Online News Prediction

Jon Doretti, Samuel Prasad Chinta, Aswin Kumar Guvvala,

Erika Larsen, Anisha Patel, Arun Sivakumar

DSC 424: Advanced Data Analysis

Professor Stephanie Besser

August 2022

Executive Summary

Due to the rise of major network platforms and the expansion of the world wide web, considered a universe of information, are popular at home and offices. In this enormous search engine, there is much information reflected in the form of articles. Suppose an article is popular, which means that it was among the "most read" articles shown online. If it is possible to predict the popularity of an online article, it may help people to know the situation of the information propagation. With this prediction model, people can even use it to construct information that is easily spread. Therefore, it is essential to predict the popularity of online news.

The reasons, why an article is successful such as quantifiable measures like the number of images an article includes and less easily quantifiable measures such as the polarity and topic of the article. To quantify the less quantifiable ideas our dataset “Online News Popularity” uses a process called Natural Language Processing (NLP). Artificial intelligence can be used to perform NLP to do what would have previously taken humans hours to do. (Nadkarni et al., 2011) (Fernandes et al., 2015)In our case, NLP was used to read through each article and determine its polarity, topic, and word count. The variables NLP created can be seen in figure 3.3.

The methods we have used here are to predict the number of shares published by mashable using multiple linear regression specifically ridge regression, and validating the importance of variables predicting the popularity of the articles using lasso regression, and reduce the multicollinearity in the dataset concerning predict the possibility of the number of people sharing the articles we have used PCA and correspondence analysis is used to analyze the relationship between the weekday and type of news channels as how people use the news channels during the weekdays represented in figure 1.2.

The limitation of the report is as the dataset used in our analysis is composed of articles published by a single website, Mashable. Expanding our analysis to different news websites and determining if the same variables play a major role in determining the popularity of the news article is an important future direction. Our Dataset falls under the social sciences category, resulting in low correlation values between dependent and independent variables. Estimation of latent variables that affect the number of shares using factor analysis or discriminant analysis is another direction. Optimization of the features in an article that improves the number of shares could be a profitable direction.

Our conclusion says from the analysis we can predict that the business channel is more popular, from Monday - Thursday. On weekends lifestyle and world news channels were more popular. We ranked optimizing the Number of images and minimizing bad keywords in an article to be the most important changes to be made to improve the number of shares achieved by the article. Grouping all the variables into 8 components, captures the 67% variance of the variables. We were able to group variables into 3 factors, positivity, unique word count and referenced article count accounting for 68.1% of the cumulative variance and that shares were always extremely positively affected or extremely negatively affected; never supporting the null hypothesis.

Abstract

With the growth of online news media, the popularity of a news article is something that is not only useful to the publisher of the article but can be useful to see what topics are of concern or interest to the public at large. Using Online News Popularity Data Set, from the UCI Machine Learning Repository, we attempted to create a model to predict the number of shares a posted article will have. Several methods were implemented to create such a model including Correspondence Analysis, Lasso Regression, Principal Component Analysis (PCA), Common

Factor Analysis (CFA), Linear Regression, and Canonical Correlation Analysis. While not all modeling was found to be fruitful PCA and CFA successfully combined the 61 variables into a handful of factors. Correspondence analysis also successfully proved that not only the topic but the day of the week an article is posted play a vital role in the overall popularity of an article. Lasso Regression provided insight into how the number of images in an article influences the number of shares it receives. The results are the first step toward fully understanding how to predict the number of shares an article will have.

Introduction

In the digital world, online news is the primary source of information. News is shared in large numbers through online social networks which usually link to news websites. It has become easier to read news on social media and news websites. News popularity depends upon various factors such as its linguistic style, relevance to the current event, number of images and videos in the news, channel history of news, etc. After reading, people like the news article, share the news, and write comments on it. They share their opinion with the public. It helps in publicizing the news article. The click of a user for news articles is influenced by many factors such as articles' position on the web page, timing, topic, text, and additional media. These factors play an important role in measuring the news' popularity. If a news article gets the maximum number of shares then that news becomes the most popular news. The findings from this analysis can be used by stakeholders and decision makes in the publication of news to ensure that the news reaches more ears.

Rapid growth in online news services has triggered a direct need for analysis and optimization in the media industry. Analysis of news articles could provide us with insight that could lead to the application of optimization methods that take advantage of the findings of our analysis. Our dataset is composed of news articles collected from one of the most popular news

websites Mashable. Data collected from the articles provide us with 58 numerical features as predictors to predict the popularity of an article which is quantified by the number of shares achieved by the article.

Literature Review

Jowkar et al.(2009) evaluated Iranian newspapers' websites based on the criteria obtained from Alexa Search Engine using correspondence analysis. The objective of this research is the evaluation of Iranian newspapers' websites based on six indexes, including traffic rank, the average number of pages viewed by users, web page downloading speed, the number of links received from other websites, and the percentage of Iranian and foreign visitors. 24 newspapers were grouped into 3 groups of high links, low links, and foreign based on Correspondence analysis.

Recent advances in Intelligent Decision Support System support decisions by predicting an outcome and utilize several machine learning Optimization methods to optimize the predicted outcome. (Fernandes et al., 2015) Empirical evidence suggests that the popularity of a news article can be predicted using features in the article, we further analyze the importance of these features in predicting the popularity of the article using the lasso regression method. (Fernandes et al., 2015)

The idea of modeling the popularity of a post is not just applicable to spreading news but is something corporations try to capitalize on as well. Moro et al. (2016) researched top cosmetic brands and using support vector machines created a model to predict the success of a company's Facebook post. The variables chosen were a category of content (i.e. special offer or direct advertisement), the total number of likes a company has on its Facebook page, the month of the post, an hour of the post, post weekday, and whether or not the post was a paid

advertisement on Facebook. (Moro et al., 2016) While there are numerous studies looking at post retrospectively this study, in particular, is unique in the fact that it focuses on predictions much like our current study. The study focuses on 790 posts that were posted on Facebook in 2014. After data preprocessing was concluded 751 posts were used in the final model (Moro et al., 2016) Support vector machine was chosen due to its ability for high accuracy over a more user friendly model such as linear regression. (Moro et al., 2016) It was found that comments were the worst predictor due to the fact that some comments are negative and some positive and it is hard to quantify sentiment. While the final model only predicted with 27% accuracy it is argued that it still has applicable use in advertising. (Moro et al., 2016) We can also gleem knowledge from this study knowing which variables were the most important predictors and see their interactions in our model's performance.

Methods

The first step was checking for Null values and there were none of them. By looking at the descriptive statistics we can observe from the fig 1a that the ranges of the values vary a lot, so we standardized the values.

Reduced the dimensions of the data by backward elimination method by removing non-important variables. Exploratory analysis of the data by data visualizations using scatter plots to understand the variables better. Correlation analysis was done to understand if linear relations between any of the variables are significant. In order to remove the noise, exploring the data for any possible outliers in any of the variables.

We have used ‘VIF’ to find the multicollinearity of the whole dataset. In the fig 4a we can see that there is multicollinearity as most of the variables are above 0.7 which means that some variables are highly correlated to each other. After calculating correlation coefficients we find the variables that are highly correlated are n_unique_tokens, n_non_strop_words and

n_non_stop_unique_tokens. This means that there is a strong correlation between these variables.

We find out that multicollinearity exists in our independent variables as we try to check VIF values of our linear regression model. As displayed in Fig.5.1a in the appendix there is heavy multicollinearity in our dataset. We calculate the cook's d distance and remove samples that are higher than the default (4/sample size) length according to the cook's distance displayed in fig.5.2.a in appendix

Correspondence Analysis is useful to understand the relationship of our categorical data, and it could also be plotted like PCA for better visualization. As a result, under the assumption that there is a relationship among the categorical variables. Each categorical variable was paired with one of the parameters of interest, weekday, and Channel type. The Chi-squared test for independence was significant for the contingency table at a 5% significance level. The scree plot of eigenvalues shows that 2 dimensions explain about 90% of the variance. For creating the Correspondence model, library "ca" was used. The Libraries "factoExtra" and "FactoMineR" were used for visualizations. To further analyze the weekday and channel type, the weekdays were broken into two groups Monday through Thursday and Friday through Sunday.

A Mosaic plot(fig.10) was also created using the "vcd" library and used to better understand the frequency of channels for each weekday. The findings from the mosaic plot are very similar to correspondence analysis, the only difference is that the Pearson residuals in the mosaic plot show if the values are above or below expected values. On Saturday and Sunday, lifestyle news channels were watched more than expected and business channels were less than expected. On Monday, entertainment and business channels have more views than expected, and lifestyle has fewer views than expected.

We utilize Lasso regression to validate the importance of variables in predicting the popularity of an article and to predict the popularity of an article. We observe that there is multicollinearity in our model in the preprocessing stage(Refer to Fig.5.1a).We stress that the main reason to utilize Lasso regression is to deal with multicollinearity and numerous outliers present in our dataset (Refer to Fig.5.2.a).We Create a new model to predict the number of shares achieved by the article and experiment with various (λ) values which is a parameter that can be tuned by running several trials.We evaluate our Lasso regression model based on MSE value.we plot the MSE values of various (λ) values (refer to Fig. 5.3). It can be noted that in the data submitted to lasso regression, outliers have not been removed.

Also we further analyze the impact of each variable on the popularity of the article by including the variables found to be important by lasso regression model in a linear regression model. In this linear regression model We carry out outlier analysis as mentioned in pre-processing (refer to fig.5.2.a) and remove outliers based on cook's distance. Using the beta coefficients of the variables in this linear regression model we explain how change in each variable affects the popularity of the article.

Backward regression is a method which starts with all predictors in the model (full model), iteratively removes the least contributive predictors. We first used backward regression to cut down on insignificant variables. The model retained with significant ones has 27 variables. The multiple linear regression model is predicted with these 27 variables. The dependent variable is the number of shares in social networks of articles published by Mashable in a period of two years.

Checking for VIF values, Checked for multicollinearity among independent variables and removed the variables with high ‘vif’ values. In fig 3b, We removed n_non_stop_words and kw_avg_avg because of their high ‘vif’ values.

Multiple linear regression is a statistical technique that uses several explanatory variables to predict the outcome of a response variable. The goal of multiple linear regression is to model the linear relationship between independent variables and response variables.

As shown in fig 3c, We used Multiple linear regression as we wanted to know the relationship between sales and other independent variables. For every unit increase in average keyword(minimum shares) the number of shares decreases by 0.038 units. If the number of links increases we can observe that the number of shares also increases by 0.031 units. If the number of minimum shares of referenced articles in Mashable increases by one unit we can observe that the shares also increase by 0.038 units. When the data channel is entertainment it negatively affects the number of shares. The average token length negatively affects the number of shares. Our R-squared value is 2 percent and this may be due to very low correlations. We can reject the null hypothesis that our model Is not significant because of the low p-value.

Interpretation for figure 3d, ,As the number of tokens in the title increases the number of shares also increases to a point where it is maximum at 10 and then decreases. Interpretation for figure 3e, As the number of links for a publication increases, the number of shares increases to reach a maximum at the number of links of 4. It then decreases continuously to go to a very low value.

Principal Component Analysis, or PCA, is a dimensionality-reduction method that is often used to reduce the dimensionality of large data sets, by transforming a large set of variables into a smaller one that still contains most of the information in the large set. Transforming the highly correlated data into eigenvectors, which are called principal components. We have used rotation

options for independent composition we use varimax and for dependent composition, we use Obilim for dependent composition.

For factorability, we have used KMO Sampling Adequacy, Bartlett's Test of Sphericity, and Cronbach's Alpha for Reliability. KMO value is 0.6, which is below the threshold point of 0.7, Bartlett's Test of Sphericity: p-value is < 2.22e-16, signifies a very small number,Cronbach's Alpha: raw_alpha = 0.34, which means that the data is highly correlated. To remove multicollinearity and determine which variables are important, we have used Principal Component Analysis (PCA). We have used both the screen plot and the eigenvalue which should be greater than one to determine the number of components for this analysis. By using the screen plot as shown in fig 4b and eigenvalues in fig 4c in the below images, we can determine the number of components as 8.

From the summary, we can see that for 8 components there is 76% of variances shown in fig 4d. There are not too many ways to visualize PCA, but the two many ways are biplot and PCA plot, in the biplot, there is a problem which shows that all the data points along with vectors of the dataset, we cannot predict any pattern there. And in PCA-plot which is shown in fig 4e, we can see that there are too many variables correlated to each other, and kw_max_avg and kw_avg_avg are away and the rest are correlated to each other.

Scaling is considered if there are small-scale effects in which our data might be highly correlated and important. The PCA calculates a new projection of the data set. And the new axis are based on the standard deviation of your variables. Now we have conducted PCA analysis, along with Varimax Rotation for factor rotation, and after multiple reruns removing a lot of variables that are not part of component analysis, we captured 76% variance among the 28

variables using 8 Principal components as shown in fig 4f. Components are named according to the variables, and these components are explained below.

The first component is named “Global subjectivity and positive polarity”, it has seven variables which has subjectivity and positive words of the data, this component consists of variables average_token_length(0.753), global_subjectivity(0.804), global_rate_positive_words(0.670), rate_positive_words(0.732), avg_positive_polarity(0.809), max_positive_polarity(0.804), global_sentiment_polarity(0.589). The second component is named “influence of keywords”, it has three variables which are the keywords of the data, this component consists of variables Kw_max_min(0.940), kw_avg_min(0.926),kw_max_avg(0.796). The third component is named “Global sentiment and negative polarity”, it has three variables which are the global rate of negative words and rate and minimum negative polarity of the data, this component consists of variables Global_rate_negative_words(0.873), rate_negative_words(0.931), min_negative_polarity(-0.641). The fourth component is named “Unique tokens” as it has three variables which are the rate of unique tokens, this component consists of variables n_unique_tokens (1.00), n_non_stop_words(1.00),n_non_stop_unique_tokens(1.00). The fifth component is named “maximum influence of keywords”, it has three variables which has minimum, maximum keywords, this component consists of variables Kw_min_min(-0.928), kw_max_max(0.942),kw_avg_max(0.687) which are keywords. The sixth component is named “self-references of shares”, it has three variables which explains about referenced articles in the dataset, this component consists of variables Self_reference_min_shares(0.848), self_reference_max_shares(0.858), self_reference_avg_sharess(0.986). The seventh component is named “data channels”, it has two variables which say about the data channel of the world and closeness to LDA topic 0, this component consists of variables data_channel_is_world(0.916), LDA_02(0.922). The eighth component is named “keywords and

closeness", it has three variables which say about the average if keywords, this components consists of variables kw_min_avg(0.678),kw_avg_avg(0.614), LDA_03(0.595), shares(0.404).

By focusing on these eight factors in PCA, the dataset should be able to better predict whether an article will be shared on social media. Moreover, the dataset can potentially increase the number of shares for each article by setting the value of each of these attributes such that it maximizes the chance that a reader will share that article.

Common Factor Analysis (CFA) was also performed. Similar to PCA, CFA is a dimensionality reduction technique. It takes a large number of variables and groups them into factors that can be used in further analysis. Unique challenges in regard to interpretation were faced when running CFA due to the fact that several variables from the data set were created using Natural Language Processing (NLP). Variables 39 through 43 in figure 3.1 played particular importance in the final model, all of which utilized the Latent Dirichlet Allocation (LDA) algorithm by finding the top five most popular topics on Mashable and related the article's closeness to the popular topics. (Fernandes et al., 2015) Fernandes et a. (2015) also used NLP to create variables 44 to 59 in figure 3.1 which focuses on the polarity of the article.

Variables 4, 5, and 6 (n_unique_tokens, n_non_stop_words, n_non_stop_unique_tokens) in figure 3.2 were found to be extremely correlated. because they used a similar NLP technique (Fernandes et al., 2015) (Nadkarni et al., 2011) After initially running CFA it was found these three variables were skewing the precision of factor selection and were therefore combined into a single variable (sum_unique) by summing them together. After data preprocessing was finished and the final CFA model completed the variables converged at six factors and explained 68.1% of the variance. After consulting the Scree Plot Fig. 1.2 and trying several different numbers of factors it was found the variance was best explained with 6 factors. Promax rotation was used due to the fact the data had multicollinearity.

Variables were also scaled since different units were used. The final output can be seen in figure 3.2.

Factors were named in accordance with which variables they included. Factor 1 was named ‘Positivity’ because it included variables regarding polarity with positive coefficients on variables regarding positive polarity and negative coefficients on negative polarity. The variables included and their coefficients were as follows: average_token_length (0.606), global_subjectivity (0.506), global_sentiment_polarity (0.726), global_rate_positive_words (0.613), rate_positive_words(0.999), and rate_negative_words. Factor 2 included two variables’ data_channel_is_tech (0.795) and LDA_00 (1.017). Since the found LDA topics were not defined and due to the factor these two variables were grouped together it can be assumed LDA topic 0 relates in some way to business even though the two only have a moderate correlation coefficient. Therefore Factor 2 was named ‘Business’. Factor 3 included the variables data_channel_is_tech(0.747), LDA_04 (0.999), LDA_03 (-0.418) using the same rationale Factor 3 was labeled ‘Technology’. Factor 4 included variables LDA_02 (-0.564), LDA_03 (1.055), and kw_avg_avg (0.484) which posed a challenge in naming since the variables in themselves are unknown in regards to which topics they include. It was decided since LDA_03 and LDA_02 were negative it was decided that the factor be called ‘Related to LDA Topic 2’. The fifth factor again had two factors data_channel_is_entertainment (0.592) and LDA_01 (1.076) and was called ‘Entertainment’. Factor 6 included the variable sum_unique. Since sum_unique was the combination of the variables dealing with the number of tokens it was named ‘Word Count’.

