The Effects of Class Label Noise on Highly-Imbalanced Big Data

Robert K. L. Kennedy and Justin M. Johnson and Taghi M. Khoshgoftaar
College of Engineering and Computer Science
Florida Atlantic University
Boca Raton, Florida 33431
rkennedy@fau.edu, jjohn273@fau.edu, khoshgof@fau.edu

***Abstract*—This study explores the effects of class label noise on a highly-imbalanced big data set by injecting varying levels of class noise into a Medicare Part B fraud detection data set. Noise parameters are used to vary the total level of class noise and the proportion of class noise between the majority and minority classes. This allows us to better understand how class noise affects imbalanced data sets and where data cleaning efforts should be focused. Four popular machine learning algorithms are evaluated using six rounds of five-fold cross-validation to determine which learners are most robust to class noise. Area under the precision-recall curve (AUPRC) results shows that negative class noise, i.e. when positive instances are incorrectly labeled as negative, has the greatest adverse effect on classification performance. Statistical results show that the XGBoost learner performs significantly better than Random Forest, Multilayer Perceptron, and Logistic Regression Learners. True positive rates and true negative rates show that there is a trade-off that occurs as the noise proportion switches between the majority and negative classes. Finally, we show that the effects of class label noise can be combatted by regularizing the XGBoost learner through shallower decision trees.**

***Index Terms*—Class Noise, Class Imbalance, Robust Learners, Big Data, Medicare, Fraud Detection**

I. Introduction

In machine learning (ML), class label noise is an important factor of data quality that negatively affects model performance [1]. A noise-free training data set used for supervised machine learning is assumed to have correctly labeled instances. Every instance x (in labeled data) has a true class y and an observed class label $\tilde{y}$. Class noise is defined when $y \neq \tilde{y}$ for a given instance (x, y). Increased levels of class noise can lead to misleading subsets of data, result in larger and more complex learners, and can significantly decrease model prediction and classification performance [1]–[4]. Not all machine learning models handle class noise equally. Some models are inherently more robust to class label noise without additional preprocessing or modified training routines.

Class label noise can occur for a variety of reasons. Noise can be introduced by the tools or methods used to collect the data or label noise can be introduced during the labeling processes [5]. For example, data labeled by human annotators can have instances mislabeled (e.g., human error or any subjectivity during labeling) or data collected by a distributed system (e.g., collection of IoT devices) can have errors introduced due to network communication issues. Frénay and Verleysen [1] define three categories of class label noise, Noise Completely at Random (NCAR), Noise at Random (NAR), and Noise Not at Random (NNAR). NCAR occurs when the label noise is independent of the class label y and data features x, NNAR occurs when noise is dependent on x and y, and NAR occurs when the probability of class noise is dependent on the true class y [1]. NAR is the most common scenario studied. The noise examined in this paper can be categorized as NAR. This type of class noise has been shown to degrade model performance and inference [6], [7].

We use a highly-imbalanced binary classification dataset. Binary datasets are composed of one positive class and one negative class. Class imbalance occurs when one class is significantly larger than the other. In most cases, the positive class of interest is the smaller group, or the minority group, and there are an abundance of negative samples that make up the majority group. Examples include disease diagnosis datasets [8], clinical data [9], image recognition [10], tweet sentiment data [11], and fraud detection datasets [12], [13] like the one used in our experiments. Datasets that are intrinsically imbalanced occur naturally, such as disease diagnosis datasets. Those that are extrinsically imbalanced are ones that are imbalanced due to external factors, such as storage procedures [14]. Imbalanced datasets do not necessarily reduce the model's performance if the imbalance is minimal, and each class is well represented [15]. However, highly-imbalanced datasets, defined as datasets that the minority class represents as few as 0.1% of the total dataset, have much more significant of an effect on model performance and present additional challenges [15], [16].

In this paper, we explore the challenges of class label noise *with* a highly-imbalanced dataset by injecting simulated class label noise into a big Medicare Part B fraud detection dataset. The original dataset is made publicly available by the Centers for Medicaid and Medicare Services (CMS) [17]. We first clean the dataset to remove any existing class label noise that may be present. The original dataset starts with roughly 8.44 million negative instances and 4,334 positive instances. Roughly 50% of the instances are removed during the cleaning process to produce one with approximately 4.22 million negative instances and 4,118 positive instances, or a positive class rate of 0.0975%. The dataset characteristics are captured in Table I. Using this cleaned subset as our base

TABLE I
MEDICARE PART B SUMMARY FOR ORIGINAL AND CLEANED DATASET.

