Group Members: Tanveer Bains, Keanna Medina, Prateek Rao, Ashley Tran, Rafael Tuazon

Predicting Malaria Outbreak Using Machine Learning Project Report

Problem Statement

Measles is a highly contagious disease that infected "3 to 4 million people in the United States...each year" before the measles, mumps, and rubella (MMR) vaccine was invented [4]. When the vaccine was invented in 1963, the United States started "a highly effective vaccination program" that led to measles being "declared eliminated...from the United States in 2000" [4]. Recent studies, however, have shown that there are "negative effects of an anti-vaccine culture on individual and population-level health" [34]. "In the case of measles vaccine, a licensed vaccine used for several decades, unfounded speculations about a possible association with autism and autism spectrum disorder led to detectable population-level decreases in measles vaccine use, in turn leading to a resurgence of measles cases, hospitalizations, and measles-related deaths..."[34]. In light of the resurgence of measles, our project attempts to predict measles outbreaks in each state within the United States based on MMR vaccination data collected by the CDC. Our project uses different supervised learning classification models (Decision Trees, Random Forest, Naive Bayes, Neural Networks, and K-Nearest Neighbors) to predict outbreaks and we compare the results to determine the model(s) that has the best predictive power. In doing so, we hope to find classification models that are effective in predicting outbreaks based on vaccination data, so that these models can be used in preventing future outbreaks within vulnerable states.

Previous Work

There exists a good amount of research on predicting epidemics through the use of mathematical models. Some notable cases that we found include feeding search engine data to artificial neural networks in order to forecast AIDS in China, a prediction model to successfully predict and act on a measles epidemic in 1997 in New Zealand, and a tool created by Carnegie Mellon University that forecasts the flu [11, 36, 37]. The AIDS prediction models in China calculated the best threshold for the Pearson correlation coefficient that they used [11]. We followed a similar take in finding the best threshold for our confusion matrix to better our outbreak prediction accuracy. In general regards to the health sector, "Support Vector Machine (SVM), Naïve Bayes, Decision Tree and Artificial Neural Network (ANN) are some of the major classifiers of Machine Learning techniques which are widely used in healthcare as decision support techniques" [40]. Likewise, we included these major models in our project for credible comparisons. The article further details in utilizing Receiver Operating Characteristics (ROC) Area to measure the accuracy of models as well, which accounted for our approach towards model comparison. We used the area under the ROC curve (AUC) to compare our classifiers because it “represents degree or measure of separability”, meaning it describes how accurate a model is at “distinguishing between classes” [12]. We collected data on outbreaks and state vaccination rates from the CDC website. We used SKLearn’s implementation of our models, recursive feature elimination, ROC curve, AUC and confusion matrix calculations. We also followed SKLearn’s examples on how to plot confusion matrices when we coded our project.

Project Decomposition

The work was divided into three stages: scraping and cleaning the data, fitting various models with the data, and comparing the models’ score values/ROC curves. After we created a list of useful datasets, Rafael cleaned the scraped data, partially through removing extraneous data points (such as fields that had NaN values) to compile the feature set. Prateek generated the target data set after collecting the data about measle outbreaks from the CDC website. Afterwards, Ashley aggregated the data and ran a feature selection process in order to help us see the relevant features. While the data was being cleaned and processed, Keanna compiled information regarding various potential models for us to use. And since the implementations of SKLearn’s models are fairly straightforward, each member was able to implement at least one model.

The models were divided as such:
Ashley: Decision Tree,
Rafael: Random Forest,
Tanveer: Multi-Layer-Perceptron, SVM
Keanna: K-Nearest Neighbors,
Prateek: Naive Bayes (Gaussian, Multinomial, and Bernoulli)

Ashley then created a new notebook file that compares the results of all of the completed models.

Experience in Coding

Much of our code makes use of the SKLearn module. We opted for the module’s implementations because of the experience we gained from the homework assignments, as well as the available documentation for tools we had not used previously (ROC curves, confusion matrix, etc). First, we attempted to visualize our results by plotting a decision boundary, but we ran into many issues. We tried using principal component analysis to dimensionally reduce our data into two dimensions, but that proved to be uninformative. Principal component analysis is a linear technique and further research led us to find out that “ linear techniques cannot adequately handle complex nonlinear data” [43]. We then tried to plot decision boundaries in pair plots of two features, but that yielded no results because the data was not separable. After discussions with Hirak, he said that we should research ROC curves. Upon researching ROC curves, we learned that the area under the curve (AUC) was one of the better ways to compare classifiers. “The area under the ROC curve, or the equivalent Gini index, is a widely used measure of performance of supervised classification rules. It has the attractive property that it side-steps the need to specify the costs of the different kinds of misclassification” [8]. We originally planned to compare the classifiers’ accuracy scores, but we learned that ”misclassification rate is often a poor criterion by which to assess the performance of classification rules” [8]. Our data contained many data points that did not have an outbreak and few data points that contained an outbreak; thus, a classifier could achieve a high accuracy score if it predicted no outbreak a majority of the time. From our research into ROC curves, we found that one of the ways to visualize classification results was to visualize a confusion matrix, which is used to calculate ROC curves. Additionally, we ran cross-validation to better our data and increase accuracy. From this, we calculated the best degree/depth to use for each model accordingly. Each model has a notebook file that contains the model's fitting, cross-validation error plot, confusion matrix plot, and ROC curve plot.

Results

The data collected from the CDC contained 8 features: state, approximate age, survey year, vaccination rate, upper confidence limit on the vaccination rate, lower confidence limit on the vaccination rate, confidence interval, and sample size. After running SKLearn’s recursive feature elimination, we found the relevant features to be approximate age, vaccination rate, upper confidence limit, and lower confidence limit; therefore, we ran our models with those features.

For all of the models, we first plotted a confusion matrix that was calculated with each classifier’s standard threshold probability of an outbreak of 0.5. To decrease the amount of false negatives, a new threshold probability was used, which was calculated to be the median value of the predicted probabilities of the outbreak class. The new threshold was used to plot an augmented confusion matrix.

After running 5-fold cross-validation on the Decision Tree classifier, a max depth of 4 was used to control the fitting of the data. The accuracy for the Decision Tree classifier with a max depth of 4 was 0.96. When plotting the ROC curve, the area under the curve was around 0.6287 (Figure 9). Figure 1 depicts the standard confusion matrix compared to the augmented confusion matrix.

