2018 IEEE International Conference on Information Reuse and Integration for Data Science

Identifying Medicare Provider Fraud with Unsupervised Machine Learning

Richard A. Bauder
Florida Atlantic University
Boca Raton, Florida, USA
rbauder2014@fau.edu

Raquel C. da Rosa
Florida Atlantic University
Boca Raton, Florida, USA
rrosa1@fau.edu

Taghi M. Khoshgoftaar
Florida Atlantic University
Boca Raton, Florida, USA
khoshgof@fau.edu

***Abstract*—**

With the increasing number of people ages 65 and older, healthcare programs are being relied on more for quality and affordable care. Given these and other factors, healthcare spending continues to increase, particularly for the elderly. Medicare is one such program affected by the aging population. Fraud in the United States (U.S.) Medicare program is an ongoing issue resulting in higher healthcare costs for beneficiaries. In this paper, we present an empirical study of several unsupervised machine learning methods to detect outliers, indicating fraudulent medical providers, using the Medicare Part B big dataset. We employ two methods, Isolation Forest and Unsupervised Random Forest, which have not previously been used for the detection of Medicare fraud, along with more commonly used methods to include Local Outlier Factor, autoencoders, and k-Nearest Neighbors. In order to validate the fraud detection performance of each method, we use the List of Excluded Individuals/Entities (LEIE) database which contains information on excluded providers. Moreover, we present details on processing the Part B data and incorporating the LEIE fraud labels. Our results indicate that Local Outlier Factor is the best outlier detection method and k-Nearest Neighbors, with 5 neighbors, and autoencoders are the worst at detecting Medicare Part B fraud.

Keywords: Medicare Part B, Outlier Detection, LEIE, Medicare Fraud, Unsupervised Machine Learning

I. Introduction

Healthcare in the U.S. is vital for the health and well-being of the general population but is becoming increasingly important for the 65 and older demographic. This is due to the rising number of elderly in the U.S. which has increased steadily since the 1960's and is expected to double from 46 million in 2015 to 98 million by 2060 [29]. Therefore, access to affordable and comprehensive healthcare for the elderly population is critical for productivity and wellness in these later years. Medicare is a subsidized U.S. government program providing insurance to over 54.3 million beneficiaries over the age of 65 or younger individuals with specific medical conditions and disabilities [3]. Because Medicare is a subsidized federal program, it is not a functioning health insurance market in the same way as private insurance companies. In 2016, U.S. healthcare spending reached $3.3 trillion, an increase of 4.3% from 2015 [33]. The Centers for Medicare & Medicaid Services (CMS) project spending will increase by 5.5% annually from 2017 to 2018, which would encompass 19.7%, or $5.7 trillion, of the U.S. economy by 2026 [32]. Of this, Medicare makes up about 20% of U.S. healthcare spending at $672.1 billion in 2016 [33]. CMS cites several reasons for the increased Medicare spending including increased enrollment, higher costs for medical goods and services, and more disposable personal income [33]. With such high levels of spending associated with Medicare, and healthcare in general, this area is a potential target for fraudulent activities. The Federal Bureau of Investigations (FBI) estimates that fraud accounts for 3-10% of all medical costs [31]. With Medicare alone accounting for 20% of healthcare spending, the detection of fraud can lead to substantial savings for U.S. healthcare programs and its beneficiaries.

To make medical-related data more accessible to help combat fraud, CMS released several Medicare datasets ranging from provider claims to prescription information. For our paper, we use the *Medicare Provider Utilization and Payment Data: Physician and Other Supplier* data, also known as Medicare Part B, which includes information on services provided to Medicare beneficiaries by physicians and other healthcare professionals. The Medicare data includes the 2012 to 2015 calendar years, with the 2015 dataset being released in June 2017. With this information, the detection of outliers, or anomalies, in provider claims can be used to flag potentially fraudulent activities [10]. This method would use a provider's procedures performed and associated payments to assess behavior indicating normal behavior or potential fraud. The providers flagged as possibly fraudulent can then be investigated further to assess culpability, thus reducing the time and number of resources needed by narrowing down the providers that require additional investigation. In our study, we perform an empirical evaluation of several unsupervised machine learning approaches to detect possible Medicare Part B fraud. Since the Part B dataset does not contain indicators or labels for fraud, we incorporate the Office of Inspector General's (OIG) List of Excluded Individuals/Entities (LEIE) database [24]. These fraud labels allow us to validate fraud detection performance. The interested reader can find more information on Medicare and Medicare fraud in [11], [38].

We employ two unsupervised machine learning algorithms, Isolation Forest and Unsupervised Random Forest, which, to the best of our knowledge, have not been used for the detection of Medicare fraud. Additionally, three other more commonly used methods (Local Outlier Factor, autoencoder,

978-1-5386-2659-7/18/$31.00 ©2018 IEEE
DOI 10.1109/IRI.2018.00051

and k-Nearest Neighbors) are incorporated into our study for a comprehensive view of outlier detection performance. For each method, we run several configurations, such as the number of trees or neighbors used. We validate and compare each method using the aforementioned fraud labels to produce a Receiver Operating Characteristic (ROC) curves and corresponding Area Under the ROC Curve (AUC). We also discuss each methods sensitivity and specificity, in relation to optimal decision thresholds along the ROC curves. The aim is to assess outlier detection method performance in identifying fraud in Medicare Part B data. Our results indicate that Local Outlier Factor outperformed all other methods with an AUC of 0.62985, with k-Nearest Neighbors (using 5 neighbors), autoencoder, and Isolation Forest being the worst performers based on AUC. The main contributions of our empirical study are as follows:

- Detail Medicare Part B big data processing (with over 37 million claims) and the integration of the LEIE for fraud labels.
- Assess the previously untested Isolation Forest and Unsupervised Random Forest methods for fraud detection on the Medicare Part B data and compare these with other more commonly used methods.
- Use real-world fraudulent providers as fraud labels to validate fraud detection performance with AUC as well as sensitivity and specificity at optimal decision thresholds.