Part B Data Set	# of Positive	# of Negative	Total
Original	4334 (0.0513%)	8,443,052	8,447,386
Cleaned	4118 (0.0975%)	4,221,526	4,225,644
Difference	-216	-4,221,526	-4,221,742

case, we then inject simulated class label noise using two parameters, Λ and Ψ. The total number of instances that class noise is to be applied is defined as Λ and the proportion of noise applied to the two classes is defined as Ψ. We vary these two parameters and create 29 noise-injected datasets. We then train four learners on each of these datasets using five-fold cross validation. Each experiment is repeated six times for statistical purposes. We use the area under precision-recall curve (AUPRC), true positive rate (TPR) and true negative rate (TNR) for our classification performance metrics. Using these, we explore how robust each learner is to varying levels of class label noise. To the best of our knowledge, we are the first to examine the effects of class label noise on a highly-imbalanced big dataset.

II. RELATED WORK

How class label noise affects learners and methods specifically for learning from noisy data has been actively studied in the literature. Frénay and Verleysen present a survey on model classification with class label noise [1]. They categorize methods for dealing with class label noise into two main categories, data-level and algorithm-level techniques. Data-level techniques address the class label noise by filtering out instances with labels that are deemed noisy. Once the noisy instances are cleaned from the dataset a model can be trained on the cleaned data. This type of technique is most popular since it is independent of the model. Examples include distance-based filters [18], [19], ensemble filters [20]–[22], or a hybrid approach [23], [24]. In this paper, we clean our dataset with a technique that would be categorized as data-level.

In contrast, algorithm-level techniques aim to use classifiers that are robust to class label noise such as [25]. Other examples of algorithm-level techniques include neural networks that are robust to class noise [26], [27], generative adversarial networks [28], and modified training of overparameterized neural networks with early stopping [29]. Prati et al. [30] present a survey on class label noise and the challenges of learning from noisy data. They focus on non-binary classification problems and argue that though most of the existing research focuses on binary classification often, real-world problems include multi-class, multi-label, and otherwise go beyond binary classification. However, Wang et al. [31] show that techniques designed for the binary classification problem can be extended to the multi-class problem. Class decomposition can be used to convert a set of binary class problems into a multi-class one. Though there has been significant research in the area of class label noise, research that focuses on class label noise in class-imbalanced data is much scarcer.

Van Hulse and Khoshgoftaar [32] study different approaches for learning from noisy and class-imbalanced data. They consider two aspects of class label noise and introduce a technique for injecting simulated class label noise into the training data. The total class label noise Λ and the proportion of class label noise between a negative class and positive class Ψ across 11 machine learning models and 7 training datasets. They were the first to introduce the Ψ noise parameter and their results show that this parameter has the greatest effect on classification performance, when using the Area Under the Receiver Operating Characteristic Curve (AUC). We expand upon the work by Van Hulse and Khoshgoftaar by extending noise-injection experiments to highly-imbalanced big data classification problems.

III. METHODOLOGY

This paper evaluates the effects of class label noise on a highly-imbalanced big data dataset. Specifically, four machine learning models are evaluated using six rounds of five-fold cross validation on a cleaned Medicare Part B dataset. Simulated class label noise is injected into the training partitions of the cleaned data set, and classification performance is evaluated on the clean test partitions.

A. Data Cleaning

We use a cleaned and pre-processed dataset derived from a publicly available Medicare Part B dataset originally made available by CMS [17]. The original dataset contains services and medical procedures that were performed by Medicare providers from 2012 through 2018. This becomes a binary fraud detection dataset when fraud labels are mapped from the List of Excluded Individuals and Entities (LEIE) [33]. The LEIE contains a list of medical providers that have been banned from providing Medicare services due to confirmed fraud-related infractions.

We aim to start with a class-label-noise free dataset in which simulated class label noise can be injected into for evaluation. The cleaned dataset (a subset of the original dataset, $D_c \in D$) is created by using an ensemble classifier noise ranking technique. A random forest (RF) classifier is trained on D using five-fold cross validation. The binary class probability predictions from the hold-out partitions are ranked. The $K\%$ lowest probabilities are removed from each of the two classes. K is independent for each class. K was varied from 0.0 to 0.5 and we determined K to be 0.5 and 0.05 for the negative and positive class, respectively, such that a new RF classifier produced an AUC ≥ 0.95. The number of instances in each class is summarized in Table I.

