WHAT MAKES A TED TALK POPULAR?

Tomer Eldor
Minerva Schools
January 26, 2018

Have you ever wondered what makes some TED talks more popular than others? Well, I've analyzed a dataset of 2550 ted talks to get some answers for this question. May aim was to explore which of my available variables of a given talk, such as the number of comments, number of languages translated, duration of the talk, number of tags, or day it was published online– are a strong predictor of its popularity, measured in number of views.

I don't believe these analyses serve as a good *causal inference*, since results wouldn't be matched with these variables, their explanatory power isn't rigorous enough. The available numerical parameters I had in hand are not sufficient for that kind of a conclusion; I couldn't match the *content that really matters* to compare apples to apples, and even with controlling with multiple regression – not all things are equal (ceteris paribus assumption is still not met). However, I was able to get a decent predictor and understand which variables are most strongly *associated* with higher view counts.

What data did I have? The dataset includes the name, title, description and URL of each of the 2550 talks, name and occupation of the main speaker, number of speakers, duration of the talk, the TED event and date it was filmed at, date it was published online, number of comments, languages translated and views. It also includes data points as the array of associated tags, ratings, and related talks, but as inside arrays and these need transformations before they can be used. For a full list, see comment.[1]

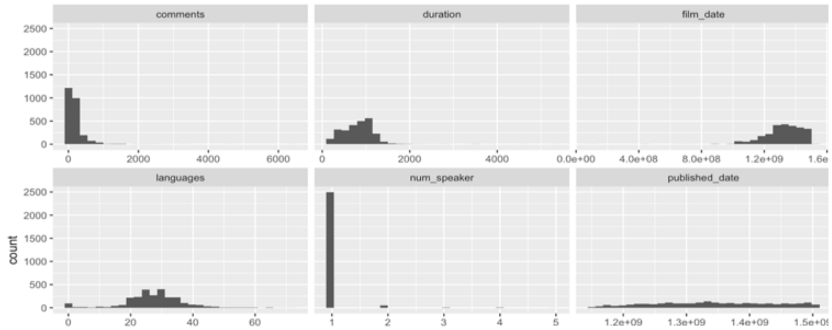
---

[1] TED Main Dataset is described more fully and accessible via Kaggle.com at this link. The variables are below:

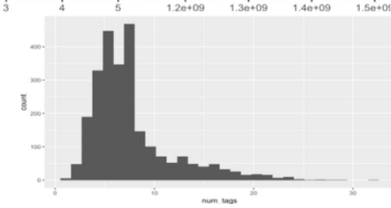
- name: The official name of the TED Talk. Includes the title and the speaker.
- title: The title of the talk
- description: A blurb of what the talk is about.
- main_speaker: The first named speaker of the talk.
- speaker_occupation: The occupation of the main speaker.
- num_speaker: The number of speakers in the talk.
- duration: The duration of the talk in seconds.
- event: The TED/TEDx event where the talk took place.
- film_date: The Unix timestamp of the filming.
- published_date: The Unix timestamp for the publication of the talk on TED.com
- comments: The number of first level comments made on the talk.
- tags: A list of the tag themes associated with the talk.
- languages: The number of languages in which the talk is available.
- ratings: A stringified dictionary of the various ratings given to the talk (inspiring, fascinating, jaw dropping, etc.)
- related_talks: A list of dictionaries of recommended talks to watch next.
- url: The URL of the talk.
- **views:** The number of views on the talk.

DATA EXPLORATION AND SUMMARY STATISTICS

1 - HISTOGRAMS

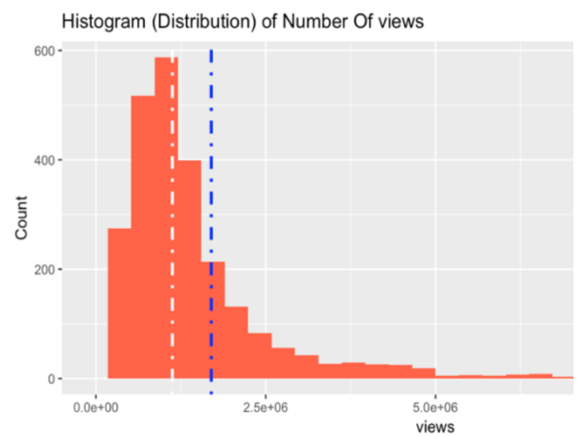


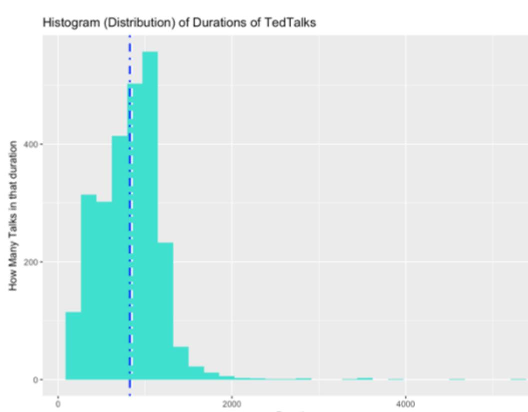
Let's examine the distributions of the key parameters that we'll be using. To the right are histograms of most of our numerical variables. Below are some more detailed histograms, with a white line for the median and blue line for the mean.

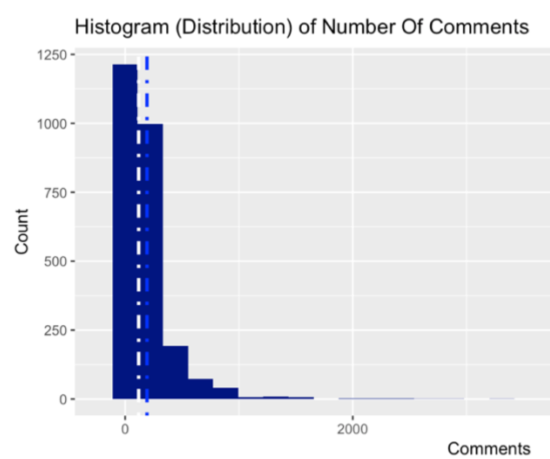


Insights and description from the distributions:

- Number of views is also a Poisson like distribution skewed leftwards
- Duration of talks is closer to a normal distribution, but with a wide right-side tail of a few talks at longer durations, around a mean of 14 minutes and median of 12 minutes. Almost all talks range between 1-18 minutes (maximum length of a normal Ted talk).
- Number of comments is a Poisson distribution (visually resembling an exponential distribution, but it is not technically since comments are measured at discrete numbers) strongly skewed to the minimum of 0 comments for the unpopular videos.
- Number of tags also is a Poisson distribution skewed leftwards, peaks between 4-7 tags.
- Number of languages of translations has a peak at 0 for unpopular talks, but mostly is between 20-40 languages offered.
- While the film dates are low before 2012, they are all published at a much more uniform rate.

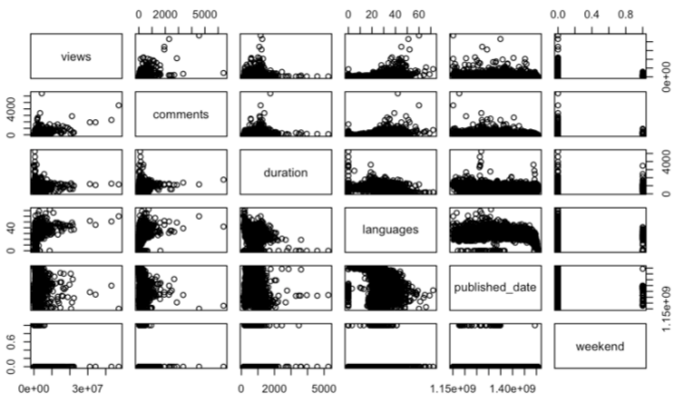


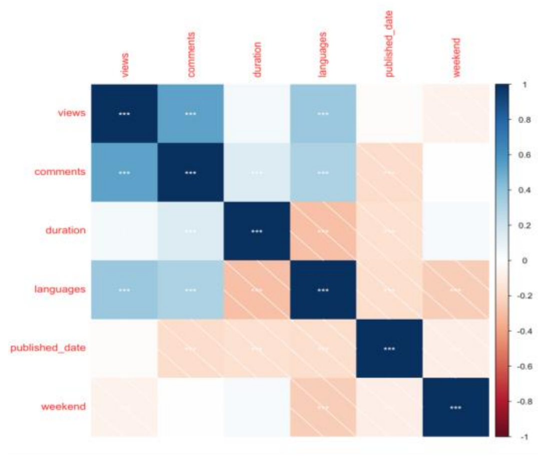




2 - CORRELATIONS BETWEEN PARAMETERES

On the left is a correlation (pairs) scatterplot matrix between each pair of numerical variables; on the right is a correlation matrix with colors representing the intensity of the correlation from 0 (white) to dark blue (+1) or dark striped red (strong negative correlation, -1), and with asterisks (***) signifying significance by p-values.



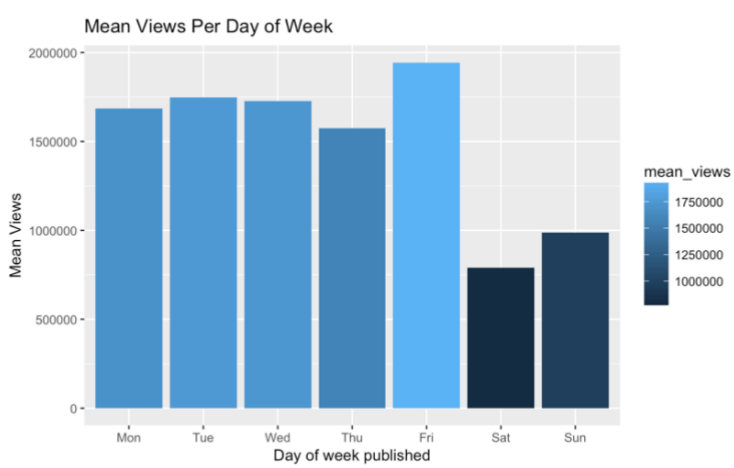


Most of the parameters don't have strong correlations.

- Naturally, there was a very high correlation between published data and filmed date. Filmed date seemed to be less associated with views numerically and logically – since the audience is more affected by the date a ted talk is released than whether it was recorded a month ago or a year ago.
- There is a relatively higher positive correlation between number of comments and views, which makes sense (more audience, more comments);
- Some positive correlation between number of languages of translation and number of views (0.38) and number of comments (0.32)
- Small negative correlation between duration and number of languages; the shorter the talk, the more translated languages there are, probably because it is easier to translate.