To the best of our knowledge, there are no other studies using Isolation Forest (IF) and Unsupervised Random Forest (URF) on big data to detect Medicare fraud. We also use these results, with those of Local Outlier Factor (LOF), autoencoder (AE), and k-Nearest Neighbors (KNN), and compare performance with fraud labels from known excluded providers.

The rest of the paper is organized as follows. In Section II, we discuss works related to our current research. In Section III, Medicare Part B and LEIE data are detailed, along with the integration of fraud labels. In Section IV, we present the results of our research and discuss the fraud detection performance. Finally, Section V presents our conclusions and possible avenues for future work.

II. Related Works

There have been other studies related to the detection of general healthcare fraud [22], [45]. Given this brevity in available studies across the healthcare domain, we focus our literature review on studies pertaining to unsupervised machine learning approaches with Medicare-related data for fraud detection. The number of available studies in the area of Medicare fraud detection are limited. In a study by Shan et al. [40], the authors applied LOF in the public health service management field. Even though they did not use Medicare data, they assessed the use of outlier detection on medical specialist groups to discover inappropriate billing. In order to validate the LOF results, a domain expert was employed for evaluation. The authors suggest that LOF is effective in identifying inappropriate billing patterns and for monitoring billing compliance. Burr et al. [15] used unsupervised machine learning techniques and multivariate outlier detection methods on Medicare claims data, provided by Florida's National Claims History. The authors compared methods to include Mahalanobis distance (with and without dimension reduction), KNN, density estimation methods, and Spearman rank ordering. They found that no methods agreed on suspicious providers, based on comparisons using a list of known fraudulent providers. KNN was also used by Weiss et al. [46] to predict target variables for prescriptions. The authors showed that the prediction of actual outcomes from peer profiles is significantly better than chance. They also found that for the 10% of physicians that prescribed the most medications, there were significant differences between the predicted and actual outcomes indicating possibly suspicious behaviors.

Works employing either IF, URF, or AE are uncommon, but particularly so for Medicare-related fraud detection. A study by Liu et al. [28] is the only work discussing IF and outlier detection with Medicare, Medicaid, and healthcare data. Even so, the authors do not provide experiments or results using IF to detection Medicare fraud. They do discuss a tool, known as the Xerox Program Integrity Validator (XPIV), which uses an interactive graph analysis technique to detect fraudulent activities. Their case study used Medicaid datasets to assess and demonstrate the XPIV tool. Random Forest has been widely used for prediction and feature selection; however, the use of Random Forest as an unsupervised model is limited. We did not find any related works on fraud detection with Medicare-related data.

Our previous research includes a preliminary study that compares several supervised and unsupervised methods to detect Medicare fraud can be found in [8]. In this study, we detect Medicare fraud in a comparative study with supervised (Gradient Boosted Machine, Random Forest, Deep Neural Network, and Naive Bayes), unsupervised (autoencoder, Mahalanobis distance, KNN, and LOF), and hybrid (multivariate regression and Bayesian probability) machine learning approaches. Four performance metrics, oversampling, and an 80:20 (majority:minority) class undersampling method were assessed during our experiments. We split the 2015 Medicare Part B data into separate provider types, with fraud labels from the LEIE database. Results indicated that the successful detection of fraudulent providers is possible, with the 80:20 sampling method demonstrated to be the best performer across all learning approaches. As expected, supervised methods performed better than unsupervised or hybrid approaches, with results fluctuating based on the sampling technique used and provider type. This includes the 80:20 sampling being better than oversampling across all provider types, and more specialized provider types, such as neurology, performing significantly better than more general provider types, such as family practice. Additionally, we have two studies [6], [9] that explore Medicare fraud detection using other unsupervised approaches. We propose a two-step approach in detecting Medicare fraud. Multivariate regression and Bayesian modeling are used to establish a baseline for expected Medicare payments per provider type. This baseline is then used as the normative case in which to compare the actual payment amounts, with deviations flagged as outliers.

The use of outlier detection methods on Medicare datasets is very limited. There are no studies that discuss experiments or results using IF or URF on detecting Medicare fraud, and

studies with AE are limited to our research group. We differ from the known related works in providing the only study to use and compare multiple unsupervised machine learning methods to detect Medicare Part B fraud. Our research is also one of the only such works to use known excluded providers as fraud labels to validate the fraud detection performance of unsupervised methods.

III. Methodology

In this section, we detail the Medicare Part B and LEIE datasets to include data processing and the mapping of fraud labels. Additionally, the outlier detection methods and ROC/AUC performance metrics are described.

