Received: 15 May 2020 | Revised: 18 May 2021 | Accepted: 26 June 2021

DOI: 10.1111/jori.12359

ORIGINAL ARTICLE

Journal of Risk and Insurance

Insurance fraud detection with unsupervised deep learning

Chamal Gomes[1] | **Zhuo Jin**[1] | **Hailiang Yang**[2]

[1]Centre for Actuarial Studies, Department of Economics, The University of Melbourne, Parkville, Victoria, Australia

[2]Department of Statistics and Actuarial Science, The University of Hong Kong, Pokfulam, Hong Kong

Correspondence
Zhuo Jin, Centre for Actuarial Studies, Department of Economics, The University of Melbourne, Parkville, VIC 3010, Australia.
Email: zjin@unimelb.edu.au

Funding information
Research Grants Council, University Grants Committee,
Grant/Award Number: 17304921

Abstract

The objective of this paper is to propose a novel deep learning methodology to gain pragmatic insights into the behavior of an insured person using unsupervised variable importance. It lays the groundwork for understanding how insights can be gained into the fraudulent behavior of an insured person with minimum effort. Starting with a preliminary investigation of the limitations of the existing fraud detection models, we propose a new variable importance methodology incorporated with two prominent unsupervised deep learning models, namely, the autoencoder and the variational autoencoder. Each model's dynamics is discussed to inform the reader on how models can be adapted for fraud detection and how results can be perceived appropriately. Both qualitative and quantitative performance evaluations are conducted, although a greater emphasis is placed on qualitative evaluation. To broaden the scope of reference of fraud detection setting, various metrics are used in the qualitative evaluation.

KEYWORDS

autoencoder, insurance fraud detection, unsupervised deep learning, variable importance, variational autoencoder

1 | INTRODUCTION

Insurance fraud is defined by Gill et al. (2005) as "knowingly making a fictitious claim, inflating a claim or adding extra items to a claim, or being in any way dishonest with the intention of gaining more than legitimate entitlement." This definition applies to insurance fraud committed by fraudulent policyholders. Our paper will focus on the analysis of fraud behavior as such. Although the types of insurance fraud can be categorized by the nature of insurance product, in a broader context, the Insurance Council of Australia (ICA) categorizes insurance fraud as opportunistic, premeditated, or fraudulent nondisclosure, either of which ultimately leads to illegitimate financial gain for the fraudulent claimant. Present discourse involving fraud classification on moral grounds may take different positions, which is not pertinent to the modeling of fraud (Derrig, 2002). Thus, for axiomatic reasons, this paper focuses on the broader notion of insurance fraud in accordance with ICA, without such further classification.

Traditional insurance fraud detection heavily relied on adept auditing and inspection (Artís et al., 2002; Dionne et al., 2009; Nian et al., 2016). However, because of technological developments and large-scale business operations, the adoption of such conventional methodologies makes the task of fraud detection impractical (Kemp, 2010). The modus operandi of criminals, which is ever so dynamic, has made it improbable, if not impossible, to identify fraudulent claims based on a fixed algorithmic criterion most of the times (Nguyen & Perez, 2020). In his paper on a holistic approach to fraud management, Wilhelm (2004) discussed this dynamic behavior of criminals in general to explain why fraud management life cycle should be more dynamic and evolving. To keep up with the rapid macroenvironmental changes, the insurance industry had to integrate automation of expeditious statistical models into their internal fraud detection systems. A study by the Statistical Analysis System found that in the United States, 75% of all insurers had integrated automated systems for fraud detection by 2016.

Sharma and Panigrahi (2013) categorized the key domains in which these models operate as follows: classification (Phua et al., 2010), clustering (Jain et al., 1999), regression (Fawcett & Provost, 1997), visualization based on trees, and anomaly-based (Noble & Cook, 2003). A list of fraud modeling methodologies is summarized in Table 1. An extraneous observation is the

TABLE 1 Fraud modeling methodologies

Model class	Modeling methodologies
Clustering	K-means clustering
	Nearest neighbors
Classification	Random forest
	Naive Bayes
	Support vector machines (SVM)
Regression	Logistic regression
	Polynomial regression
	Neural networks
Outlier detection	Isolation forest
	Gaussian mixture models
Prediction	Neural networks

possibility that under such a categorization, certain models can share several domains, for example, tree-based visualization approaches against classification and regression domains. Various statistical models have been applied to insurance sectors. See Li et al. (2008) for a survey on statistical methods for health care fraud detection. Despite varying model categories and the range of statistical models studied in fraud modeling literature, effectiveness in real-life business settings has been the major limitation owing to various reasons, for example, class imbalance of the data distribution and the inexistence of true outcomes in fraud contextualization, which render modeling attempts through conventional statistical frameworks difficult. In this data mining era with abundant availability of data, the use of machine learning algorithms is a propitious alternative.

Machine learning algorithms used for fraud detection can be primarily divided into the following categories: supervised (Khatri et al., 2020), unsupervised (Srivastava & Salakhutdinov, 2014), and semisupervised models (Van Engelen & Hoos, 2020). The significance of such a distinction is best understood by analyzing the elemental fundamentals of the two initial model classes. Supervised learning is defined as attempting to infer a function that best maps the given input data to a given output. A set of training data is submitted as system input during the model training phase. Each input is labeled with a desired output value, which essentially supervises the model; therefore, the model knows what the output is when input is fed into it. A review of fraud detection literature revealed that the most common and pertinent studied class of models are supervised machine learning models. A recent study of antifraud methodologies used for health care fraud detection by the Society of Actuaries found that almost all of the methodologies used are supervised models. Supervised learning paradigms have been well refined over the years through extensive academic research with significant model developments. Comprehending the analytical framework of the supervised models is imperative not only to evaluate performance against deep learning models, but also to understand the classification dynamics of supervised models with the expectation of better distinguishing model results for greater insights. However, such a discussion is beyond the scope of this paper.

To develop a supervised model for fraud detection, we must first obtain past information on the true occurrence of fraud and nonfraud. Based on the accuracy of the output labels, the model attempts to optimize the parameters to better identify fraud and nonfraud instances. In contrast, unsupervised learning comprises a class of analytical methods that attempt to infer a function that best characterizes the representation of the input data in the absence of any supervision. Unsupervised models based on neural networks with multiple "hidden" layers, also known as multilayer perceptrons (MLP), are often referred to as unsupervised deep learning (Goodfellow et al., 2016; LeCun et al., 2015). Hastie et al. (2009) described unsupervised learning to be more subjective than supervised learning, as there is no simple goal for analysis, such as predicting a response. Thus, it is able to better determine a feature representation of the data in the absence of any bias or misguidance precipitated by subjective output data. An unsupervised learning paradigm plays an important role in fraud detection within the domain of anomaly detection. Unlike supervised models, we do not require prior knowledge of the output labels to develop a model (Niu et al., 2019). See Chalapathy and Chawla (2019) for a review of deep learning methods for anomaly detection.

