Final Project

#Use to display   
library(readr)  
# Use Convert POSTX time stamps  
library(anytime)  
library(dplyr)  
library(data.table)  
library(topicmodels) #topic modeling functions  
library(stringr) #common string functions  
library(tidytext) #tidy text analysis  
library(tidyverse) #data manipulation and visualization  
library(scales) #used for percent scale on confusion table  
library(ggthemes)  
library(ggplot2)  
library(anytime)  
library(GGally)  
library(car)  
library(ggcorrplot)  
library(dplyr)  
library(fastDummies)  
library(corrplot)  
library(ggpubr)  
#library(ggthemr)  
#ggthemr("flat")  
#set\_swatch(c(swatch(),'purple','black'))

Load the raw data

# Description of Research Question:

The primary goal of this analysis is to assess the popularity of TED Talks and identify the associated characteristics that predict this measure. Furthermore we are intrested in comparing how predictors change with respect to a TED Talks’ topic and time of publication.

# Description of scientific or research issues:

The data intially contained a large amount of variables, which made it difficult to identify the significant variables. Furthermore, after somep preliminary analysis, we discovered that the nature of the data is not normally distributed, so it limited our ability to implment statical inference. Additonally, the nature in which the variables were recorded is ambiguous which poses the risk of bias in their measurments.  
Certain variables provided limited predicting power because they are only recorded once the talk has been published. It was our view that in order to predict the popularity of a TED Talk, we would want to consider its state of characteristics before release. Furthermore, the subjective definition of popularity poses ambiguity in a definitive measure. As investigators, we have our own opinions on what constitutes popularity. However, different audiences may vary in their opinion of this.

# Description of Data:

In order to assess how various characteristics of TED talks change with time, we needed to convert the provided publish dates into a numerically readable format that would allow observations to be classifed by day, month and year. We opted not to assess the film date as we belived the publication date more accurately reflected the audience’s opinions at that time. Below we created 3 variables of day, month, and year to assess trends over time throughout our analysis.

library(lubridate)  
# Convert time stamps to readable format  
TED.df$published\_date <- anydate(TED.df$published\_date)  
TED.df$year <- year(TED.df$published\_date)  
TED.df$month <- month(TED.df$published\_date)  
#TED.df$month = as.Date(paste0(TED.df$year,"-",TED.df$month,"-01"),"%Y-%m-%d")  
#TED.df$year = as.Date(as.character(TED.df$year), format = "%Y")  
  
TED.df$day = weekdays(TED.df$published\_date,abbreviate = TRUE)  
TED.df$DaysAvailable <-difftime("2017-09-23", TED.df$published\_date,units = c("days"))  
TED.df$DaysAvailable <- as.numeric(round(TED.df$DaysAvailable))

A characteristic we hypothesized could predict popularity was the view count of related videos. Related talks appear at the bottom of the web page; if the related talks are popular (measured by the view count), there is a high chance that after watching them people would also watch related talks. We decided to take a count of views for related talks for each individual Ted Talk, and take the average of that value.

#TED.df["id"]=NA  
#num=1  
  
#TED.df$id=seq(num, 2550, 1)  
#ted\_pop\_views=arrange(TED.df, desc(views))  
total=data.frame(matrix(ncol = 3, nrow =2550))  
colnames(total)<-c("AvgRelViews", "TotalRelViews","TotalRelated")  
r<-1  
for (i in TED.df$related\_talks){  
   
 tit=str\_match\_all(i, "(viewed\_count':\\s)(\\d\*)")[[1]][,3]  
 count=0  
 total[r,2]=TED.df$views[r]  
 x<-0  
 for (k in tit){  
 find\_views=regexpr("\\d+",k)  
 x<-x+as.numeric(regmatches(k,find\_views))  
 count=count+1  
 }  
 total[r,2]= x  
 total[r,3]= count  
 x<-x/count  
   
 total[r,1]=x  
 r=r+1  
}  
TED.df$AvgRelViews <- total$AvgRelViews  
TED.df$TotalRelViews <- total$TotalRelViews  
TED.df$RelatedCount <- total$TotalRelated

Create Sentitment Scores and Counts: In order to easily evaluate how ratings influence popularity, we decided to create the sentiment score system. This was done by observing the number of times a talk has received a certain evaluation (i.e “Persuasive”, “Informative”, etc…), determining whether it is negative or positive, and counting evaluations per number of views. Furthermore we recorded the indvidual counts of these ratings to explore if a specific feeling toward a talk better predicted its popularity.

rating\_to\_list <- function(vec){  
x <- str\_extract\_all(vec, "\\w+")  
x <- unlist(str\_extract\_all(x, "\\w+"))  
return (x)  
}  
  
  
find\_rating\_count <- function(vec, rat\_tag){  
 count <- unlist(vec)[which(unlist(vec) == rat\_tag) +2]  
 return (as.numeric(count))  
}  
ratingTypes <- c("Persuasive", "Unconvincing", "Longwinded", "Informative", "Beautiful", "Confusing", "Obnoxious", "dropping", "Ingenious", "OK", "Courageous", "Inspiring", "Funny", "Fascinating")  
rateList <-lapply(TED.df$ratings, rating\_to\_list)  
  
for (i in 1:length(ratingTypes)){  
 TED.df[ratingTypes[i]] <- sapply(rateList, find\_rating\_count, rat\_tag = ratingTypes[i])  
TED.df[ratingTypes[i]] <- replace(TED.df[ratingTypes[i]],is.na(TED.df[ratingTypes[i]]),0)  
TED.df[ratingTypes[i]] <- lapply(TED.df[ratingTypes[i]],as.numeric)  
}  
colnames(TED.df)[colnames(TED.df) == 'dropping'] <- 'jawdropping'  
  
sentPOS <- c(1,0,0,1,1,0,0,1,1,0,1,1,1,1)  
sentNEG <- c(0,1,1,0,0,1,1,0,0,0,0,0,0,0)  
TED.df["sentPOS"] <- as.matrix(TED.df[,25:38]) %\*% sentPOS  
TED.df["sentNEG"] <- as.matrix(TED.df[,25:38]) %\*% sentNEG  
TED.df["netSENT"]<- TED.df$sentPOS - TED.df$sentNEG  
TED.df["CR"] <- (TED.df$comments / TED.df$views)  
  
TED.df["sentSUM"] <- TED.df$sentPOS + TED.df$sentNEG  
TED.df["SR"] <- TED.df$sentSUM / TED.df$views  
TED.df["NEGRatio"]<- TED.df$sentNEG /TED.df$sentSUM  
  
TED.df$titleLen <- lapply(TED.df$title,FUN = nchar)  
TED.df$descriptionLen <- lapply(TED.df$description,FUN = nchar)  
  
TED.df$titleLen <- as.numeric(TED.df$titleLen)  
TED.df$descriptionLen <- as.numeric(TED.df$descriptionLen)  
  
TED.df$activity <- TED.df$CR / TED.df$SR  
#TED.df["activity"] <- TED.df$sentSUM/TED.df$comments  
#TED.df["sentQUO"]<- TED.df$sentPOS / TED.df$sentNEG

The “Tags”, “Title”, and “Description” variables contain a large amount of text. We believed that the subject nature of a video could be determined by assessing a combination of these columns. In order to employ a variable which could be used to make inferences about popularity, we attempted to group videos into topic groups based on a combination of this text data.

This process was accomplished through a Latent Dirichlet Analysis Topic Model (***REFENCE***). In order to conduct this analysis, we needed to prepare the data for the LDE model. This was accomplished by removing all the unnecessary tags, such as “TED Fellows”, “TEDEx”, “TED Brain Trust”, “TED MEd” and “talks” since they do not have a significant role in grouping topics. Additionally, we removed stop words such as “and”," “if”, “or”, etc… which did not provide interactability toward a specific topic. Finally, we removed words with less than 5 occurrences amongst all video text. We then created the document term matrix with the frequency of each unique word for each video. This was then implemented into the LDA Topic Model which will be discussed later in the analysis.

While removing unuseful text we were able to create a new variable of event type between TED Talks and TEDx. We opted to use this variable rather than the event variable in the original dataset because it is not as diluted by fragmented categories like the city or event locations in the original event variable. Because TED and TEDx are intended for different kind of public (TED covers more global issues and focuses on the global community, while TEDx is typically intended for a local community and addresses local issues), the distribution of popular videos among these two groups might differ as well.

library(quanteda)  
library(topicmodels)  
#Create Document Term Matrix  
k <-2  
text <- vector(,length = length(TED.df$tags))  
eventType <- vector(,length = length(TED.df$tags))  
  
for (i in 1:length(TED.df$tags)){  
 description <- TED.df[i,2]  
 tag <- TED.df[i,14]  
 title <- TED.df[i,15]  
 if (str\_detect(tag,"TED Fellows")){  
 tag <- str\_remove\_all(tag,"TED Fellows")  
 }  
 else if (str\_detect(tag,"TEDx")){  
 eventType[i] <-"TEDx"  
 tag <- str\_remove\_all(tag,"TEDx")  
 }  
 else if (str\_detect(tag,"TED Brain Trust")){  
 tag <- str\_remove\_all(tag,"TED Brain Trust")  
 }  
 else if (str\_detect(tag,"TEDMED")){  
 tag <- str\_remove\_all(tag,"TEDMED")  
 }  
 tag <- str\_remove\_all(tag,"TED")  
 description <- str\_remove\_all(description,"talk")  
 description <- str\_remove\_all(description,"can")  
 text[i] <- paste(description,title,tag,sep = " ")  
}  
eventType[eventType == FALSE] <- "TED"  
TED.df$text <- text  
TED.df$eventType <- eventType  
  
DTM <- dfm(TED.df$text,remove\_punct = TRUE,remove = stopwords("en") )  
doc\_freq <- docfreq(DTM)  
DTM <- DTM[,doc\_freq>=5]  
#DTM <- dfm\_tfidf(DTM)  
par\_DTM <- convert(DTM, to = "topicmodels")

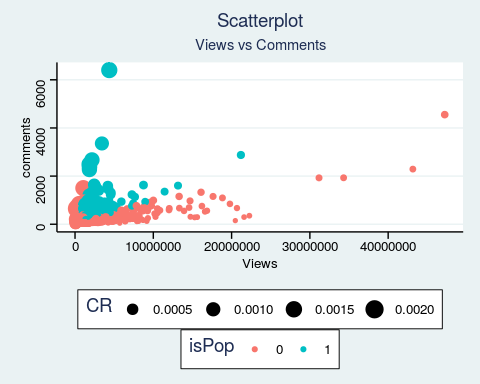
In predicting the popularity of TED talks there are a variety of ways in which popularity can be defined. Total views act as an estimate of the size of the audience the video has been able to reach. However, there is no detailed information on how long viewers watched the video for or whether they left soon after starting a video. To combat this ambiguity we opted to include a measure of user activity in our classification of popularity

We defined a new measure called the comment ratio which is the total number of comments divided by the total views of a video. The intuition behind this statistic was to judge user activity relative to the number of views. It was then decided that a talk could be deemed popular if it had both above median views and an above median comment ratio. Our logic here was, if it measured high on both these stats, it meant many individuals were viewing the video and also engaging in discussion about it.

We considered including the sentiment score calculation into this measure. However, we found that it did not offer clear differentiation between videos. we wanted the opportunity to assess the effect of individual rating types on popularity without introducing a bias towards the measure.