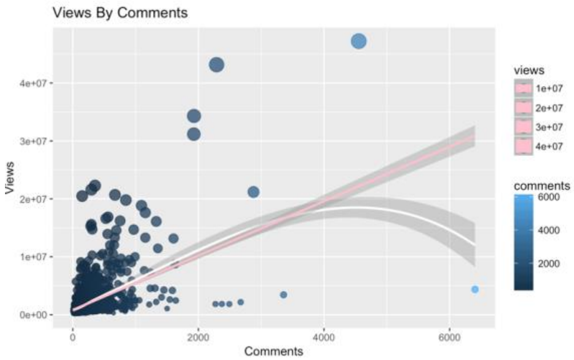
3 - CORRELATION OF PARAMTERES WITH NUMBERS OF VIEWS

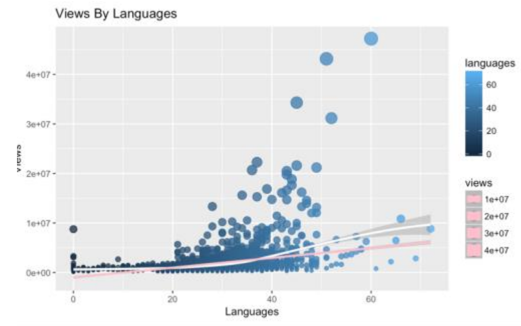
How do these variables correlate with views? Is it a linear relationship, nonlinear, or non? This is important to understand to know how to insert them into the regression if at all.

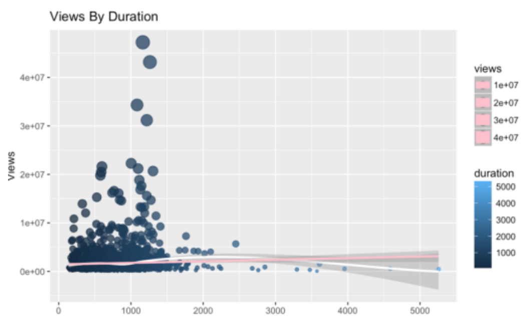


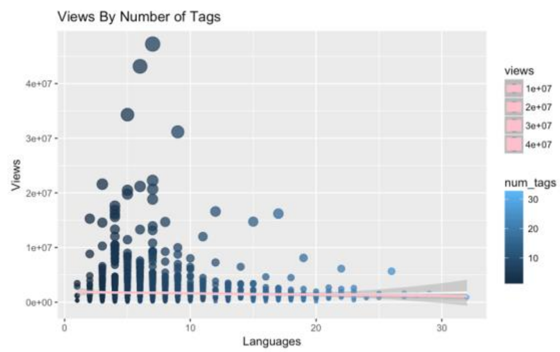
So, day of week does seem to have some association with average (mean) views! Ted Talks published on weekends seem to have much less views, with Saturday being the lowest, and Friday is the most popular day for ted talks published day.

Below are scatterplots with LOESS flexible regression in white and linear regression in pink, to see how different would a linear shape look from a flexible moving average. This shows us that usually, except for in the tales of the distributions of these variables, where there are only a couple of outliers' data, the linear model described the relationship somewhat well.









Surprisingly, duration had almost no consistent correlation with number of views; except for the fact that most popular talks were closer to 8-20 minutes. Number of comments is, obviously, very well correlated with number of views and so does number of languages – they all come from having many viewers. Thus, it is not "fair" to predict views based on these factors, and in the real world, we couldn't use these parameters to predict, since they **are not causes** for more views, but they are also a result of many views, and a cause in a reinforcing feedback loop: the more comments, the more engaged the community is around the talk and likelier to spread; the more languages, the more viewers can watch; and the more viewers, the more audience there is to comment and translate. The rest had a small linear effect, where that didn't deviate much, though small.[2]

[2] The pink lines are linear regressions while the white lines are LOESS. It seems that none of these loose much information by a linear regression versus a LOESS regression, which is arbitrarily flexible and would reveal a clear non-linear shape. while some of them do have nonlinear shapes - from a closer look, it is only in the tail where data is scarce and it is biased by the few data points there and some outliers (as in the Comments correlation). Therefore, inputting the regressor as a linear fit might be sufficiently explanatory.

MODELS AND RESULTS

All Models Compared

	Dependent variable: views							
	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
comments	4,698.788***				4,044.862***	4,071.480***	4,076.027***	3,931.536***
	(148.571)				(151.374)	(151.444)	(151.858)	(157.090)
languages		98,655.110***			60,650.310***	62,446.000***	61,949.050***	68,222.950***
		(4,792.403)			(4,468.592)	(4,505.978)	(4,660.659)	(4,986.410)
duration			325.599**					408.844***
			(132.184)					(117.251)
weekend				-836,986.200***				
				(243,014.000)				
num_tags						27,351.270***	27,156.920***	26,625.560***
						(9,536.676)	(9,549.530)	(9,529.882)
weekend							-86,147.350	-41,407.250
							(205,940.500)	(205,890.700)
Constant	798,186.600***	-997,579.200***	1,429,187.000***	1,734,403.000***	-733,892.800***	-993,507.000***	-975,622.200***	-1,455,238.000***
	(50,681.560)	(138,743.900)	(119,912.200)	(50,472.810)	(123,037.900)	(152,609.000)	(158,508.700)	(209,605.700)
Observations	2,550	2,550	2,550	2,550	2,550	2,550	2,550	2,550
R2	0.282	0.143	0.002	0.005	0.330	0.332	0.333	0.336
Adjusted R2	0.282	0.142	0.002	0.004	0.330	0.332	0.331	0.334
Residual Std. Error	2,117,652.000 (df = 2548)	2,313,944.000 (df = 2548)	2,496,000.000 (df = 2548)	2,493,173.000 (df = 2548)	2,045,392.000 (df = 2547)	2,042,496.000 (df = 2546)	2,042,827.000 (df = 2545)	2,038,364.000 (df = 2544)
F Statistic	1,000.232*** (df = 1; 2548)	423.772*** (df = 1; 2548)	6.068** (df = 1; 2548)	11.862*** (df = 1; 2548)	628.185*** (df = 2; 2547)	422.720*** (df = 3; 2546)	316.981*** (df = 4; 2545)	257.128*** (df = 5; 2544)
Note:								*p<0.1; **p<0.05; ***p<0.01

Chosen model is the last model since it had the best explanatory power in terms of R squared, adjusted R squared, p value and F-statistic, although it had only marginal improvements over model (5) with only comments and languages translated.

$$Model\ 5{:}\ Y(views) = \beta_0 + \beta_1 comments + \beta_2 languages + \epsilon$$

$$Model\ 8{:}\ Y(views) = \beta_1 comments + \beta_2 languages + \beta_3 numtags + \beta_4 isweekend + \beta_5\ duration + \epsilon$$

For predicting purposes, I would choose model 8 with all variables. For explanatory purposes, I would choose model 5 to explain that comments and languages are by far the most correlated with views and explain most of its variance.

Model 5 suggests that every additional comment is associated with 4,044 more views (p-value under 0.01) and that every additional language translated is associated with 60,650 more views (p-value under 0.01). However, the constant is negative (-733) views, which makes no sense, but that comes with the restriction of a linear model. These together explained 0.33 of the variance (both R-squared and adjusted R-squared). The F-statistic

$$\boldsymbol{Y(views) = -733 + 4044 * comments + 60650 * languages}$$

However, adding all the other variables into model 8 improved slightly the R-squared to 0.336 and Adjusted R-squared to 0.334. So, if we are after accuracy for prediction, I would use this latter model:

$$\boldsymbol{Y(views) = -1455238 + 3931 * comments + 68222 * languages + 408\ duration + 26625 * numtags + -41407 * isweekend}$$

The results, and particularly model 8, show overall significance. Most variables show significance, although weekend does not, but adding it still improved the explanatory power slightly, so I'm keeping it. F-statistic is relatively lower, and R and R-squared are not great at 0.336 and 0.334 respectively, but the best performance out of this set of models. The constant decreased much more, giving more power to the variables to raise the predicted view count. The coefficients (estimation of the effect) of comments decreased from 4044 to 3931 and was redistributed to higher coefficient for the number of languages and new coefficients for the newly added variables: 408 more views for every additional second, 26625 more views for every additional tag, and this is compensated by reducing the predicted number of views by 41407 if it was published on a weekend.

CONCLUSION AND IMPLICATIONS

For conclusion, this very limited model does not convey causal relationship well because the fundamental problem of causal inference is not well addressed with these variables, and these predictors are *not independent from the y variable,* but they are highly related (mostly comments and number of languages which are the best predictors, naturally. I don't believe that with these available numerical predictors we could have reached a causal inference. Next attempts might use the transcription of the talk to analyze the content, or audio to analyze the level of clapping, or the visuals in the talk and the clothing of the speaker to better predict using the content of the talk.

Implications

However, we can see some correlations, even if not causal. So, if you want a higher number of views for a talk, it would be likelier if you:

1. **Increase the number of comments in a talk!** (get all of your friends to comment and discuss)
2. **Increase the languages translated!** (get your friends or freelancers to translate)
3. **There is no need to make it too short!** (probably only works if the talk is really good)
4. **Tag it with more topics**!
5. **Whatever you do – do NOT publish it on a weekend**. **Publish it on a Friday**!

So, go increase your TED Talk's view count and comment if these strategies worked or not! (and tell me about it, since that would be a helpful small experiment which could reaffirm or reject these results!)

APPENDIX: TED TALKS ANALYSIS - FULL R MARKDOWN NOTEBOOK

Tomer Eldor

1/26/2018

INTRO

Let's examine a Ted Talks dataset. This Dataset, coming from Kaggle.com, contains information on 2550 "observations" - each observation being a ted talk from TED.com. Let's explore this dataset, see some of the propreties of the distribution of attributes of Ted Talks, and see if we can associate which of these paramteres are associated with (or even potentially "cause") higher views count for Ted Talks!