It is clear from the very definition that supervised modeling is proficient at modeling relationships in which the output label or variable of interest is naturally generated without the need for self-verification. An example is modeling stock prices based on historical book ratios and stock statistics. The variable of interest in this case, the stock price, is naturally present. However, when attempting to monitor insurance fraud, the predictive variable of interest is not

present in natural business settings. It requires manual verification on a case-by-case basis. Moreover, insurance fraud is a complex phenomenon whose outcomes are almost impossible to verify with absolute certainty, despite thorough investigations involving considerable time and effort. Thus, it is subject to errors of commission, omission, and misclassification, which can deter model performance (Artís et al., 2002). We should remember that insurers operate in an era where the amalgam of technology and modus operandi of criminals is ever so dynamic, such that one set algorithm cannot be used to detect fraud. The difficulty of learning these dynamic behavioral characteristics increases exponentially in terms of complexity, and supervised learning cannot, in practice, learn such complex latent variables in the data. These limitations have hindered the use of supervised modeling in insurance fraud detection, thereby allowing opportunity for an alternate and versatile modeling paradigm, which is unsupervised deep learning. There is emerging literature on developing unsupervised learning methods in insurance fraud detections. See Ekin et al. (2018, 2019) and Zafari and Ekin (2019).

The objective of the paper is not to create an exposé, nor to censure supervised models, but to scrutinize the profound applications of unsupervised deep learning models and to propose a novel variable importance methodology. Instead of mere predictions, our model aims to offer pragmatic insights into the driving factors of fraud. Many practitioners of fraud detection have often been interested in further comprehending which features constitute fraudulent characteristics (Artís et al., 1999). Existing supervised models, such as random forests, provide solutions to this question; however, owing to the inherent limitations of supervised models in insurance fraud detection, their use in this field is impractical.

The variable importance methodology proposed in this paper strictly abides by the fundamentals of unsupervised learning, as it does not utilize target variables. The proposed methodology can be applied in the fields of superannuation, insurance, investment and in other broader domains of the insurance industry with effortless extensions. For example, it can be used to understand changes in policyholder behavior, climate change impacts, and actuarial pricing behavioral analysis. Moreover, understanding the drivers of fraud through idle data is not only cost effective in the business context but also realistic. The proposed methodology leverages the ability of unsupervised deep learning models to continuously learn complex changes in user behavior through periodical model updates conducted at the click of a button.

Furthermore, it is incontrovertible that input and output data should be labeled to provide a learning basis for the supervised model to conduct any form of variable importance analysis. From determining the structure of parameterization to parameter optimization, the guidance of labels, which do not exist in natural business settings, is required. As mentioned above, labels should be manually generated, which is a costly, time consuming, and possibly unreliable exercise. Moreover, there may be implicit bias in the output labels owing to human intervention or the procedure being used. In certain events, such as workers' compensation, there is no alternative to produce true outcomes. Proactive decision making will require that information is learned in a timely manner to preserve the applicability and value of the information. In this case, by the time interminable surveys and studies of fraud behavior occur, criminals will have already changed their approach to a different domain of committing fraud, which the user has not identified. Thus, for large-scale insurers who deal with customer requests occurring at an astronomical rate, spending time and resources on these timely updates offers no return on their investment. The proposed unsupervised variable importance methodology aims to overcome such limitations within fraud detection.

This study includes a juxtaposition of the variable importance methodology on two prominent unsupervised models, namely, variational autoencoder (VAE) and autoencoder (AE)

(Baldi, 2012), on qualitative grounds. AE is a type of neural architecture tasked with encoding the input data and reconstructing the original input, with minimal error (Zheng & Peng, 2018). Meanwhile, VAE is a probabilistic variant of the AE; its objective is similar to that of AE despite the probabilistic dynamics (Arenz et al., 2020). The proposed variable importance methodology uses the imbalanced nature of fraud data as its foundation to function in the domain of anomaly/outlier detection based on the categorization offered by Sharma and Panigrahi (2013). The proposed methodology is built upon the AE architecture; hence, our discussion will begin with an introduction to AE and VAE and their model dynamics. We will then discuss several approaches of how unsupervised deep learning incorporating AE and VAE can be used in fraud classification. Variable importance methodology is then introduced, detailing the steps involved and how results should be understood. To evaluate the performance of the proposed approach, three data sets, namely, $\mathcal{D}_1$, $\mathcal{D}_2$, and $\mathcal{D}_3$, are used. Each of the data sets has specific features that capture the real insurance data.

As explained in the following sections, the ability of unsupervised deep learning models to infer complexities of and dynamic changes in criminal behavior is paramount. These models base the data representations on a probabilistic distribution to not only determine instances of fraud but also identify the drivers of fraud, which may lead to the elimination of policy loopholes and flaws. Based on the proposed methodologies, we can expand our insights into identifying drivers of fraud from a multitude of features within the AE framework and its variations. Compared with existing approaches, the proposed variable importance AE/VAE approaches outperform supervised learning in the precision of fraud detection in $\mathcal{D}_2$ in which data are all output labeled. In the $\mathcal{D}_3$ data in which the majority of data are not output labeled and supervised learning models have limited application, the proposed variable importance AE/VAE approaches provide excellent identification of key fraud drivers.

The rest of the paper is organized as follows. In Section 2, three data sets that are used for training and testing in the following sections are described. In Section 3, two primary deep learning models and their underlying dynamics, namely, VAE and AE are introduced. In Section 4, the criteria of fraud classification are presented. In Section 5, the proposed methodology for variable importance is presented and used on the three data sets. Performance of predictive results is shown in Section 6. Finally, we conclude the paper with further remarks in Section 7.

