Master 2 Economics

Econometrics, big data, statistics (EBDS)

Machine learning and statistical learning

Predicting Online News Popularity of Mashable articles

Realized by :
CAF Aimane

Directed by :
Mr. Ewen GALLIC
Mr. Sebastien LAURENT
Mr. Pierre MICHEL

Abstract

In this work, we were interested in predicting the popularity of online news based on a set of their characteristics. In this context, we used parametric methods and those of machine learning. To take into account the problems related to dimensionality and multicollinearity, we performed regularized logistic regression based on Elastic Net and a logistic regression based on Principal Component Analysis (PCA).

The machine learning methods used are CART, bagging, Random Forest, K-Nearest Neighbors, Support Vector Machine and Neural networks. Each of the methods are evaluated on a test sample for evaluation and comparison with the other methods. The choice of the best method is made on the basis of the overall accuracy rate.

Keywords— News articles, Popularity, Classification, Prediction.

Table of contents

Introduction

Online news has boomed in recent years, especially with social media. Indeed, social networks like Facebook have more than 2.41 billion active accounts, YouTube with 2 billion accounts and Instagram with 1 billion active accounts in 2019 **(Clément, 2019)**. It is also obvious that the majority of current information is information posted online. This appearing evolution of online news has been possible thanks to the great spread of smartphones and the wide distribution of mobile applications in a context of globalization. Online news is nowadays a major instrument in the dissemination of information quickly and in real time. Thus, they began to arouse great interest, with the progress of the digital world. This interest is addressed in several aspects including popularity.

The popularity of online news is generally measured by taking into account the number of interactions on the Web and social networks such as the number of shares, reviews and comments **(Fernandes et al, 2015)**. Thus defined, the popularity of online news appears to be a major issue for various actors in the digital world or elsewhere. Taking online information into account presents remarkable practical values for actors such as authors, content creators, advertisers, influencers and even political actors. For example, for press organizations, they can better understand the different types of user consumption and produce more relevant content in real time and have better organization. However, it is difficult to predict, before publication, whether an online article will be successful or popular, given all the factors to be taken into account, including the specific characteristics of the article.

Our goal in this study is to predict the popularity of online news before its publication. To do this, we will opt on the one hand for parametric econometric methods and on the other hand for machine learning techniques *(CART, Bagging, Random Forest, KNN, SVM, and neural networks)*. To carry out this study, we will adopt the following work plan : first, a detailed introduction to the study problem. Second, we will present the database and descriptive statistics. Third, we will expose the techniques used and the different results. Before concluding, we will compare our different methods.

1 Literature review

The theory of popularity prediction focused in its early days on studying the behavior of users of web content and video on demand **(Tatar et al, 2014)**. Two major theories are known on this subject. The first suggests that the request for access to web content follows a Zipf law. Indeed, for authors like **Arlitt et al (1996)** then **Chesire et al (2001)**, the degree of file sharing and therefore the distribution of popularity follows a distribution which is of Zipf type. The second assumes that the popularity of content does not follow this law. **Archarya et al (1999)** conclude that, unlike the above authors, the rankings of video titles by popularity do not follow a Zipf distribution. This study of the demand for web pages will undergo a mutation and look into models to predict the popularity of web content and videos on demand.

One of the first prediction studies that caught our attention on this subject is that of **Pinto et al (2013)** which uses machine learning techniques to predict the popularity of YouTube videos based on the number of views in the preceding days. Similarly, **Roy et al (2013)** using an SVM machine learning model and relying on data composed of 10.2 million tweets and 3.5 million YouTube videos, maintain that the social importance of video subject (context) is responsible for the sudden increase in its popularity. Their results will be useful in the multimedia field since the prediction is between two multimedia sectors (YouTube and tweeter).

Although taking into account several domains **(Roy et al, 2013)** is interesting, the proposed modeling may not be the best. Some authors propose several prediction models in the same study in order to get the best out of them. To do this, **Fernandes et al (2015)** through five models of Machine Learning that are the Random Forest, the Naive Bayes, the AdaBoost, the SVM and the KNN set up a proactive system of intelligent decision support for predict the popularity of online news. For this work, they built a database of more than 40,000 articles posted online on the Mashable site between 2013 and 2015. They conclude that the Random Forest is the best model in the context of their study with an area under the ROC curve of 0.73. Still with the same concern of predicting popularity by comparing several prediction models, **Lui (2017)** through the database of **Fernandes et al (2015)** estimates several regression and classification models to predict the online popularity of certain articles. He notes on the one hand that the GAM model gives more explanation on the popularity of the articles, and on the other hand with the classification methods where the Random Forest leads to the best predictions.

2 Data Analysis

Our data on the online news popularity prediction was created and introduced by Fernandes and al in their article published in 2015, available in UCI Machine Learning Repository. It was collected during a two year period, from January 7, 2013 to January 7, 2015. It contains 39,644 observations and 61 variables. The data description made by Fernandes and al is in the **appendix A.6**.

2.1 Description of variables

The database is made up of different variables, constituted as descriptive characteristics of an article, which can influence its popularity. We will give an overview of some variables.

Category	Features
Words	- Number of words in the title - Number of words in the article - Average word length - Rate of non-stop words - Rate of unique words
Links	- Number of links - Number of links to other articles published by Mashable
Digital Media	- Number of images - Number of videos
Time	- Day of the week - Published on a weekend ?
Keywords	- Number of keywords - Article category (Mashable data channel)
Natural language processing	- Closeness to top 5 LDA topics - Title subjectivity - Rate of positive and negative words
Target variable	- Number of article Mashable shares

Table 1 – An overview of the variables

As we notice when viewing the table above, the database includes important characteristics related to the article, such as determining the number of words, keywords, links, category, images, videos, etc. To take into account non-linearities with the target variable, we therefore created new variables through interactions, and elevations of the initial variables squared. We ended up with 78 variables.

2.2 Data cleaning

After a quick exploration of the graphical representations, we cleaned up the database using the **interquartile range** method on our main variable, which is the number of shares, located outside the interval. The calculation method is presented according to the following formula :

$$\left[\mathbf{Q1} - \mathbf{k}(\mathbf{Q3} - \mathbf{Q1}); \mathbf{Q3} + \mathbf{k}(\mathbf{Q3} - \mathbf{Q1})\right] \textit{ with } k = 1.5$$

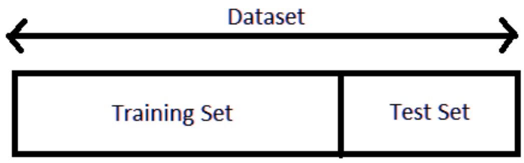
This method therefore allows us to go from 39,644 observations initially to 35,103. To possibly take into account the non-linearity between the number of shares and variables such as the number of words in the content of the articles and the number of links contained in the article, we have took the square of these variables.

Regarding the possible interactions, we suppose that the impact of each of the variables such as the number of videos, the number of images, the number of words in the content of articles can be a function of their fields of publication (data channel). To integrate this aspect, we have created interactions between these variables and the different domains.

Regarding the dependent variable which is the number of shares, we choose to categorize it into two different groups namely "Popular" and "non-popular" taking the values 0 and 1 respectively. Indeed, we consider as popular an article whose number of shares is greater than 1300 (Median of the cleaned base) otherwise if it is unpopular.

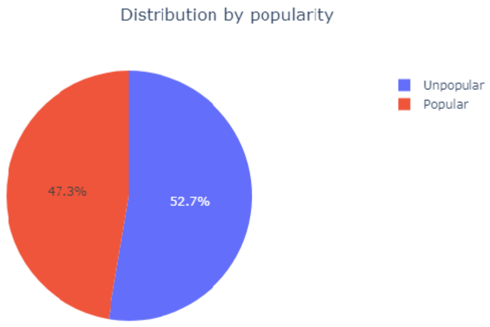
It is on this new basis that we make our different graphical representations and apply our various methods.