B. Noise Injection

The cleaned dataset is now used as a baseline dataset for the four learners and as a baseline for the noise injection to effectively explore the effects of class label noise. The noise injection procedure used in this paper was introduced by Van Hulse et al. [32]. This procedure simulates various types of class label noise that exist in real applications. How much noise and to which class the noise is added is varied. Two noise parameters are introduced, Λ and Ψ, to vary the level of simulated noise injected into each class and the proportion of noise in each class, respectively. They are both expressed as a percentage from 0-100% or 0.0-1.0. We use a binary classification dataset for this work. Thus, when a given instance has class label noise applied to it, the label is flipped from positive to negative or vice versa.

The number of instances class label noise is applied to is given by Equation 1, where $\Lambda \in \{0.0, 0.1, 0.2, 0.3, 0.4, 0.5\}$ and S^p is the number of instances in the positive class.

$$2 \times \Lambda \times S^p \tag{1}$$

The number of randomly selected instances from the positive class and negative class, which the class label noise is applied, is given in Equation 2 and 3 respectively, where $\Psi \in \{0.0, 0.25, 0.50, 0.75, 1.0\}$:

$$2 \times \Lambda \times S^p \times \Psi \tag{2}$$

$$2 \times \Lambda \times S^p \times (1 - \Psi) \tag{3}$$

C. Performance Evaluation

Four learners, XGBoost, an RF learner, Logistic Regression (LR) and a Multilayer Perceptron (MLP), are trained on the varying levels of noise-injected data using five-fold cross-validation and replicated six times each for statistical significance. We use AUPRC, TPR, and TNR as our classification performance metrics. When reporting noise-injection performance, we selected the AUPRC metric over the AUC (AUC was used in the data cleaning process) metric because related works have shown that the AUPRC metric is more informative than the AUC metric when comparing the classification performance of multiple classifiers on imbalanced data sets [34]. Unlike the AUC metric, the AUPRC metric incorporates the model's precision performance, and the precision is significantly more sensitive to false positives when data is highly imbalanced. We selected a threshold for calculating TPR and TNR that optimizes for the G-Mean (geometric mean) on the training folds. The thresholding seeks to produce a model with TPR and TNR as similar as possible but favoring TPR such that TPR is $\geq$ TNR. Favoring TPR while keeping the TPR and TNR as balanced as possible is beneficial for the fraud detection problem [35]. For statistical significance, we use Tukey's Honestly Significant Difference (HSD) test to measure the statistical significance between the different learner's classification performance and, thus, robustness to class label noise. We use a 99% confidence interval for our Tukey HSD test ($\alpha = 0.01$).

TABLE II
LEARNER AUPRC HSD RESULTS

Learner	Average AUPRC	HSD Group
XGB	0.8199	a
RF	0.7725	b
MLP	0.7488	c
LR	0.5839	d

IV. Results and Analysis

We present the AUPRC performance results as Λ and Ψ are varied across the four learners, shown in Figure 1. Table II presents the HSD test for the learners' average performance and shows that XGBoost is the best performer. Additionally, XGBoost has a relatively short training time on highly-imbalanced data [36]. Hence, we focus on the XGBoost learner when presenting the rest of the experimental results, including the TNR and TPR results. However, the LR, MLP, and RF produced similar patterns as XGBoost as shown in Figure 2, though consistently lower performance than the latter. It is also important to note all experiments where Λ = 0.5 and Ψ = 1.0 were omitted because in this edge case, the noise injection procedure would create a dataset that would consist of only one label (i.e. there is nothing for the models to learn if all labels are the same).