In the canonical correlation analysis (CCA) ten groups were created: (1) Shares, (2) Numbers, (3) Data, (4) Keywords, (5) SelfRef, (6) Day, (7) LDA, (8) Global, (9) Polarity, (10) Title. The first category, shares, consists of our only dependent variable. Shares refer to how many times an article gets shared through a social media platform. Numbers contain information about the number of videos, images, links, words, and other numerical data points pertaining to

the contents of the article. Data consists of the data channels that the dataset looked at. Six categories are included in the data: (1) Lifestyle, (2) Entertainment, (3) Business, (4) Social Media, (5) Tech, and (6) World. Keywords encompass the worst, best and average keywords ranking these keywords by the number of shares articles containing these keywords get. SelfRef incorporates data from shared articles self-referenced in Mashable. Day subsumed data about what day an article gets published and if an article is published on the weekend. LDA comprises a ranking of articles and their closeness to the five LDA topics. Global refers to the polarity, subjectivity, positive content, and negative content within an article's context. The data further separates between neutral and non-neutral tokens. Polarity is the minimum, maximum and average polarity of words within an article. Lastly, title pertains to the subjectivity, absolute subjectivity, polarity, and absolute polarity of titles.

When running the CCA, three hypotheses were tested: (1) Null hypothesis (NH), (2) alternative hypothesis - 1 (AH-1), and (3) alternative hypothesis -2 (AH-2). The NH becomes true if there is no correlation between shares and other independent variables. AH-1 is true only if shares are positively correlated to the independent variable being tested. Lastly, AH-2 denotes as true if and only if shares become negatively correlated by the independent variables. All independent groups except days were tested against the dependent variable resulting in either AH-1 or AH-2 being true. Days were excluded because the day is a categorical variable, not a metric variable. Numbers were also split up into two because of the number of data Numbers contained. Shares always had an extremely positive or extremely negative correlation. This is because CCA is supposed to test for multiple independent variables against multiple dependent variables. Within this dataset, only one dependent variable can be found. To continue the analysis the use of the data variable in place of shares was considered. This was chosen because the channel that each article is published in can give us insights into when an article is shared. When performing the CCA on data and polarity you get six correlation variates. CVs 1

through 5 are all significant, this is supported by Bartlett's Chi-Square test and Wilks Lambda test's df values (Figure 4.1).

However, because the scores of the CCA scores are low, .2258 for CV1 and .2184 for CV2, those are the only two CVs that will be focused on. All other scores are significantly lower creating less impact. Further insight can be drawn from the loadings of CVs 1 and 2. In CV 1, Entertainment and world have the strongest correlation between each x variate and show string correlation with all y variates excluding max_positive_polarity and min_negative_polarity. For visualization of CV1 and CV2 in the form of an helio plot, see figures 4.2 and 4.3. Within CV2, Business, Tech, and World news have strong x variate correlations and strong y variate correlations excluding min_positive_polarity and max_negative_polarity (Figure 4.4). Furthermore, CV1 and CV2 have very low redundancy meaning the significant variables within CV1 and CV2 have a possibility of being good predictive values (Figure 4.5).

Discussion and Results

In order to make the news popular, it is very important to look at what days the particular type of news channels are being watched the most. On days from Monday through Thursday, business news channels are more popular. On Friday, and Sunday world lifestyle is more popular and on Saturday, world news is more popular.

Monday is positively related to both Dimensions(Fig 1.3). Tuesday, Wednesday, and Thursday are positively related to Dimension 1 and negatively related to Dimension 2. Friday is weakly related to dimensions 1 and 2. Saturday is negatively related to both dimensions. Sunday is highly positively related to Dimension 2 and negatively related to dimension 1. In terms of the contribution of weekdays in both dimensions, Saturday, Sunday, and Monday are contributing the most to Dimension 1 and Dimension 2(fig 1.5).

For Channels, Business and Entertainment are surprisingly in the same quadrant(positively related to both dimensions)(Fig 1.4). Technology news is negatively related

to dimension 2 and positively related to dimension 1. Lifestyle news is positively related to Dimension 2 and negatively related to dimension 1. Social media and World news channels are negatively related to both Dimensions. Lifestyle and Business news channels contribute the most to Dimension 1. Technology, Entertainment, and Social Media contributed the most to Dimension 2 (fig 1.6).

Looking at the weekend days and Channels, It is clear that business, social media, and entertainment news channels were more popular on Friday, Saturday, and Sunday respectively. Interestingly, the cumulative variance of the Dimension 2 was increased from 26% to 36% when looking at just the weekend. 2 dimensions were able to explain 100% of the variance in the data. This could be because all the days on the weekend were highly related to dimension 2.

Looking at the weekday group, the cumulative variance increased to 77% from 61% for Dimension 1. The cumulative variance of both the eigenvalues was 100%. On Monday, people tend to watch both entertainment and business news channels. On Tuesday, people tend to watch technology news channels more. On Wednesday and Thursday, business channels are more popular. The weekdays don’t contribute a lot which is why the graph looks a little cluttered.

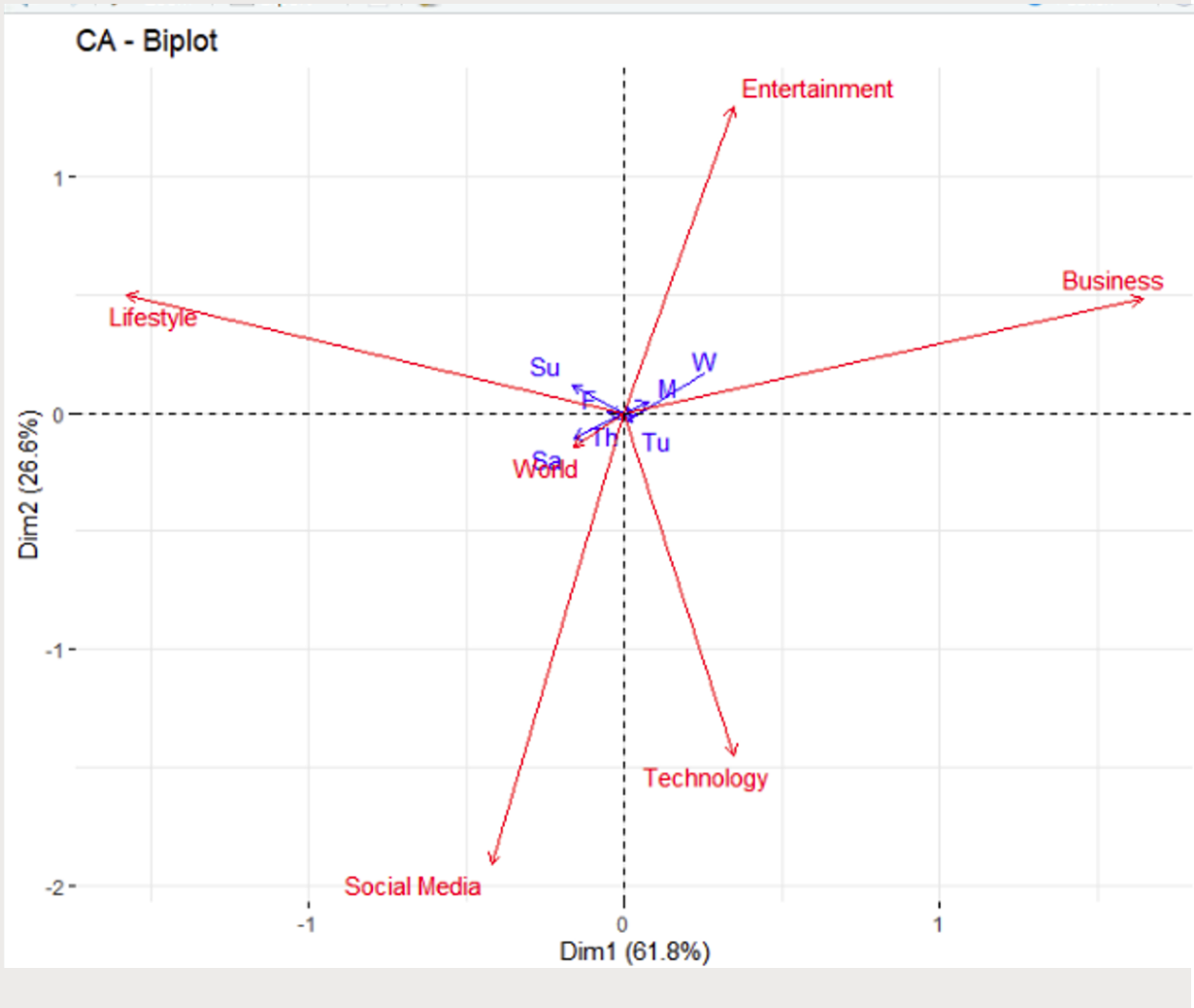


Fig. 1.7 Correspondence plot of Weekday and Channels

In order to understand which features of an article are given the most importance when predicting its popularity we analyze the importance of variables provided by lasso regression in descending order of importance. From fig.5.4.a in appendix let's interpret the coefficients of each variable and conclude its importance in predicting the number of shares achieved by the article. Global subjectivity has the highest coefficient and it conveys that the amount of facts and personal statements in the article has the most impact on the number of shares achieved by the article. We can further analyze its effect on the number of shares using box plots in fig.5.5a.The Second highest coefficient of min positive polarity conveys that the reduced positive magnitude of positive words affects the number of shares achieved by the article. The third

highest coefficient of avg negative polarity conveys that the moderate magnitude of negative words affects the number of shares achieved by the article.The fourth highest coefficient of absolute title subjectivity conveys that the title including or not including personal opinions affects the number of shares achieved by the article.Title sentiment polarity has the fifth highest coefficient and this conveys that the title being a positive or negative sentiment affects the number of shares achieved by the article. The sixth higher coefficient of average_token_length indicates that the Average length of the words in the content affects the number of shares achieved by the article.The seventh highest coefficient of title subjectivity conveys that articles varying in including personal opinions in the title affects the number of shares achieved by the article.

One of the variables among each group of dummy variables has high coefficients. Further analysis using linear regression using variables provided by linear regression could help shed light into significance of dummy variables. The slightly higher coefficient of one of the dummy variables Data_channel_is_entertainment indicates that the data channel in which the articles are published affects the number of shares achieved by the article. (further analysis is required as this could be due to it being a categorical variable, the variable days of the week also falls under the category that requires further analysis). We further analyze the effect of weekdays on shares in Correspondence analysis. To evaluate performance of our model MSE values have been plotted in Fig.5.3 and we get the best model with MSE value 11600000.

Overall we can say that our model predicts the number of shares achieved by the article with criteria such as (1) if an article includes personal opinions and/or facts as the first criteria and (2) intensity of positive words as the second criteria.

We now analyze the relationship of the most important variable in predicting popularity of an article which is global subjectivity. We visualize Its relationship with our dependent variable

number of shares to examine the effect global subjectivity value has on how much the news will be shared.

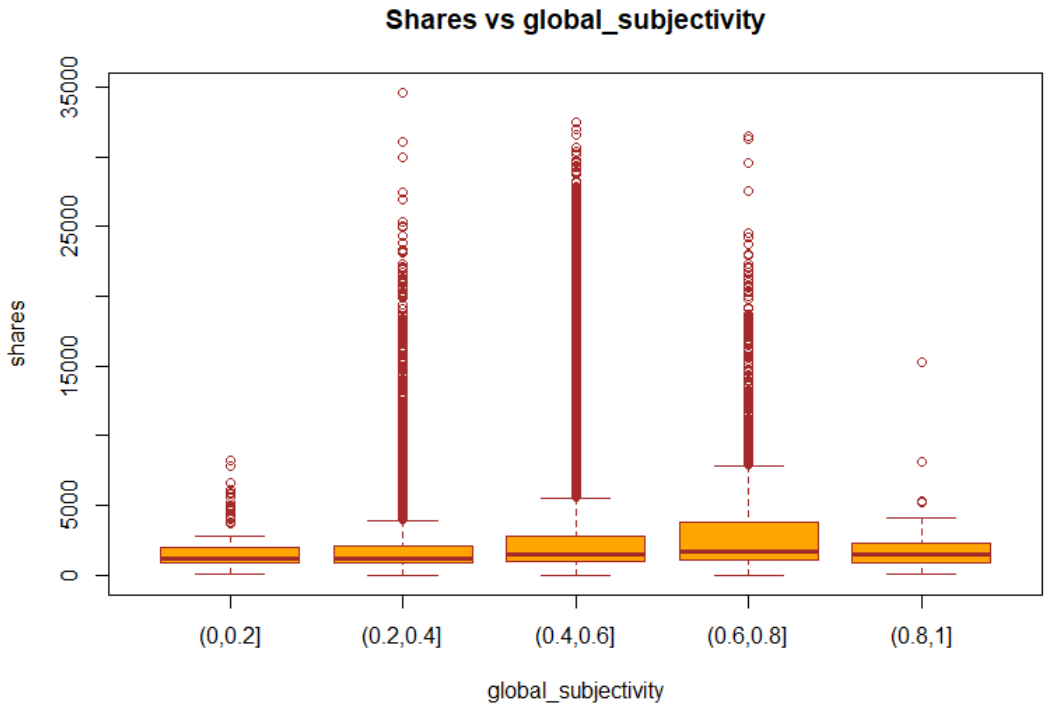


Fig.5.5.a Popularity of articles compared with amount of personal statements vs facts in the article

From fig.5.5.a. We can say that articles with just facts (0,0.2) are shared less and just personal opinions (0.8,1) are also shared less. Equal personal opinion and facts (0.2,0.4), (0.4,0.6),(0.6,0.8) are shared more.Quadrant 3 of global subjectivity (0.6-0.8) has the highest distribution of articles and it also has a long whisker indicating a wider range of shares. The outliers of (0.2,0.4), (0.4,0.6), and (0.6,0.8) are wider ranging from 5,000-35,000 shares indicating a direct effect on the number of shares achieved by the articles.The Mean of global subjectivity (0.6-0.8) is the highest mean across all articles with respect to global subjectivity

and so it can be advised to balance the facts and personal opinions to maximize the popularity of the article

One of the other variables of interest is num of images. Although it has low coefficient values, we analyze it further to find out if it has any influence on the number of shares achieved by the article as it is one of the most distinguishable features of an article. we can now visualize Its relationship with y variables shares to examine the impact of no. of images in our article on how much the news will be shared. we binned the variable no. of images and made 6 bins in total.From fig.5.5.b In appendix we can say that articles with a low number of images (0-20) are shared more.The high shares are represented as outliers for articles having (0-20) images by the box plot but the same behavior can be observed across all articles having (0-20 images)· Thus restraining the number of images used in the article to (0-20) could maximize shares

We ranked balancing facts and personal information in an article to be the most important features of an article that can be changed to improve the number of shares achieved by the article. We were able to rank this feature as the most important using the Lasso regression method.

To further explain how one unit change in one variable affects the popularity of our article we include the variables determined by lasso regression in a linear regression model and interpret the beta coefficients of the model. All 18 variables used in this linear regression model are significant as they have F-value <0.005 and the R-square value improves from 2% to 6%. Lets interpret the effect of each variable on Popularity of the article as follows, From Fig.5.3 when the article is published in Entertainment channel the popularity achieved by the article will decrease by 0.04 units , as references that achieve minimum shares are increased by one unit in the article the popularity of the article increases by 0.007 units , as global subjectivity increases by one unit in the article the popularity of the article increases by 0.001 units, as

negative words have moderate magnitude it decreases the popularity of the article by 0.003 units, as tokens in the article have moderate length it decreases the popularity of the article by 0.002 units, as number of links in the article increases by one unit the popularity of the article increases by 0.014 units, as number of links to articles published by mashable increases by one unit the popularity of the article decreases by 0.016 units, as number of images increases by one unit the popularity of the article increases by 0.016 units, when worst keyword usage that achieve low shares are used the number of shares achieved by the article will increase by 2.8 units(this maybe due to the worst keywords being new keywords that are becoming gradually trending as new articles are shared), if the articles are shared on Tuesdays the popularity of the article decreases by 0.062 units, if the articles are shared on mondays the popularity of the article decreases by 0.04 units, if the articles are shared on wednesday the popularity of the article decreases by 0.059 units, if the articles are shared on thursday the popularity of the article decreases by 0.061 units, if the articles are shared on fridays the popularity of the article decreases by 0.053 units, if the articles shared contain minimum worst keywords the popularity of the article increases by 2.8 units, if the articles shared contain minimum worst keywords the popularity of the article increases by 2.8 units, if the articles shared contains minimum best keywords the popularity of the article decreases by 1.4 units, if the articles shared contains moderate amount of average keywords the popularity of the article decreases by 0.1 units,if the articles shared contains minimum average keywords the popularity of the article decreases by 0.19 units,if the articles shared contains maximum average keywords the popularity of the article decreases by 0.11 units

Further analysis using different methods could help us examine if our findings are spurious correlation or if balancing facts and personal information in an article and making positive words subtly positive in an article has causation on increased no. of shares. In this

context if we expand our analysis to different news websites and if the same phenomenon is observed then we can be sure that our assumption is stable and in this particular context, correlation equals causation.