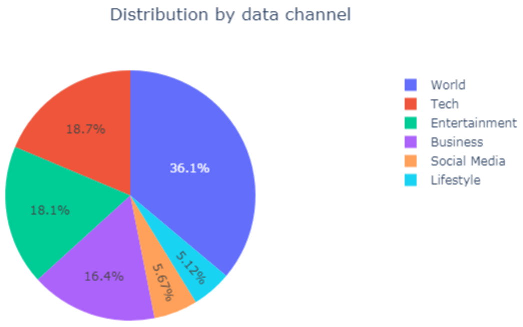
Finally for training, validation and evaluation of our different models, we have subdivided our new database into training sample (80%) and test sample (20%).



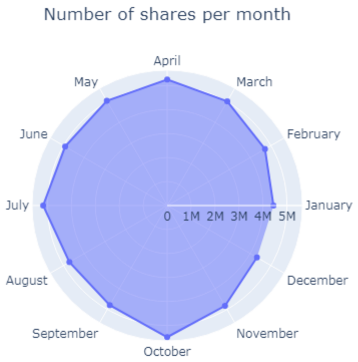
2.3 Descriptive statistics

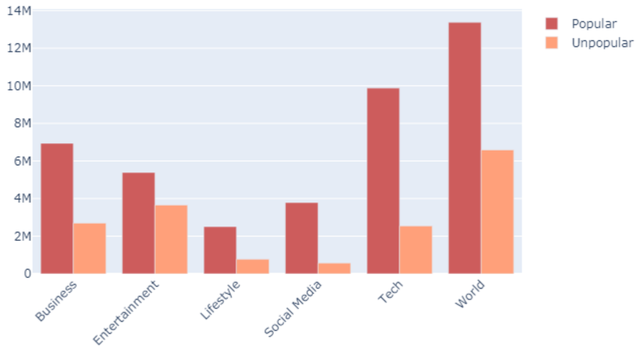
The graphical visualization of the pie charts indicates practically equal shares between popular and non-popular articles, and shows that the most common data channel category is ”World”.



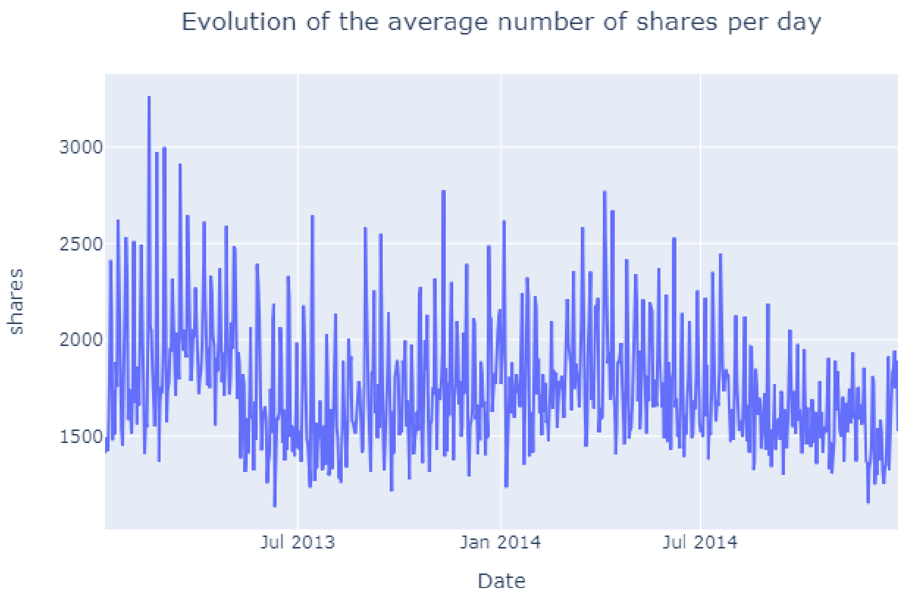


The radar chart shows a monthly distribution according to the number of sharing of articles. We note that, for two years, the most shared articles were in October, with more than 5 million number of shares. Thus, the most widespread category at Mashable is ”World”, and ”Technology” with roughly 13 Million and 10 Million shares, respectively.

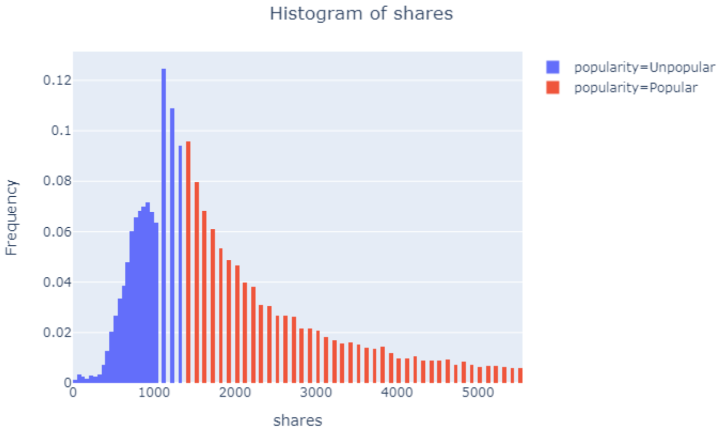




Through this graph below, we can see the average evolution of the number of shares per day. We note the existence of fluctuations and a deterioration in the average number of sharing per day over time. After July 2014, we start to revolve around the range of 1000 to 2000 shares per day.



The histogram below provides information on the distribution of shares for the two different classes. We notice that there is more concentration in the unpopular class. There is an asymmetry to the right, that is to say that the small observed values are more frequent than the higher.



2.4 Unsupervised method : K-Means

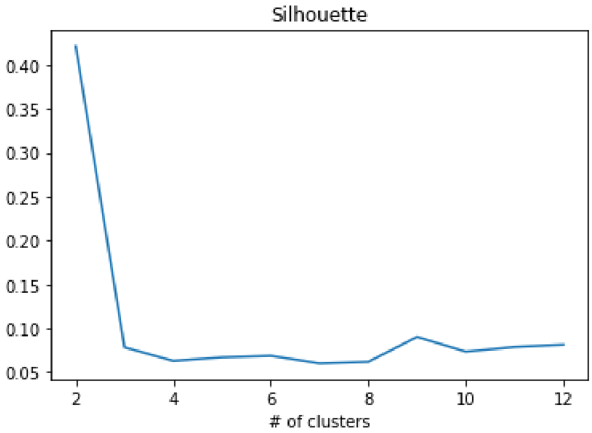
K-means clustering is one of the algorithms for automatic classification of the unsupervised family. It is widely used among clustering methods. These methods which are techniques which try to divide the data into homogeneous groups based on the similarity of the points.

The main idea of K-means is to determine the k centers, one for each group of similar individuals. This algorithm therefore makes it possible to classify each observation or individual in a suitable group by calculating the distance between each point and the centers. The choice is made on the minimum distance.

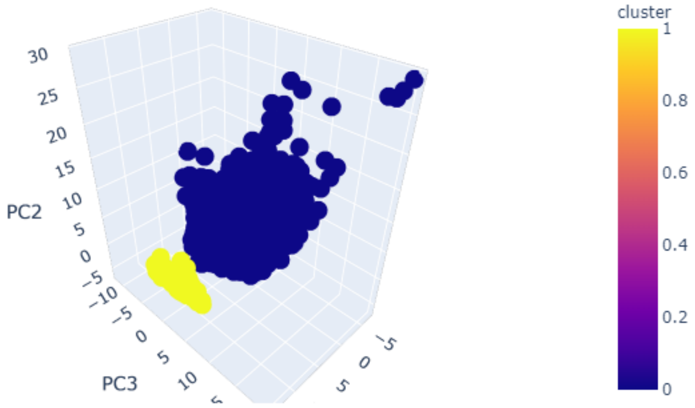
We stop the iteration process when we find that there is no change in centroid values or it reaches the maximum number of iterations given by the user. This method is used for quantitative variables which represent attributes or characteristics of points.

