UCI World Channel Data Report

Prepared for: Kaustav Chakraborty

Prepared by: Jaqueline Ortiz, Kate Roeser, Michael Mardyla, Ridha Alkhabaz, Yulin Pan

Introduction:

From our initial client meeting, our group was presented with the task of analyzing the World Channel subset of the online news popularity data provided by the University of California, Irvine. This dataset contains 39,644 observations, each being an article from Mashable, an international media platform. There were 61 attributes, 2 non-predictive which we used to divide the data into relevant subsets, and there are 58 predictor variables used to determine the popularity or number of clicks, each article received. Our target variable was “shares”, which described the number of shares an article received.

Our client, Kaustav, tasked us with performing an exploratory data analysis on the data to determine which variables are most “relevant” when it comes to determining an article's popularity. Kaustav requested we produce an interpretable model for predicting which variables are important for increasing the number of shares an article will receive. Some of the specific variables he was interested in were sentiment, LDA, media, and days of the week. Off the bat, we did find those variables to be important, but we will get into that further in the next section of our report. Additionally, after looking at the variables, we found that many variables relate to the same characteristic. We found that most of the variables described the rate and number of text, unique words, and keywords, as well as described the number of references, media, and word length used in the article. We also saw that variables described when the article was posted, their closeness to specified topics, as well as information regarding the sentiment and polarity of the article.

● url & time difference (non-predictive) (2)	● (boolean) day of week published (8)
● (boolean) data channel topic (6)	● closeness to LDA topics 0-4 (5)
● num of links/source links in article (2)	● text subjectivity and sentiment polarity (2)
● num of images/videos (2)	● rate of pos/neg words in content/tokens (4)
● avg length of words, num of keywords (2)	● min/max/avg polarity of neg/pos words (6)
● num & rate of words, unique words, non-stop words (5)	● title subjectivity/polarity and abs subjectivity/polarity level (4)
● worst/best/avg keywords min/max/avg shares (9)	● min/max/avg share of referenced article (2)
	● number of shares (**target variable**) (1)

Methods & Results:

When performing our initial data exploration, a few things were apparent. Of the 40,000 articles, only about 8,000 were articles that fell in the World Channel, so we could disregard the rest of the data set. With this initial subset of the data, we performed a standard linear regression using sklearn.linear_model in Python. The resulting R^2 was less than 0.01, so we knew that we would have to perform more modifications to the data. Then, we fit a model using only numerical predictors, i.e. predictors with more than seventeen unique entries. This has improved the R^2 value to 0.26. However, this model did have some problems regarding collinearity between some of the predictors.

First, we merged several categorical variables that indicated which week was the article published into one categorical variable. In addition, we created classes for the popularity of the articles based on which quantile of shares they fell into. One of the other ways we decided to approach the data was splitting up the remaining articles into bins based on their publication date (timedelta). We had 8 bins total: the first included articles published from t = 0 to t = 100, the second from t = 101 to t = 200, and so on, all the way up until the last bin, which included articles published from t = 700 onwards.

This broke up the data into smaller, more manageable chunks. We decided to approach making a final model in a few different ways.

Weekdays vs. Shares

The first variable our group looked into was the day of the week. We have Monday indexed at 0, and Sunday indexed at 6. In the first table below that Saturday has the highest average for the number of shares. This means that on average, articles published on Saturdays get the greatest number of shares.

	Weekday	Average
0	Monday	2,456.05
1	Tuesday	2,220.13
2	Wednesday	1,879.79
3	Thursday	2,394.01
4	Friday	2,228.41
5	Saturday	2,760.20
6	Sunday	2,605.48

Because we thought that days after posting time would greatly affect the number of shares, we created the second table that shows the average of shares based on the 8 different time ranges(bins) with the highest number in each bin highlighted. Even though some other days in the week may have slightly more shares in certain bins, it looks like Saturday still has a relatively higher average with the highest number being 7525 shares in the last bin.

