2019 IEEE 20th International Conference on Information Reuse and Integration for Data Science (IRI)

Evaluating Model Predictive Performance: A Medicare Fraud Detection Case Study

Richard A. Bauder, Matthew Herland, and Taghi M. Khoshgoftaar
College of Engineering & Computer Science
Florida Atlantic University
rbauder2014@fau.edu, mherlan1@fau.edu, khoshgof@fau.edu

Abstract

Evaluating a machine learning model's predictive performance is vital for establishing the practical usability in real-world applications. The use of separate training and test datasets, and cross-validation are common when evaluating machine learning models. The former uses two distinct datasets, whereas cross-validation splits a single dataset into smaller training and test subsets. In real-world production applications, it is critical to establish a model's usefulness by validating it on completely new input data, and not just using the cross-validation results on a single historical dataset. In this paper, we present results for both evaluation methods, to include performance comparisons. In order to provide meaningful comparative analyses between methods, we perform real-world fraud detection experiments using 2013 to 2016 Medicare durable medical equipment claims data. This Medicare dataset is split into training (2013 to 2015 individual years) and test (2016 only). Using this Medicare case study, we assess the fraud detection performance, across three learners, for both model evaluation methods. We find that using the separate training and test sets generally outperforms cross-validation, indicating a better real-world model performance evaluation. Even so, cross-validation has comparable, but conservative, fraud detection results.

Keywords: model evaluation, cross-validation, Medicare fraud detection, machine learning

Introduction

The need for accurate methods to evaluate machine learning model performance is vital in model building and deployment into real-world operational environments. There are many different evaluation methods (Bengio and Grandvalet 2004), where two popular methods include using separate training and test sets (Train_Test) and cross-validation (CV). The latter evaluation method (Prashant Gupta 2018) splits a single dataset into a number of smaller subsets of training and test sets, which allows for model evaluation without a separate, unseen test dataset. The Train_Test method is similar in concept to hold-out CV. However, the former uses completely separate training and test sets, whereas CV only incorporates the training dataset. More specifically, this distinction is important, because CV derives the test sets from the original training data and not from a separate, unseen dataset. CV can be useful when a researcher only has access to relatively small historical data, where there is not enough data to reasonably create a separate distinct test set. However, for 'Big Data' datasets, especially with uneven class representation, CV may present inaccurate model performance results.

This begs the question as to how accurate CV predictions are compared to the Train_Test evaluation results with Big Data. In general, CV tends to be less reliable in assessing model performance, with Train_Test providing more accurate results (Justin Domke 2018; Martin Schmitz 2018). Part of the reason for this lower reliability is due to CV being susceptible to overfitting the data (low bias, high variance) (Hawkins 2004), because each instance is used both for model building and evaluation. Furthermore, bias can be introduced into the CV process as models are trained on smaller datasets, which limits a model's discriminatory power (Prashant Gupta 2018). Rao et al. (Rao, Fung, and Rosales 2008) provide an argument that, "Any modeling decisions based upon experiments on the training set, even cross validation estimates, are suspect, until independently verified [by a separate and unique test set]." Therefore, given a reasonable amount of data, having separate training and test sets for model evaluation may increase accuracy and produce different results versus CV.

In order to provide model performance results for comparative analyses between Train_Test and CV methods, we present a United States (U.S.) Medicare fraud detection case study. This is an important real-world application for evaluating the fraud detection performance of machine learning models, over both methods. The importance of detecting fraud is in recovering and reducing losses due to improper or illegal activities leading to large financial losses. In 2017, U.S. healthcare spending was about $3.5 trillion and is projected to increase 4% per year through 2026 (CMS 2018c). The Federal Bureau of Investigation (FBI) estimates that fraud accounts for 3-10% of healthcare costs (Morris 2009). This indicates that, for U.S. healthcare programs, $105 to $350 billion could be recovered through effective fraud detection. Medicare is one such federal program that received approximately $600 billion

