_speaker', 'speaker_occupation', 'num_speaker', 'durat

# For converting unix dates
df$film_date = anydate(df$film_date)
df$published_date = anydate(df$published_date)
head(df)
```

```
## # A tibble: 6 x 17
##                                              name
##                                           <chr>
## 1     Ken Robinson: Do schools kill creativity?
## 2          Al Gore: Averting the climate crisis
## 3                 David Pogue: Simplicity sells
## 4             Majora Carter: Greening the ghetto
## 5 Hans Rosling: The best stats you've ever seen
## 6             Tony Robbins: Why we do what we do
## # ... with 16 more variables: title <chr>, description <chr>,
## #   main_speaker <chr>, speaker_occupation <chr>, num_speaker <int>,
## #   duration <int>, event <chr>, film_date <date>, published_date <date>,
## #   comments <int>, tags <chr>, languages <int>, ratings <chr>,
## #   related_talks <chr>, url <chr>, views <int>
```

```r
nrow(df)
```

```
## [1] 2550
```

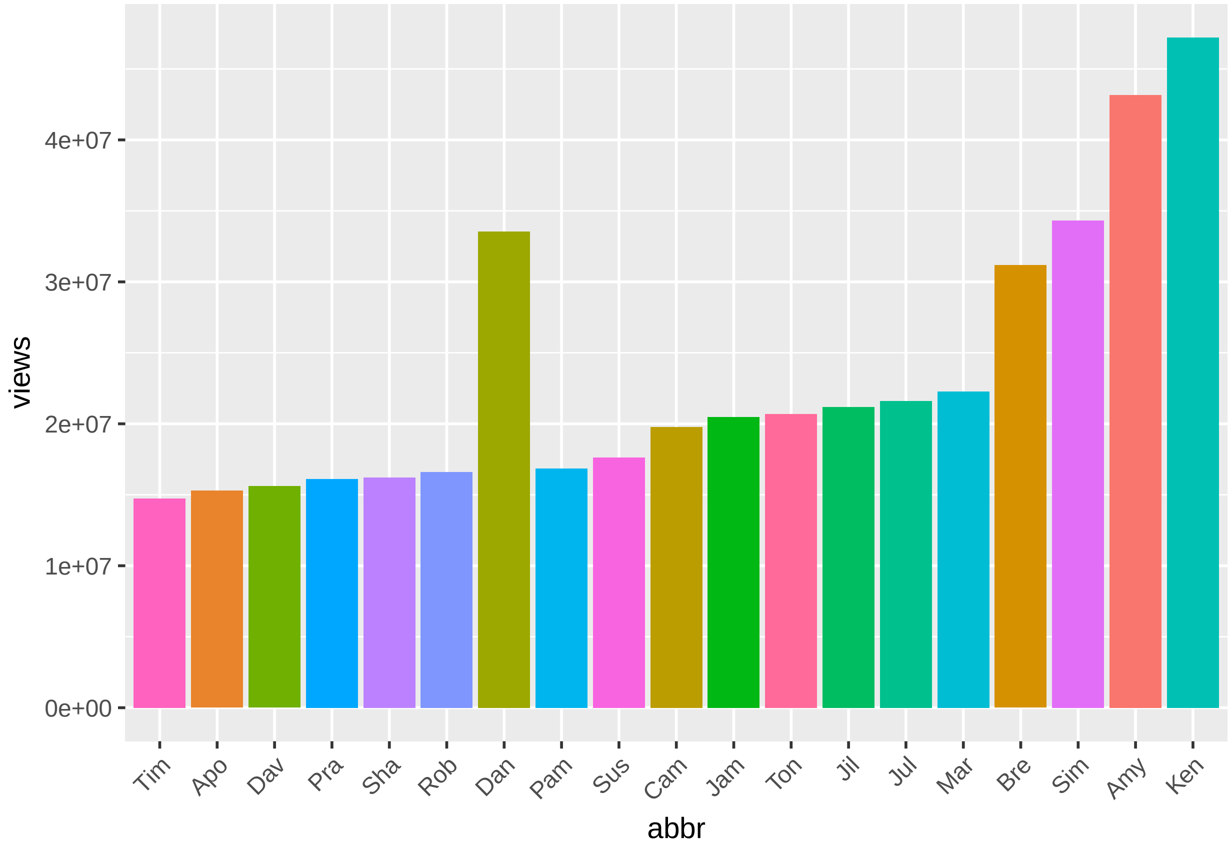
For displaying the top 20 viewed talks

```r
pop_talks = df[, c("title", "main_speaker", "views", "film_date")] %>% arrange(desc(views)) %>% head(20
pop_talks
```

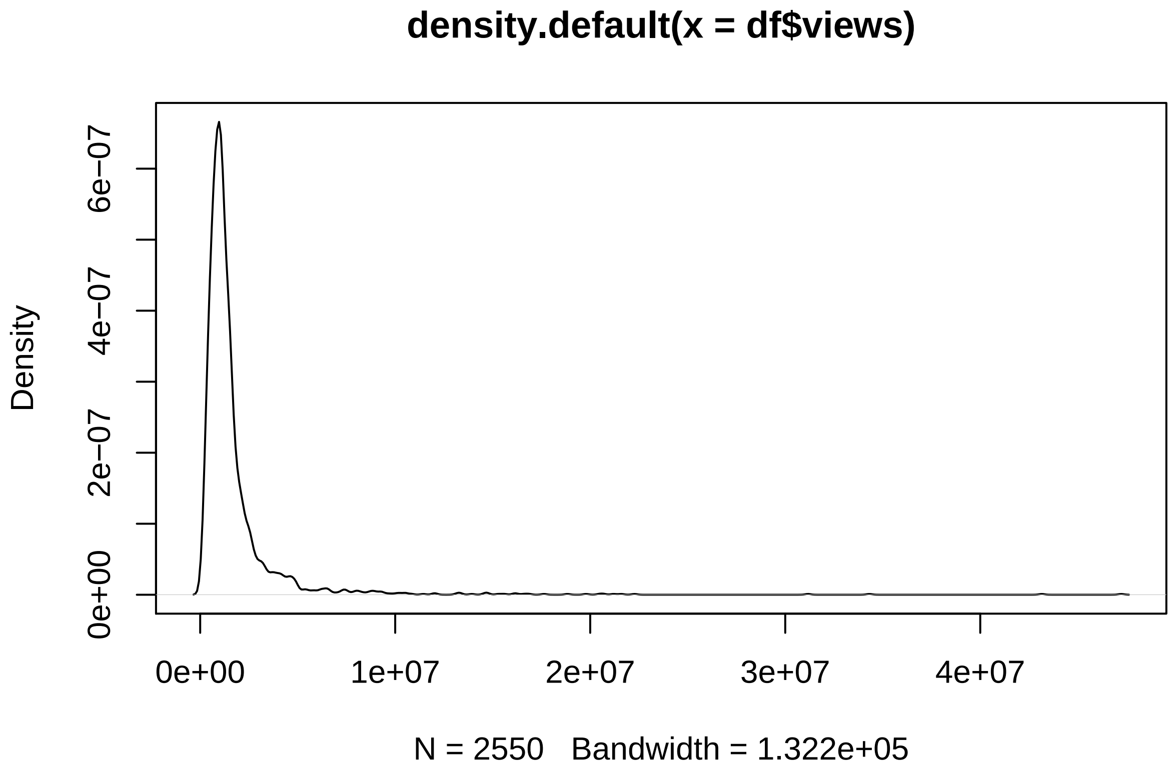
```
## # A tibble: 20 x 4
##                                                                  title
##                                                                  <chr>
##  1                                         Do schools kill creativity?
##  2                              Your body language may shape who you are
##  3                                      How great leaders inspire action
##  4                                          The power of vulnerability
##  5                                 10 things you didn't know about orgasm
##  6                               How to speak so that people want to listen
##  7                                                 My stroke of insight
##  8                                                   Why we do what we do
##  9                         This is what happens when you reply to spam email
## 10                        Looks aren't everything. Believe me, I'm a model.
## 11                                              The puzzle of motivation
## 12                                              The power of introverts
## 13                                                    How to spot a liar
## 14 What makes a good life? Lessons from the longest study on happiness
## 15                                        The happy secret to better work
## 16                         The thrilling potential of SixthSense technology
## 17                                        How I held my breath for 17 minutes
## 18                                              The art of misdirection
## 19                               Inside the mind of a master procrastinator
## 20                                        The surprising science of happiness
## # ... with 3 more variables: main_speaker <chr>, views <int>,
## #   film_date <date>
```

For visualizing the top 20 viewed talks

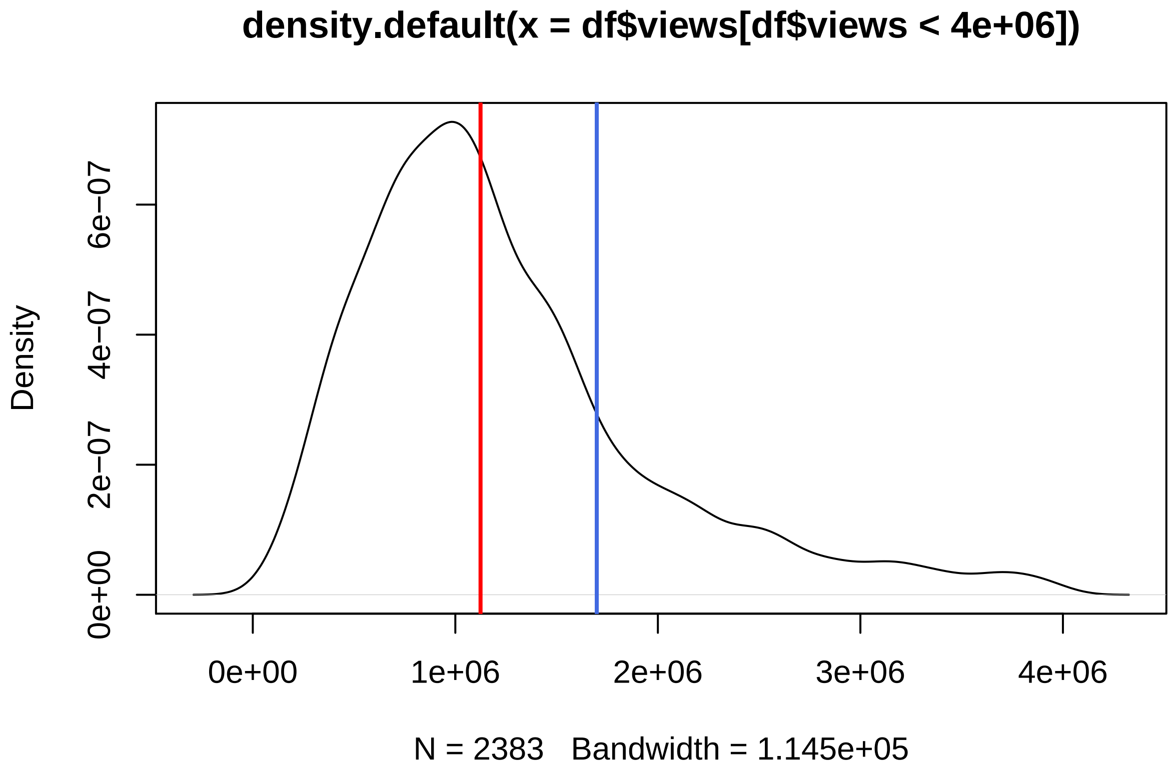
```r
pop_talks$abbr <- pop_talks$main_speaker %>% as.character() %>% substr(0,3)
ggplot(pop_talks, aes(x=reorder(abbr, views), y=views, fill=abbr)) +
  geom_bar(stat = 'identity') +
  guides(fill=FALSE) +
  labs(x="abbr") + theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust=1))
```