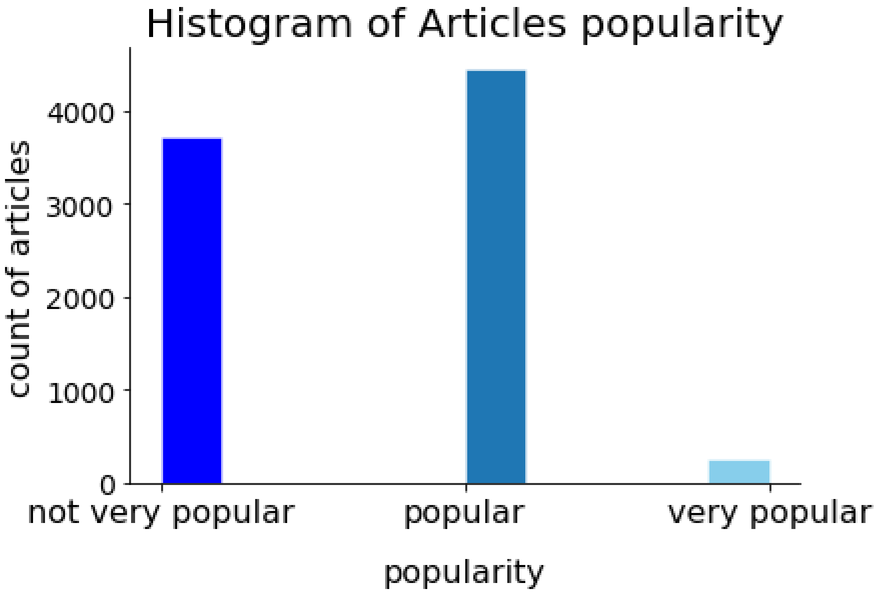
	0	1	2	3	4	5	6	7
Monday	1,753.54	2,944.30	2,874.54	2,041.17	3,587.73	1,728.99	2,725.74	1,664.09
Tuesday	2,294.37	1,931.21	2,469.99	3,340.72	1,749.32	1,541.70	2,222.61	1,499.19
Wednesday	1,659.60	2,109.14	1,817.96	2,182.40	2,211.90	1,333.58	1,879.75	1,572.02
Thursday	2,714.60	2,549.08	2,242.37	2,186.34	2,230.77	1,527.72	3,087.68	1,904.91
Friday	1,993.40	1,807.46	2,487.45	2,982.88	3,203.28	1,285.99	2,380.62	1,727.56
Saturday	2,542.65	1,919.72	3,299.51	2,414.77	3,761.17	3,617.27	3,079.23	7,525.00
Sunday	3,029.17	2,040.53	2,019.90	2,721.79	2,823.91	2,876.16	3,063.64	4,750.00

Our analysis would focus more on the most shared 10% of articles, so we also calculated the Maximum shares for most shared 10% of articles in the last bin and found out Saturday stands out by a lot. As a result, we conclude that articles posted Saturday would likely have more shares.

Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday
8,500	5,500	10,400	9,100	3,800	25,200	15,600

Distribution of Shares:

A major factor that affected our general model is the huge disparity in the volume of shares. We noticed that about half of our articles are shared less than a thousand times from the following histogram, whereas the not very popular category is articles shared less than thousands of times. Also, popular articles are shared between a thousand times to ten thousand times. Hence, it is plausible that because of the huge difference in magnitude between not very popular articles and popular articles, we have problematic general models that cannot predict popularity well.



General Models

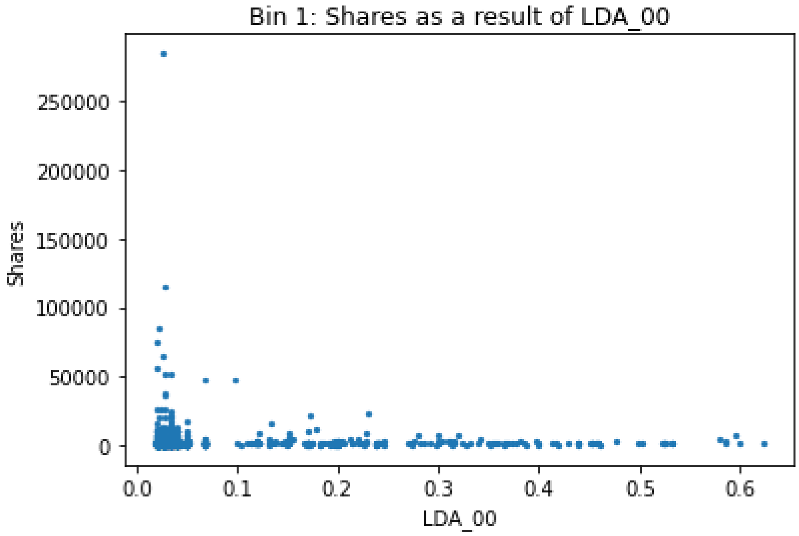
First, we used Statsmodels Python library implementation of Generalized linear models (GLM) and Ordinary Least Squares (OLS). Using GLM and OLS, we fitted all numerical predictors against shares. Their R Squared were 0.26 and 0.37. However, there were apparent collinearity problems. Furthermore, we used the bins above to collect the article in the 90th percentile across these bins. This reduced our data to 808 data points. Then, we used numerical predictors to explain shares. However, this model yielded a small R squared of 0.097. Moreover, we manually selected the twelve most significant predictors to build a new model. The new model yielded a lower R squared of 0.074. Finally, we created a model with the top ten predictors that had an effect on shares from the previous model. This model has an even lower R squared value of 0.032.