Implementing this classification rule led to the following segmentation of the data. As we can see from the scatterplot, there is a high correlation between views and comments. Thus, we took the comment ratio as one of our main variables to classify the video as popular or not popular. Here, the variable isPop was defined.

options(scipen = 999)  
medComments <- median(TED.df$comments)  
medView<- median(TED.df$views)  
medCR<- median(TED.df$CR)  
medComments <- median(TED.df$comments)  
medNR <- median(TED.df$NEGRatio)  
TED.df <- mutate(TED.df,isPop = ifelse((views > medView) & (CR> medCR),"1","0"))  
  
PopVar<- c("views","comments","CR","isPop")  
PopVar.df<- select(TED.df,PopVar)  
  
gg <- ggplot(TED.df,aes(x=views,y=comments)) + geom\_point(aes(col=isPop,size=CR)) + labs(subtitle="Views vs Comments",x="Views",y="comments",title="Scatterplot") + theme\_stata() + scale\_fill\_brewer(palette = "Set1")  
plot(gg)



This revealed the following structure of Popular Vs. Not Popular counts. We noticed that combining the elements of view counts and relative user activity by comment ratios distinguished a more elite class of 500 videos whose characteristics we wish to explore.

Our next goal was to determine the features that lead to these classifications. We first attempted a factor analysis of the quantitative variables to explore the possibility of variable reduction. However, as seen below this was unsuccessful in revealing a simple structure to the data with two factors explaining a small portion of the variance in both Bartlett’s and the Regression methods calculation of factors. Furthermore, when testing the null hypothesis that a sufficient number of factors was used in our factor analysis, the test yielded p-values close to zero.

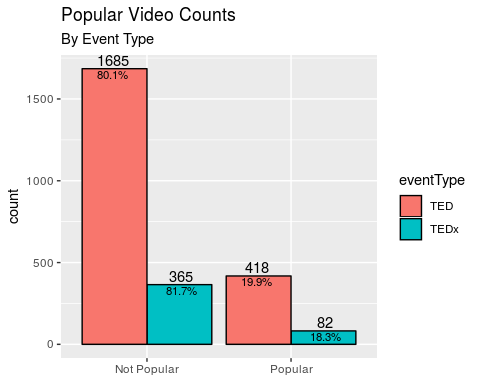
##### ------------------------------FACTOR ANALYSIS ----------------------------  
keeps <- c("isPop","AvgRelViews","RelatedCount","duration","languages","num\_speaker","DaysAvailable","titleLen","descriptionLen","CR","Persuasive","Unconvincing","Longwinded","Informative","Beautiful","Confusing","Obnoxious","jawdropping","Ingenious" ,"OK","Courageous","Inspiring","Funny","Fascinating")  
general <- select(TED.df, keeps)  
general$isPop <- as.numeric(general$isPop)  
  
  
  
fac.ted1 <- factanal(general, 2, rotation="promax", scores="Bartlett")  
#print(loadings(fac.ted))  
#plot(fac.ted$scores, pch=20)  
#par(mfrow=c(1,2))  
#qqnorm(fac.ted$scores[,1])  
#qqnorm(fac.ted$scores[,2])  
  
fac.ted2 <- factanal(general, 2, rotation="promax", scores="regression")  
print(loadings(fac.ted1),loadings(fac.ted2))

##   
## Loadings:  
## Factor1 Factor2  
## isPop 0.0   
## AvgRelViews 0.0   
## RelatedCount   
## duration   
## languages 0.0   
## num\_speaker   
## DaysAvailable 0.0   
## titleLen 0.0   
## descriptionLen 0.0   
## CR 0.0 0.0   
## Persuasive 0.8   
## Unconvincing 0.9   
## Longwinded 0.0 0.6   
## Informative 0.8   
## Beautiful 0.7 0.0   
## Confusing 0.7   
## Obnoxious 0.0 0.8   
## jawdropping 0.5   
## Ingenious 0.7   
## OK 0.5 0.0   
## Courageous 0.6   
## Inspiring 0.9   
## Funny 0.5   
## Fascinating 0.9   
##   
##   
## Loadings:  
## Factor1 Factor2  
## isPop 5.0 5.0   
## AvgRelViews   
## RelatedCount   
## duration 3.0 3.0   
## languages   
## num\_speaker   
## DaysAvailable 5.0 5.0   
## titleLen   
## descriptionLen   
## CR 3.0 3.0   
## Persuasive 0.1   
## Unconvincing 0.4   
## Longwinded 5.0 5.5   
## Informative 0.2   
## Beautiful 0.2   
## Confusing 3.0 3.0   
## Obnoxious 0.1   
## jawdropping 0.4   
## Ingenious 5.5 5.0   
## OK 0.2   
## Courageous 0.2   
## Inspiring 3.0 3.0   
## Funny 0.1   
## Fascinating 0.4   
##   
## Factor1 Factor2  
## SS loadings 141.750 141.420  
## Proportion Var 5.906 5.892  
## Cumulative Var 5.906 11.799

#plot(fac.ted$scores, pch=20)  
#par(mfrow=c(1,2))  
#qqnorm(fac.ted$scores[,1])  
#qqnorm(fac.ted$scores[,2])  
  
#promax rotation produces simple structure  
#number of speakers has been annhilated in all factors - remove from analysis  
#small covariance/correlation to the other dependent variables

Our next approach was to assess how popularity differed with different categorical variables. We first assessed how the counts of popular videos differed by eventTypes previously defined above (TED, TEDx). We did not see a sizable difference between the proportions of popular videos amongst the categories. Perhaps further segmentation later in the analysis will reveal a more interesting structure amongst these types.

percentData <- TED.df %>% group\_by(eventType) %>% count(isPop) %>% mutate(ratio=percent(n/sum(n)))  
  
ggplot(TED.df,aes(x = as.factor(isPop),fill = eventType)) + geom\_bar( position = "dodge",stat = "count",color="black") + labs(x=NULL,title ="Popular Video Counts",subtitle = "By Event Type") + geom\_text(stat = 'count',aes(label=..count..),position=position\_dodge(width=0.95), vjust=-0.25) + scale\_x\_discrete(breaks = c("0","1"),labels = c("Not Popular","Popular")) + geom\_text(data=percentData, aes(y=n,label=ratio),  
 position=position\_dodge(width=.95), vjust=1.25,size=3)

 Here, we compared the means of views, comment ratios, counts of Ratings, and counts of popular talks between the event types using MANOVA. Using Wilk’s lambda, we reject the null hypothesis that the covariance matrices are equal. Thus, inference between means is limited.

library(mvnormtest)  
data <- select(TED.df,views,CR,sentSUM,isPop,eventType)  
data$isPop <- as.numeric(data$isPop)  
data <- select(TED.df,views,CR,sentSUM,isPop,eventType)  
data$isPop <- as.numeric(data$isPop)  
res.man <- manova(as.matrix(data[,1:4]) ~ eventType,data = data)  
summary(res.man,test="Wilks")

## Df Wilks approx F num Df den Df Pr(>F)   
## eventType 1 0.99029 6.2367 4 2545 0.0000543 \*\*\*  
## Residuals 2548   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

summary(res.man)

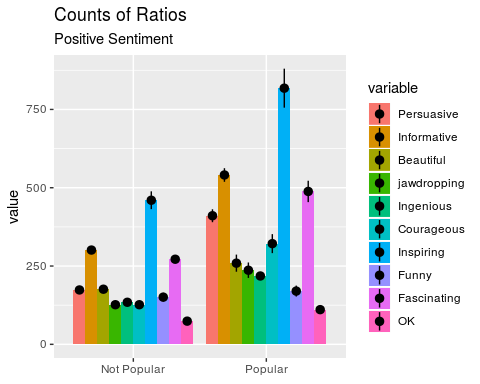
## Df Pillai approx F num Df den Df Pr(>F)   
## eventType 1 0.0097071 6.2367 4 2545 0.0000543 \*\*\*  
## Residuals 2548   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

summary.aov(res.man)

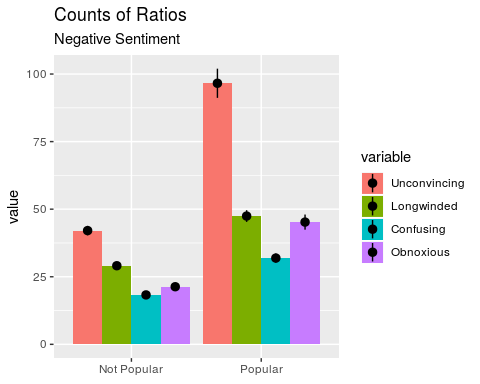
## Response views :  
## Df Sum Sq Mean Sq F value Pr(>F)  
## eventType 1 7835634935398 7835634935398 1.2554 0.2626  
## Residuals 2548 15904039857640570 6241773884474   
##   
## Response CR :  
## Df Sum Sq Mean Sq F value Pr(>F)   
## eventType 1 0.000000566 0.00000056638 22.11 0.000002711 \*\*\*  
## Residuals 2548 0.000065270 0.00000002562   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Response sentSUM :  
## Df Sum Sq Mean Sq F value Pr(>F)  
## eventType 1 4998921 4998921 0.2878 0.5917  
## Residuals 2548 44262763951 17371571   
##   
## Response isPop :  
## Df Sum Sq Mean Sq F value Pr(>F)  
## eventType 1 0.09 0.086504 0.5485 0.459  
## Residuals 2548 401.87 0.157721

We further visualized the difference of rating values between popularity by plotting the respective counts of rating types between Popular and Not Popular segments. We first highlight that the popular videos have larger counts in all rating types. This provides evidence that having larger views and comment activity corresponds with a higher number of ratings. Amongst the positive rating types, the “Inspiring” rating has a significantly greater count in both popularity groups. Amongst the negative counts, the “Uninspiring” rating shares this property of the highest count amongst popularity groups. This provides evidence of a crowding effect of perceptions by viewers. If there is already a majority of people tending towards a rating type, other viewers may also tend toward this sentiment.

library(reshape2)  
TED.df$isPop <- as.numeric(TED.df$isPop)  
basic <- c("isPop","AvgRelViews","TotalRelViews","RelatedCount","duration","languages","num\_speaker","DaysAvailable","titleLen","descriptionLen")  
Pop.basic <- select(TED.df,basic)  
sentiment <- c("isPop","Persuasive","Unconvincing","Longwinded","Informative","Beautiful","Confusing","Obnoxious","jawdropping","Ingenious" ,"OK","Courageous","Inspiring","Funny","Fascinating")  
Pop.sentiment <-select(TED.df,sentiment)  
fun\_mean <- function(x){  
 return(data.frame(y=mean(x),label=round(mean(x,na.rm=T))))}  
Pop.basic$isPop <- as.character(Pop.basic$isPop)  
Pop.sentiment$isPop <- as.character(Pop.sentiment$isPop)  
m <- melt(Pop.sentiment,id.vars='isPop',measure.vars=c("Persuasive","Informative","Beautiful","jawdropping","Ingenious" ,"Courageous","Inspiring","Funny","Fascinating","OK"))  
  
g <- ggplot(m,aes(x=isPop, y=value, fill=variable))  
g + geom\_bar(stat = "summary", fun.y = "mean",position = position\_dodge(width = 0.9)) + geom\_pointrange(stat = "summary", fun.data = "mean\_se",position = position\_dodge(width = 0.9)) + labs(x=NULL,title = "Counts of Ratios",subtitle = "Positive Sentiment") + scale\_x\_discrete(breaks = c("0","1"),labels = c("Not Popular","Popular"))



m <- melt(Pop.sentiment,id.vars='isPop',measure.vars=c("Unconvincing","Longwinded","Confusing","Obnoxious"))   
  
g <- ggplot(m,aes(x=isPop, y=value, fill=variable))  
g + geom\_bar(stat = "summary", fun.y = "mean",position = position\_dodge(width = 0.9)) + geom\_pointrange(stat = "summary", fun.data = "mean\_se",position = position\_dodge(width = 0.9)) + labs(x=NULL,title = "Counts of Ratios",subtitle = "Negative Sentiment") + scale\_x\_discrete(breaks = c("0","1"),labels = c("Not Popular","Popular"))



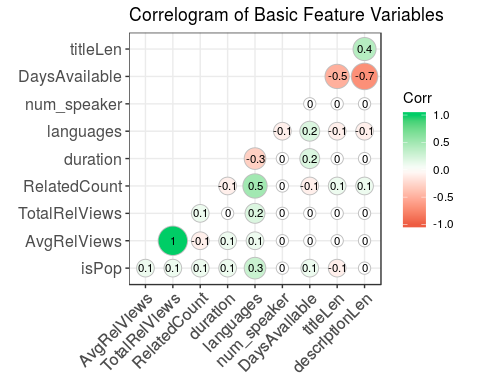
#g + geom\_boxplot(varwidth = F ) + labs(title = "Boxplot",subtitle = "Basic Varibles") + theme(axis.text.x = element\_text(angle=65, vjust=0.6)) + stat\_summary(fun.y = mean, geom="point",colour="darkred", size=3) +stat\_summary(fun.data = fun\_mean, geom="text", vjust=-0.7) + scale\_x\_discrete(breaks = c("0","1"),labels = c("Not Popular","Popular"))

In order to better understand the relative effects each rating variable count has on its respective video, we decided to convert rating counts to their proportions amongst the total number of ratings for each respective video.