```r
# For adding the histogram
plot(density(df$views))
```



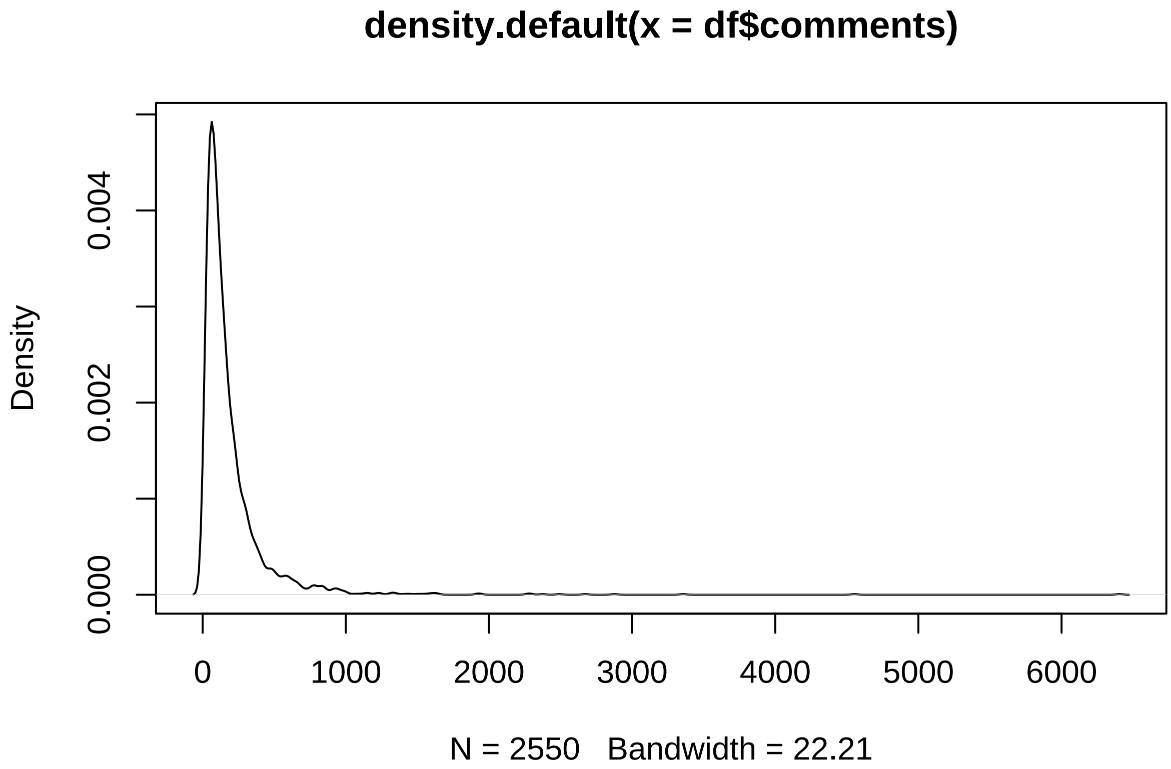
```r
plot(density(df$views[df$views<0.4e7]))
abline(v = mean(df$views),
 col = "royalblue",
 lwd = 2)
abline(v = median(df$views),
 col = "red",
 lwd = 2)
```



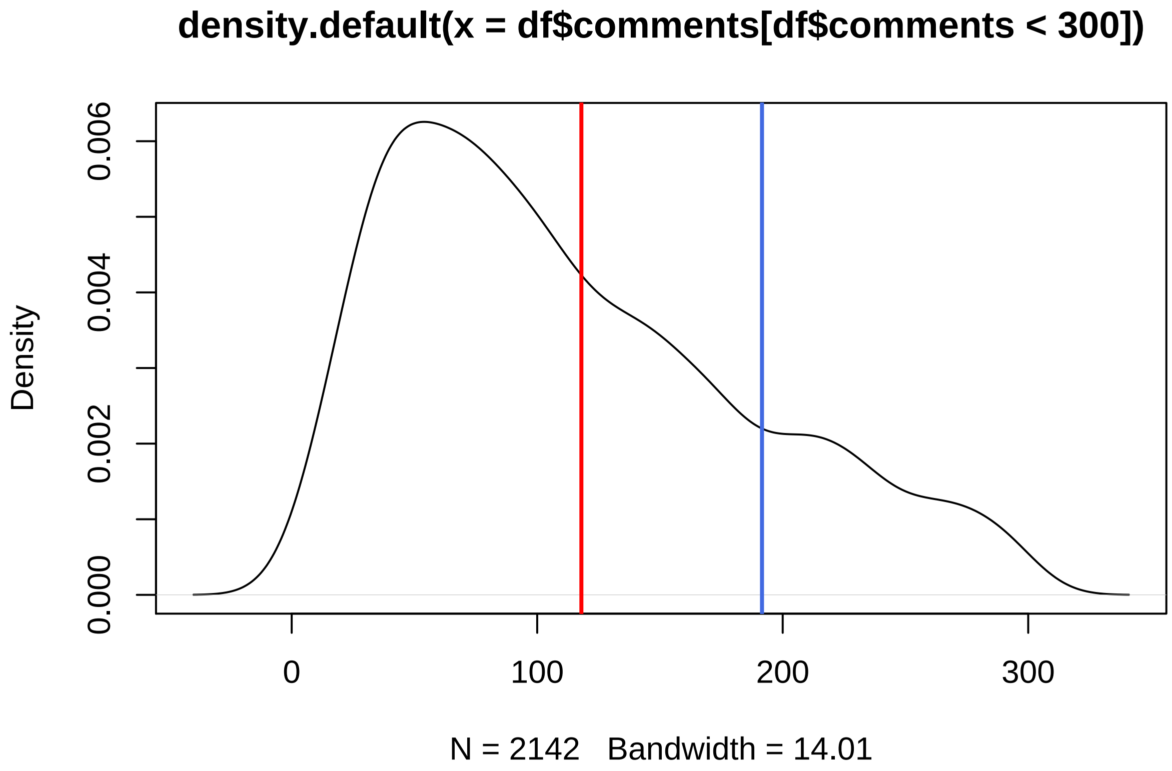
```
summary(df$views)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
##    50443   755793  1124524  1698297  1700760 47227110
```

```
# There are two talks viewed over 40 million times.

# For summarizing the comments

plot(density(df$comments))
```



```
# Since most of the talks have less than 300 comments we now replot the chart
plot(density(df$comments[df$comments < 300]))

abline(v = mean(df$comments),
 col = "royalblue",
 lwd = 2)

abline(v = median(df$comments),
 col = "red",
 lwd = 2)
```

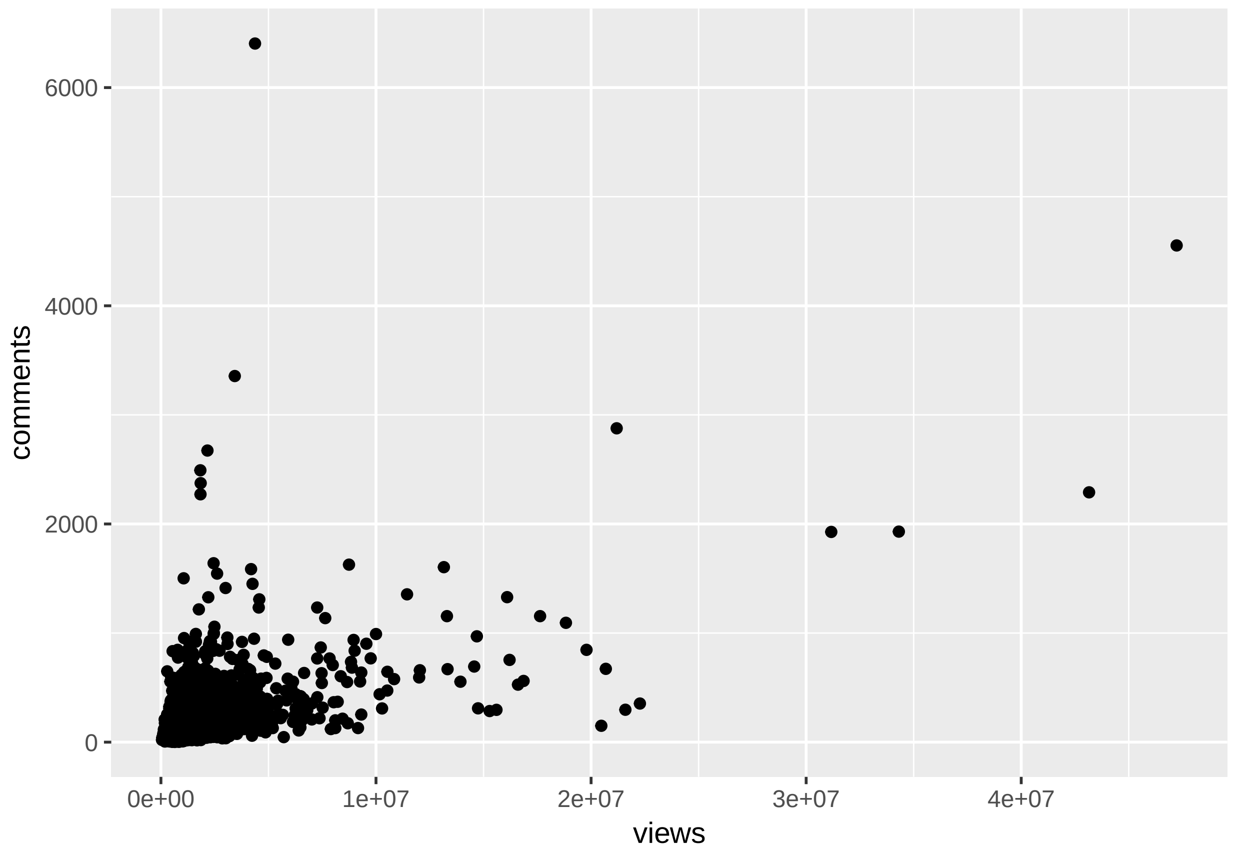