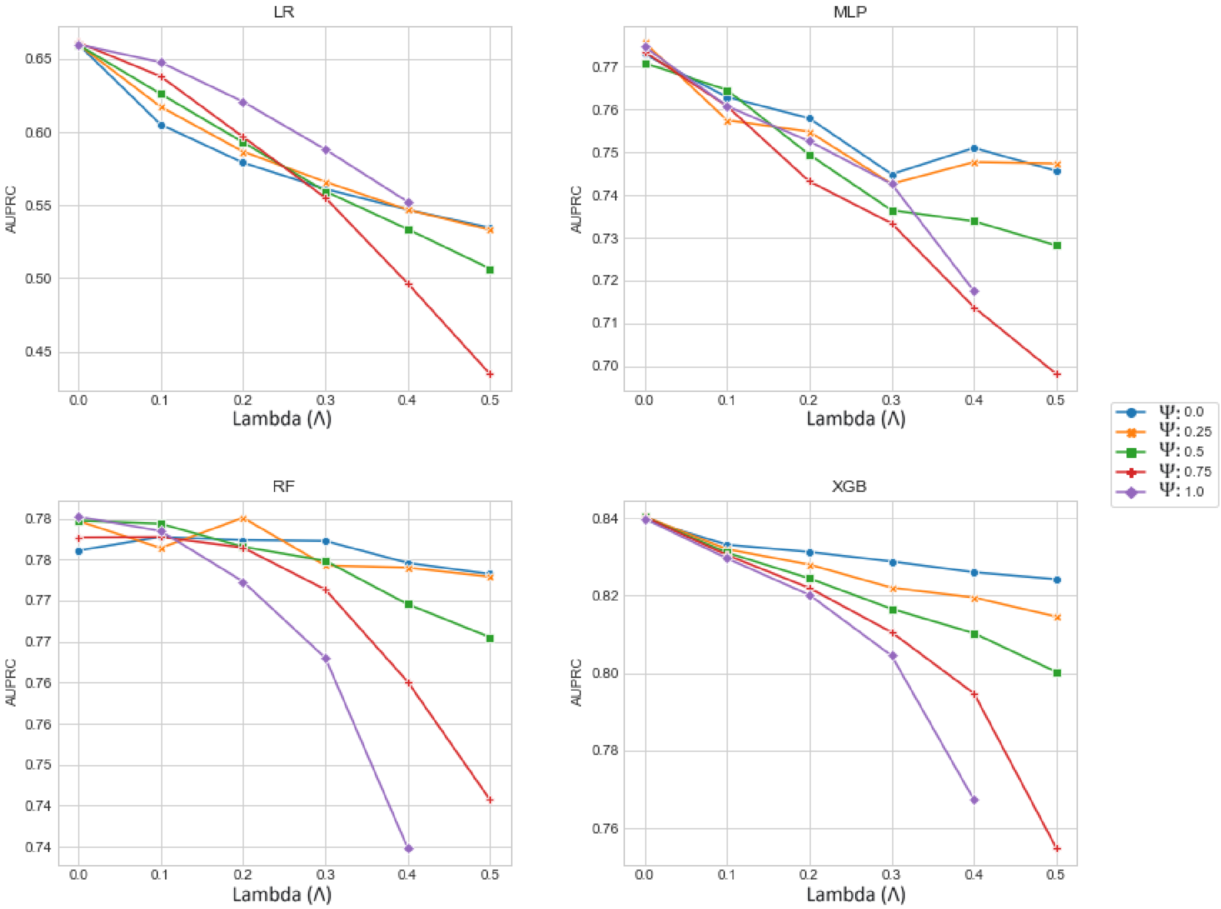
Of the four learners used, it is important to note that the RF used to evaluate its robustness to class label noise is a different RF than the one used in the data cleaning steps. The RF learner was built with 100 trees and a maximum tree depth of 8. The LR learner was built with no more than 200 iterations. The MLP architecture is 2 layers deep, 32 neurons wide using the ReLu (rectified linear unit) activation function, a learning rate of 0.001 and a dropout rate of 0.5. The XGBoost was created with 100 estimators, and a maximum depth of 8. RF and LR were created using scikit-learn [37], the MLP was created in Keras [38], and XGBoost was created using the XGBoost framework [39]. All other parameters were kept as library defaults.

A. Effects of Λ

The level of total class noise injected into the training data is examined. Figure 1 presents the experimental results, measured in AUPRC, as Λ increases from 0.0 to 0.5 for each of the four learners. When Λ = 0 each learner trains on only the cleaned dataset, i.e., no noise has been injected. This baseline case shows that the RF and MLP perform most similarly, LR is the least performant, and XGBoost performs the best with no noise. When measuring TPR and TNR the no-noise baseline case has the highest TNR with a balanced TPR, as can be seen in Figure 2. However, as the level of noise increases, TNR decreases significantly while the TPR is relatively less affected (this is dependent on Ψ).