For CCA, there need to be multiple dependent variables. This dataset lacked that, resulting in CCA becoming a bad analysis to use for this dataset. For suture research, shares can be broken down into multiple categories. As shares track the number of times an article is shared through a social media channel, shares can be broken down into shares through specific social media channels. In other words, shares can be broken down into (1) shares_via_facebook, (2) shares_via_twitter, (3) shares_via_whatsapp, and more. Although when looking at data, Entertainment, Business, Tech, and World news have significant correlation values. They also have the possibility of being good predictor values. To recreate this analysis refer to the code in Figure 4.6. In figure 4.2, you can see that positive polarity has a positive correlation with entertainment. While world news has a negative correlation with positive polarity. Meaning when it comes to world news there is a correlation with negative polarity. Figure 4.3 shows that the world has a positive correlation while almost all of the y variates have negative correlations. Tech and business both have negative correlations to maximum negative polarity, average negative polarity, and maximum positive polarity.

Common Factor Analysis further proved that entertainment, business, and technology along with variables of polarity to be important. After running CFA it was determined that the 61 variables could be paired down to 6 factors: positivity, business, technology, related to LDA topic 2, entertainment, and word count.

Based on the model results, we found following variables to be significantly positively influencing the shares: N_tokens_title, N_tokens_content, N_unique_tokens, Num_hrefs, Kw_max_min, Self_refrence_min_shares, Self_reference_max_shares, Weekday is Monday, Weekday is Saturday, Global subjectivity, Abs_title_sentiment polarity.

The following set of variables are found to be negatively influencing the shares: N_non_stop_words, Num_self_hrefs, Average token length, Data channel is lifestyle, Data channel is entertainment, Kw_avg_min, Kw_min_max, Kw_min_avg, Kw_max_avg, LDA_02 Global_rate_positive_words, Min_positive_polarity, Avg_negative_polarity.

Limitations:

Limitations and future work that were found within the study have been deduced to three points: (1) dataset is from a singular website, (2) dataset is from the social sciences category, and (3) features that improve the number of shares. The Dataset used in our analysis is composed of articles published by a single website, Mashable. Expanding our analysis to different news websites and determining if the same variables play a major role in determining the popularity of the news article is an important future direction. Our Dataset falls under the social sciences category, resulting in low correlation values between dependent and independent variables. Estimation of latent variables that affect the number of shares using factor analysis or discriminant analysis is another direction. Optimization of the features in an article that improves the number of shares could be a good direction. More shares of an article allow for more clicks, thus creating more ad revenue for the article's publication.

Conclusion

We went through the correspondence analysis, lasso regression, multiple linear regression, and the PCA. However the common factor analysis and CCA gave insight From the correspondence analysis, it was clear that on Monday - Thursday, the business channel is more popular. On weekends lifestyle and world news channels were more popular. We were able to rank the features of an article based on their importance in predicting popularity of the article using the lasso regression method. We ranked balancing personal statements and facts in an article to be the most important change to be made to improve the popularity achieved by the

article. 61PCA is used to group all the variables into 8 components, which captures the 67% variance of the variables. As the data is highly skewed, future analysis is that we are required to do mixed PCA. Using CFA, we were able to group variables into 6 factors, positivity, business, technology, related to LDA topic 2, entertainment, and word count accounting for 68.1% of the cumulative variance. CCA showed that shares were always extremely positively affected or extremely negatively affected; never supporting the null hypothesis. However, when changing the dependent variable to data, the CCA showed there are data channels Entertainment, Business, Tech, and World are all significantly correlated and have low redundancy allowing for the ability to become good predictors.

References:

Fernandes, K., Vinagre, P., & Cortez, P. (2015). A proactive intelligent decision support system for predicting the popularity of online news. Progress in Artificial Intelligence, 535–546. https://doi.org/10.1007/978-3-319-23485-4_53

Jowkar, A., & Didegah, F. (2010). Evaluating Iranian newspapers' websites using correspondence analysis. Library Hi Tech, 28(1), 119–130. https://doi.org/10.1108/07378831011026733

Moro, S., Rita, P., & Vala, B. (2016). Predicting Social Media Performance Metrics and evaluation of the impact on brand building: A Data Mining Approach. Journal of Business Research, 69(9), 3341–3351. https://doi.org/10.1016/j.jbusres.2016.02.010

Nadkarni, P. M., Ohno-Machado, L., & Chapman, W. W. (2011). Natural language processing: An introduction. *Journal of the American Medical Informatics Association*, *18*(5), 544–551. https://doi.org/10.1136/amiajnl-2011-000464

Appendix:

Preprocessing:

```
   kw_min_min        kw_max_min        kw_avg_min        kw_min_max        kw_max_max
Min.   : -1.00   Min.   :     0   Min.   :   -1.0   Min.   :     0   Min.   :     0
1st Qu.: -1.00   1st Qu.:   445   1st Qu.:  141.8   1st Qu.:     0   1st Qu.:843300
Median : -1.00   Median :   660   Median :  235.5   Median :  1400   Median :843300
Mean   : 26.11   Mean   :  1154   Mean   :  312.4   Mean   : 13612   Mean   :752324
3rd Qu.:  4.00   3rd Qu.:  1000   3rd Qu.:  357.0   3rd Qu.:  7900   3rd Qu.:843300
Max.   :377.00   Max.   :298400   Max.   :42827.9   Max.   :843300   Max.   :843300
```

Fig 1a

What is the number of shares of articles published by Mashable in two years period?

```
Residuals:
   Min     1Q Median     3Q    Max
-2.537 -0.198 -0.105 -0.006 72.033

Coefficients:
                               Estimate Std. Error t value Pr(>|t|)
(Intercept)                   4.328e-15  4.966e-03   0.000 1.000000
timedelta                     3.224e-02  6.724e-03   4.794 1.64e-06 ***
n_tokens_title                2.145e-02  5.245e-03   4.089 4.35e-05 ***
n_tokens_content              2.075e-02  8.201e-03   2.530 0.011407 *
n_unique_tokens               6.173e-01  2.748e-01   2.247 0.024672 *
n_non_stop_words             -6.151e-01  2.748e-01  -2.238 0.025222 *
num_hrefs                     2.724e-02  6.195e-03   4.397 1.10e-05 ***
num_self_hrefs               -1.972e-02  5.719e-03  -3.449 0.000564 ***
num_imgs                      1.092e-02  5.736e-03   1.903 0.057040 .
average_token_length         -2.289e-02  6.964e-03  -3.286 0.001017 **
num_keywords                  9.874e-03  5.511e-03   1.791 0.073223 .
data_channel_is_lifestyle    -1.122e-02  5.131e-03  -2.186 0.028824 *
data_channel_is_entertainment -2.795e-02  5.321e-03  -5.252 1.51e-07 ***
kw_min_min                    1.143e-02  6.359e-03   1.797 0.072360 .
kw_max_min                    3.992e-02  1.630e-02   2.449 0.014342 *
kw_avg_min                   -3.134e-02  1.596e-02  -1.963 0.049668 *
kw_min_max                   -1.264e-02  5.457e-03  -2.316 0.020586 *
kw_min_avg                   -3.827e-02  7.150e-03  -5.352 8.75e-08 ***
kw_max_avg                   -1.160e-01  1.206e-02  -9.621  < 2e-16 ***
kw_avg_avg                    2.044e-01  1.344e-02  15.201  < 2e-16 ***
self_reference_min_shares     3.843e-02  5.712e-03   6.728 1.74e-11 ***
self_reference_max_shares     1.225e-02  5.858e-03   2.091 0.036546 *
weekday_is_monday             1.518e-02  5.007e-03   3.032 0.002434 **
weekday_is_saturday           1.222e-02  5.022e-03   2.433 0.014971 *
LDA_02                       -1.805e-02  5.998e-03  -3.009 0.002626 **
global_subjectivity           2.570e-02  7.562e-03   3.398 0.000679 ***
global_rate_positive_words   -1.667e-02  6.294e-03  -2.648 0.008105 **
min_positive_polarity        -1.384e-02  5.773e-03  -2.397 0.016518 *
avg_negative_polarity        -1.729e-02  5.714e-03  -3.025 0.002485 **
abs_title_subjectivity        1.085e-02  5.547e-03   1.955 0.050535 .
abs_title_sentiment_polarity  1.224e-02  5.520e-03   2.218 0.026573 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.9887 on 39613 degrees of freedom
Multiple R-squared:  0.02319,   Adjusted R-squared:  0.02245
F-statistic: 31.35 on 30 and 39613 DF,  p-value: < 2.2e-16
```

Fig 3a

```
> VIF(model1)
                num_keywords                   kw_min_min                     num_imgs        abs_title_subjectivity
                    1.224535                     1.595792                     1.309166                     1.221462
                  kw_avg_min      self_reference_max_shares      data_channel_is_lifestyle  abs_title_sentiment_polarity
                    1.524796                     1.390810                     1.050254                     1.231105
             n_non_stop_words               n_unique_tokens                   kw_min_max         min_positive_polarity
                 1730.790412                  1729.798167                     1.203779                     1.231166
           weekday_is_monday          average_token_length          global_subjectivity                num_self_hrefs
                    1.002890                     1.854815                     1.755238                     1.286698
             n_tokens_title                    num_hrefs                    timedelta data_channel_is_entertainment
                    1.113252                     1.467056                     1.730543                     1.060451
                  kw_min_avg      self_reference_min_shares                   kw_max_avg                    kw_avg_avg
                    1.998755                     1.322072                     5.061971                     6.246303
```

Fig 3b

```
Residuals:
   Min     1Q Median     3Q    Max
-2.701 -0.200 -0.127 -0.027 72.099

Coefficients:
                               Estimate Std. Error t value Pr(>|t|)
(Intercept)                   -2.864e-15  4.988e-03   0.000  1.00000
num_keywords                   1.471e-02  5.503e-03   2.674  0.00751 **
num_imgs                       2.294e-02  5.417e-03   4.234 2.30e-05 ***
abs_title_subjectivity         1.222e-02  5.507e-03   2.218  0.02654 *
self_reference_max_shares      1.465e-02  5.862e-03   2.499  0.01245 *
data_channel_is_lifestyle     -5.698e-03  5.094e-03  -1.118  0.26336
abs_title_sentiment_polarity   1.701e-02  5.527e-03   3.077  0.00209 **
n_unique_tokens                2.753e-03  4.993e-03   0.551  0.58136
kw_min_max                    -6.918e-03  5.450e-03  -1.269  0.20433
weekday_is_monday              1.267e-02  4.994e-03   2.538  0.01116 *
average_token_length          -5.777e-02  6.444e-03  -8.966  < 2e-16 ***
global_subjectivity            5.315e-02  6.388e-03   8.321  < 2e-16 ***
num_self_hrefs                -2.520e-02  5.587e-03  -4.510 6.51e-06 ***
n_tokens_title                 1.355e-02  5.109e-03   2.653  0.00799 **
num_hrefs                      4.159e-02  5.847e-03   7.113 1.15e-12 ***
data_channel_is_entertainment -2.275e-02  5.131e-03  -4.434 9.28e-06 ***
kw_min_avg                     3.088e-02  5.559e-03   5.555 2.80e-08 ***
self_reference_min_shares      4.185e-02  5.731e-03   7.303 2.87e-13 ***
kw_max_avg                     4.383e-02  5.188e-03   8.448  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.9931 on 39625 degrees of freedom
Multiple R-squared:  0.01423,   Adjusted R-squared:  0.01379
F-statistic: 31.79 on 18 and 39625 DF,  p-value: < 2.2e-16
```

Fig 3c

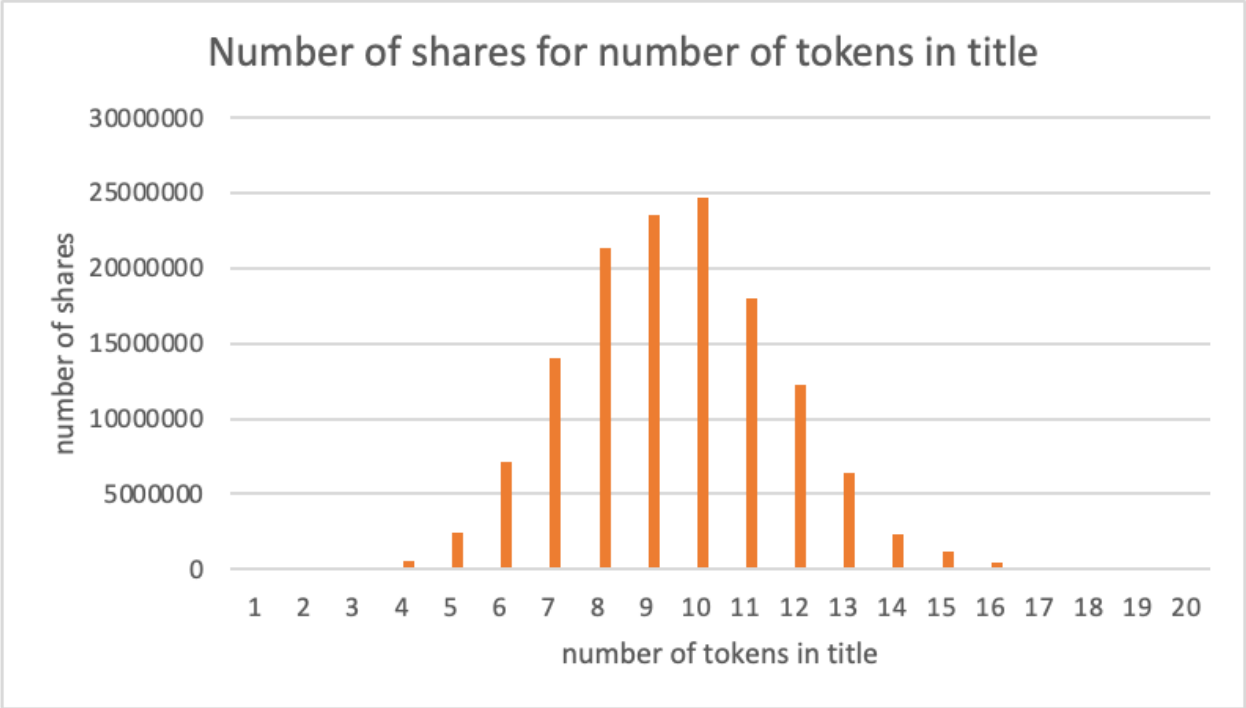

Number of shares for number of tokens in title
30000000
25000000
20000000
15000000
10000000
5000000
0
number of shares
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
number of tokens in title

Fig 3d

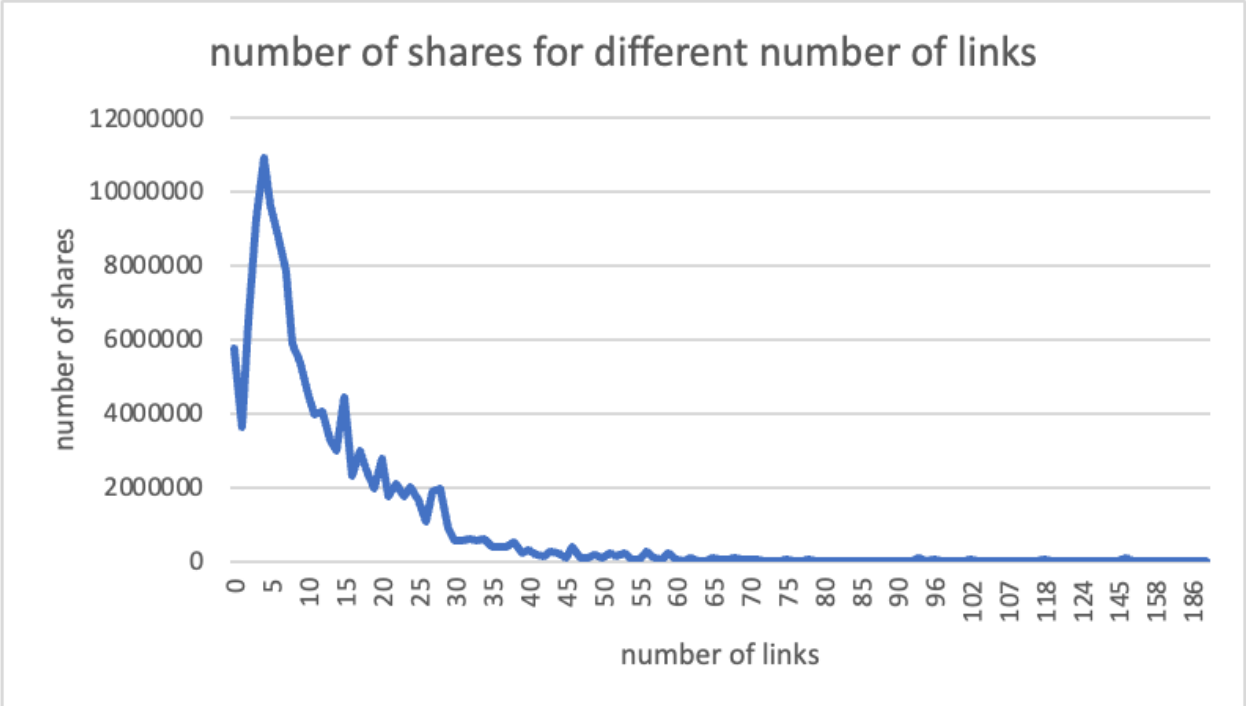

number of shares for different number of links
12000000
10000000
8000000
6000000
4000000
2000000
0
number of shares
0 5 10 15 20 25 30 35 40 45 50 55 60 65 70 75 80 85 90 96 102 107 118 124 145 158 186
number of links