In this work, we seek the possible grouping of articles in different groups based only on quantitative variables. There are 62 characteristics without the main attribute which is the number of shares.

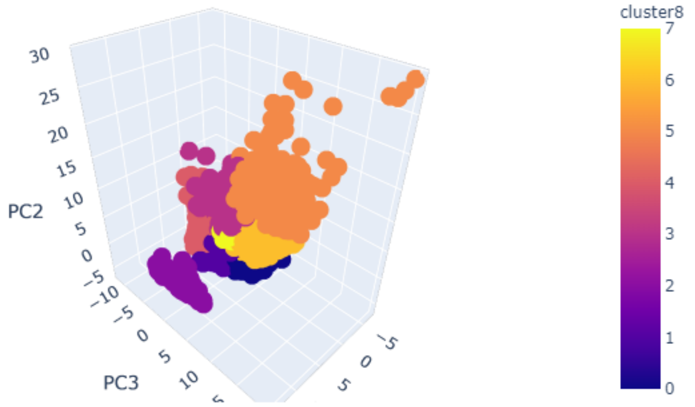
For all these 62 variables, we searched for the number of clusters using the Silhouette metric. Indeed, this indicator measures the performance of groupings. A high value of this indicator implies minimum distances between individuals in the same group and maximum distances between groups. To do this, we vary the number of clusters from 2 to 12 and measure the performance of each way to group. This result is as follows :



We can easily notice that the optimal number of clusters is 2. And for graphical representation, we proceed to the reduction of the dimensionality by making the principal components analysis. Three components are considered. The graphical representation of this grouping is as follows :



This representation clearly shows that the number of 2 clusters seems to group our observations well. We did have two groups which are well separated. To convince ourselves better, we sit down with the default parameters of the scikit-learn package where the number of clusters is 8. The result is as follows :



We note that the grouping into 8 clusters seems less to separate the groups. We note that some groups may well be pooled. Now, let's analyze each of our two clusters based on a few characteristics including the main variable (the measure of popularity)

	shares									num_hrefs	n_tokens_content	num_imgs	num_videos
	count	**mean**	**std**	**min**	**25%**	**50%**	**75%**	**max**	**count**	**mean**			
0	34133.0	1669,52	1100,72	1.0	903.0	1300.0	2100.0	5500.0	34133	10,89	562,33	4,35	1,20
1	970.0	1758,27	1184,63	4.0	906.25	1300.0	2200.0	5500.0	970	0,00	0,00	3,75	0,80

This table shows that the two groups are respectively represented by 34,133 (cluster 0) and 970 articles. We note that, on average, the articles of group 1 have no words in their content, contain fewer links, fewer images and fewer videos. Regarding the number of shares, articles in cluster 0 have a lower mean and lower variance with a minimum value of 1 share versus 4 at the level of the articles of the cluster 1. The two groups have the same median which is the same in the initial base.

These differences between groups are noticeable through other characteristics such as the number of keywords. However, a crossover with the categories of articles does not give us such information as shown in the following table :

Popularity	Clusters	kw_avg_avg
Popular	0	3185.42
	1	4155.65
Unpopular	0	2882.92
	1	4030.42

The table shows that each of our initial groups of articles (popular and unpopular) contain articles from the two clusters with the same characteristic differences. We also note that the articles in cluster 0 of the group of unpopular articles contain fewer keywords on average than those of the popular group, while we have the reverse in cluster 1.

We recall that the clustering did not take into account all the variables or characteristics of the articles, but it gives us decision tools to consider two groups of articles for the rest of this work.

3 The methods and the results

In this section, we will first try to present the different estimation methods. Next, we will try to present the different estimation results. For the purpose of predicting the popular and non-popular classes, and given that we have a classification problem. We are considering a set of methods through which we will seek to find the model with more accuracy.

3.1 Logistic regression

The first model that comes to mind during a classification problem is logistic regression. This latter, also called binomial regression, is a regression model where the variable to be explained is binary. It is based on the assumption that the logit of probability is a linear function of the different explanatory variables.

The logit of probability is nothing more than a ratio between the probability of popularity and the probability of non-popularity of the article : $\mathbf{log}\left(\frac{\mathbf{P(Y=1|X)}}{\mathbf{P(Y=0|X)}}\right)$

Let $X = (X_1, X_2, ..., X_n)$ be a set of explanatory variables which represent characteristics of the article. The model looks like this :

$$\mathbf{log}\left(\frac{\mathbf{P(Y=1|X)}}{\mathbf{P(Y=0|X)}}\right) = \beta_\mathbf{1}\mathbf{X_1} + \beta_\mathbf{2}\mathbf{X_2} + ... + \beta_\mathbf{n}\mathbf{X_n}$$

Given that the number of variables is important, and that the diagnosis of the multicollinearity problem through the correlation matrix has revealed its presence, we are considering information reduction techniques which will seek to alleviate this problem. To do this, we compare two dimension reduction techniques namely PCA and Elastic Net.

3.1.1 Logistic Regression based on PCA

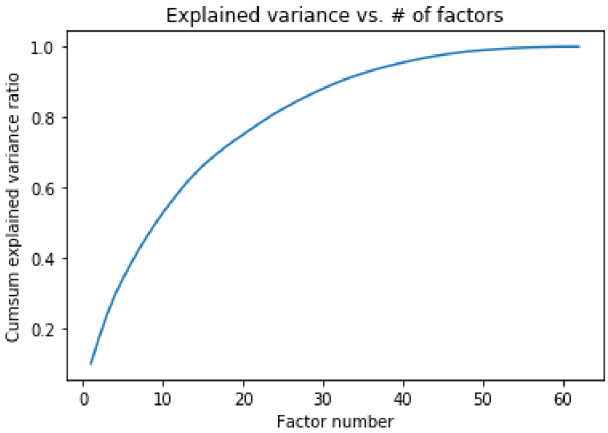
Seeking to reduce the dimensionality of a dataset is becoming more and more necessary because of the multiplication of data, even if it generates a loss in terms of information. Principal Component Analysis (PCA) is an extremely powerful tool for synthesizing information that is very useful when there is a large amount of **quantitative data** to process and interpret. It is a factorial analysis, in that it produces factors (or principal axes) which are linear combinations of the initial variables, hierarchical and independent of each other.

To avoid unit differences, variables must be standardized. The following procedure shows how to generate factorial axes based on the starting variables :

1. Standardize the X matrix : $\boldsymbol{X_c}$
2. Compute the covariance matrix : $\boldsymbol{S = \frac{1}{n-1} X_c' X_c}$
3. Calculate the eigenvalues (λ) and eigenvectors (U) of the covariance matrix

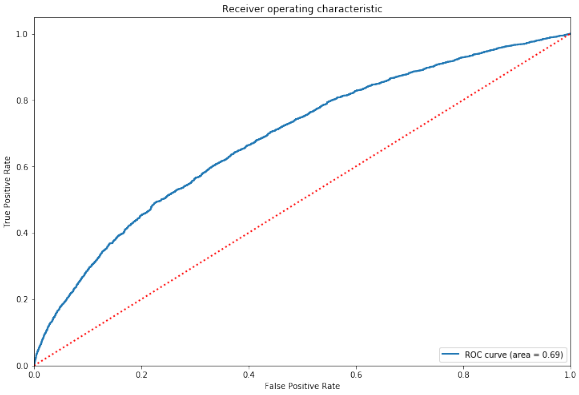
4. Keep only the largest eigenvalues (+ vectors).
5. multiply X_c by the matrix of the eigenvectors (U) to obtain the factorial axes $\boldsymbol{F} = \boldsymbol{X_c U}$

Based on the cumulative sum variance ratio criterion, we taked the first 30 factorial axes, having 88% of informations.



To evaluate our logistic regression, we use as an evaluation criterion the confusion matrix and the ROC curve.These methods are shown in the **appendix A.1 and A.2**, respectively. The accuracy of our model is 63% which seems not satisfactory as a result. The precision is 62%, which means that 62% of the numbers of popular articles have been predicted correctly. Regarding recall, we note that 56% of the number of actual popular articles have been correctly classified.