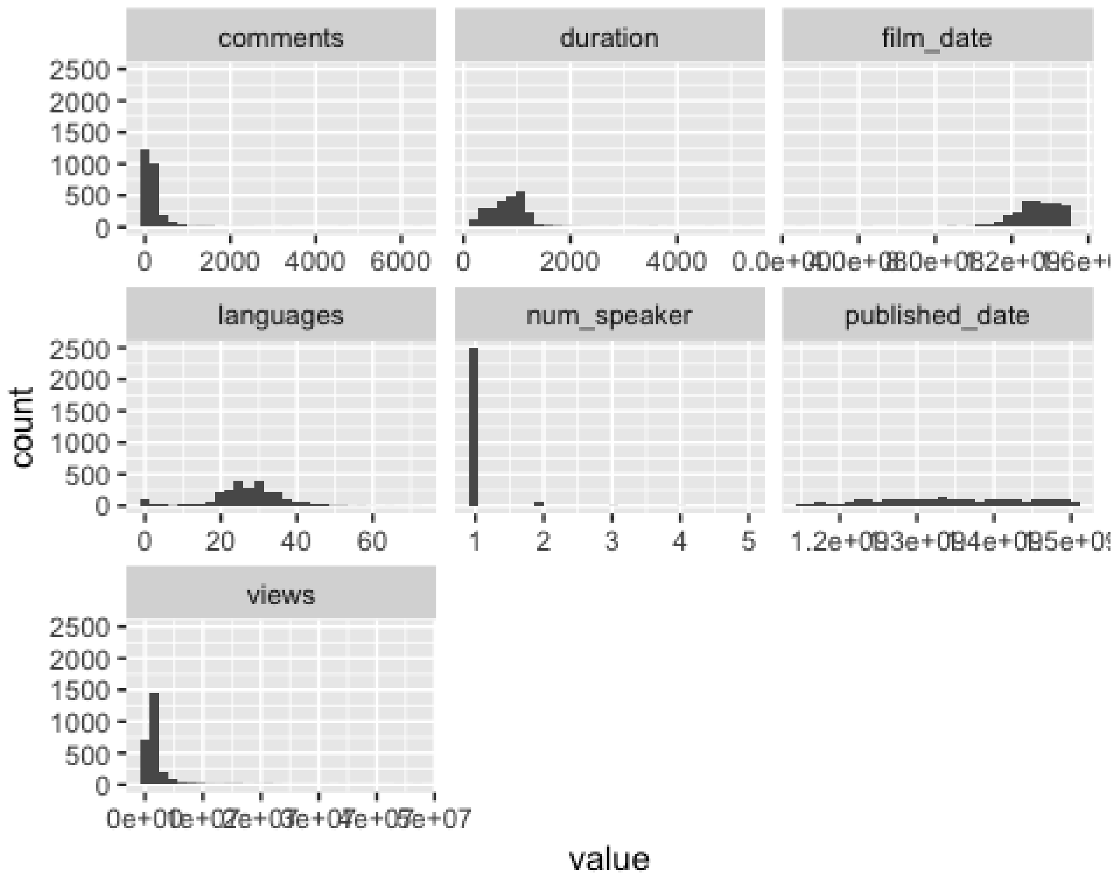
LOADING DATASET AND LIBRARIES

```
library(ggplot2) # Data visualisation
library(reshape2)
library(corrplot)
library(dplyr)
library(stringr) # String manipulation
library(anytime)
library(data.table)

ted = read.csv("./data/ted_main.csv",header=TRUE,stringsAsFactors = TRUE)
transcripts=read.csv("./data/transcripts.csv",header=TRUE,stringsAsFactors =
FALSE)
```

SUMMARY STATISTICS

Here are some in a Summary Statistics - a few tables and appropriate graphs that describe the data.

```
print(summary(ted))

melted_ted <- melt(ted)
ggplot(data = melted_ted, mapping = aes(x = value)) +
    geom_histogram(bins = 30) +  # 30 bins represented the distribution well
from trying values between 10 and 100, and is also the minimum for normal dis
tribution so it can show well if we'd have a normal distribution.
  # tried to force non-scientific notation, but the numbers are too long to r
epresent, so I removed it.
   #scale_x_continuous(labels = function(x) format(x, scientific = FALSE)) +

    facet_wrap(~variable, scales = 'free_x')
```



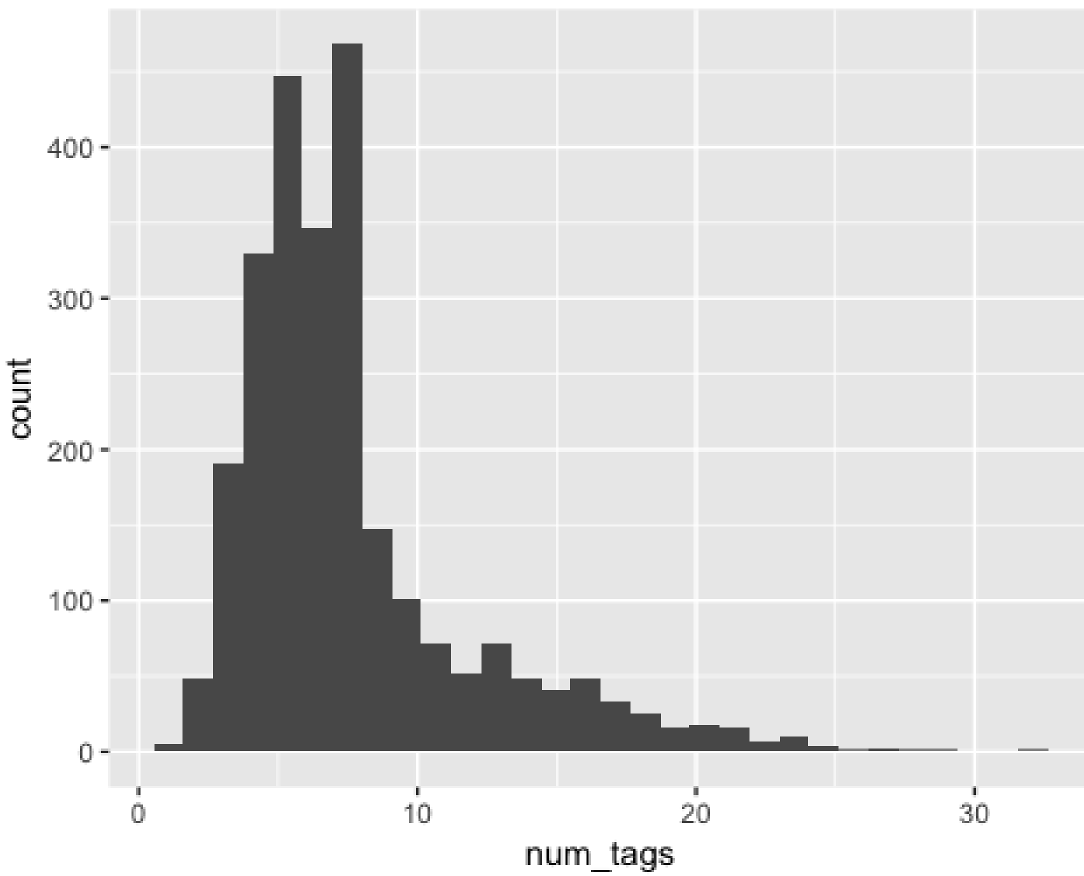
CONVERTING DATES INTO VARIABLES

```
ted$date_pub = anydate(ted$published_date)
ted$month = month(ted$date_pub)
ted$year = year(ted$date_pub)
ted$day = weekdays(ted$date_pub,abbreviate = TRUE)
head(ted,3)
```

TRYING TO CONVERT TAGS

```
typeof(ted$tags[1])

## [1] "integer"

split(as.character(ted$tags[1]),", ")

## $`, `
## [1] "['children', 'creativity', 'culture', 'dance', 'education', 'parentin
g', 'teaching']"
```

```r
#splitting did not work well. I'll just use the count of how many tags for no
w
library(stringr)
ted$num_tags <- str_count(ted$tags, ",") +1 #counting how many tags by counti
ng the number of commas and adding one. I verified it worked well.
# quick histogram of the number of tags
qplot(x=num_tags,data=ted)
```



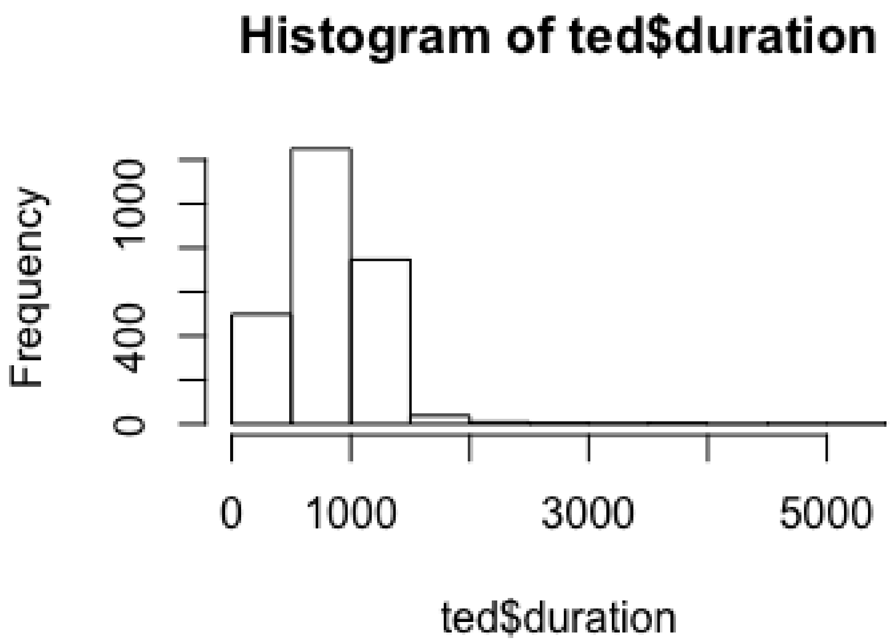
Most frequent numbers of tages are around 4-7 tags. Since the distribution is skewed, regression might work best over a polynomial or factored (cateogircal) representation of this. I'll inspect this later

HISTOGRAMS IN MORE DETAIL

DISTRIBUTION OF THE DURATION OF A TALK

```r
median_duration <- median(ted$duration)
cat("Median number of duration: ", median(ted$duration))

## Median number of duration:  848

cat("Mean number of duration: ", mean(ted$duration))
```

```
## Mean number of duration:  826.5102

# simple r histogram:
hist(ted$duration)
```