978-1-7281-1337-1/19/$31.00 ©2019 IEEE
DOI 10.1109/IRI.2019.00016

in federal funding in 2017, which was about 17% of overall healthcare spending (Kaiser Family Foundation 2017). The Medicare program, administered by the Centers for Medicaid and Medicare Services (CMS), covers part of the cost associated with healthcare, primarily serving beneficiaries over the age of 65 and some younger, disabled people and dialysis patients (U.S. Centers for Medicare & Medicaid Services 2017). For this case study, we use the *Medicare Provider Utilization and Payment Data: Referring Durable Medical Equipment, Prosthetics, Orthotics, and Supplies* (DMEPOS) big dataset. Durable medical equipment is any equipment, such as wheelchairs, canes, walkers, and kidney machines, that provides therapeutic benefits to a patient with certain medical conditions. This is an important aspect of Medicare providing its beneficiaries with a better quality of life. At present, CMS does not provide fraud labels for the DMEPOS dataset. To obtain fraud labels for the machine learning models, we use the Office of Inspector General's (OIG) List of Excluded Individuals and Entities (LEIE) (OIG 2018). More information on Medicare fraud can be found in (CMS 2018d; 2018b).

In this paper, we use three different models to assess the fraud detection with the fraud-labeled DMEPOS dataset. These results are from both Train_Test and CV methods using the Area Under the Receiver Operating Characteristic (ROC) Curve (AUC) performance metric, with differences assessed via significance testing. Our results indicate that the Train_Test evaluation method outperforms the CV method, though CV does produce comparable results. Even so, the CV fraud detection results are slightly conservative when compared to the Train_Test method. To the best of our knowledge, this is the first paper to compare model evaluation methods in a real-world Medicare fraud application domain, using the latest 2016 DMEPOS dataset (made available in June 2018) as test data. Our study's contributions are as follows:

- Summarize our unique approach in preparing the DMEPOS data and mapping the LEIE fraud labels
- Present comparative results for Train_Test and CV model evaluation methods, and demonstrate the benefit of employing the Train_Test method to evaluate Medicare fraud detection performance

The remainder of the paper is organized as follows. The Related Works section discusses works related to the current research. In the Data section, we discuss the Medicare and LEIE datasets. The learners, performance metric, and overall design of our experiment are discussed in the Experimental Design section. The results of our research are detailed in the Results and Discussion section. Finally, the Conclusion section summarizes our conclusions and possible avenues for future work.

Related Works

The use of CV, across diverse application domains, is a popular method of evaluating the predictive performance of machine learning models (Bengio and Grandvalet 2004). In our study, we employ CV to assess fraud detection performance and compare it with a Train_Test evaluation method. As such, we focus the selection of related works on the uses and shortcomings of CV. In (Justin Domke 2018), Domke discusses the inverse relationship between bias and variance experienced when employing CV. Bias occurs when the built model lacks discrimination between classes and does not fully recognize the full complexities of the data. Variance occurs when the model built is overly discriminatory, becoming overfit. Optimal CV models will balance this tradeoff in order to minimize error. Varoquaux (Varoquaux 2017) conducts a study using brain image analysis to promote awareness for the shortcomings when applying CV, specifically for Leave One Out Cross-Validation (LOOCV) and 80/20 Train/Test splits with 50 repeats. The author determines that for small sample sizes, CV results in large errors. Bengio et al. (Bengio and Grandvalet 2004) note that k-fold CV is susceptible to large degrees of variability, potentially misleading a researcher's decision during model selection. They demonstrate that determining the level of variance in k-fold CV is challenging and that there exists no variance estimation technique without bias. Their results show, in very simple cases, the bias centered around ignoring the dependencies between test errors will be relatively equal to the quantity of variance.

An experiment, by Rao et al. (Rao, Fung, and Rosales 2008), demonstrating the impact of different algorithms and increasing data dimensionality on LOOCV shows that as sample size decreases and the number of algorithms and data dimensionality increase, the effectiveness of CV to estimate generalization becomes less reliable. They recommend validating CV results with a separate test set. They warn that when a model is tuned based upon performance on a test set, this is no longer viable for simulating a real-world scenario, and the test set should only be used for evaluation. In (Kodovskỳ 2011), Kodovsky performs a study in the field of steganalysis, presenting the risks involved in utilizing CV, specifically in the JPEG domain. The author demonstrates that k-fold CV results are inadequate, with a significant difference between predicted error and real testing error. It is important to note that none of these studies assess or use big datasets or focus on Medicare fraud detection. However, if these issues with CV are found in these other domains, it is possible that some of these concerns could apply to the domain of Medicare fraud. Consequently, this could lead to the selection of potentially sub-optimal fraud detection models.