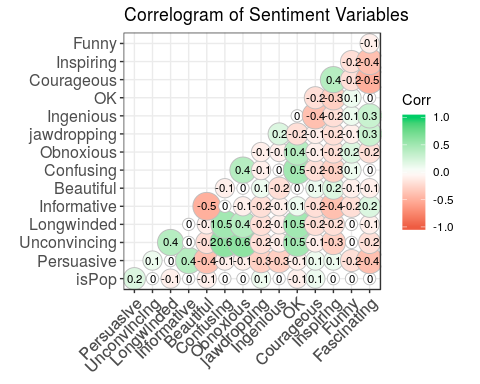
TED.df[,25:38] <- (1/TED.df$sentSUM)\*TED.df[,25:38]

To get a better sense of the relationship between popularity and the feature variables, we assessed the correlation matrix for the quantitative variables.

TED.df$isPop <- as.numeric(TED.df$isPop)   
basic <- c("isPop","AvgRelViews","TotalRelViews","RelatedCount","duration","languages","num\_speaker","DaysAvailable","titleLen","descriptionLen")  
Pop.basic <- select(TED.df,basic)  
sentiment <- c("isPop","Persuasive","Unconvincing","Longwinded","Informative","Beautiful","Confusing","Obnoxious","jawdropping","Ingenious" ,"OK","Courageous","Inspiring","Funny","Fascinating")  
Pop.sentiment <-select(TED.df,sentiment)  
  
basic.corr <- round(cor(Pop.basic[,1:10]),1)  
ggcorrplot(basic.corr, hc.order = FALSE,   
 type = "lower",   
 lab = TRUE,   
 lab\_size = 3,   
 method="circle",   
 colors = c("tomato2", "white", "springgreen3"),   
 title="Correlogram of Basic Feature Variables",   
 ggtheme=theme\_bw)



sentiment.corr <- round(cor(Pop.sentiment[,1:15]),1)  
ggcorrplot(sentiment.corr, hc.order = FALSE,   
 type = "lower",   
 lab = TRUE,   
 lab\_size = 3,   
 method="circle",   
 colors = c("tomato2", "white", "springgreen3"),   
 title="Correlogram of Sentiment Variables",   
 ggtheme=theme\_bw)

 This raises some intresting findings. Languages having the highest correlation with popularity and could indicate allowing the video to be understood by a more gloabl audience is resulting in larger groups viewing and interactig on videos.There is also a variety of minor postive and negative correlations amongst the ratings ratios. Before jumping to any conclusions it important that we consider how these effects change based on segements of categorical varibles in additon to time.

On the graph Average number views that are distributed all over years 2007 to 2018 we can see that the highest number of views occured in 2007, following by a huge decline and a small spike in 2013, again following the downwards trend to the 2018. This can be backed up by the fact that Ted platform was launched in 2007( that is why we can observe such a huge number of views), and in March 2012 Netflix announced a deal to stream series of Ted Talks(that explaines a small upward trend in the number of views, since expansion of Ted Talks to more platfoms provides access for bigger amount of population).

The graph average comments distribution per year indicates that there were major spikes of upward trend in 2011 and in 2014. Althought, the nature of the trends is not clear, we can theorize that the spike in comments could be influenced by some amount of talks that were appealing for majority of people during that time(could be caused by political or media influence). This theory could be backed up by the next graph that indicates that the most popular videos also appeared on 2011 and 2014, which proves that our choice of including the comment per view variable into our model that defines the popularity.

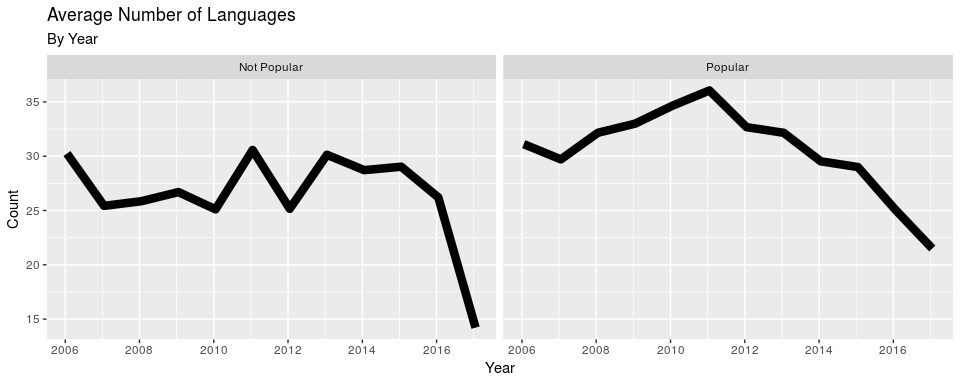
Average number of languages by years for both popular and unpopular videos show an overall decreasing trend. It can be inferred that, with increasing access to technology and English becoming a far more popular language in foreign countries, the need for the TED talks to be translated into different languages has declined. English has had great success as the lingua franca of business, travel, and international relations.

Number of views by month shows that there are less views, averaging around 300000000, in the winter months. There is an increase in views in March and a gradual decline into the summer months, ending in August. It is difficult to infer a particular trend in views by month because the audience varies so greatly. Furthermore, the count of popular and unpopular videos published by month does not vary greatly. Therefore, it cannot be inferred that number of views in any given month increases or decreases with the number of videos that were published in that month.

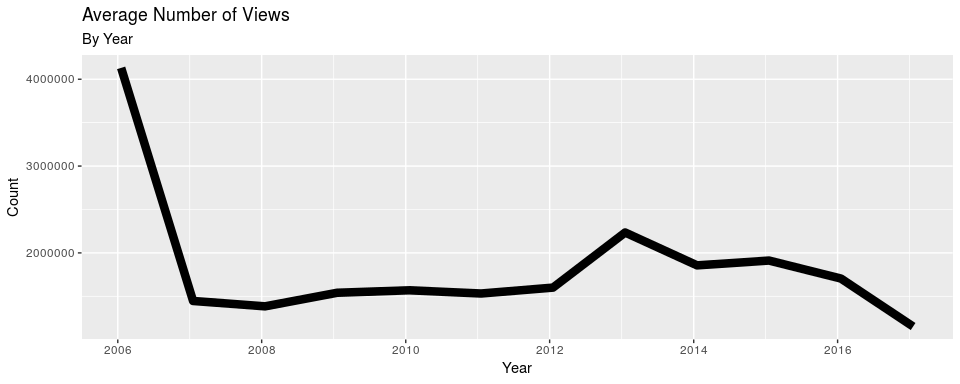
In the graph showing numbers of views by month by year, there is an upward trend of views from 2006 to 2013 and then a downward trend afterwards. In general, it seems that the month with the greatest number of views is May. This may be attributed to university students ending their winter semester in April and finding further educational value from these talks. However, as previously stated, the audience varies so greatly that it is difficult to make any specific inference about these trends without further information. Finally, the downward trend post 2013 may show that TED has declined in popularity and people are less likely to watch their videos.

The count of popular videos by month and by year shows an interesting trend. It is clear that post 2013, the number of “not popular” videos has greatly declined, to the point where they zeem virtually nonexistent in 2016 and 2017.

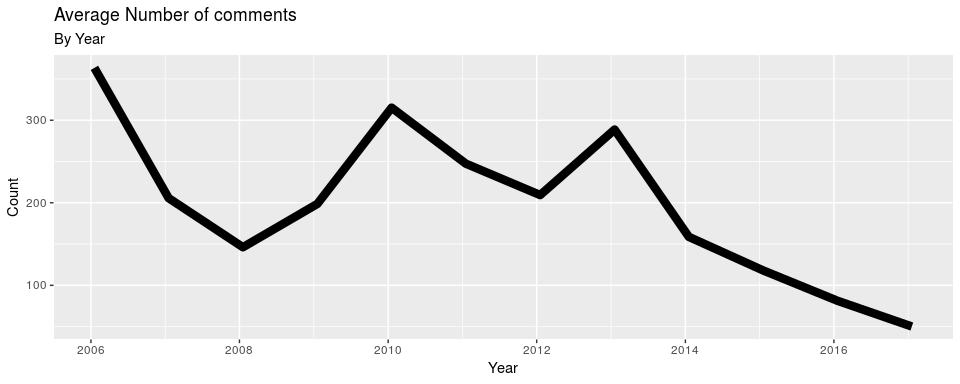
#TED.df$month = as.Date(paste0(TED.df$year,"-",TED.df$month,"-01"),"%Y-%m-%d")  
#TED.df$year = as.Date(as.character(TED.df$year), format = "%Y")  
#Year Analysis:  
# Y is numeric  
ggplot(TED.df, aes(x = as.Date(as.character(year), format = "%Y"), y = languages)) + stat\_summary(fun.y = mean, geom = "line",size=3) + labs(title = "Average Number of Languages",subtitle = "By Year",x="Year",y="Count") + scale\_x\_date(labels = date\_format("%Y")) + facet\_wrap(~mutate(TED.df,isPop=if\_else(isPop == "0","Not Popular","Popular"))$isPop)



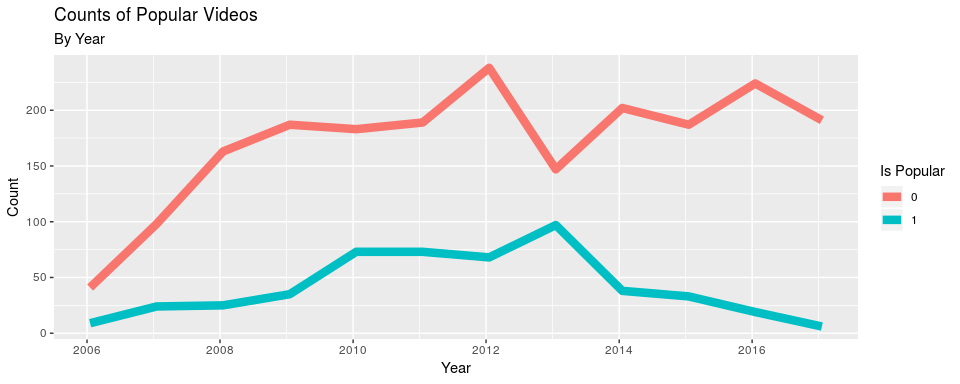
ggplot(TED.df, aes(x = as.Date(as.character(year), format = "%Y"), y = views)) + stat\_summary(fun.y = mean, geom = "line",size=3) + labs(title = "Average Number of Views",subtitle = "By Year",x="Year",y="Count") + scale\_x\_date(labels = date\_format("%Y"))