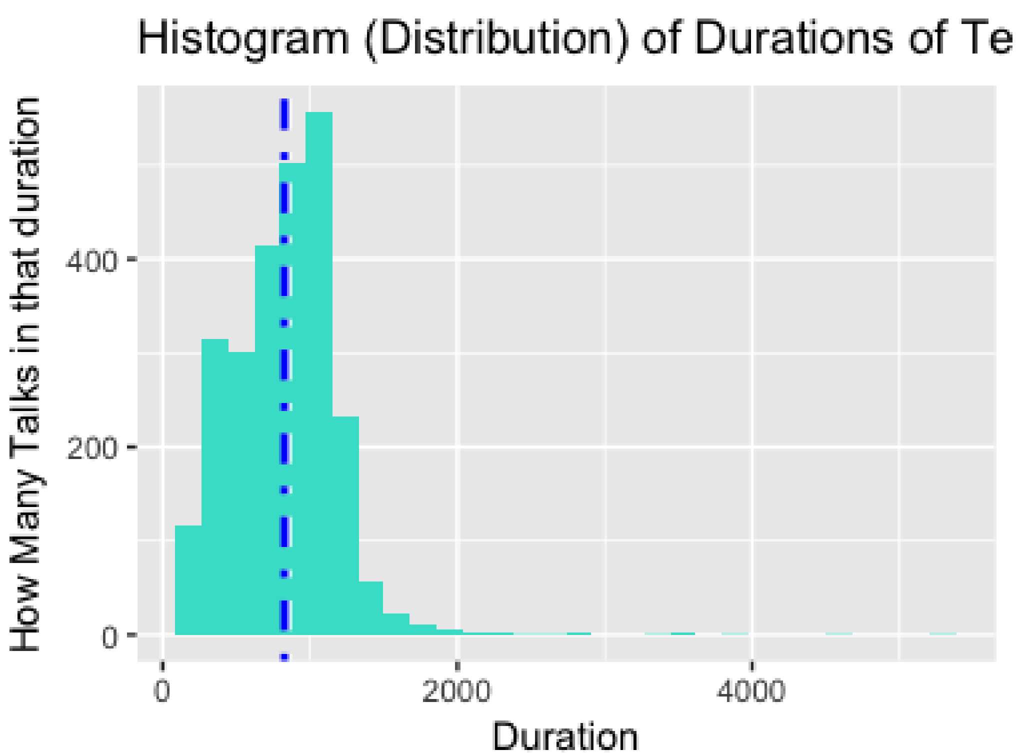

```
# a nicer histogram using ggplot, also adding median number of duration line
duration_hist = ggplot(ted,aes(duration,..count..)) +
  geom_histogram(fill="turquoise") +
  labs(x="Duration",y="How Many Talks in that duration",title="Histogram (Dis
tribution) of Durations of TedTalks") +
  #scale_x_continuous(limits=c(0,1500),breaks=seq(0,1500,150)) +
  geom_vline(aes(xintercept = median(ted$duration)),linetype=4,size=1,color="
white") +
  geom_vline(aes(xintercept = mean(ted$duration)),linetype=4,size=1,color="bl
ue")
duration_hist
```

DISTRIBUTION OF THE DURATION OF A TALK

```
median_duration <- median(ted$duration)
cat("Median number of duration: ", median(ted$duration))

## Median number of duration:  848
```

```
cat("Mean number of duration: ", mean(ted$duration))

## Mean number of duration:  826.5102

# simple r histogram:
hist(ted$duration)
```

```
# a nicer histogram using ggplot, also adding median number of duration line
duration_hist = ggplot(ted,aes(duration,..count..)) +
  geom_histogram(fill="turquoise") +
  labs(x="Duration",y="How Many Talks in that duration",title="Histogram (Dis
tribution) of Durations of TedTalks") +
  #scale_x_continuous(limits=c(0,1500),breaks=seq(0,1500,150)) +
  geom_vline(aes(xintercept = median(ted$duration)),linetype=4,size=1,color="
white") +
  geom_vline(aes(xintercept = mean(ted$duration)),linetype=4,size=1,color="bl
ue")
duration_hist
```

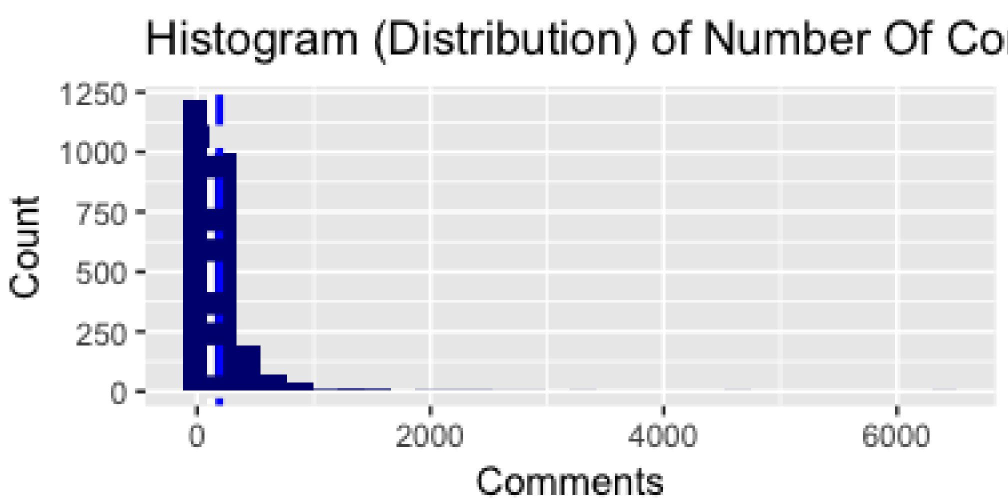


DISTRIBUTION OF THE NUMBER OF COMMENTS (IN THE DISCUSSION) PER TALK

```
median_comments <- median(ted$comments)
cat("Median number of comments: ", median(ted$comments))
```

```
## Median number of comments:  118

cat("Mean number of comments: ", mean(ted$comments))

## Mean number of comments:  191.5624

# simple r histogram:
hist(ted$comments)
```

```
# a nicer histogram using ggplot, also adding median number of comments line
comments_hist = ggplot(ted,aes(comments,..count..)) +
  geom_histogram(fill="navy") +
  labs(x="Comments",y="Count",title="Histogram (Distribution) of Number Of Co
mments") +
  #scale_x_continuous(limits=c(0,1500),breaks=seq(0,1500,150)) +
  geom_vline(aes(xintercept = median(ted$comments)),linetype=4,size=1,color="
white") +
  geom_vline(aes(xintercept = mean(ted$comments)),linetype=4,size=1,color="bl
ue")

comments_hist
```



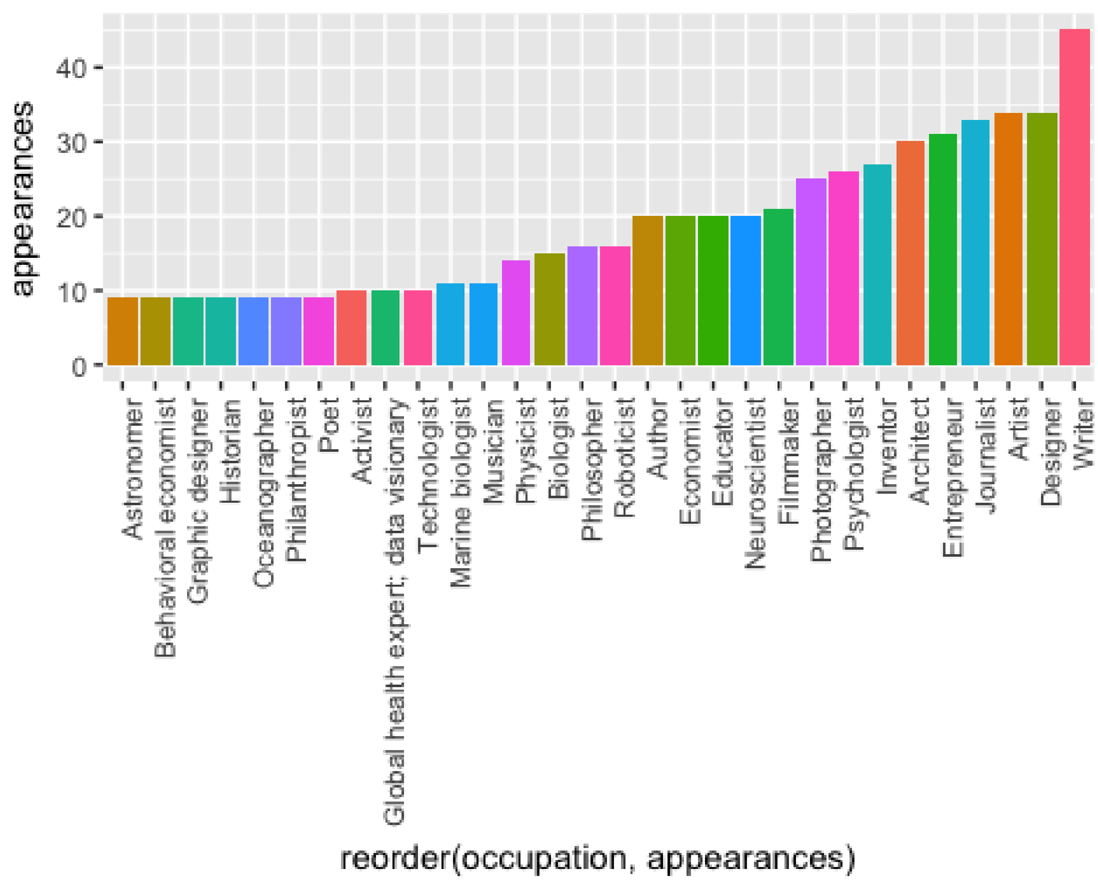
The number of comments are strongly skewed left towards 0 comments. Therefore, their mean (191) is not the most representative, but median is more representative: 118.