```r
summary(df$comments)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##     2.0    63.0   118.0   191.6   221.8  6404.0
```

For visualizing the relation between views and comments

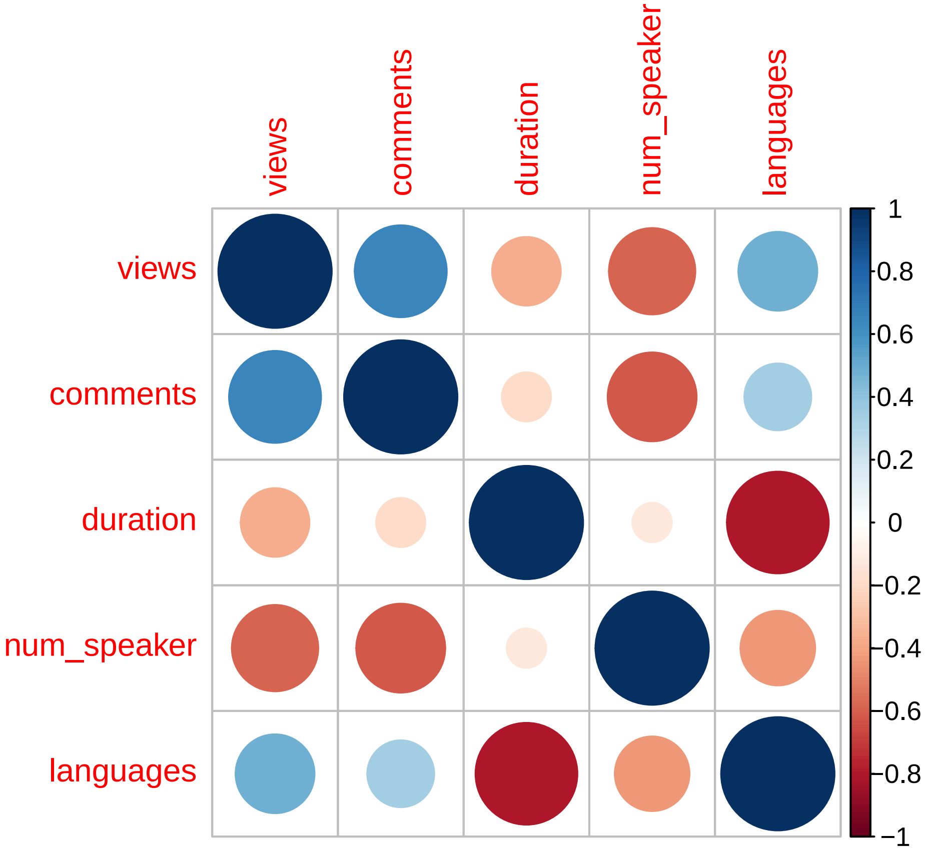
```r
ggplot(df, aes(x=views, y=comments)) +
  geom_point()
```



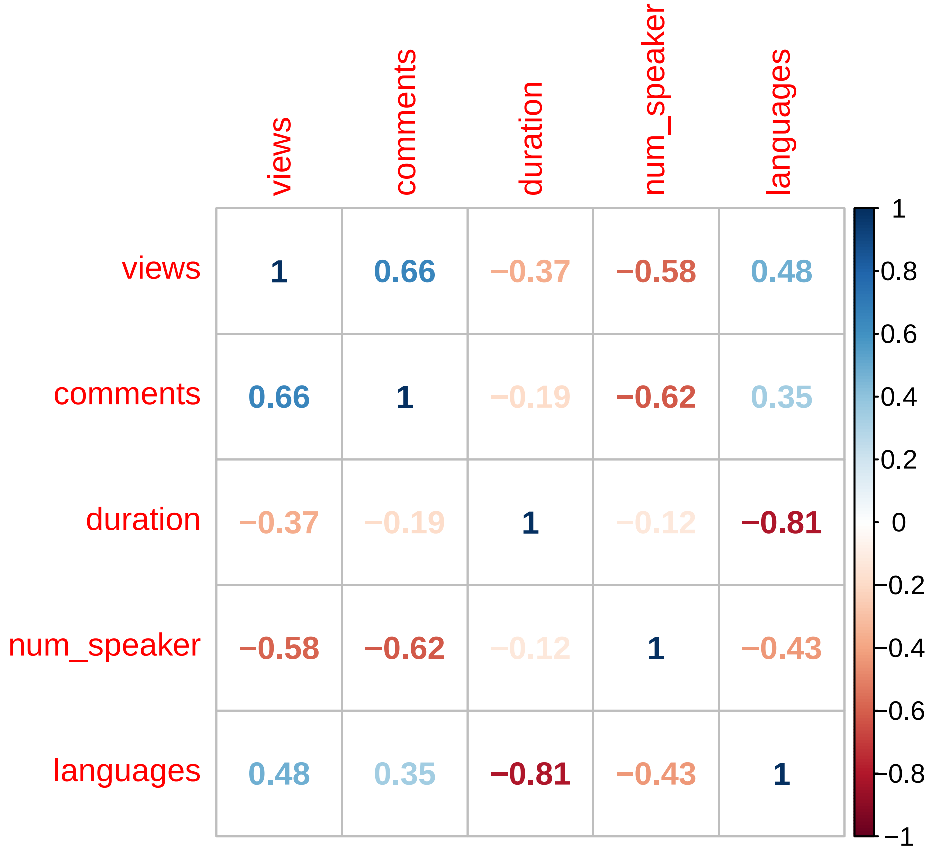
```
# For the calculation of correlation between views and comments
cor(df[, c("views", "comments")])
```

```
##               views  comments
## views    1.0000000 0.5309387
## comments 0.5309387 1.0000000
```

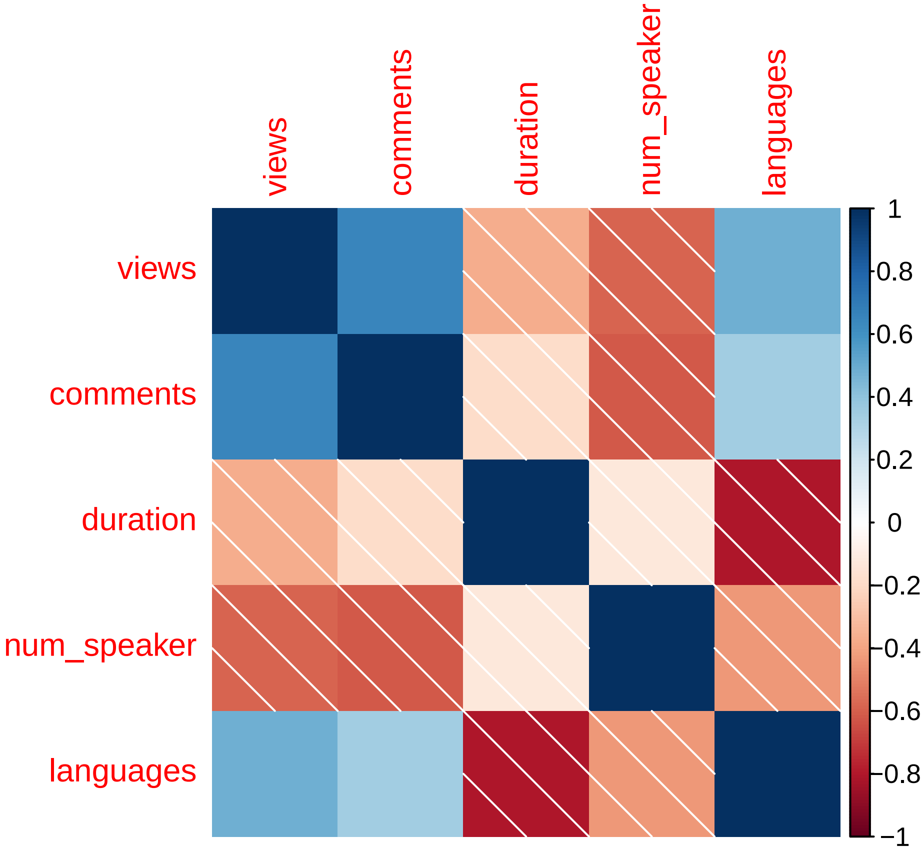
```
# Let's check all the correlations between the numeric fields, which was not analyzed on the kernel we
for_cor = cor(df[, c("views", "comments","duration","num_speaker","languages")])
M <- cor(for_cor)
View(M)
corrplot(M,method = "circle")
```



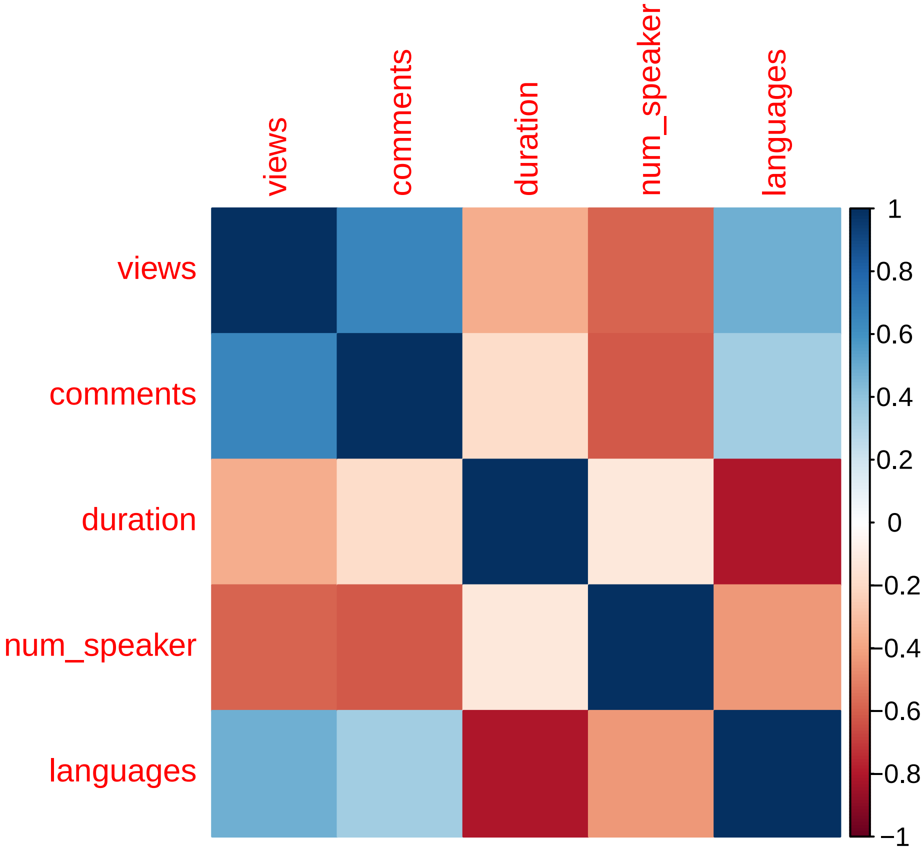
```
corrplot(M,method = "number")
```



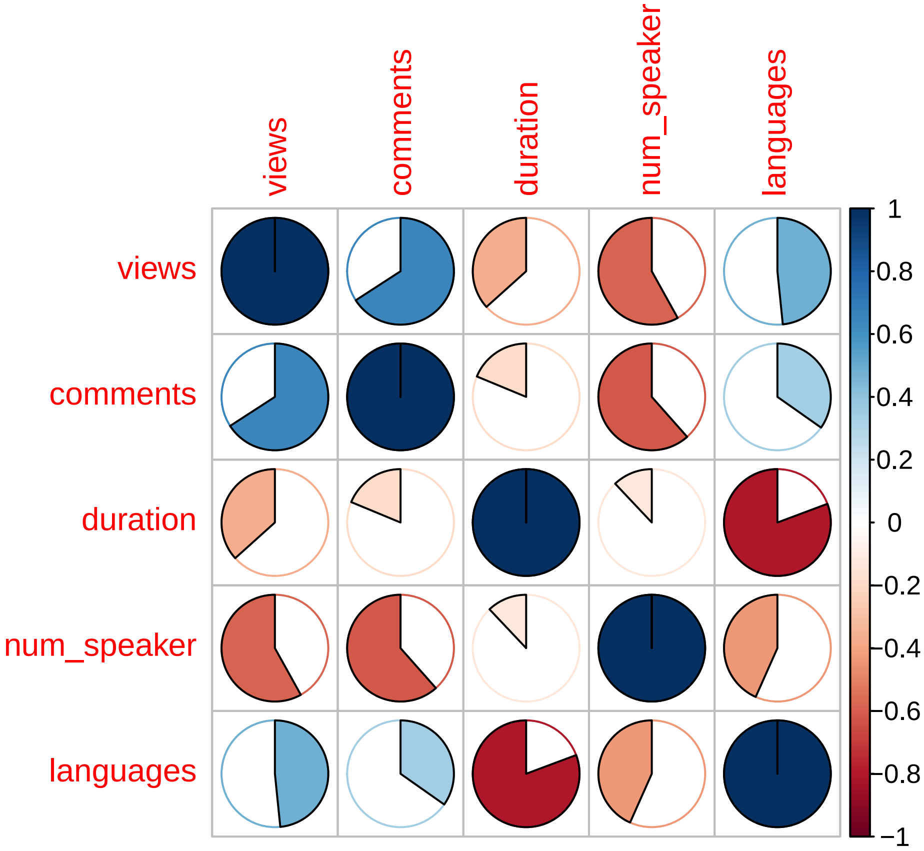
```r
corrplot(M,method = "shade")
```



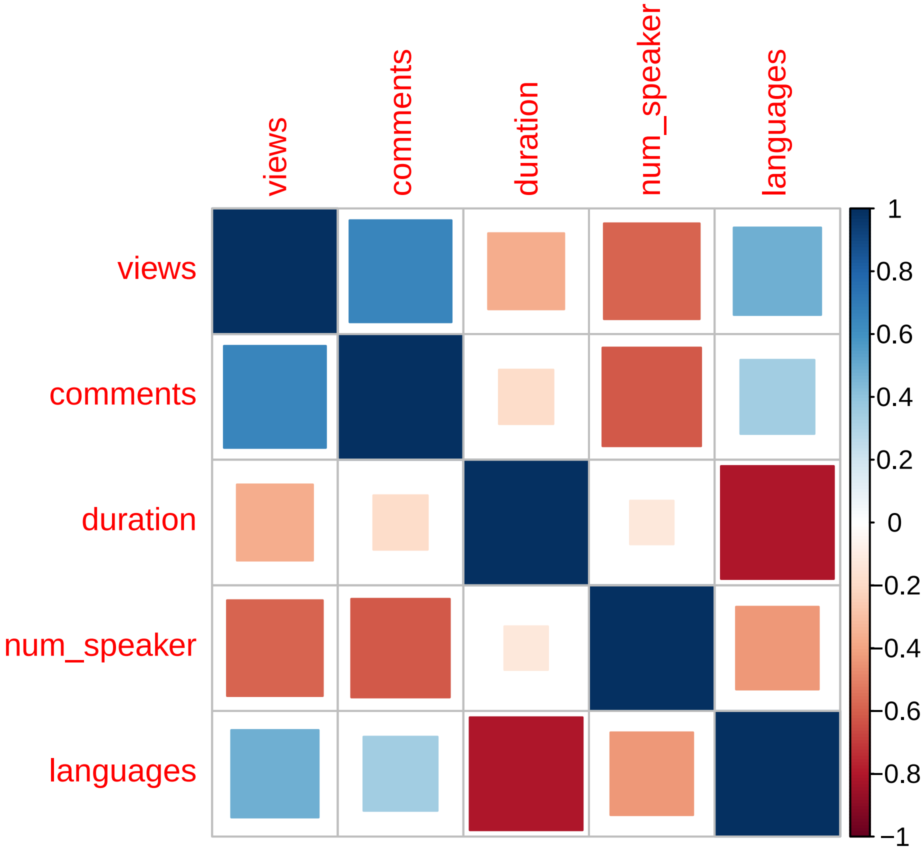
```r
corrplot(M,method = "color")
```