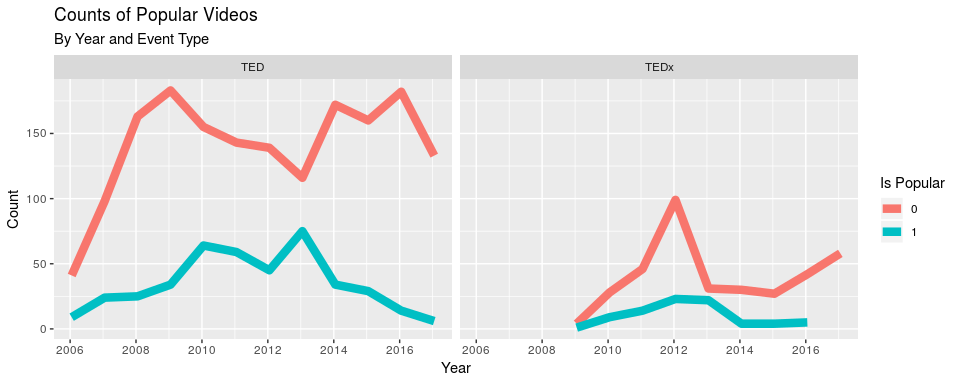
ggplot(TED.df, aes(x = as.Date(as.character(year), format = "%Y"), y = comments)) + stat\_summary(fun.y = mean, geom = "line",size=3) + labs(title = "Average Number of comments",subtitle = "By Year",x="Year",y="Count") + scale\_x\_date(labels = date\_format("%Y"))



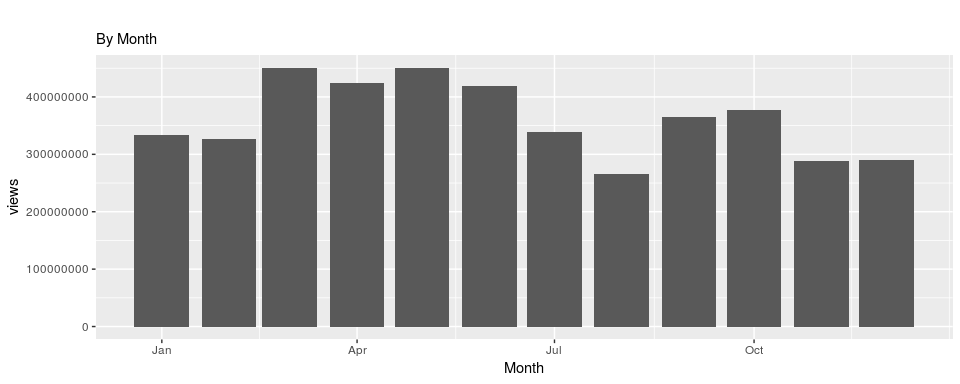
# Y is categorical  
ggplot(TED.df, aes(x = as.Date(as.character(year), format = "%Y"), color = as.factor(isPop))) + geom\_line(stat='count',size=3) + labs(title = "Counts of Popular Videos",subtitle = "By Year",x="Year",y="Count",color="Is Popular") + scale\_x\_date(labels = date\_format("%Y"))



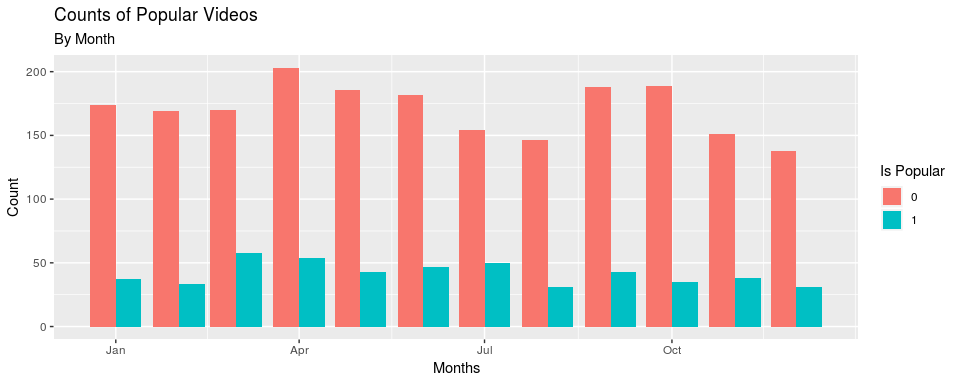
ggplot(TED.df, aes(x = as.Date(as.character(year), format = "%Y"), color = as.factor(isPop))) + geom\_line(stat='count',size=3) + labs(title = "Counts of Popular Videos",subtitle = "By Year and Event Type",x="Year",y="Count",color="Is Popular") + scale\_x\_date(labels = date\_format("%Y")) + facet\_wrap(~eventType)



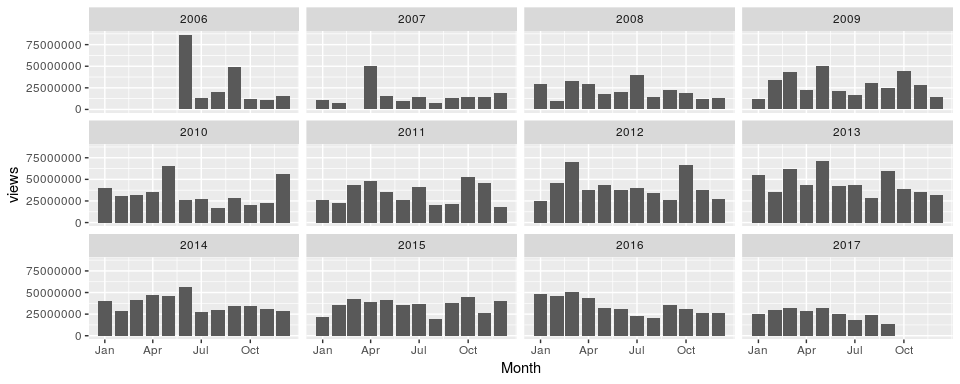
#Month Analysis:  
#Y is Numeric  
ggplot(TED.df,aes(x=as.Date(paste0("2015-",TED.df$month,"-01"),"%Y-%m-%d"),y=views)) + stat\_summary(fun.y = sum, geom = "bar")+labs(x="Month") + labs(title = "" ,subtitle ="By Month" ) + scale\_x\_date(date\_labels = "%b")



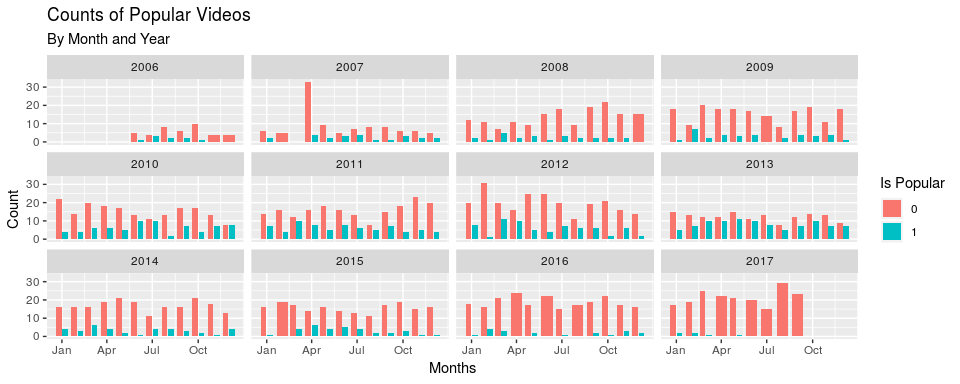
#Y is Categorical  
ggplot(TED.df, aes(x = as.Date(paste0("2015-",TED.df$month,"-01"),"%Y-%m-%d"), fill = as.factor(isPop))) + geom\_bar(stat='count',size=3,position = "dodge") + labs(title = "Counts of Popular Videos",subtitle = "By Month",x="Months",y="Count",fill="Is Popular") + scale\_x\_date(date\_labels = "%b")



#Month & Year Analysis  
ggplot(TED.df,aes(x=as.Date(paste0("2015-",TED.df$month,"-01"),"%Y-%m-%d"),y=views)) + stat\_summary(fun.y = sum, geom = "bar")+labs(x="Month") + scale\_x\_date(date\_labels = "%b") + facet\_wrap(~year)



ggplot(TED.df, aes(x = as.Date(paste0("1999-",TED.df$month,"-01"),"%Y-%m-%d"), fill = as.factor(isPop))) + geom\_bar(stat='count',position = position\_dodge2(width = NULL, preserve = c("total", "single"),  
 padding = 0.1, reverse = FALSE),size=2) + labs(title = "Counts of Popular Videos",subtitle = "By Month and Year",x="Months",y="Count",fill="Is Popular") + scale\_x\_date(date\_labels = "%b") + facet\_wrap(~year)



We now turn our attention to assessing characteristics of topic types for different videos. Using the data term matrix that was previously created, we evaluated the performance of grouping videos into different k number of topics. To briefly summarize the LDA; it is a generative probabilistic model in which objects are broken into smaller pieces that comprise them. A singular piece for a specific object may be identical or unique amongst other objects in the model. The model’s probabilistic setup is to group the pieces amongst all objects into k number of topic groups. With this, the probability of sampling a specific piece from a topic comprises of one distribution. Additionally, based on the occurrences of pieces in an object, the probability of sampling a topic type from an object can be derived.

In our situation, the videos represent the objects and the words contained in their title, description, and tags are the pieces they embody. The model aims to estimates two variables: the probability of a word belonging to a topic denoted beta and the probability of a group belonging to a topic denoted gamma. Provided a training set of data, it learns these parameters by using the method of Gibbs sampling. The algorithm goes through each video’s text data and randomly assigns each word within the text data to one of the k topic groups. This forms initial probability distributions for words in topics and topics in videos. It then goes through each word in each document and evaluates two conditional probabilities. The proportion of the words in the document have been assigned to an individual topic (P(t|v)) and the proportion of that topic to overall videos that come from that specific word (P(w|t). It then reassigns that word to one of the topic groups with probability P(t|v) x P(w|t) (the probability that topic t generate that word). After repeating this process a large number of times, it eventually reaches a steady state for both probability distributions, after which the process ends and the resulting estimates are used.

We can then classify each video by comparing the probability of the topic given the video for each topic (gamma) and assigning the topic to the highest probability.

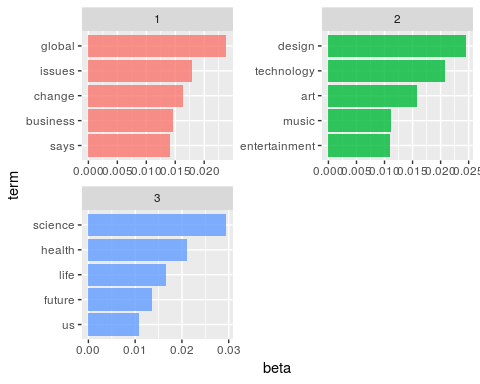
Before the model can be learned, the hyperparameter of k number of groups must be decided. In making this decsion evaluated below model results for differnt levels. Models were compared based on their individual loglikelihoods given the training data and the perplexity of each model. Perplexity, like likelihood, is a measure of how well the model predicts the data. It is, however, a measure of uncertainty in a model. A random variable with perplexity k has the same uncertianty as a k-sided dice.

n\_topics <- c(2, 4, 10, 20, 24, 26,30,32,34,36,38,40,50)  
lda\_compare <- n\_topics %>%  
 map(LDA, x = par\_DTM, control = list(seed = 747))  
data\_frame(k = n\_topics,  
 perplex = map\_dbl(lda\_compare, perplexity)) %>%  
 ggplot(aes(k, perplex)) +  
 geom\_point() +  
 geom\_line() +  
 labs(title = "Evaluating LDA topic models",  
 subtitle = "Optimal number of topics (smaller is better)",  
 x = "Number of topics",  
 y = "Perplexity")  
data\_frame(k = n\_topics,  
 perplex = map\_dbl(lda\_compare, logLik)) %>%  
 ggplot(aes(k, perplex)) +  
 geom\_point() +  
 geom\_line() +  
 labs(title = "Evaluating LDA topic models",  
 subtitle = "Optimal number of topics (larger is better)",  
 x = "Number of topics",  
 y = "score")

It is important to highlight that, with this type of unsupervised learning, it recommends not solely basing the number of topics off perplexity and log-likelihood but also to asses where the topic types have an intuitive sense to them. For example, whether words are being segmented into topics with an interpretable meaning. With this in mind, we decided on 3 groups. Seen below are the words with the highest probability of belonging to the respective topic group. The model classifies the video into one of these topic groups based on which topic corresponds to the highest probability associated with its respective words and their frequencies. In certain cases, videos have equal probabilities amongst each topic group. In this scenario, we decided to classify it as an arbitrary general topic group which does not have a focused theme.