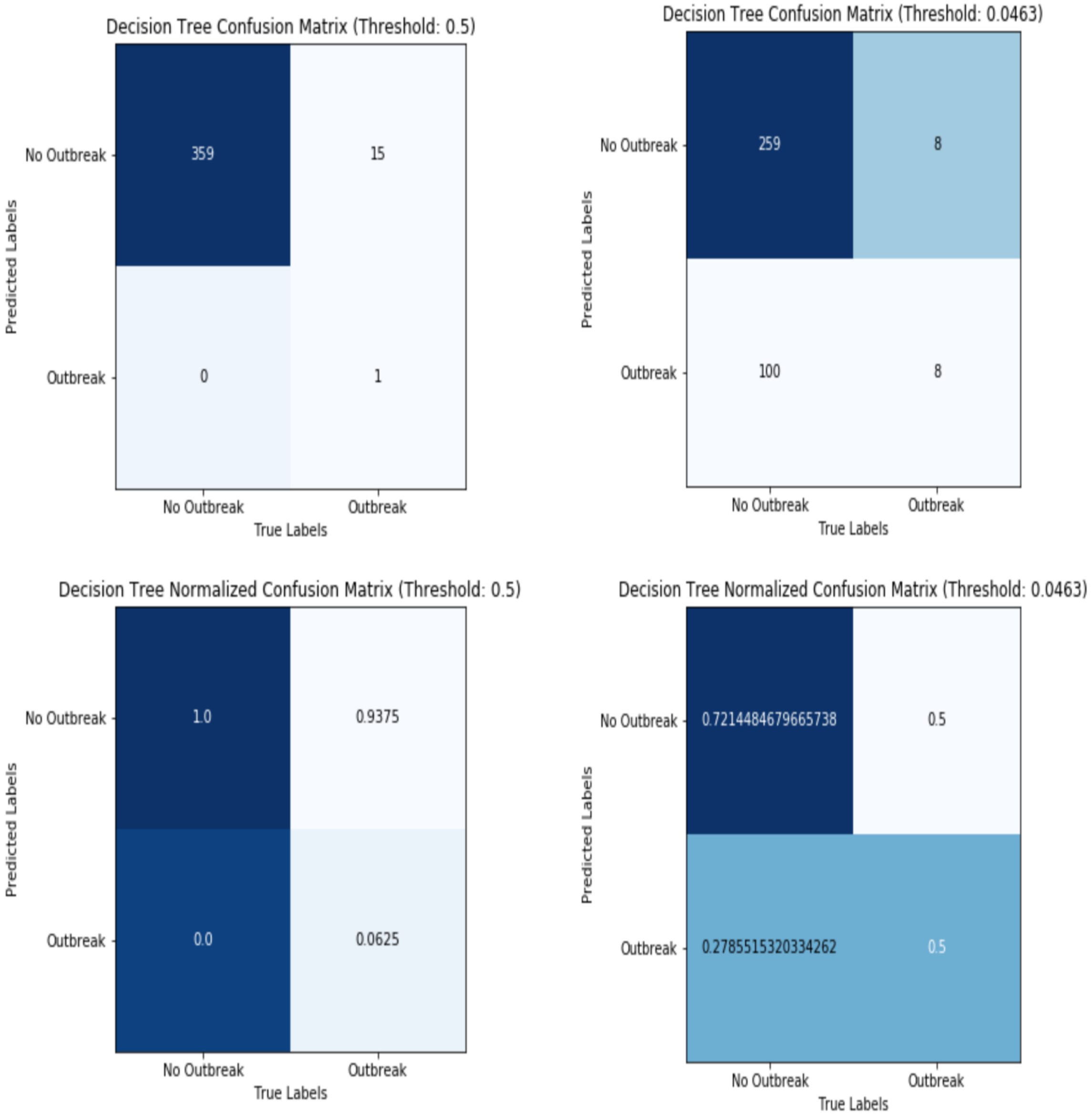

Figure 1: The confusion matrices on the left were calculated using the standard threshold of 0.5, while the confusion matrices on the right were calculated with the median threshold of 0.0463. Lowering the standard threshold to the median threshold lowered the number of false negatives from 15 to 8; however, the number of false positives increased from 0 to 100 as a result.

The Random Forest classifier was also run with 5-fold cross-validation and found that a max depth of 4 best fit the data. The accuracy of the Random Forest classifier with a max depth of 4 was around 0.9573. The area under the ROC curve was around 0.6580 (Figure 9). Figure 2 depicts the standard confusion matrix compared to the augmented confusion matrix.

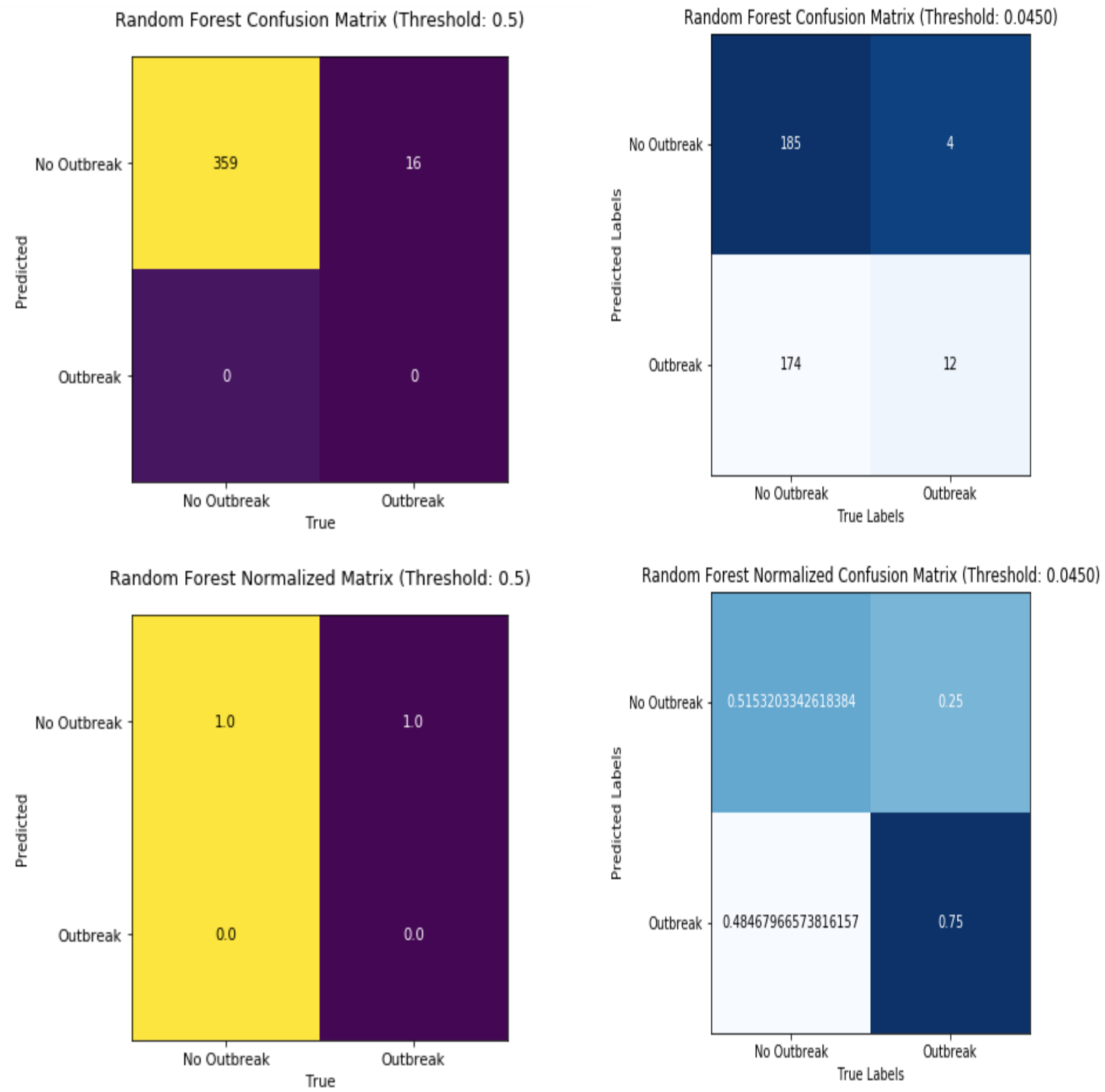

Figure 2: The confusion matrices on the left were calculated using the standard threshold of 0.5, while the confusion matrices on the right were calculated with the median threshold of 0.0450. Lowering the standard threshold to the median threshold lowered the number of false negatives from 16 to 4; however, the number of false positives increased from 0 to 174 as a result.