With the previous in mind, we decided to pursue a bins-specific model. We found more robust and statistically significant models with that framework. Moreover, it made more sense for business purposes. Meaning, when an author is writing an article, authors usually have a period of popularity. This period is the length of time where their article might get significant exposure from its posting. So, authors usually do not expect their articles to have significant additional shares after a month of publication.

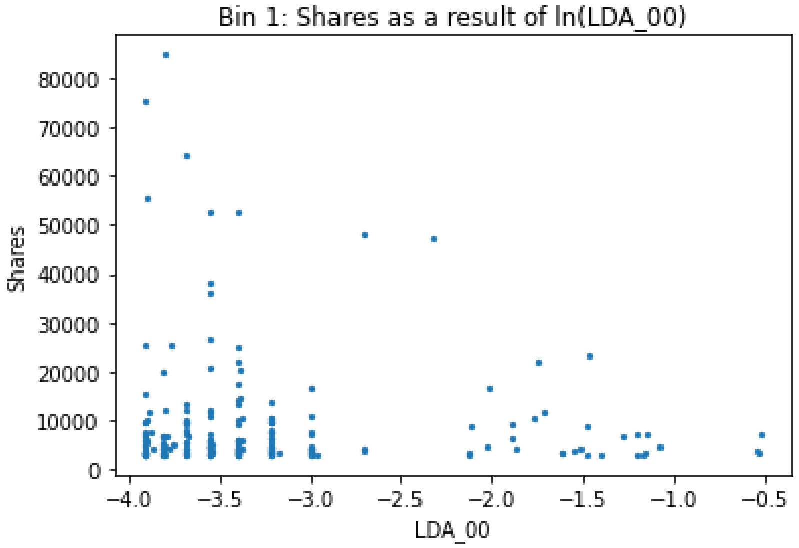
Model 1: Simple Linear Model

During our midpoint meeting, Kaustav suggested that we only look at the articles that performed particularly well. When putting articles into bins, we only added the articles from the 89.5th quantile to 99.5th quantile. By ignoring the top 0.5% of articles, we hoped to minimize the influence of outliers. Before beginning a model, there were some aspects of the data that we wanted to address.

The LDA_00 and LDA_01 variables, which were two that we were told to pay close attention to, yielded scatter plots that had a sharp curve in them.



To address this, we attempted to linearize the data by plotting shares as a result of the natural log of LDA_00 and LDA_01. As a result of the natural log transformation, we got a plot that was significantly more linear looking, especially in the top part of the graph, which we wanted to focus on.



The other main variables we wanted to look at in this model - notably presence of media, sentiment, and weekend - did not have graphs that needed to be linearized, so they were added into the model as is, along with the rest of the variables.

Using sklearn.linear_model in Python again, we performed a simple linear regression on the data in each bin, garnering the following results.

	First week	**Bin 1 Total**	**Bin 4**	**Bin 7**
# articles	52	216	86	68
R^2	0.82	0.199	0.42	0.76
Weekend	-729	1160	-3600	3000
LDA_00	-28	-3430	-3050	1000
LDA_01	-889	-1700	-2250	-280
Sentiment	-5737	-14200	-37800	-4150
Media	-434	2600	-1550	360

We also wanted to look specifically at the first week of articles published in the dataset to see if there was anything that helped an article be immediately successful. In the table, for the rows Weekend, LDA_00, LDA_01, sentiment, and media, the coefficients for the variable within each model is listed. It is important to note that although LDA_00 and LDA_01 have negative coefficients, that means that a lower LDA (which means a more relevant subject matter) increases shares of the article.

