Jing Du, Minji Kim

BZAN3307: Machine Learning, Prof. VanderWerf

Dec. 4, 2022

**Predicting Online Shopper Purchasing Intent via Machine Learning**

Problem and Client

The e-commerce industry is flourishing, with online retail sales accelerating more than ever following the COVID-19 pandemic. According to the Statista Research Department, revenue from retail e-commerce in the United States in 2022 was estimated at around $906 billion, and it is expected to exceed $1 trillion in 2023, as shown in Exhibit 1 (Chevalier). As individuals become familiar with the virtual shopping experience, more consumers are beginning to buy their products through sites; based on a 2018 survey, about 63% of shopping begins online (Think With Google). Despite the large market opportunity, e-commerce has a different set of challenges than traditional retail. As buyers have almost unlimited access to unique websites, there is much more intense competition for consumers. Furthermore, the industry has a low average conversion rate (around 2-3%); this means that if the website has 100 visitors in a month, only two or three of them complete the desired action, such as subscribing to the service or purchasing the product. Due to advances in data-tracking technology, e-commerce websites can gather information regarding each session, visit, and transaction. Firms can use machine learning to analyze key factors that predict buyer behavior, which will help firms determine the most successful strategies for increasing their conversion rates.

Our client, which we will refer to as Retailer X, has presented us with over 12,000 rows of online shopper data and has requested a highly accurate way of predicting whether or not a customer will complete a purchase, along with a way to determine profiles of likely purchasers and non-purchasers. Our primary audience is managers working in Retailer X’s marketing and sales departments, as their main responsibility is to develop sales strategies that motivate customers to complete their purchases. Our analysis via machine learning is intended to provide data-backed support for enhancing and further customizing Retailer X’s online shopping experience through identification of retailer draws and weaknesses.

Data

Our dataset, titled “Online Shoppers Intention UCI Machine Learning,” was posted on Kaggle and is from a University of Irvine study. The dataset contains 12,330 sessions, with each session representing a different website user in a one-year period. Among them, 10,422 sessions did not have a purchase, whereas 1,908 sessions did involve a purchase. There are 17 independent variables in total, including 10 numeric variables (Administrative, Administrative\_Duration, Informational, Informational\_Duration, ProductRelated, ProductRelated\_Duration, BounceRates, ExitRates, PageValues, SpecialDay) and 7 categorical variables (Month, OperatingSystems, Browser, Region, TrafficType, VisitorType, Weekend). The dependent variable that we tried to predict is Revenue, which represents whether or not the customers complete the purchase.

Before introducing the models, we will explain some of the less obvious variable names. The numeric variable SpecialDay represents the relative closeness of the browsing date to shopping periods surrounding special days or holidays. For example, a value closer to one means that the browsing date was near a special day like Christmas or Thanksgiving, factoring in dynamics like shipping time to determine how far a date must be from the special day to be zero. Because SpecialDay only contained five distinct values (0, 0.2, 0.4, 0.6, 0.8, 1), we decided to treat it as categorical and changed it into a dummy variable for analysis. PageValues is the average of the page values for the pages that a user has viewed before making a transaction. It is a way that Google Analytics measures how relevant visiting a specific page tends to be in prompting a later purchase. Additionally, Administrative refers to the number of pages that a user visits that are related to account management, such as updating passwords or payment information. Informational provides the number of pages that a user visits related to website information, such as the “Contact Us” or “About Us” pages. Administrative\_Duration and Informational\_Duration provide the time in seconds spent by users on the associated pages.

Classification Algorithms with Original Data

We decided to run all six classification algorithms and compare their overall test accuracies and complete purchase (rare event) accuracies to choose the best model for Retailer X’s needs. To run the algorithms, we made the Revenue column a binary variable, with 1 representing “complete purchase” and 0 representing “no purchase”, and we partitioned our data into 70% training and 30% testing.

Logistic regression is the first classification algorithm we implemented. Using all the categorical variables and Special Day as dummy variables, the stepwise logistic regression model returns 12 significant variables (p<=0.05), including PageValues, ExitRates, November, Product-Related Duration, TrafficType 3/8/13, December, May, March, February, and Weekend. With the stepwise logistic regression model, the training data has an accuracy of 88.67%, and it correctly predicts that the customers complete the purchase 40% of the time. In the test set, the model has an overall accuracy of 88.26% and a complete purchase accuracy of 38%.