```
corrplot(M,method = "pie")
```



```r
corrplot(M,method = "square")
```

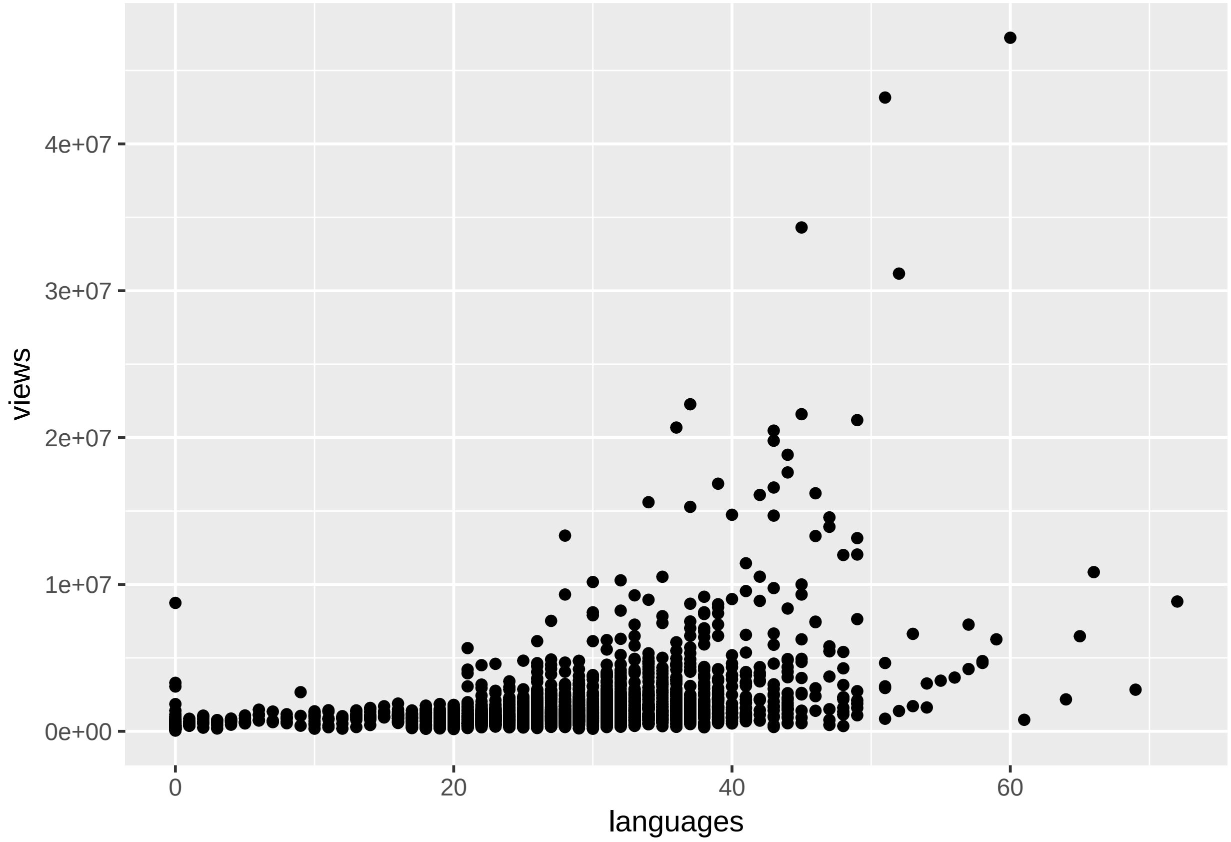


```
summary(for_cor)
```

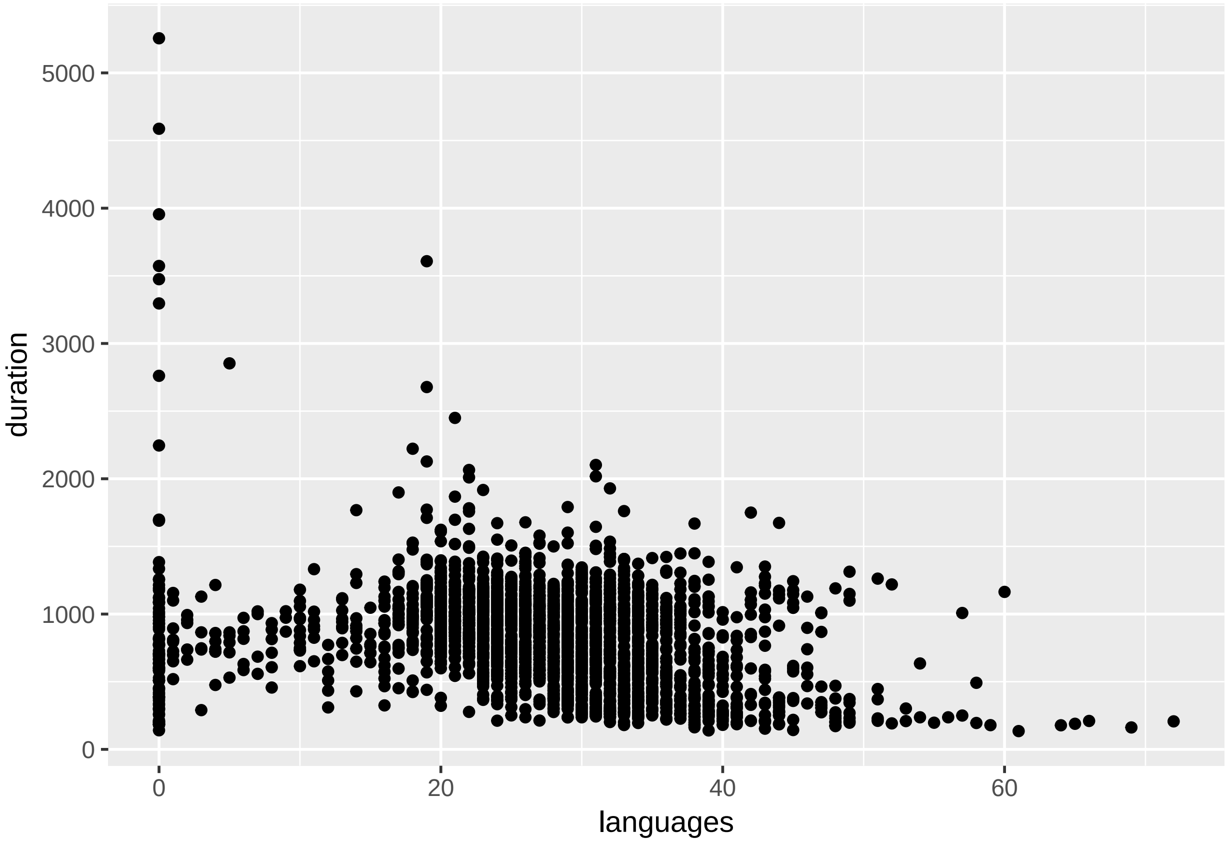
```
##       views             comments            duration
##  Min.   :-0.02639   Min.   :-0.03549   Min.   :-0.29568
##  1st Qu.: 0.04874   1st Qu.: 0.14069   1st Qu.: 0.02226
##  Median : 0.37762   Median : 0.31828   Median : 0.04874
##  Mean   : 0.38618   Mean   : 0.39089   Mean   : 0.18320
##  3rd Qu.: 0.53094   3rd Qu.: 0.53094   3rd Qu.: 0.14069
##  Max.   : 1.00000   Max.   : 1.00000   Max.   : 1.00000
##   num_speaker         languages
##  Min.   :-0.06310   Min.   :-0.2957
##  1st Qu.:-0.03549   1st Qu.:-0.0631
##  Median :-0.02639   Median : 0.3183
##  Mean   : 0.17946   Mean   : 0.2674
##  3rd Qu.: 0.02226   3rd Qu.: 0.3776
##  Max.   : 1.00000   Max.   : 1.0000
```

```
# From the visualizations above we cans see that;

# Views and comments have a mid positive relation of 66%,
# More the views more the comments is observed.

# Views and number of translations (languages) have a mid positive relation of 48%
# More the translations more the views may be observed.

# Views and number of speakers have a mid negative relation of -58%
# More speakers lead to less viewers.
```

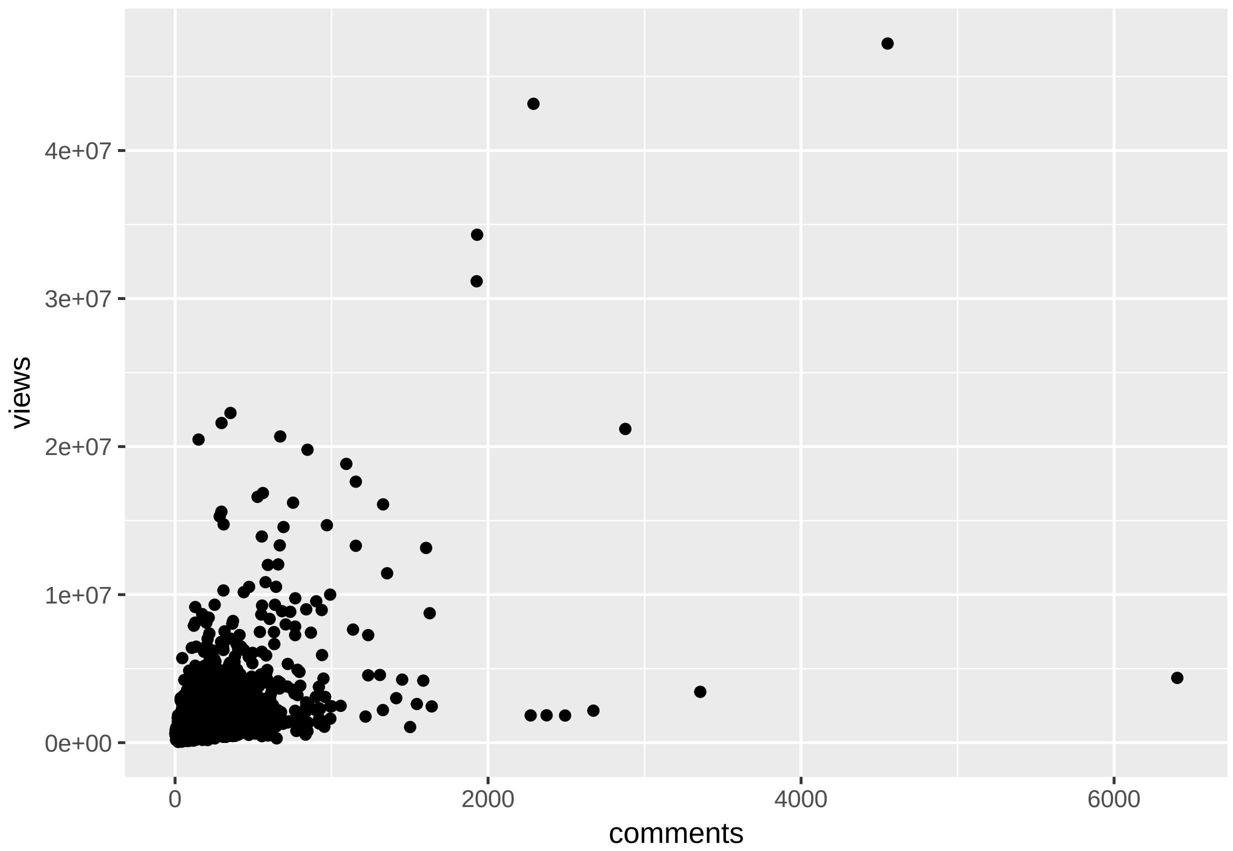
```r
# Comments and number of speakers have a mid negative relation of -62%
# More speakers lead to less comments as less views stated above.