A. Data

We use the 2012 to 2015 *Medicare Provider Utilization and Payment Data: Physician and Other Supplier* which is publicly available from the Center for Medicare and Medicaid Services [2]. Herein, we refer to this data as Medicare or Part B. Each dataset represents annual Medicare provider claims that are made available about two years after the end of the last release year. As of this study, the 2015 dataset is the latest available year being released in 2017. Medicare Part B includes information pertaining to medical claims for provider services which includes charges, payments, procedures performed, and demographic information. Each claim, or instance in the dataset, is grouped by the provider, procedure, and place of service. The National Provider Identifier (NPI) [18] is a unique code given to each provider. Each procedure performed is designated by a healthcare procedure code using the Healthcare Common Procedure Coding System (HCPCS) [17]. HCPCS codes are required by CMS to ensure that insurance claims and payments are processed in an orderly and consistent manner. The place of service simply refers to offices or facilities, such as hospitals. Note that the Medicare dataset contains values that are registered after claims payments were made, thus we assume the dataset is valid [19]. We combine the four years of available data removing any non-overlapping features, to include standardized payment and standard deviation features which are only available in 2014/2015 and 2012/2013, respectively.

The combined dataset was filtered for non-prescription data only. These include codes that are not for specific services listed on the Medicare Part B Drug Average Sales Price file [2]; thus, the procedure codes and counts are the actual provider services and not prescriptions. Additionally, we include only Medicare program participants and providers with valid NPI and HCPCS codes. As mentioned, the Medicare Part B dataset does not include fraud labels. In order to map fraud labels to the Part B data, we use the LEIE database [24] which includes physicians and other healthcare entities that are excluded from participation in federally funded healthcare programs, such as Medicare, for a certain period of time. The authority to exclude providers is via Section 1128 and 1156 of the Social Security Act [35]. In this paper, we focus on the mandatory exclusions which are presented in Table I. Even though providers are on the LEIE, 38% with fraud convictions continue to practice medicine and 21% were not suspended from medical practice despite their convictions [36].

TABLE I: LEIE mandatory exclusions

Rule Number	Description
1128(a)(1)	Conviction of program-related crimes.
1128(a)(2)	Conviction for patient abuse or neglect.
1128(a)(3)	Felony conviction due to healthcare fraud.
1128(b)(4)	License revocation or suspension.
1128(c)(3)(g)(i)	Conviction of 2 mandatory offenses.
1128(c)(3)(g)(ii)	Conviction on 3+ mandatory offenses.

In order to properly map fraud labels from the LEIE to the Part B data, we need to account for differences in how the datasets are grouped. The Part B dataset is grouped by provider and procedure, whereas the LEIE is grouped by provider only. We decided to perform a provider-level transform of the Part B dataset by aggregating over procedures and places of service per year. From the selected Part B features in Table II, we generate new numerical features representing the mean, median, sum, minimum, maximum, and standard deviation. We do this in order to reduce the loss of information caused by the aggregation. The identifiers and categorical variables remain unmodified. We select these original features from numerical values, such as payments, as well as categorical variables like gender and HCPCS code. The remaining excluded features are demographic in nature, such as address, or redundant features, such as the HCPCS code and description features. Furthermore, due to the aggregation of the data by provider, there are instances where there are only a single provider claim for a given year. This generates NA, or missing, values for the sample standard deviations. Because we know these single claims are valid (since there is an actual provider who performed a single service in only this year), we impute a standard deviation of zero to replace any NA values indicating no claims variation in that particular year. Because some outlier detection methods require all numerical values, the final modification uses one-hot encoding on the provider type (specialty) and gender categorical features. One-hot encoding uses the values, in each categorical feature, to generate dummy features with binary values which indicate the presence of this variable, assigning a value of one if present otherwise zero, versus all other dummy features. Note that any missing gender values are represented by zeros in both male and female features.

After the Part B dataset is at the provider-level, we join the Part B and LEIE datasets by NPI and map the excluded providers as fraud labels indicating either fraud or non-fraud instances. For any instance where the Medicare year is prior to the end of the exclusion period, for an excluded provider, we flag that instance as fraud, otherwise non-fraud. For instance, if a provider's exclusion start year is 2008, with a 5-year period, then the range of the exclusion period is from 2008 to 2013, which overlaps with the available Medicare years, thus 2012 and 2013 are labeled as fraud for that particular provider. Note these fraud instances include possible fraudulent activities prior to and during the exclusion period, thus capturing a majority of the possible suspicious behaviors contributing to a provider being placed on the LEIE. The final dataset has 1,417 excluded providers out of 3,693,980 instances, which is only 0.04% indicating a severely imbalanced dataset [23], [39]. Due to

TABLE II: Description of Medicare Part B features

Feature	Description	Type
npi	Unique provider identification number	Categorical
provider_type	Medical provider's specialty (or practice)	Categorical
nppes_provider_gender	Provider's gender	Categorical
line_srvc_cnt	Number of procedures/services the provider performed	Numerical
bene_unique_cnt	Number of distinct Medicare beneficiaries receiving the service	Numerical
bene_day_srvc_cnt	Number of distinct Medicare beneficiary / per day services performed	Numerical
average_submitted_chrg_amt	Average of the charges that the provider submitted for the service	Numerical
average_medicare_payment_amt	Average payment made to a provider per claim for the service performed	Numerical
exclusion	Fraud labels from the LEIE database	Categorical

a lack of big data outlier detection implementations, such as with Spark ML [30], and memory constraints, particularly for LOF and URF, we reduced the size of the dataset by 50% but, in doing so, made sure to retain the original fraud and non-fraud class distribution. This latter point is important as it indicates that any results using 50% of the data are still representative of the original distribution of known fraudulent providers. Table III summarizes the original, the aggregated (NPI-level), and reduced (final) datasets.

