Prediction of the popularity of an article

Elie G. AIDASSO Ulrich AIOUNOU Saleem B. DJIMA

January 2022

Abstract

The popularity of an article is a very important indicator for several actors (web editors, web influencers, bloggers, publishers etc.). It allows them to evaluate the reach and the impact of their contents on their audiences. The objective of this study is to propose an efficient model for predicting the popularity of an article among linear regression models, Lasso model, Ridge model, principal component analysis regression and some others machine learning models like General Additive Model(GAM) and Random Forest. After our estimations, We keep two models that have the same performance which are the Random Forest and the GAM. However, the GAM model has the advantage of being much faster than the Random Forest.

Keywords: article, popularity, blog, ridge, lasso, linear regression.

Contents

1 Introduction

With the expansion of internet, blogs and articles have become the most useful means of information. Whether it is to talk about oneself, a trending topic or any other specific issue, these articles have become the daily life of thousands of people. Moreover, the monetization of content on internet has led to the democratization of the writing profession on internet. In order to offer content that is increasingly appreciated by readers, bloggers (web writers) must respect certain characteristics. Indeed, an article includes many aspects or characteristics that allow the proposed content to be appreciated by readers. These aspects are not known by bloggers. In addition, they require a careful analysis of many characteristics of an article: the number of words, the number of words in the title of the article, the number of images, the number of pages, the number of times certain words appear, etc.

In particular, the popularity of a content (article) is the most important indicator for a web editor. It allows him to evaluate the reach and impact of his content on his audience. This analysis is important because it is one of the most important steps in a content strategy, better known as "Content Marketing Strategy". According to [4], the key indicators to analyze to know the popularity of your article are: the number of shares on social networks, the number of comments under the article, the tone of the opinions, the number of downloads, the bounce rate, the average time of a visit, etc. It becomes essential to know the common points between the most popular articles in particular: the format, the number of titles, the subject matter, the tone, and the different characteristics mentioned above.

In this sense, based on its different characteristics, it is possible to know a priori the popularity of an article. According to [1], predicting the popularity of an article on internet has become a new trend as it is important for content creators and even for activists and politicians who strive to understand or influence public opinion. This prediction step is important in making decisions before publishing content and is part of the Decision Support System. The progress in the field of artificial intelligence and in particular machine learning allows today to propose very efficient algorithms to make predictions.

In this paper, we propose different methods to predict the popularity of an article published on the Mashable website. The objective of our work is to see which method can make better predictions. To do so, we will choose between: a linear regression model, a Principal Component Analysis (PCA) regression, Ridge regression, Lasso regression , GAM regression and random

forest.

2 Data

This section presents our database and the associated descriptive statistics.

2.1 Source and description of data

The database of our study comes from the OpenML website. It summarizes a set of characteristics on a number of articles from the mashable website. These characteristics are among others the number of words in the title, the number of words in the article, the percentage of appearance of a word, the number of images, the number of videos, etc. The database contains 39644 rows and 61 variables of which 47 are quantitative and 14 are qualitative. The table below summarizes the descriptive statistics of some variables in our database. But for our study, we exclude articles that are very recent (less than 3 weeks), so our final database contains 39 016 observations.

Table 1: Summary statistics

Statistic	N	Mean	St. Dev.	Min	Max
Number of shares target	39,016	3,406.018	11,706.110	1	843,300
Number of words in the title	39,016	10.384	2.107	2	20
Number of words in the content	39,016	546.998	471.967	0	8,474
Number of links	39,016	10.892	11.246	0	187
Number of images	39,016	4.567	8.352	0	128
Number of videos	39,016	1.259	4.128	0	91

From the table 1, we can see that the maximum sharing of an article is 843,300 while the minimum sharing is 1. Regarding the average sharing of all articles, it is 3,406,018. Regarding the number of words in the title of the articles, we notice that the longest title is composed of 20 words while the shortest title includes 1 word. The average number of words in the title of all articles is 10,384. Regarding the number of words contained by the articles, it appears from this table that the largest number of words contained by an article is 8,474 while the smallest number of words contained by an article

is 0 and the average number of words contained by all articles is 546,998. Talking about the number of links, the number of images and the number of videos contained by an article, their maximums are respectively 187, 128 and 91 where their minimums are 0.

2.2 Distribution of the number of shares

The histogram 1 show us how the log of number of shares is distributed and the associative density function curve.

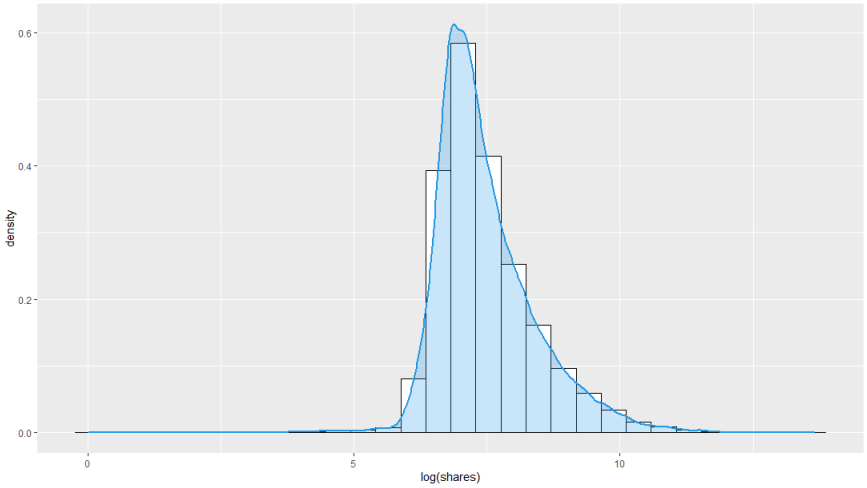

Figure 1: Histogram of number of shares

We notice that the number of shares seems to have a positive skew, meaning that the distribution is skewed to the right. The distribution is unimodal.

2.3 Description of the different types of articles published

The bar charts on 2 show the distribution of the proportion (%) of different types of articles that are published on the Mashable website.

The analysis of these different graphs, allows to highlight that the category of article the most published on the Mashable website is "World" which represents 21%. Then follow the "Business", "Technology" and "Leisure" articles, whose percentages of publications are respectively 16%, 19% and

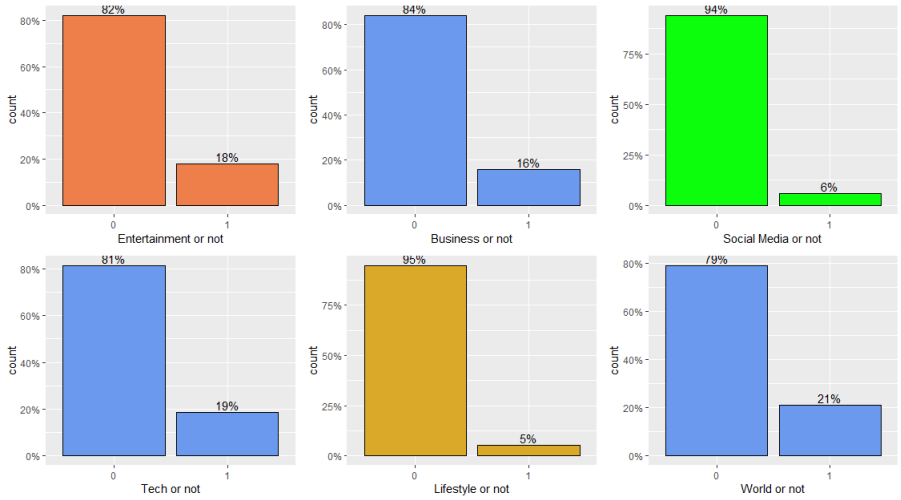

Figure 2: Types of articles publised on Mashable

18%. On the other hand, lifestyle and social networking articles are the least published on the Mashable platform.