Here, we see the following topics and their themes: \* Topic 1 has a theme of culture and art \* Topic 2 has a theme of public and international affairs \* Topic 3 has a theme of modernism and technology An observation that scored equal probabilities for each topic were assigned a topic value of 0, representing the general topic group.

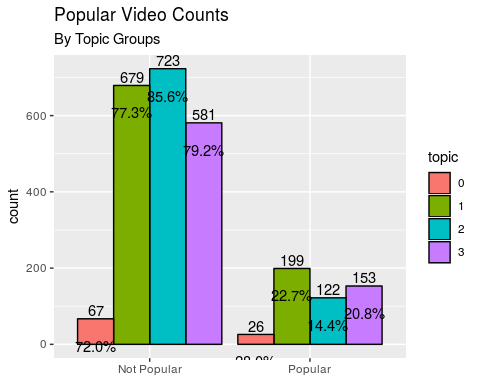
#Select model with best k  
  
lda\_model <-LDA(par\_DTM, method = "Gibbs",k = 3,control =list(seed=747))  
  
terms <- tidy(lda\_model)  
topTerms <- terms %>% group\_by(topic) %>% top\_n(5,beta) %>% ungroup() %>% arrange(topic,-beta)  
  
topTerms %>%  
 mutate(term = reorder(term, beta)) %>%  
 ggplot(aes(term, beta, fill = factor(topic))) +  
 geom\_bar(alpha = 0.8, stat = "identity", show.legend = FALSE) +  
 facet\_wrap(~ topic, scales = "free", ncol = 2) +  
 coord\_flip()



#terms(lda\_model,5)  
obs\_gamma <- tidy(lda\_model, matrix="gamma")  
obs\_gamma$document <- rep(TED.df$title,1)  
obs\_gamma$id <-seq(1, 2550, 1)  
topic <- obs\_gamma %>% group\_by(document) %>% top\_n(1,gamma)  
topic$dup <- duplicated(topic[,4])  
topic[duplicated(topic$id)|duplicated(topic$id, fromLast=TRUE),2] = "0"  
topic <- topic %>% group\_by(document) %>% sample\_n(1) %>% ungroup()  
topic$topic <- as.character(topic$topic)  
topic <-arrange(topic,document)  
#topic <- dummy\_cols(topic,select\_columns = "topic")  
topicNames <- colnames(topic)  
TED.df <- arrange(TED.df,title)  
TED.df$topic <- topic$topic

We then evaluated how popularity differs amongst the respective groups. We noticed that, among the popular videos, Topic 2 (public and international affaris) has the most view count. This makes sense since art/culture and modernism/technology are focused more on groups of people who either work in these sectors or have a personal preference towards them, while public and global issues are shared by a majority of people no matter their personal preferences or employment.

TED.df$isPop <- as.numeric(TED.df$isPop)  
basic <- c("isPop","AvgRelViews","TotalRelViews","RelatedCount","duration","languages","num\_speaker","DaysAvailable","titleLen","descriptionLen","topic")  
Pop.basic <- select(TED.df,basic)  
sentiment <- c("isPop","Persuasive","Unconvincing","Longwinded","Informative","Beautiful","Confusing","Obnoxious","jawdropping","Ingenious" ,"OK","Courageous","Inspiring","Funny","Fascinating","topic")  
Pop.sentiment <-select(TED.df,sentiment)  
percentData <- TED.df %>% group\_by(topic) %>% count(isPop) %>% mutate(ratio=percent(n/sum(n))) %>% ungroup()  
  
ggplot(TED.df,aes(x = as.factor(isPop),fill = topic)) + geom\_bar( position = "dodge",stat = "count",color="black") + labs(x=NULL,title ="Popular Video Counts",subtitle = "By Topic Groups") + geom\_text(stat = 'count',aes(label=..count..),position=position\_dodge(width=0.9), vjust=-0.25) + scale\_x\_discrete(breaks = c("0","1"),labels = c("Not Popular","Popular")) + geom\_text(data=percentData, aes(y=n,label=ratio),  
 position=position\_dodge(width=0.9), vjust=3)

 In attempting MANOVA, we see that an assumption of equal covariance was rejected by Wilks’ lambda. This may be due to the fact that the nature of data is not normal.

data <- select(TED.df,views,CR,sentSUM,isPop,topic)  
data$isPop <- as.numeric(data$isPop)  
  
res.man <- manova(as.matrix(data[,1:4]) ~ topic,data = data)  
summary(res.man,test="Wilks")

## Df Wilks approx F num Df den Df Pr(>F)   
## topic 3 0.96676 7.2111 12 6728.4 0.0000000000002864 \*\*\*  
## Residuals 2546   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

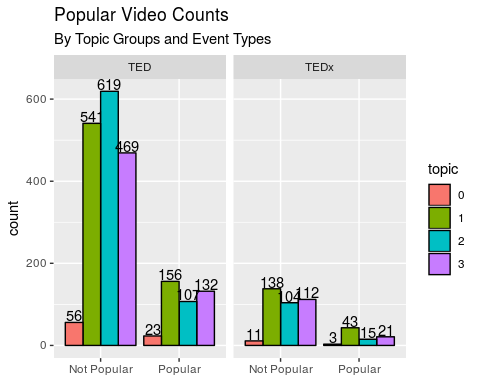
Furthermore, when comparing just the topic groups 1 to 3, the test for equal covariance was still rejected by Wilks’ Lambda. Again, since the nature of data is not normal, it makes sense that the test by Wilks’ Lambda fails.

data <- filter(data,topic != 0)  
res.man <- manova(as.matrix(data[,1:4]) ~ topic,data = data)  
summary(res.man,test="Wilks")

## Df Wilks approx F num Df den Df Pr(>F)   
## topic 2 0.96972 9.4945 8 4902 0.0000000000004064 \*\*\*  
## Residuals 2454   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

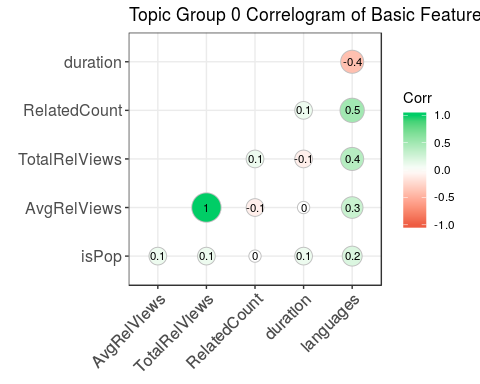
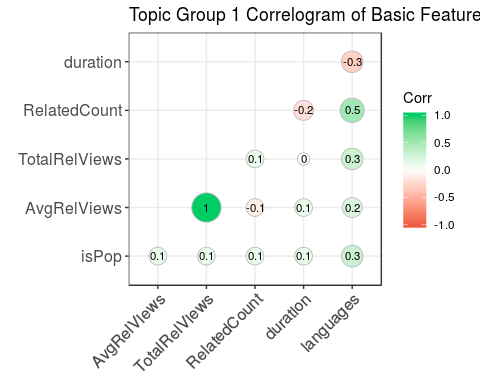
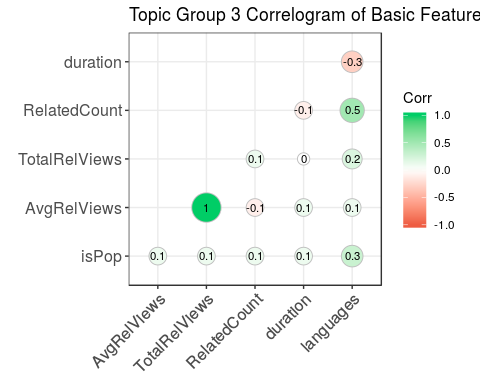
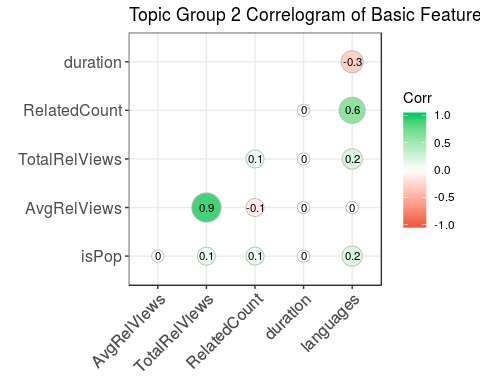
In assessing the popularity of event types amongst the topic groups, we see that Topic 2 has both the most popular TED and TEDx videos. We thought, since the TED and TEDx are intended for different kind of public (TED covers more global issues and focuses on the global community, while TEDx is typically intended for a local community and voices local issues), the distribution of popular videos among these two groups might differ as well. However, there does not appear to be any trend reversals when segmenting by event type. Again, it is possible that this is due to the fact that global/public issues are intended to be of interest to a broader audience.

ggplot(TED.df,aes(x = as.factor(isPop),fill = topic)) + geom\_bar( position = "dodge",stat = "count",color="black") + labs(x=NULL,title ="Popular Video Counts",subtitle = "By Topic Groups and Event Types" ) + geom\_text(stat = 'count',aes(label=..count..),position=position\_dodge(width=0.9),vjust=-.1 ) + scale\_x\_discrete(breaks = c("0","1"),labels = c("Not Popular","Popular")) + facet\_wrap(~eventType)

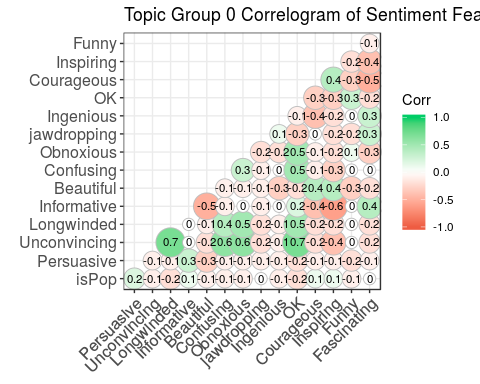
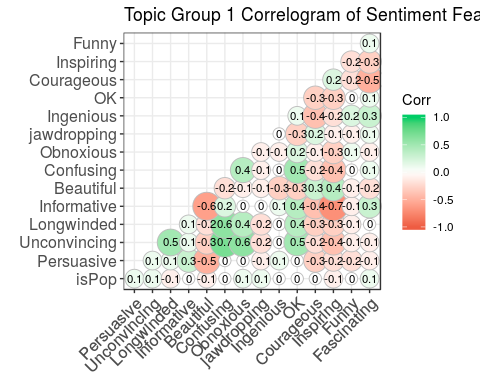
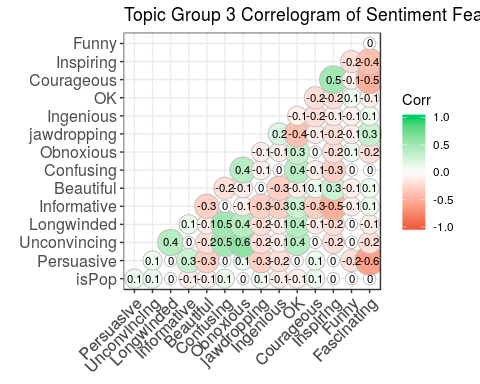
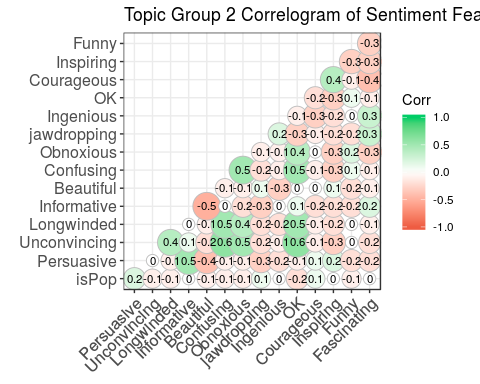


We further evaluated the relationship amongst popularity with feature variables by segments of topic groups. There are any clear major changes in correlations by topic groups. The number of languages remains the strongest correlation across topics and there are minor differences in correlations amongst the rating ratios. This makes sense since if the video is accessible for a bigger population, it will have a higher view count. We observed that, while analyzing the sentiment feature variables, the correlation between comments that express the same degree of evaluation (i.e “unconvincing” and “confusing” or “inspiring” and “courageous”) is also quite high. Again, we can theorize that the crowd effect is prevalent, since people are more likely to have the same opinion as the majority of the public.