2 | DATA

The testing of the models related to this study was based on three primary data sets. One data set is directly related to insurance fraud detection; it will be denoted as $\mathcal{D}_1$ for convenience. The $\mathcal{D}_1$ data set, initially released by Oracle, is related to auto insurance claims made by 15,478 policy holders. The data set contains 33 features and one column indicating the true outcome of the claim (fraud or not fraud). 32 of the features were categorical variables (gender of the claimant, car manufacturer, marital status, accident area, etc.). Data cleansing began by integer encoding (Helmy et al., 2018) the categorical variables to normalize the values. After normalizing the encoded variables, they were fed into the model. This is an important step, because any model, be it supervised or unsupervised, requires its categorical variables to be quantitatively encoded before being fed into a model. Although normalizing the categorical variables is not compulsory, it is recommended for a variety of practical reasons, including faster training time and minimal likelihood of being stuck in a local optima. The total number of fraud occurrences in the $\mathcal{D}_1$ data set was 218 relative to 14,900 nonfraud occurrences.

This reflects the true nature of fraudulent data, which usually indicates a high class imbalance data structure (Table 2).

The second data set, denoted $\mathcal{D}_2$ data set, contains credit card transactions made by European cardholders in September 2013. The data set was collected and analyzed through a research collaboration between Worldline and the Machine Learning Group of Université Libre de Bruxelles (ULB) on big data mining and fraud detection. The number of transactions in the data was 284,807, whereas the number of features was 29 including the fraud amount. All of the features (except for fraud amount) in the $\mathcal{D}_2$ data set were anonymized because of their confidential nature, as it was acquired from a realistic business process. Features were anonymized through principal component analysis transformation, and the only raw attributes were the true outcome of the transaction, the amount of fraud, and time of fraud. Overall, the data set consisted of 28 principal components (denoted V1–V28), fraud amount, time, and the fraud indicator. Hence, feature encoding, as was performed with $\mathcal{D}_1$ data set, was not required.

The $\mathcal{D}_2$ and $\mathcal{D}_1$ data sets were chosen for many reasons. First, we aimed to reduce the dependence on model inferences and research findings made from a single data set. Second, the $\mathcal{D}_2$ data set was of a higher quality than the $\mathcal{D}_1$ data set in relation to the objective of this paper. This was particularly appealing because unsupervised deep learning models require a profusion of data for model training (optimization) compared with supervised models. Relative to supervised models, deep learning models contain a large number of parameters that need to be optimized; hence, the availability of considerable training data is crucial for model performance. The $\mathcal{D}_1$ data set only contains 15,118 records, whereas the $\mathcal{D}_2$ data set contains 284,807 records in total.

Third, within the domain of fraud detection, the two data sets shared the same cardinality, which is trying to identify fraud cases given a series of features relating to each instance. However, the $\mathcal{D}_2$ data set reflected a more realistic overview of the imbalanced nature of fraud detection, with a greater fraud to nonfraud ratio (imbalance ratio); it contains 284,315 nonfraud instances for 492 fraud instances. Meanwhile, the $\mathcal{D}_1$ data set has an imbalance ratio of 218 fraud instances for 14,900 nonfraud instances. Therefore, the results obtained from the $\mathcal{D}_2$ data set have been used to illustrate many concepts and insights obtained throughout this paper.

Having understood the context of the data, we can assume that for each data set consisting of N data vectors of input dimension m (either $\mathcal{D}_1$ or $\mathcal{D}_2$),

$$\mathcal{D} = \{\mathbf{x}^1, \mathbf{x}^2, ..., \mathbf{x}^N\} \equiv \{\mathbf{x}^i\}_{i=1}^N, \quad \text{where} \quad \mathbf{x}^i \in \mathbb{R}^m \equiv \left\{\mathbf{x}_j^i\right\}_{j=1}^m.$$

$\{\mathbf{x}^i\}_{i=1}^N$ is assumed to be i.i.d. This is to say that each input vector in $\mathcal{D}_1/\mathcal{D}_2$ data set is independent (none of the claims/transactions are related to each other) and none are identical (all input vectors are distinct).

TABLE 2 $\mathcal{D}_1$ data set features

Month	WeekOfMonth	DayOfWeek	Make	AccidentArea
DayOfWeekClaim	MonthOfClaim	WeekOfMonthClaim	Sex	MaritalStatus
Age	Fault	AddressChangeClaim	VehicleCategory	VehiclePrice
FraudFound	PolicyNumber	RepNumber	Deductible	DriverRating
DaysPolicyAccident	DaysPolicyClaim	PastNumberOfClaims	AgeOfVehicle	BasePolicy
PoliceReportFiled	WitnessPresent	NumberOfSuppliments	AgentType	Year
NumberOfCars	PolicyType	AgeOfPolicyHolder		

The third data set, denoted $\mathcal{D}_3$ data set, contains insurance claims data provided by a leading insurance company in Spain for the period 2015–2016. Owing to data sensitivity, as was the case with $\mathcal{D}_2$ data set, features were anonymized. The data set was obtained through Mendeley Data, and it contains 272,858 data points, of which 2379 represent cases investigated and found to be fraud. Unlike the $\mathcal{D}_2$ data set, the rest of the 270,479 data points represent unanalyzed cases whose outcomes are unknown. This makes this data set less attractive than the $\mathcal{D}_2$ data set. Palacio (2018) discussed the original nature of the data in detail, and most of the data was grouped in personally identifiable information, policy attributes, and so on.

3 | MODELS AND METHODOLOGY

This section analyzes the two primary deep learning models, namely, VAE and AE, and their underlying dynamics. It includes a brief discussion of each of the models with a view of introducing the reader to MLPs.

3.1 | Autoencoder

AEs are simple learning circuits that are tasked with transforming inputs into outputs, with minimal distortion. An AE consists of an encoder function f and a decoder function g. The purpose of the encoder function f is to compress the input to a lower representation using the most salient features of the input data. Meanwhile, the decoder function g reconstructs the input from the compressed features. Thus, the entire functionality of an AE for input data $\mathbf{x}$ can be presented as follows, where $g(f(\mathbf{x})) = r(\mathbf{x})$ denotes the reconstructed input. The lower dimension of the encoder function characterizes the undercomplete AE framework; many other frameworks such as sparse AEs and denoising AEs exist, which, however, are beyond the scope of this paper.