2.4 Publication of articles by days

The figure 3 shows the publication of articles according to the days of the week.

From this graph, we can see that articles are mostly published during working days, especially on Tuesday, Wednesday and Thursday. Saturdays and Sundays (weekend) are the days where articles are the least published: 6.17 % for Saturdays and 6.89 % for Sundays.

2.5 Correlation table

Given the very large number of variables we have, we just present the 10 largest correlations. The correlation table for all quantitative variables can be found in the code file.

According to the figure 4 the most strongly correlated variables of our study are the minimum of worst keywords and the maximum of worst keywords whose correlation coefficient is higher than 0.8. We also notice a strong correlation about 0.7 between the variables weekday_is_dimanche (was the

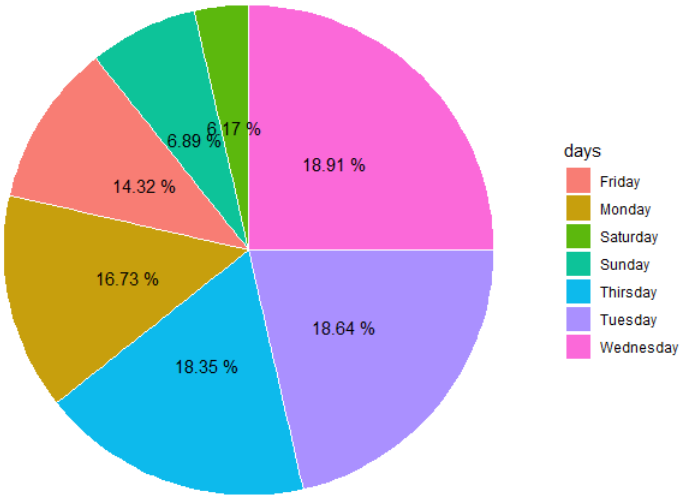

Figure 3: Publication of articles through the days of the week

article published on Sunday?) and is_week-end (was the article published on the weekend?): which is quite logical since Sunday is part of the weekend.

3 Methodology

Our work consists mainly of a series of model comparisons. We compare a multiple linear regression model, two regularization models (Lasso and Ridge), a principal component analysis regression model, a general additive model (GAM) and a random forest model. The GAM model has the secondary objective of allowing us to capture the non-linearities that escaped the linear model, in order to try to take them into account in the latter. We used R and OxMetrics as tools in the estimation of our models.

In the rest of our work, given the high number of variables in our database, we will represent the explanatory variables by X, their log by $\log(X)$ and our dependent variable by Y. Also, some of our variables are rather volatile (we considered a variance threshold of 1000) so we reduced their variability by considering their logarithm form. They are 14 in total. Among those variables, there is our dependant variable.

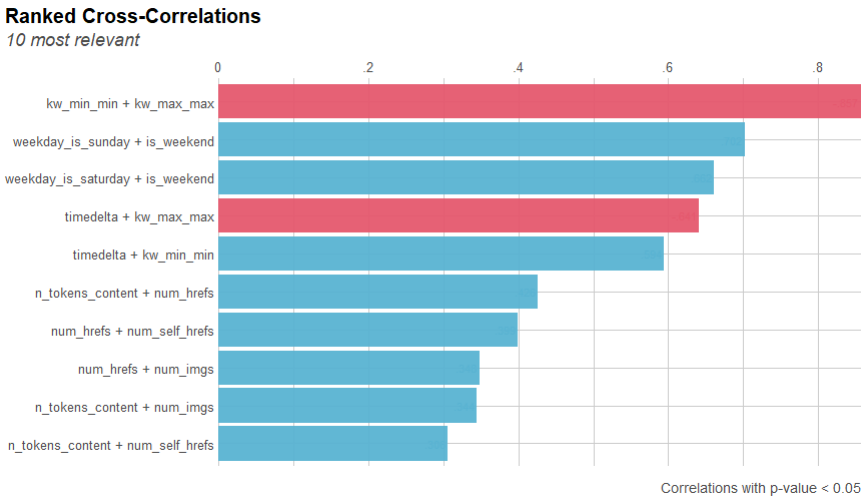

Figure 4: Correlation table

3.1 Linear regression model

The linear regression model we kept is the one obtained after a selection between three models. The three models initially specified are obtained, respectively with forward stepwise selection, a backward stepwise selection and a selection by autometrics (with a threshold of 5%). The selection criterion used for these three selection methods is the Akaïké criterion (AIC), which considers more the predictive power of the models. When the AIC won't be sufficient we can add the out-sample RMSE. The estimator of the ordinary least square model is obtained by solving the following problem:

$$\underset{\beta \in \mathbf{R}^{56}}{\operatorname{argmin}} \left(\log(Y_i) - \beta_0 - \sum_{i=1}^{44} \beta_i X_i - \sum_{i=45}^{56} \log(X_i) - \epsilon_i \right) \tag{1}$$

- X_i : represents quantitative and the binary qualitative explanatory variables
- $\log(X_i)$ represents the volatile quantitative explanatory variables. For the variables with negative modality, we consider the opposite of the logarithm of their log.

3.2 Ridge

Ridge regression is a so-called regularization regression method with a hyper-parameter λ that we determined by empirical choice rules that are discussed a bit more in the results. The estimator of the ridge model is obtained by solving the following problem:

$$\underset{\beta \in \mathbf{R}^{56}}{\operatorname{argmin}} \left(\log(Y_i) - \beta_0 - \sum_{i=1}^{44} \beta_i X_i - \sum_{i=45}^{56} \log(X_i) - \epsilon_i + \lambda \sum_{i=1}^{56} \beta_i^2 \right) \quad (2)$$

All terms in the model have the same meaning as in the model (2).

3.3 Lasso

Like the ridge model, Lasso is a regularization regression model with also a hyper-parameter λ , chosen in the same way as in the ridge model. However, Lasso has the advantage of being also a variables selection tool because its estimator shrinks to 0 the value of some coefficients whose associated variables are thus considered as not selected. The estimator is obtained by solving the following problem :

$$\underset{\beta \in \mathbf{R}^{56}}{\operatorname{argmin}} \left(\log(Y_i) - \beta_0 - \sum_{i=1}^{44} \beta_i X_i - \sum_{i=45}^{56} \log(X_i) - \epsilon_i + \lambda \sum_{i=1}^{56} |\beta_i| \right) \quad (3)$$

3.4 Regression by principal components analysis (ACP)

Principal component analysis is a data mining method that allowed us to reduce the dimension of the initial information. Beyond this reduction, the components obtained at the end of this reduction, were considered as variables that we used to make a linear regression. Moreover, since the principal component analysis only takes the qualitative variables as additional variables and they do not participate in the selection of the principal components, we considered them in the model by adding them as control variables. However, instead of adding them all in the model, we started from the regression model containing only the selected principal components and then we did a forward looking on the qualitative explanatory variables only in

order to select those that will be kept. We obtain our estimator by solving the following problem

$$\underset{\beta \in \mathbf{R}^K}{\operatorname{argmin}} \left(Y - \beta_0 - \sum_{i=1}^{K} \beta_i Z_i - \sum_{i=1}^{14} \beta_i X_i - \epsilon_i \right) \text{ where } k \leq 44 \tag{4}$$

- Z_i : is a selected principal component
- X_i : is a binary qualitative explanatory variable.
- K represents the number of selected principal components.