	Predicted	
Actual	Negative	Positive
Negative	2559	1140
Positive	1431	1891



3.1.2 Logistic regression based on Elastic Net

To take account of multicollinearity and dimensionality in regression, regularized regression is often used through various methods such as Ridge, Lasso (Least Absolute Shrinkage and Selection Operator) and Elastic Net regressions. These regularization methods allow less variance, and therefore better precision at the expense of the introduction of bias. They also each have particular advantages.

Ridge and Lasso regressions control the coefficients by adding the penalty function. In the case of logistic regression, The objective function for the penalized logistic regression uses the negative binomial log-likelihood as shown by *Trevor Hastie and Junyang Qian* in the glmnet package of R that we used for this work. The objective function is therefore the next one :

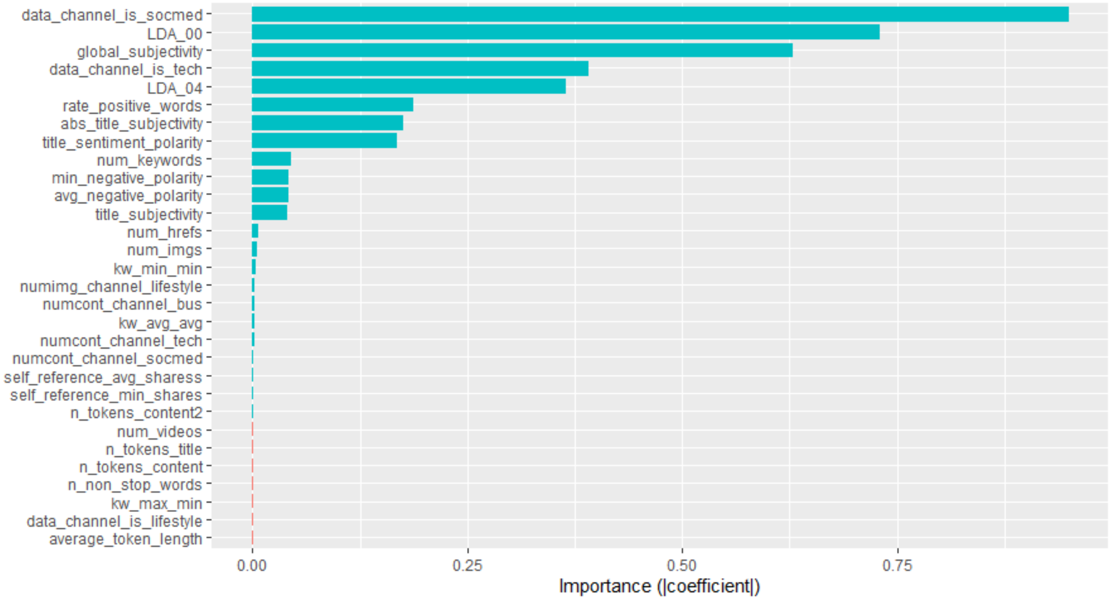
$$\min_{(\beta_0,\beta)\in\mathbb{R}^{p+1}} -\left[\frac{1}{N}\sum_{i=1}^{N} y_i \cdot (\beta_0 + x_i^T\beta) - \log(1 + e^{(\beta_0 + x_i^T\beta)})\right] + \lambda\big[(1-\alpha)||\beta||_2^2/2 + \alpha||\beta||_1\big]$$

Where λ represents the parameter of penalization or shrinkage and α the parameter of mixture between the Ridge and the Lasso which causing the Elastic Net.

Remember that Ridge keeps all the explanatory variables in the predictive model while the Lasso allows you to select the most relevant variables from the correlated variables.

For this work, we chose a fair mix between the Ridge and the Lasso by setting the α value to 0.5 and determined the optimal λ value by cross validation (CV) as shown in the graph in **appendix (A.4)**.In fact, From left to right, we chosed the second λ value which the error is no longer stable and can evolve very quickly. We had retained therefore 50 variables. The estimation results and the evolution of the coefficients are also presented in **the appendix (A.3) & (A.5)**, respectively.

The Elastic Net allows us to delete 22 variables and therefore select 50 relevant variables that allow us to predict the Popularity of articles online. The following graph shows the importance of some selected variables :



The bars in blue indicate the variables which increase the probability that an article is popular, and in red those which negatively influence this probability. For example, we notice that the number of links, the number of images, the number of keywords in an article increases its probability of being popular. On the other hand, characteristics such as the average word length, the number of words in the title and certain fields of publication reduce this probability.

Regarding the predictive power, the following confusion matrix allows us to calculate the precision of the model on a test sample :

	Predicted	
Actual	Negative	Positive
Negative	2636	1061
Positive	1387	1935

This table indicates our model manages to predict the true class of new articles with an accuracy of 65%. The Recall is 71% which indicates that this percentage of the number of actual popular articles is correctly classified. Also, the precision is 65% which means that 65% of the numbers of popular articles have been predicted correctly.

3.1.3 GETS modelling of logistic regression

According to Damian Clarke in ”General-to-specific modeling in Stata”, General to specific (GETS) modelling is a prescriptive way to select a parsimonious and model from a large set of real-world variables and enables the researcher to avoid unnecessary ambiguity or ad hoc decisions. This process involves the definition of a general model that contains all potentially important variables and then, via a series of stepwise statistical tests, the removal of empirically “unimportant” variables to arrive at the proposed specific or final model.

GETS modelling is computed in 3 steps :

1. Formulation of the general unrestricted model (GUM) ;
2. Backwards elimination of insignificant regressors ;
3. Choice of the best terminal model according to a fit criterion.

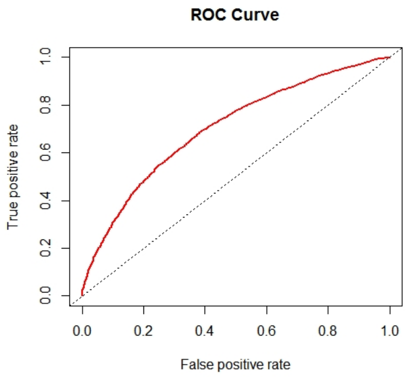
This model has two main properties : First, all the relevant regressors in the starting model (GUM) will be retained in the final model. Second, on average, irrelevant regressors will be retained, where α is the chosen significance level for t-tests.

The table below presents the final choice of the model while minimizing the criterion of Schwarz. The terminal model is taken with 42 variables.

Step	Size of non-zero coefficients	Schwarz Crtierion (SC)
1	44	1.272190
2	44	1.272083
3	48	1.273203
4	42	1.271680
5	45	1.272561
6	**42**	**1.271658**
7	42	1.271677

We will use this model to apply logistic regression. We remark that all variables are significant. We obtain an accuracy of 65%. The area under the curve is up to 70% which indicates that the model is quite satisfactory compared to the previous methods.

	Predicted	
Actual	Negative	Positive
Negative	2646	1051
Positive	1381	1941

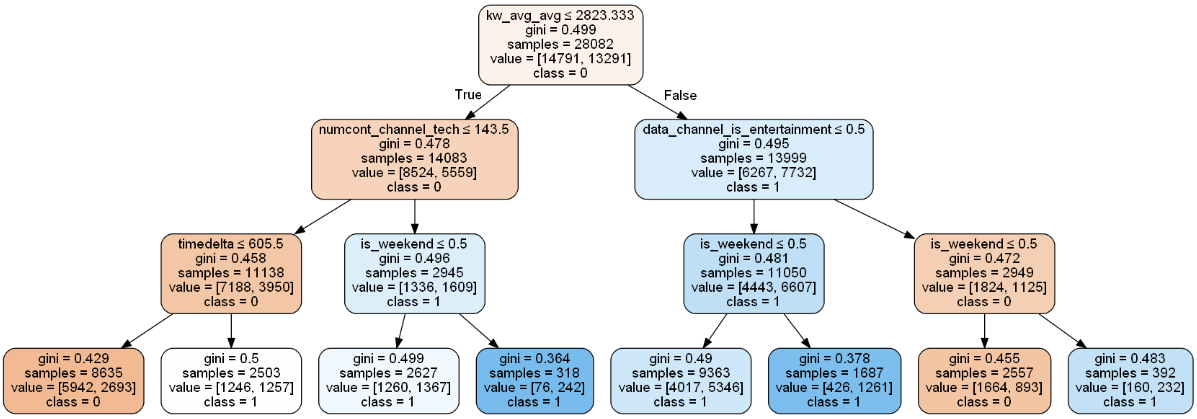


3.2 Classification and Regression Tree (CART)

CART is also a predictive model which helps to find a variable based on other labeled variables. To be more clear the tree models predict the outcome by asking a set of if-else questions. **(Source : www.medium.com)**

Decision tree methods involve recursively partitioning the input space into a number of classes in order to make predictions. This decision support tool allows us to represent in a simple and easy to interpret way.