The neural dynamics of an AE are similar to those of a typical neural network; however, there are unique characterizations with respect to the architecture of the neural network. The encoding/decoding layer consists of symmetric hidden units in each layer, following the symmetric architecture, which, however, is not an "essential" characterization of an AE. $(h_1, ..., h_k)$ represent hidden nodes of encoding layer, $(\widehat{h}_1, ..., \widehat{h}_l)$ represent hidden nodes of information bottleneck layer, whereas $(\tilde{h}_1, ..., \tilde{h}_k)$ denote the hidden nodes of the information decoding layer. The encoded/decoded data at the jth hidden layer, the weight matrix belonging to the jth hidden layer, and the bias matrix at the jth layer are denoted as $\mathbf{z}^{(j)}$, W_j, and b_j, respectively. Then, the operation during the jth hidden layer can be denoted as follows:

$$\mathbf{z}^{(j)} = \tilde{f}^j(W_j\mathbf{z}^{(j-1)} + b_j), \tag{1}$$

where $\tilde{f}^j$ denotes the selected activation function for the jth hidden layer. The ReLu and tanh activation functions are defined as follows:

$$\tanh(y) = \frac{2}{1 + e^{-2y}} - 1, \tag{2}$$

$$\mathrm{ReLu}(y) = \begin{cases} 0 & \text{for} \quad y < 0, \\ y & \text{for} \quad u \geqslant 0. \end{cases} \tag{3}$$

Because the AE depicted in Figure 1 comprises four hidden layers (including the output layer), following the analogy from Equation (1), the operations are as follows:

$$\begin{aligned}\mathbf{z}^{(1)} &= \tanh(W_1\mathbf{x} + b_1),\\ \mathbf{z}^{(2)} &= \tanh(W_2\mathbf{z}^{(1)} + b_2),\\ \mathbf{z}^{(3)} &= \tanh(W_3\mathbf{z}^{(2)} + b_3),\\ \hat{\mathbf{x}} &= \mathrm{ReLu}(W_4\mathbf{z}^{(3)} + b_4).\end{aligned} \tag{4}$$

In this particular instance, the tanh activation function was used in the encoding and decoding layers, whereas the ReLu activation was used in the output layer. The choice of activation functions varies by each situation and is quite ambiguous. In Equation (4) above, $\hat{\mathbf{x}}$ denotes the reconstruction layer, which represents the reconstructed input in the final output layer (4th hidden layer). The dimensions of the respective weight and bias matrices were $W^{(1)} \in \mathbb{R}^{k\times m}$, $W^{(2)} \in \mathbb{R}^{l\times k}$, $W^{(3)} \in \mathbb{R}^{k\times l}$, $W^{(4)} \in \mathbb{R}^{m\times k}$, $b^{(1)} \in \mathbb{R}^{k\times 1}$, $b^{(2)} \in \mathbb{R}^{l\times 1}$, $b^{(3)} \in \mathbb{R}^{k\times 1}$, and $b^{(4)} \in \mathbb{R}^{m\times 1}$. The edges between any two layers represent the in-degree of any node in the rightmost layer cells corresponding to the dimension of the leftmost layer, thus conforming with the matrix dimensions above.

An important characterization of an AE is that the input and reconstruction layer follow the same dimensionality, reiterating the objective of an AE to reconstruct the input, which requires similar dimensions. The latent compressed layer, also known as an information bottleneck, is of

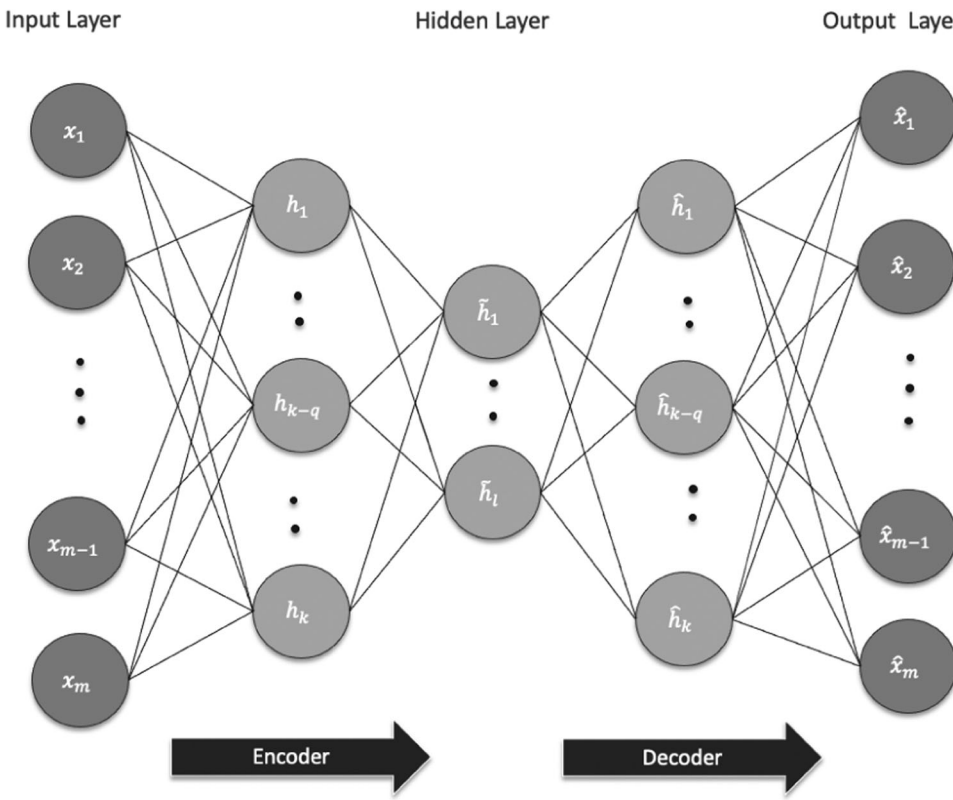

FIGURE 1 Autoencoder architecture

particular interest; it represents the salient features of the input feature space at a much-compressed dimensional space. It is from this feature space that the decoder attempts to reconstruct the input space. Therefore, the dimension of the information bottleneck determines how much freedom is given to the encoder to learn important features. The loss function used for training the AE was the Mean-Square-Error Equation (5); the Adam optimizer (Kingma & Ba, 2014) was used for stochastic gradient optimization based on the Mean-Square-Error loss.

$$J(W_1, b_1, W_2, b_2, W_3, b_3, W_4, b_4) = \min \|\mathbf{x} - g(f(\mathbf{x}))\|_2 = \sum_{i=1}^{N} (\hat{\mathbf{x}}^i - \mathbf{x}^i)^2. \tag{5}$$

In the context of anomaly detection, an AE is first trained on a training data set using the back propagation algorithm (LeCun et al., 1988). Next, the AE is evaluated on a test set consisting of the fraud and nonfraud cases to analyze the reconstruction error. Analysis will show that the corresponding reconstruction error of fraud differs from that of nonfraud cases, because the AE was calibrated to reconstruct nonfraud cases with minimal reconstruction error. Following careful consideration of the insurer's position, a reconstruction error threshold is set to flag fraud and nonfraud cases based on the reconstruction error. Typically, the threshold is set to reduce the false-positive rate; however, this may differ based on the type of the insurer and their costing structure. The threshold setting process is discussed in detail in the following sections.