We can consider the number of components K to be kept here, as a hyper parameter of our model, which can be determined by cross - validation or any other classical method of determination of turning parameter. But here in the framework of our work, K designates the first K principal components so the cumulative percentage of the explained variability is 94 % (this is totally arbitrary).

3.5 General additive Model (GAM)

The GAM model is a non-parametric estimation based on the assumption that the effect of the different variables are decomposable. We estimate it in order to see its performance in the prediction of the popularity of an article but also to use it to capture the possible non-linearities in the linear regression model. The specification of the model thus described is :

$$\log(Y_i) = \sum_{i=1}^{14} X_i + \sum_{i=15}^{44} m_i(X_i) + \sum_{i=45}^{56} m_i(\log(X_i)) \tag{5}$$

m_i represents the estimated density function associated with the variable X_i

4 Random Forest

The Random Forest model is a machine learning algorithm inspired by Bagging and has the reputation of having a great performance for complex prediction problems. It consists in averaging several regression decision trees.

The trees are obtained from boostrap samples. The hyper-parameters of this model are the number of decision trees, the size of the boostrap sample and the number of random variables drawn at each node of the tree. We planned to use the k-fold validation to choose these hyper-parameters but, the technical resources we have does not allow us to do this, because the model takes too much time to run. So we kept the default parameters recommended in the documentation of the package we used(randomForest in R) . Namely, 500 decision trees and $\sqrt{p}$ number of variables considered at each node, where p represents the total number of variables.

5 Results of models

5.1 Linear regression models

The results of the linear regression model (2) are in the following table:

Table 2: Results of linear models

	Dependent variable:		
	Popularity		Popularity
	Forward	Backward	Autometrics
	(1)	(2)	(3)
kw_avg_avg	0.588***	0.583***	0.589***
	(0.040)	(0.040)	(0.040)
LDA_02	−0.215***	−155.192***	−0.118***
	(0.044)	(56.899)	(0.044)
LDA_03		−154.989***	0.066**
		(56.896)	(0.033)
data_channel_is_entertainment	−0.280***	−0.288***	−0.293***
	(0.022)	(0.020)	(0.020)
kw_avg_max	−0.087***	−0.086***	−0.090***
	(0.020)	(0.020)	(0.020)

self_reference_avg_sharess	0.255^{***}	0.258^{***}	0.257^{***}
	(0.039)	(0.042)	(0.039)
num_hrefs	0.006^{***}	0.006^{***}	0.006^{***}
	(0.001)	(0.001)	(0.001)
average_token_length	-0.085^{***}	-0.066^{***}	-0.093^{***}
	(0.010)	(0.019)	(0.009)
weekday_is_saturday	-0.014		
	(0.030)		
global_subjectivity	0.395^{***}	0.417^{***}	0.376^{***}
	(0.070)	(0.072)	(0.064)
data_channel_is_socmed	0.068^{**}	0.048^{*}	
	(0.033)	(0.027)	
num_self_hrefs	-0.009^{***}	-0.009^{***}	-0.009^{***}
	(0.002)	(0.002)	(0.002)
kw_min_min	-0.039^{***}	-0.038^{***}	-0.040^{***}
	(0.005)	(0.005)	(0.005)
data_channel_is_tech	0.035		
	(0.034)		
LDA_04	0.020	-154.941^{***}	0.118^{***}
	(0.043)	(56.898)	(0.035)
num_imgs	0.003^{***}	0.003^{***}	0.003^{***}
	(0.001)	(0.001)	(0.001)
weekday_is_tuesday	-0.301^{***}	-0.294^{***}	-0.293^{***}
	(0.024)	(0.019)	(0.019)
abs_title_sentiment_polarity	0.072^{**}	0.072^{**}	
	(0.036)	(0.036)	

weekday_is_thursday	-0.282^{***}	-0.275^{***}	-0.276^{***}
	(0.024)	(0.019)	(0.019)
weekday_is_wednesday	-0.281^{***}	-0.274^{***}	-0.273^{***}
	(0.024)	(0.019)	(0.019)
min_positive_polarity	-0.281^{***}	-0.267^{***}	-0.340^{***}
	(0.083)	(0.086)	(0.080)
LDA_01	-0.071^{**}	-155.056^{***}	
	(0.033)	(56.896)	
weekday_is_friday	-0.226^{***}	-0.219^{***}	-0.220^{***}
	(0.025)	(0.020)	(0.020)
weekday_is_monday	-0.217^{***}	-0.210^{***}	-0.210^{***}
	(0.024)	(0.020)	(0.020)
kw_max_max	-0.097^{***}	-0.096^{***}	-0.094^{***}
	(0.018)	(0.017)	(0.017)
kw_min_avg	-0.011^{***}	-0.011^{***}	-0.011^{***}
	(0.002)	(0.002)	(0.002)
kw_max_avg	-0.115^{***}	-0.114^{***}	-0.117^{***}
	(0.026)	(0.026)	(0.026)
self_reference_max_shares	-0.175^{***}	-0.176^{***}	-0.176^{***}
	(0.029)	(0.031)	(0.029)
self_reference_min_shares	-0.052^{***}	-0.053^{***}	-0.053^{***}
	(0.012)	(0.012)	(0.012)
abs_title_subjectivity	0.158^{***}	0.156^{***}	0.146^{***}
	(0.033)	(0.033)	(0.033)
kw_avg_min	0.019^{***}	0.019^{***}	0.019^{***}

	(0.006)	(0.006)	(0.006)
data_channel_is_bus	-0.233^{***}	-0.255^{***}	-0.276^{***}
	(0.034)	(0.028)	(0.024)
LDA_00	0.275^{***}	-154.699^{***}	0.390^{***}
	(0.045)	(56.898)	(0.043)
data_channel_is_world	-0.130^{***}	-0.153^{***}	-0.164^{***}
	(0.034)	(0.026)	(0.025)
data_channel_is_lifestyle	-0.123^{***}	-0.147^{***}	-0.160^{***}
	(0.036)	(0.026)	(0.025)
min_negative_polarity	-0.061^{**}	-0.061^{**}	-0.054^{***}
	(0.024)	(0.027)	(0.021)
title_subjectivity	0.061^{**}	0.061^{**}	0.092^{***}
	(0.025)	(0.025)	(0.019)
title_sentiment_polarity	0.054^{**}	0.053^{**}	0.070^{***}
	(0.023)	(0.023)	(0.021)
n_tokens_title	0.005^{*}	0.005^{*}	
	(0.003)	(0.003)	
rate_negative_words	-0.137^{**}		
	(0.060)		
n_tokens_content		-0.019^{*}	
		(0.011)	
n_unique_tokens		-0.221^{***}	
		(0.081)	
global_sentiment_polarity	-0.154	-0.141	
	(0.097)	(0.098)	

rate_positive_words		0.128**	
		(0.060)	
Constant	6.349***	161.349***	6.346***
	(0.122)	(56.902)	(0.117)
AIC	70091.433	70089.985	70093.946
RMSE	0.884	0.884	0.884
Nb var	40	41	33
Observations	27,337	27,337	27,337
R^2	0.120	0.120	0.119
Adjusted R^2	0.118	0.119	0.118
Residual Std. Error	0.871	0.871	0.871
F Statistic	92.869***	90.696***	112.030***
Note:			*p<0.1; **p<0.05; ***p<0.01

The three models obtained have the same out-sample RMSE error, i.e. (0.884). We can therefore say that the three models have almost the same performance in terms of prediction. The selection methods, forward stepwise, backward stepwise and autometrics lead to models with respectively 40, 41 and 33 explanatory variables out of 56. In fact, instead of having 58 variables, we only consider 56, because we have put the variable weekday_is_sunday as a reference for the other days of the week, and exclude the variable is_weekend to avoid multicollinearity in our models.

