Stacked generalizations in imbalanced fraud data sets using resampling methods

Kathleen R Kerwin¹ and Nathaniel D Bastian¹,²

Abstract
Predicting fraud is challenging due to inherent issues in the fraud data structure, since the crimes are committed through trickery or deceit with an ever-present moving target of changing modus operandi to circumvent human and system controls. As a national security challenge, criminals continually exploit the electronic financial system to defraud consumers and businesses by finding weaknesses in the system, including in audit controls. This study uses stacked generalization using meta or super learners for improving the performance of algorithms in step one (minimizing the algorithm error rate to reduce its bias in the learning set) and then in step two the results are input into the meta learner with its stacked blended output (with the weakest algorithms learning better). A fundamental key to fraud data is that it is inherently not systematic, and an optimal resampling methodology has yet not been identified. Building a test harness, for all permutations of algorithm sample set pairs, demonstrates that the complex, intrinsic data structures are all thoroughly tested. A comparative analysis on fraud data that applies stacked generalizations provides useful insight to find the optimal mathematical formula for imbalanced fraud data sets necessary to improve upon fraud detection for national security.

Keywords
Stacked generalizations, imbalanced data, intelligent resampling, machine learning, meta learners, ensemble methods, fraud detection

1. Introduction
Predicting fraud is challenging due to inherent issues in the fraud data structure, since the crimes are committed through trickery or deceit with an ever-present moving target of changing modus operandi to circumvent human and system controls. In addition to these complex and unsystematic characteristics, there is extreme imbalance in the data due to the class distributions between fraudulent and normal transactions.

According to the Association of Certified Fraud Examiners (ACFE), businesses worldwide lose 5% of their revenues annually to fraud from internal and external sources.¹ "Fraud consists of some deceitful practice or willful device, resorted to with intent to deprive another of his right, or in some manner to do him an injury."¹ Businesses are exposed to internal fraud by employees, as well as fraud targeted at their business.

The effect on our domestic economy is profound. Fraud is becoming increasingly profitable and being committed by organized crime where the exploitation of discovered weaknesses in the financial system are taken advantage of until controls are put in place.² Criminals then search for other means to exploit businesses, which means an ever-moving target for investigators to prevent and detect.

¹Department of Data Science, School of Professional Studies, Northwestern University, USA
²Army Cyber Institute, United States Military Academy, USA

Corresponding author:
Kathleen R Kerwin, Department of Data Science, School of Professional Studies, Northwestern University, Evanston, IL 60201, USA.
Email: krkerwin@sbcglobal.net
Crime rings study how financial institutions operate and work around controls by fragmenting transactions into discrete units between organizations to obfuscate complex criminal collaborative efforts. The failure of the ability to see the more comprehensive picture of the illegal activity results in institution losses due to these crimes being written off or written down, as well as the perpetrators being able to disguise the crime typologies and identity of the criminals. Legal barriers in sharing information across financial institutions and law enforcement agencies (unless regulations trigger 314 (b) of the US Patriot Act based on suspicious activity or evidence of a crime) further encourage crime rings to exploit the inability of coordinated tracking and investigating. The Clearinghouse paper on “Reforms to the AML/CFT framework for national security and law enforcement” recommended illuminating and enhancing the range of information sharing.

Looking from the outside in, criminals try any and everything to defraud businesses and individuals from their assets. From the inside out, criminals pay insiders for their knowledge, expertise, and assistance in identifying ways to find loopholes and circumvent manual and system controls. The only limitation for criminals in committing fraud is their creativity and ingenuity. Criminals continually exploit the electronic financial system to defraud consumers and businesses by finding weaknesses in the system, including in audit controls. When these weaknesses are found and fixed, criminals simply update their strategy and find other means of committing crimes. Since these fraud events are continually changing when discovered, a business and information processing management system needs to be created that not only identifies similar transactions to known fraud cases but also keeps up with how criminals change their habits by recognizing new fraud cases.

1.1. National security and transactional organized crime

The greatest risk is from organized crime, which in the age of increased information technology and communication stems from the global digital economy and infrastructure. The decentralized nature of the internet erases borders in cyberspace, encouraging transnational organized crime. Criminals find weaknesses in security at all levels, from individual businesses to national security and exploit, them until controls and updated regulations are put in place. Criminals more typically do not reside in the USA because they have a higher chance of being caught and held to answer due to our law enforcement, legal, and regulatory system, as well as being in physical proximity for being apprehended, which does not require an extradition request.

Crime as a Service (CaaS) permits criminal entrepreneurs to sell their services and products to those willing to pay that normally would not have access to these deep skills. Examples are money laundering and cybercrime (cryptoware and network intrusion software for data theft), including links to terrorism. When criminal activity is intermingled with legal activity the endeavor can be preserved as an ongoing business, which compounds the threat. National security requires a cyber security strategy from a transnational organized crime perspective.

The United Nations Office of Drugs and Crime (UNODC) estimates that approximately 2–5% of global Gross Domestic Product, or $800 billion–$2 trillion in current US dollars, is money laundered globally based on predicate crimes that include trafficking in drugs, psychotropics, stolen goods, arms, humans, racketeering, terrorism, terrorism financing, sexual exploitation, counterfeiting of currency or goods, corruption, bribery, fraud, environmental crimes, murder, kidnapping, theft, extortion, forgery, piracy, smuggling, insider trading, market manipulations, and tax crimes, for example. Dysfunctional countries with weak socio-political institutions are vulnerable to corruption and human rights violations and are magnets for organized crime, since they offer safe havens because they do not have the legal and technical infrastructures to fight back. The failure of necessary protections being in place enables organized crime to become illegal self-perpetuating industries without fear of being detected or prevented.

Fighting back requires a long-term plan with a global coordinated outreach and synchronization among countries at supervisory, regulatory, investigatory, and law enforcement levels, such as the international Financial Action Task Force (FATF) and Egmont Financial Intelligence Units (FIUs) group.

The FATF, established in 1989 as an intergovernmental agency located at the Office of Economic Co-operation and Development (OECD) in Paris, which builds the world’s blueprint for Anti-money Laundering (AML) controls, established core principles with the founding principles called the FATF 40 Recommendations, completes technical evaluations and assessment for each country submitting to become an organization member, and monitors and implements recommendations. There are 39 member countries, in addition to FATF-style Regional Bodies (FSRBs) with protocols for assisting and encouraging non-member countries to join. The US Department of Treasury is the local FATF within the USA.

FIUs are voluntary international cooperative investigatory groups, established in 1995, that set principles and policies, as well as training and enabling communication while sharing technology and investigations among member countries, promoting the founding of new FIUs. FinCEN is the FIU within the USA.
The Clearinghouse paper provides a roadmap on how domestic intergovernmental agencies can improve their working arrangements and agreements. The more efficiently domestic and international agencies can work together the more effectively they can fight against transnational organized crime. Each organization updates their processes on a regular basis, although often far behind the frontlines of the ever-changing criminal inventiveness to work around any improvements.

1.2. Defining fraud issues

Machine learning does not find fraud. Machine learning algorithms learn using known fraud cases and then make inferences on new data, finding similar configurations that may be fraudulent transactions. These results are then sent to fraud investigators for confirmation, which is a labor-intensive endeavor. Due to the extraordinarily high volume and complexity of transactions that take place daily in business, machine learning has become an invaluable tool used to identify potential fraud cases. Fraudulent transactions are rare cases of interest marked as the positive minority class with the normal transactions categorized as the negative majority class. The skewed data set requires special handling for the class imbalance, which favors the majority class, often ignoring the minority class due to the large disparity in their relative numbers; this leads to inaccurate metrics and results. The two-class problem requires supervised machine learning using binary classification modeling.

Identification of illegal activity may be made through system monitoring for thresholds or by using filters, software matching fraud typologies, anomaly detection, or human detection by a customer complaint, law enforcement, or a tip from a customer-facing employee, among other sources. One of the major challenges is that fraudulent transactions may appear as ordinary, normal transactions but are unlawfully committed through trickery or deceit. For example, identity theft may appear as a genuine business operation with the fraud pattern identified from a source not within the electronic event. The transaction may be identified as fraudulent by law enforcement detective work with the fraud pattern not entirely appearing within the electronic business data or operation. If the fraud pattern information is not added into the electronic event flagged as fraudulent in the data set, it will predict similar non-fraudulent examples as True Positive (TP) transactions (which is the definition of a False Positive (FP)). Compare this example to cancer detection, where the actual lab test observations can be included in the data set with the indicators of cancer inherently existing in the data. For machine learning to identify similar patterns of fraud, additional relevant information may need to be reintroduced into the feature attributes or values.

An added complication for fraud patterns is the possibility of a lack of consistent structure, since criminals are working to obscure their identity, attempting to get away with the crime that uses a “throw everything at the wall approach” until something works, while making the paper trail as complex as possible to unravel. Contrast this with fields in which data set observations in math, science, or technology have systematic patterns with predictable organization. Although ever-changing, regular and repeated order may be found in the fraud typology and possible electronic systems paths to accomplish the crime before controls limit access.

The ACFE and the Association of Certified Money Laundering Specialists (ACAMS) have documented fraud typologies. These roadmaps are useful for understanding the crime and electronic fraud patterns. Further, they are useful for ascertaining which data fields must be added to the data set, in addition to the fields that must be developed to optimize the predictive capabilities of the machine learning system to increase accuracy while reducing FPs and False Negatives (FNs).

1.3. Research objective

This is an investigative study designed to build a baseline foundation for future work in the application of machine learning for fraud detection. It is not a survey of all approaches, although the groundwork is supported by a vast survey of available research in machine learning for classification and resampling methods in handling the principle issue facing the study of fraud: how to deal with imbalanced data sets. The contributions of this study help answer the following research questions.

1. Do stacked generalizations in combination with resampling methods improve the identification of rare events in fraud data sets?
2. If so, which stack generalization resampling combinations work better than others?

