CMSC 35300 - Graduate Project

Sergio Olalla, Michael Wagner, Núria Adell Raventós

December, 2022

1 Introduction

The goal of this project is to test whether Elastic Net regularization might improve current regularization methods used in text classification. We conduct this analysis in the context of identifying potentially unreliable news articles. As a text classification problem, feature engineering includes several natural language processing techniques that are outlined in the methodology. We focus on logistic regression, which has performed well in other text classification problems. Our work compares elastic net with ridge, lasso regularization methods, and explores potential variation between these methods with respect to our high-dimensional data.

2 Literature Review

Logistic regression methods have shown high performance in text classification problems amongst different statistical and machine learning (ML) models (Hassan, S. U., Ahamed, J., Ahmad, K., 2022; Suneera, C. M. and Prakash, J., 2020). In the context of fake news detection, Katsaros, D., Stavropoulos, G., Papakostas, D. (2019) analyze the performance of a number of ML methods. Among the regression-based methods available, only lasso regularized regression is tested. While this is a sensible choice, lasso presents some limitations. Lasso regularization increases sparsity, reducing the high dimensionality problem. However, it can be problematic when some features are highly correlated. Traditionally, ridge regression was used to deal with this issue in problems where the number of features was smaller than the number of observations ($p < n$). However, the performance of ridge models degrades when $p > n$. Elastic net regularization provides a compromise between the lasso and ridge penalties. Marafino, B., Boscardin, J.B., and Dudley, R.A. successfully implement elastic net regularization for feature selection in biomedical text classification. Following their methodology, we test elastic net regularization in a fake news detection classifier with different ratios of L1 and L2 regularizations.

3 Data

We work with a dataset that contains 20.800 news articles which are classified as reliable or not. It includes the following fields: article id, title, author, text, and label (1 if reliable, 0 if not). For the purpose of this project, we use only the text and label columns. The dataset is balanced with respect to the labels.

Retrieved from: www.kaggle.com/competitions/fake-news/data

4 Methodology

This project builds a binary classification model to determine whether an article is fake news. We build the model predictors from the article text, and implement the following text pre-processing. We remove punctuation, numbers, and stop-words, and perform stemming. Following, we implement tf-idf (term frequency-inverse document frequency) word vectorization, which will correspond to our model predictors (see equations below). To reduce the volume of terms and improve efficiency, we test different vectorization methods, such as excluding words that appear in 50 - 100% of the documents and words that appear in less than 5 to 30 documents.

$$tf_{t,d} = \frac{\text{number of times term } t \text{ appears in document } d}{\text{total number of terms in document } d}$$

$$idf_t = \log \frac{\text{total number of documents}}{\text{number of documents with term } t}$$

$$tf - idf_{t,d} = tf_{t,d} \times idf_t$$

We have a binary outcome to predict, so we implement a logistic regression model for binary classification. We compare three different regularization techniques: ridge, lasso, and elastic net. We build the different models from the ground up using Numpy. For each of these, we compare different shrinkage parameters λ, and for the elastic net, we also test different ratios of $L1 - L2$ penalties (α). Note that $\alpha = 1$ for ridge regularization and $\alpha = 0$ for lasso regularization.

Logistic function:

$$\sigma(x) = \frac{1}{1 + e^{-X\hat{w}}}$$

Weights with Lasso regularization:

$$\hat{w} = \operatorname{argmin} x_{\mathbf{w}} \|\mathbf{y} - \sigma(X\mathbf{w})\|_2^2 + \lambda\|\mathbf{w}\|_1$$

Weights with ridge regularization:

$$\hat{w} = \operatorname{argmin} x_{\mathbf{w}} \|\mathbf{y} - \sigma(X\mathbf{w})\|_2^2 + \frac{1}{2}\lambda\|\mathbf{w}\|_2^2$$

Weights with elastic net regularization:

$$\hat{w} = \operatorname{argmin} x_{\mathbf{w}} \|\mathbf{y} - \sigma(X\mathbf{w})\|_2^2 + \lambda\alpha\|\mathbf{w}\|_1 + \frac{1}{2}\lambda(1-\alpha)\|\mathbf{w}\|_2^2$$

We split the data 80-20% into train and test sets. To find the optimal λ and α, we implement 3-fold cross-validation on the train data. We calculate the log loss across the folds for each relevant combination of λ and α and select the parameters associated with the lowest log loss.

Log loss for the ith point:

$$l_i = \begin{cases} -\ln(\sigma(x)), & \text{if } y_i = 1 \\ -\ln(1 - \sigma(x)), & \text{if } y_i = 0 \end{cases}$$

which can be rewritten as:

$$l_i = y_i(-\ln(\sigma(x))) + (1 - y_i)(1 - \ln(\sigma(x)))$$

Given the large volume of features (corresponding to words in the article text), we train the model by means of stochastic gradient descent (SGD). See the equations for the gradient and weight update step below. We initialize each weight with a random value between 0 and 1, and we test multiple step sizes and a range of the number of epochs to find the optimal training process.

Elastic net gradient:

$$\nabla_w l_i = -(y_i - \sigma(x_i))x_i + sign(w)\alpha\lambda + (1 - \alpha)\lambda w$$

Note that L2=0 for lasso regularization gradient and L1=0 for ridge regularization gradient.

Weights update:

$$w^{k+1} = w^k - \tau \nabla_w l_i(w^k)$$

Finally, based on the calculated log losses, we use the winning models with ridge, lasso, and elastic net regularization to classify the test data and compare the results. The final evaluation is based on the accuracy of the predicted values.