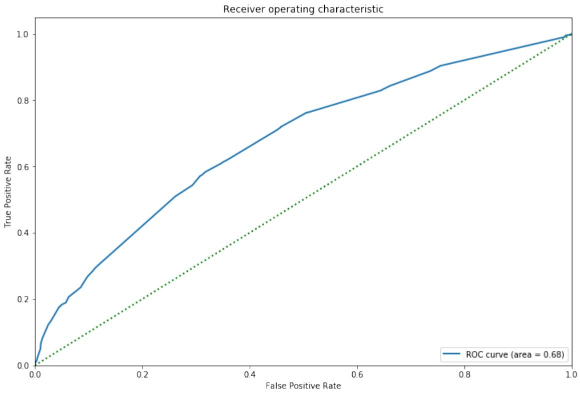
For analysis purposes, we used the Gini criterion to split the decision tree. We note that Gini is based on the probability of misclassifying the observation if we randomly choose its class of belonging. The determination of the depth of the tree was determined using the GridSearchCV function which is a cross validation which makes it possible to determine the optimal parameters.



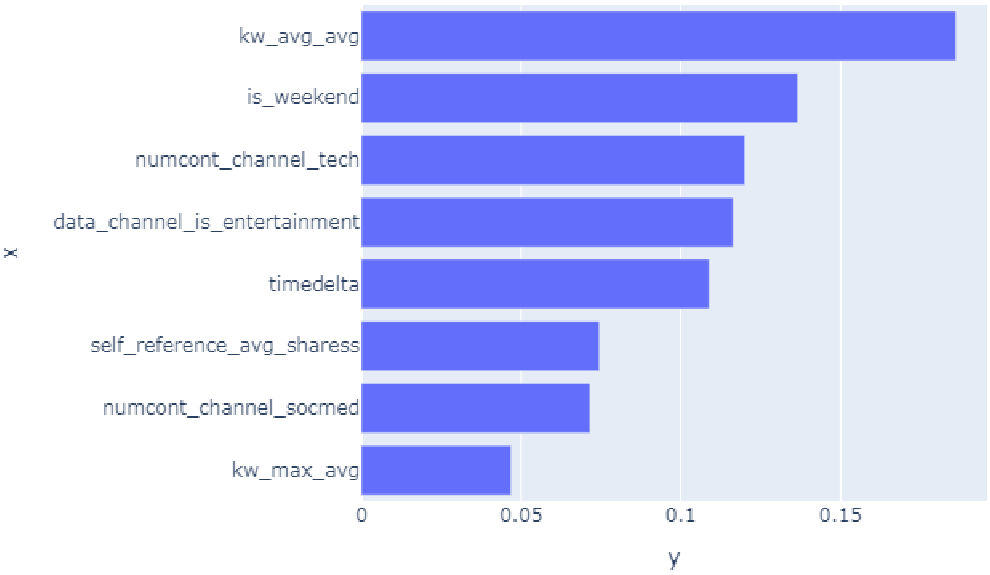
As explained above, the optimal depth of the tree obtained by the GridSearchCV is equal to 6. For a good visualization, we present an overview of the tree with a depth of 3. We notice that the most important variable in the discrimination is at the top of the tree, and is ”average keyword”. It divides the 28,082 observations initially into two groups of 14,791 and 13,291. At the second level of the tree, classification is ensured by variables such as ”numcount_channel_tech” and ”data_channel_is_entertainment”. The process is the same until the final classification.

For most of our nodes, we see a Gini impurity index around 0.5. Which means that in general we have about a 50% chance that our new random values will be misclassified. Our lowest Gini indices are observed at the terminal nodes with misclassification probabilities of 37% and 36%. These values are very high. However, since the optimal depth was 6 we can understand such a result.

	Predicted	
Actual	Negative	Positive
Negative	2545	1154
Positive	1411	1911



The confusion matrix tells us about the accuracy which is 63%.The ROC curve below provides information on the probability of classifying the types of articles. The area under the curve measures the ability to distinguish between the 2 types of articles (Popular and unpopular). This capacity is up to 68% which means that the model is not satisfactory enough.



The graph above presents the 9 most important variables, and at the top the keyword average, followed by the weekend effect variable and the number of words in the content of articles published in the field of technology.

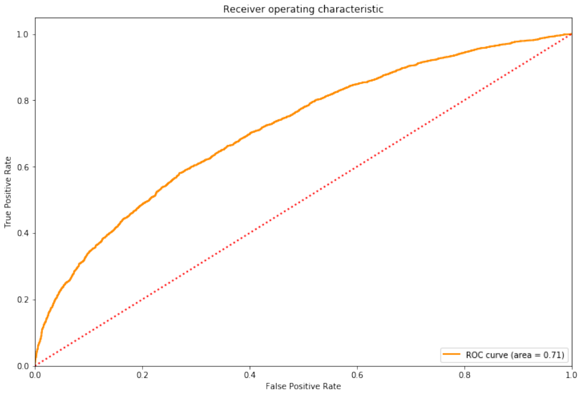
3.3 Bootstrap Aggregating (Bagging)

Given that the CART algorithm is very sensitive to training data, bagging comes to overcome this problem by basing on different training samples constituted by the bootstrap. The principle of the latter is to create ”new samples” by drawing at random from the old sample, with replacement. The estimators thus obtained are used for a majority ”vote”.

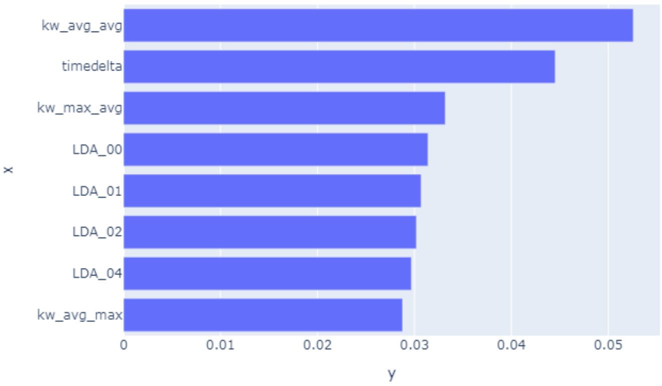
The parameters necessary for the application of bagging are maximum of depth and the size of the subsamples. For the optimal choice of these parameters, we apply cross validation using the GridSearchCV under Python.

We obtain an accuracy of 65%. The area under the curve is up to 71% which indicates that the model is quite satisfactory compared to the previous methods.

	Predicted	
Actual	Negative	Positive
Negative	2544	1155
Positive	1278	2044



The graph below which describes the importance of the variables is roughly similar to that of CART. The most important variables for the bagging algorithm are those related to keyword, the days between the article publication and the dataset acquisition and closeness to top 5 LDA topics.