In typical machine learning efforts, the stakeholders define the problem and provide data to the data science team. The modeling output is then returned to the stakeholders for analysis. In the case of fraud, the model that best generalizes the data set may not be finding meaningful new cases of fraud. This work explores how information processing and management solutions ensure the predictive modeling results reflect the correct classification matrix values.

This paper is organized as follows. Firstly, we review the relevant literature related to handling imbalanced data sets, to include discussion around the application of stacked generalizers and the contributions of this study. Secondly, we describe the methodology used in the study,
particularly around methods for handling imbalanced data sets at both the data level and algorithm level, the cost-sensitive learning (CSL) and ensemble methods. We also discuss the evaluation metrics employed along with the computational experimentation. Thirdly, we provide the results of our stacked generalizations using resampling methods for fraud detection. Fourthly, we discuss our findings as well as next steps. We then conclude our work and pose some future research directions.

2. Related work

Handling imbalance is not a matter of simply separating classes. Imbalance requires consideration of the data structure, small data set size with small minority samples that cause overfitting and deficient generalization,14 data quality, small disjuncts as a result of sparsity of data,15 weak separation and class overlap at the boundary,16 feature space sparseness caused by the imbalance of classes,17 di vide and conquer17 class separation that may cause data fragmentation,18 detection of rare events that may have a negative influence, causing noise,19 and that imbalance predisposes the minority class to small disjuncts that are more disposed to poor performance and errors caused by bias,19 noise, missing attributes, and data size.19 Another opinion regarding class imbalance is that the “volume, complexity of data, class imbalance, concept drift, class overlap, and class mislabeling” are challenges. “Data quality and detection accuracy” are closely linked, requiring the management of feature selection while keeping the detection rate high and reducing FPs.20

If the initial distribution method is not optimal, then which is? Weiss and Provost21 addressed this issue in a thorough but not exhaustive study when determining which class distribution to select for the reduced training set, which was a condensed version for data access and cost-sensitive purposes. Their testing showed that the minority class had a disproportionate number of errors and determined that a sampling strategy with the initial data set measured by accuracy and a rebalanced version using the receiver operating characteristic (ROC) curve performed as well but not worse. Chawla22 agreed with their assessment that the best distribution may not be the initial distribution method used when the differences are not significant. Sesmero23 found that imbalanced data sets are not suitable for model selection, or a combination of both.17 Data-level resampling rebalances the ratio between minority and majority classes, since their disparity in numbers has a higher misclassification cost for minority samples than the majority, which leads to metric evaluation errors.23 Classification of the minority class does not improve with the duplication of samples in random over-sampling23–25. A discovery using a more intelligent and focused approach called SMOTE skews the minority over-samples just slightly, making the decision boundary between classes less specific and, thus, improving predictive performance.23 This technique can be used to under-sample the majority with a similar affect.

A means to balance the cost of misclassifying the minority class without the use of resampling is by CSL, which makes the cost of misclassifying the minority class greater than the majority, rebalancing the relationship.26 This study found that random over- and under-sampling performed similarly to CSL. A study by Maalouf and Trafalis19 also found that CSL and resampling achieved similar results. Weiss et al.27 found that CSL performed better with more data. Examples are MetaCost, AdaCost, CSBI, CSB2, RareBoost, AdaC1, AdaC2, and AdaC3. Improvements in feature selection can be made by association rules rather than correlation alone.28 This improves performance by reducing the data set. Reducing dimensionality may also lose invaluable data that can be combined using association rules, which reduces noise.20 Zheng et al.29 found that imbalanced data sets are not suited to traditional feature selection and instead also proposed combining features. Ensemble applications include more advanced cross-validation techniques that incorporate more successful model selection approaches.30 Ting and Witten31 noted that class probabilities are a fundamental requirement for applying stacked generalizers, which were found to be better performers than single classifiers. Methods with data preprocessing include RUSBoost, SMOTEBoost, SMOTEBagging, EasyEnsemble, BalanceCascade, RareBoost, and EUSBoost. Hybrid combinations appear to compensate for the weaknesses inherent in a single routine. Many algorithms enable a cost weighting parameter to be set. Examples are the cost-sensitive ensemble,32 cost-sensitive Support Vector Machine (SVM) and Query by Committee (QBC) using the SVM, C.45, NB, and AdaBoost. There are often different and opposing outcomes documented in research conclusions, which may be a reflection of the differences in the data set structure, source, size, variances in development decisions, and the combination of data sets used when the differences are not significant. Sesmero
et al.\textsuperscript{33} also noted many contradictions in research results with no consensus regarding optimal stacking parameters. They believed these could be reasons why the technique is used less often than straightforward bagging and boosting in real-world problems. The complex evaluation process, as well as high computational costs, contribute to reasons why the technique is less often applied. Classification problems can be completed off-line, making the higher overhead for stacking still a viable option.

Naimi and Balzer\textsuperscript{34} note that stacking is increasingly used in epidemiology, but it is observed that prediction is separate from the recognition of causal effects. The complexity of algorithms and stacks of algorithms is opaque and considered a black box rather than an identification of causal relationships. The unknowns that are not distinguished by the meta learner in the data set include “confounders, instrumental variables, mediators, colliders, and the exposure.” Therefore, one needs to carefully think through the causal structure rather than taking the results at face value. Wong et al.\textsuperscript{35} used a cost-sensitive Neural Network (NN) ensemble method of Stacked Denoising Autoencoders on six data sets, as well as Cost-sensitive Deep Neural Network (CSDNN) and Cost-sensitive Deep Neural Network Ensemble (CSDE) models, against several typically used machine learning algorithms, including Logistic Regression (LR), NN, SVM, Bayesian Network (BN), Decision Tree (DT), boosting algorithms (LogitBoost, AdaBoost, RUSBoost, RBBost, SMOTEBoss, Bagging, Voting, and BalanceCascade), and cost-sensitive algorithms (AdaCost and MetaCost). The test results showed that the CSDE and CSDNN models performed first and third best, respectively, for True Positive Rate (TPR), True Negative Rate (TNR), and area under the curve (AUC) metrics in average rankings, as well as low generalization gaps between training and test results.

Increasingly, stacked generalization in conjunction with data-level resampling is used as the method of choice in scientific applications and where machine learning is being applied to improve generalization in the data set. Given the work of Chawla et al.\textsuperscript{23} that developed the SMOTE technique, this study develops a similar solution for fraud detection by leveraging the $k$-nearest neighbor ($k$NN) method\textsuperscript{36} and applying mathematical formulas that identify the inherent characteristics of fraud data using distance formulas. Further, Wang et al.\textsuperscript{37} studied the automated feature selection processes, which found that the quality of the data set and its subsequent performance are better matched with the actual object of prediction. Fraud investigators can better find the best features based on the fraud incident, while confirming that the necessary data is contained in the data set. The automated feature selection algorithm assumes that all necessary data is present and the best-performing algorithm would determine the best features. This study assumes that all the necessary data is not present and must be confirmed through reverse engineering the actual fraud event. An automated feature selection would improve prediction after the confirmation and validation process by fraud investigators. Finally, Ting and Witten\textsuperscript{38} conclude that multiple classifiers are better than single classifiers, which is similar to the findings in this study.

3. Methodology

The scarcity of publicly available fraud data sets is one of the most significant hurdles to overcome in the development of machine learning models for fraud detection. This study investigates stacked generalizations on fraud imbalanced data sets with resampled data. Given the unique characteristics of fraud data relative to other publicly available two-class data sets, including data other than fraud data would not serve the research conditions. Although fraud typology patterns may repeat, fraud data may not be systematic nor orderly and can follow complex deceptive schemes to commit the crime, as well as obfuscate the identity of the perpetrator. Publicly available two-class data sets generally are sourced from science studies and have complicated systematic and/or orderly patterns. The binary classification separation of the distributions into classes would be relevant, but the relationships in the data set structure and metrics from the resampling and predictions would be less applicable.

Rather than focus on experiments across data sets, the study strategy is centered on a within the data set comparative analysis tactical approach regarding how data- and algorithm-level techniques perform in combination, as well as how to select the best-performing machine learning model. A single data set is used to delve deeper into the changes and relationships that arise in preprocessing through the algorithm prediction phases. Only one data source met the study requirements, which was a credit card fraud data set (creditcard.csv) from the Machine Learning Group at the Université Libre de Bruxelles in Brussels, Belgium.\textsuperscript{39} The data set has 284,807 entries, of which 492 are known to be fraudulent and 284,315 are normal; this data set has an extreme imbalance ratio (IR) of 0.17%, making its analysis challenging. The scarcity of publicly available fraud data sets is one of the most significant hurdles to overcome in the development of machine learning models for fraud detection. This study investigates stacked generalizations on fraud imbalanced data sets with resampled data. Given the unique characteristics of fraud data relative to other publicly available two-class data sets, including data other than fraud data would not serve the research conditions. Although fraud typology patterns may repeat, fraud data may not be systematic nor orderly and can follow complex deceptive schemes to commit the crime, as well as obfuscate the identity of the perpetrator. Publicly available two-class data sets generally are sourced from science studies and have complicated systematic and/or orderly patterns. The binary classification separation of the distributions into classes would be relevant, but the relationships in the data set structure and metrics from the resampling and predictions would be less applicable.

