Online News Popularity Data

Julian Murillo, Valeria Park
Chapman University, MGSC 310 – Statistical Models in Business
Final Project

I. Data Introduction

For our final project, we used the Online News Popularity dataset from the UCI Machine Learning Repository. The data was obtained from Mashable.com, a global, multi-platform media and entertainment company that is powered by its own proprietary technology. The dataset consists of 39,644 articles and 61 variables, with 58 predictive attributes, 2 non-predictive, and 1 target variable. The target variable is the amount of shares from each article because increasing online exposure increases the amount of traffic on the website and therefore, increases revenue for the company. Since Mashable is not as well known as other media platforms, this analysis can help the company understand how to reach a bigger audience.

In order to understand how to optimize the shares, we originally looked at all 58 predictive attributes: number of words in the title, number of words in the content, rate of unique words in the content, rate of non-stop words in the content, rate of unique non-stop words in the content, number of links, number of links to other articles published by Mashable, number of images, number of videos, average length of the words in the content, number of keywords in the metadata, data channel (dummies), best keyword, worst keyword, average keyword, shares of referenced articles in Mashable, article publish day (dummies), closeness to LDA topic 0-4, text subjectivity, text sentiment polarity, rate of positive/negative words, polarity of positive/negative words, title subjectivity, title polarity, absolute subjectivity level, absolute polarity level.

II. Initial Filtering

Since the dataset was already clean, we did not have to create any new variables or dummy variables. However, after running several different models, we noticed there was a major

problem with the target variable. In fact, when we ran a linear regression model, the adjusted R-squared was a mere 2.17%. In addition, we implemented more robust methods like a ridge, lasso, and elastic net model. The ridge model yielded very small coefficients, while the lasso and elastic net models did not select any of the 58 potential predictor variables. In other words, all three models concluded that none of the variables were important predictors. Thus, we took a look at the target variable, and the summary statistics showed a large variance, standard deviation, and that the target was not normally distributed, meaning that the amount of shares varied drastically for each article (Fig. 1). Some articles had less than 5 shares, while others had more than 500,000 shares. Given the extreme variation in the dataset, we created a popular article variable that classified articles as popular or not popular, depending on whether it had more or less shares than the median number of shares (1,400).

Moreover, we realized there were too many predictive attributes, so we filtered out insignificant variables through several steps. First, we looked at the variance inflation factor (VIF) to identify any multicollinearity problems. We removed the highly correlated variables with a VIF of over 10. Then, we ran a lasso model and elastic net model to select the best predictors for our binary variable "popular_article". We cross-referenced the selected variables and chose the ones with the most impact on the amount of shares. The selected variables were the number of links, number of links to other articles published by Mashable, number of keywords, data channels, article publish days, global subjectivity, minimum of positive words, title subjectivity, title sentiment polarity, absolute title subjectivity, and closeness to LDA topics. Reducing the amount of predictors allows for a parsimonious model that is more simple and accurate.

III. Summary Statistics

The summary statistics of the variables were not particularly compelling, especially because half of them were dummy variables with a minimum of 0 and a maximum of 1 (Fig. 1). As mentioned before, the amount of shares undoubtedly showed that the data was not normally distributed due to several extreme outliers. Another interesting variable was the number of external links, which

had a minimum of 0, a maximum of 304, and a median of 8. These statistics indicate that there are outliers when it comes to the number of external links as well, and that most of the articles on average, had around 10 external links. The summary statistics for the number of Mashable links also demonstrate several outliers, with a minimum of 0, a maximum of 116, a median of 3, and a mean of 3.29. Overall, the summary statistics of the variables confirm, once again, that the dataset is not normally distributed.

IV. Predictive Models and Results

After creating a new dataset with all 21 relevant variables, we found predictions using a logistic model and random forest model. We specifically chose these 2 models because they perform the best with classification problems.

Logistic Regression (cut-off: .5)

Once we developed a logistic model and generated probability scores we decided that a probability cut-off of .5 was appropriate given the distribution of the scores, with the median being .47, the max being .97 and the min being .07. The logistic model we developed for the training set yielded a False Positive Rate of 36.53%, a True Positive Rate of 61.93%, a True Negative Rate of 63.46% and an overall Accuracy of 62.71%. As for the test set, the same logistic model (derived from the training set) yielded a False Positive Rate of 36.24%, a True Positive Rate of 62.47%, a True Negative Rate of 63.75% and an overall Accuracy of 63.12%. From these figures we were able to conclude that our logistic model was slightly overfit, but to truly evaluate the predictive power of the model we needed to perform further robustness checks.

Random Forest Model

For the random forest model, we used 100 trees, as indicated by the plot as the optimal number where the error rate plateaus. The predictions for the training set yielded a False Positive Rate of 49.49%, a True Positive Rate of 48.89%, a True Negative Rate of 50.51%, and an overall Accuracy of 49.70%. The predictions for the testing set yielded a False Positive Rate of 34.75%,

a True Positive Rate of 62.12%, a True Negative Rate of 65.25%, and an overall Accuracy of 63.69%. Evidently, these numbers indicate that the random forest model was overfit and that the random forest model did not perform significantly better than the logistic model.