3.2 | Variational autoencoder

VAE is a type of probabilistic model, where by the encoder is regularized to extract key properties from training data to generate meaningful samples. Neural networks are designed to approximate the posterior distribution for latent variables and to find optimal variational lower bound. The analogy of AE is derived through the method by which a nonprobalistic model is structured to achieve the objective (inference and generative networks). A general VAE framework with multivariate Gaussian distributed latent variable is presented in details as follows. Readers who are interested in practical applications of the proposed approach may safely skip the technical parts to the next section.

VAE is a type of generative learning model that performs Bayesian inferences for modeling the underlying probability distribution of the training data $p_\theta(\mathbf{x})$ parameterized by θ. Based on the latent variable $\mathbf{z}$, the probability distribution of the data can be defined as follows:

$$p_\theta(\mathbf{x}) = \int p_\theta(\mathbf{x}|\mathbf{z}) p_\theta(\mathbf{z}) d\mathbf{z}. \tag{6}$$

Approximating the above marginal likelihood by means of expectation maximizing or mean-field variational bound algorithm is intractable for reasons discussed by Kingma and Welling (2013). The notion of VAE is to infer $p_\theta(\mathbf{z})$ from $p_\theta(\mathbf{z}|\mathbf{x})$. However, we first have to infer the distribution $p_\theta(\mathbf{z}|\mathbf{x})$, which is inferred using variational inference. That is, we approximate $p_\theta(\mathbf{z}|\mathbf{x})$ with a variational posterior $q_\phi(\mathbf{z}|\mathbf{x})$, which is of a simpler distribution. The divergence between the variational posterior and the true posterior is measured in terms of the reverse Kullback–Leibler (KL) divergence, which measures the information loss when attempting to approximate p using q. The reverse KL divergence is defined as follows:

$$\mathbb{D}[q_\phi(\mathbf{z}|\mathbf{x}) \| p_\theta(\mathbf{z}|\mathbf{x})] = \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}[\log q_\phi(\mathbf{z}|\mathbf{x}) - \log p_\theta(\mathbf{z}|\mathbf{x})]. \tag{7}$$

Owing to the asymmetry of the KL divergence, it is important to discuss the difference between reverse and forward KL divergence, and why within the VAE framework, reverse KL divergence ($\mathbb{D}[p_\theta(\mathbf{z}|\mathbf{x}) \| q_\phi(\mathbf{z}|\mathbf{x})]$) is used. Forward KL divergence contributes loss when

$p_\theta(\mathbf{z}|\mathbf{x}) > 0, \forall\ \mathbf{z}$. If $p_\theta(\mathbf{z}|\mathbf{x}) = 0$, despite the divergence between $p_\theta(\mathbf{z}|\mathbf{x})$ and $q_\phi(\mathbf{z}|\mathbf{x})$, there would be no consequence because it does not contribute any loss under forward KL. This would result in $q_\phi(\mathbf{z}|\mathbf{x})$ taking a zero-avoiding form during the optimization process. In contrast, minimizing reverse KL loss results in a zero-forcing/mode-seeking behavior because $q_\phi(\mathbf{z}|\mathbf{x})$ must be 0 when $p_\theta(\mathbf{z}|\mathbf{x}) = 0$, causing it to concentrate on the mode. Attempting to fit a unimodal distribution to a multimodal distribution with reverse KL would result in high false-negatives. Having set up the foundation for reverse KL divergence, we derive the objective function of VAE from Equation (7) as follows:

$$\begin{aligned}\mathbb{D}[q_\phi(\mathbf{z}|\mathbf{x})||p_\theta(\mathbf{z}|\mathbf{x})] &= \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}[\log q_\phi(\mathbf{z}|\mathbf{x}) - \log p_\theta(\mathbf{z}|\mathbf{x})] \\ &= \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}[\log q_\phi(\mathbf{z}|\mathbf{x}) - \log p_\theta(\mathbf{x}, \mathbf{z})] + \log p_\theta(\mathbf{x}).\end{aligned} \tag{8}$$

Rearranging Equation (8) yields,

$$\mathcal{L}_{\theta,\phi}(\mathbf{x}^i) = \log p_\theta(\mathbf{x}^i) - \mathbb{D}[q_\phi(\mathbf{z}|\mathbf{x}^i)||p_\theta(\mathbf{z}|\mathbf{x}^i)] = \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}^i)}[\log p_\theta(\mathbf{x}^i, \mathbf{z}) - \log q_\phi(\mathbf{z}|\mathbf{x}^i)], \tag{9}$$

which denotes the variational lower bound, also known as the evidence lower bound (ELBO) for data vector $\mathbf{x}^i$. By Jensen's inequality $\mathbb{D}[q_\phi(\mathbf{z}|\mathbf{x}^i)||p_\theta(\mathbf{z}|\mathbf{x}^i)] \geqslant 0$, which then $\Rightarrow\ \mathcal{L} \leqslant \log p_\theta(\mathbf{x}^i)\ \Rightarrow\ \mathcal{L}$ is a strict lower bound. As $\log p_\theta(\mathbf{x}^i)$ does not depend on q, this equation additionally shows that maximizing the ELBO on the right-hand side (RHS) maximizes $\mathbb{D}[q_\phi(\mathbf{z}|\mathbf{x}^i)||p_\theta(\mathbf{z}|\mathbf{x}^i)]$. Hence, instead of minimizing the KL divergence between the true and variational posteriors (computationally intractable), we can simply maximize the ELBO, which is computationally tractable. A further simplification of the RHS of Equation (9) yields

$$\mathcal{L}_{\theta,\phi}(\mathbf{x}^i) = \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}^i)}[\log p_\theta(\mathbf{x}^i|\mathbf{z})] - \mathbb{D}[q_\phi(\mathbf{z}|\mathbf{x}^i)||p_\theta(\mathbf{z})], \tag{10}$$

which provides an intuitive explanation for the VAE objective function. Equation (9) can now be interpreted as an attempt to model the true distribution under some error term ($\mathbb{D}[q_\phi(\mathbf{z}|\mathbf{x}^i)||p_\theta(\mathbf{z}|\mathbf{x}^i)]$), which is found by maximizing the mapping from latent space to data ($\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}^i)}[\log p_\theta(\mathbf{x}^i|\mathbf{z})]$) and minimizing the difference between the sample and latent distribution ($\mathbb{D}[q_\phi(\mathbf{z}|\mathbf{x}^i)||p_\theta(\mathbf{z})]$).

