



# PoliticalLens: An R-shiny Application to Evaluate Political Bias in American News



Mitchell E. Daniels, Jr.  
School of Business

Gabriel Ponsot, Conrad Lin, Caleb Brunton, Umar Yasser, Matthew A. Lanham  
Purdue University, Mitch Daniels School of Business

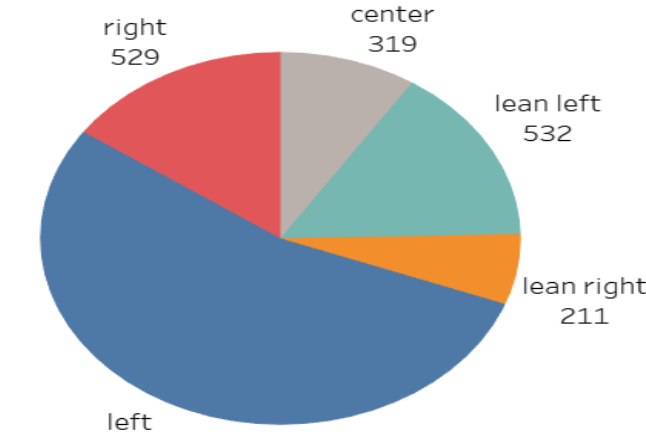
gponsot@purdue.edu; conradsl@purdue.edu; cbrunton@purdue.edu; uyasser@purdue.edu; lanhamm@purdue.edu

## ABSTRACT

This study develops an R-shiny application called PoliticalLens that estimates the political bias from news articles. The motivation for this research is that the public trust in traditional news media has been at an all time low. According to a 2024 Gallup poll, only 31% of Americans expressed a “great deal” or “fair amount” of confidence in the media to report the news “fully, accurately and fairly [1].” News has become so politically polar that we posit there is a significant market for those who would like to have articles reviewed for political neutrality prior to deciding to read them. PoliticalLens is trained on thousands of recent news articles from popular news sources and labeled as politically left, center, or right. Using this training set and subsequent textual analysis, an end user can provide a new article which and be informed if the article is politically center or not.

## INTRODUCTION

Division in the news today can be seen almost everywhere and across every single topic. The version you get depends on the platform that you follow. A common issue is selective reporting – news outlets will focus only on certain stories based on their editorial stance. Another issue is the rampant spread of misinformation, as well as disinformation, across social media. The consequences of this include, but are not limited to, social divisions, political polarization and civic disengagement. Take the issue of gun control as an example. Liberal outlets like The Guardian or HuffPost advocate for gun control and reform whereas conservative media like Fox News or The National Review often stress the importance of the right to bear arms.