Rather than focus on experiments across data sets, the study strategy is centered on a within the data set comparative analysis tactical approach regarding how data- and algorithm-level techniques perform in combination, as well as how to select the best-performing machine learning model. A single data set is used to delve deeper into the changes and relationships that arise in preprocessing through the algorithm prediction phases. Only one data source met the study requirements, which was a credit card fraud data set (creditcard.csv) from the Machine Learning Group at the Université Libre de Bruxelles in Brussels, Belgium.\textsuperscript{39} The data set has 284,807 entries, of which 492 are known to be fraudulent and 284,315 are normal; this data set has an extreme imbalance ratio (IR) of 0.17%, making its analysis challenging. The scarcity of publicly available fraud data sets is one of the most significant hurdles to overcome in the development of machine learning models for fraud detection. This study investigates stacked generalizations on fraud imbalanced data sets with resampled data. Given the unique characteristics of fraud data relative to other publicly available two-class data sets, including data other than fraud data would not serve the research conditions. Although fraud typology patterns may repeat, fraud data may not be systematic nor orderly and can follow complex deceptive schemes to commit the crime, as well as obfuscate the identity of the perpetrator. Publicly available two-class data sets generally are sourced from science studies and have complicated systematic and/or orderly patterns. The binary classification separation of the distributions into classes would be relevant, but the relationships in the data set structure and metrics from the resampling and predictions would be less applicable.

Rather than focus on experiments across data sets, the study strategy is centered on a within the data set comparative analysis tactical approach regarding how data- and algorithm-level techniques perform in combination, as well as how to select the best-performing machine learning model. A single data set is used to delve deeper into the changes and relationships that arise in preprocessing through the algorithm prediction phases. Only one data source met the study requirements, which was a credit card fraud data set (creditcard.csv) from the Machine Learning Group at the Université Libre de Bruxelles in Brussels, Belgium.\textsuperscript{39} The data set has 284,807 entries, of which 492 are known to be fraudulent and 284,315 are normal; this has an extreme imbalance ratio (IR) of 0.17%, making its development results directly applicable to the main investigative study objective. Because only a single data set was examined in this study, no conclusions can be drawn until more fraud data sets are studied in future research efforts.

3.1. Handling imbalanced data sets

In this study, a combination of data-level resamples (in addition to a full data set) and 11 algorithm classifier pairs were used to handle class imbalance for a combination of 88 baseline models. The algorithm descriptions are
In kNN, an odd number for method since it does not model the training data but uses Data-level imbalance handling happens at an initial phase to rebalance the class IRs, which then go through the algorithm-level handling phase. Seven resampling techniques were chosen for prepossessing: random over- and under-sampling, SMOTE, SMOTEENN, BorderlineSMOTE, SMOTETomek, and ADASYN (Adaptive Synthetic Sampling Technique), in addition to using the full data set for comparison purposes. Random over-sampling keeps the majority class and randomly over-samples the minority class by the class difference with replacement. The result is that the new minority samples are duplicates of existing samples causing overfitting. Random under-sampling (RUS) keeps the minority class and randomly under-samples (removing entries) from the majority class, losing valuable data in the routine, which can be done with or without replacement.

The SMOTE technique over-samples the minority using a more intelligent sampling application that improves prediction using kNN classification. It is considered a Lazy method since it does not model the training data but uses both training and test data in the “just in time” prediction. In kNN, an odd number for \( k \) is selected in order that the voting algorithm always has a winner. The first point is chosen from which the \( k \) number of neighbors is calculated of each minority class feature vector by the shortest distance. Then, each point votes positioned on its own class with the majority vote, deciding to which class the initial point belongs. This information is kept in an array for the next step.

The minority class is then over-sampled, with the new synthesized minority vector not being a duplication of previous entries and, thus, reducing overfitting. The number of over-samples, with the majority class less than the minority class, are calculated, and each new sample is taken from the distance of the line segment between the minority class feature vector sample selected and its nearest neighbor (from the stored array) measurement. This new sample is then multiplied by a random number between 0 and 1. These new synthetic samples create a less specific and general decision boundary line, improving classification prediction. The higher the \( k \) value the more even the decision boundary. The lower the \( k \) value the rougher the decision boundary.

There are many different distance measures that can be applied, with the most common being the Euclidean distance, which is the ordinary straight-line distance between two points \((x_1, y_1)\) and \((x_2, y_2)\). Other possible options are building kNN from scratch by means of a similar SMOTE-like approach using another distance measurement. The optimal distance measure for a data set may be determined by the data structure and mathematical relationships of the data at a more abstract plane.

For example, in a National Health Institute study the Euclidean, cosine, Chi square, and Minkowsky distance measures were compared and it was found that the Chi square distance performed better for medical domain data sets. Sentimentality prediction found that the cosine improved the discovery of boundaries between classes. The optimal distance measurement for fraud transactions has not been explored and is very challenging by the lack of systematic and organized ways fraud is committed and the deficiency of publicly available data sets for researchers to access for their work. On reflection, finding that the cosine supports sentimentality prediction, mathematical relationships would not have been an obvious choice because of the complexity and boundless possible ways of writing text. Therefore, the possibility of recognizing a core mathematical relationship for fraud typologies is quite promising.

The other intelligent resampling techniques implemented in this study, SMOTETomek, SMOTEENN, BORDER LINESMOTE, and ADASYN, are explained in greater detail in Table 1.

3.1.2. Algorithm level. Any number of algorithms can be applied with or without resampling techniques. The test strategy in this study was to compare a diverse but not a comprehensive list of machine learning classifiers, including those based on probability (Naive Bayes), classification (DT - C4.5, kNN, and SVM), and NNs (multi-layer perceptron (MLP)). Naive Bayes (GaussianNB) has its foundation in the Bayes rule named after Thomas Bayes (1702–1761), which provides the logic for conditional probability. The Naive Bayes algorithm can be built manually and is the ratio of two frequencies.

DT classifier examples include ID3, C4.5, and CART, which are the three fundamental DTs. C4.5 was developed from ID3 (Iterative Dichotomer 3); both were discovered by Ross Quinlan and build DTs piece by piece in a top-down fashion, determining the best advantage. kNN separates classes by first calculating the number of nearest neighbors then measures the distance from each point to its nearest neighbors to vote upon to which class the point should belong. For a more detailed explanation, see SMOTE above.

SVMs seek to find a plane that separates the classes, which is a concept that was first introduced by Vladimir Vapnik PhD in 1990. Twenty years after it was introduced, it is commonly accepted as one of the best classification methods. The goal is to design a hyperplane to separate a binary classifier labeled as a positive or negative point. The MLP Classifier is a MLP classifier NN
back propagation approach that learns the structure in the data without first imposing assumptions on the data. The learning by the hidden belief layer is enhanced in multiple layers in a progressive recursive function to gain learning patterns successively from the data.

### 3.1.3. Cost-sensitive learning

Handling imbalanced data sets is essentially a weighting problem that has been solved with straightforward, as well as creative, resampling solutions. CSL uses weighting on the full imbalanced data set, which has been found to be very effective. The cost of misclassification on the minority class is weighted or penalized more heavily than a correct classification. The true costs include the costs of misclassification, testing, and human intervention, as represented in the cost matrix (Table 2) initially developed by Hand et al. and Bahnsen et al. The administrative costs of a FP ($C_{FPi} = C_{admin}$) are equal to the TP ($C_{TPi} = C_{admin}$), since in the latter instance the account holder needs to be contacted. The cost of a FN ($C_{FNi} = 100C_{admin}$), missing the fraud, is significantly greater and represented by the actual loss of the transaction amount ($C_{FNI} = A_{MTI}$).

There are two techniques of CSL that can be applied through machine learning. The direct method alters accuracy-based algorithms in a way that includes cost weighting for misclassification. Since the algorithm is developed in a way that rewards lower costs, raising the weight of misclassification will over correct and, therefore, correctly classify. Weights of misclassification that are higher for the FPs than FNs vary inversely as the class size increases. The indirect method applies a wrapper to the algorithm, effectively altering how the misclassification is weighted. One primary issue is that the initial weighting cost is unknown, requiring that a starting point be determined.

For example, a direct method cost-sensitive tree decides how to build branches being controlled by entropy introduced by an information cost factor in selecting how to add attributes. Information theory, usually referenced as entropy, quantifies irregularity and measures class distribution consistency. In this study, the Random Forest algorithm was used with a cost-sensitive weighting parameter. CSL can be used on the whole data set, as well as with resampling, which can be an evaluation point to check whether preprocessing improves performance.

### 3.1.4. Ensemble methods

Ensemble methods can be used with classifiers (AdaBoost, BaggingClassifier, EasyEnsemble, RUSBoost, GradientBoostingMachine (GBM), etc.), with tacked generalizers, or with other hybrid approaches. They are often referred to as a mixture of experts that include cross-validation and voting strategies to improve prediction outcomes. The BaggingClassifier is an ensemble meta-estimator that divides the sample into randomly selected groups with replacement. Selection strategies include cross-validation with bagging that uses majority voting rather than winner-takes-all routines. The predictions are combined, which may overfit with large trees. In Boosting, weak classifiers with higher error rates are combined with reweighting with voting that makes them strong classifiers. Strong and weak classifiers are measured by their error rates, with the former with a rate close to zero and a 100% prediction accuracy and the latter an error rate less than or equal to the chance of a 50:50 outcome.

The goal is to find a solution that reduces variance and, therefore, does not overfit and does not increase bias, which cannot resolve difficult machine learning challenges. Weak and strong classifiers may not always work in the same consistent way in the same areas. By first categorizing the strong classifier with the lowest error rates rather than a weak classifier, boosting can find weights to improve performance through maximizing the likelihood of the inverse of the weights equaling one, resulting in the reduction of overfitting.