The Support Vector Machine classifier was run with degree 3 and had an accuracy score around 0.9573. The area under the ROC curve was around 0.4506 (Figure 9). Figure 3 depicts the standard confusion matrix compared to the augmented confusion matrix.

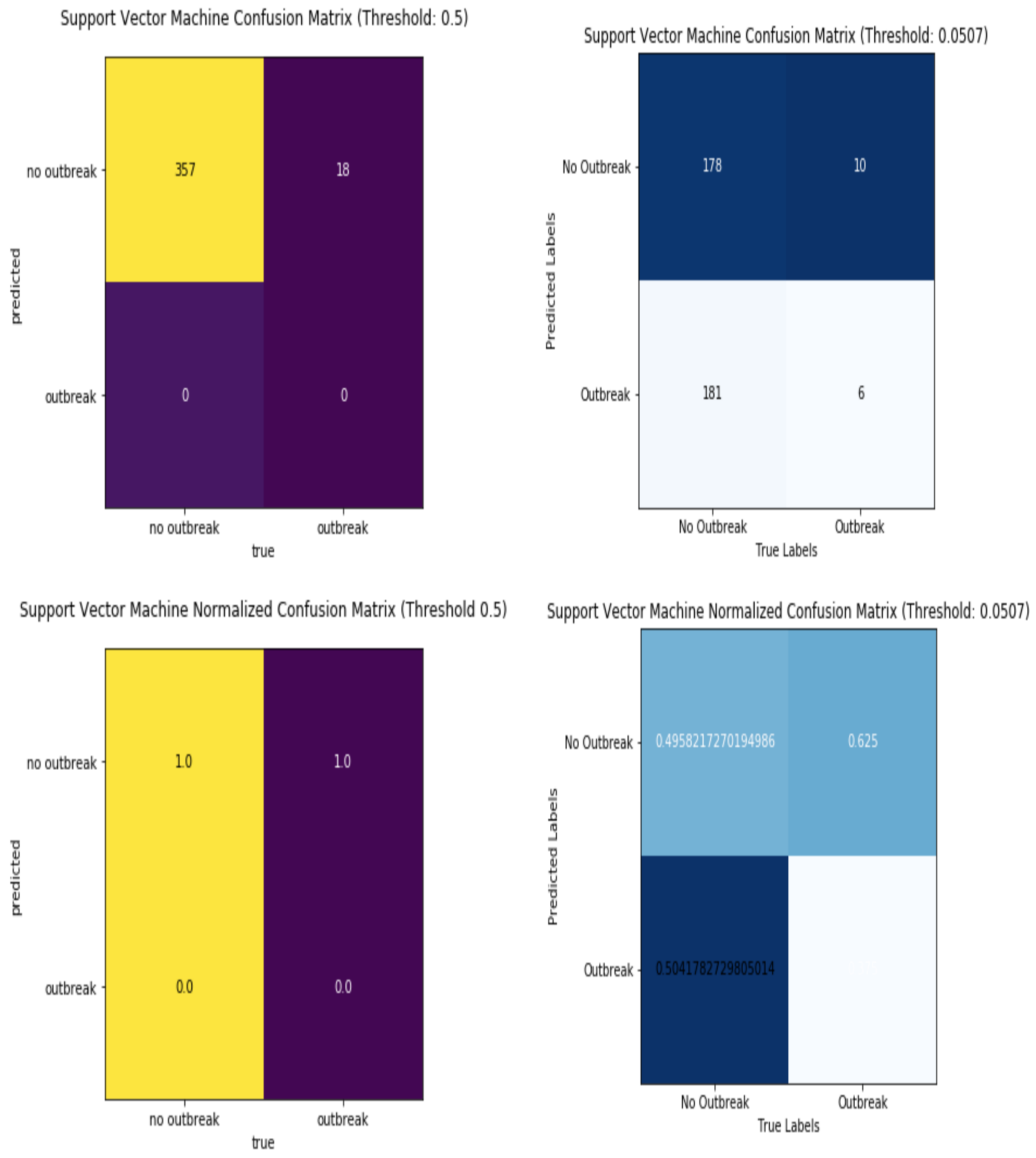

Figure 3: The confusion matrices on the left were calculated using the standard threshold of 0.5, while the confusion matrices on the right were calculated with the median threshold of 0.0507. Lowering the standard threshold to the median threshold lowered the number of false negatives from 16 to 10; however, the number of false positives increased from 0 to 181 as a result.

The Multi-layer Perceptron Neural Network classifier was also run with 5-fold cross-validation and it found that the best neuron count was 3. It had an accuracy score around 0.9573. The area under the ROC curve was around 0.5380 (Figure 9). Figure 4 depicts the standard confusion matrix compared to the augmented confusion matrix.