Fig 3e

```
> vif(M1)
Error in vif.default(M1) : there are aliased coefficients in the model
```

Fig.5.1a

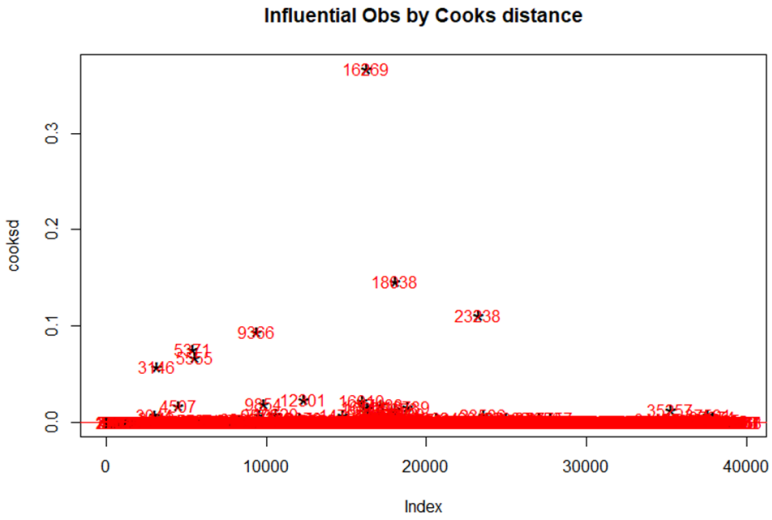


Fig.5.2.a

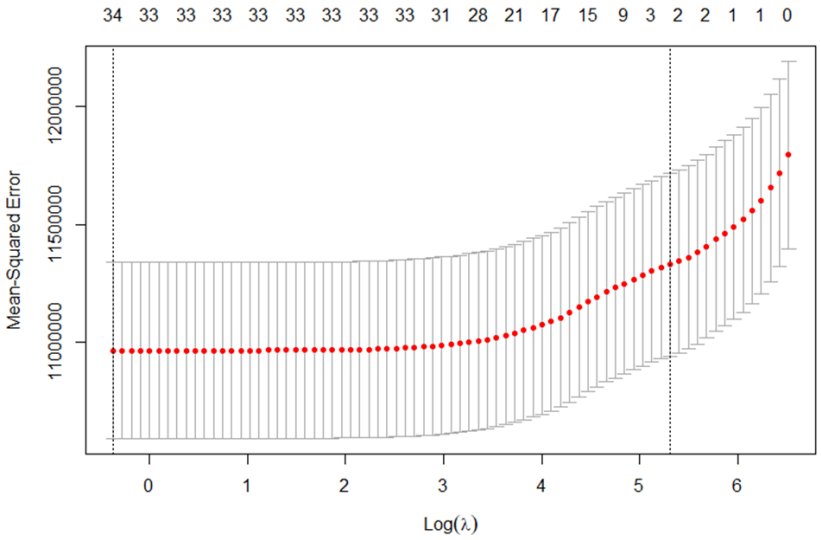


Fig 5.3

```
> best_lambda
[1] 0.5751992
> plot(cv_model)
> best_model <- glmnet(x, y, alpha = 1, lambda = best_lambda)
> coef(best_model)
30 x 1 sparse Matrix of class "dgCMatrix"
                                        s0
(Intercept)                    848.5472198809
kw_min_min                       1.7806715052
kw_avg_max                      -0.0013067024
kw_min_avg                      -0.1942711206
kw_max_avg                      -0.1266185186
kw_avg_avg                       1.0753626723
kw_min_max                      -0.0007994473
global_subjectivity           1451.3714172013
title_sentiment_polarity       286.8386096295
abs_title_subjectivity         335.5901273878
self_reference_min_shares        0.0276934674
title_subjectivity             124.5449785391
data_channel_is_lifestyle     -204.4290585567
data_channel_is_entertainment -527.5292010769
data_channel_is_bus           -151.5508939589
average_token_length          -210.0556682721
num_videos                       3.8965675451
num_hrefs                       10.0356906984
num_self_hrefs                 -11.3641203779
num_imgs                         6.7226074550
n_tokens_title                  11.7230365361
n_tokens_content                 0.0438734119
weekday_is_monday             -314.1378448334
weekday_is_tuesday            -546.8797441206
weekday_is_wednesday          -531.2199662051
weekday_is_thursday           -517.9435811831
weekday_is_friday             -413.2314806019
min_positive_polarity         -902.8096194829
avg_negative_polarity         -352.6405499551
max_negative_polarity          246.7164253382
```

Fig. 5.4

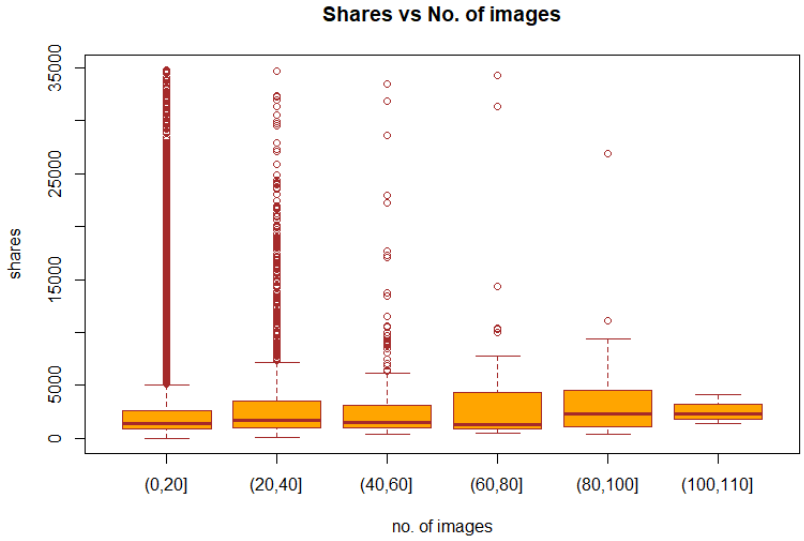


Fig.5.5.b

```
Call:
lm(formula = shares ~ ., data = newdata)

Residuals:
   Min     1Q Median     3Q    Max
-19797  -1559   -939     46  33104

Coefficients:
                               Estimate Std. Error t value Pr(>|t|)
(Intercept)                   1.048e+03  1.424e+02   7.357 1.92e-13 ***
data_channel_is_entertainment -4.498e+02  4.941e+01  -9.104  < 2e-16 ***
kw_min_min                     2.827e+00  2.740e-01  10.319  < 2e-16 ***
kw_min_max                    -1.402e-03  3.426e-04  -4.091 4.30e-05 ***
kw_min_avg                    -1.990e-01  2.244e-02  -8.869  < 2e-16 ***
kw_max_avg                    -1.174e-01  6.357e-03 -18.464  < 2e-16 ***
kw_avg_avg                     1.018e+00  3.422e-02  29.753  < 2e-16 ***
self_reference_min_shares      7.956e-03  9.510e-04   8.365  < 2e-16 ***
weekday_is_monday             -4.047e+02  6.854e+01  -5.904 3.57e-09 ***
weekday_is_tuesday            -6.205e+02  6.699e+01  -9.263  < 2e-16 ***
weekday_is_wednesday          -5.916e+02  6.699e+01  -8.832  < 2e-16 ***
weekday_is_thursday           -6.115e+02  6.724e+01  -9.095  < 2e-16 ***
weekday_is_friday             -5.343e+02  7.094e+01  -7.532 5.11e-14 ***
global_subjectivity            1.812e+03  2.155e+02   8.411  < 2e-16 ***
avg_negative_polarity         -3.253e+02  1.649e+02  -1.973  0.04845 *
average_token_length          -2.676e+02  2.927e+01  -9.144  < 2e-16 ***
num_hrefs                      1.436e+01  1.933e+00   7.427 1.13e-13 ***
num_self_hrefs                -1.601e+01  5.321e+00  -3.009  0.00262 **
num_imgs                       1.662e+01  2.436e+00   6.823 9.01e-12 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3673 on 39274 degrees of freedom
Multiple R-squared:  0.06179,   Adjusted R-squared:  0.06136
F-statistic: 143.7 on 18 and 39274 DF,  p-value: < 2.2e-16
```

Fig. 5.6

```
Canonical Correlations:

       CV 1        CV 2        CV 3        CV 4        CV 5        CV 6
0.2258133590 0.2184185561 0.1575749098 0.0755044743 0.0519119888 0.0008313091

Shared Variance on Each Canonical Variate:

       CV 1        CV 2        CV 3        CV 4        CV 5        CV 6
5.099167e-02 4.770667e-02 2.482985e-02 5.700926e-03 2.694855e-03 6.910748e-07

Bartlett's Chi-Squared Test:

         rho^2      Chisq df Pr(>X)
CV 1 5.0992e-02 5.3422e+03 36 <2e-16 ***
CV 2 4.7707e-02 3.2677e+03 25 <2e-16 ***
CV 3 2.4830e-02 1.3302e+03 16 <2e-16 ***
CV 4 5.7009e-03 3.3360e+02  9 <2e-16 ***
CV 5 2.6949e-03 1.0699e+02  4 <2e-16 ***
CV 6 6.9107e-07 2.7392e-02  1 0.8685
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Figure 4.1

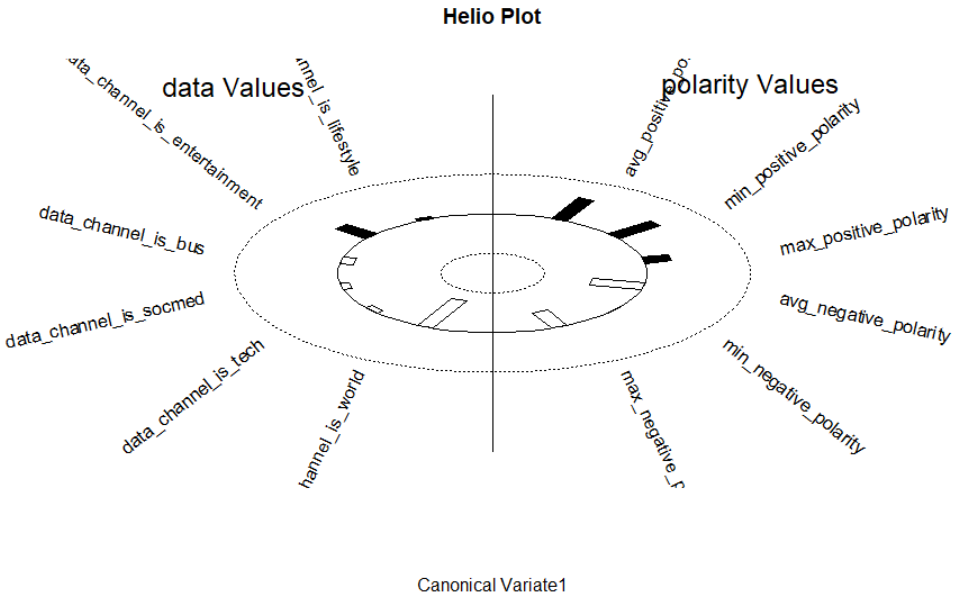


Figure 4.2

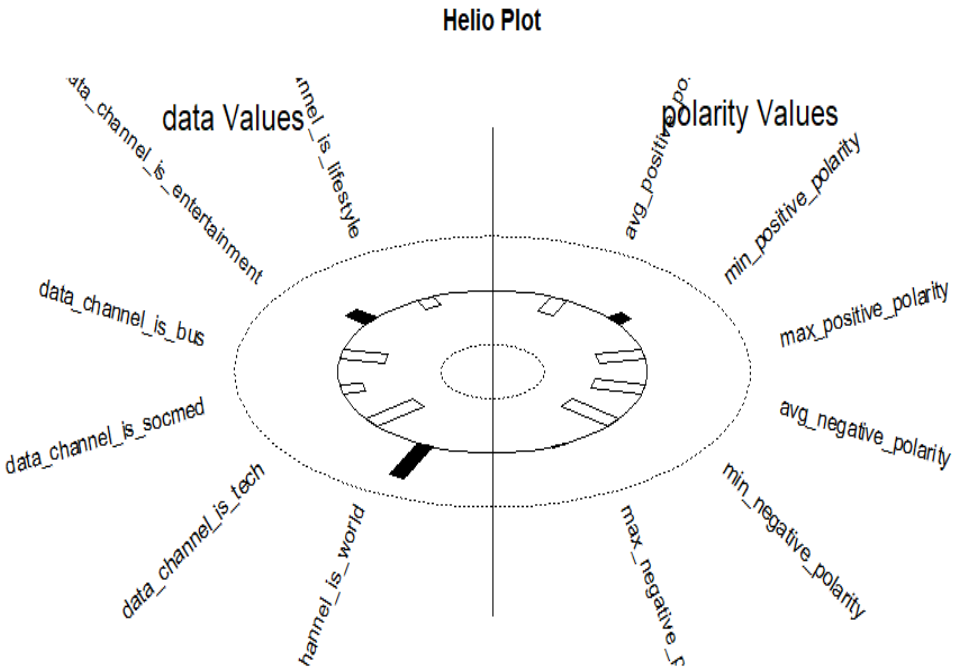


Figure 4.3

```
Structural Correlations (Loadings):

        X Vars:
                                    CV 1       CV 2        CV 3        CV 4       CV 5         CV 6
data_channel_is_lifestyle      0.05126900 -0.1828773 -0.29326141 -0.24777090  0.8783627  0.212164257
data_channel_is_entertainment  0.37705939  0.2514640 -0.72385965 -0.30446971 -0.4217929 -0.002864913
data_channel_is_bus           -0.13220200 -0.4495425 -0.02277258  0.50028945 -0.2352518  0.688681786
data_channel_is_socmed        -0.09160011 -0.2247477 -0.28476109  0.58302604  0.1320335 -0.708982935
data_channel_is_tech          -0.07792524 -0.5806149  0.42157790 -0.58242768 -0.1905224 -0.321815663
data_channel_is_world         -0.75102707  0.6403279  0.13893832 -0.03750349  0.0722121  0.003683664

        Y Vars:
                           CV 1         CV 2        CV 3       CV 4         CV 5        CV 6
avg_positive_polarity  0.58488771 -0.311938257 -0.26403553 -0.2176627  0.662790391 -0.06494592
min_positive_polarity  0.51793373  0.155404434  0.58459297 -0.4220862  0.379309987 -0.20931400
max_positive_polarity  0.26582794 -0.457605717 -0.63928356 -0.4561219  0.181092892 -0.26534362
avg_negative_polarity -0.51934694 -0.502976354  0.51315090 -0.1788703 -0.006917571  0.42652933
min_negative_polarity  0.01225821 -0.578378183  0.68595285  0.1218974 -0.289723449  0.30983616
max_negative_polarity -0.43198209  0.007198987 -0.08450613 -0.5343251  0.035202025  0.72073285
```

Figure 4.4

```
        X | Y:
        CV 1         CV 2         CV 3         CV 4         CV 5         CV 6
6.295638e-03 8.717691e-03 3.677360e-03 1.030851e-03 4.777618e-04 1.296387e-07

        Y | X:
        CV 1         CV 2         CV 3         CV 4         CV 5         CV 6
9.667116e-03 7.302438e-03 6.460494e-03 7.277591e-04 3.149335e-04 1.054833e-07


Aggregate Redundancy Coefficients (Total Variance
Explained by All CVs, Across Sets):

        X | Y: 0.02019943
        Y | X: 0.02447285
```

Figure 4.5

```r
library(yacca)

#Read in Data
setwd("C:/Users/jdoretti/Documents/DSC 424/Project")

ONP = read.csv("OnlineNewsPopularity.csv", header = TRUE, sep = ",")
head(ONP)