Medicare Data

The release of Medicare-related datasets to the public enabled studies on fraud detection leveraging advanced analytics and machine learning (Sheshasaayee and Thomas 2018; Waghade and Karandikar 2018; Herland, Bauder, and Khoshgoftaar 2018; Herland, Khoshgoftaar, and Bauder 2018; Herland, Bauder, and Khoshgoftaar 2019; Bauder, Khoshgoftaar, and Hasanin 2018; Bauder and Khoshgoftaar 2018a; 2018b). The majority of these studies have used Medicare Part B data (Feldman and Chawla 2015; Chandola, Sukumar, and Schryver 2013; Khurjekar, Chou, and Kha-

Table 1: Description of selected DMEPOS dataset features

Feature	Description	Type
referring_npi*	Unique provider identification number	Categorical
referring_provider_type	Medical provider's specialty (or practice)	Categorical
referring_provider_gender	Provider's gender	Categorical
number_of_suppliers	Number of suppliers used by provider	Numerical
number_of_supplier_beneficiaries	Number of beneficiaries associated by the supplier	Numerical
number_of_supplier_claims	Number of claims submitted by a supplier due to an order by a referring order	Numerical
number_of_supplier_services	Number of services/products rendered by a supplier	Numerical
avg_supplier_submitted_charge	Average payment submitted by a supplier	Numerical
avg_supplier_medicare_pmt_amt	Average payment awarded to suppliers	Numerical
exclusion	Fraud labels from the LEIE database	Categorical

**not used in model training*

sawneh 2015) which focuses on physician claims for medical procedures or services performed. The data used in our experiment is from the Centers for Medicare and Medicaid Services (CMS) and encompasses the 2013 to 2016 calendar years (CMS 2018d). The *Medicare Provider Utilization and Payment Data: Referring Durable Medical Equipment, Prosthetics, Orthotics. and Supplies* (DMEPOS) contains information on products and services provided to beneficiaries to include utilization, payment (allowed amount and Medicare payment), and submitted charges organized by National Provider Identifier (NPI) (CMS 2016), Healthcare Common Procedure Coding System (HCPCS) code (CMS 2018a) and supplier rental indicator. This Medicare dataset contains values that are recorded after claims payments were made and with that, we assume that the Medicare dataset was appropriately recorded and cleansed by CMS. There are no known studies focusing primarily on Medicare DMEPOS fraud detection.

For the Train_Test and CV methods, the training dataset consists of all available years of Medicare data prior to 2016. We do this by appending each annual dataset to each other for only matching features. The test dataset, for Train_Test only, encompasses the 2016 data with the same features as the training dataset (2017 and 2018 data are not available). There are a number of features available in the DMEPOS dataset, for which we select those specifically related to provider claims information and readily usable by machine learning models. The remaining features are excluded for several reasons. For instance, features used for identification purposes, such as the NPI, or those used for filtering the data, like Medicare participation, are not included in model building. Repetitive and redundant features such as physician names, addresses, and other demographic information are not included, as they provide no further discriminative value. We also removed several features containing missing or constant values. Table 1 lists the DMEPOS features used in our study. Note that the 'exclusion' feature contains the fraud labels. Details on these Medicare features are found in "Public Use File: A Methodological Overview" (Office of Enterprise Data and Analytics 2018).

In order to obtain labels indicating fraudulent providers, we incorporate excluded providers from the LEIE dataset (LEIE 2018). The LEIE only includes NPI-level, or provider-level, exclusions, with no details on procedures (HCPCS codes) that contribute to potential fraud. The exclusions are categorized by various rule numbers, which indicate severity as well as the length of time of each exclusion. We selected the providers with mandatory exclusions, as seen in Table 2.