In total 29 variables are all selected by the methods. Most of the selected variables are significant.

Regarding the performance of our three models, the prediction error on the test sample is the same, i.e. 0.884. Thus this criterion alone is not sufficient to decide between the models. However, by coupling the AIC criterion and the R^2 to the RMSE, we keep as final model of the linear regression the model obtained with the backward stepwise whose AIC criterion is a little weaker than others methods.

5.2 Ridge

Given the penalty parameter which is an important hyper-parameter in ridge, our first objective was its determination. So we trained the model by letting

the package (glmnet in R) test 100 different values for lambda. This allows us to obtain the figure 5 which represents the picture of the penalization of the coefficients

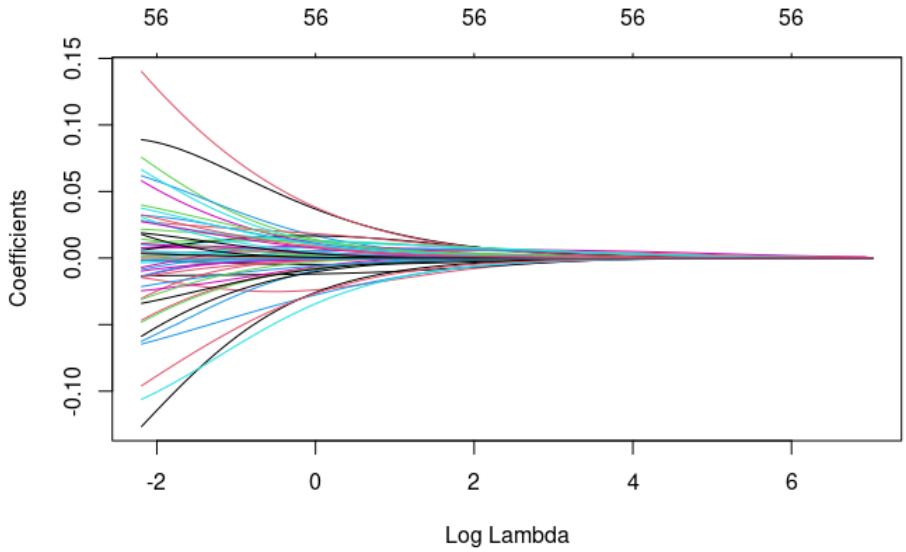

Figure 5: Ridge path coefficients

From the figure 5 we have to choose a value of λ at which the coefficients start to be stable. Although graphically we can claim a value of λ that is considered optimal, this way of choosing is certainly not suitable. For this purpose, we propose another more formal and objective way of choosing λ, which is the k-fold cross validation. Thus, after having trained the model again by a 10-fold cross validation, we obtain the figure 6 which represents the evolution of MSE according to the log of lambda.

For the choice of the optimal value of λ, two rules of thumb are available according to the literature:

- choose the value of λ associated with the smallest value of MSE , which corresponds graphically to the 1^{er} vertical dotted line (Min rule) ;
- the standard deviation rule, which consists in choosing the largest value of lambda, whose average error is lower than the upper limit of the

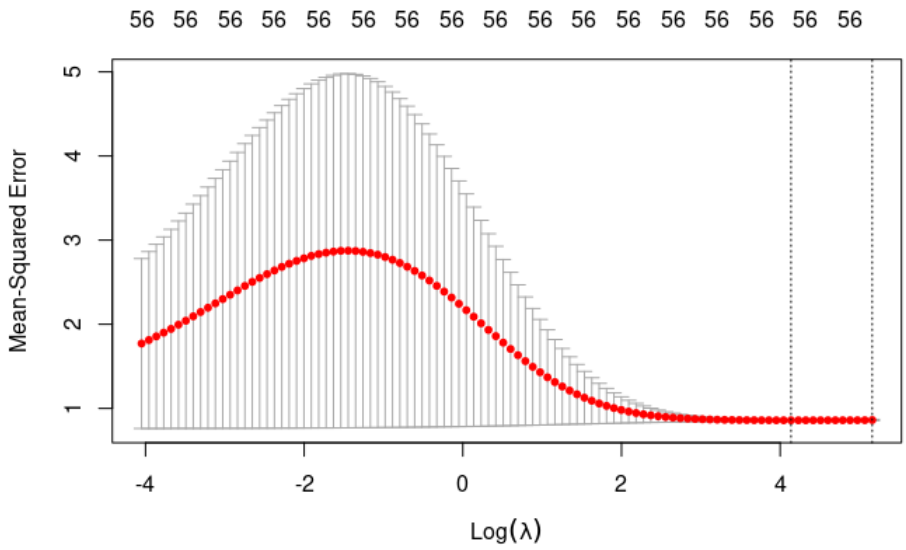

Figure 6: Ridge Cross Validation MSE path

confidence interval of the optimal error (std rule). This criterion corresponds to the second vertical dotted line.

To make our choice, we decided to evaluate the model with the different values on our test sample in order to choose the model that does better. The result is recorded in the table 3.

Table 3: Performance comparison of the two λ selection criteria

	Min rule	Std rule
lambda	62.624	174.255
RMSE	0.939	0.942

From the analysis of the table 3, we notice that the two models do almost equally well on the test sample, however the model having with the lambda selection under the criterion "Min rule" does a little better than the second model. It is therefore this first model that we will keep for the rigde regression

model. The table A.1 available in appendix presents the different parameters obtained for the final model.

5.3 Lasso

As in the case of the ridge model, the first thing we did was to determine the hyper-parameter λ. We adopted the same procedure as in the case of the ridge model. The curve of the coefficients evolution is available in figure 8 in appendix and also the one of the MSE evolution in cross validation as a function of the lambda values in figure 9. The comparison of the performances according to the two lambda selection criteria is presented in the table 4

Table 4: Performance comparison of the two λ selection criteria for Lasso

	Min rule	Std rule
lambda	0.043	0.052
RMSE	0.909	0.912

From the table 4, we notice that the model obtained with the Min rule (RMSE = 0.909) does better than the std rule(RMSE=0.912). We therefore retain the Lasso model with $\lambda = 0.043$ as final model. Moreover we know that the Lasso model is also a variable selection tool, because unlike ridge it can shrink the value of the coefficient of the unselected variables to 0. So we can compare it with the regression model.