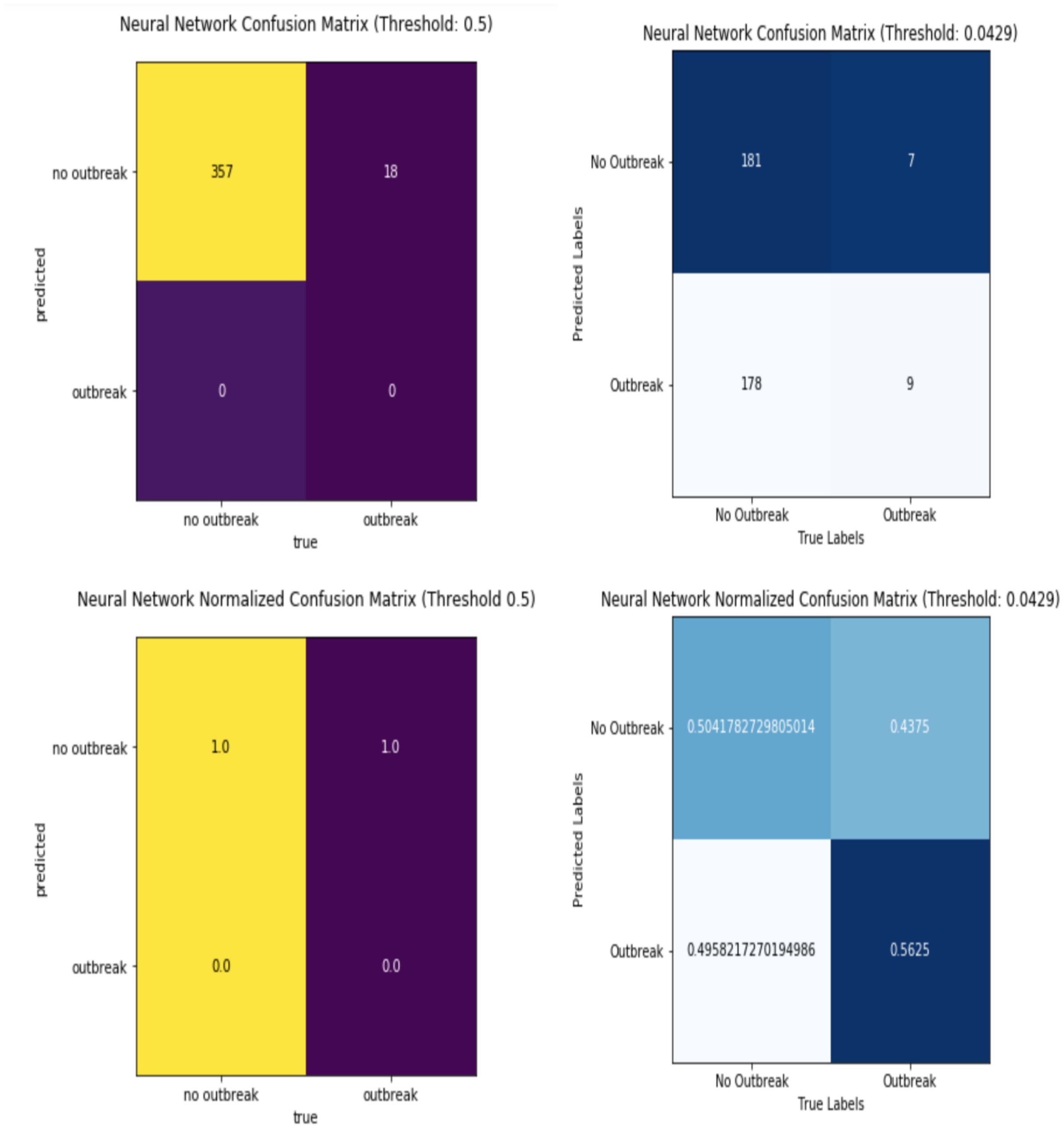

Figure 4: The confusion matrices on the left were calculated using the standard threshold of 0.5, while the confusion matrices on the right were calculated with the median threshold of 0.0429. Lowering the standard threshold to the median threshold lowered the number of false negatives from 16 to 7; however, the number of false positives increased from 0 to 178 as a result.

The K-Nearest Neighbor classifier used the cross_val_score from the sklearn.model_selection library and it found that the best number of neighbors was 6. The K-Nearest Neighbor classifier had an accuracy score of around 0.9573. The area under the ROC curve was around 0.5303 (Figure 9). Figure 5 depicts the standard confusion matrix compared to the augmented confusion matrix.

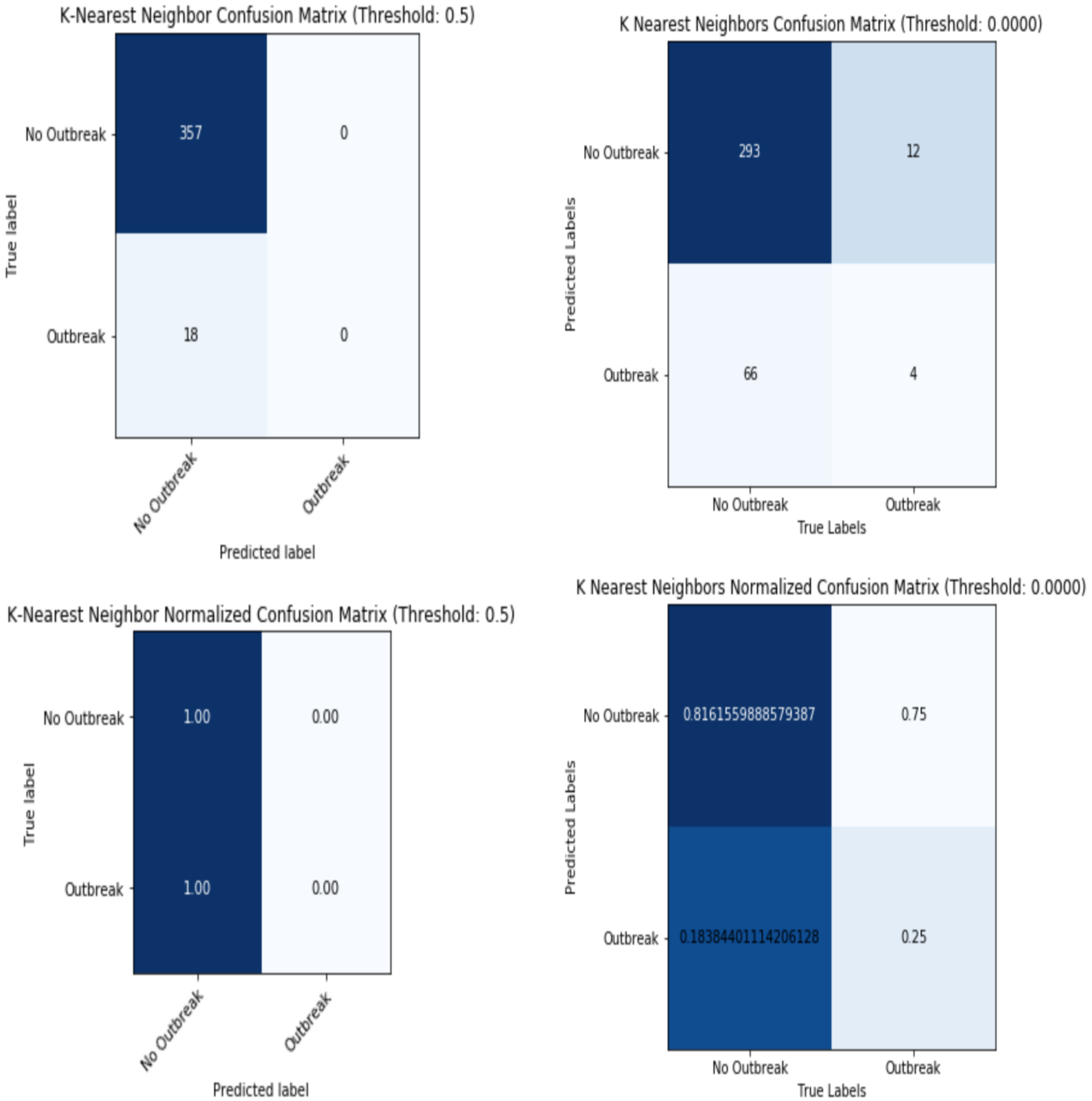

Figure 5: The confusion matrices on the left were calculated using the standard threshold of 0.5, while the confusion matrices on the right were calculated with the median threshold of 0.0. Lowering the standard threshold to the median threshold lowered the number of false negatives from 16 to 12; however, the number of false positives increased from 0 to 66 as a result.

We ran three different Naive Bayes classifiers: Gaussian, Multinomial, and Bernoulli. The Gaussian Naive Bayes classifier had an accuracy score of around 0.9633 and an area under the ROC curve of around 0.5555 (Figure 9). Figure 6 depicts the standard confusion matrix compared to the augmented confusion matrix for the Gaussian Naive Bayes classifier. The Multinomial Naive Bayes classifier had an accuracy score of around 0.9633 and an area under the ROC curve of around 0.5632 (Figure 9). Figure 7 depicts the standard confusion matrix compared to the augmented confusion matrix for the Multinomial Naive Bayes classifier. Finally, the Bernoulli Naive Bayes classifier had an accuracy score of around 0.9633 and an area under the ROC curve of around 0.5530 (Figure 9). Figure 8 depicts the standard confusion matrix compared to the augmented confusion matrix for the Bernoulli Naive Bayes classifier.