At Λ = 0.1, the AUPRC performance for LR, MLP, and XGBoost start to reduce. This trend continues for the remainder of the experiments. However, RF, generally, does not reduce in performance until $\Lambda \geq 0.2$. From that point it trends similar

Fig. 1. Noise Levels and AUPRC Performance



to the other learners. As the total class label noise is increased, the learner performance consistently reduces. The rate at which the performance degrades (the general slope of the graphs) for RF and XGBoost are somewhat similar and the rate for LR and the MLP are somewhat similar, with respect to an increasing class label noise. Finally, the level of class noise is at a maximum when Λ = 0.5, where all learners perform their worst.

A Tukey's HSD test for the average AUPRC across Λ and Ψ values is presented in Table II. The averages show that the XGBoost performs significantly better than the other three learners overall and is most resistant to class label noise with respect to AUPRC. The HSD groupings show that rankings of the learners are all statistically significant from each other.

B. *Effects of* Ψ

The proportion of the total class noise applied to the two classes in the training data is examined. When Ψ = 0, all noise is applied to the negative (majority) class, when Ψ = 0.5 the number of instances noise is applied to is equal between the two classes, and when Ψ = 1.0 all the noise is applied to the positive (minority) class. In this dataset, the positive class contains the instances with a fraud label and the instances with a no-fraud label are in the negative class.

For a given level of Λ, it would be expected that a learner's classification performance would reduce as the proportion of class noise in the minority class is increased when training with a highly-imbalanced dataset. However, only XGBoost shows this behavior consistently for all levels of class imbalance, when measuring AUPRC (see Figure 1). RF exhibits this behavior only with the higher levels of class noise and the MLP learner exhibits this behavior the least. However, with higher levels of class noise, there is a clear distinction between low and high Ψ levels for MLP, RF and XGB. For example, when Λ = 0.4 and $\Psi \in \{0.0, 0.25\}$ all four learners perform significantly better than when Λ = 0.4 and $\Psi \in \{0.75, 1.0\}$. This would also hold true for Λ = 0.5, but 0.5 and 1.0 experiments were omitted, as previously mentioned. Another point of interest is how similar the AUPRC is for $\Psi \in \{0, 0.25\}$ across all levels of noise for the LR, MLP and RF learners. The performance metrics between these two are more similar than the other levels of Ψ. This is likely because at the low levels of Ψ, most of the class label noise is injected into the negative class. In this dataset, this class has the vast majority of samples, as seen in Table I. LR performs best when Ψ = 1.0, which is interesting because that is when all minority class labels have noise (i.e. they are all mislabeled) suggesting it is better at dealing with high levels of minority

Fig. 2. XGB Classification Results Measured as TPR (left) and TNR (right).

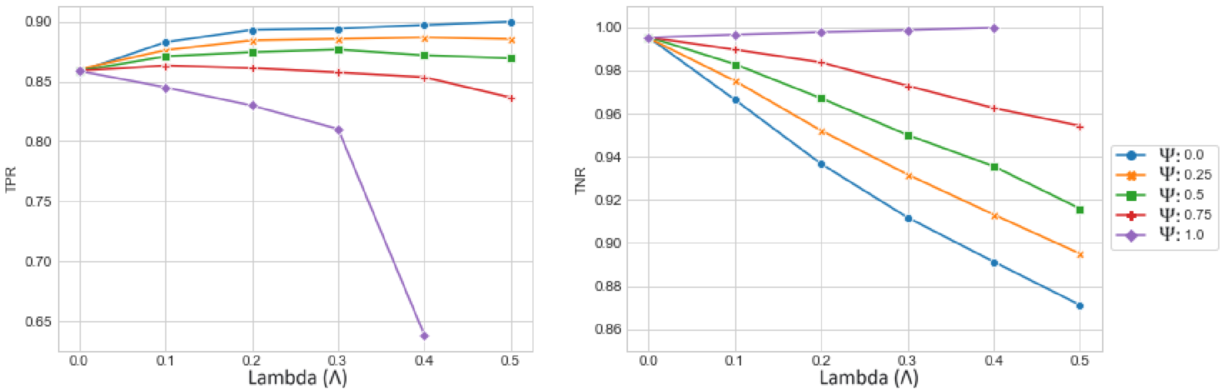
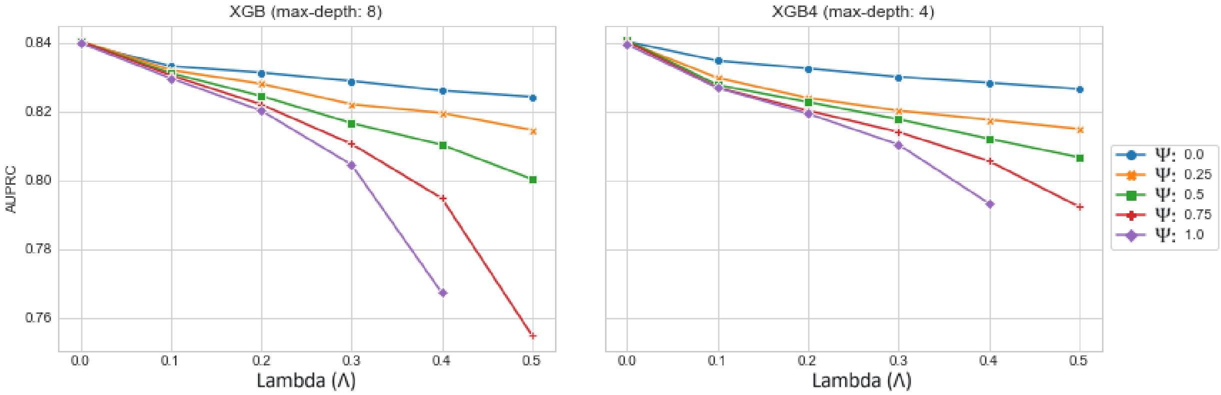