# Duration and languages have a mid to high negative relation of -81%
# Usually longer duraions lead to lower number of translations of the talks.


# Visualizing many possible relations
ggplot(df, aes(x=languages, y=views)) + geom_point()
```



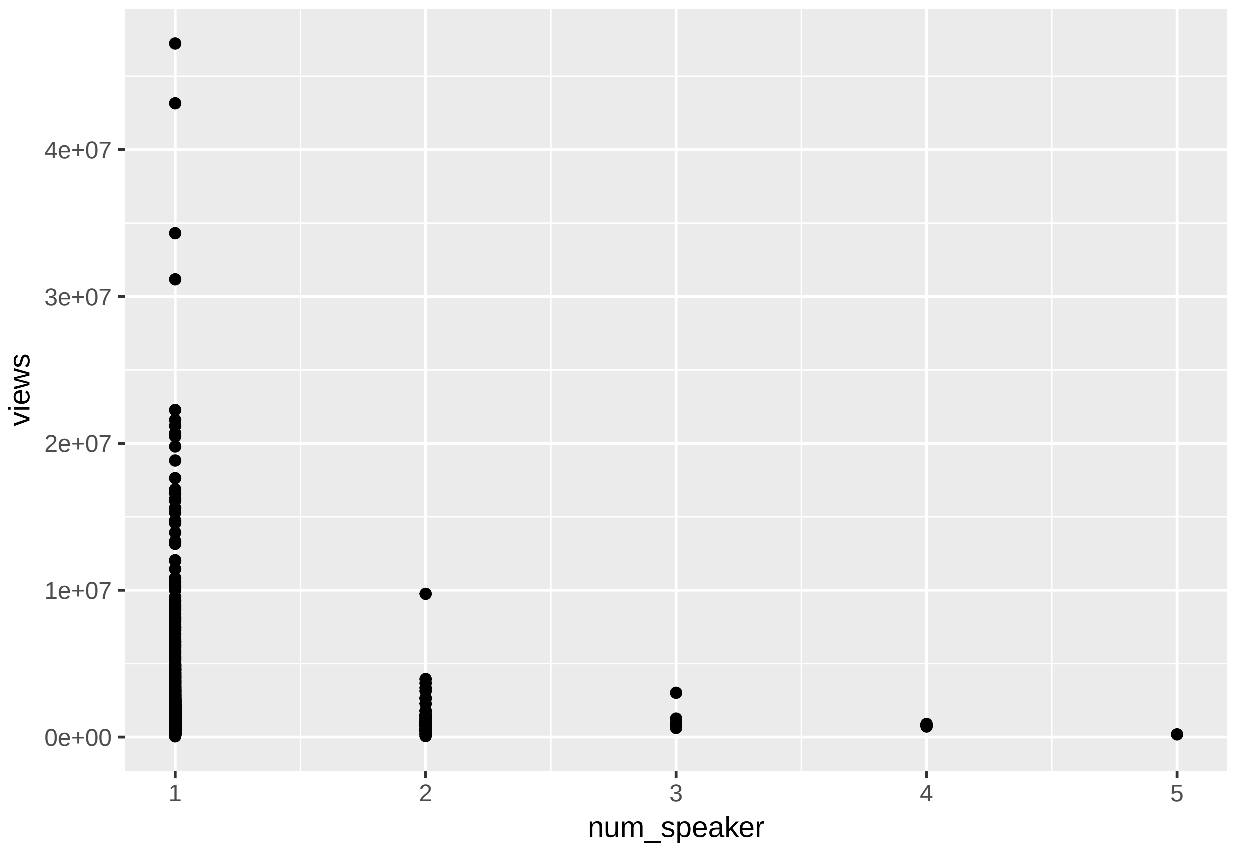
```r
ggplot(df, aes(x=languages, y=duration)) + geom_point()
```



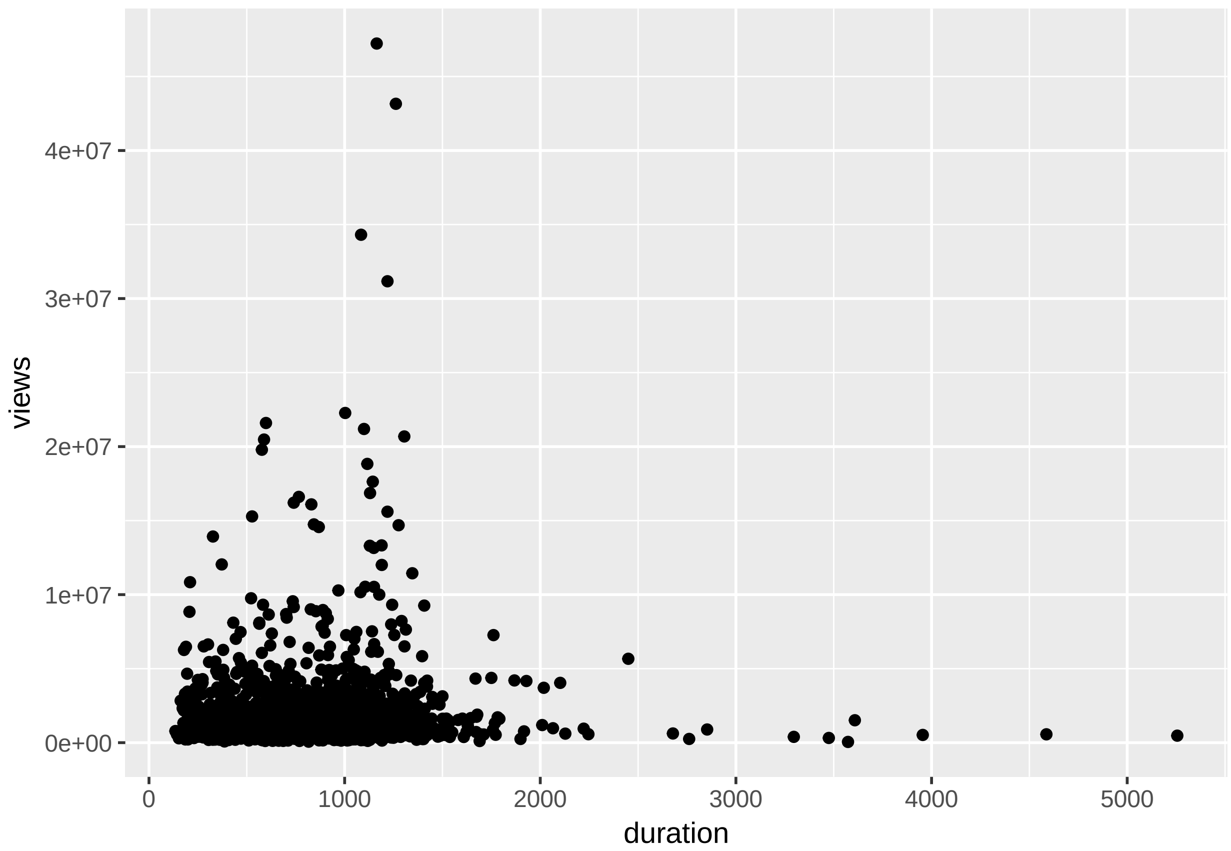
```r
ggplot(df, aes(x=comments, y=views)) + geom_point()
```



```r
ggplot(df, aes(x=num_speaker, y=views)) + geom_point()
```

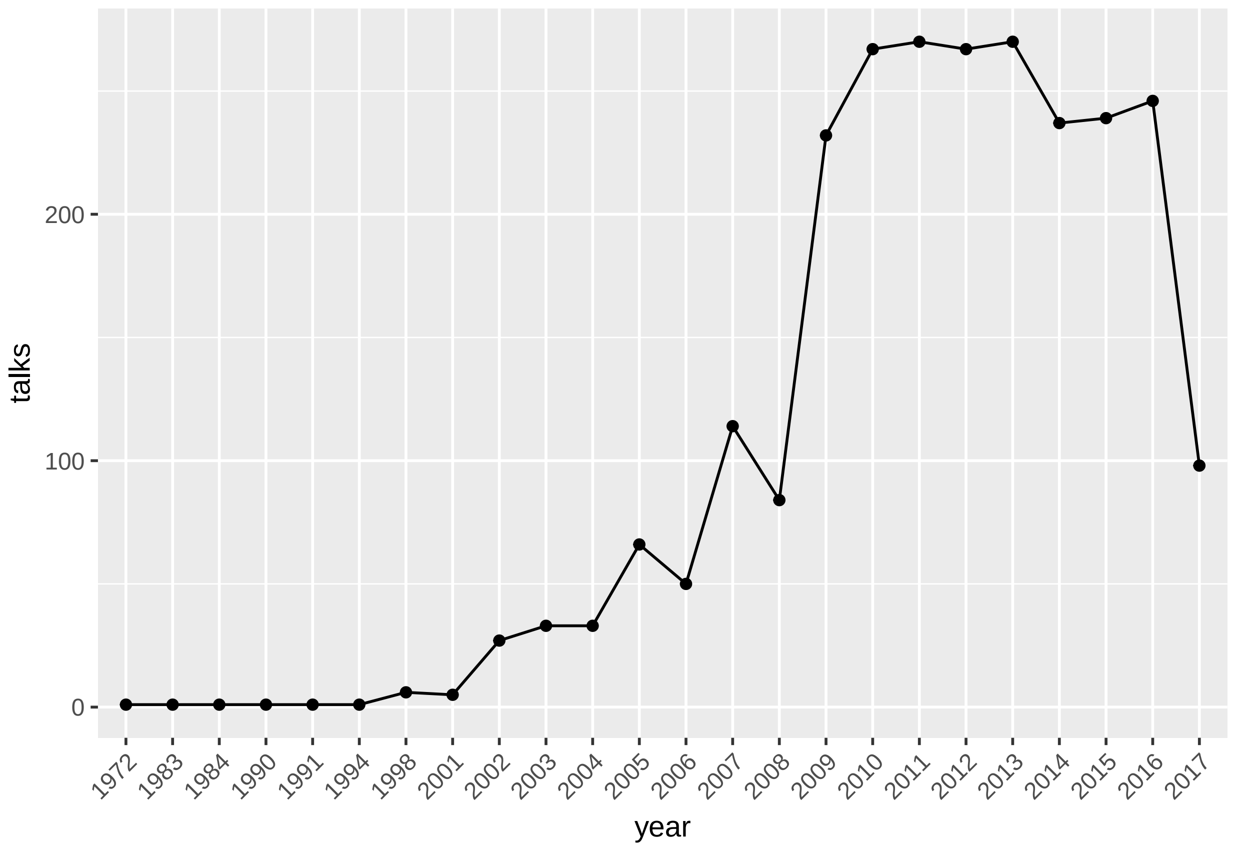


```r
ggplot(df, aes(x=duration, y=views)) + geom_point()
```



For displaying the number of talks by years

```
df$year <- year(df$film_date)
year_df <- data.frame(table(df$year))
colnames(year_df) <- c("year", "talks")
ggplot(year_df, aes(x=year, y=talks, group=1)) + geom_line() + geom_point() + theme(axis.text.x = elemen
```