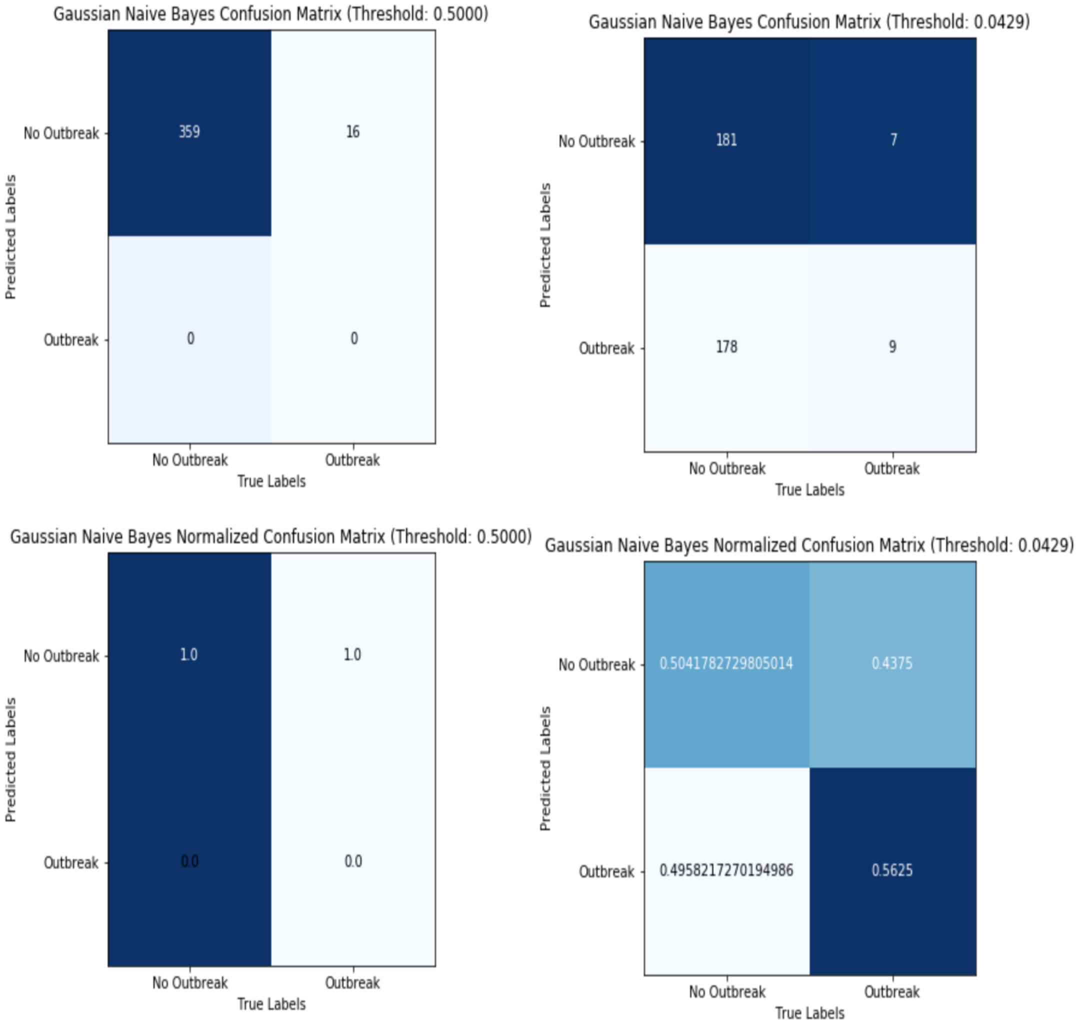

Figure 6: The confusion matrices on the left were calculated using the standard threshold of 0.5, while the confusion matrices on the right were calculated with the median threshold of 0.0429. Lowering the standard threshold to the median threshold lowered the number of false negatives from 16 to 7; however, the number of false positives increased from 0 to 178 as a result.

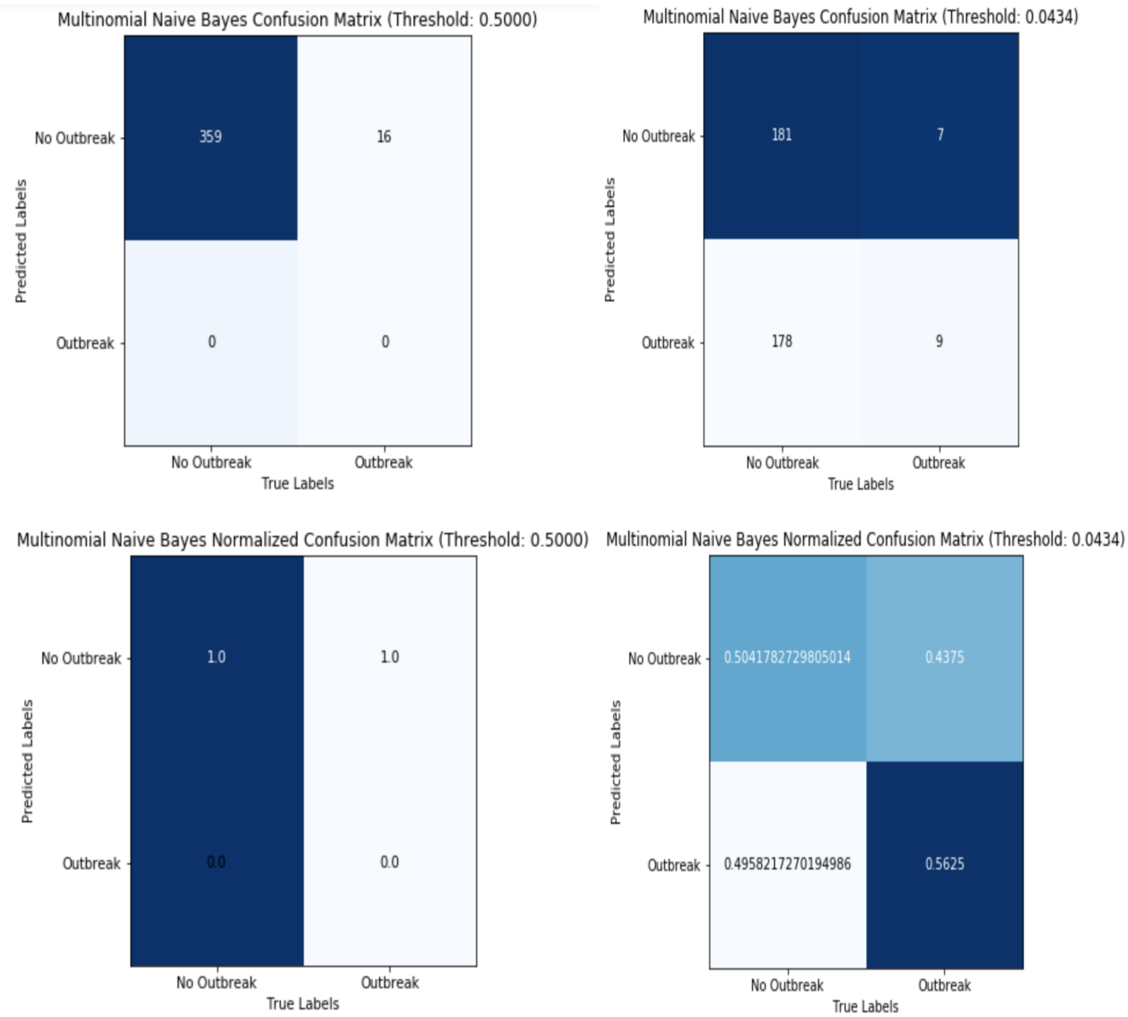

Figure 7: The confusion matrices on the left were calculated using the standard threshold of 0.5, while the confusion matrices on the right were calculated with the median threshold of 0.0434. Lowering the standard threshold to the median threshold lowered the number of false negatives from 16 to 7; however, the number of false positives increased from 0 to 178 as a result.