To optimize the objective function in Equation (10), VAE models the parameters of the approximate posterior $q_\theta(\mathbf{z}|\mathbf{x}^i)$ using an inference network denoted $f(\mathbf{x}^i, \phi)$, whereas the parameters of the true posterior $p_\theta(\mathbf{x}^i|\mathbf{z})$ are modeled using a generative network denoted $g(\mathbf{x}^i, \theta)$. Hence, the analogy of AEs to VAE. Generative and inference networks model the parameter space of the distribution rather than the value of the distribution itself. Stochastic gradient descent (SGD) is used to optimize the network parameters. We can change the order of expectation and derivative for the generative network due to the independence of θ and $q_\phi(\mathbf{z}|\mathbf{x}^i)$. Then the gradient of loss function with respect to the generative network parameter θ is found as follows:

$$\begin{aligned}\nabla_\theta\, \mathcal{L}_{\theta,\phi}(\mathbf{x}^i) &= \nabla_\theta\, \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}^i)}[\log p_\theta(\mathbf{x}^i, \mathbf{z}) - \log q_\phi(\mathbf{z}|\mathbf{x}^i)] \\ &= \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}^i)}[\nabla_\theta(\log p_\theta(\mathbf{x}^i, \mathbf{z}) - \log q_\phi(\mathbf{z}|\mathbf{x}^i))] \\ &\simeq \nabla_\theta(\log p_\theta(\mathbf{x}^i, \mathbf{z}) - \log q_\phi(\mathbf{z}|\mathbf{x}^i)) \\ &= \nabla_\theta(\log p_\theta(\mathbf{x}^i, \mathbf{z})).\end{aligned} \tag{11}$$

However, the condition of changing the order of expectation and derivative may not be true when finding gradients with respect to the inference network parameter ϕ, since the expectation is taken with respect to $q_\phi(\mathbf{z}|\mathbf{x}^i)$,

$$\begin{aligned}\nabla_{\boldsymbol{\phi}}\,\mathcal{L}_{\theta,\boldsymbol{\phi}}(\mathbf{x}^i) &= \nabla_{\boldsymbol{\phi}}\,\mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x}^i)}[\log p_\theta(\mathbf{x}^i,\mathbf{z}) - \log q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x}^i)] \\ &\neq \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}^i)}[\nabla_{\boldsymbol{\phi}}(\log p_\theta(\mathbf{x}^i,\mathbf{z}) - \log q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x}^i))].\end{aligned} \tag{12}$$

To overcome the computational complexity involved in obtaining the gradients w.r.t $\boldsymbol{\phi}$, the reparameterization trick is introduced, which uses the change of variable method to improve the efficiency of Monte Carlo while allowing SGD (Kingma & Welling, 2013; Rezende et al., 2014). It is done by expressing the approximate posterior as a differential transformation of the random variable $\epsilon \sim p(\epsilon)$, whereby $\epsilon \perp \boldsymbol{\phi}, \mathbf{x}^i$. Under the reparameterization $\tilde{\mathbf{z}} = g(\epsilon, \boldsymbol{\phi}, x)$, we can replace an expectation with respect to $q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x}^i)$ with another with respect to $p(\epsilon)$ as follows:

$$\begin{aligned}\mathcal{L}_{\theta,\boldsymbol{\phi}}(\mathbf{x}^i) &= \mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x}^i)}[\log p_\theta(\mathbf{x}^i,\mathbf{z}) - \log q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x}^i)] \\ &= \mathbb{E}_{p(\epsilon)}[\log p_\theta(\mathbf{x}^i,\tilde{\mathbf{z}}^{i,l}) - \log q_{\boldsymbol{\phi}}(\tilde{\mathbf{z}}^{i,l}|\mathbf{x}^i)],\end{aligned} \tag{13}$$

where the posterior is sampled using $\tilde{\mathbf{z}}^{\,i,l} = g_{\boldsymbol{\phi}}(\mathbf{x}^{(i)}, \epsilon^{(l)})$, l denoting each sample. For the distribution and the posterior distribution of the latent variable $\mathbf{z}$, which are $p_\theta(\mathbf{z})$ and $q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x}^i)$, the common choice of distribution is isotropic normal for two reasons. First, the relationship among variables in the latent variable space is simpler than that in the original input data space, and second, it allows a greater computational simplicity. The distributions of the likelihood $p_\theta(\mathbf{x}^i|\mathbf{z})$ are determined by the data types. Thus, if the data are binary, the Bernoulli distribution is used, whereas if the data are continuous, the multivariate Gaussian distribution is used. In the context of fraud detection, because data are continuous, the multivariate Gaussian distribution is chosen. Thus, $\mathcal{N}(0, I)$ is chosen as the latent variable distribution, whereas $q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x}^i)$ is chosen to follow a multivariate Gaussian distribution $\mathcal{N}(\mu^i, \Sigma^i)$ with latent dimension k, where Σ^i is restricted to a diagonal matrix. μ^i and Σ^i are produced by the inference network, taking $\mathbf{x}^i$ as its input. Next step would be to derive the KL divergence between two multivariate Gaussian distributions to solve for the second term in the VAE objective function from Equation (10). For example, KL divergence between $\mathcal{N}_1(\mu_1, \Sigma_1)$ and $\mathcal{N}_2(\mu_2, \Sigma_2)$ is found as follows:

$$\begin{aligned}\mathbb{D}[\mathcal{N}_1||\mathcal{N}_2] &= \mathbb{E}_{\mathcal{N}_1}\left[\frac{1}{2}\log\frac{|\Sigma_2|}{|\Sigma_1|} - \frac{1}{2}(\mathbf{x}-\mu_1)^T\Sigma_1^{-1}(\mathbf{x}-\mu_1) + \frac{1}{2}(\mathbf{x}-\mu_2)^T\Sigma_2^{-1}(\mathbf{x}-\mu_2)\right] \\ &= \frac{1}{2}\log\frac{|\Sigma_2|}{|\Sigma_1|} - \frac{1}{2}\mathbb{E}_{\mathcal{N}_1}\left[(\mathbf{x}-\mu_1)^T\Sigma_1^{-1}(\mathbf{x}-\mu_1)\right] + \frac{1}{2}\mathbb{E}_{\mathcal{N}_1}\left[(\mathbf{x}-\mu_2)^T\Sigma_2^{-1}(\mathbf{x}-\mu_2)\right] \\ &= \frac{1}{2}\log\frac{|\Sigma_2|}{|\Sigma_1|} - \frac{1}{2}\mathrm{tr}\{I_k\} + \frac{1}{2}(\mu_1-\mu_2)^T\Sigma_2^{-1}(\mu_1-\mu_2) + \frac{1}{2}\mathrm{tr}\left\{\Sigma_2^{-1}\Sigma_1\right\} \\ &= \frac{1}{2}\left[\log\frac{|\Sigma_2|}{|\Sigma_1|} - k + \mathrm{tr}\left\{\Sigma_2^{-1}\Sigma_1\right\} + (\mu_2-\mu_1)^T\Sigma_2^{-1}(\mu_2-\mu_1)\right].\end{aligned} \tag{14}$$