TABLE III: Medicare Part B dataset summary

Medicare Part B dataset	Instances	Features
Original	37,147,213	30
NPI-level	3,693,980	35
NPI-level (one-hot-encoded)	3,693,980	126
NPI-level 50% reduced (one-hot encoded)	1,846,990	126

B. Outlier Detection Methods

In this subsection, we discuss each outlier detection method. We use and compare five different methods: IF, LOF, URF, AE, and KNN. Many outlier detection approaches generate normal profiles in order to flag values that significantly deviate from the acceptable patterns. LOF [14] is a well-known unsupervised outlier detection method that calculates the local density deviation of a given data point with respect to its neighbors. It is considered local because the outlier score is determined by how isolated an observation is with respect to the surrounding neighborhood. Locality is given by k-Nearest Neighbors, whose distance is used to estimate the local density. By comparing the local density of an observation to the local densities of its neighbors, it is possible to identify samples that have a considerably lower density than their neighbors which are considered outliers. The KNN [47] algorithm is a relatively simple and robust method that looks at the k-Nearest Neighbors around some particular value to determine which neighbors are most similar, based on their distance to points in a training dataset. In our case, we calculate the distances between all instances (cases) prior to any prediction, thus employ KNN in an unsupervised manner. Different measures can be used to determine the distance, such as Euclidean, Mahalanobis, Cosine, or Jaccard. For this study, we use Euclidean distance. Observations having large distances from their neighbors, where the neighbors indicate patterns of normal behavior, are seen as outliers.

The last method that leverages normal patterns and marks observations outside of this normal profile as outliers is the autoencoder. An autoencoder [34] is a neural network, applied as an unsupervised learner, using backpropagation to replicate the input as the output. With the sparsity constraint enforced, an autoencoder automatically learns useful features of the unlabeled training data [21]. An autoencoder is composed of an encoder and decoder employed to learn the patterns from the data to generate representative features of that data. These features are then used to reconstruct the original input data patterns, with the reconstruction error indicating the divergence in the model's prediction relative to the original input. In this study, we incorporate "bottleneck" training creating a middle-hidden layer that is very small [16]. We use three hidden layers where the middle hidden layer is the "bottleneck" with two nodes to reduce the dimensionality of the standardized input data to encourage the network to generalize and try to discern non-fraud and fraud patterns in the input data. We use two activation functions in our study, Rectifier Linear Unit (RELU) and Hyperbolic Tangent (Tanh), both employing a dropout rate of 0.5 to create a more generalizable model less likely to overfit the data [41].

The remaining two outlier detection methods do not assume or generate normal profiles from neighborhoods or the input data. IF [27] uses a random forest of decision trees to isolate outliers. IF isolates observations by randomly selecting a feature, and then randomly selecting a split value between the maximum and the minimum values of the selected feature. Since recursive partitioning can be represented by a tree structure, the number of splits required to isolate a sample is equivalent to the path length from the root node to the terminal node. This path length averaged over a forest of many random trees is a measure of abnormality. Random partitioning produces noticeable shorter paths for outliers. When a forest of random trees collectively produces shorter path lengths for particular observations, these are highly likely to be outliers. URF [4], similar to IF, uses decision trees but incorporates a different methodology than IF in detecting outlying values. This methodology assumes that any dataset should be distinguishable from a randomly generated version of itself. The approach taken is to represent the original dataset as class one and create a synthetic dataset of the same size as class two. For this study, we used the tutorial and case studies available

in [4] where a synthetic dataset is generated randomly based on the original dataset. The two datasets together form a two-class classification problem that can be modeled using classical supervised RF. A proximity matrix is used to analyze the number of instances in which two observations are distributed into the same child node of an RF tree. These child node distributions are averaged across *t* trees with the output being an *n x n* matrix of proximity scores, where n is the number of samples. These proximity scores are then used to assess outliers.

In our study, we use the R programming language [37], with specific packages for IF [26], URF [25], KNN [13], and LOF [43]. The autoencoder, however, is part of the deep learning function in the h2o R package [42]. As previously stated, we run several different configurations for each of the described methods. The configurations were chosen based on preliminary experiments and recommendations from the cited literature and R packages. LOF is configured with the following *k* neighbors: 10, 20, 40, 80, and 100, and KNN has k values of 1 and 5, based on prior research [8]. Both IF and URF are configured with 100 trees. The autoencoder network configurations include the following combinations of three hidden layers and *n* nodes per layer with 125 inputs (the number of features minus the exclusion variable): 50-2-50, 100-2-100, and 200-2-200. Tanh or RELU activation functions with a dropout rate of 0.5 are used for each network combination. Retaining the default settings in h2o, L1 and L2 regularization are not used. Note that any other parameters not listed use the default configurations. In this paper, each method incorporates the parameter modifications in the method name, so LOF with 40 neighbors is denoted as 'LOF40', IF with 100 trees is 'IF100', AE with 50 nodes and RELU is 'AE50_RectifierWithDropout'. The remaining methods follow the same naming convention.

C. Performance Metrics

To assess outlier detection performance, we use the Receiver Operating Characteristic (ROC) curve [12], [20]. This is a commonly used representation of the performance of a binary classifier. The ROC curve plots the true positive rate (TPR), or sensitivity, versus the false positive rate (FPR), or 1minus specificity (where specificity is the true negative rate or TNR), across all possible decision thresholds. Performance can be visually assessed from the plot of the ROC curves. A curve with a steeper slope that is closer to the upper left corner of the plot indicates better classification performance. To get a more concise measure of performance from the ROC curves, we compute the Area Under the ROC Curve (AUC), where a value of 1 indicates a perfect model, i.e. 100% sensitivity (no false negatives) and 100% specificity (no false positives), and a value of 0.5 which is equivalent to taking a random guess. In this study, to provide a more complete picture of Part B outlier detection performance, we use ROC curves, AUC, and the sensitivity and specificity values selected at the best decision thresholds. These optimal threshold estimates are determined using Youden's Index [1], which maximizes the distance to the identity line via the following equation: $\max(\text{sensitivities} + \text{specificities})$. Note that these thresholds are the result of each method's returned outlier values, e.g. distance or reconstruction error, thus not necessarily directly comparable without normalization or transformation [7].