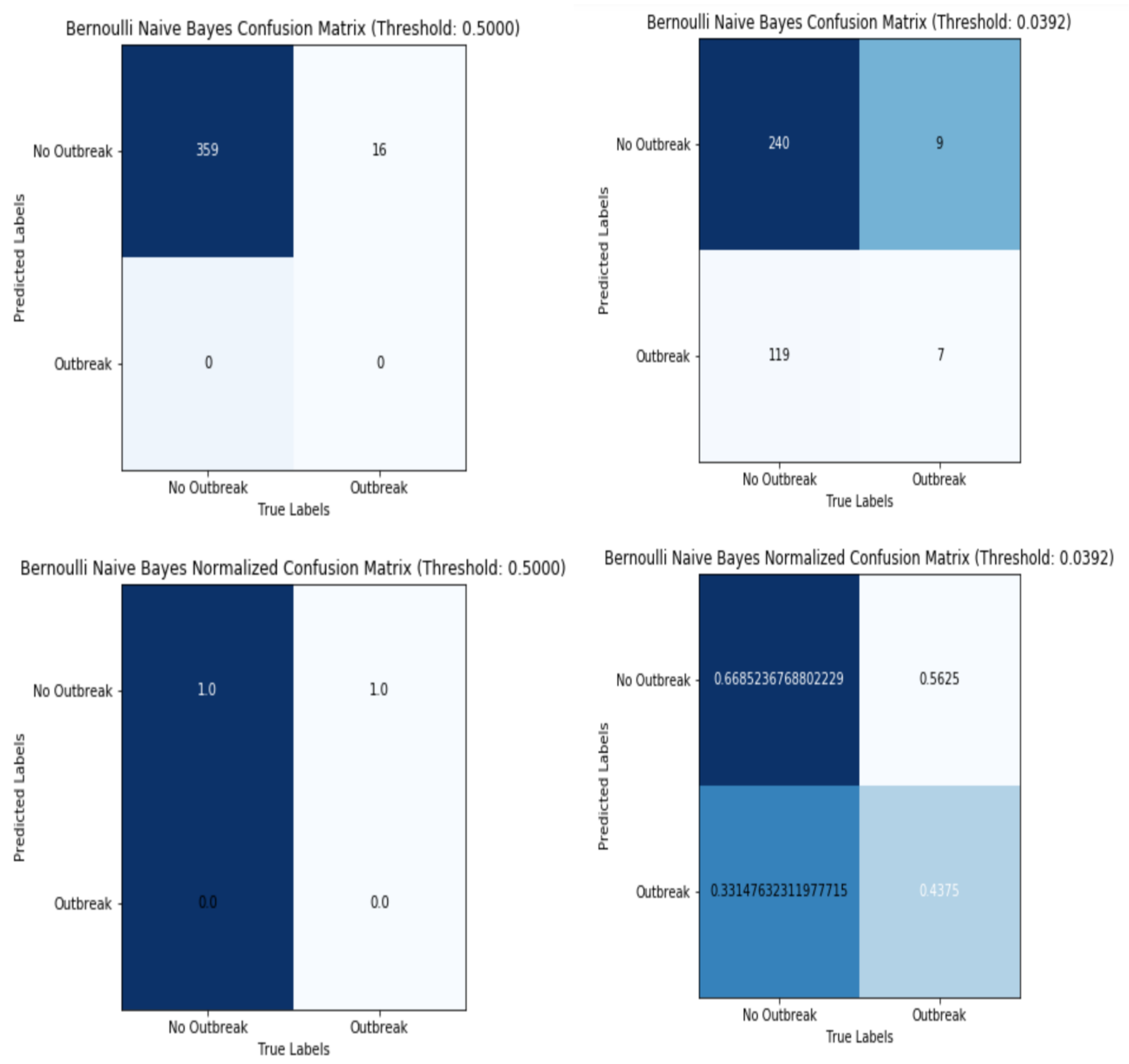

Figure 8: The confusion matrices on the left were calculated using the standard threshold of 0.5, while the confusion matrices on the right were calculated with the median threshold of 0.0392. Lowering the standard threshold to the median threshold lowered the number of false negatives from 16 to 9; however, the number of false positives increased from 0 to 119 as a result.

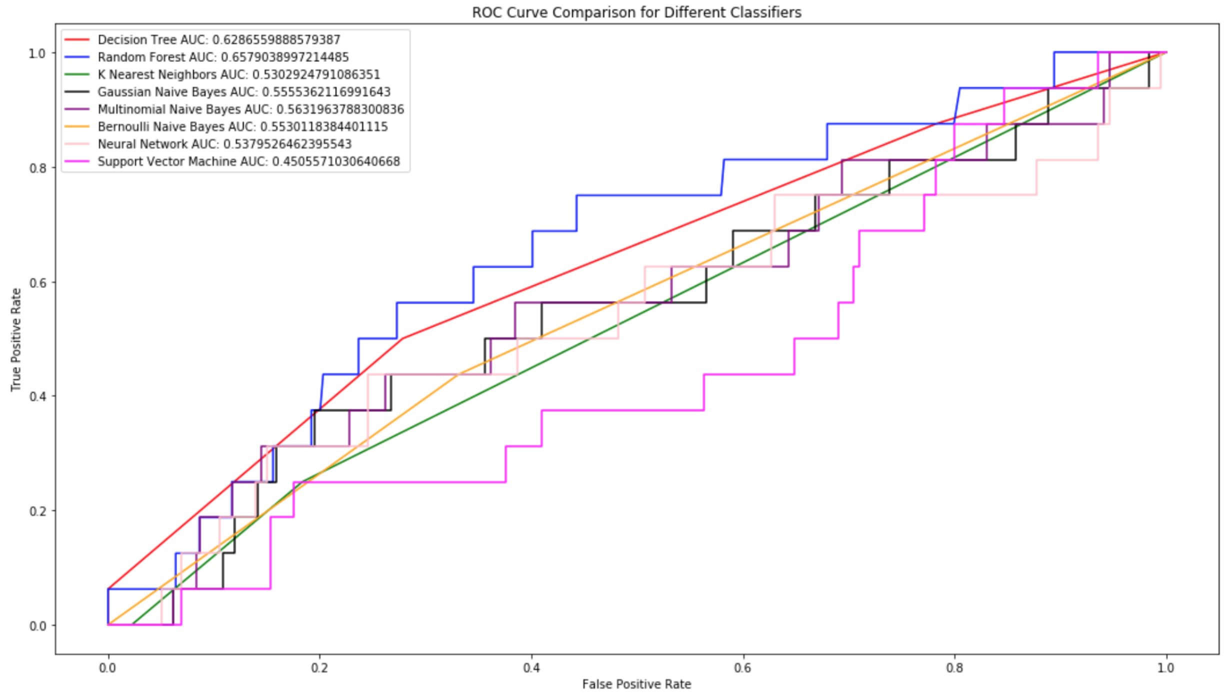

Figure 9: The graph shows the ROC curves for each of the classification models. The legend contains the area under the curve (AUC) for each of the ROC curves as well.