#See the first six lines of the data
head(ONP)

names(ONP)

shares = ONP[, 61]
numbers = ONP[, 3:7]
numbers2 = ONP[, 8:13]
data = ONP[, 14:19]
keyword = ONP[, 20:28]
selfRef = ONP[, 29:31]
day = ONP[, 32:39]
LDA = ONP[, 40:44]
global = ONP[, 45:50]
polarity = ONP[, 51:56]
title = ONP[, 57:60]

#Numbers
# This gives us the cannonical correlates, but no significance tests
c = cca(shares,numbers)
summary(c)

#CV1
helio.plot(c, cv=1, x.name="shares Values",
y.name="numbers Values")

#Function Names
ls(c)

# Perform a chi-square test on C
c
ls(c)
c$chisq
c$df
summary(c)
round(pchisq(c$chisq, c$df, lower.tail=F), 3)

#Numbers
# This gives us the cannonical correlates, but no significance tests
cc = cca(shares,numbers2)
summary(cc)

#CV1
helio.plot(cc, cv=1, x.name="shares Values",
           y.name="numbers2 Values")
```

```r
# Perform a chi-square test on C
cc
ls(cc)
cc$chisq
cc$df
summary(cc)
round(pchisq(cc$chisq, cc$df, lower.tail=F), 3)

#Data
# This gives us the cannonical correlates, but no significance tests
c2 = cca(shares,data)
summary(c2)

#CV1
helio.plot(c2, cv=1, x.name="shares Values",
           y.name="Data Values")

# Perform a chi-square test on C2
c2
ls(c2)
c2$chisq
c2$df
summary(c2)
round(pchisq(c2$chisq, c2$df, lower.tail=F), 3)

#Keywords
# This gives us the cannonical correlates, but no significance tests
c3 = cca(shares,keyword)
summary(c3)

#CV1
helio.plot(c3, cv=1, x.name="shares Values",
           y.name="keyword Values")


# Perform a chi-square test on C2
c3
ls(c3)
c3$chisq
c3$df
summary(c3)
round(pchisq(c3$chisq, c3$df, lower.tail=F), 3)

#selfRef
# This gives us the cannonical correlates, but no significance tests
c4 = cca(shares,selfRef)
summary(c4)

#CV1
helio.plot(c4, cv=1, x.name="shares Values",
           y.name="selfRef Values")

# Perform a chi-square test on C2
c4
ls(c4)
c4$chisq
c4$df
summary(c4)
```

```r
round(pchisq(c4$chisq, c4$df, lower.tail=F), 3)

# #day - could not use because categorial
# c5 = cca(shares,day)
# summary(c5)
#
# #CV1
# helio.plot(c5, cv=1, x.name="shares Values",
#            y.name="day Values")
#
#
# # Perform a chi-square test on C2
# c5
# ls(c5)
# c5$chisq
# c5$df
# summary(c5)
# round(pchisq(c5$chisq, c5$df, lower.tail=F), 3)

#LDA
# This gives us the cannonical correlates, but no significance tests
c6 = cca(shares,LDA)
summary(c6)

#CV1
helio.plot(c6, cv=1, x.name="shares Values",
           y.name="LDA Values")

#Function Names
ls(c6)

# Perform a chi-square test on C2
c6
ls(c6)
c6$chisq
c6$df
summary(c6)
round(pchisq(c6$chisq, c6$df, lower.tail=F), 3)

#global
# This gives us the cannonical correlates, but no significance tests
c7 = cca(shares,global)
summary(c7)

#CV1
helio.plot(c7, cv=1, x.name="shares Values",
           y.name="global Values")

#Function Names
ls(c7)

# Perform a chi-square test on C2
c7
ls(c7)
c7$chisq
c7$df
summary(c7)
round(pchisq(c7$chisq, c7$df, lower.tail=F), 3)
```

```r
#polarity
# This gives us the cannonical correlates, but no significance tests
c8 = cca(shares,polarity)
summary(c8)

#CV1
helio.plot(c8, cv=1, x.name="shares Values",
           y.name="polarity Values")

#Function Names
ls(c8)

# Perform a chi-square test on C2
c8
ls(c8)
c8$chisq
c8$df
summary(c8)
round(pchisq(c8$chisq, c8$df, lower.tail=F), 3)

#title
# This gives us the cannonical correlates, but no significance tests
c9 = cca(shares, title)
summary(c9)

#CV1
helio.plot(c9, cv=1, x.name="shares Values",
           y.name="title Values")

#Function Names
ls(c9)

# Perform a chi-square test on C2
c9
ls(c9)
c9$chisq
c9$df
summary(c9)
round(pchisq(c9$chisq, c9$df, lower.tail=F), 3)

#idv - data, title
c9 = cca(data, title)
summary(c9)

#CV1
helio.plot(c9, cv=1, x.name="data Values",
           y.name="title Values")

#CV2
helio.plot(c9, cv=2, x.name="data Values",
           y.name="title Values")

#CV3
helio.plot(c9, cv=3, x.name="data Values",
           y.name="title Values")
```

```r
#CV4
helio.plot(c9, cv=4, x.name="data Values",
           y.name="title Values")

#Function Names
ls(c9)

# Perform a chi-square test on C2
c9
ls(c9)
c9$chisq
c9$df
summary(c9)
round(pchisq(c9$chisq, c9$df, lower.tail=F), 3)

#idv - data, keyword
c9 = cca(data, keyword)
summary(c9)

#CV1
helio.plot(c9, cv=1, x.name="data Values",
           y.name="keyword Values")

#CV2
helio.plot(c9, cv=2, x.name="data Values",
           y.name="keyword Values")

#CV3
helio.plot(c9, cv=3, x.name="data Values",
           y.name="keyword Values")

#CV4
helio.plot(c9, cv=4, x.name="data Values",
           y.name="keyword Values")

#Function Names
ls(c9)

# Perform a chi-square test on C2
c9
ls(c9)
c9$chisq
c9$df
summary(c9)
round(pchisq(c9$chisq, c9$df, lower.tail=F), 3)

#idv - data, selfref
c9 = cca(data, selfRef)
summary(c9)

#CV1
helio.plot(c9, cv=1, x.name="data Values",
           y.name="selfRef Values")

#CV2
helio.plot(c9, cv=2, x.name="data Values",
           y.name="selfRef Values")
```

```
#CV3
helio.plot(c9, cv=3, x.name="data Values",
           y.name="selfRef Values")

#Function Names
ls(c9)

# Perform a chi-square test on C2
c9
ls(c9)
c9$chisq
c9$df
summary(c9)
round(pchisq(c9$chisq, c9$df, lower.tail=F), 3)


#idv - data, polarity
c9 = cca(data, polarity)

#CV1
helio.plot(c9, cv=1, x.name="data Values",
           y.name="polarity Values")

#CV2
helio.plot(c9, cv=2, x.name="data Values",
           y.name="polarity Values")

#CV3
helio.plot(c9, cv=3, x.name="data Values",
           y.name="polarity Values")

#CV3
helio.plot(c9, cv=6, x.name="data Values",
           y.name="polarity Values")


# Perform a chi-square test
c9
ls(c9)
c9$chisq
c9$df
summary(c9)
round(pchisq(c9$chisq, c9$df, lower.tail=F), 3)
```

Figure 4.6

Figure 3.1

0	url	URL of the article (non-predictive)
1	timedelta	Days between the article publication and the dataset acquisition (non-predictive)

2	n_tokens_title	Number of words in the title
3	n_tokens_content	Number of words in the content
4	n_unique_tokens	Rate of unique words in the content
5	n_non_stop_words	Rate of non-stop words in the content
6	n_non_stop_unique_tokens	Rate of unique non-stop words in the content
7	num_hrefs	Number of links
8	num_self_hrefs	Number of links to other articles published by Mashable
9	num_imgs	Number of images
10	num_videos	Number of videos
11	average_token_length	Average length of the words in the content
12	num_keywords	Number of keywords in the metadata
13	data_channel_is_lifestyle	Is data channel 'Lifestyle'?
14	data_channel_is_entertainment	Is data channel 'Entertainment'?
15	data_channel_is_bus	Is data channel 'Business'?
16	data_channel_is_socmed	Is data channel 'Social Media'?
17	data_channel_is_tech	Is data channel 'Tech'?
18	data_channel_is_world	Is data channel 'World'?
19	kw_min_min	Worst keyword (min . shares)
20	kw_max_min	Worst keyword (max. shares)
21	kw_avg_min	Worst keyword (avg. shares)
22	kw_min_max	Best keyword (min. shares)
23	kw_max_max	Best keyword (max. shares)
24	kw_avg_max	Best keyword (avg. shares)
25	kw_min_avg	Avg. keyword (min
26	kw_max_avg	Avg. keyword (max
27	kw_avg_avg	Avg. keyword (avg
28	self_reference_min_shares	Min. shares of referenced articles in Mashable
29	self_reference_max_shares	Max. shares of referenced articles in Mashable
30	self_reference_avg_sharess	Avg. shares of referenced articles in Mashable
31	weekday_is_monday	Was the article published on a Monday?
32	weekday_is_tuesday	Was the article published on a Tuesday?
33	weekday_is_wednesday	Was the article published on a Wednesday?
34	weekday_is_thursday	Was the article published on a Thursday?

35	weekday_is_friday	Was the article published on a Friday?
36	weekday_is_saturday	Was the article published on a Saturday?
37	weekday_is_sunday	Was the article published on a Sunday?
38	is_weekend	Was the article published on the weekend?
39	LDA_00	Closeness to LDA topic 0
40	LDA_01	Closeness to LDA topic 1
41	LDA_02	Closeness to LDA topic 2
42	LDA_03	Closeness to LDA topic 3
43	LDA_04	Closeness to LDA topic 4
44	global_subjectivity	Text subjectivity
45	global_sentiment_polarity	Text sentiment polarity
46	global_rate_positive_words	Rate of positive words in the content
47	global_rate_negative_words	Rate of negative words in the content
48	rate_positive_words	Rate of positive words among non-neutral tokens
49	rate_negative_words	Rate of negative words among non-neutral tokens
50	avg_positive_polarity	Avg. polarity of positive words
51	min_positive_polarity	21 Min. polarity of positive words
52	max_positive_polarity	Max. polarity of positive words
53	avg_negative_polarity	Avg. polarity of negative words
54	min_negative_polarity	Min. polarity of negative words
55	max_negative_polarity	Max. polarity of negative words
56	title_subjectivity	Title subjectivity
57	title_sentiment_polarity	Title polarity
58	abs_title_subjectivity	Absolute subjectivity level
59	abs_title_sentiment_polarity	Absolute polarity level
60	shares	Number of shares (target)

Figure 3.2

```
Loadings:
                           Factor1 Factor2 Factor3 Factor4 Factor5 Factor6
average_token_length        0.606
global_subjectivity         0.506
global_sentiment_polarity   0.726
global_rate_positive_words  0.613
rate_positive_words         0.999
rate_negative_words        -0.518
data_channel_is_bus                 0.795
LDA_00                              1.017
data_channel_is_tech                        0.747
LDA_04                                      0.999
LDA_02                                     -0.418  -0.564
LDA_03                                              1.055
data_channel_is_entertainment                                0.592
LDA_01                                                       1.076
sum_unique                                                           0.998
kw_avg_avg                                          0.484
global_rate_negative_words

               Factor1 Factor2 Factor3 Factor4 Factor5 Factor6
SS loadings      2.959   2.005   1.957   1.917   1.740   0.996
Proportion Var   0.174   0.118   0.115   0.113   0.102   0.059
Cumulative Var   0.174   0.292   0.407   0.520   0.622   0.681
```

Figure 3.3 (Fernandes et al., 2015)

Feature	Type (#)
Words	
Number of words in the title	number (1)
Number of words in the article	number (1)
Average word length	number (1)
Rate of non-stop words	ratio (1)
Rate of unique words	ratio (1)
Rate of unique non-stop words	ratio (1)
Links	
Number of links	number (1)
Number of Mashable article links	number (1)
Minimum, average and maximum number of shares of Mashable links	number (3)
Digital Media	
Number of images	number (1)
Number of videos	number (1)
Time	
Day of the week	nominal (1)
Published on a weekend?	bool (1)

Feature	Type (#)
Keywords	
Number of keywords	number (1)
Worst keyword (min./avg./max. shares)	number (3)
Average keyword (min./avg./max. shares)	number (3)
Best keyword (min./avg./max. shares)	number (3)
Article category (Mashable data channel)	nominal (1)
Natural Language Processing	
Closeness to top 5 LDA topics	ratio (5)
Title subjectivity	ratio (1)
Article text subjectivity score and its absolute difference to 0.5	ratio (2)
Title sentiment polarity	ratio (1)
Rate of positive and negative words	ratio (2)
Pos. words rate among non-neutral words	ratio (1)
Neg. words rate among non-neutral words	ratio (1)
Polarity of positive words (min./avg./max.)	ratio (3)
Polarity of negative words (min./avg./max.)	ratio (3)
Article text polarity score and its absolute difference to 0.5	ratio (2)

Target	Type (#)
Number of article Mashable shares	number (1)

Code for Principal Component Analysis:

```
#Libraries
library(Hmisc) #Describe Function
library(psych) #Multiple Functions for Statistics and Multivariate Analysis
library(GGally) #ggpairs Function
library(ggplot2) #ggplot2 Functions
library(vioplot) #Violin Plot Function
library(corrplot) #Plot Correlations
library(REdaS) #Bartlett's Test of Sphericity
library(psych) #PCA/FA functions
library(factoextra) #PCA Visualizations
library("FactoMineR") #PCA functions
library(ade4) #PCA Visualizations
################################################################################

#Set Working Directory
setwd('C:/Users/Samuel/Downloads/Advaced Data Analysis/Final Project')

#Read in Datasets
RawData <- read.csv(file="OnlineNewsPopularity.csv", header=TRUE, sep=",")

#Check Sample Size and Number of Variables
dim(RawData)
#39644-Sample Size and 61 variables

#Show for first 6 rows of data
head(RawData)

#Names of the data
names(RawData)

#Check for Missing Values (i.e. NAs)
#For All Variables
sum(is.na(RawData))
#0 total missing values (0 cells with missing data)

describe(RawData)

#Create new subsets of data
Data <- RawData[,c(5:7,12,19:22,24:31,42,43,45:51,53,55,61)]

library(psych)
describe(Data)

#Create Initial Linear Regression Model with Enter Method
model1 <- lm(shares ~ ., data=Data)
model1

library(car)
#Check VIF
vif(model1)

# PCA_Plot functions
PCA_Plot = function(pcaData)
{
  library(ggplot2)

  theta = seq(0,2*pi,length.out = 100)
```

```
  circle = data.frame(x = cos(theta), y = sin(theta))
  p = ggplot(circle,aes(x,y)) + geom_path()

  loadings = data.frame(pcaData$rotation, .names = row.names(pcaData$rotation))
  p + geom_text(data=loadings, mapping=aes(x = PC1, y = PC2, label = .names, colour = .names, fontface="bold")) +
    coord_fixed(ratio=1) + labs(x = "PC1", y = "PC2")
}

PCA_Plot_Secondary = function(pcaData)
{
  library(ggplot2)

  theta = seq(0,2*pi,length.out = 100)
  circle = data.frame(x = cos(theta), y = sin(theta))
  p = ggplot(circle,aes(x,y)) + geom_path()

  loadings = data.frame(pcaData$rotation, .names = row.names(pcaData$rotation))
  p + geom_text(data=loadings, mapping=aes(x = PC3, y = PC4, label = .names, colour = .names, fontface="bold")) +
    coord_fixed(ratio=1) + labs(x = "PC3", y = "PC4")
}

PCA_Plot_Psyc = function(pcaData)
{
  library(ggplot2)

  theta = seq(0,2*pi,length.out = 100)
  circle = data.frame(x = cos(theta), y = sin(theta))
  p = ggplot(circle,aes(x,y)) + geom_path()

  loadings = as.data.frame(unclass(pcaData$loadings))
  s = rep(0, ncol(loadings))
  for (i in 1:ncol(loadings))
  {
    s[i] = 0
    for (j in 1:nrow(loadings))
      s[i] = s[i] + loadings[j, i]^2
    s[i] = sqrt(s[i])
  }

  for (i in 1:ncol(loadings))
    loadings[, i] = loadings[, i] / s[i]

  loadings$.names = row.names(loadings)

  p + geom_text(data=loadings, mapping=aes(x = PC1, y = PC2, label = .names, colour = .names, fontface="bold")) +
    coord_fixed(ratio=1) + labs(x = "PC1", y = "PC2")
}

PCA_Plot_Psyc_Secondary = function(pcaData)
{
  library(ggplot2)

  theta = seq(0,2*pi,length.out = 100)
  circle = data.frame(x = cos(theta), y = sin(theta))
  p = ggplot(circle,aes(x,y)) + geom_path()
```