With the classification tree model, we attempted several different values for the complexity parameters (CPs) from 0.005 to 0.03 at steps of 0.001. As the CP gets smaller, both accuracy and the probability of overfitting increase, so we weighed the extent of accuracy improvement with the overfitting issue when selecting a final CP. With CP from 0.011 to 0.03, the classification trees return the same training accuracy of 90.08%. Lowering CP to 0.005 only increases the accuracy by 0.948%. We were satisfied with the training accuracy at 90.08% with CP of 0.03, and we did not want to risk more overfitting for this small improvement. The model correctly predicts the complete purchase 68% of the time. In the test set, the tree gives an overall accuracy of 89.43% and a complete purchase accuracy of 65%.

Then, we normalized all the numeric variables and used the significant variables from the stepwise logistic regression to train the KNN model. We tried multiple different values for k as the number of neighbors from 2 to 18. KNN with k=2 provides the highest accuracy at 92.61%, whereas KNN with k=3 predicts correctly 92.60% of the time. However, KNN with k=2 gives a 53% complete purchase accuracy, whereas KNN with k=3 has 68% accuracy for complete purchase. Therefore, we decided to use k=3 for our KNN model. In the test set, this model returns an overall accuracy of 88.64% and a complete purchase accuracy of 53%.

Similarly, we changed the number of hidden layers, ranging from 2 to 13, for the neural network. Our analysis shows a general pattern that as the number of layers increases, the model becomes more accurate at predicting the events. We examined the accuracy improvement between each number of layers and found that beyond 11 layers, the training accuracy does not increase much. Therefore, we decided to implement the neural network with 11 hidden layers. This model provides us a training accuracy at 97.74% and a complete purchase accuracy at 69%. In the test set, the overall accuracy is 89.19% and the complete purchase accuracy is 57%.

There is no hyperparameter to be manipulated with the naive Bayes model. Removing all numeric variables and converting all dummies into factors, the model provides overall training accuracy of 82.71%, and the accuracy for complete purchase is 10%. In the test set, the overall accuracy is 83.16% and the complete purchase accuracy is 8%.

The last basic algorithm we implemented is the support vector machine, and we tried multiple different costs from 5 to 80 at steps of 5. We found that beyond a cost of 25, the training accuracy increases at a very small step, so to avoid overfitting, we decided to pick the cost at values between 25 and 30. Cost of 29 provides higher accuracy for complete purchases than the lower costs, so we decided to build the SVM model with a cost of 29. The resulting overall training accuracy is 88.37%, and the complete purchase accuracy is 34%. In the test set, the overall accuracy is 87.94%, and the complete purchase accuracy is 31%.

The comparison table for all six classification algorithms can be found in Exhibit 2a in the appendix. Using both the overall accuracies and complete purchase accuracies in the test set as the indicators of model performance for future prediction, we found that the classification tree with complexity parameter of 0.03 returns the highest accuracies in both criteria. However, this is not the final model that we would recommend to Retailer X, since it does not fully address the rare event problem and overfitting.

Classification Algorithms with Oversampled Data

After considering the real-life situation, we realized that the customers who eventually complete a purchase are as important as the other majority customers who abandon the transactions because these customers increase the profits for the e-commerce websites. However, even the best model so far only provides a rare event accuracy of 65%. Therefore, in the following section, we tried to improve our accuracy on customers who make a purchase.

We began by adding the performance measure of Area Under the Curve (AUC), since this measure is affected more strongly by rare events. Then, we used the oversampling method to duplicate the complete purchase events so that both events get equal attention from the algorithms. After oversampling, we used the six classification algorithms and the same hyperparameters to examine to what extent oversampling improves the rare event accuracies.

For stepwise logistic regression, all the previously significant variables still hold for the oversampled data except for Traffic Type 3. In addition, Other Visitor Type, Administrative Pages, Informational Pages, October, Region 2/5/9, Special Day 0.2/0.6, Traffic Type 2/4/5/10/11/20, Browser 2/3/6, Operating Systems 2/4/7 are also found to be significant. With the oversampled data, the stepwise logistic regression’s overall training accuracy drops to 82.55%. However, the complete purchase accuracy increases to 79% in the training data, and the AUC is 82.60%. In the test set, the AUC is 80.68%, and the complete purchase accuracy is 76%.