3.4 Random Forest

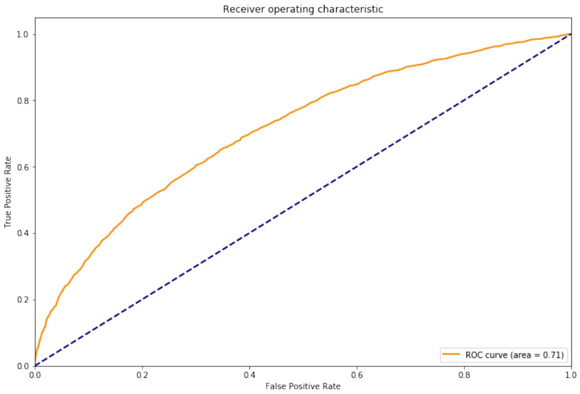
Random Forest is an extension over bagging. It takes one extra step where in addition to taking the random subset of data, it also takes the random selection of features rather than using all features to grow trees.**(Source : www.towardsdatascience.com)**

If we suppose that we have a very strong covariate in the sample, then, most or all trees will have this variable in the root node, and therefore all of the trees will be quite similar to each other. So, by selecting randomly a set of features, we will tend to decorrelate the trees.

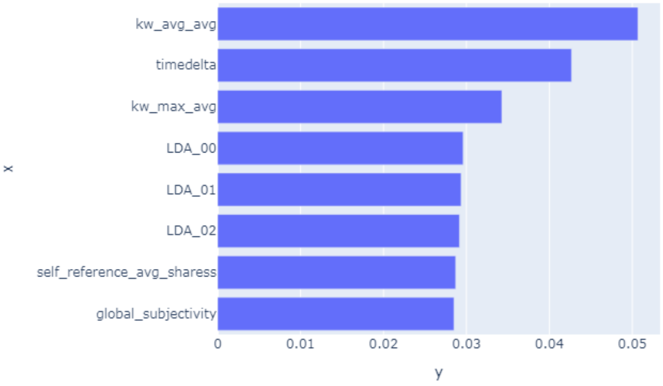
The implementation of random forest requires the choice of the number of variables to be selected randomly, and the depth of the trees. By using the cross validation with GridSearchCV, the number of variables selected is 58 variables.

Generally, there was a small improvement in the level of the Accuracy with 66%. The area under the curve is up to 71% which indicates that the model is quite satisfactory.

	Predicted	
Actual	Negative	Positive
Negative	2567	1132
Positive	1304	2018



The most important variables for the random forest algorithm are those related to keyword, the days between the article publication and the dataset acquisition, closeness to top 5 LDA, Avg. shares of referenced articles in Mashable topics and Text subjectivity.



3.5 K-Nearest Neighbors

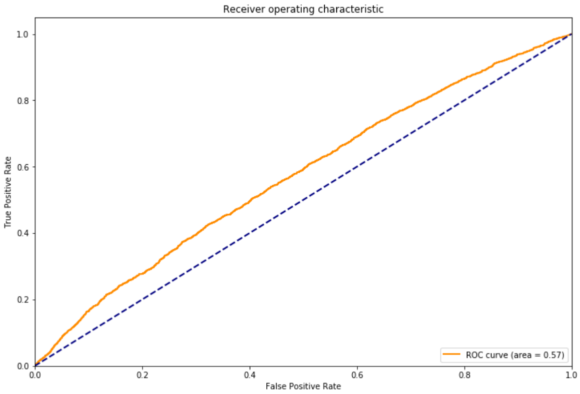
The K-Nearest Neighbrs is a machine learning algorithm that is used in both regression and classification. It is a supervised machine learning method. The idea of this algorithm is to be based on the K neighbors closest to the point studied in order to predict the class or the group to which it belongs. K represents the number of neighbors. This method can be summarized in this sentence : **Tell me who your friends are and i will tell you who you are**.

In the case of classification, the algorithm is based on the class mode of the K nearest neighbors, in order to be able to predict the class of membership. The KNN does not have a statistical model, it is non-parametric. This algorithm, for its operation requires training data. It has the advantage of being simple to implement and intuitive.

It is generally important to optimally determine three parameters in order to be able to train the KNN algorithm in Python software because these influence the results of the model. First, the optimal number of neighbors K and then, the metrics that will be used to calculate the distances between the observations and finally, the weight that will be used for the prediction.

After having initiated a process through the GridSearchCV which allows us to determine the optimal parameters with the Python language, we obtain the optimal neighbor number K = 9, the optimal metric is Manhattan, and the weight is the distance. All other parameters have been taken by default.

	Predicted	
Actual	Negative	Positive
Negative	2428	1271
Positive	1853	1469



Since the accuracy is 56%, and the area under the curve is 57%. We can say that the method tends to assign random predictions.

3.6 Support Vector Machine

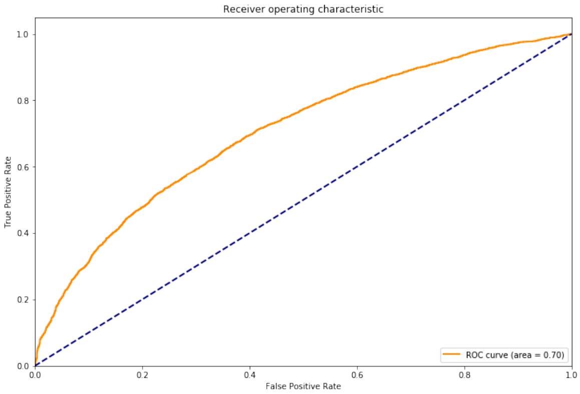
SVM (Vector Machine Support) is a supervised Machine Learning algorithm that can be used in both regression and classification. The idea of this method is to find the hyperplane that optimally separates the data while maximizing the margin. In other words, a hyperplane is the one that maximizes in the case of this study, the distance between the group of popular articles and that of unpopular articles. The hyperplane is defined as the line which allows to classify the data correctly. The vector supports are the values of the dataset that are close to the hyperplane.

Some data is linearly separable while others are linearly non-separable. When a dataset is non-linearly separable, you must use kernel functions. The purpose of the kernel technique is to provide a kernel that will allow data to be separated with non-linear decision limits such as the polynomial kernel and the radial basic function.

To implement the SVM algorithm, the hyperparameters must be determined by making a combination of these parameters. The latter greatly influence the results. Using the Scikit Learn package of Python software, we can cite, the inverse of the regularization parameter C, the choice of the kernel and the coefficient of the gamma kernel.

As part of our project, we were able to determine these parameters optimally through the cross-validation of GridSearchCV, even if we were limited by the algorithm's computation time. In fact, the calculation time was 769.2 Minutes. The parameters found are $C = 1$ and the kernel function rbf with $gamma = 0.01$.

	Predicted	
Actual	Negative	Positive
Negative	2557	1142
Positive	1402	1920



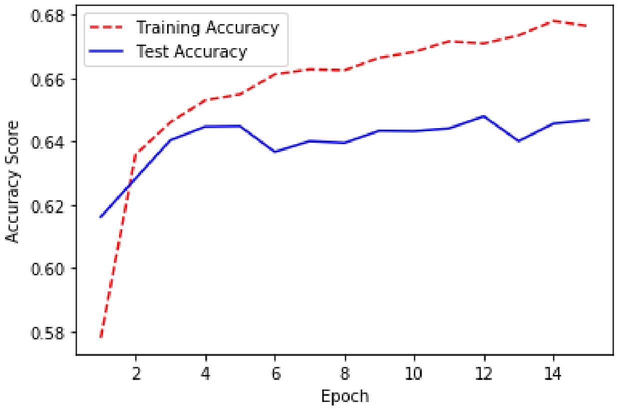
The accuracy of the model is 0.65, therefore the SVM method allows us to correctly predict the popularity of a new article at 65%. The ROC curve indicates that the probability of differentiating two articles, one popular and the other unpopular, is 70%.

3.7 Neural Networks

Neural networks are built on a biological paradigm. The function of neural networks in machine learning is to quickly classify and learn to improve them. For this work, we used **Keras** from the Python software to build a structure.