Variable Selection

We utilized variable selection in our model buildup with log-transformation for certain variables to help make the model more accurate when predicting the final value (share). We want a model to have the fewest variables, lowest Akaike Information Criterion (AIC) value, and highest adj r-squared value. After using both forward and backward selection, we found out that the backward selection model has fewer variables(14 vs.18) with lower AIC values(371.2617 vs. 377.034) compared to the forward selection model. Even though the forward selection model provides a slightly higher R-squared value(0.9355 vs. 0.9293), it provides a lower adjusted r-squared value(0.7529 vs. 0.8373). We wanted to focus more on the adjusted r-squared value since the adjusted r-squared is a modified version of r-squared that adjusts for predictors that are not significant in a regression model. A lower adjusted r-squared indicates that the additional input variables are not adding value to the model. In this case, the additional 4 variables in the forward selection model do not contribute much to the model, we would like to use the backward selection model as our final model.

Conclusion

For the simple linear model, the following conclusions were drawn.

	First week	Bin 1 Total	Bin 4	Bin 7
Weekend	Did not help	Helped	Did not help	Helped
Low LDA_00	Helped (weak)	Helped	Helped	Did not help
Low LDA_01	Helped	Helped	Helped	Helped (weak)
Negative Sentiment	Helped	Helped	Helped	Helped
Presence of Media	Did not help	Helped	Did not help	Helped

Some conclusions were a little surprising, specifically how having a low LDA did not help articles published over 2.5 years ago. A reason for that could be that the topics corresponding to each LDA could change over time - so the most popular overall topics when the dataset was created may not have been the most popular topics when the old articles were written. Other trends were consistent across all the different date bins, notably the negative sentiment polarity. This means that across different times, negative articles were frequently shared more.

After a more in depth analysis, using variable selection, we have a model with an adjusted R-squared value of 0.8373 and a p-value of 0.0004374. The formula is “Shares = global_subjectivity + what_weekday + LDA_00 + LDA_01 + LDA_02 + log(LDA_03) + log(LDA_04) + num_keywords” with some important variable coefficients of LDA_00(-67720), LDA_02(-71607.6), num_keywords(-2482.7) and what_weekday5(-16470.0).

Model Exploration

Once we found a predictive model, we wanted to ensure that our model was the most relevant. We did this by comparing our final model with different recommended models. Using a for loop, we created selection models between shares and every singular attribute. After collecting all the models adjusted R^2, we found that the num_self_refs attribute produced the highest R^2 at 0.4234. As previously stated, our model produced a backward selected adjusted R^2 value of 0.8373. Thus we know that our model showed a stronger relationship between the article attribute and shares. We also looked at the relationship between the client-specified variables of interest and shares. After finding that the attributes sentiment, day of the week posted, and media did not strongly influence the shares variable, we moved on to our last model of interest. Our final model included models with the attributes media and day of the week, so we decided to add the last variable of interest, media, into our final model. We found that it had a higher forward selection adj R^2, but the backward selection adj R^2 was the same as the final model. Thus we know that adding media to the model did not improve the relationship, so to reduce the possibility of overfitting, we left our final model the same. Thus we are confident that our model does the best possible job at predicting article characteristics that increase the number of shares that an article receives.

Discussion and Conclusions:

After finding our conclusions, we are able to have recommendations for our client to improve their business. First, we recommend that articles should be posted on Saturdays, as they received more shares than articles posted any other day of the week. Next we would recommend that they post articles with a topic closely related to LDA_00 instead of other topics. Lastly, we would also recommend that they post articles with negative sentiment polarity to achieve higher share numbers. We are interested to see how the clients' shares increase after taking our advice.

We discussed areas of improvement or additional information that would have helped us create a more accurate model. We concluded that having articles more evenly distributed by posting date would be beneficial. We found that after dividing the articles by timedelta, the smallest subset of data had 27 observations while the average was 107. We also think that considering the length of period since posted as a constraint, we could predict popularity much better. We would want to know how long the client wants to receive shares, whether it be right when posted, or to receive shares years after posting, to have relevant subsets of data to test and yield accurate results.

Appendix & Code:

Attached here is the code representing the simple linear model. The code is commented to help understand what is happening.

Here is code for data cleaning and general models.

Here is the code used to perform variable selection.