Table 2: LEIE exclusion rules

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

We transformed the DMEPOS dataset to the NPI- or provider-level by aggregating over the HCPCS procedure codes and place of service. This was necessary because the current LEIE only lists providers with no indication of the procedure code associated with a particular exclusion. Presently, there is no known publicly available dataset which includes fraud labels by provider and by each procedure performed, but future research will look at ways to mitigate this lack of data through majority voting or methods of NPI-level data aggregation. When both datasets are at the NPI-level, we map fraud labels to the DMEPOS by joining the LEIE information via NPI and year. All matching providers are labeled with fraud, otherwise non-fraud. Additionally, we consider any provider with claims prior to and during their exclusion period as fraud, while accounting for any early waiver and reinstatement dates for which we label as non-fraud. These steps for mapping fraud labels help reduce the potential over counting of fraudulent providers due to overlapping or expired exclusion periods. Thus, we can be reasonably confident, with the stated assumptions, that we capture a fair number of fraud labels for the corresponding excluded providers. Additional details on data processing and fraud labeling can be found in (Bauder and Khoshgoftaar 2018b). Table 3 summarizes the final DMEPOS datasets.

Table 3: Training and Test dataset summary

	Dataset	Features	Non-Fraud	Fraud	% Fraud
Train	DMEPOS	145	862,792	635	0.074%
Test	DMEPOS	119	290,548	75	0.026%

Experimental Design

For our experiments, we built and tested three different machine learning models to classify fraudulent Medicare provider claims. To effectively handle the larger DMEPOS dataset, we employ Spark on top of a Hadoop YARN cluster (Apache 2018a). The Apache Spark 2.3.0 Machine Learning Library has eight classifiers available (Apache 2018b). From this library, we built one non-tree model, Logistic Regression (LR) (Apache 2018d), and two tree-based models, Random Forest (RF) (Apache 2018b) and Gradient-Boosting Trees (GBT) (Apache 2018c). Unless specified otherwise, we used default configurations for each learner. These models are scored using the AUC performance metric (Bekkar, Djemaa, and Alitouche 2013; Richter and Khoshgoftaar 2019). AUC is a popular measure of model performance, providing a general idea of predictive potential of a binary classifier. The ROC curve is used to characterize the trade-off between true positive rate and false positive rate and depicts a learner's performance across all decision thresholds, i.e. a value between 0 and 1 that separate the classes. AUC is a single value that ranges from 0 to 1, where a perfect classifier provides an AUC value of 1.

To evaluate the fraud detection performance for these models, we employ Train_Test and CV methods. For CV, we use stratified k-fold CV where k=5. Evaluation with CV is done on a single dataset by splitting the dataset into k folds. A model is built on k-1 folds and evaluated on the remaining fold. This process is repeated until each fold has been used to evaluate the model, with the final result being the average of the scores across all of the k folds. The stratification ensures all folds have approximately the same ratio of class representation as in the original dataset, which could be important for evaluating highly imbalanced datasets. We repeat the CV process 10 times, to help reduce bias from bad random draws, with the final model evaluation being the average score over these 10 repeats. It is important to note that CV is applied to the training data and denoted as Train_CV.

As mentioned, the Train_Test method uses one dataset for model building, with a separate, unique dataset for evaluation. The instances in the test set are completely new instances, never used in model creation, unlike the CV method. Prior experimentation with the Train_Test method is necessary for implementing real-world applications answering whether, based on past occurrences, a model accurately predicts new occurrences. More specifically, this method will determine whether, based on historical (or prior) DMEPOS information (prior to 2016), providers can be classified as fraud or non-fraud given new information (2016).

Results and Discussion

In Table 4, we present the average AUC scores for the Train_Test and Train_CV methods. The boldfaced values show the highest AUC scores per method. On average, the Train_Test method results are consistently better than those employing Train_CV. These results indicate that all models perform similarly well using the Train_Test method, with more noticeable variability in model fraud detection performance with Train_CV. The less variable results, across models, for the Train_Test method seem to demonstrate a more accurate result in predicting DMEPOS claims fraud.