Table 5: Selected variables from OLS and Lasso

var	Ols	Lasso
(Intercept)	161.349	
abs_title_sentiment_polarity	0.072	
abs_title_subjectivity	0.156	
average_token_length	-0.066	-0.002
data_channel_is_bus	-0.255	
data_channel_is_entertainment	-0.288	
data_channel_is_lifestyle	-0.147	
data_channel_is_socmed	0.048	
data_channel_is_world	-0.153	-0.03

Table 5: Selected variables from OLS and Lasso

var	Ols	Lasso
global_sentiment_polarity	-0.141	
global_subjectivity	0.417	
kw_avg_avg	0.583	0.092
kw_avg_max	-0.086	
kw_avg_min	0.019	0.015
kw_max_avg	-0.114	0.013
kw_max_max	-0.096	
kw_max_min		0.006
kw_min_avg	-0.011	
kw_min_max		0.001
kw_min_min	-0.038	
LDA_00	-154.699	
LDA_01	-155.056	
LDA_02	-155.192	
LDA_03	-154.989	
LDA_04	-154.941	
min_negative_polarity	-0.061	
min_positive_polarity	-0.267	
n_tokens_content	-0.019	-0.02
n_tokens_title	0.005	
n_unique_tokens	-0.221	
num_hrefs	0.006	0.007
num_imgs	0.003	0.005
num_keywords		0.006
num_self_hrefs	-0.009	-0.006
num_videos		0.001
rate_positive_words	0.128	
self_reference_avg_sharess	0.258	0.011
self_reference_max_shares	-0.176	
self_reference_min_shares	-0.053	0.013
title_sentiment_polarity	0.053	
title_subjectivity	0.061	
weekday_is_friday	-0.219	
weekday_is_monday	-0.21	
weekday_is_thursday	-0.275	

Table 5: Selected variables from OLS and Lasso

var	Ols	Lasso
weekday_is_tuesday	-0.294	
weekday_is_wednesday	-0.274	
==========	==========	==========
Number of var	41	15

The Lasso model has the reputation of selecting too many variables, and not necessarily only the relevant ones. However we notice from the table 5 that the backward stepwise selection used in linear regression selected more variables than the Lasso model, 42 against 15. The Lasso model selected far fewer variables this time. However, what is important to us is the predictive power between our two models; what we compare in the section 8

5.4 Principal components analysis

As explained in the subsection 3.4, we have first determined the principal components, then we have regressed the number of shares on the components that contain the 94% of information, in total the first 24 components in our case, as shown in the table 9 available in appendix. The result of the regression thus made is summarized in the following table 6:

Table 6: Principales components analysis regression

	Dependent variable:
	shares
Dim.1	0.033***
	(0.002)
Dim.2	0.044***
	(0.003)
Dim.3	0.060***
	(0.003)
Dim.4	0.001

Table 6: Principales components analysis regression

	Dependent variable:
	shares
	(0.003)
Dim.5	0.043^{***}
	(0.003)
Dim.6	0.019^{***}
	(0.003)
Dim.7	0.052^{***}
	(0.004)
Dim.8	-0.026^{***}
	(0.004)
Dim.9	0.033^{***}
	(0.004)
Dim.10	-0.051^{***}
	(0.005)
Dim.11	-0.058^{***}
	(0.005)
Dim.12	0.029^{***}
	(0.005)
Dim.13	0.009^{*}
	(0.006)
Dim.14	-0.028^{***}
	(0.006)
Dim.15	-0.051^{***}

Table 6: Principales components analysis regression

	Dependent variable:
	shares
	(0.005)
Dim.16	0.030^{***}
	(0.005)
Dim.17	-0.038^{***}
	(0.005)
Dim.18	0.046^{***}
	(0.005)
Dim.19	-0.004
	(0.005)
Dim.20	-0.011^{*}
	(0.006)
Dim.21	0.033^{***}
	(0.006)
Dim.22	0.077^{***}
	(0.006)
Dim.23	0.008
	(0.006)
Dim.24	-0.032^{***}
	(0.006)
data_channel_is_entertainment	-0.332^{***}
	(0.018)
is_weekend	0.292^{***}

Table 6: Principales components analysis regression

	Dependent variable:
	shares
	(0.014)
data_channel_is_bus	-0.202^{***}
	(0.029)
data_channel_is_socmed	0.172^{***}
	(0.028)
data_channel_is_tech	0.092^{***}
	(0.028)
weekday_is_friday	0.077^{***}
	(0.013)
weekday_is_monday	0.071^{***}
	(0.012)
data_channel_is_world	-0.074^{***}
	(0.028)
data_channel_is_lifestyle	-0.048
	(0.029)
Constant	7.496^{***}
	(0.018)
RMSE	0.889
Observations	39,016
R^2	0.108
Adjusted R^2	0.107
Residual Std. Error	0.881 (df = 38982)
F Statistic	142.626^{***} (df = 33; 38982)

All the principal components are significant, except the 19^{th} and the 23^th at a threshold of 10%. In addition, the 13^{th} and the 20^{th} are not significant at a threshold of 1%. As for the qualitative variables, 9 were kept selected by the forward stepwise and they are all significant at a threshold of 1%.

6 General Addictive Model (GAM)

The GAM model we obtained has an out-sample RMSE of 0.864. All tables and graphs of the estimation are available in the section A.4 in the appendix. The confidence intervals of our non-linear components that we estimated are not significant, except for those of the variables "Rate of unique words in the content" (n_unique_token), "Rate of non-stop words in the content" (n_non_stop_words) and "Rate of unique non-stop words in the content" (n_non_stop_unique_token) as shown in figure 10 in appendix. However, these obtained components do not present any non-linearity. Thus the GAM model obtained is nothing but the equivalent of a linear model containing all the explanatory variables.

7 Random Forest

The only relevant result we present on the random forest is its out-sample RMSE. It is presented in the comparison section of all the models.

8 Comparison between all models

In this section, we compare the performance of the different models and choose the one we keep among the six (06) proposed models. The summary is recorded in the following table 7:

In the end, from the table 7, the model that does best in terms of performance, with the RMSE selection criteria, is either the Random Forest model or the GAM model. We list these two models because they have same RMSE, in fact the difference of 0.001 could come from randomization of the sample. So any model among both could be retained.

Another interesting result that we obtain here is that the PCA model as we have proposed it also performs quite well, and very close to the OLS model. Indeed, we noticed in our simulations that as we increase the threshold of

Table 7: Performance comparison between all models

models	RMSE	var
OLS	0.884	41
Ridge	0.939	56
Lasso	0.912	12
PCA	0.889	33
GAM	0.864	56
Random Forest	0.863	-

variance explained for the selection of the components, in other words, when we increase the number of principal components in the model, the RMSE reduces (but slightly), to the point where the PCA regression exceeds the OLS model in performance already from the insertion of the 36^{th} principal component. The result of this simulation is shown in the appendix in table 10. This means that PCA regression can be a very good alternative, when we are interested in the prediction of a quantitative variable and we have a large number of variables, especially when a small number of the first principal components explain a large variability of the data (which is not the case here), because we will thus have a model with very few explanatory variables (thus faster in execution) but which does as well as a sophisticated model with many more variables.

9 Conclusion

Our study focuses on predicting the popularity of an article published on the Mashable website. The objective of the study is to have a good prediction of the popularity of an article published on Mashable by comparing six (06) econometric models that are a linear regression model, a Ridge model, a Lasso model, a regression model by principal component analysis, a random forest model and a GAM model. We have measured the popularity of an article using as proxy the number of shares of the article. The study is structured in tree (03) main parts which are the presentation of the data of the study, the methodology of the study and the results.