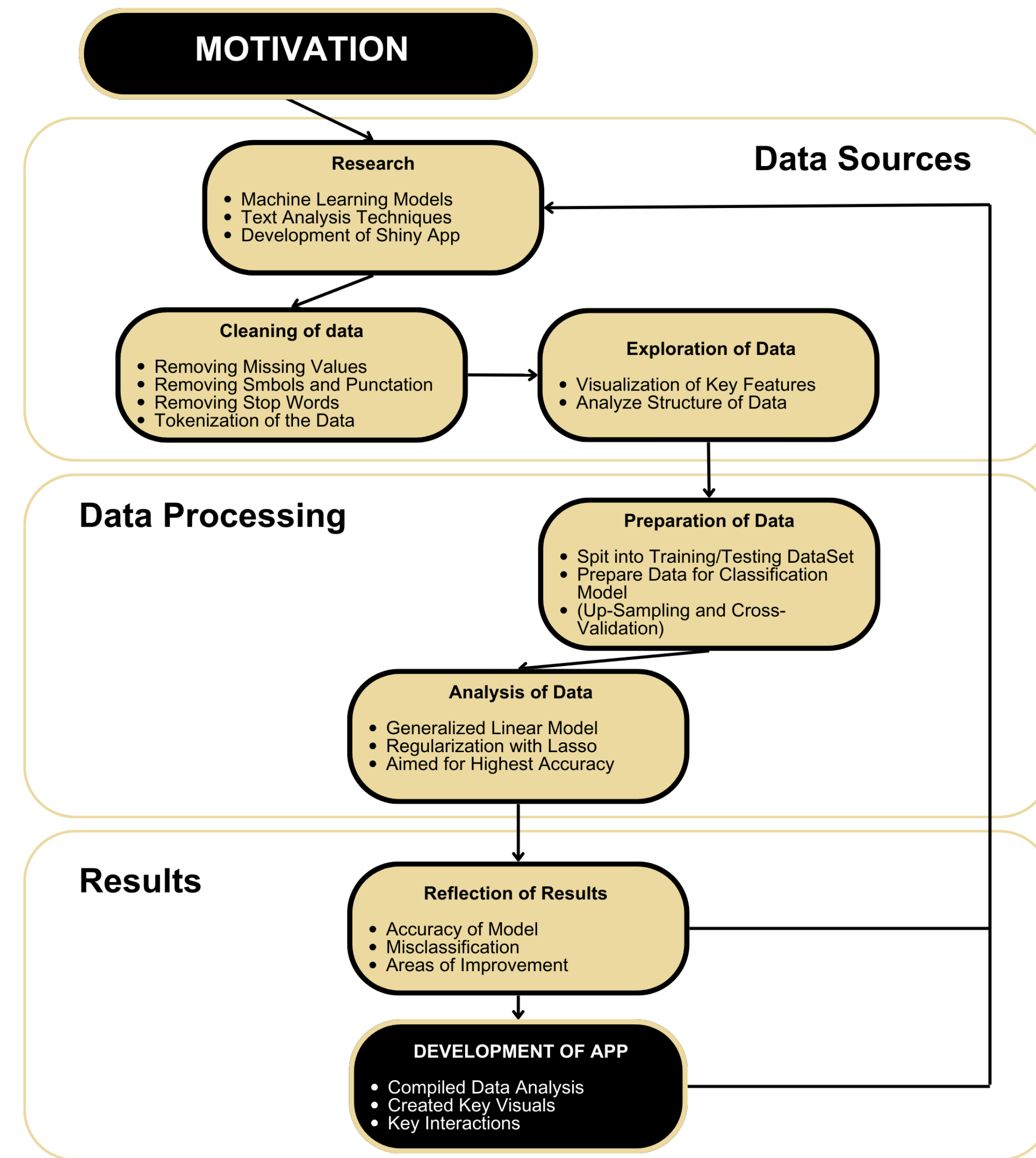
## RESEARCH OBJECTIVES

- How can we develop methods to reduce uncertainty in identifying and assessing media bias?
- How can we create accessible tools that enables users to track and compare bias across different media sources?

## PROCESS

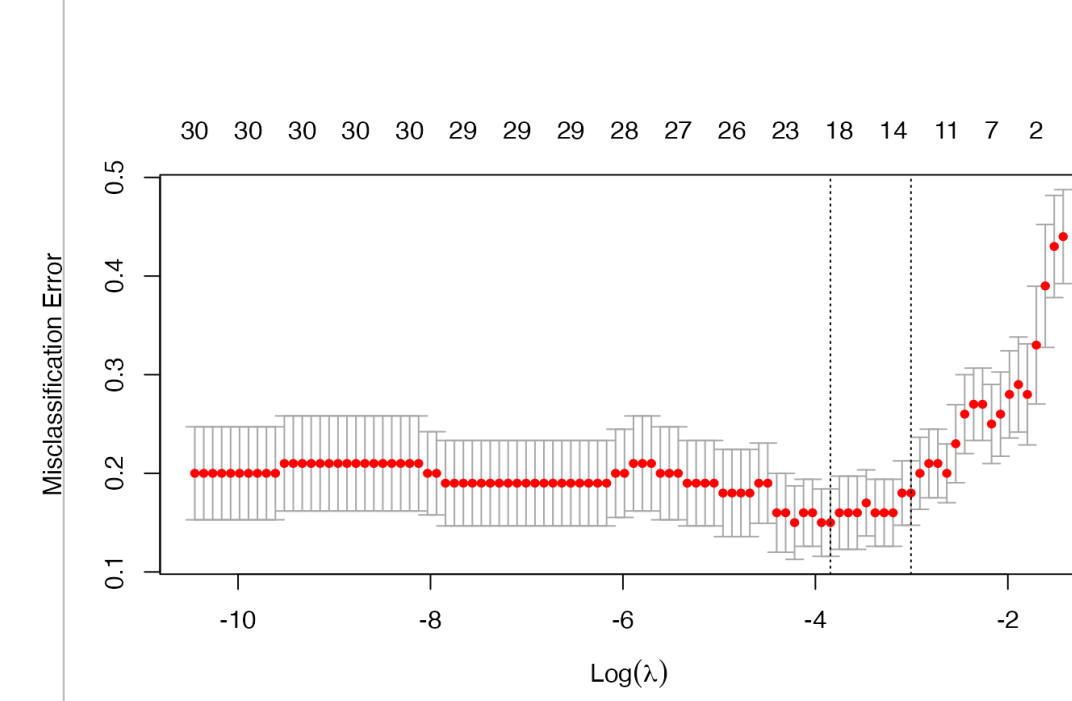
Our initial process involved developing a lexicon sentiment database through text analysis. This proved to be a faulty and unreliable analysis tool. We then conducted research into more accurate methods and found a supervised machine learning would be more effective. We implemented better data cleaning procedures and a more accurate model. The final model we decided on was a Generalized Linear Model with a Lasso (least absolute shrinkage and selection operator) regression. This lasso effect has a shrinking effect on some variables taking them out of the analysis and reducing bias. The model can then better classify the text and articles to certain areas of Political Bias. Our analysis methods will be explained further in the statical review section. We used a few sources in our process, which will be in the acknowledgements.