With the classification tree and with the same CP of 0.03, the overall training accuracy decreases to 84.99% with an AUC of 85.04%, whereas the complete purchase accuracy increases to 81%. In the test set, the AUC is 84.16%, and the complete purchase accuracy is 80%.

KNN shows a different pattern. With the oversampled data, KNN with k=3 increases its overall training accuracy to 95.29% and has a rare event accuracy of 100%. The training AUC is 95.23%. However, the oversampled KNN model is dramatically overfitted. In the test data, the model returns an overall accuracy of 65.87% and a complete purchase accuracy of 47%, which is even worse than the original model, and the AUC is 65.28%.

Using the oversampled data, the neural network with 11 hidden layers gives a slightly lower overall accuracy of 91.96% in the training data. The rare event accuracy increases to 95% and the training AUC is 91.91%. In the test data, the AUC is 81.21%, and the complete purchase accuracy is 78%. There is some overfitting with the neural network at 11 hidden layers, since there is a large gap between training and test accuracy.

The naive Bayes model returns a training AUC of 65.34% and rare event accuracy of 69%. In the test data, AUC is 63.53%, and the complete purchase accuracy is 66%.

The support vector machine with cost of 29 decreases the overall training accuracy to 84.33% but increases the complete purchase accuracy to 83%. The AUC for training is 84.35%. With test data, SVM gives an AUC of 80.20% and a rare event accuracy of 75%.

The comparison table for all six classification algorithms with the oversampled data can also be found in Exhibit 2b. With the oversampled data, the classification tree still provides the highest overall accuracy, complete purchase accuracy, and AUC in the test data.

Advanced Methods for Overfitting

One problem with oversampling using exact replicas of the rare events is that the probability of overfitting can increase. To ensure that the models with certain hyperparameters will perform consistently across different datasets, we implemented the k-fold cross-validation method to let the algorithms select the best hyperparameters for us in the classification tree, KNN, and neural network.

With the 10-fold classification tree, the algorithm selects the complexity parameter of 0.001096792 to have the highest average accuracy of 87.30% in the k-fold process. This model provides a training AUC of 89.04% and a complete purchase accuracy of 90%. In the test data, the AUC is 84.09%, and the complete purchase accuracy is 82%.

With 10-fold KNN, the algorithm selects the number of neighbors for prediction to be one. The average accuracy across the 10 folds is 94.57%. Training the model on the entire training set, we found the AUC to be 100%, and the model predicts all rare events correctly. However, in the test set, AUC drops dramatically to 62.97%, and the rare event accuracy drops to 34%.

With 10-fold neural network, the algorithm selects 12 from 1 to 13 as the number of layers that provides the highest average accuracy of 88.39%. Trained on the entire training set, the model provides an AUC of 90.46% and rare event accuracy of 93%. The test AUC is 79.65%, and the model correctly predicts 74% of the rare events in the test data.

Since the classification tree is consistently the best model for prediction, we also implemented the random forest from the bagging ensemble model to see if combining multiple trees will produce a better result. While it predicts at 100% accuracy and AUC with the training set, the AUC with the test data is 80.23%, and it only captures 67% of the rare events. Therefore, the oversampled random forest did not outcompete the oversampled k-fold classification tree in terms of test overall accuracy, rare event accuracy, and AUC. Thus, we decided to recommend the oversampled k-fold classification tree above all others to Retailer X.

Our Recommended Model

After comparing the six basic classification models and performing several more advanced machine learning techniques to deal with overfitting and the rare events problem, we determined that the best model for our client would be the oversampled k-fold classification tree with a complexity parameter of 0.001096792. This model is displayed in Exhibit 3. We weighed this decision primarily based on overall accuracy, rare events accuracy, AUC, probability of overfitting, and the ability for our client to quickly develop rules of thumb for online shopper purchasing behavior. Even though the oversampled classification tree with CP of 0.03 provides slightly better AUC, its rare event accuracy is lower than our k-fold tree, and k-fold cross validation provides the advantage of lower probability of fitting the test data by luck.