Adaptive Boosting (AdaBoost) works in a similar way in that CSL adjusts weights of misclassification with correct classification and, in fact, the boosting mechanism...
was developed to adapt CSL with boosting strategies,\textsuperscript{59} incorporating a weighted majority vote deciding the result. It can be used on the whole data set, which may be an evaluation point to check whether using AdaBoost improves performance. The GBM is a boosted NN classifier that makes no assumptions about the data (i.e., it is non-parametric) and bases the outcome from relationships learned in the modeling process. The GBM builds on the boosting techniques using gradient descent to find the local minimum of the function and then builds new base learners iteratively to improve accuracy.\textsuperscript{60}

EasyEnsemble uses RUS of the majority class that iterates through the samples, first training then combining the model predictions.\textsuperscript{61} RUSBoost applies RUS of the majority class before using boosting techniques of weighting then majority voting. RUSBoosting is a simpler sampling technique compared to SMOTEBoost, which applies the more complex SMOTE technique with the kNN classifier.\textsuperscript{62} EasyEnsemble and RUSBoost both use bagging as the principal approach and AdaBoost as the classifier.\textsuperscript{61}

Stacked generalizers provide an additional layer of generalization error correction, improving upon using a single classifier, which is helpful because even the smallest improvement in fraud detection can have significant and far-reaching ramifications. Designing the machine learning system includes combining data- and algorithm-level development using the stacked generalization architecture. The purpose is to test baseline and stacked classifier pairs with meta learners and combine their outcomes to compare which machine learning models have the optimal performance. Meta learners of stacks outperform baseline learners but are not statistically significant, although the improvement of the total cost of losses in the transactions in production may be substantial.

The Stacked Generalizer is a hybrid method introduced by David Wolpert to minimize the error rate using ensemble stacking to learn the bias in the initial individual classifiers with the output of the initial models.\textsuperscript{38} This is referred to as level-0, which then provides the input into the level-1 meta-learner phase that corrects for the bias identified using cross-validated prediction, improving majority voting or weighting rather than using the winner-takes-all strategy, thereby enhancing the accuracy of the classifiers being stacked.\textsuperscript{30}

Stacking ensembles combine bagging and boosting with different algorithms, in which the strengths and weaknesses in each model are weighted, thus promoting the weak classifiers and demoting the stronger classifiers in each resampling round. This focuses the algorithm on fixing misclassifications with only the improvements in accuracy included.\textsuperscript{58} Rather than using a single result from an algorithm, stacked generalizations (see the architecture depicted in Figure 1) combine method results, improving on the bias of each in the collection.\textsuperscript{30} The bias of the individual classifiers are learned and then filtered out in the subsequent level-1 phase.\textsuperscript{30} Wolpert\textsuperscript{30} noted that the rules for selecting level-0 and level-1 classifiers were unknown at the time of his writing in 1992. Yan and Han\textsuperscript{63} noted that stacking generalizations are not usually applied to imbalanced data sets, which is the core subject matter of this study.

### 3.2. Evaluation metrics

A machine learning model learns the patterns from the training set, which then makes inferences on test data to determine whether the new instances match these patterns (see the classification via the confusion matrix in Table 3). The actual and predicted outcomes are categorized as TP (both occurred and predicted), FP (did not occur but was predicted, which is an incorrect prediction), true negative (neither occurred nor predicted), and FN (occurred but not predicted, representing an actual fraud that was missed by the machine learning classifier). The numbers of each category are aggregated to build meaningful statistics. The classification matrix also provides a threshold variable parameter from greater than zero to 100%, with the default being 50% or the prediction being no better than chance. The higher the threshold, the fewer predictions will be included in the matrix, although with more applicable predictions.

The most commonly used metric is the scalar accuracy value $(TP + TN) / (TP + TN + FP + FN)$, which is not useful for imbalanced data sets since the minority class is under-represented and is not a good measure of classifier performance.\textsuperscript{64,65} Since the misclassification costs of FPs (cost of a fraud investigator reviewing the event) and FNs (total loss of the transaction if the FN is found to be accurate) are not included. The AUC, an array of the entire threshold range of accuracy, is more effective and efficient in gauging classifier performance. Once the data is rebalanced, accuracy becomes more useful.

Precision and recall, which are used to build the F-score, are more applicable to the unequal class distributions in skewed data since they include FPs and FNs. Precision measures the percentage of correct predictions with respect to the total correctly predicted, determining when correctly predicted and how often and measuring the
exactness. High precision is preferred, and it should be used when the cost of FPs is high, such as through reputation loss of falsely accusing a customer of fraud. Precision is used as the $y$-axis in the precision–recall curve. Recall (also known as sensitivity and TPR) calculates the proportion of positives correctly predicted; a low recall is preferred. Recall is used as the $x$-axis in the precision–recall curve and $y$-axis in the ROC curve. The difference between precision and recall are the FP and FN in the numerator. FPs have associated investigative costs that can be interpreted to be direct overhead costs, and FNs are opportunity costs of having missed the prediction, which is a loss of the value of the business event that could have been spent elsewhere. Quantifying the direct and opportunity costs provides a more consequential means of discovering the weight of the precision versus recall trade-off.66

The F1-score calculates the trade-off between precision and recall that is moderated by using a threshold greater than 0–100%. The higher the precision (if correctly predicted, how often and exact) the lower the recall (correct predictions). The preference is an F1-score of 1 with a high precision and high recall, which identifies the optimal balance, including with imbalanced data sets.67 The False Positive Rate (FPR) computes incorrect predictions or positive predictions incorrectly classified. The FPR is used in the $x$-axis of the ROC curve. The AUC quantifies the ROC performance and summarizes accuracy for the entire test result outcomes, which represent the threshold range rather than using just the scalar accuracy value.65 The visualization is a graph with the TPR on the $y$-axis and the FPR (i.e., $1 -$ specificity) on the $x$-axis. The optimal cutoff is where sensitivity is relatively equal to specificity. The diagonal line from (0,0) to (0,1) is the 50% threshold representing random selection. The AUC is indifferent to skewed data and can be summarized using a scalar AUC value.65

Although some metrics may not be relevant or useful for evaluating classifier performance on imbalanced data sets, the prehandling technique of resampling balances the data distribution and modifies the original shortcomings of these measurements, although perhaps not mitigating them fully since the change in balance is not so much adding new data as using creative techniques to equalize the number of observations in each class. Before the data is relabeled, the AUC may be overly optimistic;68 therefore, the

---

**Figure 1.** Stacked generalization machine learning architecture. AUC: area under the curve; ROS: random over-sampling; MLP: multi-layer perceptron; SMOTE: Synthetic Minority Over-sampling Technique; kNN: k-nearest neighbor; NB: Naive Bayes; CSL: cost-sensitive learning; SVM: Support Vector Machine; GBM: GradientBoostingMachine.

**Table 3.** Machine learning model classification matrix.

| Actual | $y_i = 1$ | $y_i = 0$ |
|--------|-----------|-----------|
| Predicted $y' = 1$ | TruePositive | FalsePositive |
| $y' = 0$ | FalseNegative | TrueNegative |

Kerwin and Bastian
precision–recall curve is a better tool for evaluation of machine learning classifier performance on skewed data, providing more discriminating information.

3.3. Computational experimentation

A reusable and reconfigurable test harness was built with viable algorithm and sampling options to evaluate a variety of resampling plans, baseline machine learning classifiers, and meta learners (each of which can be easily substituted) to help keep a broad range of combination options available to investigate the best production configuration for the top project strategy. The testing consists of the following: (1) preprocessing for resampling of the data set; (2) modeling and predicting the level-0 baseline classifier resample pairs; and (3) building stacks from the level-0 findings to be predicted with the level-1 meta learners. Eight data samples were teamed with each of 11 machine learning classifiers, totaling 88 level-0 baseline models, using the following resampling strategies and classifiers (see Table 4). After the baseline level-0 models were predicted, their outcomes were sorted by the AUC and F1-score to find the strongest and weakest classifier resample pairs, with eight stacks developed for test purposes.

- Resampling mechanism: full, ROS, RUS, SMOTE, SMOTETomek, SMOTEENN, BorderlineSMOTE, ADASYN.
- Machine learning classifier: kNN, GaussianNB, C4.5, CSL, SVM, EasyEnsemble, RUSBoost, AdaBoost, GBM, Bagging, and MLP.

These eight stacks were remodeled against each of the 11 meta-learner classifiers with the original baseline entries also present in the level-1 phase for contrast, which totaled (88 + 16) 104 predictions. The results of the level-0 and level-1 predictions were analyzed and then combined to see if and where improvements were obtained with data resamples and classifiers, and whether level-1 predictions provided notable benefits, reducing FPs and FPs while increasing TPs. The final outcome was sorted in order by the AUC, F1-score, and accuracy.

4. Modeling results

The model development findings were reviewed in three phases: (1) level-0 baseline; (2) level-1 stack generalizations with meta learners; and 3) combining both level-0 and level-1. From Table 5, the top 17 entries produced an AUC of 96%, of which 13 utilized more informed resampling techniques (ADASYN, SMOTE, and SMOTE variants), one full data set, and three ROS/RUS resamples. The top six entries had very similar metrics. The top three predictions were applied with the MLP classifier with a SMOTE/ BORDERLINESMOTE or random over-sampling (ROS) approach, each returning very similar metrics. The ROS MLP entry would most likely be chosen for production on cost factor alone, since the ROS technique has a lower computation processing and time overhead.

The 104 predictions for the meta learners and stack elements displayed three predictions with AUC metrics of 97% (see Table 6), which was followed by a group of 24 entries with an AUC of 96%. Twenty-four of the top 27 AUC entries were meta learners, with the best groups representing seven of the eight different stack combinations and the top AUC models representing three stacks types.

From the constructed combinations, the meta learners appear to have improved seven of the eight stacks (1, 2, 3, 5, 6, 7, and 8). The top entry had a reasonable FP statistic, with the second and third having much higher FP values of 523 and 4918, respectively, creating a less likely scenario that these would be the selected model (see Table 7). The top three models had very different TP and FP values, suggesting uncertainty around which model should be selected with data resamples and classifiers, and whether level-1 predictions provided notable benefits, reducing FPs and FPs while increasing TPs. The final outcome was sorted in order by the AUC, F1-score, and accuracy.