Pop.basic$isPop <- as.numeric(Pop.basic$isPop)  
Pop.sentiment$isPop <- as.numeric(Pop.sentiment$isPop)  
for(i in unique(Pop.basic$topic)) {  
 topic.df <- subset(Pop.basic, topic==i)  
 corr<- round(cor(topic.df[,1:6]),1)  
 gg <- ggcorrplot(corr, hc.order = FALSE,   
 type = "lower",   
 lab = TRUE,   
 lab\_size = 3,   
 method="circle",   
 colors = c("tomato2", "white", "springgreen3"),   
 title= paste("Topic Group",i,"Correlogram of Basic Feature Variables"),  
 ggtheme=theme\_bw)  
 plot(gg)  
}



for(i in unique(Pop.basic$topic)) {  
 topic.df <- subset(Pop.sentiment, topic==i)  
 corr<- round(cor(topic.df[,1:15]),1)  
 gg <- ggcorrplot(corr, hc.order = FALSE,   
 type = "lower",   
 lab = TRUE,   
 lab\_size = 3,   
 method="circle",   
 colors = c("tomato2", "white", "springgreen3"),   
 title= paste("Topic Group",i,"Correlogram of Sentiment Feature Variables"),  
 ggtheme=theme\_bw)  
 plot(gg)  
 }

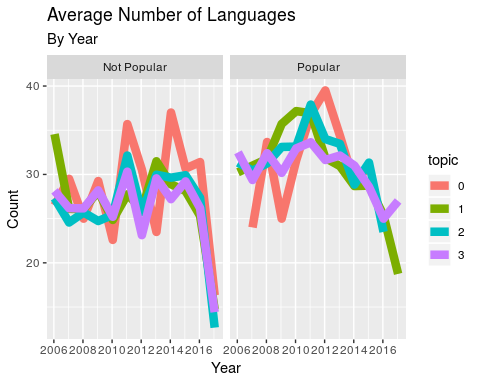


We decided to analyze how the number of views of different topic groups changes over time. As we can see, views for all topic groups are following roughly the same pattern as the popular videos in general. However, it is important to note, that for the general topic group 0 and topic group 3 (art/culture) there were two significant spikes in 2013 and 2014, respectively. Again, we are uncertain of the nature of such behavior, but we can theorize that during these years there were events that spiked peoples’ interests in these topics or there were certain videos in these categories that attracted a lot of attention. The latter theory could be backed up by the graph counts of popular videos by year and topic group, since there is a spike in popular videos in 2014 for group 3.

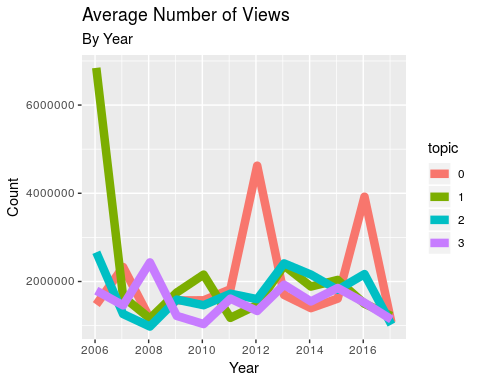
If we look at the graph that shows us the average number of comments per group topic we can see, that group 2 is generally the most discussed and has spikes in 2008, 2011 and 2014. Group 0, 1, and 3 roughly follow the same pattern, but it is important to note that the largest amount of comments belongs to group 0 in 2014.

We observed that the graphs that portray the counts of videos by topic by month show that group 2 tends to publish more videos from April to June, while group 1 has a spike in videos in September. On the graph showing counts of videos by month and year, we can see that the biggest spike for group 2 was in 2007, group 1 had the highest counts in 2011 and 2012, and starting 2014 group 2 took the lead. This may be indicative that, while TED Talks were originally intended for professional masses, later on, they expanded to cover topics of interest for the majority of the public.

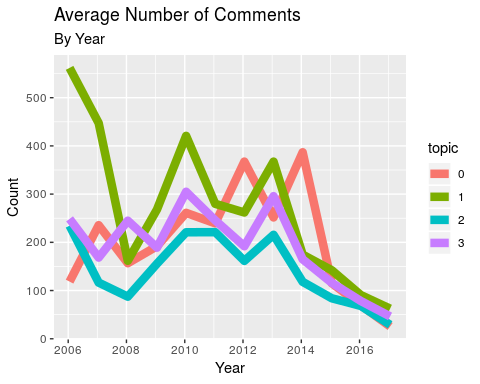
# Y is numeric  
ggplot(TED.df, aes(x = as.Date(as.character(year), format = "%Y"), y = languages, color=topic)) + stat\_summary(fun.y = mean, geom = "line",size=3) + labs(title = "Average Number of Languages",subtitle = "By Year",x="Year",y="Count") + scale\_x\_date(labels = date\_format("%Y")) + facet\_wrap(~mutate(TED.df,isPop=if\_else(isPop == "0","Not Popular","Popular"))$isPop)



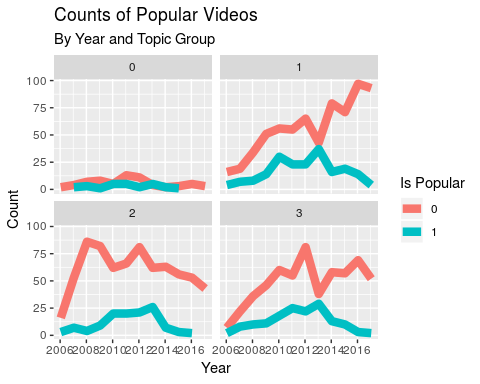
ggplot(TED.df, aes(x = as.Date(as.character(year), format = "%Y"), y = views,color=topic)) + stat\_summary(fun.y = mean, geom = "line",size=3) + labs(title = "Average Number of Views",subtitle = "By Year",x="Year",y="Count") + scale\_x\_date(labels = date\_format("%Y"))



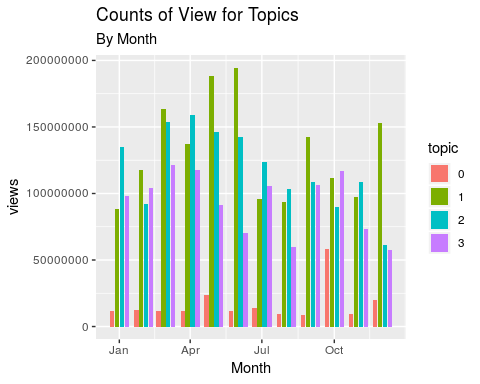
ggplot(TED.df, aes(x = as.Date(as.character(year), format = "%Y"), y = comments,color=topic)) + stat\_summary(fun.y = mean, geom = "line",size=3) + labs(title = "Average Number of Comments",subtitle = "By Year",x="Year",y="Count") + scale\_x\_date(labels = date\_format("%Y"))



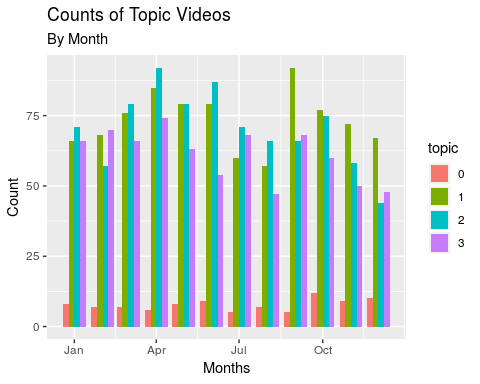
# Y is categorical  
ggplot(TED.df, aes(x = as.Date(as.character(year), format = "%Y"), color = as.factor(isPop))) + geom\_line(stat='count',size=3) + labs(title = "Counts of Popular Videos",subtitle = "By Year and Topic Group",x="Year",y="Count",color="Is Popular") + scale\_x\_date(labels = date\_format("%Y")) + facet\_wrap(~topic)



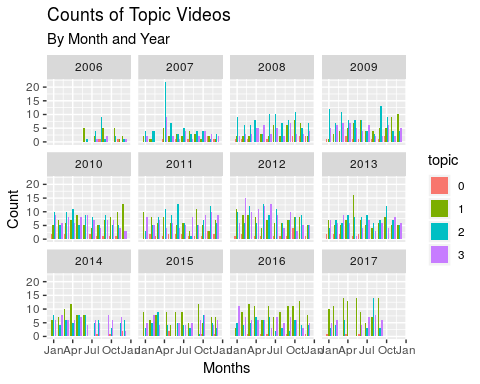
#Month Analysis:  
#Y is Numeric  
ggplot(TED.df,aes(x=as.Date(paste0("2015-",TED.df$month,"-01"),"%Y-%m-%d"),y=views,fill=topic)) + stat\_summary(fun.y = sum, geom = "bar",position = position\_dodge2(width = NULL, preserve = c("total", "single"),padding = 0.1, reverse = FALSE),size=2) +labs(x="Month") + labs(title = "Counts of View for Topics" ,subtitle ="By Month" ) + scale\_x\_date(date\_labels = "%b")



#Y is Categorical  
ggplot(TED.df, aes(x = as.Date(paste0("2015-",TED.df$month,"-01"),"%Y-%m-%d"), fill = topic)) + geom\_bar(stat='count',size=3,position = "dodge") + labs(title = "Counts of Topic Videos",subtitle = "By Month",x="Months",y="Count") + scale\_x\_date(date\_labels = "%b")



#Month & Year Analysis  
  
ggplot(TED.df, aes(x = as.Date(paste0("1999-",TED.df$month,"-01"),"%Y-%m-%d"), ,fill =topic)) + geom\_bar(stat='count',position = position\_dodge2(width = NULL, preserve = c("total", "single"),  
 padding = 0.1, reverse = FALSE),size=2) + labs(title = "Counts of Topic Videos",subtitle = "By Month and Year",x="Months",y="Count") + scale\_x\_date(date\_labels = "%b") + facet\_wrap(~year)



To help structure our consideration of predictors of popularity we implemented a forward elimination process with logistic regression. We began by splitting our data in 80% training and 20% testing using stratified random sampling. Our strata sizes were selected proportionally based on the number of popular and unpopular talks available amongst the entire set. We sampled 400 Popular classified videos and 1640 Unpopular classified videos to be used in the training set. The remainder would then be used as a test set to evaluate the trained models.