```
  loadings = as.data.frame(unclass(pcaData$loadings))
  s = rep(0, ncol(loadings))
  for (i in 1:ncol(loadings))
  {
    s[i] = 0
    for (j in 1:nrow(loadings))
      s[i] = s[i] + loadings[j, i]^2
    s[i] = sqrt(s[i])
  }

  for (i in 1:ncol(loadings))
    loadings[, i] = loadings[, i] / s[i]

  loadings$.names = row.names(loadings)

  print(loadings)
  p + geom_text(data=loadings, mapping=aes(x = PC3, y = PC4, label = .names, colour = .names, fontface="bold")) +
    coord_fixed(ratio=1) + labs(x = "PC3", y = "PC4")
}

#Test KMO Sampling Adequacy
library(psych)
KMO(Data)
#Overall MSA =  0.6
#> 0.7 so good size

#Test Bartletts Test of Sphericity
library(REdaS)
bart_spher(Data)
#p-value < 2.22e-16 (Very Small Number)

#Test for Reliability Analysis using Cronbachs Alpha
library(psych)
alpha(Data,check.keys=TRUE)
#raw_alpha = 0.34

#scaling
scale(Data, center = TRUE, scale = TRUE)

#Create PCA
p = prcomp(Data, center=T, scale=T)
p

#Check Scree Plot
plot(p)
abline(1, 0)

#Check PCA Summary Information
summary(p)
print(p)

#Check PCA visualizations
plot(p) #Scree Plot
PCA_Plot(p) #PCA_plot1
PCA_Plot_Secondary(p) #PCA_Plot2
```

```r
print(p)

#Check PCA visualizations
plot(p) #Scree Plot
PCA_Plot(p) #PCA_plot1
PCA_Plot_Secondary(p) #PCA_Plot2
#biplot(p) #Biplot

#Calculating the Varimax Rotation Loadings manually
rawLoadings = p$rotation %*% diag(p$sdev, nrow(p$rotation), nrow(p$rotation))
print(rawLoadings)
v = varimax(rawLoadings)

#Options available under varimax function
ls(v)
v

# The Psych package has a wonderful PCA function that allows many more options
# including build-in factor rotation, specifiying a number of factors to include
# and automatic "score" generation

#Best Way to Conduct PCA Analysis
p2 = psych::principal(Data, rotate="varimax", covar=FALSE, nfactors=8, scores=TRUE)
p2
print(p2$loadings, cutoff=.4, sort=T)

#PCAs Other Available Information
ls(p2)
p2$values
table(p2$values>1)
p2$communality
p2$rot.mat

#Calculating scores
scores <- p2$scores
scores_1 <- scores[,1]
round(cor(scores),2)
cor(scores)
summary(scores)
summary(scores_1)
scores_2 <- scores[,2]
summary(scores_2)
scores_3 <- scores[,3]
summary(scores_3)
scores_4 <- scores[,4]
summary(scores_4)
scores_5 <- scores[,5]
summary(scores_5)
scores_6 <- scores[,6]
summary(scores_6)
scores_7 <- scores[,7]
summary(scores_7)
scores_8 <- scores[,8]
summary(scores_8)
#scores_9 <- scores[,9]

```

```
> vif(model1)
           n_unique_tokens            n_non_stop_words   n_non_stop_unique_tokens       average_token_length
                9098.959602                1834.601182                6664.746542                  11.037340
     data_channel_is_world                  kw_min_min                  kw_max_min                  kw_avg_min
                  3.469243                    3.815688                   10.963693                  10.636205
                kw_max_max                  kw_avg_max                  kw_min_avg                  kw_max_avg
                  4.317188                    2.214699                    1.943231                   6.598788
                kw_avg_avg  self_reference_min_shares  self_reference_max_shares self_reference_avg_sharess
                  9.186415                    6.523240                    7.945276                  18.507755
                    LDA_02                      LDA_03       global_subjectivity  global_sentiment_polarity
                  3.578020                    1.774273                    2.675626                   6.622439
global_rate_positive_words global_rate_negative_words       rate_positive_words       rate_negative_words
                  4.469316                    6.185731                   19.799463                  17.439766
     avg_positive_polarity      max_positive_polarity      min_negative_polarity
                  4.473633                    3.091487                    2.174302
```

4a Vif of PCA

```
> p2$values
 [1] 4.243797e+00 3.459755e+00 3.089399e+00 2.998675e+00 2.442622e+00 2.270547e+00 1.970207e+00 1.039764e+00 9.752135e-01
[10] 8.289227e-01 7.524125e-01 6.515765e-01 5.845887e-01 5.680623e-01 4.915693e-01 3.915534e-01 3.581188e-01 2.209365e-01
[19] 1.602651e-01 1.381901e-01 1.049818e-01 8.042400e-02 7.491527e-02 4.184274e-02 3.265389e-02 2.852312e-02 4.171938e-04
[28] 6.574365e-05
> table(p2$values>1)

FALSE  TRUE
   20     8
```

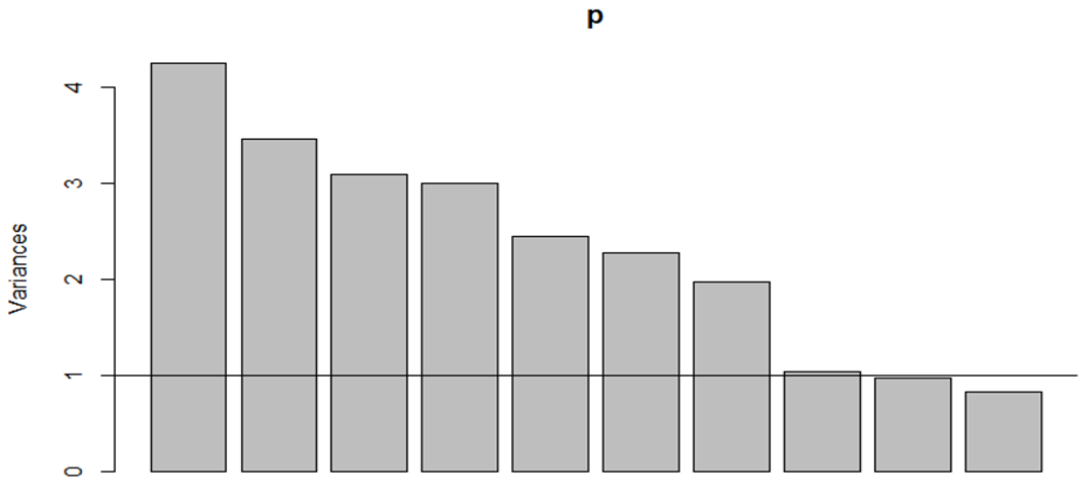
4c

```
> #Check PCA Summary Information
> summary(p)
Importance of components:
                          PC1    PC2    PC3    PC4     PC5     PC6     PC7     PC8     PC9   PC10    PC11    PC12    PC13    PC14
Standard deviation     2.0600 1.8600 1.7577 1.7317 1.56289 1.50683 1.40364 1.01969 0.98753 0.9105 0.86742 0.80720 0.76458 0.75370
Proportion of Variance 0.1516 0.1236 0.1103 0.1071 0.08724 0.08109 0.07036 0.03713 0.03483 0.0296 0.02687 0.02327 0.02088 0.02029
Cumulative Proportion  0.1516 0.2751 0.3855 0.4926 0.57979 0.66089 0.73125 0.76838 0.80321 0.8328 0.85969 0.88296 0.90384 0.92413
                          PC15    PC16    PC17    PC18    PC19    PC20    PC21    PC22    PC23    PC24    PC25    PC26    PC27
Standard deviation     0.70112 0.62574 0.59843 0.47004 0.40033 0.37174 0.32401 0.28359 0.27371 0.20455 0.18070 0.16889 0.02043
Proportion of Variance 0.01756 0.01398 0.01279 0.00789 0.00572 0.00494 0.00375 0.00287 0.00268 0.00149 0.00117 0.00102 0.00001
Cumulative Proportion  0.94168 0.95567 0.96846 0.97635 0.98207 0.98701 0.99076 0.99363 0.99630 0.99780 0.99896 0.99998 1.00000
                          PC28
Standard deviation     0.008108
Proportion of Variance 0.000000
Cumulative Proportion  1.000000
```

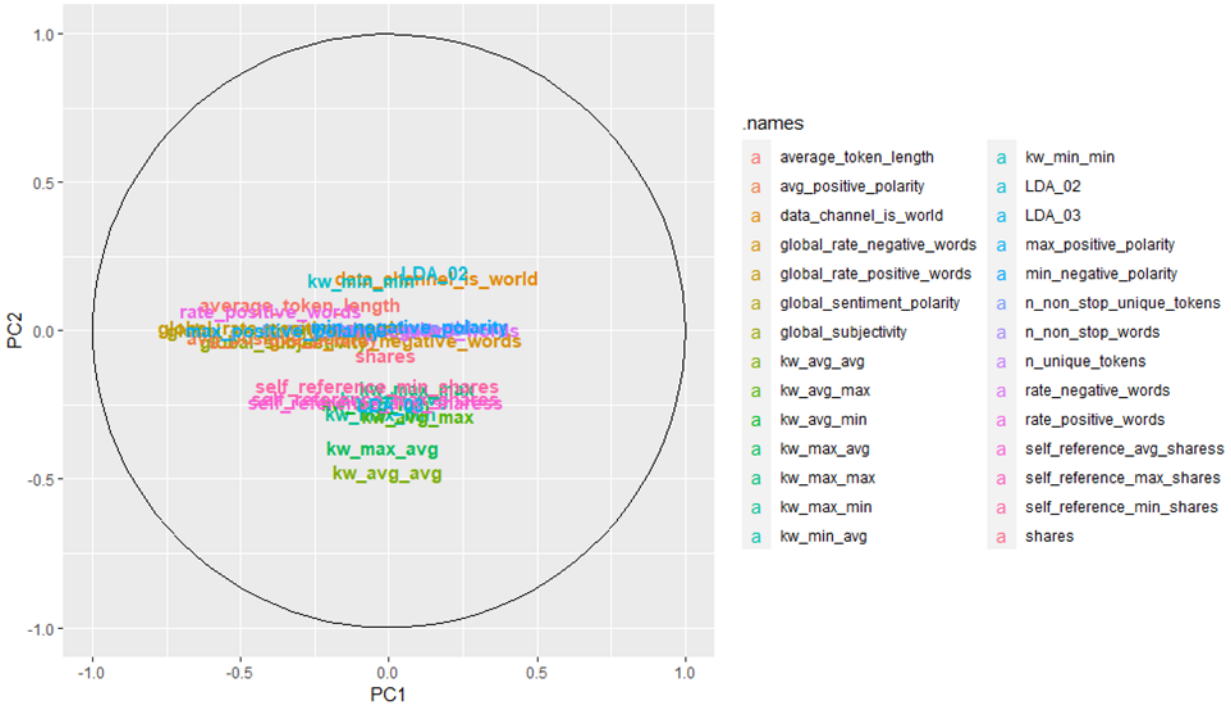
4d

```
> print(p2$loadings, cutoff=.4, sort=T)

Loadings:
                          RC1    RC3    RC4    RC2    RC6    RC5    RC7    RC8
average_token_length       0.753
global_subjectivity        0.804
global_rate_positive_words 0.670
rate_positive_words        0.732 -0.560
avg_positive_polarity      0.809
max_positive_polarity      0.804
global_sentiment_polarity  0.589 -0.700
global_rate_negative_words        0.873
rate_negative_words               0.931
min_negative_polarity            -0.641
n_unique_tokens                          1.000
n_non_stop_words                         1.000
n_non_stop_unique_tokens                 1.000
kw_max_min                                      0.940
kw_avg_min                                      0.926
kw_max_avg                                      0.796
self_reference_min_shares                              0.848
self_reference_max_shares                              0.858
self_reference_avg_sharess                             0.986
kw_min_min                                                    -0.928
kw_max_max                                                     0.942
kw_avg_max                                                     0.687         0.444
data_channel_is_world                                                 0.916
LDA_02                                                                0.922
kw_min_avg                                                                   0.678
kw_avg_avg                                             0.599                 0.614
LDA_03                                                                       0.595
shares                                                                       0.404

                 RC1   RC3   RC4   RC2   RC6   RC5   RC7   RC8
SS loadings    4.029 3.028 3.000 2.747 2.469 2.395 2.109 1.737
Proportion Var 0.144 0.108 0.107 0.098 0.088 0.086 0.075 0.062
Cumulative Var 0.144 0.252 0.359 0.457 0.545 0.631 0.706 0.768
>
```

4f


p
Variances
0
1
2
3
4

4b


PC2
PC1
1.0
0.5
0.0
-0.5
-1.0
kw_min_min
LDA_02
data_channel_is_world
average_token_length
rate_positive_words
shares
self_reference_min_shares
kw_avg_max
kw_max_avg
kw_avg_avg
.names
average_token_length
avg_positive_polarity
data_channel_is_world
global_rate_negative_words
global_rate_positive_words
global_sentiment_polarity
global_subjectivity
kw_avg_avg
kw_avg_max
kw_avg_min
kw_max_avg
kw_max_max
kw_max_min
kw_min_avg
kw_min_min
LDA_02
LDA_03
max_positive_polarity
min_negative_polarity
n_non_stop_unique_tokens
n_non_stop_words
n_unique_tokens
rate_negative_words
rate_positive_words
self_reference_avg_sharess
self_reference_max_shares
self_reference_min_shares
shares

4e

Code for CFA

```
#######################
#Final Project CFA
#######################

library(DescTools) #VIF Function
library(REdaS)
library(psych)
library(FactoMineR)
library(ade4)

#import data
news_pop <- (OnlineNewsPopularity)
summary(news_pop)


#######################
#Data Preprocessing
#######################

#checking for correlation
model1 <- lm(shares ~ data_channel_is_bus, news_pop$lda_00,data_channel_is_tech  , data=news_pop)
model1
pairs(news_pop2)
VIF(model1)
cor(model1)
cor(news_pop$data_channel_is_bus , news_pop$lda_00)
pairs( ~news_pop$num_hrefs + news_pop$num_self_hrefs + news_pop$num_imgs + news_pop$num_videos)

#combining correlated variables into one variable
news_pop$sum_unique <- (news_pop$n_unique_tokens + news_pop$n_non_stop_words + news_pop$n_non_stop_unique_tokens)

#removing categorical or highly correlated variables
news_pop2 <- (news_pop[,-c(1,2,5,6,7,20,21,23,24,26,27,29,30,32:38,52,53,55,56,61)])
summary(news_pop)

#Test KMO Sampling Adequacy
KMO(news_pop)
#Overall MSA =  0.5

#Test Bartlett's Test of Sphericity
bart_spher(news_pop)
#p-value < 2.22e-16


#Test for Reliability Analysis using Cronbach's Alpha
alpha(news_pop,check.keys=TRUE)
#raw_alpha = 0.37
```

```
#######################
#Created to see Scree Plot
#######################

p = prcomp(news_pop2, center=T, scale=T)
p

#Check Scree Plot
plot(p)
abline(1, 0)

p2 = psych::principal(news_pop2, rotate="promax", nfactors=5, scores=TRUE)
p2
print(p2$loadings, cutoff=.4, sort=T)

#######################
#Conducting Factor Analysis
#######################

fit = factanal(news_pop2, factor = 6 , rotation = "promax")
print(fit$loadings, cutoff=.1, sort=T)
summary(fit)

#removing unused
news_pop3 <- (news_pop[,-c(3,31,39,58,1,2,4:11,13:14,17,19:27,29,30,32:38,51:55,56:57,59,61,60)])
fit3 = factanal(news_pop3, factor = 6,rotation = "promax")
print(fit3$loadings, cutoff=.4, sort=T)
```

Appendix for Correspondence Analysis

```
> chisq.test(new_data)