IV. Results and Discussion

In this section, we present the Part B fraud detection results of our study with discussions on the performance of each outlier detection method. Table IV shows the performance results for all methods sorted in a descending order by AUC. Overall, based on these results, LOF, for any configuration of neighbors, outperforms all other methods including IF, which was previously shown to outperform LOF [27]. In fact, IF has one of the lower AUC scores, near 0.50, which is almost a random guess. The autoencoders also performed poorly being similar in AUC to IF. On the other hand, URF performs well relative to the other methods. KNN5 has the worst overall performance akin to randomly guessing the fraud or non-fraud labels, which could be due to the high class imbalance.

In Figure 1, we focus the results on the top performers by AUC for each outlier detection method, which depicts the optimal decision threshold and AUC value for each method. Note since all configurations returned the same AUC scores for the autoencoder with the hyperbolic tangent activation function, we chose the configuration with 50 nodes because this setup requires less computational resources than those with more nodes. Additionally, for brevity, we shortened the autoencoder names where AE100_RectifierWithDropout and AE50_TanhWithDropout are replaced with AE100R and AE50T, respectively. The ROC curves depict the optimal decision thresholds, as seen in Table IV, with colored points indicating where each method exhibits the best fraud detection performance. We examine each method's sensitivity and specificity, at the optimal decision thresholds, to provide additional details on detection capabilities.

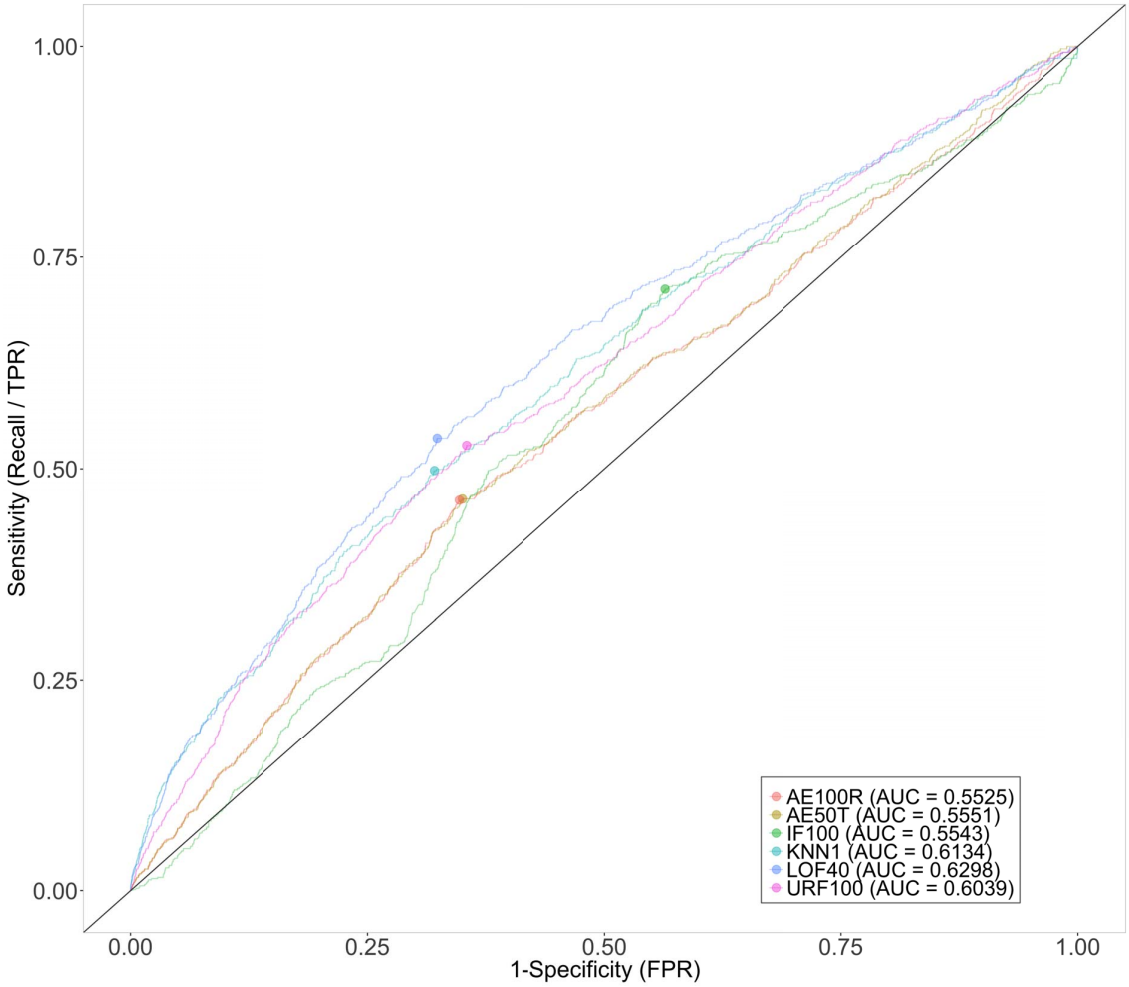
For the best overall performing method, LOF40, over 50% of fraud instances were correctly identified (which is the second best overall) along with about 68% of non-fraud instances. So in this case, a little less than half of the fraudulent providers are missed and slightly over 30% of the non-fraudulent providers are wrongly considered to be fraudulent. Even so, given the highest AUC relative to the other methods, this is the best combination of correctly detected fraudulent and non-fraudulent providers. One possible explanation for the better performance of LOF is due to the use of local density, thus better able to differentiate amongst similar (closer) provider claims to be marked as either non-fraud or fraud. With that said, one known issue of LOF, which may have adversely affected the performance, is that it can sometimes be ineffective when regions of different densities are not clearly separable [5]. KNN is similar to LOF in general methodology using information based on the neighbors and performs similarly, with being KNN1 being the second best overall performing method. The detection of actual fraudulent providers is lower than LOF, but KNN1 is slightly better in correctly detecting non-fraudulent providers. URF has a similar fraud detection rate as LOF, 54% versus 53%, but a higher rate of incorrectly identifying non-fraudulent providers. The use of the original and generated datasets in URF seems to indicate some discernible demarcation between fraud and non-fraud instances based on the proximity (more instances in the same terminal node across the forest) determined by the Random Forest model.