Let's explore occupations. Let's first see 10 most popular occuptions

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

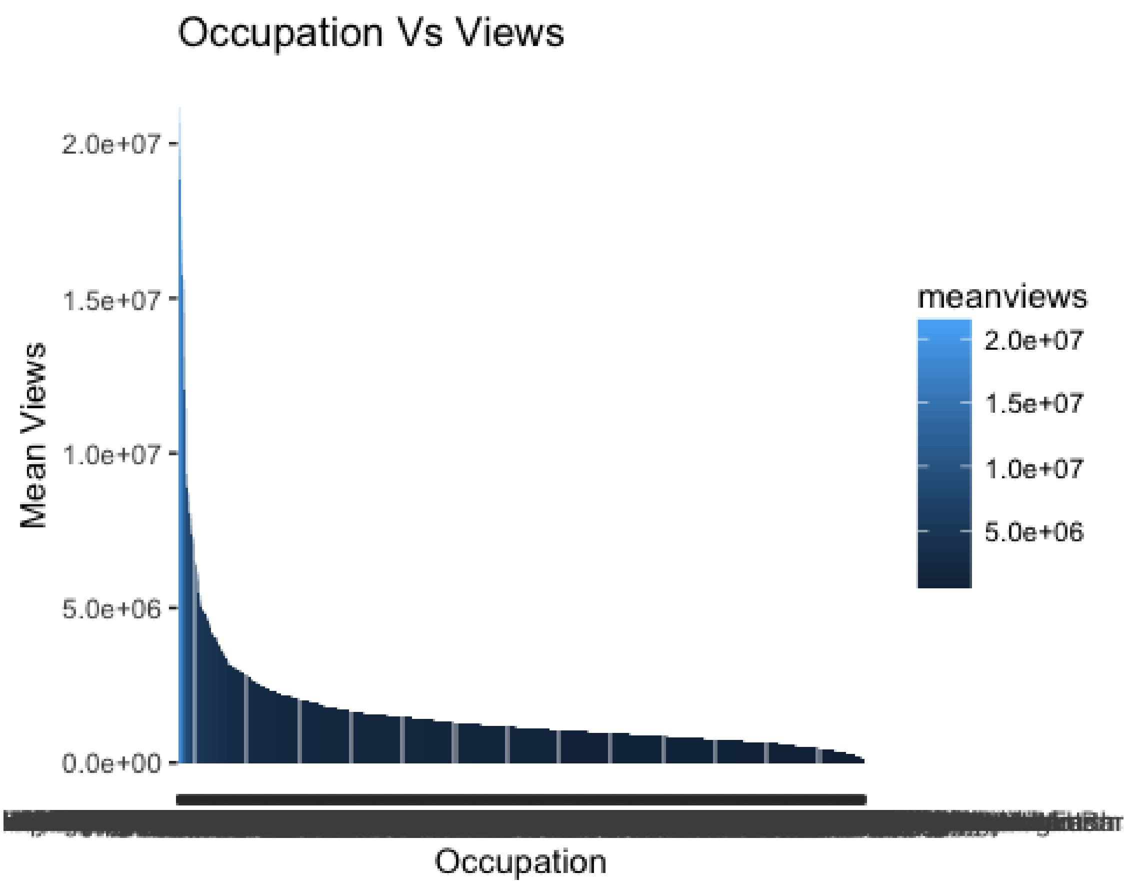
```
##      occupation appearances
## 1        Writer          45
## 2        Artist          34
## 3      Designer          34
## 4    Journalist          33
## 5  Entrepreneur          31
## 6     Architect          30
## 7      Inventor          27
## 8  Psychologist          26
## 9  Photographer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

```
ggplot(head(occupation_df,30),
       aes(x=reorder(occupation, appearances),
           y=appearances, fill=occupation)) +
  theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
  geom_bar(stat="identity") +
  guides(fill=FALSE)
```




```
ted_occupation_summary = ted %>% group_by(speaker_occupation) %>% summarise(m
eanviews=mean(views)) %>% arrange(desc(meanviews))
```

```
ggplot(ted_occupation_summary,
  aes(factor(speaker_occupation,levels=speaker_occupation), meanviews,fill=me
anviews))+
  geom_bar(stat="identity")+
  labs(x="Occupation" ,y="Mean Views",title="Occupation Vs Views")
```

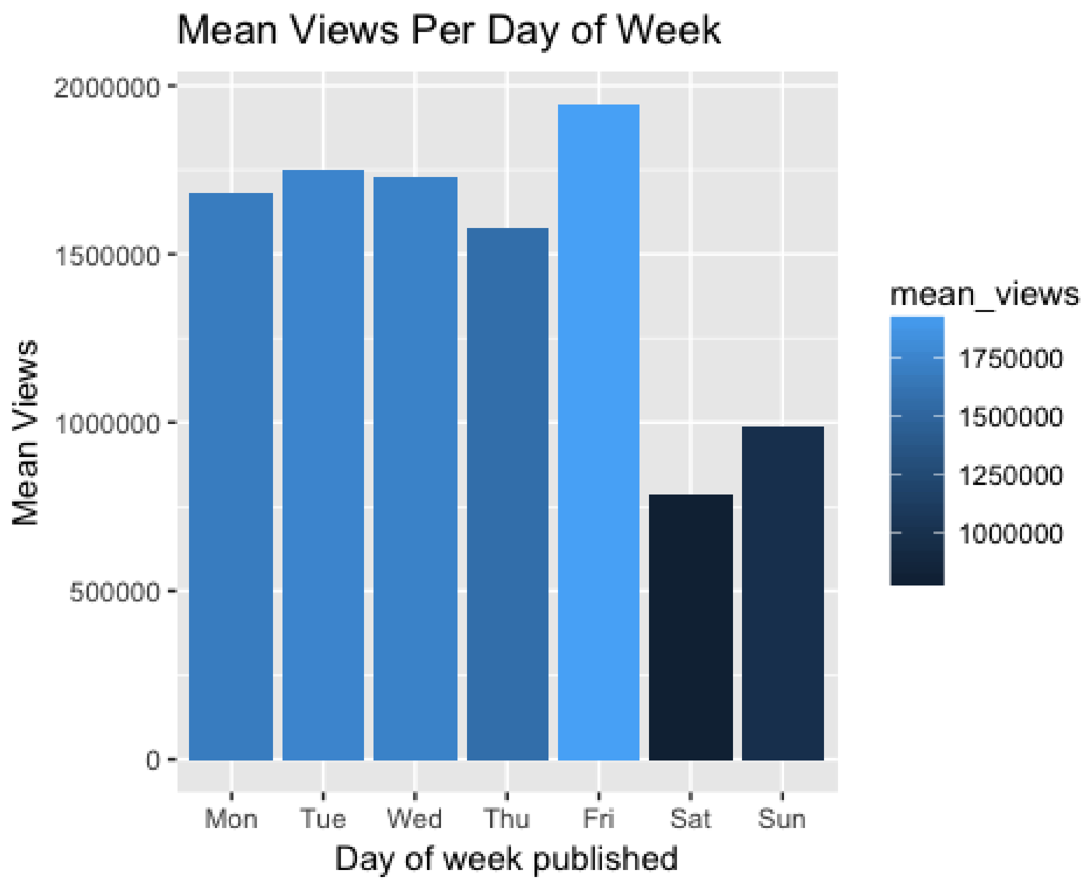



```
  #scale_fill_brewer(name="Occupation",palette = "Set2")#+
  #scale_y_discrete(labels=scales::comma)
```

Days of week Published vs Views

```
# group by day and count total views, using dpyler
ted_by_day = ted %>% group_by(day) %>% summarise(mean_views=mean(views)) %>%
arrange(desc(mean_views))
ted_by_day$day <- factor(ted_by_day$day, levels = c("Mon","Tue","Wed","Thu","
Fri","Sat","Sun"))

ggplot(data = ted_by_day, aes(x = factor(day), y = mean_views, fill = mean_vi
ews)) +
```

```
  geom_bar(stat="identity") +
  labs(x="Day of week published",y="Mean Views",title="Mean Views Per Day of
Week")
```

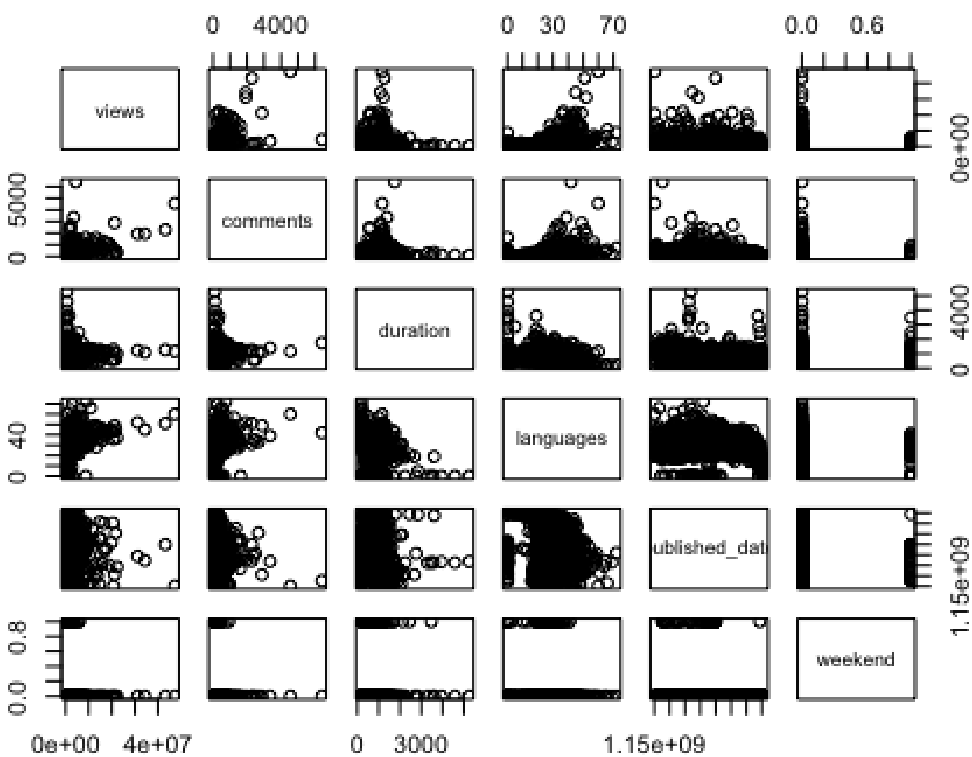


So day of week does seem to have some association with average (mean) views! Ted Talks published on weekends seem to have much less views, with Saturday being the lowest, and Friday is the most popular day for ted talks published day. Since it seems that the major effect is "Weekend or not", I'm creating a binary dummy variable for is it weekend or not to regress upon later.

```
# creating a is_weekend variable
library(chron)
ted$weekend <- as.numeric(is.weekend(ted$date_pub))

colnames(ted)

##  [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"              "date_pub"
```

```
## [19] "month"                "year"                 "day"
## [22] "num_tags"             "weekend"
```