Conclusion

From our results, we found that all of the classification models had high accuracy scores of around 0.96. At first glance, it seems that all models perform similarly. However, we've learned that standard accuracy is not the best measurement. The problem lied in our data where the distribution is inherently skewed. Further analysis of the confusion matrices that were calculated with a threshold of 0.5, showed that only the Decision Tree classifier correctly identified an outbreak; however, it only predicted 1 out of 16 outbreaks. The classifiers incorrectly predicted all of the outbreaks to be false negatives, despite having a high accuracy score. To prevent overfitting to a single class and better the accuracy, we looked at the area under the curve (AUC) of the ROC curve. Illustrated from the graph above, ROC-AUC details the performance of each model in a much clearer way.

From Figure 9, we can see that the Random Forest had the best AUC value of around 0.6580 (Figure 9). This is reflected in its predictions when the threshold was lowered to the median value of the probabilities for the outbreak class; it had the highest amount of true positives at 12 out of 16 outbreaks (Figure 2).We believe that this is because a Random Forest classifier "builds a multitude of decision trees and merges them to produce an accurate and stable prediction" [6]. It also performs its own feature selection over multiple features, providing a wider diversity and builds a better model [39].

The second best model was the Decision Tree classifier with an AUC value of around 0.6287 (Figure 9). Like the Random Forest classifier, the Decision Tree classifier performs its own feature selection, so it can decide which features contribute the most to classification and their relative importance [13]. Unlike the Random Forest classifier, however, it only splits the data set on one feature, which could be why it performed worse than the Random Forest classifier [13].

The three Naive Bayes classifiers were the next best models; however, they are known as “bad estimator[s]” and it is a common mistake to consider its probability outputs as valid [10] We believe it did not perform as well because Naive Bayes classifiers work best on small data sets (ours is a large data set of over 1000 data points) and it has the underlying assumption that all features are independent [10]. Since the upper confidence limit and lower confidence limits were correlated with vaccination rate, they had to be removed to prevent over inflating the importance of vaccination rate [10]. This led to the classifiers being tested on two features, age and vaccination rate, which may not be as accurate as the models that trained and tested on all of the features.

The next classifier was the Multi-layer Perceptron Neural Network classifier with an AUC value of around 0.5380 (Figure 9). We believe that this is because our data set, while large for other classifiers, is too small for the Multi-layer Perceptron Neural Network classifier to work accurately. The classifier requires at least thousands if not millions of labeled samples and we only had a bit over 1000 data points [7].

The K Nearest Neighbors classifier came after the Multi-layer Perceptron Neural Network with an AUC value of around 0.5303 (Figure 9). We believe that the K-Nearest Neighbor did not perform as well because it finds the most common classification, which was the non-outbreak class for our data set [2]. It did not predict the outbreak class because there were so few points that were true outbreaks.

Finally, the worst classifier for our data set was actually the Support Vector Machine classifier with an AUC of around 0.4506 (Figure 9). We believe that this is because Support Vector Machine classifiers are not suited for large data sets with overlapping classes [1]. There was no clear hyperplane for the Support Vector Machine classifier to find, so it did not work as well for our data set [1].

Our project attempted to find classifiers that could predict outbreaks in the United States based on vaccination data collected by the CDC. We compared the classifiers and ranked the classifiers in terms of accuracy based on the AUC of the ROC curve, but we do not think that our application of the models is suited for practical use. While we adjusted the threshold to lower the number of false negatives, we subsequently had a large increase in false positives. Though false positives are not as detrimental as false negatives in the case of an outbreak, we predicted less than 8 percent of true outbreaks out of all of our positive predictions. It would still be hard to try and prevent an outbreak if most of the outbreaks we predicted were false positives. This is because of the fact that most of our models had an AUC close to 0.5, meaning that our models had "no class separation capacity whatsoever" [12]. We, however, did learn more about machine learning and classification models through this project and we hope that we will be able create projects that can be applicable to the real world as we continue our studies.

Bibliography

1. Aylien,N.B. (2016, July). Support Vector Machines: A Simple Explanation. https://www.kdnuggets.com/2016/07/support-vector-machines-simple-explanation.html
2. Bronshtein, A. (2017, April 11). A Quick Introduction to K-Nearest Neighbors Algorithm. Retrieved from https://medium.com/@adi.bronshtein/a-quick-introduction-to-k-nearest-neighbors-algorithm-62214cea29c7
3. Centers for Disease Control and Prevention. (2018, October 11). 1995 through 2017 Childhood Measles, Mumps, and Rubella (MMR) Vaccination Coverage Trend Report. Retrieved from https://www.cdc.gov/vaccines/imz-managers/coverage/childvaxview/data-reports/mmr/trend/index.html
4. Centers for Disease Control and Prevention. (2018, February 5). Measles History. Retrieved from https://www.cdc.gov/measles/about/history.html.
5. Centers for Disease Control and Prevention. (2018, October 2). Measles, mumps, and rubella (MMR) vaccination coverage among adolescents 13-17 years by State, HHS Region, and the United States, National Immunization Survey-Teen (NIS-Teen), 2008 through 2017. Retrieved from https://www.cdc.gov/vaccines/imz-managers/coverage/teenvaxview/data-reports/mmr/trend/index.html
6. Donges, N. (2018, February 22). The Random Forest Algorithm. Retrieved from https://towardsdatascience.com/hype-disadvantages-of-neural-networks-6af04904ba5b
7. Donges, N. (2018, April 17). Pros and Cons of Neural Networks. Retrieved from https://towardsdatascience.com/hype-disadvantages-of-neural-networks-6af04904ba5b
8. Hand, D., & Till, R. (2001). A Simple Generalisation of the Area Under the ROC Curve for Multiple Class Classification Problems. *Machine Learning. Volume 45.* pp 171–186. https://link.springer.com/content/pdf/10.1023%2FA%3A1010920819831.pdf
9. Liang, F., Guan, P., Wu, W., & Huang, D. (2018, June 25). Forecasting influenza epidemics by integrating internet search queries and traditional surveillance data with the support vector machine regression model in Liaoning, from 2011 to 2015. *Peerj*. https://peerj.com/articles/5134/?utm_source=TrendMD&utm_campaign=PeerJ_TrendMD_1&utm_medium=TrendMD

10. Muller, M. (2018, February 28). Naive bayes Classification With Sklearn. Retrieved from https://blog.sicara.com/naive-bayes-classifier-sklearn-python-example-tips-42d100429e44
11. Nan, Y., & Gao, Y. (2018, July 11). A machine learning method to monitor China's AIDS epidemics with data from Baidu trends. *PLoS ONE. Volume* 12 (Issue 7). https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0199697
12. Narkhede S. (2018, June 26). Understanding AUC-ROC Curve. Retrieved from https://towardsdatascience.com/understanding-auc-roc-curve-68b2303cc9c5
13. Navlani, A. (2018, December 28). Decision Tree Classification in Python. https://www.datacamp.com/community/tutorials/decision-tree-classification-python
14. Nsoesie, E.O., Beckman, R. Marathe, M., & Lewis, B. (2011, January). Prediction of an Epidemic Curve: A Supervised Classification Approach. *Stat Commun Infect Dis. Volume 3* (Issue 1): 5. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3445421/
15. Pendregosa, F., Varo-quaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). Cross-Validation: Evaluating Estimator Performance. *Journal of Machine Learning Research. Volume* 12. Pp 2825-2830. https://scikit-learn.org/stable/modules/cross_validation.html
16. Pendregosa, F., Varo-quaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). Scikit-learn.ensemble.RandomForestClassifier: Random Forest Machine Learning in Python. *Journal of Machine Learning Research. Volume* 12. Pp 2825-2830. https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html
17. Pendregosa, F., Varo-quaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). sklearn.naive_bayes: Naive Bayes. *Journal of Machine Learning Research. Volume* 12. Pp 2825-2830. https://scikit-learn.org/stable/modules/classes.html#module-sklearn.naive_bayes
18. Pendregosa, F., Varo-quaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). sklearn.naive_bayes.BernoulliNB. *Journal of Machine Learning Research. Volume* 12. Pp 2825-2830. https://scikit-learn.org/stable/modules/generated/sklearn.naive_bayes.BernoulliNB.html#sklearn.naive_bayes.BernoulliNB
19. Pendregosa, F., Varo-quaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). sklearn.naive_bayes.MultinomialNB. *Journal of Machine Learning Research. Volume*