Unlike the aforementioned methods, both IF and AE methods have AUC values around 0.55 which is close to a random

TABLE IV: Fraud detection performance results

Detection Method	AUC	Sensitivity (TPR)	Specificity (TNR)	Optimal Threshold
LOF40	0.6298	0.5362	0.6768	1.0861
LOF80	0.6287	0.5943	0.6107	1.0690
LOF20	0.6284	0.6596	0.5443	1.0480
LOF100	0.6254	0.5362	0.6573	1.0856
LOF10	0.6201	0.4156	0.7750	1.1229
KNN1	0.6134	0.4979	0.6796	248.1108
URF100	0.6039	0.5277	0.6458	0.4483
AE200_TanhWithDropout	0.5551	0.4638	0.6503	100964.1483
AE50_TanhWithDropout	0.5551	0.4638	0.6503	100963.6701
AE100_TanhWithDropout	0.5551	0.4638	0.6503	100963.9570
IF100	0.5543	0.7121	0.4361	0.3444
AE100_RectifierWithDropout	0.5525	0.4624	0.6535	82028.2278
AE50_RectifierWithDropout	0.5418	0.4397	0.6665	46864.5218
AE200_RectifierWithDropout	0.5386	0.4638	0.6304	33013.1876
KNN5	0.5186	0.5966	0.4381	222.9148

Fig. 1: ROC curves for the top outlier detection methods with optimal thresholds and AUCs



guess (e.g. choosing to decide fraud and non-fraud based on a flipping a coin). The AE method has the lowest rate of fraud detection, but able to identify non-fraud instances similarly to URF. A possible reason for the poor performance of AE, especially in detecting fraudulent providers, is that AE reduces the initial original feature set to two abstract features which are supposed to represent the fraud and non-fraud instances, but this abstraction is poor in representing and recreating the original features and correctly identifying fraudulent providers. In this case, the AE's ability to replicate its input, via the bottleneck approach, does not perform well in detecting fraudulent providers. IF is the worst performing method based on AUC, but interestingly it has the highest sensitivity of 71%, thus correctly identifying the majority of fraudulent providers. A significant issue is the low specificity which implies that IF identities too many non-fraudulent providers as fraudulent. This can drastically affect the quality and usability of a fraud detection technique by over-inflating the possible number of fraud instances. IF uses the property that outliers are more susceptible to isolation, thus outliers can be detected as observations that have short expected path lengths (i.e. fewer number of splits) across the forest. In our case, it appears that there are too many splits occurring to isolate the actual fraud instances, suggesting that the observations are not in such sparse regions.

Even though LOF performs well relative to the methods used in this study, an AUC of 0.630 is still a fairly low score and may not suitable for some uses in real-world Medicare Part B fraud detection. One possible explanation for these lower AUC scores is that lack of known fraudulent providers to use as fraud labels for validation, creating a highly imbalanced dataset [44]. Increasing the number of labels could increase both sensitivity and specificity in one or more of the outlier detection methods, since more actual fraud (and consequently non-fraud) instances are known and available. Another reason could be the lack of discriminating patterns in the Medicare Part B data. An unanticipated result to come from our study is the high sensitivity of IF at the optimal threshold. This shows promise in detecting fraudulent providers but is hampered by low specificity, thus too many false positives.

V. Conclusion

Healthcare is a tempting target for unscrupulous individuals or groups due to the large amount of money involved and the overall complexity of the system. Fraud in the Medicare program, due in part to the increasing elderly population, is a continuing problem for which agencies, such as CMS, and other parties are trying to minimize. Even though fraud has been a concern in Medicare for many years, using publicly available Medicare data to detect and mitigate possible fraudulent activities is still relatively new. Efforts to employ effective machine learning solutions to combat Medicare fraud can reduce both fraud-related events and the resources required to investigate possible fraud cases.