Fig. 3. Regularized XGB AUPRC Results



class labels in a highly imbalanced dataset. The LR learner's performance, at $\Lambda = 0.1$, are opposite to what is expected, i.e. as Ψ decreases performance decreases. However, LR is overall the lowest performing learner in this paper.

Figure 2 illustrates the trade-off between TPR and TNR rates across Λ and Ψ levels using the XGBoost learner. As the overall level of noise increases, the TPR and TNR rates are generally inversely proportional. One exception is when $\Psi = 0.75$ and $\Lambda = 0.4$. At this point both the TPR and TNR rates get worse and total class noise increases. For $\Psi \in \{0.5, 0.25, 0.0\}$ the inverse relation is clear. Though the TPR increase is slight, this shows that when most of the class noise is in the majority class XGBoost is robust against class noise, with respect to TPR. However, this comes at the cost of TNR. Maximizing TPR when detecting fraudulent medical providers is not the most optimal when considering the cost of TNR. As TNR decreases, more fraudulent cases go by undetected. This suggests that efforts should be directed to reducing noise in the negative class when using TNR as a guiding performance metric.

In an effort to improve classification performance and robustness to class label noise, an additional XGBoost learner (with a smaller maximum tree depth of 4) was trained. Figure 3 shows that the XGBoost's classification performance is significantly improved when reducing the maximum tree depth from 8 to 4. This holds true for all values of Ψ and Λ but most noticeably with higher Ψ values. For example, at the highest level of class noise, $\Lambda = 0.5$, there is slight improvement in AUPRC when $\Psi = 0$ but the level of improvement increases as Ψ increases. When $\Psi = 0.75$, or when 75% of the minority class labels are noisy, the smaller XGBoost learner has the largest AUPRC improvement over the larger one. This shows that XGBoost learner with a smaller max tree depth is significantly more robust to class label noise than a larger one. The deeper trees are more prone to overfitting to the class noise and the shallower trees overfit less and generalize to the clean test data better. Additionally, the smaller learner is most robust when the minority class has most of the class noise. This is advantageous for Medicare fraud detection because it suggests that that the quality of class labels for the class that is most scarce is less important, with respect to the smaller XGBoost tree depth.

V. Conclusion

Our work examined the effects of class label noise in a highly-imbalanced big dataset. We used a Medicare Part B fraud detection dataset with over 8 million instances and an imbalance rate of 0.0975%. The dataset was cleaned of any prior class label noise using an ensemble classifier noise ranking technique, and then simulated class label noise was injected for evaluation on four ML learners: Logistic Regression, Random Forest, an MLP, and XGBoost.