The entire classification tree is very complicated with its numerous branches. To provide an overview, some of the most prominent variables where splits occur are PageValues, Months, ProductRelated Duration, Informational Duration, ExitRates, BounceRates, and Administrative. The greatest benefit of this model is how it is extremely accurate, in terms of both overall and rare-event-specific accuracy. This would allow our client to examine hypothetical online shoppers and determine with a very high level of accuracy whether they would or would not make a purchase. Another benefit of our chosen model is how our client could develop quick rules of thumb for purchasing behavior, through running hypothetical customers in the classification tree. For instance, a customer who has an average page value lower than 0.034 and buys in either December, February, June, March, May, or October, would have only a 5% probability of making a purchase. To improve this, the client could implement more intensive marketing campaigns during these months and work to direct customers to pages with higher page values.

Additionally, the processed training and test data sets had significantly fewer columns representing independent variables than many of the other models. This model’s processed data had only 17 columns as independent variables, since the classification tree can run without needing to make dummy variables. For instance, in comparison, the basic neural network model required creation of dummy variables and thus had 72 columns as independent variables, so we would have needed to either place much more emphasis on dimension reduction or deal with the issue of very low model efficiency with such a model.

Recommendation Limitations

While we believe our chosen model will best serve the client’s needs, our recommendation has a few limitations that we should keep in mind. To begin, the algorithm is a relatively complex tree with lots of different branches. It can take some time to run through the tree and may be confusing for some clients to use if they are not familiar with introductory statistics and concepts like how to utilize a decision tree. Unfortunately, the classification tree relies on the relationships between independent variables, so it does not clearly show the overall most significant stand-alone variables. Therefore, if Retailer X wanted to identify the significance of variables on their own and apply this to identify potential areas of improvement, a model like stepwise regression would better meet the client’s needs. However, our model meets Retailer X’s main needs of high accuracy and determining overall profiles of particular customers who tend to make online purchases, as the significance of some elements in these profiles often relies upon the presence of other elements.

Another potential limitation with the joint application of oversampling and k-fold cross validation is that the model might be overoptimistic, which means that “the classiﬁcation performance in the test sets will be similar to the one obtained in the training sets, not because the model is able to correctly generalize to the test data, but rather because there are similar patterns in both training and test partitions” (Santos et al., 2018). With cross validation, the training set is partitioned into k folds. The kth fold is left out for testing while the other folds are used to train the model, and the process is repeated for each fold. However, during the k-fold partition, the exact replicas will appear in both the training folds and test fold, leading to the problem of overoptimism. A more logical way to avoid this problem is to conduct oversampling within cross validation so that the replicas appear in the same fold. Unfortunately, we do not have sufficient knowledge with R to perform this step at this point.

Future Research and Improvements

After we acquire sufficient knowledge about how to manipulate steps within k-fold cross validation, we would need to fix the aforementioned issue with oversampling and cross validation. Additionally, it could be interesting to further investigate factors that contribute to a customer leaving the website without making a purchase, rather than focusing on the rare event of clients making purchases. Moreover, if our client could provide information on the non-numerical equivalents of the region, traffic type, operating system, and browser variables, we could provide more in-depth analysis of why certain variables tended to be significant, based on outside research of e-commerce demographic and browsing trends.

Works Cited

Chevalier, Stephanie. “Global Retail e-Commerce Sales 2026.” *Statista*, 21 Sept. 2022, <https://www.statista.com/statistics/379046/worldwide-retail-e-commerce-sales/>