### Table 4. Stacks of classifier and resample pairs from level-0 models.

| Metric | Stack | Classifier 1                      | Classifier 2                      |
|--------|-------|-----------------------------------|-----------------------------------|
| F1-score #1 | w-ADASYN | EasyEnsemble | s-BORDERLINESMOTE kNN |
| F1-score #2 | w-BORDERLINESMOTE RUSBoost | | w-SMOTE GaussianNB |
| F1-score #3 | s-SMOTEENN MLP | | s-SMOTE CSL |
| F1-score #4 | tree-RUS CSL | | tree-RUS C4.5 |
| AUC #5 | w-BORDERLINESMOTE C4.5 | | s-ROS MLP |
| AUC #6 | w-ADASYN BAG | | w-SMOTE SVM |
| AUC #7 | s-SMOTE Tomek GaussianNB | | s-SMOTEENN AdaBoost |
| AUC #8 | tree-ROS C4.5 | | tree-full CSL |

ADASYN: Adaptive Synthetic Sampling Technique; SMOTE: Synthetic Minority Over-sampling Technique; kNN: k-nearest neighbor; RUS: random under-sampling; NB: Naïve Bayes; MLP: multi-layer perceptron; CSL: cost-sensitive learning; ROS: random over-sampling; AUC: area under the curve; SVM: Support Vector Machine.
that is representative of optimal performance while representing statistics that are meaningful for identifying fraud.

When combining the level-0 and level-1 predictions, the top three models were the same as the top level-1 predictions, with an AUC of 97%. In comparing the classification metrics of the models with 96% AUC to the top 97% AUC models, the top 11 96% models had better FP statistics than two of the three 97% AUC models. In addition, the 96% AUC MLP ROS model used a single algorithm. As depicted in Table 8, the top 44 models had AUC values of 96% or 97%, of which 24 were meta learners, three were stacks, and 17 were baseline single algorithms.

The meta learners with informed resampling methods continued to demonstrate optimal metric performance in a combined level-0 and level-1 comparative analysis, achieving the top three models AUCs of 97% and improving on the best baseline single algorithm AUC of 96% (see Figure 2). In this combined list, three models were baseline level-0 pairs in positions 4–6 that performed very closely to the top three meta learners by AUC with better FP statistics, which makes model selection convoluted.

A noteworthy observation comes from the top three performing models, which are all meta learners representing stacks 6, 1, and 7. Stack 6 was comprised of the two weakest classifier pairs from AUC values and the Ensemble GBM classifier. Stack 1 combined the strongest and weakest classifier pairs from the F1-score with GaussianNB. Stack 7 consisted of two strong classifier resample pairs by the AUC with EasyEnsemble. Stacking by combining weaker classifiers based on the AUC returns better performance, which proved true in this case with the second and third highest models, 1 and 7, having much higher FP statistics.

The first and only time that the full data set was listed appears in position 44/104 for the level-1 phase and 11 times in the combined list, starting at 31/192, which demonstrates that resampling did have a marked prediction improvement. CSL did not perform well; however, this may be due to the classifier choice and/or parameters chosen, since CSL does well in other stacking generalization studies.

Given the range of FPs with similar AUC scores, this demonstrates that the best machine learning model results

| Test run | Classifier | TP | FP | FN | TN | Accuracy | F1 rank | F1-score | AUC |
|----------|------------|----|----|----|----|----------|---------|----------|-----|
| 0SMOTE   | MLP        | 142| 20 | 43 | 113,718 | 0.9994 | 8 | 0.8184 | 0.96 |
| 0ROS     | MLP        | 142| 28 | 41 | 113,712 | 0.9994 | 9 | 0.8046 | 0.96 |
| 0BLSMOTE | MLP        | 150| 27 | 46 | 113,700 | 0.9994 | 10 | 0.8043 | 0.96 |
| 0BLSMOTE | GBM        | 147| 28 | 49 | 113,699 | 0.9993 | 11 | 0.7925 | 0.96 |
| 0SMOTEEEN| AdaBoost   | 131| 28 | 65 | 113,699 | 0.9992 | 12 | 0.7380 | 0.96 |
| 0BLSMOTE | AdaBoost   | 127| 28 | 69 | 113,699 | 0.9991 | 13 | 0.7237 | 0.96 |
| 0BLSMOTE | GaussianNB | 171| 671| 25 | 113,056 | 0.9939 | 14 | 0.3295 | 0.96 |
| 0SMOTETomek | GaussianNB | | | | | | | |
| 0BLSMOTE | GaussianNB | 170| 731| 31 | 112,991 | 0.9933 | 15 | 0.3086 | 0.96 |
| 0ROS     | GaussianNB | 160| 697| 23 | 113,043 | 0.9937 | 16 | 0.3077 | 0.96 |
| 0full    | GaussianNB | 163| 706| 29 | 113,025 | 0.9934 | 17 | 0.3073 | 0.96 |
| 0SMOTEEEN| GaussianNB | 169| 735| 27 | 112,992 | 0.9933 | 18 | 0.3072 | 0.96 |
| 0ADASYN  | GaussianNB | 149| 668| 26 | 113,080 | 0.9939 | 19 | 0.3040 | 0.96 |
| 0SMOTE   | GaussianNB | 155| 695| 30 | 113,043 | 0.9936 | 20 | 0.2996 | 0.96 |
| 0BLSMOTE | EasyEnsemble | 178| 4874| 18 | 108,853 | 0.9571 | 21 | 0.0678 | 0.96 |
| 0SMOTEEEN| EasyEnsemble | 177| 6876| 19 | 106,851 | 0.9395 | 22 | 0.0488 | 0.96 |
| 0ADASYN  | EasyEnsemble | 159| 7304| 16 | 106,444 | 0.9357 | 23 | 0.0416 | 0.96 |

Table 5. Level-0 baseline with highest area under the curve (AUC; above 96%).

| Count | Type | Metric | Stack | Classifier 1 | Classifier 2 |
|-------|------|--------|-------|--------------|--------------|
| 1     | Meta | AUC    | #6    | w-ADASYN BAG | w-SMOTE SVM  |
| 1     | Meta | F1-score | #1    | w-ADASYN EasyEnsemble | s-BORDERLINE-SMOTE kNN |
| 1     | Meta | AUC    | #7    | s-SMOTETomek GaussianNB | s-SMOTE_ENN AdaBoost |

Table 6. Top level-1 by area under the curve (AUC; above 97%).

TP: True Positive; FP: False Positive; FN: False Negative; SMOTE: Synthetic Minority Over-sampling Technique; ROS: random over-sampling; RUS: random under-sampling; ADASYN: Adaptive Synthetic Sampling Technique; MLP: multi-layer perceptron; GBM: GradientBoostingMachine; NB: Naïve Bayes.
are not definitive from statistics alone and need to be sent to fraud investigators for review; this will be the final determination of the success or failure of the machine learning modeling process. This is primarily due to the unique conditions found in fraud data in that these crimes are committed through trickery and deceit, as well as being a moving target due to criminals constantly learning to game the system as soon as they play out each back door that closes when system controls are updated. Choosing a model on statistics alone, without reviewing if the metrics actually match confirmed fraudulent transactions during at least an initial exploration and development effort with periodic reviews, would be a guess based on numbers alone. As depicted in Figure 3, the precision-recall curve demonstrates that the statistics within each test run show great variation within the individual classification statistics.

### 5. Discussion

The question is, which are the better predictions? After fraud investigation review, delving more deeply into the classification matrix metrics will provide a better understanding of the most useful model(s). Once the classification values are verified and the machine learning algorithms are rerun, leading to the selection of the best model, then the processing time and computational resources are evaluated.

The first step in the review of the model results incorporates the fraud investigator’s assessment of TP, FP, and FN events for the top models, which is a necessary labor-intensive process before the final model selection. In reviewing the top performers of the combined final list, there is quite a difference in their classification statistics for the top AUC performers. The FP values range from 19 to 4918, which translates into a variance of 4899 more administrative case reviews for fraud investigators. The TP and FP results should be analyzed for why they failed to accurately predict, and all TP findings should be manually evaluated and assessed for possible formal investigations being established to find whether a crime was committed and then identify who committed the crime. An interdisciplinary process comprised of the fraud investigation, accounting, database, and data science teams can efficiently accomplish the continual feedback loop to support these investigations. As fraud investigators make improved