Our results show that the combination of class label noise and a highly-imbalanced datasets presents its own challenges. This is observed through a significant reduction in classification performance across all levels of noise, all proportions of noise, and across all learners, as measured by AUPRC, TPR, and TNR. The LR learner exhibited interesting behavior in which it seemed more resilient to noise in the minority class when noise levels are low (not observed in other learners). However, it had the lowest AUPRC by a significant margin for all results. The RF and MLP learners had similar AUPRC results to each other with the RF being more affected by which class the noise was injected to. XGBoost had the highest AUPRC by a significant margin. As the overall noise level increased, the AUPRC reduced significantly. XGBoost did have the most consistent reduction in performance as both noise parameters were increased (e.g., the performance is consistently affected by both the level of class noise and the proportion of noise between the two classes). Lastly, as measured in TPR and TNR, the XGBoost results show that its robustness to class label noise can be improved by reducing the complexity of the architecture. Future work in replicating these experiments using more ML models and further exploring potential ways to increase model robustness would be worthwhile.

Acknowledgment

The authors would like to thank the reviewers in the Data Mining and Machine Learning Laboratory at Florida Atlantic University.

References

[1] B. Frénay and M. Verleysen, "Classification in the presence of label noise: a survey," *IEEE transactions on neural networks and learning systems*, vol. 25, no. 5, pp. 845–869, 2013.

[2] C. Seiffert, T. M. Khoshgoftaar, J. Van Hulse, and A. Folleco, "An empirical study of the classification performance of learners on imbalanced and noisy software quality data," *Information Sciences*, vol. 259, pp. 571–595, 2014.

[3] J. L. Leevy, T. M. Khoshgoftaar, R. A. Bauder, and N. Seliya, "A survey on addressing high-class imbalance in big data," *Journal of Big Data*, vol. 5, no. 1, p. 42, 2018.

[4] R. A. Bauder and T. M. Khoshgoftaar, "A study on rare fraud predictions with big medicare claims fraud data," *Intelligent Data Analysis*, vol. 24, no. 1, pp. 141–161, 2020.

[5] D. F. Nettleton, A. Orriols-Puig, and A. Fornells, "A study of the effect of different types of noise on the precision of supervised learning techniques," *Artificial intelligence review*, vol. 33, no. 4, pp. 275–306, 2010.

[6] J. Zhang and Y. Yang, "Robustness of regularized linear classification methods in text categorization," in *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, 2003, pp. 190–197.

[7] K. M. Ali and M. J. Pazzani, "Error reduction through learning multiple descriptions," *Machine learning*, vol. 24, no. 3, pp. 173–202, 1996.

[8] R. B. Rao, S. Krishnan, and R. S. Niculescu, "Data mining for improved cardiac care," *Acm Sigkdd Explorations Newsletter*, vol. 8, no. 1, pp. 3–10, 2006.

[9] A. N. Richter and T. M. Khoshgoftaar, "Building and interpreting risk models from imbalanced clinical data," in *2018 IEEE 30th International Conference on Tools with Artificial Intelligence (ICTAI)*. IEEE, 2018, pp. 143–150.

[10] M. Kubat, R. C. Holte, and S. Matwin, "Machine learning for the detection of oil spills in satellite radar images," *Machine learning*, vol. 30, no. 2, pp. 195–215, 1998.

[11] J. D. Prusa, T. M. Khoshgoftaar, and N. Seliya, "Enhancing ensemble learners with data sampling on high-dimensional imbalanced tweet sentiment data," in *The Twenty-Ninth International Flairs Conference*, 2016.

[12] W. Wei, J. Li, L. Cao, Y. Ou, and J. Chen, "Effective detection of sophisticated online banking fraud on extremely imbalanced data," *World Wide Web*, vol. 16, no. 4, pp. 449–475, 2013.

[13] M. Herland, T. M. Khoshgoftaar, and R. A. Bauder, "Big data fraud detection using multiple medicare data sources," *Journal of Big Data*, vol. 5, no. 1, pp. 1–21, 2018.

[14] H. He and E. A. Garcia, "Learning from imbalanced data," *IEEE Transactions on knowledge and data engineering*, vol. 21, no. 9, pp. 1263–1284, 2009.

[15] B. Krawczyk, "Learning from imbalanced data: open challenges and future directions," *Progress in Artificial Intelligence*, vol. 5, no. 4, pp. 221–232, 2016.

[16] G. M. Weiss, "Mining with rarity: a unifying framework," *ACM Sigkdd Explorations Newsletter*, vol. 6, no. 1, pp. 7–19, 2004.