## METHODOLOGY



## ANALYTICAL RESULTS

In this analysis, we used a multinomial logistic regression model with a lasso penalty, implemented using the glmnet engine in the tidymodels framework. Our goal was to classify political bias into five categories—Left, Center-Left, Center, Center-Right, and Right—based on textual features extracted from news articles. To prepare the text data, we tokenized and cleaned the articles, then transformed them into numerical features using a document-term matrix. This matrix was the input for our penalized regression model. The lasso penalty (L1 regularization) is particularly useful for high-dimensional data like text, where there are thousands of potential predictors (words). It shrinks less informative coefficients to zero, effectively selecting only the most relevant words for classification. We tuned the penalty parameter (lambda) using cross-validation, selecting the value that gave the highest accuracy. After fitting the model across folds, we used collected the predictions to gather results and filtered them by the best-performing penalty value. We then evaluated performance using a confusion matrix and visualized it as a heatmap to understand where the model made correct and incorrect predictions. Overall, this approach balances interpretability and predictive power, allowing us to identify key terms associated with political leanings while managing model complexity. With this method we were able to achieve an accuracy of 78.5%. The areas of Misclassification can be seen in Fig 2.



$$\min_{\beta_0, \beta} \frac{1}{N} \sum_{i=1}^N w_i l(y_i, \beta_0 + \beta^T x_i) + \lambda [(1 - \alpha) \|\beta\|_2^2 / 2 + \alpha \|\beta\|_1]$$

The algorithm would use a lambda value, resulting in a certain amount of bias on variables. This results in error in the classification, where a new lambda value would be calculated. We set this at a limit of ten in our study. Shown by the graph the best lambda is the dip in the curve. Equation shows how the error is minimized from the lambda value.