### Table 7. Level-I meta learner with stack predictions by the area under the curve (AUC).

| Test run | Classifier     | TP  | FP  | FN  | TN  | Accuracy | F1-score | AUC  |
|----------|----------------|-----|-----|-----|-----|----------|----------|------|
| 6metalearner | GBM  | 96  | 19  | 46  | 85,281 | 0.9992   | 0.7471  | 0.97  |
| 1metalearner | GaussianNB | 122 | 523 | 22  | 84,775 | 0.9936   | 0.3092  | 0.97  |
| 7metalearner | EasyEnsemble | 133 | 4918| 13  | 80,378 | 0.9423   | 0.0511  | 0.97  |
| 5metalearner | GBM  | 116 | 17  | 40  | 85,269 | 0.9993   | 0.8028  | 0.96  |
| 7metalearner | MLP  | 108 | 16  | 38  | 85,280 | 0.9994   | 0.8000  | 0.96  |
| 1metalearner | MLP  | 108 | 18  | 36  | 85,280 | 0.9994   | 0.8000  | 0.96  |
| 7metalearner | GBM  | 109 | 19  | 37  | 85,277 | 0.9993   | 0.7957  | 0.96  |
| 5stackROS  | MLP  | 140 | 27  | 48  | 113,708 | 0.9993    | 0.7887  | 0.96  |
| 8metalearner | MLP  | 106 | 21  | 40  | 85,275 | 0.9993   | 0.7765  | 0.96  |
| 1metalearner | GBM  | 105 | 22  | 39  | 85,276 | 0.9993   | 0.7749  | 0.96  |
| 2metalearner | MLP  | 102 | 19  | 42  | 85,279 | 0.9993   | 0.7698  | 0.96  |
| 8metalearner | AdaBoost | 93  | 21  | 53  | 85,275 | 0.9991   | 0.7154  | 0.96  |
| 6metalearner | MLP  | 90  | 22  | 52  | 85,278 | 0.9991   | 0.7087  | 0.96  |
| 2metalearner | AdaBoost | 88  | 24  | 56  | 85,274 | 0.9991   | 0.6875  | 0.96  |
| 2metalearner | GaussianNB | 127 | 502 | 17  | 84,796 | 0.9939   | 0.3286  | 0.96  |
| 7metalearner | GaussianNB | 124 | 485 | 22  | 84,811 | 0.9941   | 0.3285  | 0.96  |
| 5metalearner | GaussianNB | 134 | 526 | 22  | 84,760 | 0.9936   | 0.3284  | 0.96  |
| 2stackSMOTE | GaussianNB | 179 | 719 | 30  | 112,995 | 0.9934   | 0.3234  | 0.96  |
| 8metalearner | GaussianNB | 127 | 521 | 19  | 84,775 | 0.9937   | 0.3199  | 0.96  |
| 6metalearner | GaussianNB | 120 | 503 | 22  | 84,797 | 0.9939   | 0.3137  | 0.96  |
| 3metalearner | GaussianNB | 125 | 547 | 20  | 84,750 | 0.9934   | 0.3060  | 0.96  |
| 7stackSMOTETomek | GaussianNB | 158 | 727 | 29  | 113,009 | 0.9934   | 0.2947  | 0.96  |
| 3metalearner | RUSBoost | 124 | 669 | 21  | 84,628 | 0.9919   | 0.2644  | 0.96  |
| 2metalearner | RUSBoost | 127 | 1068 | 17 | 84,230 | 0.9873   | 0.1897  | 0.96  |
| 2metalearner | EasyEnsemble | 131 | 4272 | 13 | 81,026 | 0.9498   | 0.0577  | 0.96  |
| 6metalearner | EasyEnsemble | 130 | 5053 | 12 | 80,247 | 0.9407   | 0.0489  | 0.96  |
| 1metalearner | EasyEnsemble | 128 | 5492 | 16 | 79,806 | 0.9355   | 0.0445  | 0.96  |

TP: True Positive; FP: False Positive; FN: False Negative; FP: False Positive; ROS: random over-sampling; SMOTE: Synthetic Minority Over-sampling Technique; GBM: GradientBoostingMachine; NB: Naive Bayes; MLP: multi-layer perceptron; RUS: random under-sampling.
decisions, changes may be made to the data set, variables, or observations to improve predictions so that the results are more meaningful rather than just having better generalization performance.

A strong reciprocal relationship between the data science and fraud investigation teams creates a continual improvement update loop based on the initial briefing and debriefing from each team at the start and completion of their work in order to capture and update all lessons learned and best practices. Feedback from the investigation team may provide useful information for data scientists in how to improve the quality of the data set to support investigation outcomes. Machine learning is not the end goal but one excellent tool that is a link in a chain of many work efforts to help recognize fraudulent business operations.

### Table 8. Combined level-0 and level-1 predictions by the area under the curve (AUC).

| Test run       | Classifier     | TP  | FP  | FN  | TN  | Accuracy  | F1-score | AUC  |
|----------------|----------------|-----|-----|-----|-----|-----------|----------|------|
| 6metalearner   | GBM            | 96  | 19  | 46  | 85,281 | 0.9992    | 0.7471   | 0.97 |
| 1metalearner   | GaussianNB     | 122 | 523 | 22  | 84,775 | 0.9936    | 0.3092   | 0.97 |
| 7metalearner   | EasyEnsemble   | 133 | 4918| 13  | 80,378 | 0.9423    | 0.0511   | 0.97 |
| 0SMOTE        | MLP            | 142 | 20  | 43  | 113,718 | 0.9994    | 0.8184   | 0.96 |
| 0ROS          | MLP            | 142 | 28  | 41  | 113,712 | 0.9994    | 0.8046   | 0.96 |
| 0BLSMOTE      | MLP            | 150 | 27  | 46  | 113,700 | 0.9994    | 0.8043   | 0.96 |
| 5metalearner   | GBM            | 116 | 17  | 40  | 85,269 | 0.9993    | 0.8028   | 0.96 |
| 7metalearner   | MLP            | 108 | 16  | 38  | 85,280 | 0.9994    | 0.8000   | 0.96 |
| 1metalearner   | MLP            | 108 | 18  | 36  | 85,280 | 0.9994    | 0.8000   | 0.96 |
| 7metalearner   | GBM            | 109 | 19  | 37  | 85,277 | 0.9993    | 0.7957   | 0.96 |
| 0BLSMOTE      | GBM            | 147 | 28  | 49  | 113,699 | 0.9993    | 0.7925   | 0.96 |
| 5stackROS     | MLP            | 140 | 27  | 48  | 113,708 | 0.9993    | 0.7887   | 0.96 |
| 8metalearner   | MLP            | 106 | 21  | 40  | 85,275 | 0.9993    | 0.7765   | 0.96 |
| 1metalearner   | GBM            | 105 | 22  | 39  | 85,276 | 0.9993    | 0.7749   | 0.96 |
| 2metalearner   | MLP            | 102 | 19  | 42  | 85,279 | 0.9993    | 0.7698   | 0.96 |
| 0SMOTEENN     | AdaBoost       | 131 | 28  | 65  | 113,699 | 0.9992    | 0.7380   | 0.96 |
| 0BLSMOTE      | AdaBoost       | 127 | 28  | 69  | 113,699 | 0.9991    | 0.7237   | 0.96 |
| 8metalearner   | AdaBoost       | 93  | 21  | 53  | 85,275 | 0.9991    | 0.7154   | 0.96 |
| 6metalearner   | MLP            | 90  | 22  | 52  | 85,278 | 0.9991    | 0.7087   | 0.96 |
| 2metalearner   | AdaBoost       | 88  | 24  | 56  | 85,274 | 0.9991    | 0.6875   | 0.96 |
| 0BLSMOTE      | GaussianNB     | 171 | 671 | 25  | 113,056 | 0.9939    | 0.3295   | 0.96 |
| 2metalearner   | GaussianNB     | 127 | 502 | 17  | 84,796 | 0.9939    | 0.3286   | 0.96 |
| 7metalearner   | GaussianNB     | 124 | 485 | 22  | 84,811 | 0.9941    | 0.3285   | 0.96 |
| 5metalearner   | GaussianNB     | 134 | 526 | 22  | 84,760 | 0.9936    | 0.3284   | 0.96 |
| 2stackSMOTE    | GaussianNB     | 179 | 719 | 30  | 112,995 | 0.9934    | 0.3234   | 0.96 |
| 0SMOTETomek    | GaussianNB     | 171 | 694 | 30  | 113,028 | 0.9936    | 0.3208   | 0.96 |
| 8metalearner   | GaussianNB     | 127 | 521 | 19  | 84,775 | 0.9937    | 0.3199   | 0.96 |
| 6metalearner   | GaussianNB     | 120 | 503 | 22  | 84,797 | 0.9939    | 0.3137   | 0.96 |
| 0RUS           | GaussianNB     | 170 | 731 | 31  | 112,991 | 0.9933    | 0.3086   | 0.96 |
| 0ROS           | GaussianNB     | 160 | 697 | 23  | 113,043 | 0.9937    | 0.3077   | 0.96 |
| 0full          | GaussianNB     | 163 | 706 | 29  | 113,025 | 0.9935    | 0.3073   | 0.96 |
| 0SMOTETomek    | GaussianNB     | 169 | 735 | 27  | 112,992 | 0.9933    | 0.3072   | 0.96 |
| 3metalearner   | GaussianNB     | 125 | 547 | 20  | 84,750 | 0.9934    | 0.3060   | 0.96 |
| 0ADASYN        | GaussianNB     | 149 | 668 | 26  | 113,080 | 0.9939    | 0.3004   | 0.96 |
| 0SMOTE         | GaussianNB     | 155 | 695 | 30  | 113,043 | 0.9936    | 0.2996   | 0.96 |
| 7stackSMOTETomek| GaussianNB     | 158 | 727 | 29  | 113,009 | 0.9934    | 0.2947   | 0.96 |
| 3metalearner   | RUSBoost       | 124 | 669 | 21  | 84,628 | 0.9919    | 0.2644   | 0.96 |
| 2metalearner   | RUSBoost       | 127 | 1068| 17  | 84,230 | 0.9873    | 0.1897   | 0.96 |
| 0BLSMOTE       | EasyEnsemble   | 178 | 4874| 18  | 108,853 | 0.9571    | 0.0678   | 0.96 |
| 2metalearner   | EasyEnsemble   | 131 | 4272| 13  | 81,026 | 0.9498    | 0.0577   | 0.96 |
| 6metalearner   | EasyEnsemble   | 130 | 5053| 12  | 80,247 | 0.9407    | 0.0489   | 0.96 |
| 0SMOTETomek    | EasyEnsemble   | 177 | 6876| 19  | 106,851 | 0.9395    | 0.0488   | 0.96 |
| 1metalearner   | EasyEnsemble   | 128 | 5492| 16  | 79,806 | 0.9355    | 0.0445   | 0.96 |
| 0ADASYN        | EasyEnsemble   | 159 | 7304| 16  | 106,444 | 0.9357    | 0.0416   | 0.96 |

TP: True Positive; FP: False Positive; FN: False Negative; TN: True Negative; ROS: random over-sampling; SMOTE: Synthetic Minority Over-sampling Technique; ADASYN: Adaptive Synthetic Sampling Technique; GBM: Gradient Boosting Machine; NB: Naïve Bayes; MLP: multi-layer perceptron; RUS: random under-sampling.
When the best model is chosen, a second step integrates cost–benefit analysis for computational performance. The smallest gains in statistics could mean serious savings to the next step in the process. Stacking generalizations have a development and processing overhead that needs to be weighed between operation costs and predictive gains to justify their implementation in production. Each predictive gain may represent a future lower overhead in administrative costs associated with a fraud investigation review or an increase in fraud being successfully identified and/or prosecuted. Additional analysis of the new fraud cases found may include fraud event cost evaluation. Fraud events represented in a classification matrix should be viewed from a higher business perspective. An incident of fraud is a statistic, but the value of each fraudulent transaction can vary within a substantial range of cost and/or damage.