```
# For listing the 15 most popular speakers
speaker_df <- data.frame(table(df$main_speaker))
colnames(speaker_df) <- c("main_speaker", "appearances")
speaker_df <- speaker_df %>% arrange(desc(appearances))
head(speaker_df, 15)
```

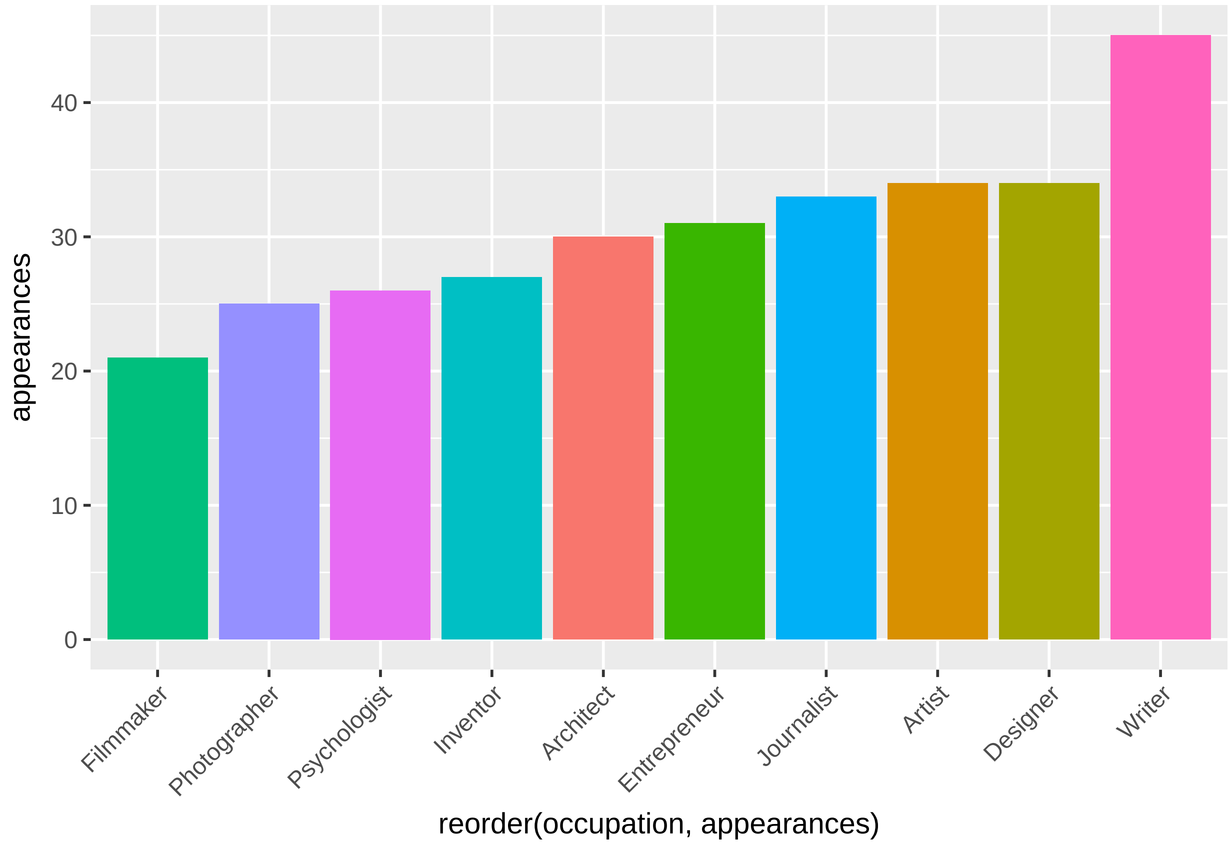
```
##            main_speaker appearances
## 1          Hans Rosling           9
## 2         Juan Enriquez           7
## 3         Marco Tempest           6
## 4                 Rives           6
## 5            Bill Gates           5
## 6           Clay Shirky           5
## 7            Dan Ariely           5
## 8  Jacqueline Novogratz           5
## 9        Julian Treasure           5
## 10  Nicholas Negroponte           5
## 11              Al Gore           4
## 12       Barry Schwartz           4
## 13       Chris Anderson           4
## 14          Dan Dennett           4
## 15          David Pogue           4
```

To find the most popular occupations among speakers

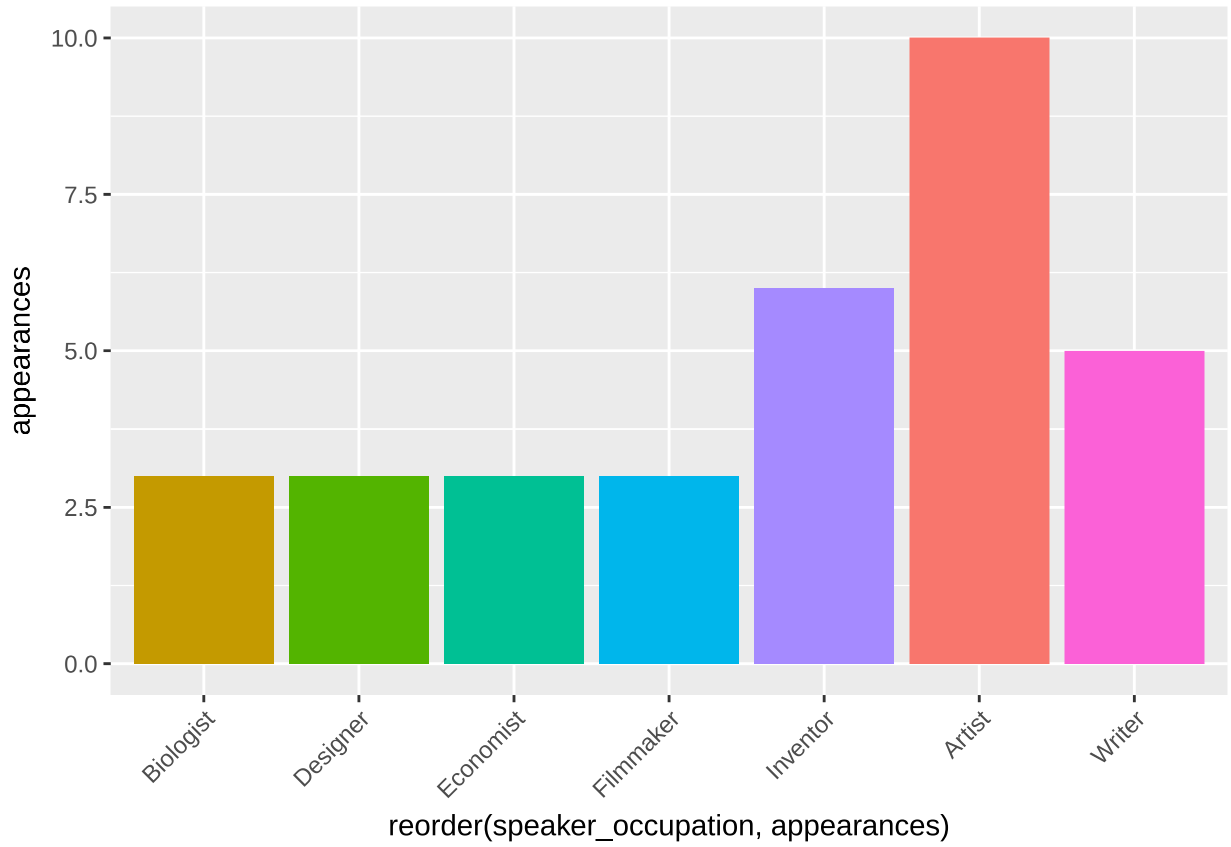
```
occupation_df <- data.frame(table(df$speaker_occupation))
colnames(occupation_df) <- c("occupation", "appearances")
occupation_df <- occupation_df %>% arrange(desc(appearances))
head(occupation_df, 10)
```

```
##       occupation appearances
## 1         Writer          45
## 2         Artist          34
## 3       Designer          34
## 4     Journalist          33
## 5   Entrepreneur          31
## 6      Architect          30
## 7       Inventor          27
## 8   Psychologist          26
## 9   Photographer          25
## 10    Filmmaker           21
```

```
# To plot the popular occupations on a barchart

ggplot(head(occupation_df,10), aes(x=reorder(occupation, appearances),
                                   y=appearances, fill=occupation)) +
    geom_bar(stat="identity") + guides(fill=FALSE) + theme(axis.text.x = element_text(angle = 45, vjust
```



```
# To find the most popular occupations per event which was not analyzed on the kernel we used as a basi
occup_dfLast <- data.frame(table(df$event, df$speaker_occupation))
colnames(occup_dfLast) <- c("event", "speaker_occupation", "appearances")
occup_dfLast <- occup_dfLast %>% arrange(desc(appearances))
View(occup_dfLast)

# To plot those occupations on a barchart

ggplot(head(occup_dfLast,10), aes(x=reorder(speaker_occupation, appearances), y=appearances, fill=speake
```



Number of speakers by talks

```
table(df$num_speaker)
```

```
##
##    1    2    3    4    5
## 2492   49    5    3    1
```

```
# Let's list the 3 talks which had 4 speakers
df[df[,'num_speaker'] == 4, c('title', 'description', 'main_speaker', 'event')]
```

```
## # A tibble: 3 x 4
##                                                     title
##                                                     <chr>
## 1      The interspecies internet? An idea in progress
## 2 An interview with the founders of Black Lives Matter
## 3 Political common ground in a polarized United States
## # ... with 3 more variables: description <chr>, main_speaker <chr>,
## #   event <chr>
```

Number of talks by events

```r
event_df <- data.frame(table(df$event))
colnames(event_df) <- c("event", "talks")
event_df <- event_df %>% arrange(desc(talks))
head(event_df, 10)
```

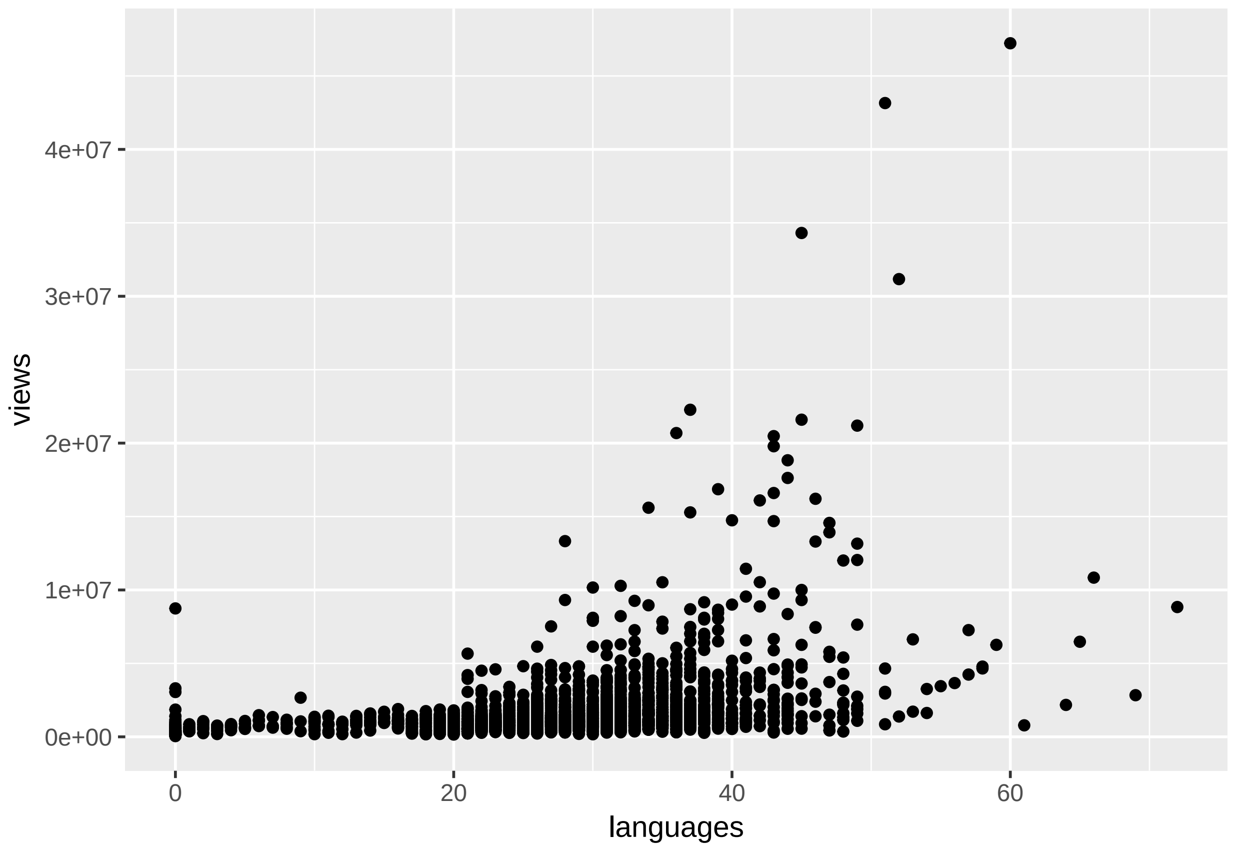
```
##             event talks
## 1         TED2014    84
## 2         TED2009    83
## 3         TED2013    77
## 4         TED2016    77
## 5         TED2015    75
## 6         TED2011    70
## 7  TEDGlobal 2012    70
## 8         TED2007    68
## 9         TED2010    68
## 10 TEDGlobal 2011    68
```

Languages