Logsitic Regression Analysis

set.seed(123)  
TED.df$index <- seq(1,2550,1)  
notPopular <- filter(TED.df,isPop == 0)  
notPopularSample <- notPopular %>% sample\_n(1640)  
Popular <- filter(TED.df, isPop== 1)  
PopularSample <- Popular %>% sample\_n(400)  
train <- rbind(notPopularSample,PopularSample)  
test <- TED.df[-train$index,]  
  
accuracy <- function(model,train,test){  
 predictions <- predict(model,type = "response")  
 predTrain.df <- data.frame(p=predictions)  
 predTrain.df <- mutate(predTrain.df,isPop = if\_else(p >= .5,1,0))  
 TrainAcc <- 1 - (sum(abs(train$isPop -predTrain.df$isPop)) / nrow(train))  
   
 predictions <- predict(model,newdata = test,type = "response")  
 predTest.df <- data.frame(p=predictions)  
 predTest.df <- mutate(predTest.df,isPop = if\_else(p >= .5,1,0))  
  
 TestAcc <- 1 - (sum(abs(test$isPop - predTest.df$isPop)) / nrow(test))  
 result <- c(TrainAcc,TestAcc)  
 return(result)  
}  
  
modelResults <- data.frame(Variables=rep("a",8),AIC=rep(0,8),TrainACC = rep(0,8),TestACC=rep(0,8))  
modelResults$Variables <- as.character(modelResults$Variables)  
colnames(modelResults) <- c("Variables","AIC","TrainACC", "TestACC")  
  
  
z <- 1

We began by training a binomial logistic regression on each individual quantitative and categorical variable. For each model, we checked whether the p-values for all coefficients were less than 0.1. Then the AIK score was reported along with its training and test accuracy. After the first round of variable selection was completed the variable which h minimized the AIK was selected to be used in the two variable sequences. We continued this process iteratively. At our 3rd variable selection, p-value significance was then lowered to .05. We continued iteratively selecting the variable that met this criterion and minimized AIK up until a nine variable model. This is when no additional variable selections had p-values that were significant.

#8 Variable Models  
varList <- c('num\_speaker','year','month','day','AvgRelViews','RelatedCount','Longwinded','Informative','Beautiful','Confusing','Obnoxious','Ingenious', 'Inspiring','Funny','eventType','topic','titleLen','descriptionLen')  
  
results <- matrix(,30,2)  
index <- 1  
n <- length(varList)  
for(i in 1:n){  
 fmla <- as.formula(paste0("isPop ~ languages + Persuasive + OK + Unconvincing + Fascinating + Courageous + jawdropping + ",varList[i]))  
 model <- glm(fmla,data = train,family = binomial)  
 sum <-summary(model)  
 if (all(sum$coefficients[,4] < .075)){  
 results[index,1] <- paste0("languages + Persuasive + OK + Unconvincing + Fascinating + Courageous + jawdropping + ", varList[i])  
 results[index,2] <- as.numeric(sum$aic)  
 index <- index + 1  
 }  
}  
colnames(results)<- c("Variables","AIC")  
results <- as.data.frame(results)  
(results <-arrange(results,AIC))

## Variables  
## 1 languages + Persuasive + OK + Unconvincing + Fascinating + Courageous + jawdropping + AvgRelViews  
## 2 languages + Persuasive + OK + Unconvincing + Fascinating + Courageous + jawdropping + RelatedCount  
## 3 languages + Persuasive + OK + Unconvincing + Fascinating + Courageous + jawdropping + Obnoxious  
## 4 <NA>  
## 5 <NA>  
## 6 <NA>  
## 7 <NA>  
## 8 <NA>  
## 9 <NA>  
## 10 <NA>  
## 11 <NA>  
## 12 <NA>  
## 13 <NA>  
## 14 <NA>  
## 15 <NA>  
## 16 <NA>  
## 17 <NA>  
## 18 <NA>  
## 19 <NA>  
## 20 <NA>  
## 21 <NA>  
## 22 <NA>  
## 23 <NA>  
## 24 <NA>  
## 25 <NA>  
## 26 <NA>  
## 27 <NA>  
## 28 <NA>  
## 29 <NA>  
## 30 <NA>  
## AIC  
## 1 1700.58422943165  
## 2 1701.65562700101  
## 3 1703.19641752564  
## 4 <NA>  
## 5 <NA>  
## 6 <NA>  
## 7 <NA>  
## 8 <NA>  
## 9 <NA>  
## 10 <NA>  
## 11 <NA>  
## 12 <NA>  
## 13 <NA>  
## 14 <NA>  
## 15 <NA>  
## 16 <NA>  
## 17 <NA>  
## 18 <NA>  
## 19 <NA>  
## 20 <NA>  
## 21 <NA>  
## 22 <NA>  
## 23 <NA>  
## 24 <NA>  
## 25 <NA>  
## 26 <NA>  
## 27 <NA>  
## 28 <NA>  
## 29 <NA>  
## 30 <NA>

results$Variables <- as.character(results$Variables)  
r <- results[1,]  
modelResults[z,1] <- r[1]  
fmla <- fmla <- as.formula(paste0("isPop ~ ",results[1,1]))  
m <- glm(fmla,train,family = binomial)  
modelResults[z,2] <- m$aic  
acc <- accuracy(m,train,test)  
modelResults[z,3] <- acc[1]  
modelResults[z,4] <- acc[2]  
z <- z + 1  
modelResults

## Variables  
## 1 languages  
## 2 languages + Persuasive  
## 3 languages + Persuasive + OK  
## 4 languages + Persuasive + OK + Unconvincing  
## 5 languages + Persuasive + OK + Unconvincing + Fascinating  
## 6 languages + Persuasive + OK + Unconvincing + Fascinating + Courageous  
## 7 languages + Persuasive + OK + Unconvincing + Fascinating + Courageous + jawdropping  
## 8 languages + Persuasive + OK + Unconvincing + Fascinating + Courageous + jawdropping + AvgRelViews  
## AIC TrainACC TestACC  
## 1 1867.682 0.7990196 0.7941176  
## 2 1817.471 0.7985294 0.7901961  
## 3 1776.918 0.7985294 0.7901961  
## 4 1751.485 0.8004902 0.7882353  
## 5 1733.698 0.8039216 0.7921569  
## 6 1709.862 0.8122549 0.7941176  
## 7 1704.291 0.8107843 0.7941176  
## 8 1700.584 0.8098039 0.7941176

#9 Variable Models  
varList <- c('num\_speaker','year','month','day','RelatedCount','Longwinded','Informative','Beautiful','Confusing','Obnoxious','Ingenious', 'Inspiring','Funny','eventType','topic','titleLen','descriptionLen')  
  
results <- matrix(,30,2)  
index <- 1  
n <- length(varList)  
for(i in 1:n){  
 fmla <- as.formula(paste0("isPop ~ languages + Persuasive + OK + Unconvincing + Fascinating + Courageous + jawdropping + AvgRelViews + ",varList[i]))  
 model <- glm(fmla,data = train,family = binomial)  
 sum <-summary(model)  
 if (all(sum$coefficients[,4] < .075)){  
 results[index,1] <- paste0("languages + Persuasive + OK + Unconvincing + Fascinating + Courageous + jawdropping + AvgRelViews + ", varList[i])  
 results[index,2] <- as.numeric(sum$aic)  
 index <- index + 1  
 }  
}  
colnames(results)<- c("Variables","AIC")  
results <- as.data.frame(results)  
(results <-arrange(results,AIC))

## Variables  
## 1 languages + Persuasive + OK + Unconvincing + Fascinating + Courageous + jawdropping + AvgRelViews + RelatedCount  
## 2 languages + Persuasive + OK + Unconvincing + Fascinating + Courageous + jawdropping + AvgRelViews + Obnoxious  
## 3 <NA>  
## 4 <NA>  
## 5 <NA>  
## 6 <NA>  
## 7 <NA>  
## 8 <NA>  
## 9 <NA>  
## 10 <NA>  
## 11 <NA>  
## 12 <NA>  
## 13 <NA>  
## 14 <NA>  
## 15 <NA>  
## 16 <NA>  
## 17 <NA>  
## 18 <NA>  
## 19 <NA>  
## 20 <NA>  
## 21 <NA>  
## 22 <NA>  
## 23 <NA>  
## 24 <NA>  
## 25 <NA>  
## 26 <NA>  
## 27 <NA>  
## 28 <NA>  
## 29 <NA>  
## 30 <NA>  
## AIC  
## 1 1699.00634330707  
## 2 1699.54639458958  
## 3 <NA>  
## 4 <NA>  
## 5 <NA>  
## 6 <NA>  
## 7 <NA>  
## 8 <NA>  
## 9 <NA>  
## 10 <NA>  
## 11 <NA>  
## 12 <NA>  
## 13 <NA>  
## 14 <NA>  
## 15 <NA>  
## 16 <NA>  
## 17 <NA>  
## 18 <NA>  
## 19 <NA>  
## 20 <NA>  
## 21 <NA>  
## 22 <NA>  
## 23 <NA>  
## 24 <NA>  
## 25 <NA>  
## 26 <NA>  
## 27 <NA>  
## 28 <NA>  
## 29 <NA>  
## 30 <NA>

In comparing the top models containing 1 to 8 variables, we noticed that AIK continually decreases, training accuracy continually increases but at a diminishing rate, and training accuracy peaks at the 6 variable model but then begins to decrease with further additional variables.

Since the training accuracy peaks at the model with the 6 variables, in order to avoid overfitting and make the model as simple as possible, we decided to choose the model with 6 variables.

# Best Model  
modelResults

## Variables  
## 1 languages  
## 2 languages + Persuasive  
## 3 languages + Persuasive + OK  
## 4 languages + Persuasive + OK + Unconvincing  
## 5 languages + Persuasive + OK + Unconvincing + Fascinating  
## 6 languages + Persuasive + OK + Unconvincing + Fascinating + Courageous  
## 7 languages + Persuasive + OK + Unconvincing + Fascinating + Courageous + jawdropping  
## 8 languages + Persuasive + OK + Unconvincing + Fascinating + Courageous + jawdropping + AvgRelViews  
## AIC TrainACC TestACC  
## 1 1867.682 0.7990196 0.7941176  
## 2 1817.471 0.7985294 0.7901961  
## 3 1776.918 0.7985294 0.7901961  
## 4 1751.485 0.8004902 0.7882353  
## 5 1733.698 0.8039216 0.7921569  
## 6 1709.862 0.8122549 0.7941176  
## 7 1704.291 0.8107843 0.7941176  
## 8 1700.584 0.8098039 0.7941176

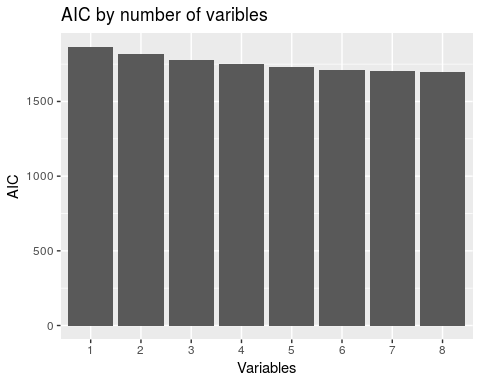
modelResults$numVar <- c(1,2,3,4,5,6,7,8)  
modelResults$TestACC <- as.numeric(modelResults$TestACC)  
modelResults$TrainACC <- as.numeric(modelResults$TrainACC)  
fmla <- as.formula(paste0("isPop ~ languages + Persuasive + OK + Unconvincing + Fascinating + Courageous"))  
model <- glm(fmla,data = train,family = binomial)  
summary(model)

##   
## Call:  
## glm(formula = fmla, family = binomial, data = train)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -2.2240 -0.6573 -0.4586 -0.1970 3.1810   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -6.177330 0.420110 -14.704 < 0.0000000000000002 \*\*\*  
## languages 0.101236 0.008172 12.388 < 0.0000000000000002 \*\*\*  
## Persuasive 7.757706 0.947174 8.190 0.00000000000000026 \*\*\*  
## OK -14.541009 2.230128 -6.520 0.00000000007018643 \*\*\*  
## Unconvincing 10.206662 1.616551 6.314 0.00000000027217290 \*\*\*  
## Fascinating 7.197613 1.102445 6.529 0.00000000006631151 \*\*\*  
## Courageous 5.060338 0.985048 5.137 0.00000027894018045 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 2019.3 on 2039 degrees of freedom  
## Residual deviance: 1695.9 on 2033 degrees of freedom  
## AIC: 1709.9  
##   
## Number of Fisher Scoring iterations: 5