In our research, we present an empirical study that evaluates five unsupervised machine learning methods in detecting Medicare Part B fraud. We discuss any necessary processing on the Part B data and incorporate the LEIE database to map fraud labels for validation. AUC is the primary metric used to assess fraud detection performance with further discussions around the sensitivity and specificity of each method at their optimal decision threshold. We evaluated IF and URF, which have not previously been tested on Medicare Part B data, along with LOF, URF, AE, and KNN. The LOF40 method exhibited the best overall fraud detection performance with a 0.630 AUC. KNN5 was the worst method with an AUC of 0.519 which is ostensibly performing random guesses to assign fraud or non-fraud to providers. URF100 performed well as did KNN1, relative to the methods used in this study. IF performed poorly but had the highest sensitivity, at the optimal decision threshold, correctly identifying 71% of fraudulent providers but had a high false positive rate. The AUC scores across all methods are relatively low, which could be attributed to the lack of known fraudulent providers. Thus, incorporating additional fraud labels is left as future work. Additional future work includes using the full dataset, versus only 50%, with more outlier detection methods.

Acknowledgment

We would like to thank the reviewers in the Data Mining and Machine Learning Laboratory at Florida Atlantic University. Additionally, we acknowledge partial support by the NSF (CNS-1427536). Opinions, findings, conclusions, or recommendations in this paper are the authors' and do not reflect the views of the NSF.

References

[1] (2017) Coordinates of a ROC curve. [Online]. Available: https://www.rdocumentation.org/packages/pROC/versions/1.10.0/topics/coords/

[2] "Centers for Medicare and Medicaid Services: Research, Statistics, Data, and Systems," 2018. [Online]. Available: https://www.cms.gov/research-statistics-data-and-systems/research-statistics-data-and-systems.html

[3] "US Medicare Program," 2018. [Online]. Available: https://www.medicare.gov

[4] N. L. Afanador, A. Smolinska, T. N. Tran, and L. Blanchet, "Unsupervised random forest: a tutorial with case studies," *Journal of Chemometrics*, vol. 30, no. 5, pp. 232–241, 2016.

[5] C. C. Aggarwal, "Outlier analysis," in *Data mining*. Springer, 2015, pp. 237–263.

[6] R. A. Bauder and T. M. Khoshgoftaar, "A probabilistic programming approach for outlier detection in healthcare claims," in *Machine Learning and Applications (ICMLA), 2016 15th IEEE International Conference on*. IEEE, 2016, pp. 347–354.

[7] R. A. Bauder and T. M. Khoshgoftaar, "Estimating outlier score probabilities," in *Information Reuse and Integration (IRI), 2017 IEEE International Conference on*. IEEE, 2017, pp. 559–568.

[8] R. A. Bauder and T. M. Khoshgoftaar, "Medicare fraud detection using machine learning methods," in *Machine Learning and Applications (ICMLA), 2017 16th IEEE International Conference on*. IEEE, 2017, pp. 858–865.

[9] R. A. Bauder and T. M. Khoshgoftaar, "Multivariate outlier detection in medicare claims payments applying probabilistic programming methods," *Health Services and Outcomes Research Methodology*, vol. 17, no. 3-4, pp. 256–289, 2017.

[10] R. A. Bauder, T. M. Khoshgoftaar, A. Richter, and M. Herland, "Predicting medical provider specialties to detect anomalous insurance claims," in *Tools with Artificial Intelligence (ICTAI), 2016 IEEE 28th International Conference on*. IEEE, 2016, pp. 784–790.

[11] R. A. Bauder, T. M. Khoshgoftaar, and N. Seliya, "A survey on the state of healthcare upcoding fraud analysis and detection," *Health Services and Outcomes Research Methodology*, vol. 17, no. 1, pp. 31–55, 2017.

[12] M. Bekkar, H. K. Djemaa, and T. A. Alitouche, "Evaluation measures for models assessment over imbalanced datasets," *J Inf Eng Appl*, vol. 3, no. 10, 2013.

[13] A. Beygelzimer, S. Kakadet, J. Langford, S. Arya, D. Mount, and S. Li, *FNN: Fast Nearest Neighbor Search Algorithms and Applications*, 2013, r package version 1.1. [Online]. Available: https://CRAN.R-project.org/package=FNN

[14] M. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander, "Lof: identifying density-based local outliers," in *ACM sigmod record*, vol. 29, no. 2. ACM, 2000, pp. 93–104.

[15] T. Burr, C. Hale, and M. Kantor, "Fraud detection in medicare claims: A multivariate outlier detection approach," Los Alamos National Lab., NM (United States), Tech. Rep., 1997.

[16] J. Chen, S. Sathe, C. Aggarwal, and D. Turaga, "Outlier detection with autoencoder ensembles," in *Proceedings of the 2017 SIAM International Conference on Data Mining*. SIAM, 2017, pp. 90–98.

[17] CMS. HCPCS - General Information. [Online]. Available: https://www.cms.gov/Medicare/Coding/MedHCPCSGenInfo/index.html

[18] CMS. National provider identifier standard (npi). [Online]. Available: https://www.cms.gov/Regulations-and-Guidance/Administrative-Simplification/NationalProvIdentStand/

[19] CMS Office of Enterprise Data and Analytics. (2017) Medicare Fee-For-Service Provider Utilization & Payment Data Physician and Other Supplier. [Online]. Available: https://www.cms.gov/Research-Statistics-Data-and-Systems/Statistics-Trends-and-Reports/Medicare-Provider-Charge-Data/Downloads/Medicare-Physician-and-Other-Supplier-PUF-Methodology.pdf

[20] T. Fawcett, "An introduction to roc analysis," *Pattern recognition letters*, vol. 27, no. 8, pp. 861–874, 2006.

[21] J. Heaton, "Ian goodfellow, yoshua bengio, and aaron courville: Deep learning," 2017.