Table 4: Average AUC results by method and model

Method	Learner	DMEPOS
Train_Test	Gradient Boosted Trees	**0.78281**
	Logistic Regression	0.78088
	Random Forest	0.77105
Train_CV	Gradient Boosted Trees	0.72789
	Logistic Regression	**0.74120**
	Random Forest	0.70525

A comparison of the differences in average AUC scores between both methods shows that Train_Test is, on average, 0.05347 points higher than with the CV method. This could indicate the high variance in the Train_CV process, leading to less consistent results across models. Moreover, it is less likely, given the large size of the DMEPOS dataset, that the Train_Test method would overfit the data. From these results, we can see that leveraging a Train_Test model evaluation method provides better and more consistent results in predicting fraud using the DMEPOS big dataset.

To better determine if these differences in average performance between Train_Test and Train_CV are significant, we use both ANalysis Of VAriance (ANOVA) (Gelman and others 2005) and Tukey's HSD tests (Tukey 1949). ANOVA is a statistical test determining whether the means of several groups (or factors) are equal. Tukey's HSD test determines factor means that are significantly different from each other. This test compares all possible pairs of means using a method similar to a t-test, where statistically significant differences are grouped by assigning different letter combinations (e.g. group 'a' is significantly better than group 'b'). We perform a 2-factor ANOVA with model and evaluation method as factors. The ANOVA results indicate that both model and evaluation method, as well as their interactions, are significant at a 95% confidence interval. The Tukey's HSD results, as seen in Table 5, show that the Train_Test method is significantly better than Train_CV, and confirms that Train_Test has less average variability in model evaluation results. With regards to the learners, LR is significantly better than the other models. This is because LR is close to GBT using the Train_Test method, and the best performer with Train_CV. Overall, we see that both results are close with CV eliciting a more conservative model evaluation.

Conclusion

From our study, we demonstrated the efficacy of both Train_Test and CV methods in evaluating model perfor-

Table 5: Tukey's HSD test results

Factor	Level	AUC	std	Min	Max	Q25	Q50	Q75	groups
Method	Train_Test	0.77825	0.00750	0.76117	0.79329	0.77348	0.78088	0.78088	a
Method	Train_CV	0.72478	0.02383	0.66261	0.77675	0.70651	0.72651	0.74342	b
Learner	LR	0.74782	0.02107	0.70641	0.78088	0.73162	0.74729	0.76060	a
Learner	GBT	0.73705	0.02750	0.69289	0.79329	0.71654	0.73334	0.75110	b
Learner	RF	0.71621	0.03078	0.66261	0.78544	0.69573	0.70979	0.72967	c

mance. The use of a real-world U.S. Medicare fraud detection case study was used to evaluate three different machine learning models, and provide comparative analyses of both model evaluation methods. This particular case study is useful, not only for this comparison, but also because minimizing losses due to Medicare fraud is critical in ensuring program viability and quality of care. The major difference between the Train_Test and CV evaluation methods is that the former uses separate and distinct datasets for model building and validation. CV, however, only incorporates a single dataset for both building and validating a model. From our case study results, we show that the Train_Test method outperforms the Train_CV method, across all models, in predicting DMEPOS claims fraud. From the results of the Train_Test method, we conclude that the use of machine learning models on Medicare claims can accurately predict potential fraud for a new, unseen year (2016) based on historical (prior to 2016) data. Overall, the Train_Test method performed significantly better than the Train_CV method, though the average results were generally comparable. Therefore, if necessary, CV is a reasonable substitute for a practitioner or researcher when the appropriate resources to evaluate models using a Train_Test method are unavailable. Future work will include additional Medicare big datasets and use methods to lessen the impact of class imbalance.

Acknowledgment

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

Apache. 2018a. Apache spark.

Apache. 2018b. Classification and regression.

Apache. 2018c. Ensembles.

Apache. 2018d. Linear methods.

Bauder, R. A., and Khoshgoftaar, T. M. 2018a. The effects of varying class distribution on learner behavior for medicare fraud detection with imbalanced big data. *Health information science and systems* 6(1):9.

Bauder, R. A., and Khoshgoftaar, T. M. 2018b. A survey of medicare data processing and integration for fraud detection. In *Information Reuse and Integration (IRI), 2018 IEEE 19th International Conference on*, 9–14. IEEE.

Bauder, R. A.; Khoshgoftaar, T. M.; and Hasanin, T. 2018. Data sampling approaches with severely imbalanced big data for medicare fraud detection. In *2018 IEEE 30th international conference on tools with artificial intelligence (ICTAI)*, 137–142. IEEE.