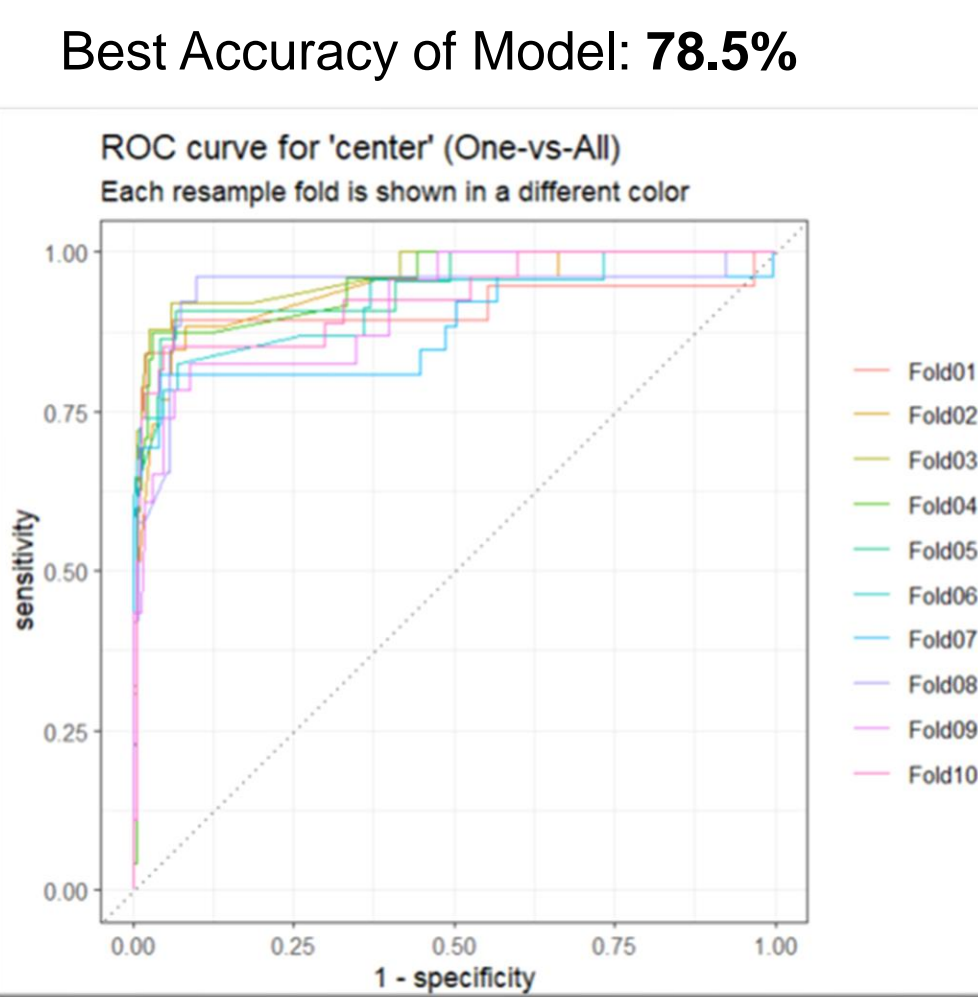


Fig 1. ROC Curve

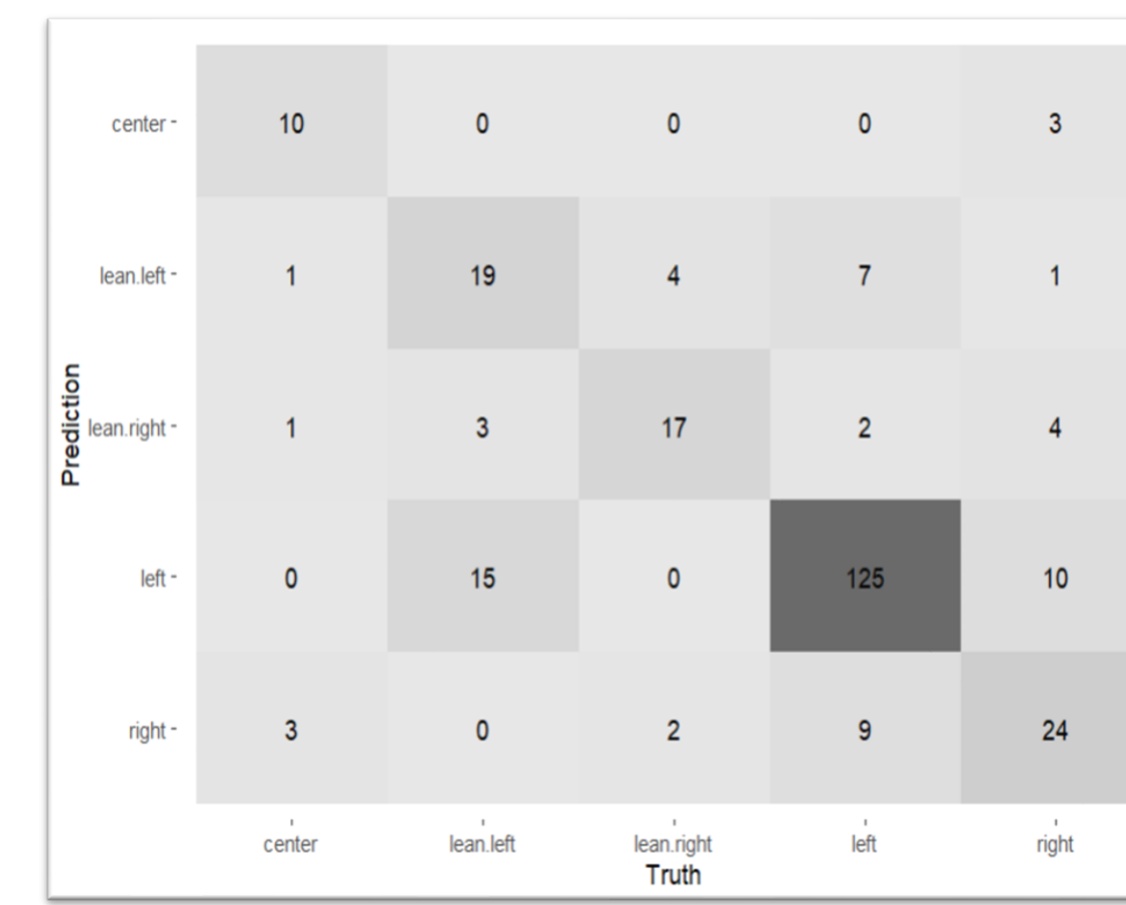
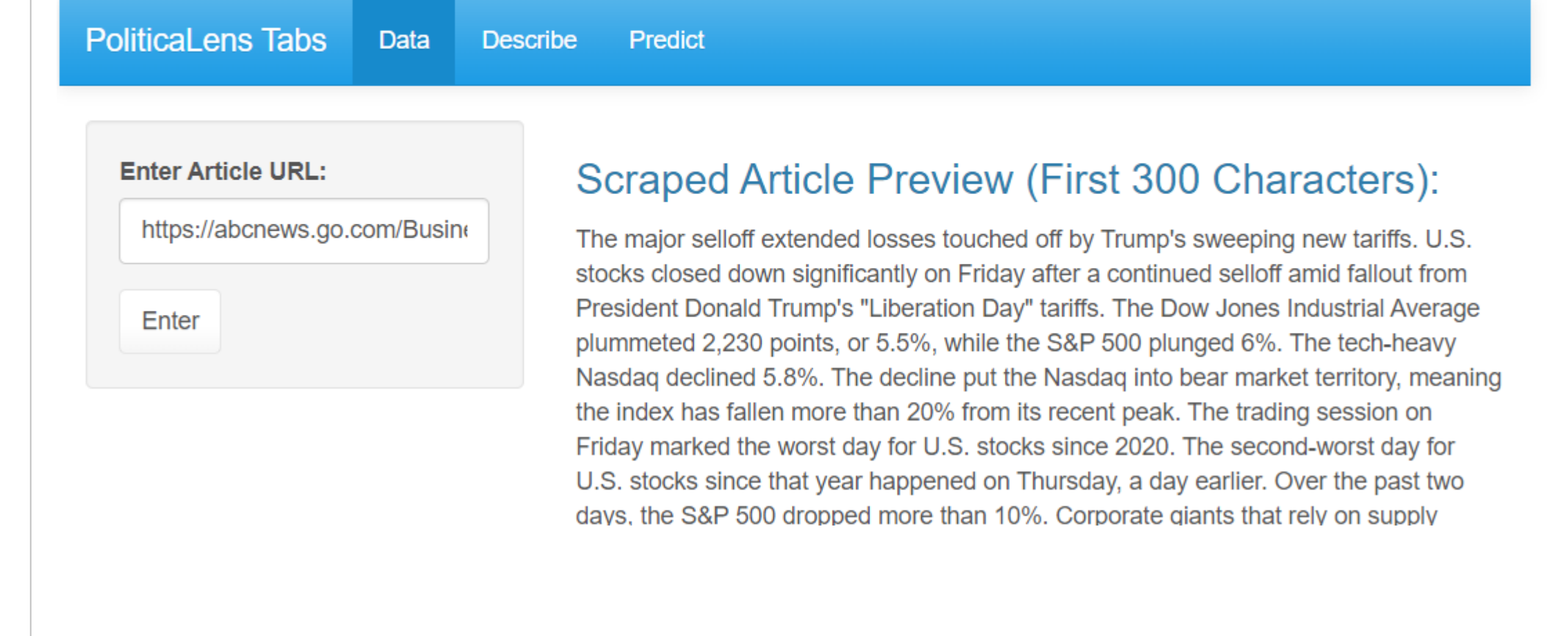