The neural network is made up of 4 layers including two intermediaries, an input layer and an output layer. The input layer is composed of 76 neurons (the number of explanatory variables), the intermediate layers consist respectively of 64 and 32 neurons with the activation function ”relu” and the output layer comprises a neuron with the sigmoid function. We defined 15 epochs on subsamples of 1000 randomly selected articles. For the network evaluation we use the same test sample as in the other methods. This structure was retained after several tests of different structures with various activation functions.

After training our model, we evaluated it on the test sample which gives us an accuracy rate of 64%. The evolution of accuracy according to training levels is illustrated in the following graph :



We can easily notice that the model learns better and better until the 5th epoch from which the precision is no longer increasing but almost constant. This explains this low accuracy rate. These results could be due to the nature of our data or the definition of the network structure.

4 Comments

4.1 The choice of the best method

The application of these methods allowed us to see the predictive power in several aspects. Overall, this predictive power does not know much difference through these different methods. It hovers around 64% except in the case of KNN where it is 56%.

It should especially be noted that these scores were obtained on the new articles which were not used to build the different models. This sample of articles was randomized. This fits well with our objective which is to predict the popularity of an article before its publication based on its characteristics.

The following table presents the different methods and their predictive power :

Methods	Accuracy (%)	AUC (%)
Logistic Regression based on PCA	63	69
Logistic Regression based on Elastic Net	65	70
GETS modelling of Logistic Regression	65	70
CART	63	68
Bagging	65	71
Random Forest	66	71
K-Nearest Neighbors (KNN)	56	57
SVM	65	70
Neural Networks	64	70

From this table, we can remember that random forest is the best method with a predictive power of 66%. We must also recognize parametric methods for their particular advantages and their close proximity to this value of 66%.

Parametric methods show us remarkable performances. Indeed, in addition to the reduction in dimensionality and the advantages in terms of interpretations that the Elastic Net and GETS regression offer, it is as effective in comparison with the non-parametric methods which are more complex.

The other big surprise in this work is the performance of the neural network which has a predictive power lower than the Elastic Net and GETS regression. Our expectation was that he would have the best performance.

In view of all this, we are tempted to retain the Elastic Net and GETS regression but to remain faithful to our initial objective, we will retain the random forest as the best method.

These results correspond well to those obtained by Fernandes and al., 2015 who also selected random forest as the best method with an accuracy of 67%. Let us recall that their approach is little different from ours with regard to the constitution of test sample, the definition of the popularity of the articles (A threshold of 1400 shares) as well as the number of variables considered in the model. We had created new variables and all our methods were based on the cleaned base which includes 35103 instead of the initial base. We also have the advantage of making parametric methods which give us more visibility on the impact of each of the characteristics of the articles.

The pooling of these different methods allows us to remember that the important characteristics to take into account to predict the popularity of an article before its publication are : The characteristics linked to keywords, closeness to top 5 LDA, characteristics linked to content, the publication area. More particularly by considering the Elastic Net regression, we can retain that the number of links in an article, the number of images, the number of keywords, the field of publication (social media), the LDA_00, the positive word rate , the number of words in the article content, the subjectivity of the title increases the probability that an article will be popular when it is published.

In contrast, the rate of negative words, the number of internal links to the site, the fields of publication (Entertainment and business) compared to the domains, the number of unique words decrease the probability that an article is popular.

4.2 Recommendations

The results imply some recommendations for Mashable. While waiting to improve the predictive model better, Mashable can use the random forest method to predict news before it goes online. This will allow actors, content creators to make any corrections to optimize the number of future shares. Among other things, the process involves :

- Good definition of the days of publication of articles ;
- Optimization on the number of links mentioned in the article ;
- Good orientation of the articles with regard to the fields of publications *(in particular, the fields of social networks and technologies)* ;
- Optimization of the number of videos, images and words of the content of the articles according to each field of publication ;
- Optimization of characteristics related to keywords and content size ;
- Optimization compared to the rate of positive and or negative words ;
- Optimization in relation to the size and sensitivity of the article titles ;
- Adapting articles to current events.

Conclusion

The purpose of this study was to predict the online popularity of new articles before they are published. This field has experienced a notable evolution in recent years with the spread of smartphones and the internet, which to arouse more and more interest including ours. To be able to predict this popularity, we used the Mashable database which contains 39,644 articles for the period from January 2013 to January 2015. It is made up of 61 variables divided into six categories of variables in addition to the dependent variable. We have created 17 new variables using interactions and squared power. This allowed us to have 78 variables. The target variable for the study is the number of shares categorized as popular and unpopular. We used parametric econometric methods and machine learning methods.

Thus, at the level of the parametric regression, we found that on the one hand, the principal component analysis had an accuracy of 63%, and on the other hand, the Elastic Net and GETS selected 50 and 42 respectively of our variables on the 72 of departures with an accuracy of 65%. The ElasticNet and GETS have the greatest precision in parametric methods.

Regarding the machine learning methods, six methods could be estimated. The CART is therefore 63% accurate. Then, Bagging classifies our articles with an accuracy of 65%. The Random Forest has an accuracy of 66%. These three models specify that the keyword average is the most important variable. Furthermore, the KNN method manages to predict our articles with an accuracy of 56%. Finally, the SVM and the neural networks both indicate an accuracy of 65%. We see from the results that, at the level of Machine Learning methods, the KNN method is the one that manages to classify our data the least since it has the lowest precision, and the Random Forest is the one that best classifies the articles with an accuracy of 66%. We therefore retain this method in the context of this study.

This level of precision is not as high, however most of the studies that have used this data have obtained precision around 60 to 70%. Indeed, authors like Fernandes and al (2015) obtained an accuracy of 67% with the Random Forest.

The low level of precision may be due to the failure to take into account certain important variables absent in our database or the complexity of the field of online news. Other methodologies may provide more precision.

A Appendix

A.1 The confusion matrix

The confusion matrix is a matrix that measures the quality of a classification model. Each row corresponds to an actual class, each column corresponds to an estimated class. A variety of measures can be derived from the confusion matrix.

	Predicted	
Actual	Positive	Negative
Positive	TP	FN
Negative	FP	TN

Table 2 – The confusion matrix's presentation

The accuracy is calculated as the number of all correct predictions divided by the total number of the dataset. It is calculated as follows :

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}$$

The precision, called also positive predictive value is calculated as the number of correct positive predictions divided by the total number of positive predictions.

$$Precision = \frac{TP}{TP + FP}$$

The recall, called also the sensitivity, or true positive rate is as the number of correct positive predictions divided by the total actual number of positives.