Even if the fraud investigator’s review does not justify using stacked generalizations for production, the cost and effort of building a test harness to evaluate a wide range of

Figure 2. Receiver operating characteristic (ROC) curve of the combined results. GBM: GradientBoostingMachine; NB: Naive Bayes; AUC: area under the curve; MLP: multi-layer perceptron; SMOTE: Synthetic Minority Over-sampling Technique.

Figure 3. Precision–recall (PR) curve of the combined results. GBM: GradientBoostingMachine; NB: Naive Bayes; MLP: multi-layer perceptron; SMOTE: Synthetic Minority Over-sampling Technique.
algorithm and sampling options to be used for research and development (R&D) for project and process improvements may be justified. This is when more formal studies on the effectiveness and efficiency of the machine learning process are conducted. The data set is reviewed to see if the necessary attributes exist to find the fraudulent transactions, finding which resampling routines and classifiers are actually achieving results supported by the fraud investigator’s conclusions, and when test strategies are explored to further improve machine learning performance. The cost and effort of R&D is weighed against yielding process improvements and updates to best practices that demonstrate reduced budgets for fraud investigator review and more successfully identified and/or prosecuted fraud cases.

5.1. Next steps

There are several areas of focus with potential areas of study to improve predictive performance. The first is a more theoretical inquiry into finding the best kNN distance measurement for sampling a fraud data set to optimize the classification process using informed resampling, such as SMOTE or its variants. Another is to add more fraudulent events in the data set from known fraud cases in the business industry in an accumulative manner rather than using only existing current fraud transactions.

This accumulative method can assist industry-wide solutions (using reverse engineering of the fraud event with case management, accounting, and other supportive fraud detection tools and team members) where the fraud typology can be outlined with possible system access points and routines, including multiple event tracing. This evidence can be placed in an overall fraud transaction framework then further divided by fraud type and industry for production. Creating a knowledge base can reuse this information, supporting businesses at an industry level rather than building solutions using only their own one-off experience. A similar framework was recently successfully accomplished for managing data analytics in internal audit in partnership with the ACFE, Internal Audit Association (IIA), American Institute of Certified Public Auditors (AICPA), and private industry. Internal audit had been overwhelmed by the volume and complexity of transactions being handled daily by businesses and agencies. Only by creating a higher-level consortium to oversee and strategize from an industry-wide perspective was a solution available. This was a monumental effort and took approximately five years. The exact same initial conditions are facing businesses and agencies responsible for identifying fraudulent events in their electronic systems.

The best practices and lessons learned from creating a framework and viable development solutions for internal audit can be leveraged to get an industry-wide handle on fraud transactions. Since internal and external audit have a due diligence requirement to build the finding and managing of fraudulent transactions into their work flow, there exists a strong possibility that the internal audit data analytics solutions have reusable solutions that already focus on fraud. These same internal audit data analytics solutions are facing businesses and agencies that need improved identification of fraud solutions. In some cases, the internal audit, fraud investigations, and machine learning data scientists are already working together. Building a formal group and procedures for them to work together on an industry-wide strategy would be required before formal funding would be approved. The low percentage of fraud being found along with the high percentage of revenue lost worldwide would justify the cost of investing in such a solution.

The internal audit data analytics initiative included financial transaction reviews. The problem space for identifying the requirements raised the work descriptions to a more abstract level, which did not include confidential customer information. This means of solving a technical problem, without triggering prohibitions on information sharing across financial institutions, set an excellent precedent to guide the planning requirements for a similar project focused on building a framework around fraud typologies as they occur in business. A byproduct of having a consortium of businesses and professional practice organizations working together is the ability to piece together the transactions broken up by crime rings for the purpose of giving them a smokescreen for their criminal enterprise. An industry-wide framework would also be able to support cross-industry, federal, and state agencies, and law enforcement collaborations handling elaborate organized crime ring schemes that are committed between financial institutions and national borders.

6. Conclusion

This study was an exploratory effort to be used as a basis for more in-depth research into discovering the effect of stacked generalization on resampled fraud imbalance data and how to ensure the best machine learning model selection. It identified three areas of focus to improve the application of machine learning methods on the handling of fraudulent transactions. The first was the opportunity to discover a mathematical formula that would best generalize the fraud data structure to be used as a preferred resampling method, which is a theoretical solution. This effort needs cleaned fraud data sets to be made available to researchers. The second was to find an effective stacked generalization formula using combined resampling methods and algorithms. One of the key issues that needs to be an integral part of any machine learning solution is that the metrics themselves do not identify fraud. The results
need to be reviewed and confirmed by fraud investigators. Therefore, findings made by fraud investigators validate the best models and/or provide feedback to data scientists to make the necessary changes to improve the process and make determinations on model selection rather than by using metrics alone. Finally, the next steps referred to an organizational collaborative effort to discover a framework to handle fraud cases by building business consortiums working together and sharing their discoveries. Such a solution is hindered by the intrinsic issue that most fraud research is conducted within businesses and not in a collaborative effort due to the confidentiality of the data; however, this perspective is changing and business consortiums are being established to work through industry solutions, although their data is not available to research groups where most of the theoretical breakthroughs in general are taking place.

Another discovery was that applying machine learning to fraudulent transactions has similar characteristics to the work being done in epidemiology. In neither case did automatically optimizing machine learning generalization necessarily find the rare events needed without further scrutiny. Naimi and Balzer noted that the complexity of stacks obscured causal relationships in the input data and output outcomes. As a result, a more thoroughly thought out understanding of their relationships needs to be the primary focus of the research rather than just a simple generalization performance improvement on the data set.

Our internal domestic fraud strategy and implementation merge into how our country works with our international partners worldwide. The improvements made domestically reflect directly how our systems and processes interact with other countries and international organizations and it is a matter of national security that they are up-to-date and at the cutting edge, working seamlessly in the hand-off across organizations.

Data accessibility statement
The creditcard.csv public data set is available at https://www.kaggle.com/ananta/credit-card-data. Figures have been uploaded to SAGE.

Funding
This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

ORCID iDs
Kathleen R Kerwin https://orcid.org/0000-0002-8065-2628
Nathaniel D Bastian https://orcid.org/0000-0001-9957-2778

Notes
1. See the Law Dictionary https://thelawdictionary.org/fraud/
2. Typology is defined as classification according to general type.
3. Distance measurements include Euclidean, Mahalanobis, Haversine, Hamming, Canberra, Bray-Curtis, Jaccard, Matching, Dice, Kulinski, RogersTanimoto, RussellRao, SokalMichener, and SokalSneath, Hellinger Distance.