First, we get the linear regression model by using respectively the forward stepwise, backward stepwise and autometrics selection methods. After the implementation of these three selection methods and based particularly on the AIC criterion, the best model retained is the linear regression model obtained by the backward stepwise. This model allowed us to select 41 explanatory variables out of 56 and has a RMSE of 0.884.

Next, we implemented the variable regularization models that are the ridge regression and the lasso regression. Also, with the regression by principal component analysis we get a model with 33 variables having a RMSE of 0.889 when we decide to select the components that provide 94% of the information. However, we notice that with this model as we increase the principal components, we have a reduction of the RMSE.

Finally, we also estimate others machine learning models like Random Forest and GAM. Both models have the same performance in our prediction goal so they are all eligible as the final model.

References

[1] K. Fernandes, P. Vinagre, and P. Cortez. "A Proactive Intelligent Decison Support System for Predicting the Popularity of Online News". In: (September 2015). URL: https://www.researchgate.net/publication/283510525_A_Proactive_Intelligent_Decision_Support_System_for_Predicting_the_Popularity_of_Online_News.

[2] Marek Hlavac. "Stargazer: Well-Formatted Regression and Summary Statistics Tables". In: (). URL: https://CRAN.R-project.org/package=stargazer.

[3] Yan Holtz. In: (2021). URL: https://www.r-graph-gallery.com.

[4] Redacteur. "Mesurer la popularité de son contenu : 8 indicateurs clés (redacteur.com)". In: (2021). URL: https://www.redacteur.com/blog/mesurer-la-popularite-de-son-contenu/.

A Appendix

A.1 Coefficients of ridge regression

Table 8: Final Ridge regression results

n_tokens_title	-0.0005
n_tokens_content	-0.001
n_unique_tokens	0.0002
n_non_stop_words	0.0002
n_non_stop_unique_tokens	0.0001
num_hrefs	0.005
num_self_hrefs	-0.001
num_imgs	0.004
num_videos	0.001
average_token_length	-0.001
num_keywords	0.001
data_channel_is_lifestyle	0.0001
data_channel_is_entertainment	-0.0005
data_channel_is_bus	-0.0001
data_channel_is_socmed	0.0002
data_channel_is_tech	0.0003
data_channel_is_world	-0.001
kw_min_min	0.0004
kw_max_min	0.001
kw_avg_min	0.001
kw_min_max	0.001
kw_max_max	0.0002
kw_avg_max	0.0005
kw_min_avg	0.001
kw_max_avg	0.001
kw_avg_avg	0.001
self_reference_min_shares	0.003
self_reference_max_shares	0.003
self_reference_avg_sharess	0.003
weekday_is_monday	-0.00003
weekday_is_tuesday	-0.0002

weekday_is_wednesday	-0.0001
weekday_is_thursday	-0.0001
weekday_is_friday	0.00000
weekday_is_saturday	0.0002
LDA_00	0.0002
LDA_01	-0.0002
LDA_02	-0.001
LDA_03	0.0004
LDA_04	0.0002
global_subjectivity	0.0001
global_sentiment_polarity	0.0001
global_rate_positive_words	0.00001
global_rate_negative_words	-0.00000
rate_positive_words	0.00002
rate_negative_words	-0.0001
avg_positive_polarity	0.00003
min_positive_polarity	-0.00001
max_positive_polarity	0.00004
avg_negative_polarity	-0.00003
min_negative_polarity	-0.00001
max_negative_polarity	-0.00002
title_subjectivity	0.0002
title_sentiment_polarity	0.0002
abs_title_subjectivity	0.00000
abs_title_sentiment_polarity	0.0002

A.2 Supplement on PCA

Table 9: Cumulative explained percentages of components

	eigenvalue	variance.percent	cumulative.variance.percent
Dim.1	5.613	12.757	12.757
Dim.2	4.221	9.593	22.351
Dim.3	3.515	7.988	30.338
Dim.4	3.003	6.824	37.162
Dim.5	2.680	6.092	43.254

Table 9: Cumulative explained percentages of components

	eigenvalue	variance.percent	cumulative.variance.percent
Dim.6	2.426	5.513	48.767
Dim.7	2.266	5.150	53.917
Dim.8	1.879	4.270	58.188
Dim.9	1.806	4.104	62.292
Dim.10	1.592	3.618	65.910
Dim.11	1.453	3.303	69.213
Dim.12	1.336	3.036	72.249
Dim.13	1.213	2.757	75.006
Dim.14	1.151	2.616	77.622
Dim.15	1.104	2.510	80.132
Dim.16	0.931	2.115	82.247
Dim.17	0.851	1.933	84.180
Dim.18	0.779	1.771	85.951
Dim.19	0.755	1.716	87.667
Dim.20	0.658	1.494	89.162
Dim.21	0.632	1.437	90.599
Dim.22	0.603	1.370	91.969
Dim.23	0.569	1.293	93.261
Dim.24	0.512	1.163	94.425
Dim.25	0.450	1.023	95.447
Dim.26	0.428	0.973	96.420
Dim.27	0.323	0.735	97.155
Dim.28	0.257	0.583	97.738
Dim.29	0.246	0.558	98.296
Dim.30	0.187	0.424	98.721
Dim.31	0.125	0.284	99.004
Dim.32	0.093	0.211	99.216
Dim.33	0.079	0.179	99.395
Dim.34	0.061	0.138	99.533
Dim.35	0.059	0.134	99.667
Dim.36	0.045	0.101	99.768
Dim.37	0.042	0.095	99.863
Dim.38	0.029	0.065	99.928
Dim.39	0.019	0.044	99.973

Table 9: Cumulative explained percentages of components

	eigenvalue	variance.percent	cumulative.variance.percent
Dim.40	0.011	0.025	99.997
Dim.41	0.001	0.002	100.000
Dim.42	0.0002	0.0004	100.000
Dim.43	0.00003	0.0001	100
Dim.44	0	0	100