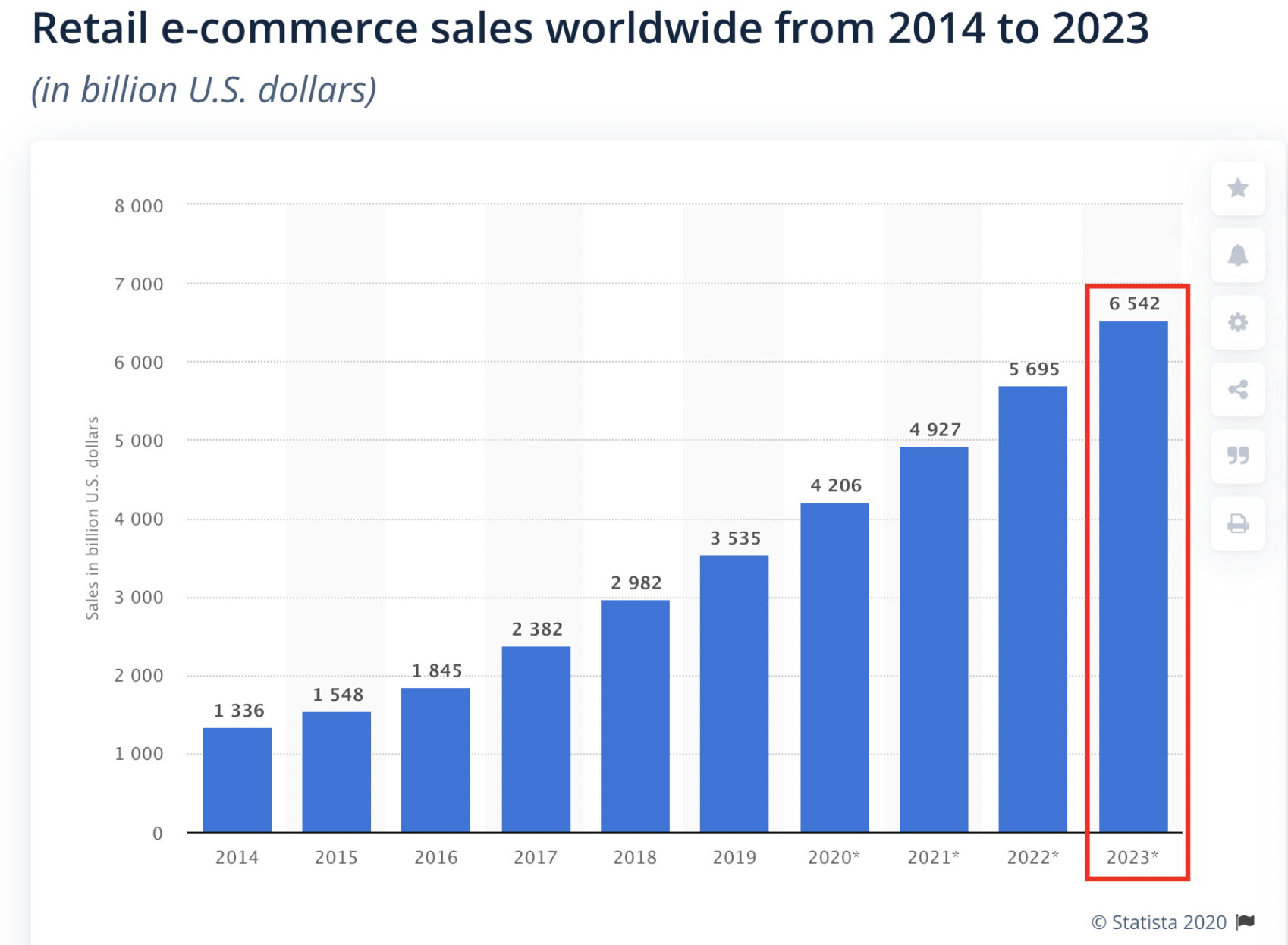
“Online Shopping Behavior Statistics - Think with Google.” *Google*, Google, <https://www.thinkwithgoogle.com/consumer-insights/consumer-trends/online-shopping-beheavior-statistics/>

Sakar, C.O., Polat, S.O., Katircioglu, M. *et al.* Real-time prediction of online shoppers’ purchasing intention using multilayer perceptron and LSTM recurrent neural networks. *Neural Comput & Applic* 31, 6893–6908 (2019). <https://doi.org/10.1007/s00521-018-3523-0>

Santos, Miriam & Soares, Jastin & Henriques Abreu, Pedro & Araujo, Helder & Santos, Joao. (2018). Cross-Validation for Imbalanced Datasets: Avoiding Overoptimistic and Overfitting Approaches. IEEE Computational Intelligence Magazine. 13. 59-76. 10.1109/MCI.2018.2866730.<https://www.researchgate.net/publication/328315720_Cross-Validation_for_Imbalanced_Datasets_Avoiding_Overoptimistic_and_Overfitting_Approaches>

**Appendix**

**Exhibit 1:** Growth in worldwide retail e-commerce sales in the last ten years



**Exhibit 2:** Overall and rare event Accuracies for basic and advanced classification models

Exhibit 2a:

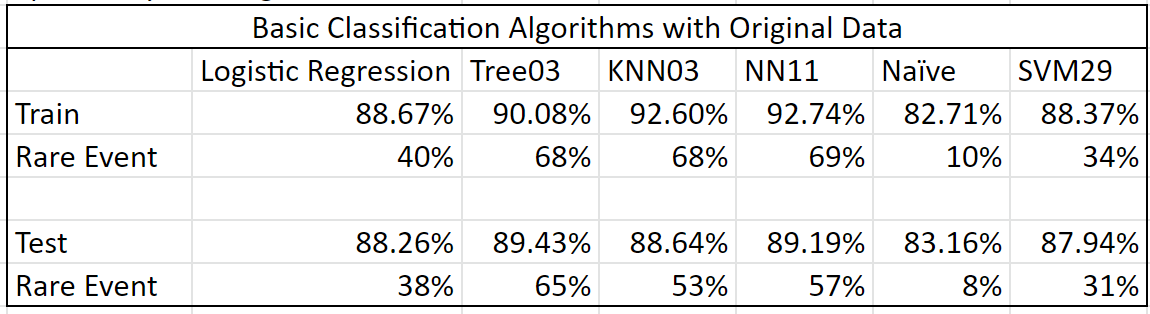


Exhibit 2b:

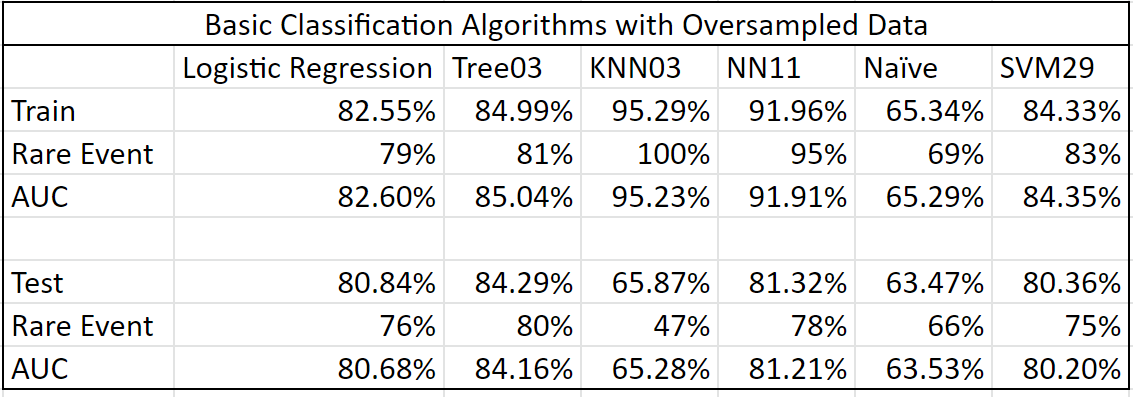


Exhibit 2c:

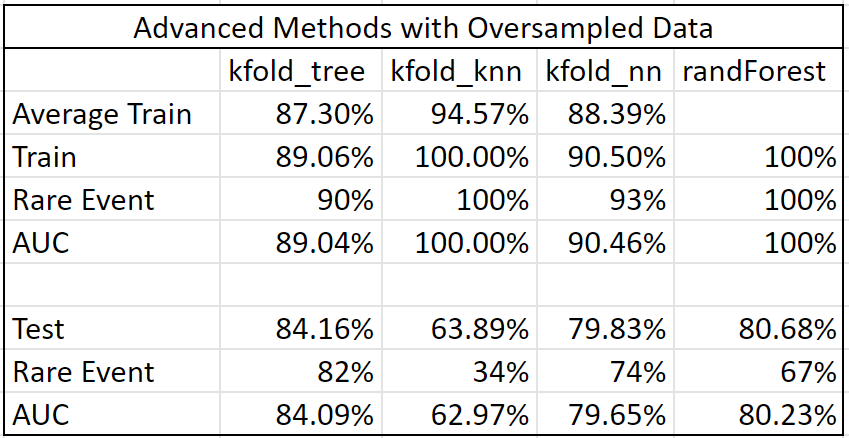


Exhibit 3a: Recommended model, first halfChart, diagram

Description automatically generated

Exhibit 3b: Recommended model, second half