References
1. ACFE. Professional training promoting the development and implementation of sound anti-money laundering policies and procedures, https://www.acfe.com/report-to-the-nations/2018 (2018, accessed March–May 2019).
2. Mills C. Predictive analytics in fraud and AML. J Fin Complian 2017; 1: 17–26.
3. Swecker C and Brothers B. Crime rings have grown to a prevalent and formidable threat, association of certified anti-money launderer specialists (ACAMS) webinar training, https://lmsacams.becker.com/media/813985 (2015, accessed March–May 2019).
4. Swecker C and Brothers B. AML: a new approach to fight complex money laundering schemes, association of certified anti-money launderer specialists (ACAMS) webinar training, https://lmsacams.becker.com/media/816463 (2015, accessed March–May 2019).
5. The Clearing House (At the Center of Banking Since 1853). A new paradigm: redesigning the U.S. AML/CFT framework to protect national security and aid law enforcement, https://www.theclearinghouse.org/~media/TCH/Documents/TCH% 20WEEKLY/2017/20170216_TCH_Report_AML_CFT_ Framework_Re design.pdf (2017, accessed March–May 2019).
6. Tropina T. Cyber crime and organized crime, http://Gmagazine.unicri.it/?p=310 (2019, accessed March–May 2019).
7. VSCPA. Heads up: cybercriminals are businesspeople, https://www.vscp a.com/article/heads-cybercriminals-are-businesspeople (2019, accessed March–May 2019).
8. Europol S. Serious and organized crime threat assessment, identifying the priorities in the fight against major crime, https://www.europol.europa.eu/socta-report (2017, accessed March–May 2019).
9. UNODC. Money-laundering globalization, http://www.unodc.org/unodc/en/money-laundering/globalization.html (2019, accessed March–May 2019).
10. GSDHC. Organised crime, violence and development, https://gsdrc.org/wp-content/uploads/2016/08/Org_crime_ violence_dev.pdf (2016, accessed March–May 2019).
11. FATF. Financial action task force, http://www.fatf-gafi.org/home/ (1989, accessed March–May 2019).
12. FIU. Financial investigatory unit, https://egmontgroup.org/ (1995, accessed March–May 2019).
13. ACAMS. https://www.acams.org/en (2019, accessed March–May 2019).
14. Yusof R, Kasmiran KA, Mustapha A, et al. Techniques for handling imbalanced datasets when producing classifier models. J Theor Appl Inform Technol 2017; 95: 1425–1440.
15. Japkowicz N. Learning from imbalanced data sets: a comparison of various strategies. *AAAI workshop on learning from imbalanced data sets* 2000; 68: 10–15.
16. Kotsiantis S, Kanellopoulos D, Pintelas P, et al. Handling imbalanced data sets: a review. *GESTS Int Trans Comput Sci Eng* 2006; 30: 25–36.
17. Chawla NV. Data mining for imbalanced datasets: an overview. In: Maimon O and Rokach L (eds) *Data mining and knowledge discovery handbook*. Boston, MA: Springer, 2005, pp 853–867.
18. Friedman JH, Kohavi R and Yun Y. Lazy decision trees. In: *AAAI/IAAI*, vol. 1. 4 August 1996, pp.717–724. Cambridge, MA: AAAI Press/MIT Press.
19. Maalouf M and Trafalis TB. Rare events and imbalanced datasets: an overview. *Int J Data Mining Model Manag* 2011; 3: 375–388.
20. Chen Z, Teoh EN, Nazir A, et al. Machine learning techniques for anti-money laundering (AML) solutions in suspicious transaction detection: a review. *Knowl Inform Syst* 2018; 57: 245–285.
21. Weiss GM and Provost F. Learning when training data are costly: the effect of class distribution on tree induction. *J Artif Intell Res* 2003; 19: 315–354.
22. Chawla NV. C4. 5 and imbalanced data sets: investigating the effect of sampling method, probabilistic estimate, and decision tree structure. In: *proceedings of the ICML*, vol. 3, 2003, p.66.
23. Chawla NV, Bowyer KW, Hall LO, et al. Smote: synthetic minority over-sampling technique. *J Artif Intell Res* 2002; 16: 321–357.
24. Ling CX and Li C. Data mining for direct marketing: problems and solutions. In: *Kdd*, volume 98. pp.73–79.
25. Japkowicz N. The class imbalance problem: significance and strategies. In: *proceedings of the international conference on artificial intelligence (IC-AI 2000)*, Las Vegas, NV, 26 June 2000.
26. McCarthy K, Zabar B and Weiss G. Does cost-sensitive learning beat sampling for classifying rare classes? In: *proceedings of the 1st international workshop on utility-based data mining*, pp.69–77. New York: Association for Computing Machinery.
27. Weiss GM, McCarthy K and Zabar B. Cost-sensitive learning vs. Sampling: which is best for handling unbalanced classes with unequal error costs? *Dmin* 2007; 7: 24.
28. Rajeswari K. Feature selection by mining optimized association rules based on a priori algorithm. *Int J Comput Appl* 2015; 119.
29. Zheng Z, Wu X and Srihari R. Feature selection for text categorization on imbalanced data. *ACM Sigkdd Explorations Newsletter* 2004; 6: 80–89.
30. Wolpert DH. Stacked generalization. *Neural Network* 1992; 5: 241–259.
31. Ting KM and Witten IH. Issues in stacked generalization. *J Artif Intell Res* 1999; 10: 271–289.
32. Zhang Y and Wang D. A cost-sensitive ensemble method for class-imbalanced datasets. In: *Abstract and applied analysis, volume* 2013. Hindawi.
33. Sesmero MP, Ledezma AI and Sanchis A. Generating ensembles of heterogeneous classifiers using stacked generalization. *Wiley Interdisc Rev Data Mining Knowl Discov* 2015; 5: 21–34.
34. Naimi A and Balzer L. Stacked generalization: an introduction to super learning. *Eur J Epidemiol* 2018; 33: 459–464.
35. Wong ML, Seng K and Wong PK. Cost-sensitive ensemble of stacked denoising autoencoders for class imbalance problems in business domain. *Expert Syst Appl* 2020; 141.
36. Winston P and Ortiz L. 6.034f boosting notes, 2009. Patrick Winston and Luis Ortiz Draft of November 17, 2009, MIT - Massachusetts Institute of Technology.
37. Wang J, You J, Li Q, et al. Extract minimum positive and maximum negative features for imbalanced binary classification. *Patt Recognit* 2012; 45: 1136–1145.
38. Ting KM and Witten IH. Stacked generalization: when does it work? In: *international joint conferences on artificial intelligence*, University of Waik, 1997.
39. Bontempi G and Lemaerts T. Machine Learning Group - ULB Machine Learning Group (MLG), a research unit of the Computer Science Department of the Faculty of Sciences. Creditcard.csv dataset, https://www.kaggle.com/mlg-ulg (2004, accessed March–May 2019).
40. Hu LY, Huang MW, Ke SW, et al. The distance function effect on k-nearest neighbor classification for medical data-sets. *SpringerPlus* 2016; 5: 1–9.
41. Winston P.MIT Open Courseware. Artificial intelligence course lecture 10:Introduction to learning, nearest neighbors, https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-034-artificial-intelligence-fall-2010/lecture-videos/lecture-10-introduction-to-learning-nearest-neighbors/ (2010, accessed March–May 2019).
42. Tomek I. Two modifications of CNN. *IEEE Trans Syst Man Cybern* 1976; 6: 769–772.
43. Han H, Wang WY and Mao BH. Borderline-smote: a new over-sampling method in imbalanced data sets learning. In: *international conference on intelligent computing*, pp.878–887. Springer.
44. He H, Bai Y, Garcia EA, et al. ADASYN: adaptive synthetic sampling approach for imbalanced learning. In: *2008 IEEE international joint conference on neural networks (IEEE world congress on computational intelligence)*, 2008, pp.1322–1328. IEEE.
45. Quinlan JR. Decision trees as probabilistic classifiers. In: *proceedings of the fourth international workshop on machine learning*, 1987, pp.31–37. Elsevier.
46. Sharma S, Agrawal J and Sharma S. Classification through machine learning technique: C4. 5 algorithm based on various entropies. *Int J Comput Appl* 2013; 82.
47. James G, Witten D, Hastie T, et al. *An introduction to statistical learning, volume* 112. New York: Springer, 2013.
48. Ng A. Deep Learning School on September 24/25, 2016, Video file. *Nuts and bolts of applying deep learning*, https://www.youtube.com/watch?v=F1ka6a13S9t (2016, accessed March–May 2019).
49. Turney PD. Types of cost in inductive concept learning. *arXiv preprint cs/0212034* 2002.
50. Shilbayeh SA, et al. Cost sensitive meta-learning. PhD Thesis, University of Salford, 2015.
51. Hand DJ, Whitrow C, Adams NM, et al. Performance criteria for plastic card fraud detection tools. J Oper Res Soc 2008; 59: 956–962.
52. Bahnsen AC, Villegas S, Aouada D, et al. Fraud detection by stacking cost-sensitive decision trees. Data Science for Cyber-Security. World Scientific, 2017.
53. Lomax S and Vadera S. A survey of cost-sensitive decision tree induction algorithms. ACM Comput Surv 2013; 45: 1–35.
54. Shannon CE. A mathematical theory of communication. Bell Syst Techn J 1948; 27: 379–423.
55. Lomax SE, et al. Cost-sensitive decision tree learning using a multi-armed bandit framework. PhD Thesis, University of Salford, 2013.
56. Efron B. Computers and the theory of statistics: thinking the unthinkable. SIAM Rev 1979; 21: 460–480.
57. Kraus A. Recent methods from statistics and machine learning for credit scoring. PhD Thesis, LMU, 2014.
58. RAY S. 5 easy questions on ensemble modeling everyone should know, https://www.analyticsvidhya.com/blog/2015/09/questions-ensemble-modeling/ (2015, accessed March–May 2019).
59. Yin QY, Zhang JS, Zhang CX, et al. An empirical study on the performance of cost-sensitive boosting algorithms with different levels of class imbalance. Math Prob Eng 2013; 2013.
60. Natekin A and Knoll A. Gradient boosting machines, a tutorial. Frontier Neurorobot 2013; 7: 21.
61. Liu XY, Wu J and Zhou ZH. Exploratory undersampling for class-imbalance learning. IEEE Trans Syst Man Cybern B 2008; 39: 539–550.
62. Seiffert C, Khoshgoftaar TM, Van Hulse J, et al. Rusboost: a hybrid approach to alleviating class imbalance. IEEE Trans Syst Man Cybern A Syst Hum 2009; 40: 185–197.
63. Yan J and Han S. Classifying imbalanced data sets by a novel re-sample and cost-sensitive stacked generalization method. Math Prob Eng 2018; 2018.
64. Fawcett T and Provost F. Adaptive fraud detection. Data Mining Knowl Discov 1997; 1: 291–316.
65. Fawcett T. An introduction to roc analysis. Patt Recognit Lett 2006; 27: 861–874.
66. Author U. When is precision more important over recall?, https://datascience.stackexchange.com/questions/30881/when-is-precision-more-important-over-recall/31085#31085 (2018, accessed March–May 2019).
67. Ng A. Lecture 11.4 – machine learning system design – trading off precision and recall, https://www.youtube.com/watch?v=W5meQnGACGo (2016, accessed March–May 2019).
68. He H and Garcia EA. Learning from imbalanced data. IEEE Trans Knowl Data Eng 2009; 21: 1263–1284.
69. Joshi MV, Agarwal RC and Kumar V. Predicting rare classes: can boosting make any weak learner strong? In: proceedings of the eighth ACM SIGKDD international conference on knowledge discovery and data mining, 2002, pp.297–306.

Author biographies

Kathleen R Kerwin has had a diverse background in law enforcement, as well as large application and system software development and management working for a top tiered bank and two Big 5 Consulting firms. She is a Certified Fraud Examiner, as well as a Certified AntiMoney Laundering Specialist. She holds a MS degree in data science from Northwestern University, where she completed thesis research in predictive modeling for credit risk scoring.

Nathaniel D Bastian is a decision analytics professional whose expertise lies in the scientific discovery and translation of actionable insights into effective decisions using algorithms, techniques, tools and technologies from operations research, data science, artificial intelligence, systems engineering, and economics to design, develop, deploy and operationalize intelligent decision-support systems and models for descriptive, predictive and prescriptive analytics. He holds a PhD degree in industrial engineering and operations research from the Pennsylvania State University, MEng degree in industrial engineering from Penn State, MS degree in econometrics and operations research from Maastricht University, and BS degree in engineering management (electrical engineering) with honors from the U.S. Military Academy at West Point. His primary research interest is computational stochastic optimization and learning for making inferences and decisions under uncertainty.