Thus, with prior being $\mathcal{N}(0, I)$, VAE objective function from Equation (10) could be expressed as follows:

$$\begin{aligned}\mathcal{L}_{\theta,\phi}(\mathbf{x}^i) &= \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}^i)}[\log p_\theta(\mathbf{x}^i|\mathbf{z})] - \mathbb{D}[q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x}^i)||p_\theta(\mathbf{z})] \\ &= \frac{1}{L}\sum_{l=1}^{L}\log p_\theta(\mathbf{x}^i|\tilde{\mathbf{z}}^{i,l}) + \frac{1}{2}(-\mathrm{tr}(\Sigma^i) - (\mu^i)^\top(\mu^i) + k + \log\det(\Sigma^i)),\end{aligned} \tag{15}$$

where $\tilde{\mathbf{z}}^{i,l} = g_{\phi}(\mathbf{x}^i, \epsilon^l) = \mu^i + \Sigma^i \odot \epsilon^l$ is sampled L times using the auxiliary noise variable $\epsilon^l \sim \mathcal{N}(0, I)$. The decoder function $\log p_{\theta}(\mathbf{x}^i|\tilde{\mathbf{z}}^{i,l})$ is implemented through a Gaussian MLP consisting of three decoding layers in $\mathcal{D}_1$ and five decoding layers in the case of $\mathcal{D}_2$, with varying dimension and symmetric encoding/decoding architecture.

4 | FRAUD CLASSIFICATION

As a prelude to the main objective of the paper, it is important to understand how fraud classification will be conducted using AE/VAEs. Hence, this section is structured to acquaint users with how to infer insights from RE/log probability distributions, allowing them to make classifications. Given that $\mathcal{D}_1$ and $\mathcal{D}_3$ are insurance-related data sets and that $\mathcal{D}_3$ is of a higher quality, results from $\mathcal{D}_2$ and $\mathcal{D}_3$ are used in the section for illustration.

4.1 | AE for classification

The first step toward fraud classification is to obtain the aggregate reconstruction error (A-RE) distribution for the underlying data. A-RE is calculated as the $\sum_{j=1}^{m}(\mathbf{x}_j^i - \hat{\mathbf{x}}_j^i)$ for each test instance (20% of the data withheld during the training phase). Figures 2 and 3 illustrate the A-RE distribution produced by AE on the testing data set. This far, while developing AE and testing it on the testing data set, it has not yet used the output labels; thus, it is independent of the output labels. From Figures 2 (more so) and 3, we deduce that a major proportion of testing instances consist of low A-RE, while some tend to have high A-RE. The objective of the AE was to encode and decode training data, which were extremely skewed toward nonfraud instances. In other words, the AE was calibrated for reconstructing nonfraud instances with minimum A-RE as possible. It could then be postulated that low A-RE indicates instances consisting of nonfraud characterization, whereas high A-RE instances are indicative of fraud characterizations. This will allow us to set a threshold for A-RE to classify fraud and nonfraud instances. For example, for threshold T, A-RE $< T$ are classified as nonfraud, whereas A-RE $\geqslant T$ are classified as fraud. With the presence of a testing data set with output labels, as is the case with $\mathcal{D}_2$ and $\mathcal{D}_3$, validation of the threshold can be conducted at convenience. A-RE distributions depicted in Figures 2 and 3 are matched against the true outcome labels of the testing data, shown by Figures 3 and 4.

Figures 4 and 5 shown above substantiate the model dynamics, and the sound judgment used upon the A-RE stands true to a large extent. That is, a major proportion of nonfraud and fraud instances consisted of low A-RE and high A-RE, respectively. Although some fraud instances showed similar characterizations to the nonfraud instances (fraud instances having low A-RE), a higher proportion of fraud instances tended to have characterizations that varied from those of nonfraud instances, that is, high A-RE. This observation was more evident in $\mathcal{D}_2$ than in $\mathcal{D}_3$. As was explained in the Data section, the $\mathcal{D}_3$ data set, unlike the $\mathcal{D}_1$ and $\mathcal{D}_2$ data sets, contains 2,379 actual fraud findings, with the rest of the data unidentified. In the context of fraud versus unidentified, $\mathcal{D}_3$ contains a larger proportion of unidentified cases indicative of fraud characterizations, which is plausible. Therefore, using this methodology, we were able to obtain a distribution of the binary state representation of unlabeled data through salient feature extraction with no requirement of output labels and user interference regarding explicit parameterization. This enabled us to make insightful decisions, which was not possible in the prior state of idle data.

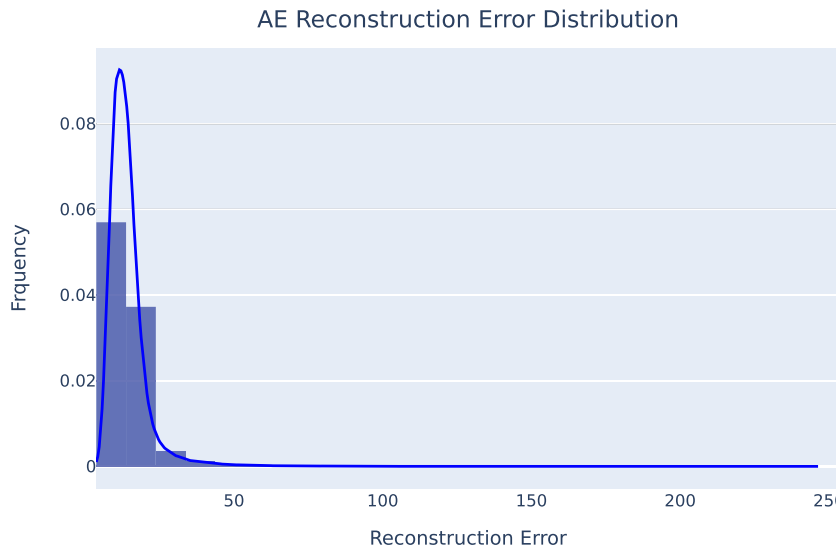

FIGURE 2 $\mathcal{D}_2$ data set: Prior validation. AE, autoencoder [Color figure can be viewed at wileyonlinelibrary.com]