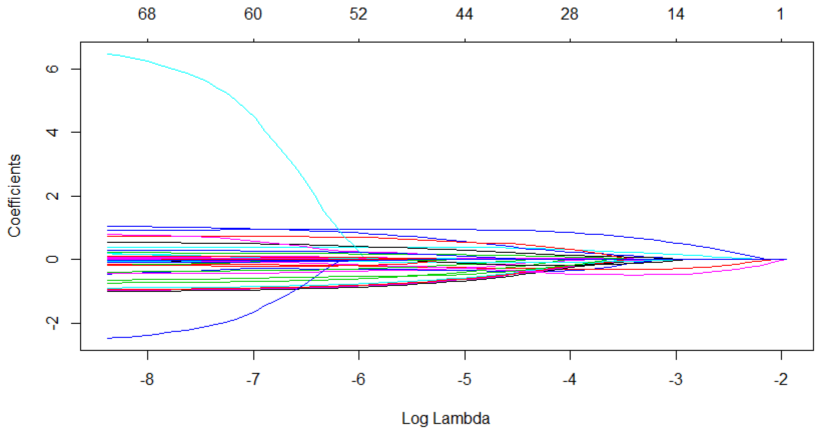
$$Recall = \frac{TP}{TP + FN}$$

A.2 The ROC Curve

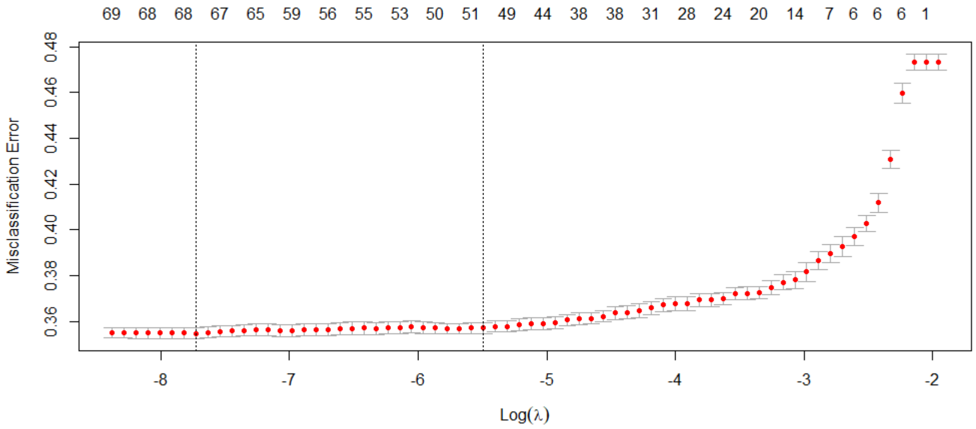
The ROC (receiver operating characteristic) curve is a graphic representation of the relationship between sensitivity (rate of good classification of positives) and specificity (rate of good classification of negatives) for each threshold value considered **(Hanley JA, McNeil BJ ., 1982)**.

This graph relates the proportion of true positives to the proportion of false positives for all of the test threshold values. The more the curve deviates from the first bisector, the more perfect the discrimination. If the curve merges with the first bisector, the model assigns scores at random. The area under the ” AUC ” curve indicates the ability of the model to discriminate between true positives and false positives. The more the AUC increases the more perfect the discrimination.

A.3 The evolution of the coefficients of Elastic Net



A.4 Optimal determination of lambdas by cross validation

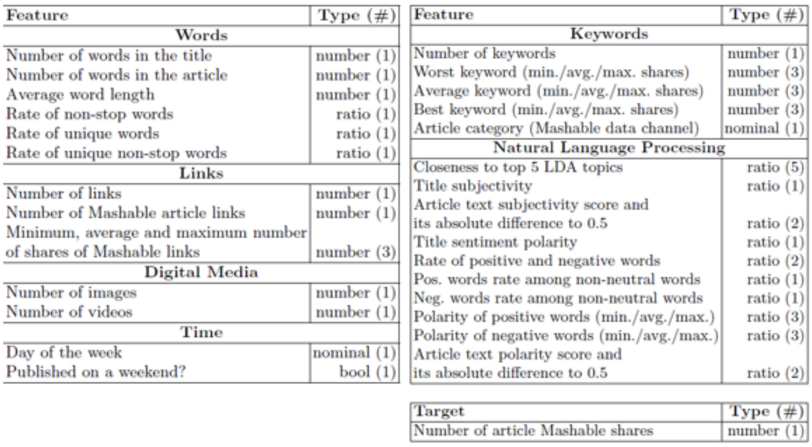


A.5 Estimation result of Elastic Net

ELASTIC NET

Variables	Coefficients
(Intercept)	-1.047242e+00
timedelta	-3.904907e-05
n_tokens_title	.
n_tokens_content	.
n_unique_tokens	-3.024164e-01
n_non_stop_words	.
n_non_stop_unique_tokens	-3.424945e-01
num_hrefs	5.622361e-03
num_self_hrefs	-7.951008e-03
num_imgs	4.545322e-03
num_videos	.
average_token_length	.
num_keywords	4.337695e-02
data_channel_is_lifestyle	.
data_channel_is_entertainment	-2.246754e-01
data_channel_is_bus	-2.384524e-01
data_channel_is_socmed	9.483232e-01
data_channel_is_tech	3.901229e-01
kw_min_min	2.242600e-03
kw_max_min	.
kw_avg_min	-4.144783e-06
kw_min_max	-4.936051e-07
kw_max_max	-5.500866e-08
kw_avg_max	-2.465007e-07
kw_min_avg	.
kw_max_avg	-4.845161e-05
kw_avg_avg	4.097876e-04
self_reference_min_shares	1.978548e-06
self_reference_max_shares	.
self_reference_avg_sharess	2.403213e-06
weekday_is_monday	-6.835953e-01
weekday_is_tuesday	-7.481584e-01
weekday_is_wednesday	-7.820131e-01
weekday_is_thursday	-7.216472e-01
weekday_is_friday	-5.192193e-01
LDA_00	7.284108e-01
LDA_01	-7.374077e-02
LDA_02	-2.148744e-01
LDA_03	.
LDA_04	3.631617e-01
global_subjectivity	6.269351e-01
global_sentiment_polarity	.
global_rate_positive_words	.
global_rate_negative_words	.
rate_positive_words	1.863565e-01
rate_negative_words	-2.823292e-02
avg_positive_polarity	-1.106605e-01
min_positive_polarity	-5.124061e-01
max_positive_polarity	.
avg_negative_polarity	4.021466e-02
min_negative_polarity	4.122419e-02
max_negative_polarity	.
title_subjectivity	3.913008e-02
title_sentiment_polarity	1.664444e-01
abs_title_subjectivity	1.741567e-01
abs_title_sentiment_polarity	.
n_tokens_content2	5.491458e-09
num_hrefs2	.
numvid_channel_lifestyle	-1.328195e-03
numvid_channel_entertainment	-5.049437e-03
numvid_channel_bus	.
numvid_channel_socmed	.
numvid_channel_tech	.
numcont_channel_lifestyle	.
numcont_channel_entertainment	-4.234130e-07
numcont_channel_bus	4.776818e-04
numcont_channel_socmed	9.263086e-05
numcont_channel_tech	3.702432e-04
numimg_channel_lifestyle	1.639896e-03
numimg_channel_entertainment	-3.618969e-03
numimg_channel_bus	.
numimg_channel_socmed	-1.461698e-02
numimg_channel_tech	-8.170022e-03

A.6 Presentation of the database



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)

Source : Fernandes and Al.,2015

Bibliography

[1] Acharya, S., Smith, B., Parnes, P. (1999). Characterizing User Access To Videos On The World Wide Web. Proceedings of SPIE - The International Society for Optical Engineering

[2] Arlitt, M., Williamson. (1997). Internet Web servers : workload characterization and performance implications. IEEE/ACM Transactions on Net-working, 5(5), 631-645.

[3] Chesire, M., Wolman, A., Voelker, G. M., Lev, H. M. (2001). Measurement and Analysis of a Streaming-Media Workload. USENIX Association.

[4] Clarke, Damian, 2014. "General-to-specific modeling in Stata," Stata Journal, StataCorp LP, vol. 14(4).

[5] Fernandes, K., Vinagre, P., Cortez, P. (s.d.). A Proactive Intelligent Decision Support System for Predicting the Popularity of Online News. Springer International, 535-546.

[6] Hastie, T., Tibshirani, R., Friedman, J. (2009). The Elements of Statistical Learning : Data Mining, Inference, and Prediction. Springer Series in Statistics.

[7] Liu, Z. (2017). Statistical Models to Predict Popularity of News Articles on Social Networks. ST.LOUIS : WASHINGTON UNIVERSITY .

[8] Mekouar, S., Zrira, N., Bouyakhf, E.-H. (2017). Popularity Prediction of Videos in YouTube as Case Study : A Regression Analysis Study. Association for Computing Machinery, 4503-4582.

[9] Roy Suman Deb, T. M., Zeng, W., Li, a. S. (2013). Towards Cross-Domain Learning for Social Video Popularity Prediction. IEEE TRANSACTIONS ON MULTIMEDIA, 16(5), 1255-1267.

[10] Tatar, A., Amorim, M. D., Fdida, S., Antoniadis, P. (2014). A survey on predicting the popularity of web content. Journal of Internet Services and Applications, 1-20.

[11] http ://www.medium.com

[12] http ://www.towardsdatascience.com