Fig 2. Confusion Matrix of glmnet Model

## SHINY APP

### PoliticalLens: An R-shiny Application to Evaluate Political Bias in American News



The app has three main functions. The user will input an article and the app will strip the text from the webpage. The text will be visible and users will be able to see it. The next tab will display a histogram that will show the most frequent words in the article. Finally, the article will be input into the algorithm we developed. This will lead to a classification of the article based on the algorithm we developed.

## CONCLUSIONS

Our App allows users to be able to detect the bias of any article, ensuring a more holistic reading experience and allowing users to have the power to make their own conclusions based on the information available. This can potentially drive media literacy by ensuring that critical thinking triumphs over misinformation and readers are aware of the truth. This can also potentially force news outlets to refine their policies and become more aware of the information that they feed out. We also believe that there are several areas of improvement. The algorithm could be more accurate, since more than 20% of articles are being misclassified. We would also like to stretch to other forms of media, not only articles. An implementation that could work with twitter would be extremely informative and useful for users.

## ACKNOWLEDGEMENTS

We would like to thank Professor Matthew Lanham for this opportunity, his guidance, and support on this project. Also to Emil Hvitfeldt and Julia Sage, for help through their book.

### Sources Used

- Wickham, Hadley. *Mastering Shiny: Build Interactive Apps, Reports, and Dashboards Powered by R*. O'Reilly Media, 2021, <https://mastering-shiny.org>
- Hvitfeldt, Emil, and Julia Silge. *Supervised Machine Learning for Text Analysis in R*. CRC Press, 2022, <https://smltar.com>.