```r
summary(df$languages)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.00   23.00   28.00   27.33   33.00   72.00
```

```r
# Does more languages cause more views? There is a medium correlation rate of 0,38 as we also analyzed
ggplot(df, aes(x=languages, y=views)) + geom_point()
```



```
cor(df[, c("languages","views")])
```

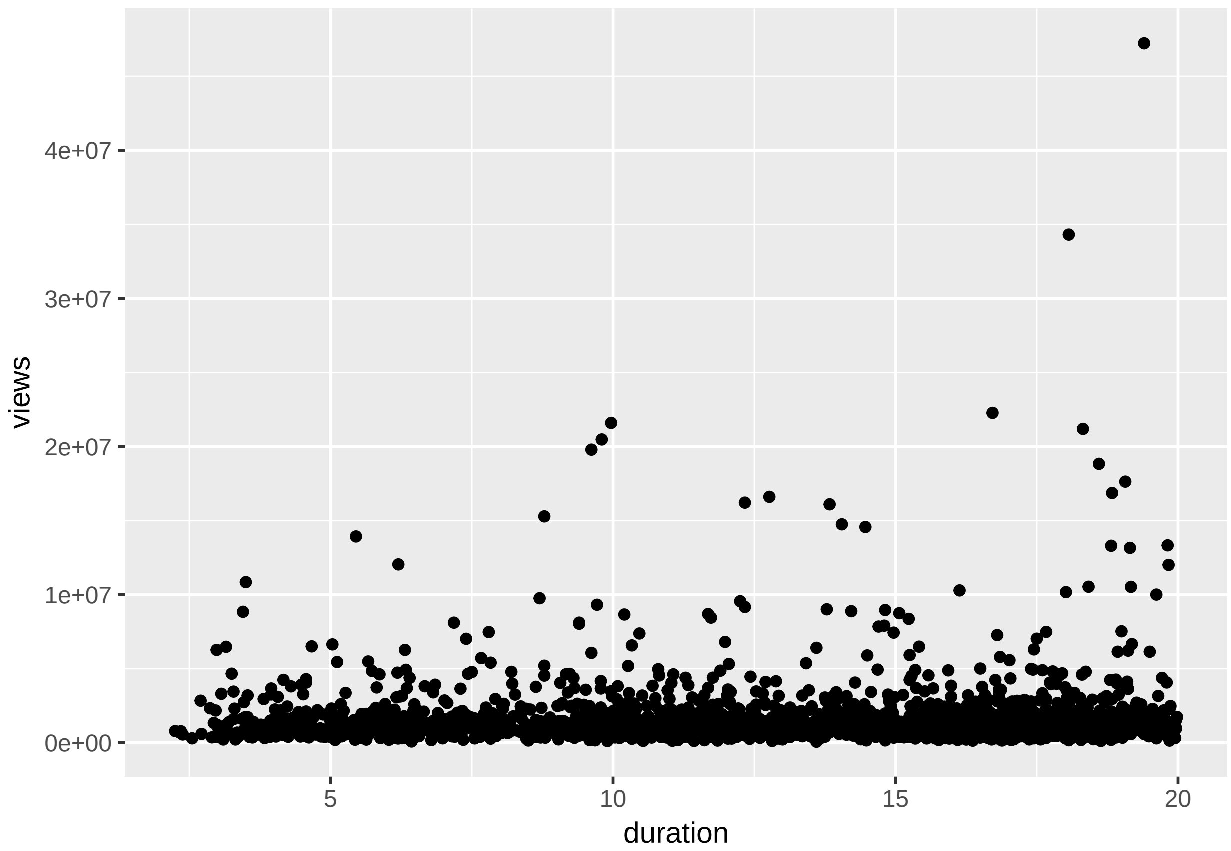
```
##           languages     views
## languages 1.0000000 0.3776231
## views     0.3776231 1.0000000
```

Distribution of talk durations

```
df$duration <- df$duration/60
summary(df$duration)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   2.250   9.617  14.133  13.775  17.446  87.600
```

```
# Since the talks average to less than 14 mins we would like to plot the talks shorter than 20 mins
ggplot(df[df$duration<20,], aes(x=duration, y=views)) + geom_point()
```



```
df2 <- read.csv("C:/AT/tedtalks/transcripts.csv", stringsAsFactors = FALSE)
head (df2)
```

```
##
## 1
```

```
nrow (df2)
```

```
## [1] 2467
```

```
df3 <- base::merge(df, df2, by="url")
head(df3)
```

```
##                                                                                url
## 1     https://www.ted.com/talks/9_11_healing_the_mothers_who_found_forgiveness_friendship\n
## 2                          https://www.ted.com/talks/a_j_jacobs_year_of_living_biblically\n
## 3                               https://www.ted.com/talks/a_robot_that_flies_like_a_bird\n
## 4                              https://www.ted.com/talks/a_ted_speaker_s_worst_nightmare\n
## 5                            https://www.ted.com/talks/a_whistleblower_you_haven_t_heard\n
## 6 https://www.ted.com/talks/aakash_odedra_a_dance_in_a_hurricane_of_paper_wind_and_light\n
##                                                                                    name
## 1 Aicha el-Wafi + Phyllis Rodriguez: The mothers who found forgiveness, friendship
## 2                                          AJ Jacobs: My year of living biblically
## 3                                   Markus Fischer: A robot that flies like a bird
## 4                               Improv Everywhere: A TED speaker's worst nightmare
## 5                                 Geert Chatrou: A whistleblower you haven't heard
## 6                   Aakash Odedra: A dance in a hurricane of paper, wind and light
##                                              title
## 1   The mothers who found forgiveness, friendship
## 2                    My year of living biblically
## 3                  A robot that flies like a bird
## 4                 A TED speaker's worst nightmare
## 5               A whistleblower you haven't heard
## 6 A dance in a hurricane of paper, wind and light
##
## 1 Phyllis Rodriguez and Aicha el-Wafi have a powerful friendship born of unthinkable loss. Rodriguez
## 2
## 3
## 4
## 5
## 6                                                Choreographer Aakash Odedra is dyslexic and has alwa
##                         main_speaker         speaker_occupation num_speaker
## 1 Aicha el-Wafi + Phyllis Rodriguez               9/11 mothers           1
## 2                         AJ Jacobs                     Author           1
## 3                    Markus Fischer                   Designer           1
## 4                 Improv Everywhere Social energy entrepreneur           1
## 5                     Geert Chatrou                   Whistler           1
## 6                     Aakash Odedra              Choreographer           1
##    duration              event  film_date published_date comments
## 1  9.900000      TEDWomen 2010 2010-12-12     2011-05-02      149
## 2 17.666667            EG 2007 2007-12-02     2008-07-17      583
## 3  6.316667     TEDGlobal 2011 2011-07-15     2011-07-22      440
## 4  3.816667            TED2012 2012-03-01     2012-03-09      324
## 5 11.933333 TEDxRotterdam 2010 2010-06-04     2011-02-11       93
## 6  9.833333     TEDGlobal 2014 2014-10-21     2014-12-05       48
##                                                                                    tags
## 1               ['culture', 'friendship', 'global issues', 'parenting', 'terrorism']
## 2 ['comedy', 'culture', 'entertainment', 'humanity', 'humor', 'religion', 'writing']
## 3        ['animals', 'biomechanics', 'biomimicry', 'design', 'robots', 'technology']
## 4                                ['entertainment', 'performance', 'performance art']
## 5                    ['TEDx', 'entertainment', 'live music', 'music', 'performance']
## 6                                                  ['dance', 'music', 'performance']
##   languages
## 1        32
## 2        39
## 3        45
## 4        51
## 5        31
## 6        39
##
```

```
## 1            [{'id': 10, 'name': 'Inspiring', 'count': 385}, {'id': 1, 'name': 'Beautiful', 'count':
## 2 [{'id': 22, 'name': 'Fascinating', 'count': 531}, {'id': 3, 'name': 'Courageous', 'count': 345}, {
## 3 [{'id': 23, 'name': 'Jaw-dropping', 'count': 1487}, {'id': 22, 'name': 'Fascinating', 'count': 119
## 4     [{'id': 2, 'name': 'Confusing', 'count': 186}, {'id': 7, 'name': 'Funny', 'count': 1423}, {'id
## 5           [{'id': 23, 'name': 'Jaw-dropping', 'count': 216}, {'id': 22, 'name': 'Fascinating', 'co
## 6             [{'id': 1, 'name': 'Beautiful', 'count': 423}, {'id': 25, 'name': 'OK', 'count': 92},
##
## 1 [{'id': 968, 'hero': 'https://pe.tedcdn.com/images/ted/202850_800x600.jpg', 'speaker': 'Inge Missm
## 2
## 3
## 4                                                                                               [{'i
## 5
## 6
##     views year
## 1  820976 2010
## 2 2291701 2007
## 3 6264902 2011
## 4 2950307 2012
## 5 1917442 2010
## 6  817014 2014
##
and avoid that## 1they get married and suffer as much as I did, well this is something good. This is why I'm here in fro
## 2 I thought I'd tell you a little about what I like to write. And I like to immerse myself in my top
```

```r
wc <- function(x){
  #x <- as.character(x)
  words <- strsplit(x[[1]], split="\\s+")
  return(length(words[[1]]))
}

df3$wc <- sapply(df3$transcript, wc)
summary(df3$wc)
```

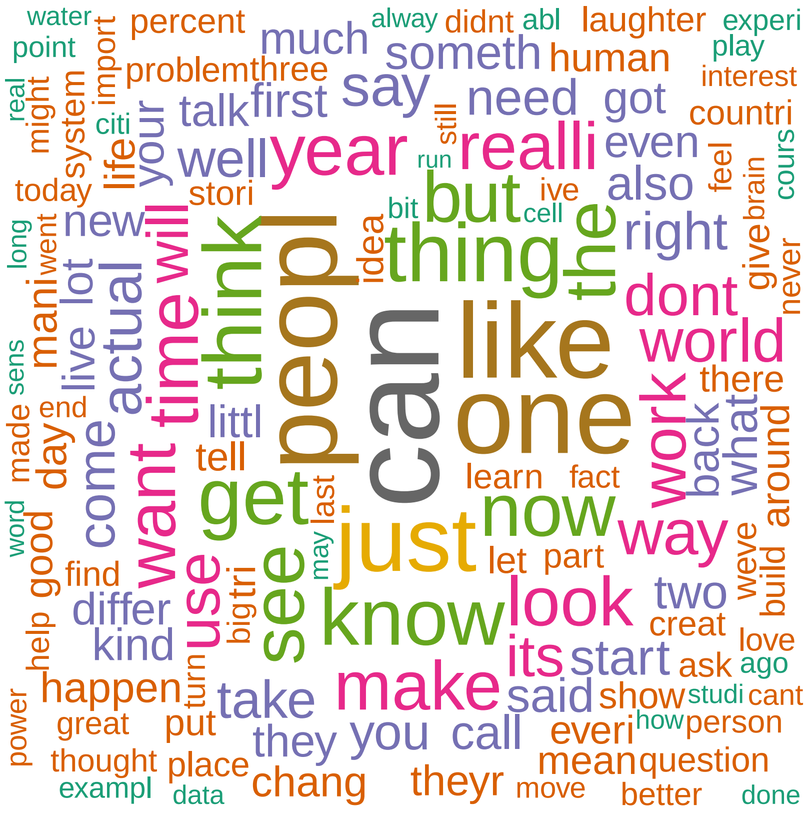
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       1    1332    2028    2040    2707    9044
```

```r
# Word Cloud

texts <- df3$transcript
#texts <- iconv(texts, to = "utf-8")
corpus <- Corpus(VectorSource(texts))
corpus <- tm_map(corpus, PlainTextDocument)
corpus <- tm_map(corpus, removePunctuation)
corpus <- tm_map(corpus, removeWords, stopwords('english'))
corpus <- tm_map(corpus, stemDocument)
corpus <- tm_map(corpus, removeWords, c("and", "this", "there"))
corpus <- Corpus(VectorSource(corpus))
dtm <- TermDocumentMatrix(corpus)
m <- as.matrix(dtm)
v <- sort(rowSums(m),decreasing=TRUE)
d <- data.frame(word = names(v),freq=v)
```

```r
head(d, 10)
```

```
##        word  freq
## and     and 42885
## can     can 24129
## one     one 20275
## like   like 19814
## peopl peopl 19527
## just   just 16098
## thing thing 14545
## think think 14370
## that   that 13963
## get     get 13840
```

```r
d <- d[-which(d$word %in% c("and","this","that")),]
set.seed(1234)

  # For temporarily disabling warnings
  oldw <- getOption("warn")
  options(warn = -1)

wordcloud(words = d$word, freq = d$freq, min.freq = 1,
          max.words=200, random.order=FALSE, rot.per=0.35,
          colors=brewer.pal(8, "Dark2"))
```