```
col_numeric =  c(17,1,3,6,10,23) # comments,duration,film_date,languages,view
s)
ted_numeric = ted[,col_numeric]
pairs(ted_numeric)
```

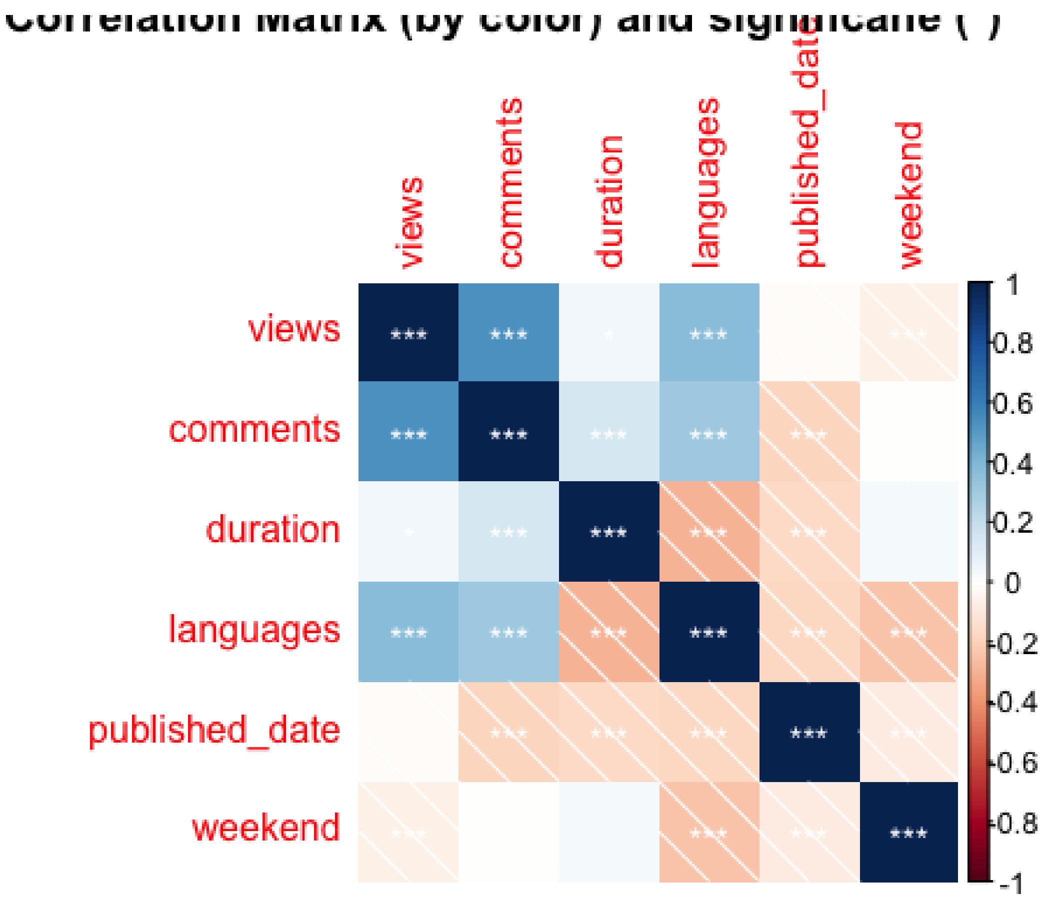


We see some correlations there but not many clear ones; between views, comments and languages. No clear corerlation between views and duration or published date.

```
cor_matrix <- cor(ted_numeric)
res <- cor.mtest(ted_numeric, conf.level = .95)
res

## $p
##               [,1]          [,2]         [,3]         [,4]         [,5]
## [1,]  0.000000e+00 1.803322e-185 1.383469e-02 3.140578e-87 3.657160e-01
## [2,] 1.803322e-185  0.000000e+00 9.558285e-13 3.950633e-61 2.869772e-21
## [3,]  1.383469e-02  9.558285e-13 0.000000e+00 1.279484e-52 2.819913e-17
## [4,]  3.140578e-87  3.950633e-61 1.279484e-52 0.000000e+00 2.370426e-18
## [5,]  3.657160e-01  2.869772e-21 2.819913e-17 2.370426e-18 0.000000e+00
```

```
## [6,]  5.820279e-04  6.932328e-01 1.003110e-01 8.566019e-35 2.459542e-05
##                [,6]
## [1,] 5.820279e-04
## [2,] 6.932328e-01
## [3,] 1.003110e-01
## [4,] 8.566019e-35
## [5,] 2.459542e-05
## [6,] 0.000000e+00
##
## $lowCI
##              [,1]        [,2]         [,3]       [,4]        [,5]
## [1,]  1.000000000  0.50247793  0.009942831  0.3438467 -0.05669662
## [2,]  0.502477926  1.00000000  0.102436668  0.2829638 -0.22314183
## [3,]  0.009942831  0.10243667  1.000000000 -0.3307018 -0.20382475
## [4,]  0.343846739  0.28296377 -0.330701766  1.0000000 -0.20925668
## [5,] -0.056696624 -0.22314183 -0.203824754 -0.2092567  1.00000000
## [6,] -0.106608323 -0.04661768 -0.006273864 -0.2764461 -0.12185536
##              [,6]
## [1,] -0.106608323
## [2,] -0.046617685
## [3,] -0.006273864
## [4,] -0.276446094
## [5,] -0.121855358
## [6,]  1.000000000
##
## $uppCI
##             [,1]       [,2]        [,3]       [,4]        [,5]        [,6]
## [1,]  1.00000000  0.5582501  0.08739150  0.4104235  0.02091130 -0.02933472
## [2,]  0.55825006  1.0000000  0.17853504  0.3527427 -0.14818904  0.03101040
## [3,]  0.08739150  0.1785350  1.00000000 -0.2598468 -0.12833640  0.07127682
## [4,]  0.41042349  0.3527427 -0.25984683  1.0000000 -0.13391282 -0.20328629
## [5,]  0.02091130 -0.1481890 -0.12833640 -0.1339128  1.00000000 -0.04476215
## [6,] -0.02933472  0.0310104  0.07127682 -0.2032863 -0.04476215  1.00000000
```