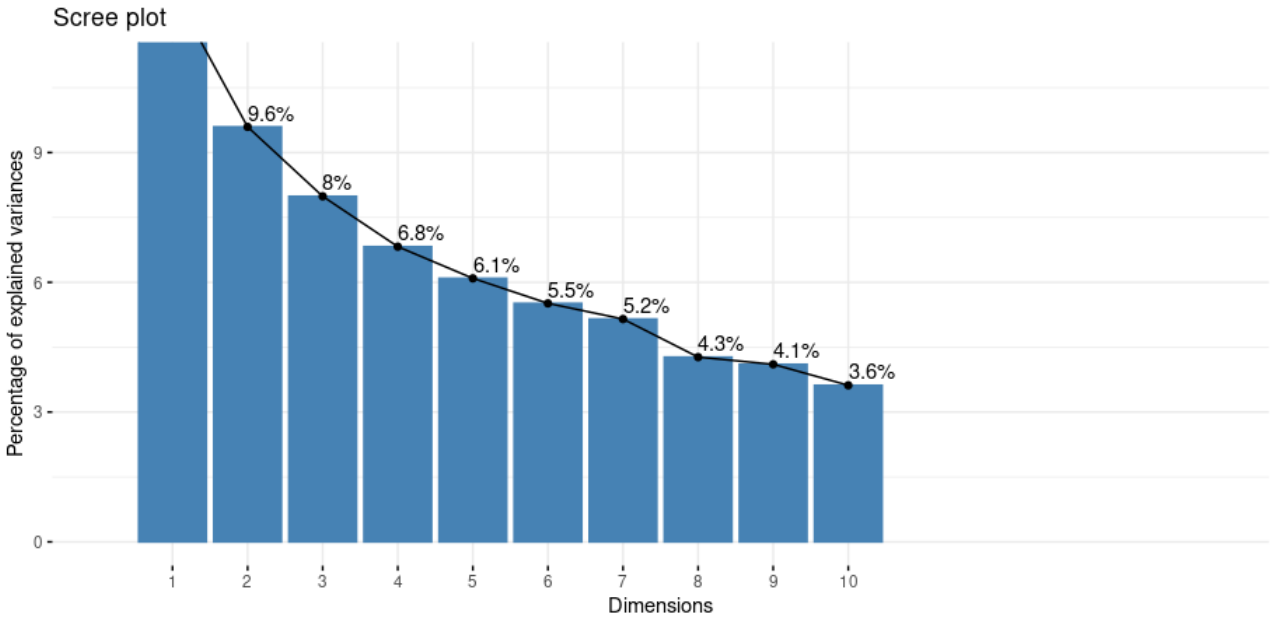

Figure 7: Cumulative explained percentage histogramme

Table 10: MSE in function of the numbers of principal components

Number of component	RMSE
1	0.907
2	0.905

Table 10: MSE in function of the numbers of principal components

Number of component	RMSE
3	0.903
4	0.903
5	0.902
6	0.901
7	0.897
8	0.897
9	0.897
10	0.895
11	0.894
12	0.894
13	0.894
14	0.894
15	0.893
16	0.893
17	0.893
18	0.892
19	0.892
20	0.892
21	0.891
22	0.889
23	0.889
24	0.889
25	0.889
26	0.887
27	0.886
28	0.885
29	0.885
30	0.884
31	0.884
32	0.884
33	0.884
34	0.884
35	0.884

Table 10: MSE in function of the numbers of principal components

Number of component	RMSE
36	0.883
37	0.882
38	0.882
39	0.882
40	0.882
41	0.882
42	0.882
43	0.881
44	0.881

A.3 Supplement on Lasso

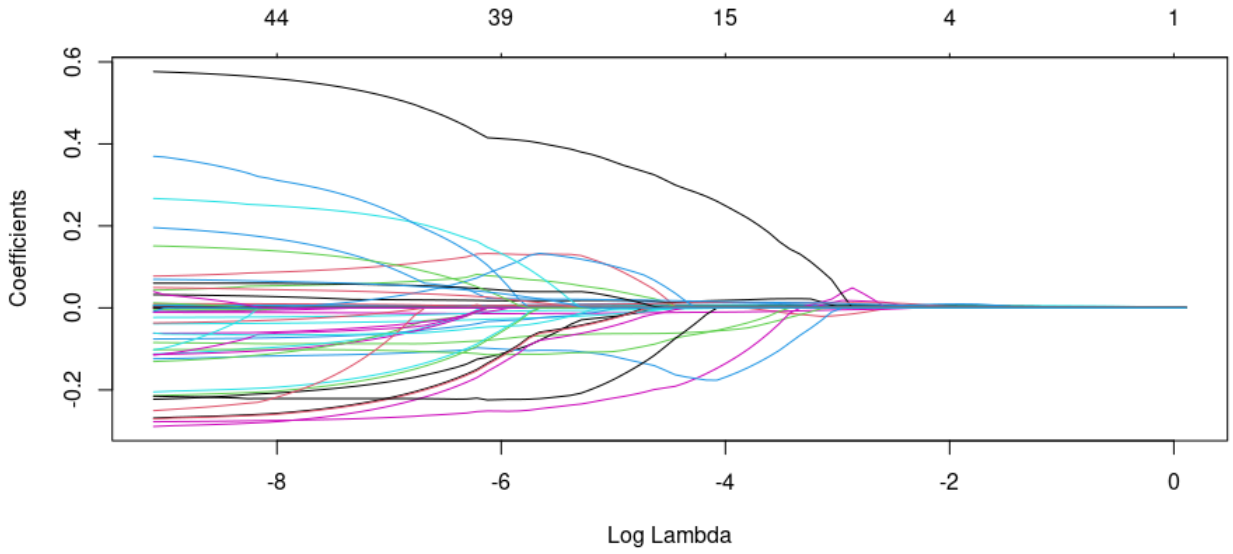

Figure 8: Lasso path coefficients

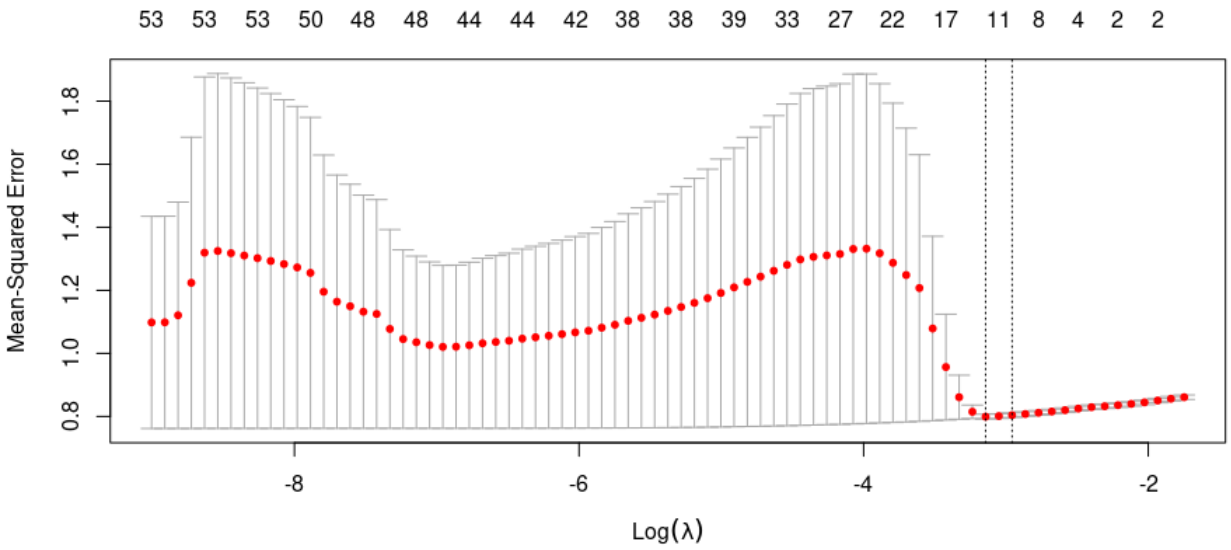

Figure 9: Lasso cross validation MSE path

A.4 Supplement on GAM

Table 11: Results of the linear components of the GAM model

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	7.710	0.029	261.739	0
data_channel_is_lifestyle	-0.074	0.036	-2.057	0.040
data_channel_is_entertainment	-0.127	0.024	-5.311	0.00000
data_channel_is_bus	-0.085	0.035	-2.419	0.016
data_channel_is_socmed	0.095	0.035	2.752	0.006
data_channel_is_tech	0.122	0.034	3.572	0.0004
data_channel_is_world	0.007	0.035	0.189	0.850
weekday_is_monday	-0.227	0.024	-9.636	0
weekday_is_tuesday	-0.296	0.023	-12.778	0
weekday_is_wednesday	-0.275	0.023	-11.841	0
weekday_is_thursday	-0.276	0.023	-11.896	0
weekday_is_friday	-0.222	0.024	-9.213	0