Bekkar, M.; Djemaa, H. K.; and Alitouche, T. A. 2013. Evaluation measures for models assessment over imbalanced datasets. *J Inf Eng Appl* 3(10).

Bengio, Y., and Grandvalet, Y. 2004. No unbiased estimator of the variance of k-fold cross-validation. *Journal of machine learning research* 5(Sep):1089–1105.

Chandola, V.; Sukumar, S. R.; and Schryver, J. C. 2013. Knowledge discovery from massive healthcare claims data. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, 1312–1320. ACM.

CMS. 2016. National Provider Identifier Standard (NPI).

CMS. 2018a. HCPCS - General Information.

CMS. 2018b. Medicare fraud & abuse: Prevention, detection, and reporting.

CMS. 2018c. National health expenditure projections 2017-2026.

CMS. 2018d. Research, statistics, data, and systems.

Feldman, K., and Chawla, N. V. 2015. Does medical school training relate to practice? evidence from big data. *Big Data* 3(2):103–113.

Gelman, A., et al. 2005. Analysis of variance why it is more important than ever. *The annals of statistics* 33(1):1–53.

Hawkins, D. M. 2004. The problem of overfitting. *Journal of chemical information and computer sciences* 44(1):1–12.

Herland, M.; Bauder, R. A.; and Khoshgoftaar, T. M. 2018. Approaches for identifying us medicare fraud in provider claims data. *Health care management science* 1–18.

Herland, M.; Bauder, R. A.; and Khoshgoftaar, T. M. 2019. The effects of class rarity on the evaluation of supervised healthcare fraud detection models. *Journal of Big Data* 6(1):21.

Herland, M.; Khoshgoftaar, T. M.; and Bauder, R. A. 2018. Big data fraud detection using multiple medicare data sources. *Journal of Big Data* 5(1):29.

Justin Domke. 2018. Overfitting, model selection, cross validation, bias-variance.

Kaiser Family Foundation. 2017. The facts on medicare spending and financing.

Khurjekar, N.; Chou, C.-A.; and Khasawneh, M. T. 2015. Detection of fraudulent claims using hierarchical cluster analysis. In *IIE Annual Conference. Proceedings*, 2388. Institute of Industrial and Systems Engineers (IISE).

Kodovskỳ, J. 2011. On dangers of cross-validation in steganalysis. Technical report, Citeseer.

LEIE. 2018. Office of inspector general leie downloadable databases.

Martin Schmitz. 2018. When cross validation fails.

Morris, L. 2009. Combating fraud in health care: an essential component of any cost containment strategy. *Health Affairs* 28(5):1351–1356.

Office of Enterprise Data and Analytics. 2018. Medicare fee-for-service provider utilization & payment data referring durable medical equipment, prosthetics, orthotics and supplies public use file: A methodological overview.

OIG. 2018. Office of inspector general exclusion authorities.

Prashant Gupta. 2018. Cross-validation in machine learning.

Rao, R. B.; Fung, G.; and Rosales, R. 2008. On the dangers of cross-validation. an experimental evaluation. In *Proceedings of the 2008 SIAM International Conference on Data Mining*, 588–596. SIAM.

Richter, A., and Khoshgoftaar, T. 2019. Efficient learning from big data for cancer risk modeling: A case study with melanoma. *Computers in biology and medicine* 110:29–39.

Sheshasaayee, A., and Thomas, S. S. 2018. A purview of the impact of supervised learning methodologies on health insurance fraud detection. In *Information Systems Design and Intelligent Applications*. Springer. 978–984.

Tukey, J. W. 1949. Comparing individual means in the analysis of variance. *Biometrics* 99–114.

U.S. Centers for Medicare & Medicaid Services. 2017. The offical U.S. Governement site for Medicare.

Varoquaux, G. 2017. Cross-validation failure: small sample sizes lead to large error bars. *Neuroimage*.

Waghade, S. S., and Karandikar, A. M. 2018. A comprehensive study of healthcare fraud detection based on machine learning. *International Journal of Applied Engineering Research* 13(6):4175–4178.