12. Pp 2825-2830. https://scikit-learn.org/stable/modules/generated/sklearn.naive_bayes.MultinomialNB.html#sklearn.naive_bayes.MultinomialNB

20. Pendregosa, F., Varo-quaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). sklearn.naive_bayes.GaussianNB. *Journal of Machine Learning Research. Volume* 12. Pp 2825-2830. https://scikit-learn.org/stable/modules/generated/sklearn.naive_bayes.GaussianNB.html#sklearn.naive_bayes.GaussianNB
21. Pendregosa, F., Varo-quaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). Receiver Operating Characteristic. *Journal of Machine Learning Research. Volume* 12. Pp 2825-2830. https://scikit-learn.org/stable/auto_examples/model_selection/plot_roc.html
22. Pendregosa, F., Varo-quaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). Decision Trees. *Journal of Machine Learning Research. Volume* 12. Pp 2825-2830. https://scikit-learn.org/stable/modules/tree.html
23. Pendregosa, F., Varo-quaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). sklearn.tree.DecisionTreeClassifier. *Journal of Machine Learning Research. Volume* 12. Pp 2825-2830. https://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeClassifier.html
24. Pendregosa, F., Varo-quaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). sklearn.neighbors.NearestNeighbors. https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.NearestNeighbors.html
25. Pendregosa, F., Varo-quaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). sklearn.neural_network.MLPClassifier. *Journal of Machine Learning Research. Volume* 12. Pp 2825-2830. https://scikit-learn.org/stable/modules/generated/sklearn.neural_network.MLPClassifier.html#sklearn.neural_network.MLPClassifier
26. Pendregosa, F., Varo-quaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A.,

Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). Neural Network Models (Supervised). *Journal of Machine Learning Research. Volume* 12. Pp 2825-2830. https://scikit-learn.org/stable/modules/neural_networks_supervised.html

27. Pendregosa, F., Varo-quaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). sklearn.metrics.roc_curve. *Journal of Machine Learning Research. Volume* 12. Pp 2825-2830. https://scikit-learn.org/stable/modules/generated/sklearn.metrics.roc_curve.html
28. Pendregosa, F., Varo-quaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). sklearn.metrics.confusion_matrix. *Journal of Machine Learning Research. Volume* 12. Pp 2825-2830. https://scikit-learn.org/stable/modules/generated/sklearn.metrics.confusion_matrix.html
29. Pendregosa, F., Varo-quaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). sklearn.metrics.auc. *Journal of Machine Learning Research. Volume* 12. Pp 2825-2830. https://scikit-learn.org/stable/modules/generated/sklearn.metrics.auc.html
30. Pendregosa, F., Varo-quaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). Confusion Matrix. *Journal of Machine Learning Research. Volume* 12. Pp 2825-2830. https://scikit-learn.org/stable/auto_examples/model_selection/plot_confusion_matrix.html
31. Pendregosa, F., Varo-quaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). sklearn.feature_selection.RFE. *Journal of Machine Learning Research. Volume* 12. Pp 2825-2830. https://scikit-learn.org/stable/modules/generated/sklearn.feature_selection.RFE.html
32. Pendregosa, F., Varo-quaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). Support Vector Machines. *Journal of Machine Learning Research. Volume* 12. Pp 2825-2830. https://scikit-learn.org/stable/modules/svm.html
33. Pendregosa, F., Varo-quaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). sklearn.svm.SVC. *Journal of Machine Learning Research. Volume* 12. Pp 2825-2830. https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html#sklearn.svm.SVC

34. Poland, G.A., Jacobson, R.M, & Ovsyannikova, I.G. (2009, May). Trends affecting the future of vaccine development and delivery: The role of demographics, regulatory science, the anti-vaccine movement, and vaccinomics. *Vaccine. Volume* 27 (Issue 22-26). pp 3240-3244. https://www.sciencedirect.com/science/article/pii/S0264410X09000851
35. Ray, S. (2017, September 13). Understanding Support Vector Machine algorithm from examples (along with code). Retrieved from https://www.analyticsvidhya.com/blog/2017/09/understaing-support-vector-machine-example-code/
36. Roberts, M.G., & Tobias, M.I. (1999, November 12). Predicting and preventing measles epidemics in New Zealand: application of a mathematical model. *Epidemiol. Infect. Volume 2000* (Issue 124). pp 279–287. https://www.cambridge.org/core/services/aop-cambridge-core/content/view/3A3F52E5A25CEA2ECD544EE1952A25B2/S0950268899003556a.pdf/predicting_and_preventing_measles_epidemics_in_new_zealand_application_of_a_mathematical_model.pdf
37. Rosenfeld, R. Case Study: Improving Epidemiological Forecasting with the Centers for Disease Control and Prevention (CDC). Retrieved from https://cdsl.cs.cmu.edu/case-studies/computational-biology-and-epidemiology/using-machine-learning-epidemiological
38. Sadalik, A. Kautz, H., & Silenzio, V. (2012, January). Modeling Spread of Disease from Social Interactions. *Proceedings of the Sixth International AAAI Conference on Weblogs and Social Media.* https://www.researchgate.net/publication/255567678_Modeling_Spread_of_Disease_from_Social_Interactions
39. Sem Spirit. (2016). Random Forest Classification. Retreived from http://www.semspirit.com/artificial-intelligence/machine-learning/classification/random-forest-classification/https://www.aaai.org/ocs/index.php/ICWSM/ICWSM12/paper/view/4493/4999
40. Sharma, V., Kumar, A., Panat, L., Karajkhede, G., & Lele, A. (2015, December). Malaria Outbreak Prediction Model Using Machine Learning. *International Journal of Advanced Research in Computer Engineering & Technology. Volume 4* (Issue 12). https://pdfs.semanticscholar.org/2ce3/631949498f6f40cf3b9ab4096c082f1d0047.pdf
41. Skymind. A Beginner's Guide to Neural Networks and Deep Learning. Retrieved from https://skymind.ai/wiki/neural-network
42. Soni, D. (2018, March 12). Introduction to k-Nearest-Neighbors. Retrieved from https://towardsdatascience.com/introduction-to-k-nearest-neighbors-3b534bb11d26
43. Van Der Maaten, L., Postma, E. & Van den Herik, J. (2009). Dimensionality reduction: a comparative review. *J Mach Learn Res*. *Volume* 10. pp 66-71. http://www.math.chalmers.se/Stat/Grundutb/GU/MSA220/S18/DimRed2.pdf
44. Webgraphviz. Webgraphviz is GraphViz in the Browser. Retrieved from http://www.webgraphviz.com/