[17] Centers For Medicare & Medicaid Services. (2019) Medicare provider utilization and payment data. [Online]. Available: https://www.cms.gov/research-statistics-data-and-systems/statistics-trends-and-reports/medicare-provider-charge-data

[18] D. L. Wilson, "Asymptotic properties of nearest neighbor rules using edited data," *IEEE Transactions on Systems, Man, and Cybernetics*, no. 3, pp. 408–421, 1972.

[19] I. Tomek, "An experiment with the edited nearest-neighbor rule," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. SMC-6, no. 6, pp. 448–452, 1976.

[20] S. Verbaeten and A. Van Assche, "Ensemble methods for noise elimination in classification problems," in *International workshop on multiple classifier systems*. Springer, 2003, pp. 317–325.

[21] T. M. Khoshgoftaar and P. Rebours, "Improving software quality prediction by noise filtering techniques," *Journal of Computer Science and Technology*, vol. 22, no. 3, pp. 387–396, 2007.

[22] J. A. Sáez, M. Galar, J. Luengo, and F. Herrera, "Inffc: an iterative class noise filter based on the fusion of classifiers with noise sensitivity control," *Information Fusion*, vol. 27, pp. 19–32, 2016.

[23] J. S. Sánchez, R. Barandela, A. I. Marqués, R. Alejo, and J. Badenas, "Analysis of new techniques to obtain quality training sets," *Pattern Recognition Letters*, vol. 24, no. 7, pp. 1015–1022, 2003.

[24] J. Koplowitz and T. A. Brown, "On the relation of performance to editing in nearest neighbor rules," *Pattern Recognition*, vol. 13, no. 3, pp. 251–255, 1981.

[25] N. Lawrence and B. Schölkopf, "Estimating a kernel fisher discriminant in the presence of label noise," in *18th International Conference on Machine Learning (ICML 2001)*. Morgan Kaufmann, 2001, pp. 306–306.

[26] B. Han, Q. Yao, X. Yu, G. Niu, M. Xu, W. Hu, I. Tsang, and M. Sugiyama, "Co-teaching: Robust training of deep neural networks with extremely noisy labels," *arXiv preprint arXiv:1804.06872*, 2018.

[27] J. M. Johnson and T. M. Khoshgoftaar, "The effects of data sampling with deep learning and highly imbalanced big data," *Information Systems Frontiers*, vol. 22, no. 5, pp. 1113–1131, 2020.

[28] T. Kaneko, Y. Ushiku, and T. Harada, "Label-noise robust generative adversarial networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 2467–2476.

[29] M. Li, M. Soltanolkotabi, and S. Oymak, "Gradient descent with early stopping is provably robust to label noise for overparameterized neural networks," in *International conference on artificial intelligence and statistics*. PMLR, 2020, pp. 4313–4324.

[30] R. C. Prati, J. Luengo, and F. Herrera, "Emerging topics and challenges of learning from noisy data in nonstandard classification: a survey beyond binary class noise," *Knowledge and Information Systems*, vol. 60, no. 1, pp. 63–97, 2019.

[31] S. Wang and X. Yao, "Multiclass imbalance problems: Analysis and potential solutions," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 42, no. 4, pp. 1119–1130, 2012.

[32] J. Van Hulse and T. M. Khoshgoftaar, "Knowledge discovery from imbalanced and noisy data," *Data & Knowledge Engineering*, vol. 68, no. 12, pp. 1513–1542, 2009.

[33] Office of Inspector General. (2019) Leie downloadable databases. [Online]. Available: https://oig.hhs.gov/exclusions/exclusions_list.asp

[34] T. Saito and M. Rehmsmeier, “The precision-recall plot is more informative than the roc plot when evaluating binary classifiers on imbalanced datasets,” *PloS one*, vol. 10, no. 3, p. e0118432, 2015.

[35] J. M. Johnson and T. M. Khoshgoftaar, “Thresholding strategies for deep learning with highly imbalanced big data,” in *Deep Learning Applications, Volume 2*. Springer, 2021, pp. 199–227.

[36] J. Hancock and T. M. Khoshgoftaar, “Performance of catboost and xgboost in medicare fraud detection,” in *2020 19th IEEE International Conference on Machine Learning and Applications (ICMLA)*. IEEE, 2020, pp. 572–579.

[37] scikit-learn, “scikit-learn,” http://scikit-learn.org/stable/.

[38] F. Chollet *et al.*, “Keras,” https://keras.io, 2015.

[39] T. Chen and C. Guestrin, “Xgboost: A scalable tree boosting system,” in *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, 2016, pp. 785–794.