        Pearson's Chi-squared test

data:  new_data
X-squared = 348.48, df = 30, p-value < 2.2e-16
```

Fig.1.1 Chisquared test for channel and weekday

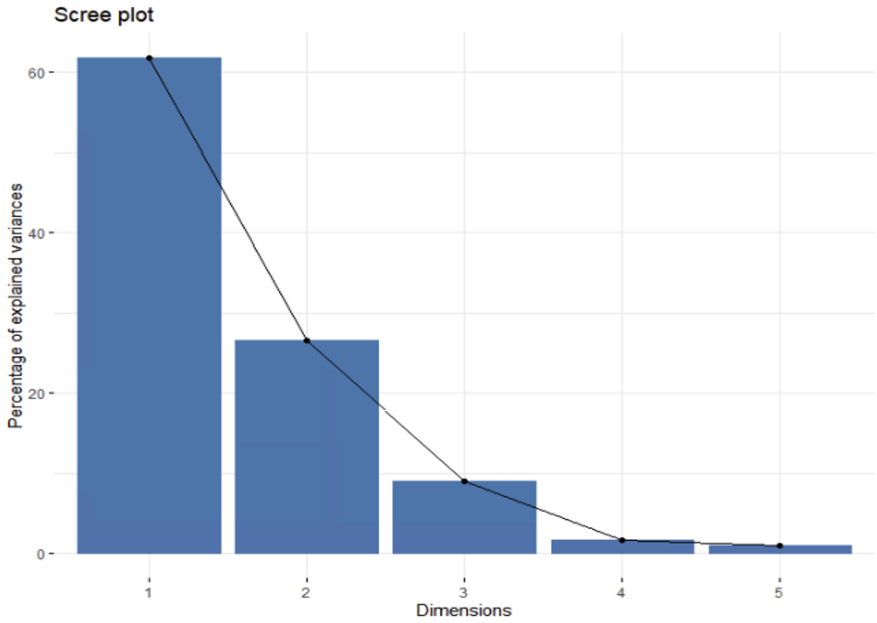


Fig. 1.2 Scree plot of correspondence analysis

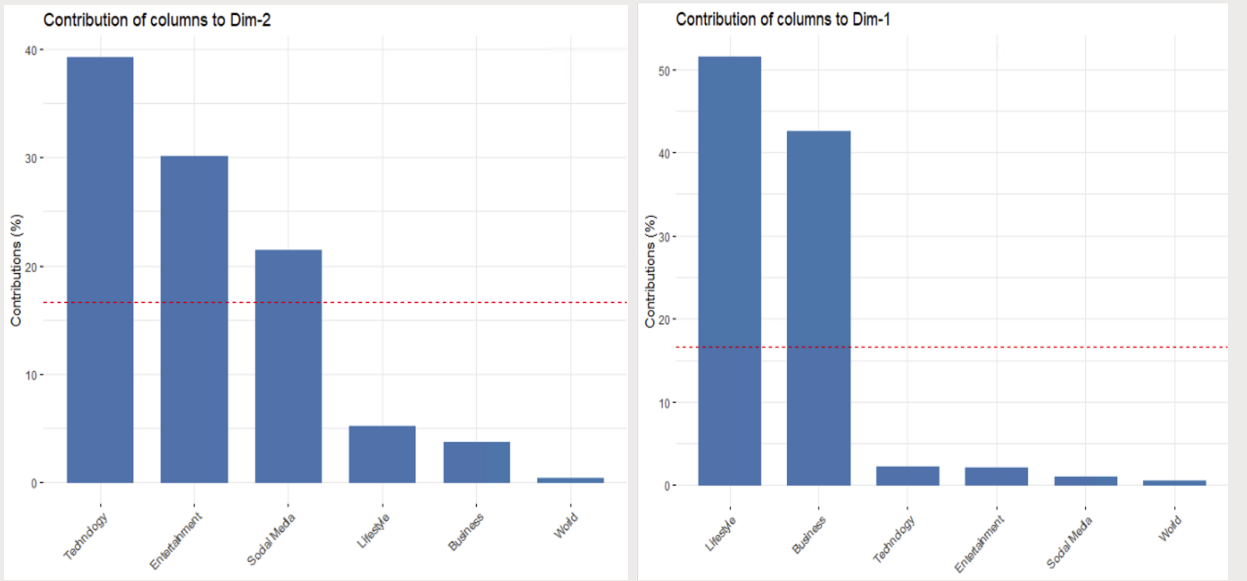


Fig 1.3 Contribution of rows

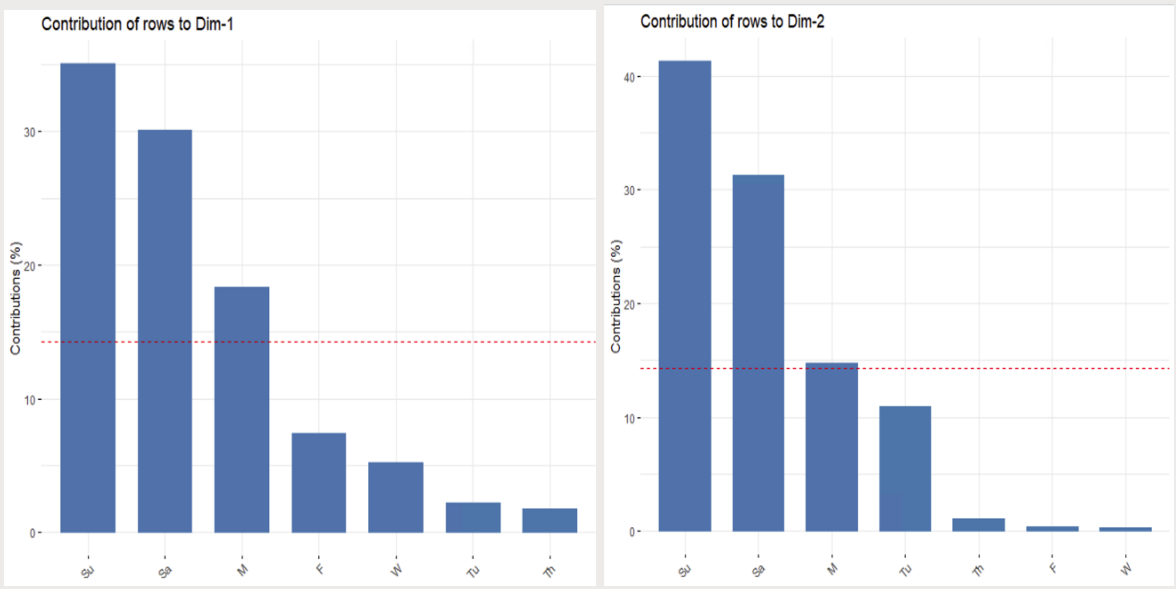


Fig 1.4: Contribution of columns

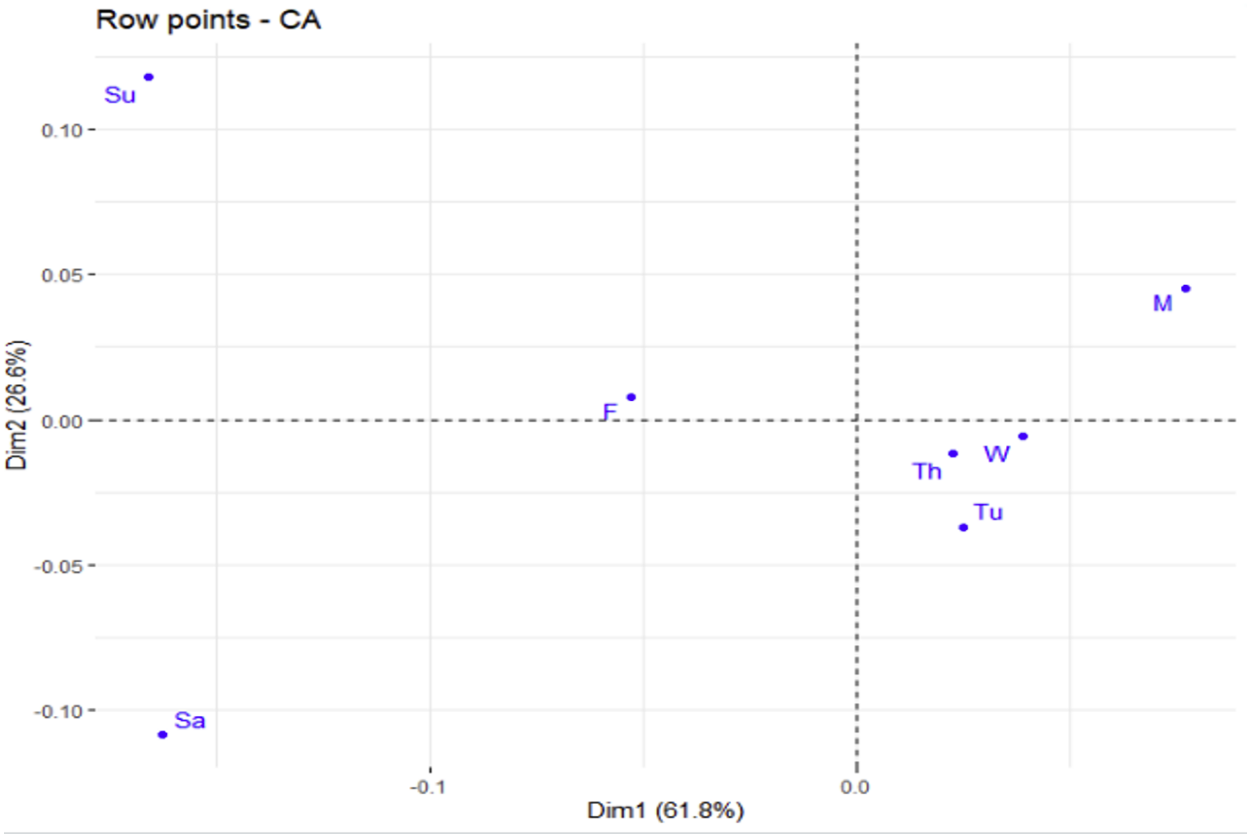


Fig 1.5: Correlation of rows with both dimensions

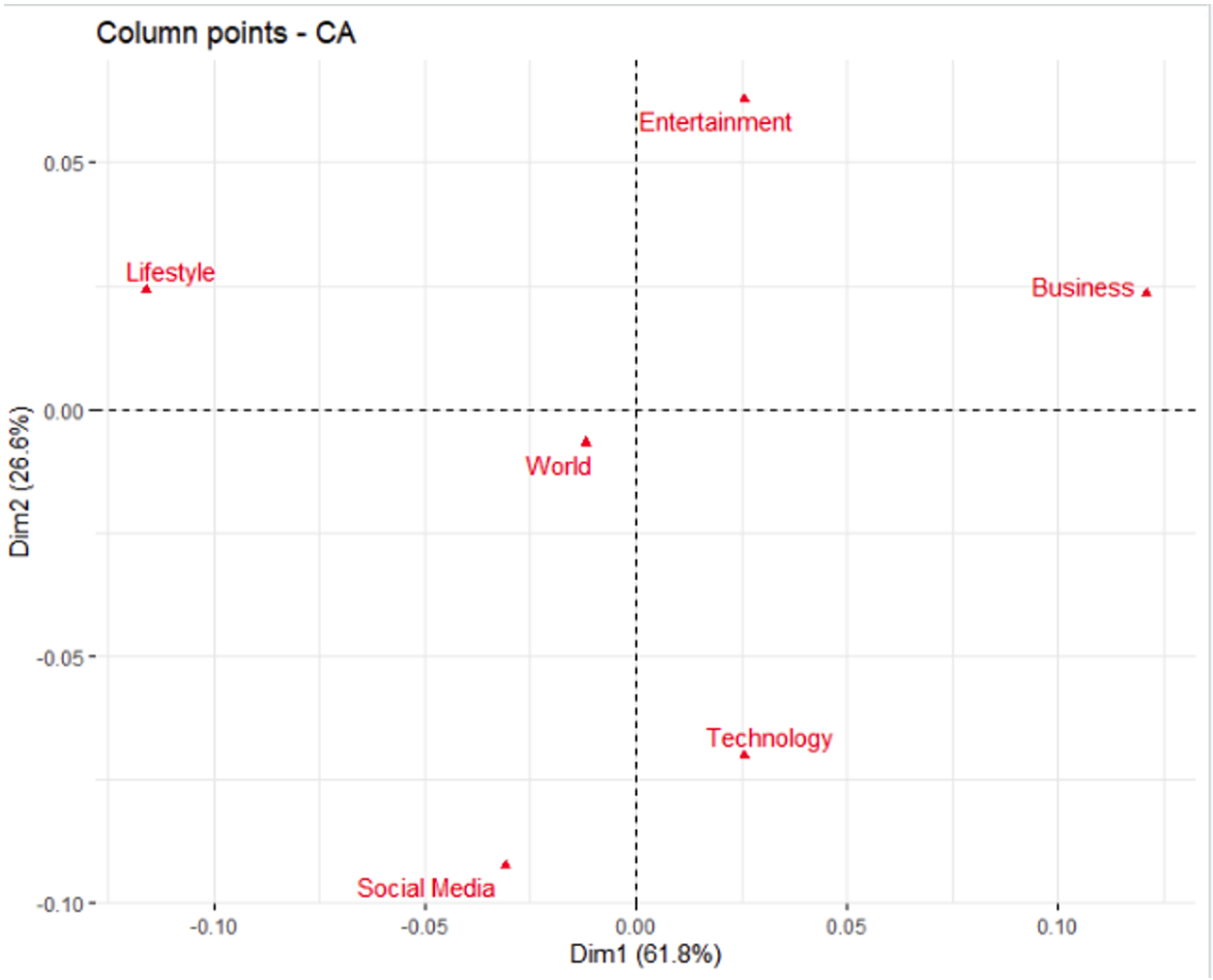


Fig1.6: Correlation of columns with both dimensions

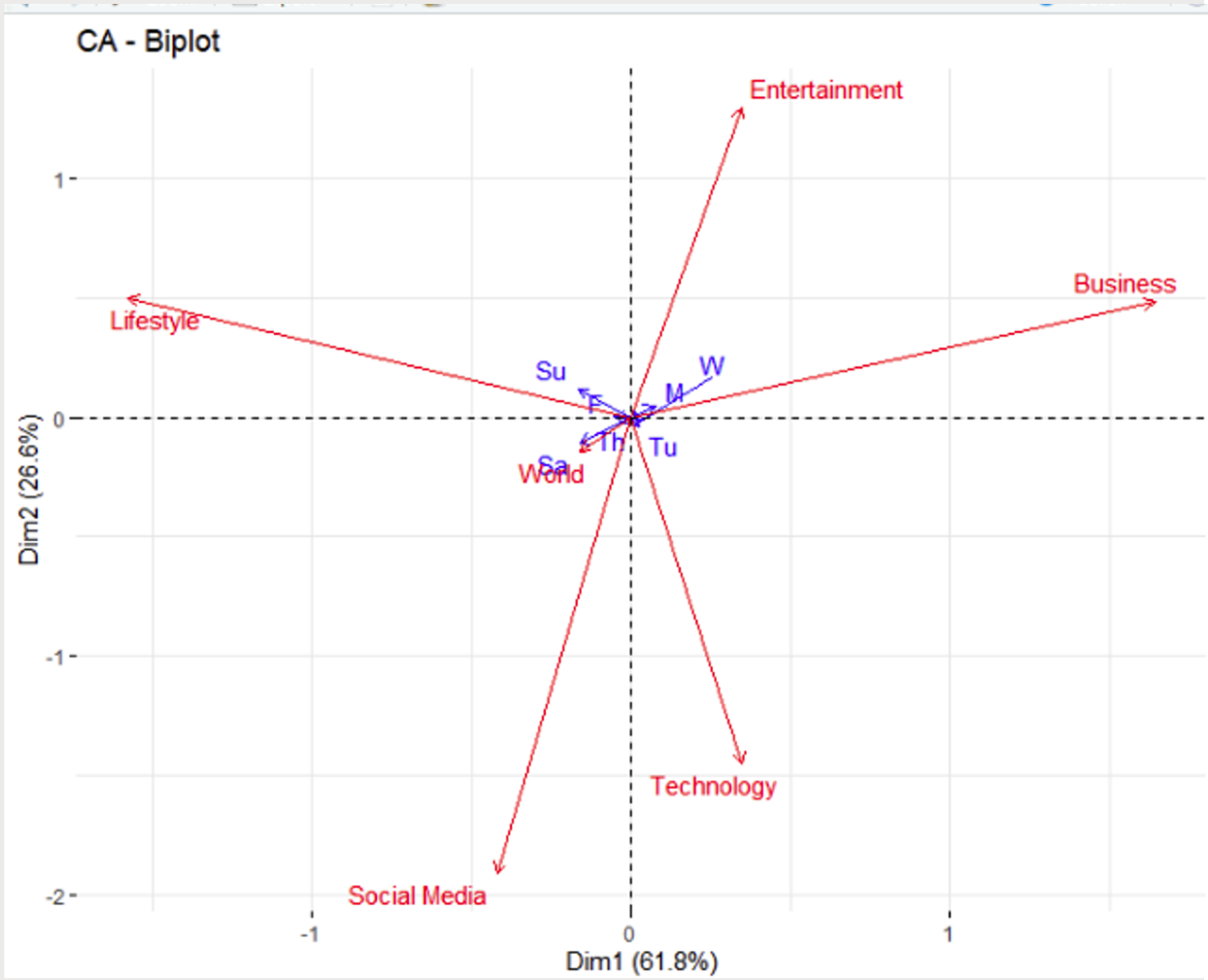


Fig:1.7 Biplot of Weekday and Channels

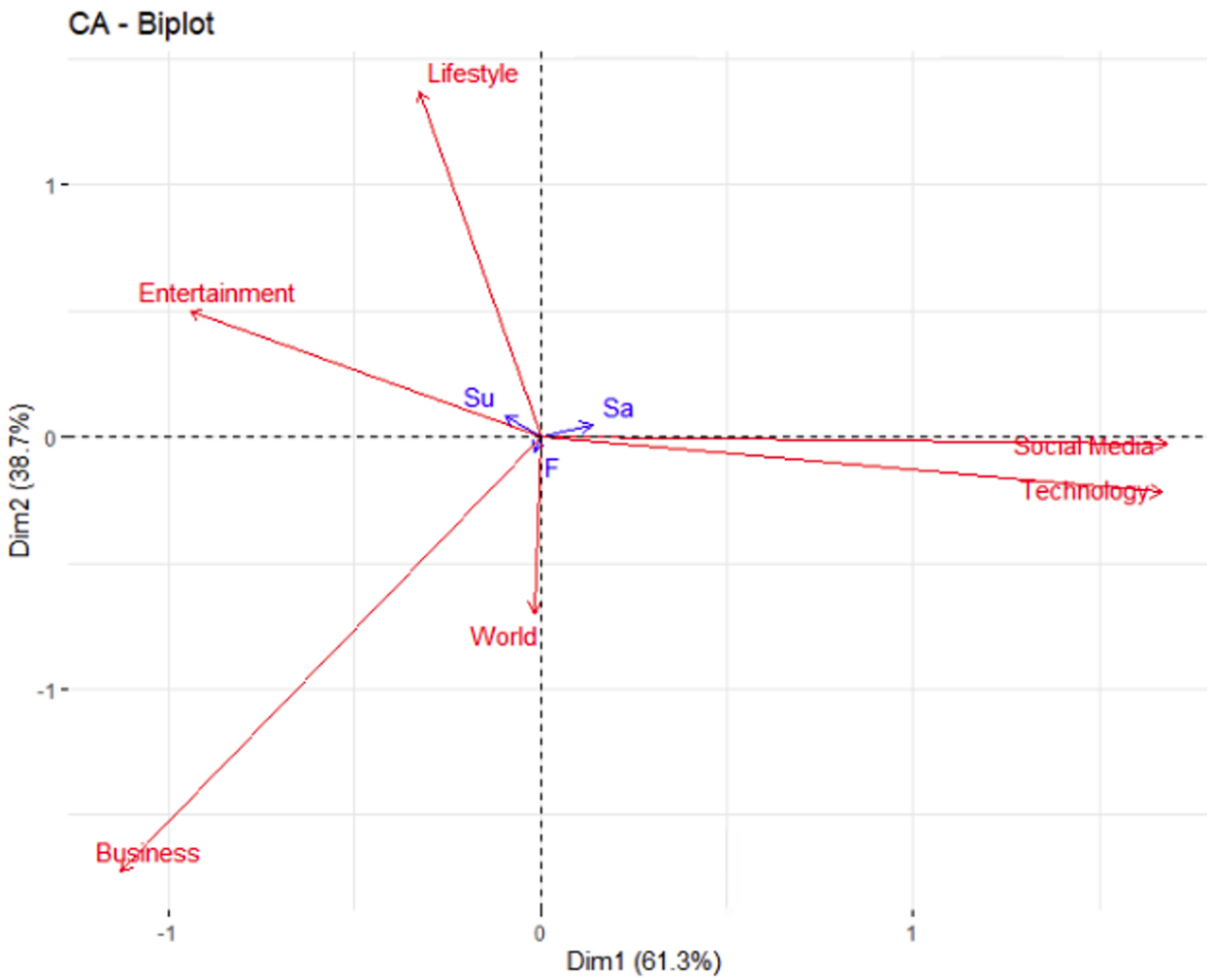


Fig 1.8: Biplot of Weekday and Channels for weekend

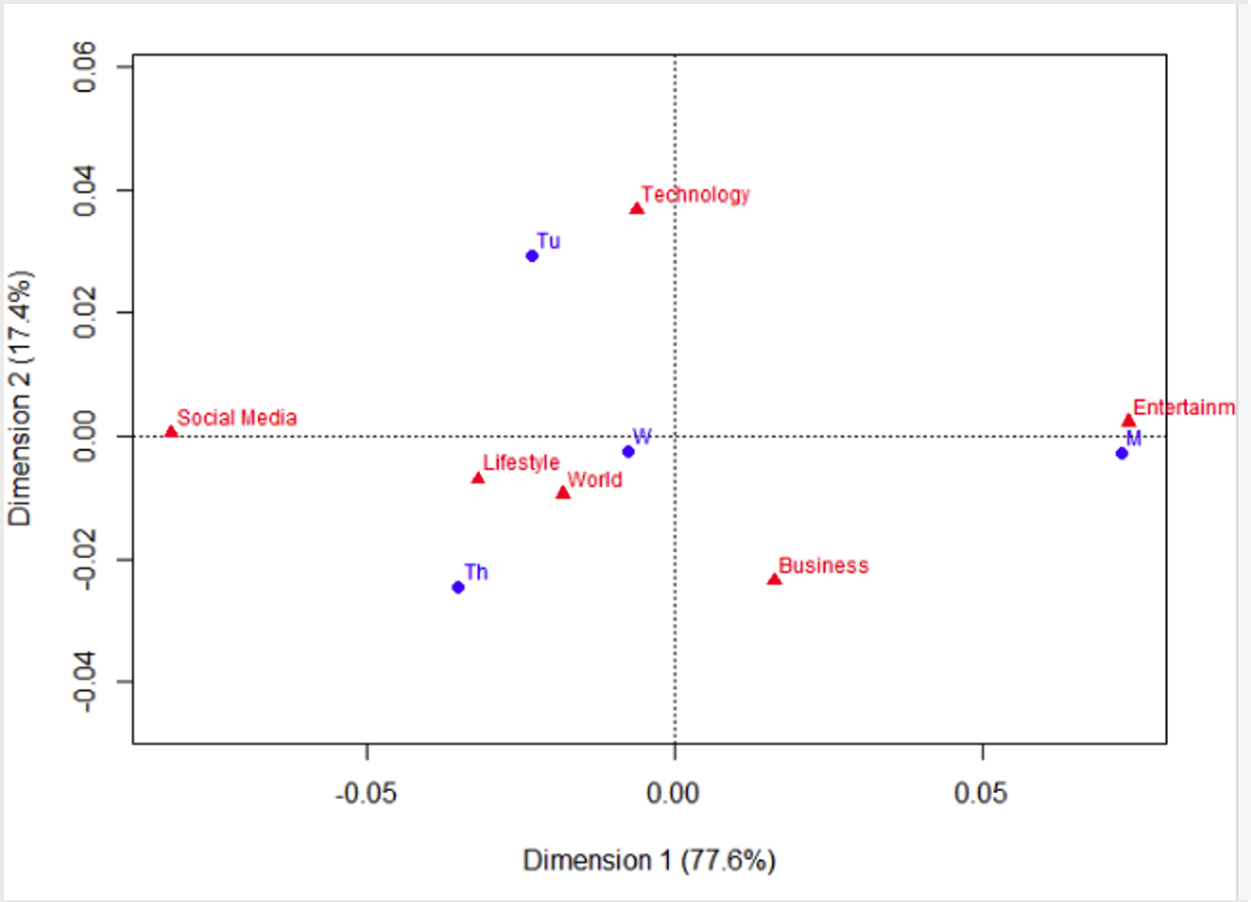


Fig 1.9: Correspondence plot of channel and weekday for Monday through Thursday

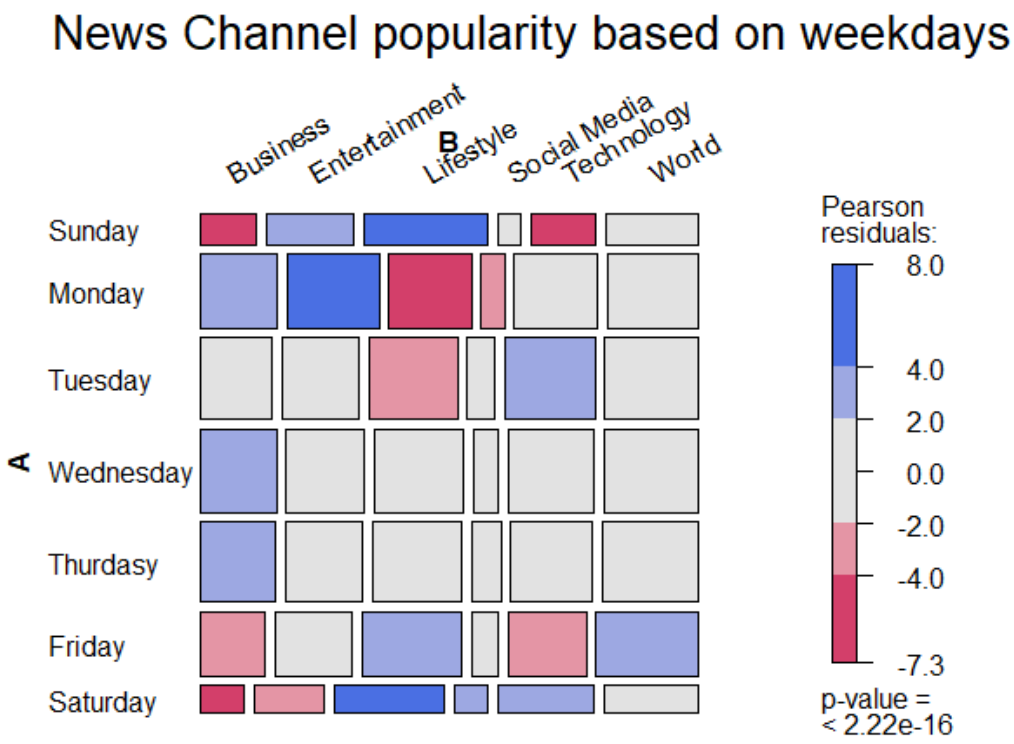


Fig 1.10: Mosaic plot of News channel and weekday
Appendix:

Python code for correspondence Analysis:
Converted 7 dummy categorical variables into a categorical variable and then created a contingency table for it.