```
corrplot::corrplot(cor_matrix,method="shade",bg="white",title="Correlation Ma
trix (by color) and significane (*)",
                   p.mat = res$p,  sig.level = c(.001, .01, .05), pch.cex = .
9, insig = "label_sig", pch.col = "white")
```

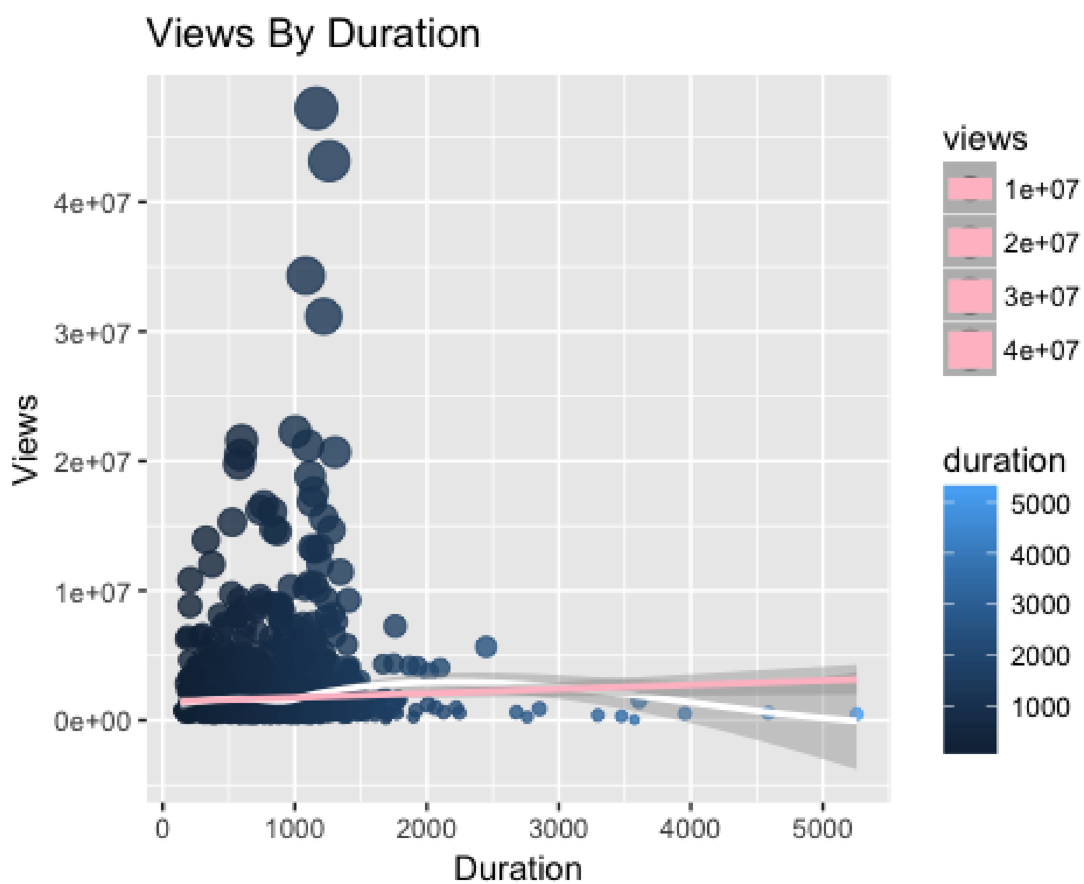


These correlation plots and correlation matrix show us the following conclusions: * There is a relatively higher positive correlation between number of comments and views, which makes sense (more audience, more comments); * Some positive correlation between number of languages of translation and number of views (0.38) and number of comments (0.32) * Small negative correlation between duration and number of languages; the shorter the talk, the more tranlsated languages there are, probably because it is easier to translate.

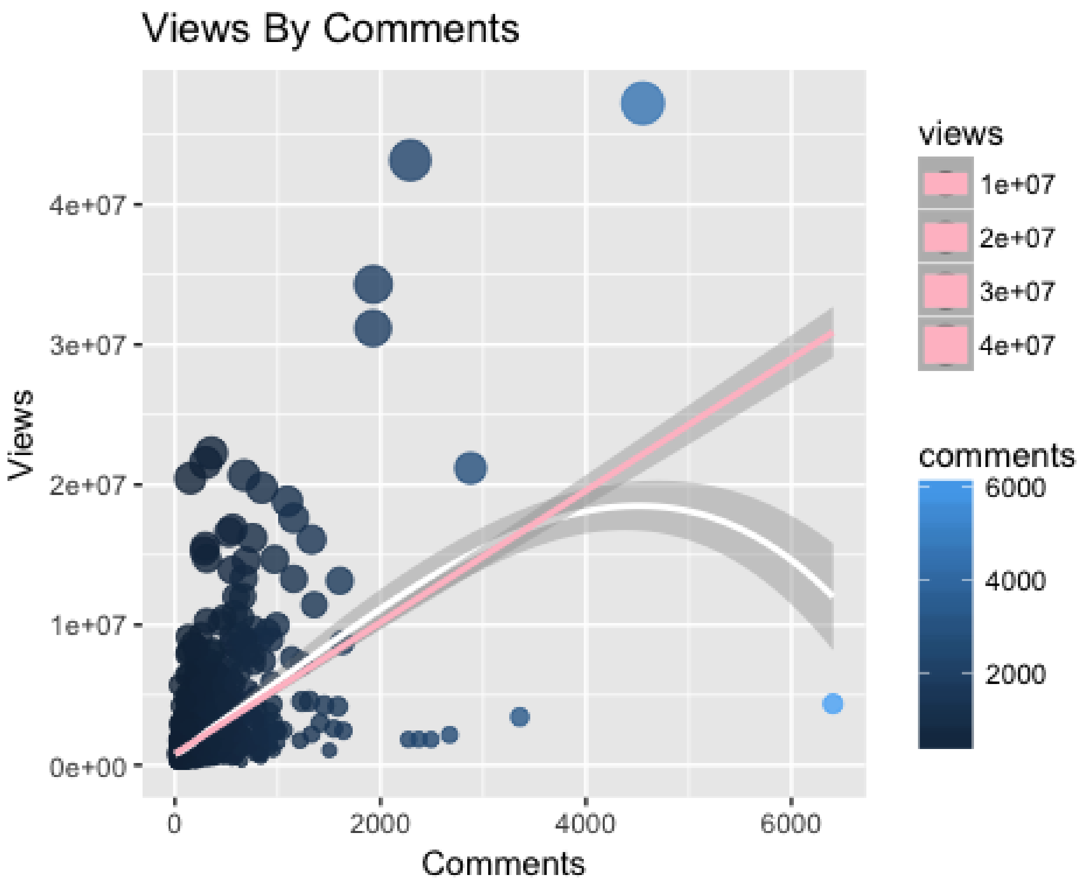
CORRELATIONS WITH VIEWS AND BETWEEN PARAMETERS

Here I'll start inspecting what are the correlations between vriables and the dependent variable, and what kind of relationship would suit to include in the regression between them (linear? polynomial? factor? none?)

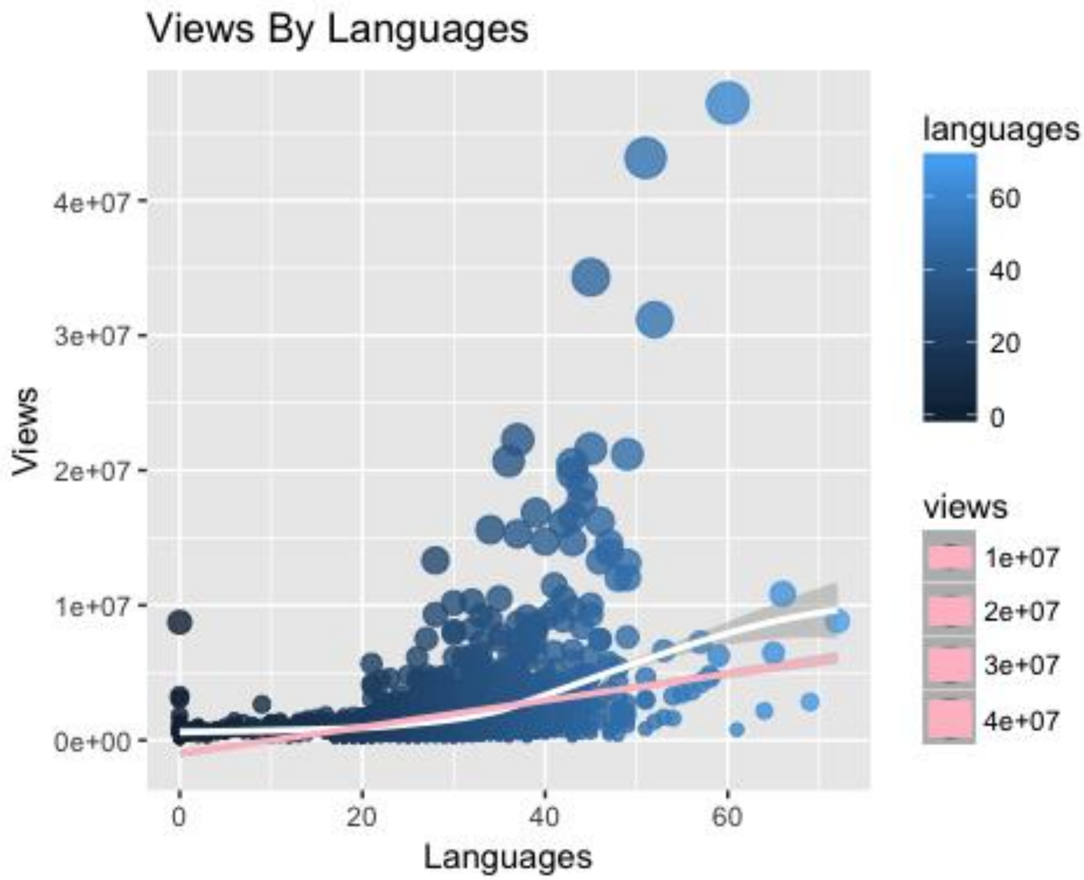
```
# Views By Duraion
ggplot(ted, aes(duration, views, size = views, col = duration)) +
  geom_point(alpha=0.8) +
  geom_smooth(method = loess, colour="White") +
  geom_smooth(method = lm, colour="Pink") +
  labs(x="Duration",y="Views",title="Views By Duration") #+
```



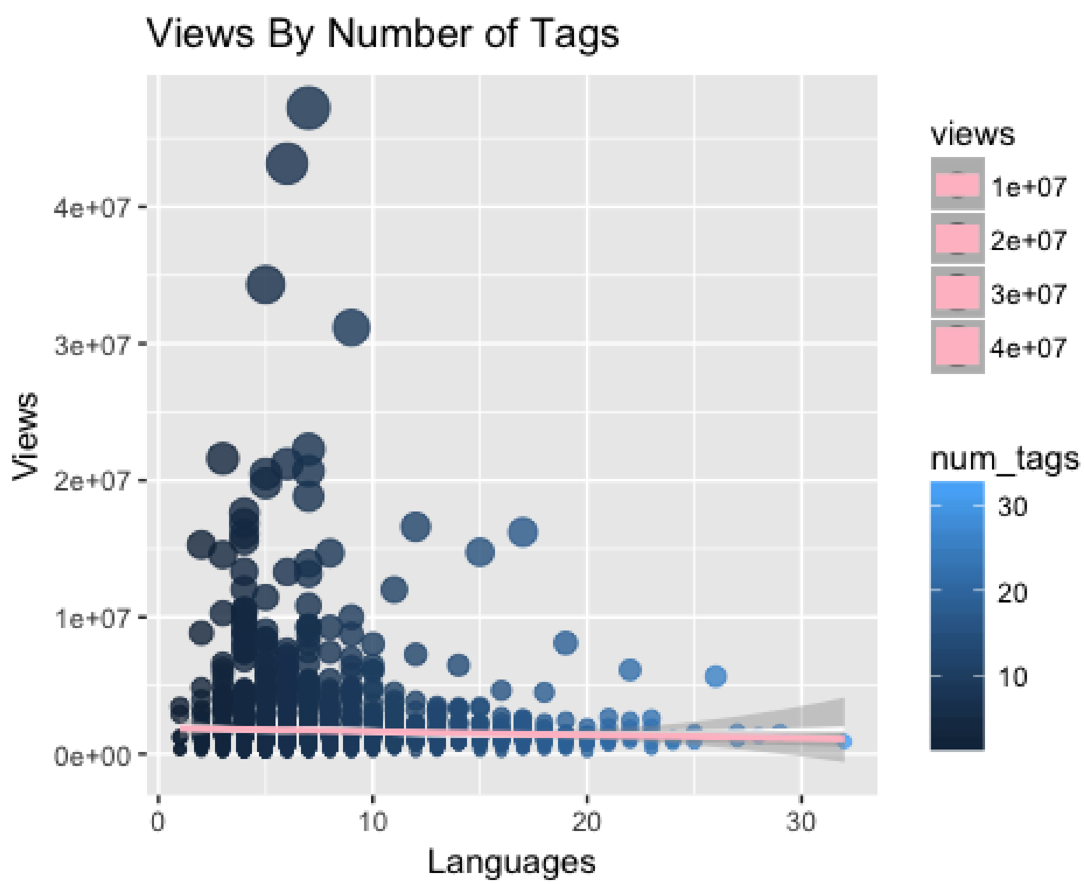

```r
# Views By Comments
ggplot(ted, aes(comments, views, size = views, col = comments)) +
  geom_point(alpha=0.8) +
  geom_smooth(method = loess, colour="White") +
  geom_smooth(method = lm, colour="Pink") +
  labs(x="Comments",y="Views",title="Views By Comments") #+
```




```
# Views By Languages
ggplot(ted, aes(languages, views, size = views, col = languages)) +
  geom_point(alpha=0.8) +
  geom_smooth(method = loess, colour="White") +
  geom_smooth(method = lm, colour="Pink") +
  labs(x="Languages",y="Views",title="Views By Languages")
```




```
# Views By Number of Tags
ggplot(ted, aes(num_tags, views, size = views, col = num_tags)) +
  geom_point(alpha=0.8) +
  geom_smooth(method = loess, colour="White") +
  geom_smooth(method = lm, colour="Pink") +
  labs(x="Languages",y="Views",title="Views By Number of Tags")
```



It seems that non of these loose much information by a linear regression versus a LOESS regression, which is arbitrarily flexible and would reveal a clear non-linear shape. while some of them do have nonlinear shapes - from a closer look, it is only in the tail where data is scarce and it is biasd by the few datapoints there and some outliers (as in the Comments correlation). Therefore, inputting the regressor as a linear fit might be sufficiently explanatory.

SPECIFY YOUR MODEL AND PERFORM A REGRESSION AND PRESENT THE RESULTS TABLE AND 1 SHORT PARAGRAPH.

```
fit1 <- lm(views  ~ comments, data = ted)
summary(fit1)

##
## Call:
## lm(formula = views ~ comments, data = ted)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -26514433   -667711   -254429    229873  31596994
##
## Coefficients:
```

```
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 798186.6    50681.6   15.75   <2e-16 ***
## comments      4698.8      148.6   31.63   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2118000 on 2548 degrees of freedom
## Multiple R-squared:  0.2819, Adjusted R-squared:  0.2816
## F-statistic:  1000 on 1 and 2548 DF,  p-value: < 2.2e-16
```

```
fit2 <- lm(views  ~ languages, data = ted)
summary(fit2)
```

```
##
## Call:
## lm(formula = views ~ languages, data = ted)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -4235090  -887422  -418475   287948 42305382
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -997579     138744   -7.19 8.47e-13 ***
## languages      98655       4792   20.59  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2314000 on 2548 degrees of freedom
## Multiple R-squared:  0.1426, Adjusted R-squared:  0.1423
## F-statistic: 423.8 on 1 and 2548 DF,  p-value: < 2.2e-16
```

```
fit3 <- lm(views  ~ duration, data = ted)
summary(fit3)
```

```
##
## Call:
## lm(formula = views ~ duration, data = ted)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2667314  -932675  -554748    28483 45418926
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept) 1429186.7   119912.2  11.919   <2e-16 ***
## duration        325.6      132.2   2.463   0.0138 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2496000 on 2548 degrees of freedom
## Multiple R-squared:  0.002376,   Adjusted R-squared:  0.001984
## F-statistic: 6.068 on 1 and 2548 DF,  p-value: 0.01383