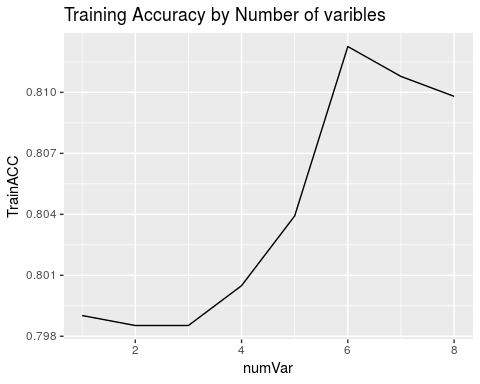
# Evaluate Accuracy on the full data set  
(accuracy(model,train,TED.df))

## [1] 0.8122549 0.8086275

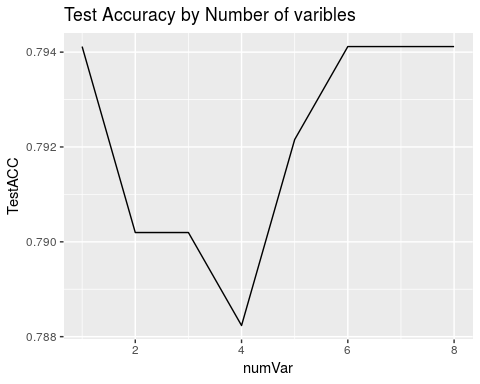
ggplot(modelResults,aes(x=Variables,y=AIC)) + geom\_bar(stat='identity') + labs(title="AIC by number of varibles") + scale\_x\_discrete(labels = c('1','2','3','4','5','6','7','8'))



ggplot(modelResults,aes(x=numVar,y=TrainACC)) + geom\_line() + labs(title="Training Accuracy by Number of varibles")



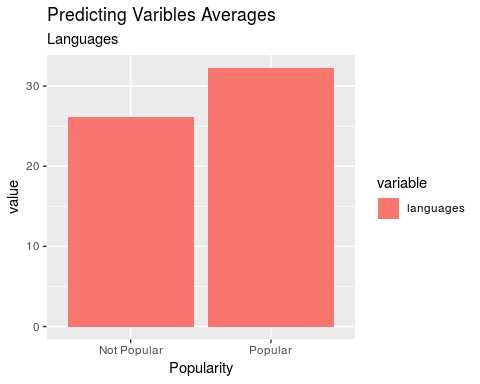
ggplot(modelResults,aes(x=numVar,y=TestACC)) + geom\_line() + labs(title="Test Accuracy by Number of varibles")



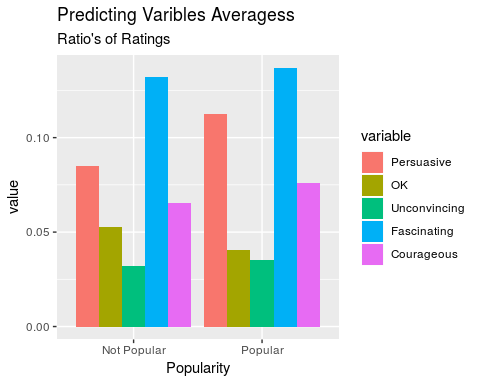
The average count of the number of languages is slightly higher amongst popular videos. This agrees with our model’s predictors as it assigns positive but small coefficient of 0.096298 to rating count of number languages. This also supports the notion that expanding the accessibility of potential viewers probably could increase the popularity of a TED Talk.

In the averages of rating ratios, we see that “Persuasive”, “Courageous”, and “Fascinating” stand out in being larger amongst popular videos. This is in line with our model’s predictors as it assigns positive coefficients to each of these variables. This gives support to the idea that new creators of content will have a greater possibility of having a popular TED Talk if their video can deliver a message containing these themes.

TED.df$isPop <- as.character(TED.df$isPop)  
m <- melt(TED.df,id.vars='isPop',measure.vars=c('languages'))  
g <- ggplot(m,aes(x=isPop, y=value, fill=variable))  
g + geom\_bar(stat = "summary", fun.y = "mean",position = position\_dodge(width = 0.9)) + labs(title ="Predicting Varibles Averages" ,subtitle = "Languages" , x="Popularity") + scale\_x\_discrete(labels = c("Not Popular","Popular"))



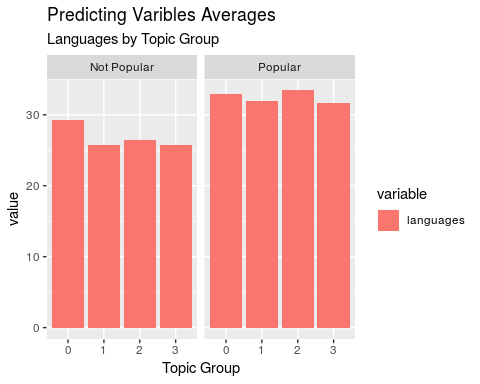
m <- melt(TED.df,id.vars='isPop',measure.vars=c('Persuasive', 'OK','Unconvincing','Fascinating','Courageous'))   
  
g <- ggplot(m,aes(x=isPop, y=value, fill=variable))  
g + geom\_bar(stat = "summary", fun.y = "mean",position = position\_dodge(width = 0.9)) + labs(title = "Predicting Varibles Averagess",subtitle = "Ratio's of Ratings", x="Popularity") + scale\_x\_discrete(breaks = c("0","1"),labels = c("Not Popular","Popular"))



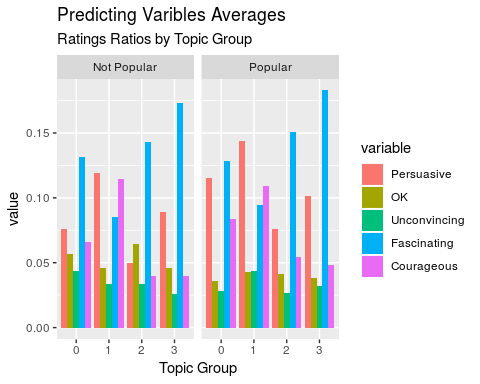
TED.df <- mutate(TED.df,isPop= ifelse(isPop =="1","Popular","Not Popular"))

We see that the average ratios for the rating “Fascinating” are highest in topic 3, associated with science and technology, for both popularity categories. This offers some support that viewers of both popular and not popular videos will be more fascinated by this subject matter. Additionally, topic 1, associated with culture and art, has the largest average ratio for “Persuasive”. Overall the averages of ratings do not change drastically between popularity groups when segmented by topic types. This was in line with our predictive model as it did not find topic type have stronger predictive power.

m <- melt(TED.df,id.vars=c('topic','isPop'),measure.vars=c('languages'))  
g <- ggplot(m,aes(x=topic, y=value, fill=variable))  
g + geom\_bar(stat = "summary", fun.y = "mean",position = position\_dodge(width = 0.9)) + labs(title ="Predicting Varibles Averages" ,subtitle = "Languages by Topic Group" , x="Topic Group") + facet\_wrap(~isPop)



m <- melt(TED.df,id.vars=c('topic','isPop'),measure.vars=c('Persuasive', 'OK','Unconvincing','Fascinating','Courageous'))  
g <- ggplot(m,aes(x=topic, y=value, fill=variable))  
g + geom\_bar(stat = "summary", fun.y = "mean",position = position\_dodge(width = 0.9)) + labs(title ="Predicting Varibles Averages" ,subtitle = "Ratings Ratios by Topic Group" , x="Topic Group") + facet\_wrap(~isPop)

 We noticed that the number of languages offered over time for both popular and not popular groups decreases as we move closer to 2017. This makes sense as more recent videos will have had less time to been translated into many languages. However, it is notable that the popular groups have a greater number of languages available for each time period. This supports our model’s predictions as it assigns a positive coefficient to the number of available languages.

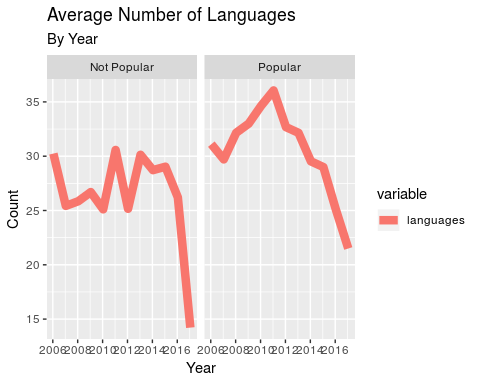
In analyzing the ratings ratios over time we see that the average ratio of “Persuasive” ratings remains consistently higher than that of non-Popular over all time periods. This provides support that, in order for a TED Talk to be popular, it needs to persuade the viewer of there message they are trying to deliver.

Additionally, the average ratio of “Fascinating” was in the top two average ratios proportions until 2015, whereafter it fell significantly. Our interpretation of this was that, with more and more quality content being created by different TED Talks over the years, the expectations of viewers have risen over time and viewers are not as easily persuaded as before.

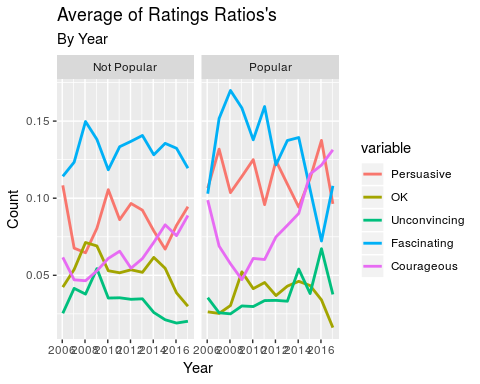
In contrast to this, the average ratio of the rating “Unconvincing” has been trending upwards in recent years. Our model assigns this ratio a positive coefficient. Our intuition behind this is if there is a divide in users opinions on whether a TED Talk is persuasive or unconvincing it will instigate debate amongst the comments and increase the user activity in the process.

We see that the average ratio of “Courageous” rating has continuously increased from 2010 onward which supports the idea that shift in theme preference towards speakers whose topic takes courage to address. We see that the “Okay” rating is consistently lower in the popular category over all years. This is in line with our prediction model which assigns a negative coefficient to ratios of okay. The intuition behind this is that videos that are not able to generate a more extreme opinion from viewers will fail to attract larger viewer activity.

m <- melt(TED.df,id.vars=c('topic','isPop','year'),measure.vars=c('languages'))  
  
ggplot(m, aes(x = as.Date(as.character(year), format = "%Y"), y = value, color=variable)) + stat\_summary(fun.y = mean, geom = "line",size=3) + labs(title = "Average Number of Languages",subtitle = "By Year",x="Year",y="Count") + scale\_x\_date(labels = date\_format("%Y")) + facet\_wrap(~isPop)



m <- melt(TED.df,id.vars=c('topic','isPop','year'),measure.vars=c('Persuasive', 'OK','Unconvincing','Fascinating','Courageous'))  
ggplot(m, aes(x = as.Date(as.character(year), format = "%Y"), y = value, color=variable)) + stat\_summary(fun.y = mean, geom = "line",size=1) + labs(title = "Average of Ratings Ratios's",subtitle = "By Year",x="Year",y="Count") + scale\_x\_date(labels = date\_format("%Y")) + facet\_wrap(~isPop)

 Our findings suggest that future TED Talks will have a greater probability of generating higher views and comments by offering the video in a greater number of languages. Furthermore, our findings suggest that the probability of popularity increases if the speaker is able to either fully persuade their audience or spark a debate amongst viewers who agree and disagree with the message. This will help generate a greater number of views and comments. In recent years, the “Courageous” rating ratio has been associated with popular videos. However, it is recommended that these trends are continued to be watched over the coming year; as with any preference they can change over time.