5 Results

The results shown in this section are the result of a model built with maximum $df = 0.5$ and minimum $df = 30$ in the tf-idf factorization, in other words, we have excluded words that appear in more than 50% of the documents in the train dataset and words that appear in less than 30 documents. Furthermore, the step size is 0.01 and the number of epochs is 10.000. Note that due to computational power constraints, these numbers are chosen after testing limited ranges of parameters.

Figure 1 shows the average validation set log loss from the 3-fold cross validation for each λ and α tested. Lasso regularization and elastic net with a small alpha are the models with the smallest log loss, while ridge regularization has the highest. However, these difference decreases as λ becomes smaller. In terms of the λ value, a λ smaller than 1e-04 leads to a higher log loss, below this value, the log loss does not vary significantly.

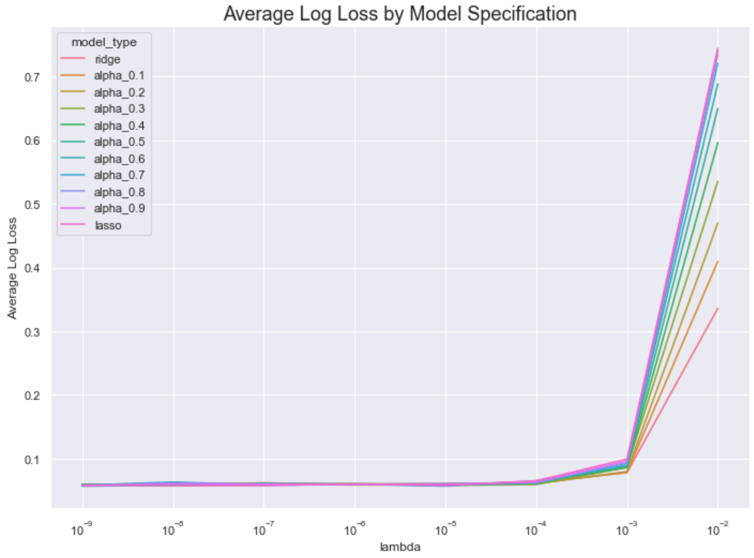

Figure 1: Validation Average Log Loss by Model Specification

Following, we take the λ and α from the best performing models in the k-fold cross validation process, we re-train the models with these parameters and all the train data, and implement the classification for the train and test sets. See the optimal parameter values and the results below:

	λ	α	Train Accuracy	Train Test
Ridge	1e-09	0	0.8244	0.5631
Lasso	1e-08	1	0.8272	0.56
Elastic Net	1e-09	0.6	0.8172	0.5580

We find the optimal λ's are similar across models and the optimal α for the elastic net is 0.60, which corresponds to a slightly higher L1 to L2 regularization ratio. When looking at the accuracy both for the train and test sets, both ridge and lasso regularizations perform very similarly. The accuracy of elastic net is slightly lower than compared to using the individual regularization parameters. The results still suggest some overfitting, implying that increasing the regularization parameters could potentially improve the test results. The following section discusses some of the limitations of this project that we believe, when solved, could improve these results.

6 Limitations

Given the computational power constraints of running this analysis in our local machines and the high volume of features in the dataset, we have evaluated limited ranges for the parameters. A more robust analysis would study how the performance of our models is sensitive to the chosen parameters, including a higher range of the minimum and maximum *df* in the tf-idf vectorization, and more combinations of λ and α values.

Similarly, we tested a limited range of gradient descent step sizes and number of epochs. Optimal λ's and α's were selected by evaluating the average log loss of the validation data from cross-validation with a model built with a number of epochs that was lower than the number of features. Increasing this number would likely find that currently selected α's and λ's are suboptimal. Additional work could improve this analysis by increasing the number of epochs, and conducting a rigorous step-size sensitivity analysis.

Furthermore, we have limited the cross-validation process to 3 folds given our computational constraints. Increasing the number of folds would increase the observations in each training set, which would likely improve the choice of optimal parameters.

Finally, regarding feature engineering, while tf-idf is widely used for this type of problem, other natural language processing methods may help improve the results, including n-gram information. Overall, we believe this further exploration of the optimal methodology could lead to improved parameter selection and higher test accuracy.

7 Conclusion

We do not find evidence of the more robust performance of the elastic net regularization compared to lasso and ridge regularization in the context of Fake News detection. Our results with training data log losses are aligned with the current literature that shows that L1 regularization appears as a more sensible choice than L2 regularization in the context of text classification. However, we do not see that a compromise between ridge and lasso (elastic net) necessarily improves Lasso performance in training data.

Our results of the testing data show that the three regularization methods present similar accuracy results. Further work could explore how the performance of the evaluated regularization methods might vary among other performance metrics such as sensitivity or specificity.

8 Bibliography

Hassan, S. U., Ahamed, J., Ahmad, K. (2022). Analytics of machine learning-based algorithms for text classification. Sustainable Operations and Computers, 3, 238-248.

Katsaros, D., Stavropoulos, G., Papakostas, D. (2019, October). Which machine learning paradigm for fake news detection?. In 2019 IEEE/WIC/ACM International Conference on Web Intelligence (WI) (pp. 383-387). IEEE.

Marafino, B. J., John Boscardin, W., Adams Dudley, R. (2015). Efficient and sparse feature selection for biomedical text classification via the elastic net: Application to ICU risk stratification from nursing notes. Journal of Biomedical Informatics, 54, 114–120. https://doi.org/10.1016/j.jbi.2015.02.003

Suneera, C. M. and Prakash, J. (2020). Performance Analysis of Machine Learning and Deep Learning Models for Text Classification. IEEE 17th India Council International Conference (INDICON), 2020, pp. 1-6, doi: 10.1109/INDICON49873.2020.9342208.