Table 11: Results of the linear components of the GAM model

	Estimate	Std. Error	t value	Pr(>|t|)
weekday_is_saturday	-0.021	0.029	-0.743	0.457

Table 12: Results of the non-linear component of the GAM model

	edf	Ref.df	F	p-value
s(n_tokens_title)	1.761	2.190	5.623	0.003
s(n_tokens_content)	4.436	4.842	4.071	0.001
s(n_unique_tokens)	4.191	4.714	4.925	0.006
s(n_non_stop_words)	1.000	1.000	0.778	0.378
s(n_non_stop_unique_tokens)	1.330	1.589	0.235	0.736
s(num_hrefs)	1.746	2.165	23.210	0
s(num_self_hrefs)	4.424	4.781	3.325	0.018
s(num_imgs)	4.243	4.671	3.551	0.004
s(num_videos)	3.172	3.700	6.114	0.0001
s(average_token_length)	1.000	1.000	3.692	0.055
s(num_keywords)	4.645	4.923	4.618	0.0003
s(kw_min_min)	2.926	3.223	22.910	0
s(kw_max_min)	4.634	4.917	4.368	0.001
s(kw_avg_min)	4.990	4.998	4.631	0.0003
s(kw_min_max)	1.000	1.000	41.591	0
s(kw_max_max)	3.739	4.286	4.501	0.001
s(kw_avg_max)	2.377	3.052	9.197	0.00000
s(kw_min_avg)	3.954	4.504	17.601	0
s(kw_max_avg)	4.456	4.828	16.738	0
s(kw_avg_avg)	3.856	4.490	64.278	0
s(self_reference_min_shares)	4.804	4.964	2.874	0.017
s(self_reference_max_shares)	3.993	4.472	5.464	0.0005
s(self_reference_avg_sharess)	3.934	4.439	7.193	0.00000
s(LDA_00)	1.442	1.805	1.920	0.250
s(LDA_01)	1.597	1.984	4.815	0.006
s(LDA_02)	2.740	3.329	5.141	0.001
s(LDA_03)	2.931	3.534	2.847	0.032

Table 12: Results of the non-linear component of the GAM model

	edf	Ref.df	F	p-value
s(LDA_04)	1.360	1.713	1.006	0.241
s(global_subjectivity)	2.859	3.582	4.388	0.003
s(global_sentiment_polarity)	1.000	1.000	0.778	0.378
s(global_rate_positive_words)	1.000	1.000	3.837	0.050
s(global_rate_negative_words)	3.543	4.233	1.825	0.126
s(rate_positive_words)	1.000	1.000	0.779	0.378
s(rate_negative_words)	1.000	1.000	0.172	0.679
s(avg_positive_polarity)	1.000	1.000	0.564	0.453
s(min_positive_polarity)	2.382	2.944	6.448	0.0003
s(max_positive_polarity)	1.000	1.000	0.890	0.346
s(avg_negative_polarity)	4.358	4.776	2.952	0.037
s(min_negative_polarity)	1.876	2.335	0.437	0.664
s(max_negative_polarity)	2.165	2.706	3.280	0.032
s(title_subjectivity)	1.000	1.000	7.092	0.008
s(title_sentiment_polarity)	2.316	2.740	2.447	0.136
s(abs_title_subjectivity)	2.287	2.770	5.921	0.001
s(abs_title_sentiment_polarity)	2.600	3.143	1.107	0.372

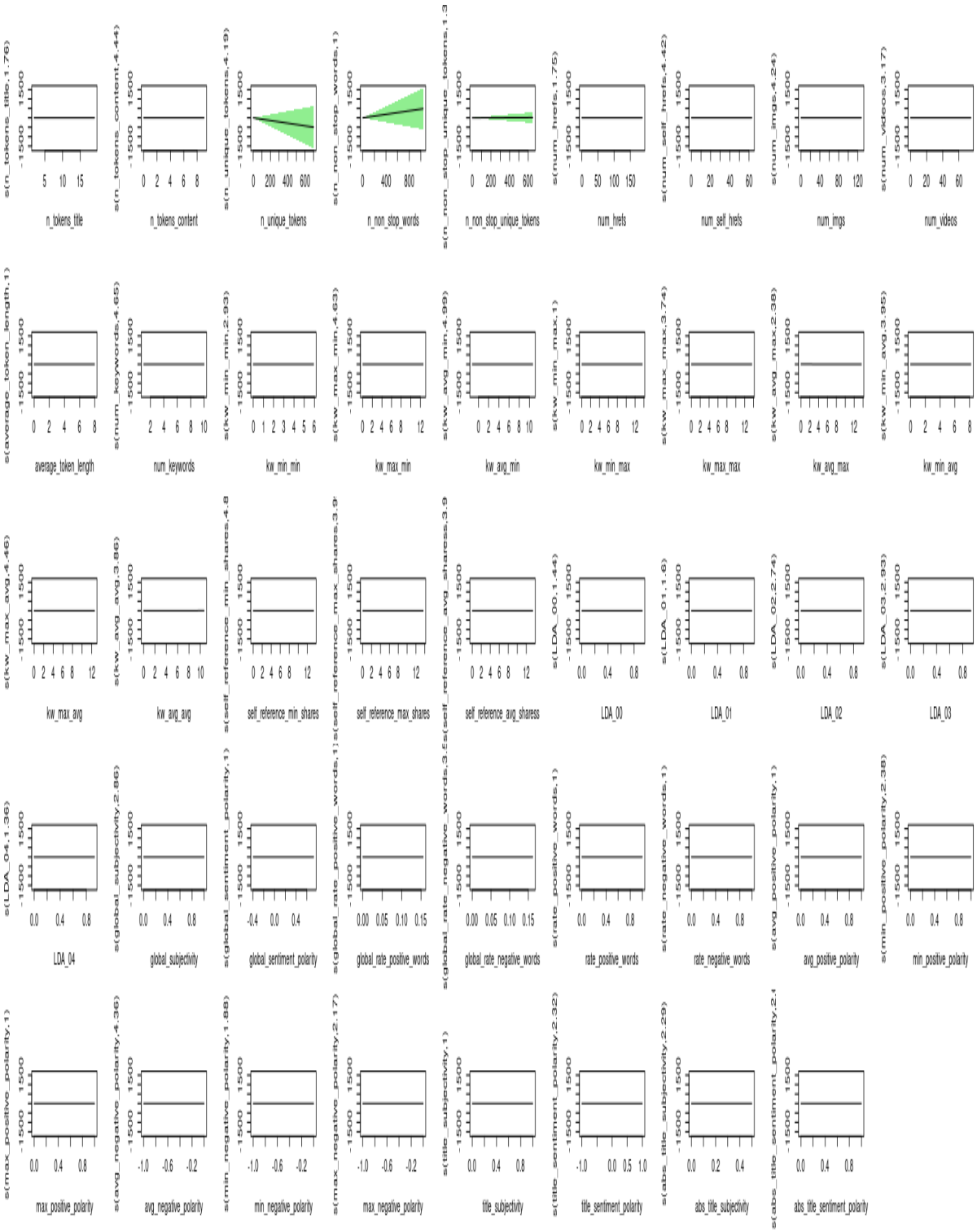

Figure 10: Non-linear components curves