[22] H. Joudaki, A. Rashidian, B. Minaei-Bidgoli, M. Mahmoodi, B. Geraili, M. Nasiri, and M. Arab, "Using data mining to detect health care fraud and abuse: a review of literature," *Global journal of health science*, vol. 7, no. 1, p. 194, 2015.

[23] T. M. Khoshgoftaar, C. Seiffert, J. Van Hulse, A. Napolitano, and A. Folleco, "Learning with limited minority class data," in *Machine Learning and Applications, 2007. ICMLA 2007. Sixth International Conference on*. IEEE, 2007, pp. 348–353.

[24] LEIE. (2017) Office of inspector general leie downloadable databases. [Online]. Available: https://oig.hhs.gov/exclusions/index.asp

[25] A. Liaw and M. Wiener, "Classification and regression by randomforest," *R News*, vol. 2, no. 3, pp. 18–22, 2002. [Online]. Available: https://CRAN.R-project.org/doc/Rnews/

[26] F. T. Liu, *Isolation Forest*, 2009, r package version 0.0-26. [Online]. Available: https://r-forge.r-project.org/projects/iforest/

[27] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation forest," in *Data Mining, 2008. ICDM'08. Eighth IEEE International Conference on*. IEEE, 2008, pp. 413–422.

[28] J. Liu, E. Bier, A. Wilson, J. A. Guerra-Gomez, T. Honda, K. Sricharan, L. Gilpin, and D. Davies, "Graph analysis for detecting fraud, waste, and abuse in healthcare data," *AI Magazine*, vol. 37, no. 2, p. 33, Apr 2016.

[29] M. Mather, L. A. Jacobsen, and K. M. Pollard, *Population Bulletin*, vol. 70, no. 2, pp. 1–53, 2015.

[30] X. Meng, J. Bradley, B. Yavuz, E. Sparks, S. Venkataraman, D. Liu, J. Freeman, D. Tsai, M. Amde, S. Owen *et al.*, "Mllib: Machine learning in apache spark," *The Journal of Machine Learning Research*, vol. 17, no. 1, pp. 1235–1241, 2016.

[31] L. Morris, "Combating Fraud In Health Care: An Essential Component Of Any Cost Containment Strategy," 2009. [Online]. Available: https://www.healthaffairs.org/doi/abs/10.1377/hlthaff.28.5.1351

[32] National Health Expenditure Projections 2017-2026. (2018) Centers for Medicare & Medicaid Services. [Online]. Available: https://www.cms.gov/Research-Statistics-Data-and-Systems/Statistics-Trends-and-Reports/NationalHealthExpendData/Downloads/ForecastSummary.pdf

[33] National Health Expenditures 2016 Highlights. (2018) Centers for Medicare & Medicaid Services. [Online]. Available: https://www.cms.gov/Research-Statistics-Data-and-Systems/Statistics-Trends-and-Reports/NationalHealthExpendData/Downloads/highlights.pdf

[34] A. Ng, ""sparse autoencoder"," *CS294A Lecture notes*, vol. 72, no. 2011, pp. 1–19, 2011.

[35] OIG. (2018) Office of inspector general leie exclusion authorities. [Online]. Available: https://oig.hhs.gov/exclusions/authorities.asp

[36] V. Pande and W. Maas, "Physician medicare fraud: characteristics and consequences," *International Journal of Pharmaceutical and Healthcare Marketing*, vol. 7, no. 1, pp. 8–33, 2013.

[37] R Core Team, *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria, 2018. [Online]. Available: https://www.R-project.org/

[38] J. O. Savino and B. E. Turvey, "Chapter 5 - medicaid/medicare fraud," in *False Allegations*, B. E. Turvey, J. O. Savino, and A. C. Mares, Eds. San Diego: Academic Press, 2018, pp. 89 – 108. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9780128012505000057

[39] C. Seiffert, T. M. Khoshgoftaar, J. Van Hulse, and A. Napolitano, "Mining data with rare events: a case study," in *Tools with Artificial Intelligence, 2007. ICTAI 2007. 19th IEEE International Conference on*, vol. 2. IEEE, 2007, pp. 132–139.

[40] Y. Shan, D. W. Murray, and A. Sutinen, "Discovering inappropriate billings with local density based outlier detection method," in *Proceedings of the Eighth Australasian Data Mining Conference-Volume 101*. Australian Computer Society, Inc., 2009, pp. 93–98.

[41] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[42] T. H. team, *h2o: R Interface for H2O*, 2017, r package version 3.16.0.2. [Online]. Available: https://CRAN.R-project.org/package=h2o

[43] L. Torgo, *Data Mining with R, learning with case studies*. Chapman and Hall/CRC, 2010. [Online]. Available: http://www.dcc.fc.up.pt/~ltorgo/DataMiningWithR

[44] J. Van Hulse, T. M. Khoshgoftaar, and A. Napolitano, "Experimental perspectives on learning from imbalanced data," in *Proceedings of the 24th international conference on Machine learning*. ACM, 2007, pp. 935–942.

[45] S. S. Waghade and A. M. Karandikar, "A comprehensive study of healthcare fraud detection based on machine learning," *International Journal of Applied Engineering Research*, vol. 13, no. 6, pp. 4175–4178, 2018.

[46] S. M. Weiss, C. A. Kulikowski, R. S. Galen, P. A. Olsen, and R. Natarajan, "Managing healthcare costs by peer-group modeling," *Applied Intelligence*, vol. 43, no. 4, pp. 752–759, 2015.

[47] I. H. Witten, E. Frank, M. A. Hall, and C. J. Pal, *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, 2016.