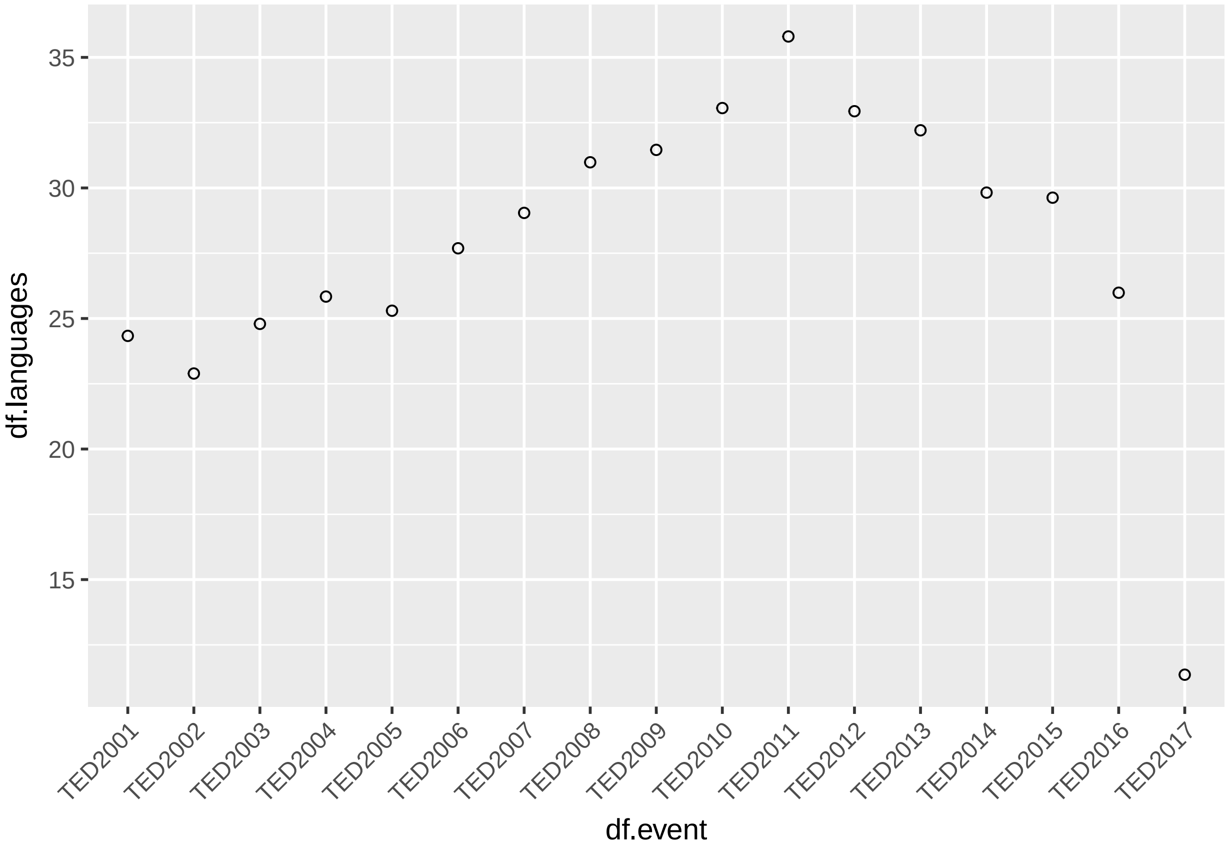
```r
  # For re-enabling warnings
  options(warn = oldw)
```

```
# Average number of languages per event, and then per TED2xxx event.

evntlang <- data.frame(df$event, df$languages)
avg <- aggregate(.~df.event, data=evntlang, mean)

avg_sub <- avg[avg$df.languages > 0, ]
avgsub_last <- avg_sub[substr(avg_sub$df.event, 1, 4) == "TED2", ]
head (avgsub_last)
```

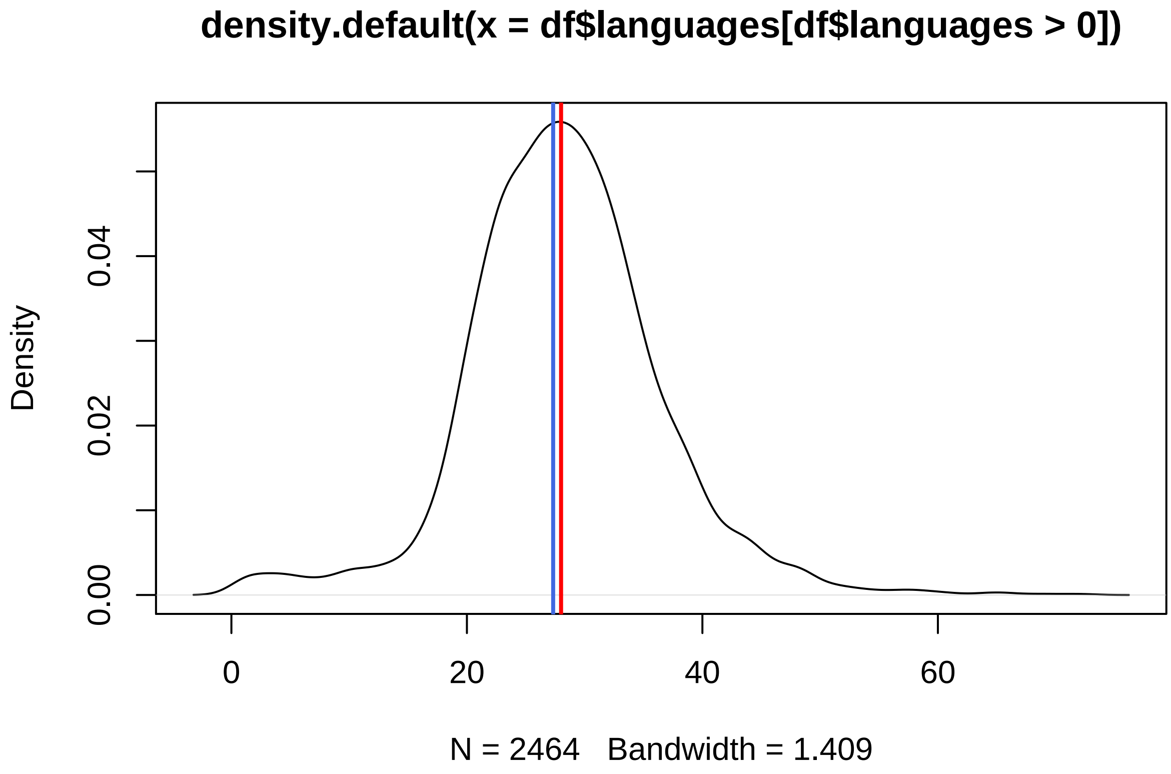
```
##    df.event df.languages
## 75  TED2001     24.33333
## 76  TED2002     22.89286
## 77  TED2003     24.79412
## 78  TED2004     25.83871
## 79  TED2005     25.29730
## 80  TED2006     27.68889
```

```
graph_sub <- ggplot(avgsub_last, aes(x=df.event, y=df.languages)) + geom_point(shape=1) + geom_smooth(me
graph_sub <- graph_sub + stat_smooth() + theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust=
graph_sub
```

```
## `geom_smooth()` using method = 'loess'
```



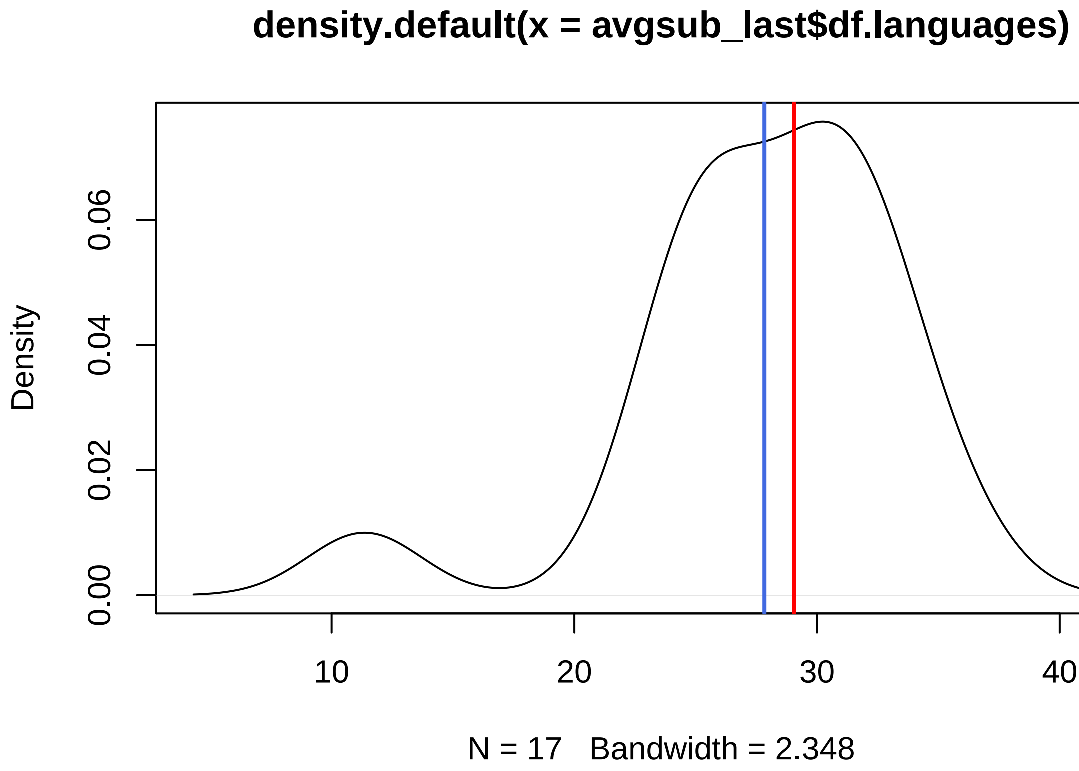
```
plot(density(df$languages[df$languages > 0]))

abline(v = mean(df$languages),
 col = "royalblue",
 lwd = 2)

abline(v = median(df$languages),
 col = "red",
 lwd = 2)
```



```
summary(df$languages)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.00   23.00   28.00   27.33   33.00   72.00
```

```
# Average number and histogram of languages per TED2xxx event.

plot(density(avgsub_last$df.languages))

abline(v = mean(avgsub_last$df.languages),
 col = "royalblue",
 lwd = 2)

abline(v = median(avgsub_last$df.languages),
 col = "red",
 lwd = 2)
```



```
summary(avgsub_last$df.languages)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##   11.36   25.30   29.04   27.83   31.46   35.80
```