```
import pandas as pd
import numpy as np
data = pd.read_csv("/content/OnlineNewsPopularity.csv")
channel = data.iloc[:, 13:19]
weekday = data.iloc[:,31:38]
#combine dummy variables into single categorical variable.
channel = channel.idxmax(axis =1)
weekday = data.iloc[:, 31:38].idxmax(axis = 1)

new_data = pd.concat([data, weekday, channel], axis =1 )
new_data = pd.DataFrame(new_data)
df = new_data.iloc[:, 61:63]
df.rename(columns = {0: "weekday", 1: "channel"}, inplace = True)
```

```
corres_data = pd.crosstab(index=df['weekday'], columns=df['channel'],
margins=True)
```

R code for Correspondence Analysis:

```
library(vcd)
library(ca)
library(tibble)
library(factoextra)
library(FactoMineR)
################################################

data = corres_data1
new_data = data[1:7, 2:7]
chisq.test(new_data)
data_matrix = column_to_rownames(data, var = "weekday")
new_data = data.matrix(data_matrix)

mosaic(new_data, shade=TRUE, legend=TRUE)

fit = ca(new_data)
plot(fit)
plot(fit, mass=T, contrib="absolute",
    map="rowgreen", arrows=c(T, T))

summary(fit)

#Inspect Eigenvalues
fviz_eig(fit)


#Inspect correlations
fviz_ca_row(fit, repel = TRUE)
fviz_ca_col(fit, repel = TRUE)

#Contribution of rows
fviz_contrib(fit, choice = 'row', top = 10)
fviz_contrib(fit, choice = 'row', top = 10, axes = 2)

#Contribution of columns
fviz_contrib(fit, choice = 'col', top = 10)
```

```
fviz_contrib(fit, choice = 'col', top = 10, axes = 2)

#Biplots
fviz_ca_biplot(fit, repel = TRUE)
fviz_ca_biplot(fit, map = "rowprincipal", arrow = c(T,T), repel = TRUE)



weekend = new_data[c(1,6,7),]
chisq.test(weekend)
fit_weekend = ca(weekend)
summary(fit_weekend)

fviz_ca_biplot(fit_weekend, map = "rowprincipal", arrow = c(T,T), repel = TRUE)


weekday = new_data[c(2:5),]
chisq.test(weekday)
fit_weekday = ca(weekday)
summary(fit_weekday)
fviz_ca_biplot(fit_weekday, map = "rowprincipal", arrow = c(T,T), repel = TRUE)
```

Code for Lasso regression:

```
data = read.csv("C:\\Users\\Arun sivakumar\\Desktop\\OnlineNewsPopularity.csv", header=T)

#Examine correlation

cor(data)

#Removing non-predictive features: url and timedelta

 data = select(data, -1,-2)

#plot outliers

 abline(h = 4/sample_size, col="red")

text(x=1:length(cooksd)+1, y=cooksd, labels=ifelse(cooksd>4/sample_size,

                                        names(cooksd),""), col="red")

influential <- as.numeric(names(cooksd)[(cooksd > (4/sample_size))])
```

```
#first model
M1 <- lm(shares ~., data=data)
summary(M1)
#examine VIF
vif(M1)
#data prep for lasso regression
y <- data$shares
x <- data.matrix(data[, c(
   "n_tokens_title",
 "n_tokens_content"      ,       "n_unique_tokens"      ,        "n_non_stop_words"
 "n_non_stop_unique_tokens"  ,   "num_hrefs"          ,          "num_self_hrefs"
"num_imgs"           ,          "num_videos"             ,      "average_token_length"
"num_keywords"      ,           "data_channel_is_lifestyle" ,   "data_channel_is_entertainment"
"data_channel_is_bus"   ,       "data_channel_is_socmed"  ,     "data_channel_is_tech"
"data_channel_is_world"  ,      "kw_min_min"           ,        "kw_max_min"
"kw_avg_min"       ,            "kw_min_max"        ,           "kw_max_max"
 "kw_avg_max"        ,           "kw_min_avg"         ,          "kw_max_avg"
 "kw_avg_avg"          ,         "self_reference_min_shares" ,   "self_reference_max_shares"
"self_reference_avg_sharess"  , "weekday_is_monday"      ,      "weekday_is_tuesday"
 "weekday_is_wednesday"     ,    "weekday_is_thursday"   ,       "weekday_is_friday"
"weekday_is_saturday"       ,   "weekday_is_sunday"      ,     "is_weekend"
"LDA_00"          ,             "LDA_01"             ,         "LDA_02"
"LDA_03"             ,          "LDA_04"             ,         "global_subjectivity"
```

```
"global_sentiment_polarity"   ,  "global_rate_positive_words"  ,  "global_rate_negative_words"
"rate_positive_words","rate_negative_words"  ,"avg_positive_polarity"

"min_positive_polarity" ,"max_positive_polarity"     , "avg_negative_polarity"

"min_negative_polarity"  ,"max_negative_polarity" ,"title_subjectivity"

"title_sentiment_polarity"  ,"abs_title_subjectivity" , "abs_title_sentiment_polarity"

  )])

#lasso regression model

cv_model <- cv.glmnet(x, y, alpha = 1)

best_lambda <- cv_model$lambda.min

best_lambda

plot(cv_model)

best_model <- glmnet(x, y, alpha = 1, lambda = best_lambda)

coef(best_model)
```

Viz:

```
data_bin <- data %>% mutate(images_bin = cut(num_imgs, breaks=c(0, 20, 40, 60,
80,100,110)))

boxplot(data_bin$shares~data_bin$num_imgs,

data=data_bin,main="Shares vs No. of images",ylab="shares",
xlab="no. of images",col="orange", border="brown")


global_data <- data %>% mutate(global_bin = cut(global_subjectivity, breaks=c(0, 0.2, 0.4, 0.6,
0.8,1.0)))

boxplot(global_data$shares~global_data$global_bin,
       data=global_data,
       main="Shares vs global_subjectivity",
       ylab="shares",
xlab="global_subjectivity",
       col="orange",
       border="brown"
)
```

```
#R code for Bckward Regression and Multiple Linear Regression:
data = read.csv("/Users/aswin/OnlineNewsPopularity.csv",header=TRUE)
df = subset(data, select = -c(url,data_channel_is_tech,weekday_is_sunday))
df
df

#checking for null values
sum(is.na(df))
#Descriptive statistics
summary(df)
boxplot(df$shares,
        ylab = "hwy",
        main = "Boxplot of highway miles per gallon"
)
#scaling the data
scale_data <- as.data.frame(scale(df))
scale_data
#backward selection
full = lm(shares ~ ., data=scale_data)
full
train_Backward = step(full, direction="backward")
summary(train_Backward)
train_Backward$anova
#multiple linear regression
model1 <- lm(shares ~ num_keywords + kw_min_min + num_imgs + abs_title_subjectivity
+kw_avg_min + self_reference_max_shares+ data_channel_is_lifestyle +
abs_title_sentiment_polarity+ n_non_stop_words+ n_unique_tokens+
kw_min_max+min_positive_polarity+ weekday_is_monday+ average_token_length+
global_subjectivity+ num_self_hrefs+n_tokens_title+
num_hrefs+timedelta+data_channel_is_entertainment+kw_min_avg+
self_reference_min_shares+kw_max_avg+kw_avg_avg,data=scale_data)
summary(model1)
VIF(model1)
#multiple linear regression after checking for vif values
model2 <- lm(shares ~ num_keywords  + num_imgs + abs_title_subjectivity  +
self_reference_max_shares+ data_channel_is_lifestyle + abs_title_sentiment_polarity+
n_unique_tokens+ kw_min_max+ weekday_is_monday+ average_token_length+
global_subjectivity+ num_self_hrefs+n_tokens_title+
num_hrefs+data_channel_is_entertainment+kw_min_avg+
self_reference_min_shares+kw_max_avg,data=scale_data)
summary(model2)
VIF(model2)
c<-cor(scale_data, method="spearman")
c
```