# Adding date related information using just a weekday binary
fit4 <- lm(views ~ ted$weekend, data=ted)
summary(fit4)

##
## Call:
## lm(formula = views ~ ted$weekend, data = ted)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1683960  -934543  -576696     5606 45492707
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1734403      50473  34.363  < 2e-16 ***
## ted$weekend  -836986     243014  -3.444 0.000582 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2493000 on 2548 degrees of freedom
## Multiple R-squared:  0.004634,   Adjusted R-squared:  0.004243
## F-statistic: 11.86 on 1 and 2548 DF,  p-value: 0.000582

# weekend was significant, with a large slope (-836986) but didn't explain da
ta well alone.

fit5 <- lm(views  ~ num_tags, data = ted)
summary(fit5)

##
## Call:
## lm(formula = views ~ num_tags, data = ted)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1710677  -959176  -550980    24372 45516020
```

```
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1886196      99359   18.98   <2e-16 ***
## num_tags      -25015      11474   -2.18   0.0293 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2497000 on 2548 degrees of freedom
## Multiple R-squared:  0.001862,   Adjusted R-squared:  0.00147
## F-statistic: 4.753 on 1 and 2548 DF,  p-value: 0.02933
```

The comments and languages where significantly and (relatively highly) positively correlated with views. Day of week seeemed to have some correlation - weekend days reduced the views. I therefore just regress on a binary variable weekend or weekday, since the nunmber of the day of the week didn't have a clear correlation.

EXCLUDED VARIABLES

```
# Excluded anlaysis

summary(lm(views  ~ event, data = ted))

…..
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2448000 on 2195 degrees of freedom
## Multiple R-squared:  0.1735, Adjusted R-squared:  0.04021
## F-statistic: 1.302 on 354 and 2195 DF,  p-value: 0.0003659

# the specific ted location / event conference did not show any significant e
ffects on views

summary(lm(views  ~ ted$date_pub, data = ted))

##
## Call:
## lm(formula = views ~ ted$date_pub, data = ted)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1685735  -957633  -557580    12248 45437906
##
## Coefficients:
```

```
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2334019.30  704368.13   3.314 0.000934 ***
## ted$date_pub     -40.88      45.19  -0.905 0.365669
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2499000 on 2548 degrees of freedom
## Multiple R-squared:  0.0003212,  Adjusted R-squared:  -7.116e-05
## F-statistic: 0.8186 on 1 and 2548 DF,  p-value: 0.3657

# the date published didn't explain at all and had a very small coefiicient w
hich wasn't significant


fit_occ <- lm(views ~ factor(ted$speaker_occupation), data = ted)
summary(fit_occ)

##
## Call:
## lm(formula = views ~ factor(ted$speaker_occupation), data = ted)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -13896573   -217360         0         0  36053705
##
## Coefficients:
 …….
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2606000 on 1091 degrees of freedom
## Multiple R-squared:  0.5344, Adjusted R-squared:  -0.08788
## F-statistic: 0.8588 on 1458 and 1091 DF,  p-value: 0.9965
```

Only a few occupations were significant in predicting view counts: factor(ted$speaker_{o}ccupation)Author/educator < 2e - 16 *** factor(ted$speaker_occupation)Autonomous systems pioneer 0.012050 * factor(ted$speaker_{o}ccupation)Beatboxer 0.006905 ** factor(ted$speaker_occupation)Blogger 0.001010 ** factor(ted$speaker_{o}ccupation)Bionicsdesigner 0.022110 * factor(ted$speaker_occupation)Anthropologist, expert on love 0.023640 *

These are correlated with a number of the occupations of the most popular ted talk; for example, Author/educator is the occupation of Ken Robinson, who speaks at the most popular Ted Talk of all times and many other popular talks. However, the regression still wasn't successful, and while the R squared was 0.53, the adjusted R-squared was -0.08, because of the huge amount of predictors.

COMBINING REGRESSORS INTO MULTIPLE REGRESSION

```r
# Adding variables one-by-one
fit6 <- lm(views  ~ comments + languages, data = ted)
fit7 <- lm(views  ~ comments + languages + num_tags, data=ted)
fit8 <- lm(views  ~ comments + languages + num_tags + weekend , data=ted)
fit9 <- lm(views  ~ comments + languages + num_tags + weekend + duration, dat
a=ted)

summary(fit1) #just comments

##
## Call:
## lm(formula = views ~ comments, data = ted)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -26514433   -667711   -254429    229873  31596994
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 798186.6    50681.6   15.75   <2e-16 ***
## comments      4698.8      148.6   31.63   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2118000 on 2548 degrees of freedom
## Multiple R-squared:  0.2819, Adjusted R-squared:  0.2816
## F-statistic:  1000 on 1 and 2548 DF,  p-value: < 2.2e-16

summary(fit6)

##
## Call:
## lm(formula = views ~ comments + languages, data = ted)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -23341925   -730945   -226309    394521  31533398
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -733892.8   123037.9  -5.965 2.79e-09 ***
## comments       4044.9      151.4  26.721  < 2e-16 ***
## languages     60650.3     4468.6  13.573  < 2e-16 ***
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2045000 on 2547 degrees of freedom
## Multiple R-squared:  0.3303, Adjusted R-squared:  0.3298
## F-statistic: 628.2 on 2 and 2547 DF,  p-value: < 2.2e-16
```

```r
summary(fit7)
```

```
##
## Call:
## lm(formula = views ~ comments + languages + num_tags, data = ted)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -23464948   -738986   -220033    353133  31476369
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -993507.0   152609.0  -6.510 9.01e-11 ***
## comments       4071.5      151.4  26.884  < 2e-16 ***
## languages     62446.0     4506.0  13.858  < 2e-16 ***
## num_tags      27351.3     9536.7   2.868  0.00416 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2042000 on 2546 degrees of freedom
## Multiple R-squared:  0.3325, Adjusted R-squared:  0.3317
## F-statistic: 422.7 on 3 and 2546 DF,  p-value: < 2.2e-16
```

```r
summary(fit8)
```

```
##
## Call:
## lm(formula = views ~ comments + languages + num_tags + weekend,
##     data = ted)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -23490106   -738708   -216928    350143  31474583
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -975622.2   158508.7  -6.155  8.7e-10 ***
## comments       4076.0      151.9  26.841  < 2e-16 ***
## languages     61949.1     4660.7  13.292  < 2e-16 ***
```

```
## num_tags      27156.9     9549.5   2.844  0.00449 **
## weekend      -86147.3   205940.5  -0.418  0.67575
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2043000 on 2545 degrees of freedom
## Multiple R-squared:  0.3325, Adjusted R-squared:  0.3315
## F-statistic:   317 on 4 and 2545 DF,  p-value: < 2.2e-16
```

```r
summary(fit9)
```

```
##
## Call:
## lm(formula = views ~ comments + languages + num_tags + weekend +
##     duration, data = ted)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -23061497   -729849   -236805    353088  31452340
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1455238.1   209605.7  -6.943 4.86e-12 ***
## comments        3931.5      157.1  25.027  < 2e-16 ***
## languages      68223.0     4986.4  13.682  < 2e-16 ***
## num_tags       26625.6     9529.9   2.794 0.005247 **
## weekend       -41407.3   205890.7  -0.201 0.840626
## duration         408.8      117.3   3.487 0.000497 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2038000 on 2544 degrees of freedom
## Multiple R-squared:  0.3357, Adjusted R-squared:  0.3344
## F-statistic: 257.1 on 5 and 2544 DF,  p-value: < 2.2e-16
```

PERFORM TESTS FOR SIGNIFICANCE OF THE PARAMETERS AND PRESENT THE RESULTS (1-2 SHORT PARAGRAPHS).

```r
library(stargazer)
stargazer(fit1, fit2, fit3, fit4, fit6, fit7, fit8, fit9, type="text", title=
"All Models Compared", align=TRUE, no.space=TRUE, font.size = "footnotesize")
```

```
##
## All Models Compared
```

```
## ============================================================================
============================================================================
============================================================================
=
##
                                          Dependent variable:


##                    ----------------------------------------------------
----------------------------------------------------------------------------
----------------------------------------------------------------------------
-
##
                                                 views


##                                 (1)                        (2)
           (3)                        (4)                        (5)
            (6)                        (7)                        (8)

## ----------------------------------------------------------------------------
----------------------------------------------------------------------------
----------------------------------------------------------------------------
-
## comments                   4,698.788***
                                                           4,044.862***
       4,071.480***                4,076.027***               3,931.536***

##                               (148.571)
                                                             (151.374)
         (151.444)                  (151.858)                  (157.090)

## languages                                               98,655.110***
                                                           60,650.310***
       62,446.000***               61,949.050***              68,222.950***

##                                                           (4,792.403)
                                                             (4,468.592)
        (4,505.978)                (4,660.659)                (4,986.410)

## duration
        325.599**
                                                              408.844***
```

```
##
        (132.184)
                                                                           (117.251)

## weekend
                                          -836,986.200***


##
                                           (243,014.000)


## num_tags

       27,351.270***                27,156.920***                26,625.560***

##

        (9,536.676)                  (9,549.530)                  (9,529.882)

## weekend

                                     -86,147.350                  -41,407.250

##

                                    (205,940.500)                (205,890.700)

## Constant                        798,186.600***               -997,579.200***
    1,429,187.000***                1,734,403.000***             -733,892.800***
      -993,507.000***                 -975,622.200***              -1,455,238.000***

##                                  (50,681.560)                 (138,743.900)
      (119,912.200)                  (50,472.810)                 (123,037.900)
       (152,609.000)                 (158,508.700)                (209,605.700)

## ----------------------------------------------------------------------------
-----------------------------------------------------------------------------
-----------------------------------------------------------------------------
-
## Observations                         2,550                         2,550
          2,550                          2,550                         2,550
           2,550                          2,550                         2,550
```

```
## R2                              0.282                           0.143
          0.002                          0.005                           0.330
           0.332                          0.333                           0.336

## Adjusted R2                     0.282                           0.142
          0.002                          0.004                           0.330
           0.332                          0.331                           0.334

## Residual Std. Error  2,117,652.000 (df = 2548)  2,313,944.000 (df = 2548)
2,496,000.000 (df = 2548) 2,493,173.000 (df = 2548) 2,045,392.000 (df = 2547)
 2,042,496.000 (df = 2546) 2,042,827.000 (df = 2545) 2,038,364.000 (df = 2544
)
## F Statistic         1,000.232*** (df = 1; 2548) 423.772*** (df = 1; 2548)
 6.068** (df = 1; 2548)   11.862*** (df = 1; 2548)  628.185*** (df = 2; 2547)
 422.720*** (df = 3; 2546) 316.981*** (df = 4; 2545) 257.128*** (df = 5; 2544
)
## =============================================================================
==============================================================================
==============================================================================
=
## Note:

                                                               *p<0.1; **p<0.05; ***p<0.0
1
```