V. Robustness Checks

Logistic Regression

After running our logistic regression model, generating predictions, generating a confusion matrix, and calculating Sensitivity, Specificity, and Accuracy for both the test and training sets, we wanted to perform a few robustness checks for our model. To do this, we constructed a Receiver Operator Curve (R.O.C plot) for our test and training sets which illustrate the effect that changing the cut-off probability has on the True Positive and False Positive rates. Further, from these R.O.C plot we calculated the Area Under the Curve (AUC). The training R.O.C yielded an AUC of .6722 and the test R.O.C yielded an AUC of .6788. In other academic literature, a general rule of thumb is that an AUC above .7 is considered acceptable. Seeing as our AUC for both the test and training sets were just under this threshold, the predictive power of our model is relatively weak, being only about 17.88% better than chance.

Random Forest Model

For the random forest model, the ROC for the training set yielded an AUC of .4942, meaning that it is no better than chance, and the ROC for the testing set yielded an AUC of 0.6814, which is still under the threshold. Once again, these different results for the test and training sets reveal how overfit the random forest model is and prove the inefficacy of the model.

VI. Revisions

After presenting in class, we decided to try the recommended revisions, hoping to get better results. Following the same steps as before to filter out irrelevant predictors, we ran OLS and logistic regression models using the top 10% of the articles that had the most amount of shares

(3,964 total obs.). The selected variables for this subset were the number of words in the titles, number of words in the content, number of links, number of links to other articles published by Mashable, number of keywords, data channels, number of keywords in the metadata, global subjectivity, rate of positive words, polarity of positive words, polarity of negative words, title subjectivity, title sentiment polarity, and title subjectivity. The summary statistics for the target variable were better with a minimum of 6,200, a maximum of 843,300, a standard deviation of 33,049, a median of 10,800, and a mean of 18,295. Although this was better than the original dataset with all the articles, this still shows how irregularly distributed the data is.

OLS Regression

The regular OLS regression model resulted in an adjusted R-squared of 1.26% for the training set and 0.37% for the testing set, which was very low, despite only looking at the top 10% of the articles . When running the predictions, the RMSE was 28,267 for the training set and 43,176 for the testing set. In addition, we implemented the OLS regression model using k-folds. The adjusted R-squared was 1.04% for the training set and 1.22% for the testing set. The RMSE was 28,450 for the training set and 44,251 for the testing set. Clearly the results from the linear regression were not significantly better than before. Therefore, we tried running a logistic regression model by classifying the top 10% of the variables.

Logistic Regression (cut-off: .5)

For our revised logistic regression model, the variable selection and modeling processes were identical, the only difference being the data frame used. From the logistic model generated from this data, the training set yielded a True Positive rate of 57.89% , a True Negative rate of 57.62%, a false positive rate of 42.38% and an overall accuracy of 57.75%. The test set yielded a True Positive rate of 58.49% , a True Negative rate of 59.65%, a false positive rate of 40.34% and an overall accuracy of 59.031%. As for model performance, the training AUC was 0.5981 and the test AUC was 0.6266. Once again even after the filtering process reiterated (using shrinkage methods Lasso and Elastic Net) our revised model to predict the top 10% of popular articles was over-fit and performed worse than our original logistic model. This model being

only 9.81% better than chance for the training set and 12.66% better than chance for the test set. In summary, our revised logistic model performed worse than our original logistic model which was created using all observations instead of the top 10%. The implications of this are that the difficulty in predicting article popularity stem from the variation in shares per article, as the distribution of shares for even the 90th percentile of articles had immense variation as well/unpredictability.

VII. Conclusion

Overall, this analysis showed the difficulty in predicting the popularity of an article based on the amount of shares. It seems to be rather subjective when it comes to readers determining whether or not they want to share an article. Also, this dataset was not normally distributed and had multiple outliers across various attributes, which made the analysis even more difficult. Having access to the number of views per article, the timespan of how long the articles have been posted for, and the demographics of the viewers would have been more helpful. Perhaps even changing the target variable from the number of shares to the number of views would be easier to predict.

Nonetheless, we were still able to obtain useful information regarding the relationship between the amount of shares and other predictive attributes. Going based off of our best model (initial logistic), in a real world application, it would probably be more costly to suffer losses in revenue from a false positive rather than a false negative. Producing and funding an article which you believed to be popular, turning out that it wasn't would result in not only a loss in regard to expenses, but a loss in regard to a potential investment that could have been made in a different article. If we were to make any recommendations based off of our best model, we would suggest including five or more keywords in your article, publish your article on a friday (92% greater likelihood), and cover topics such as social media (105% greater likelihood) and technology (44% greater likelihood).