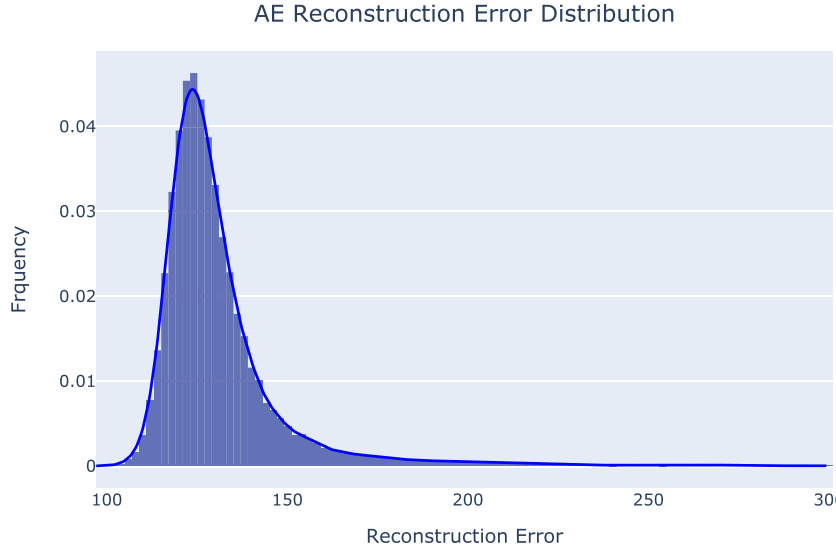

FIGURE 3 $\mathcal{D}_3$ data set: Prior validation. AE, autoencoder [Color figure can be viewed at wileyonlinelibrary.com]

4.2 | VAE for classification

As was the case with AE, A-RE produced by VAE could be used for fraud classification, as it reflects a similar characterization in terms of the shape of the distribution. Figures 6 and 7 illustrate the A-RE distributions in the absence of output labels. As was the case with AE, we must observe the VAE objective before decision making. VAE was tasked with optimizing the model architecture to maximize the probability assigned for the training data. Training data, with our priori

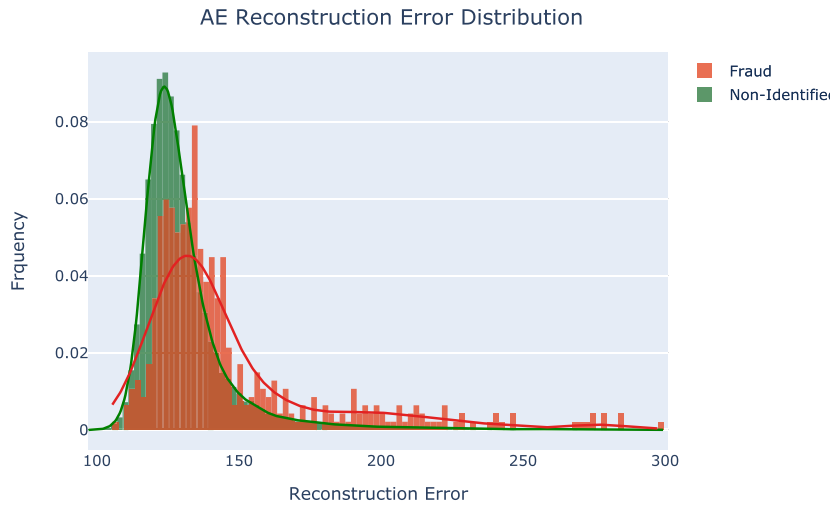

FIGURE 4 $\mathcal{D}_3$ data set: Postvalidation. AE, autoencoder [Color figure can be viewed at wileyonlinelibrary.com]

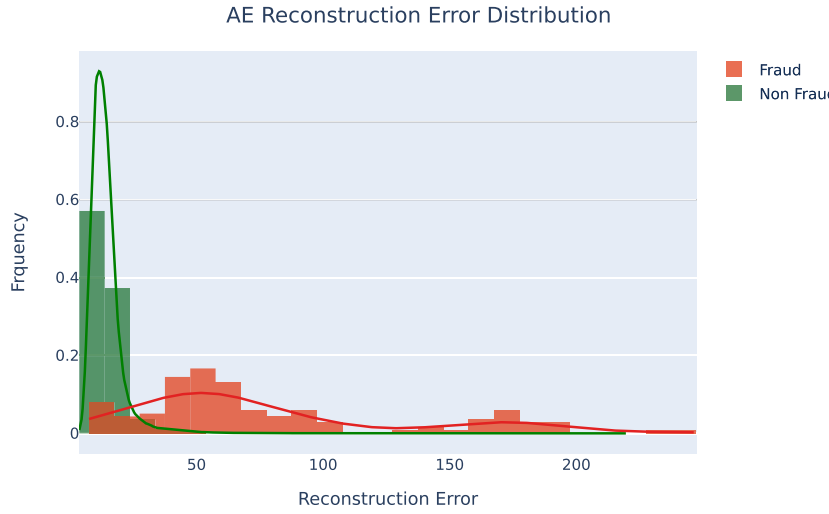

FIGURE 5 $\mathcal{D}_2$ data set: Postvalidation. AE, autoencoder [Color figure can be viewed at wileyonlinelibrary.com]

assumption, is extremely skewed toward nonfraud data, which implies the VAE is optimized for assigning higher probabilities to nonfraud instances. As previously discussed, the VAE does this through salient feature extraction from the heavily skewed data distribution (exploited from the asymmetric lower dimension of the inference network as opposed to the visible dimension). Therefore, it is correct to think of the A-RE distribution as being representative of the salient features. As was the case with AE, we can postulate that a low A-RE bound will allow for nonfraud instances, whereas a higher A-RE bound will allow for fraud instances on probabilistic grounds. However, having a validation data set with output labels will allow us to choose the optimized A-RE threshold. Figures 8 and 9 illustrate the A-RE distributions produced on testing data (Figures 6 and 7) matched against the true outcome labels.

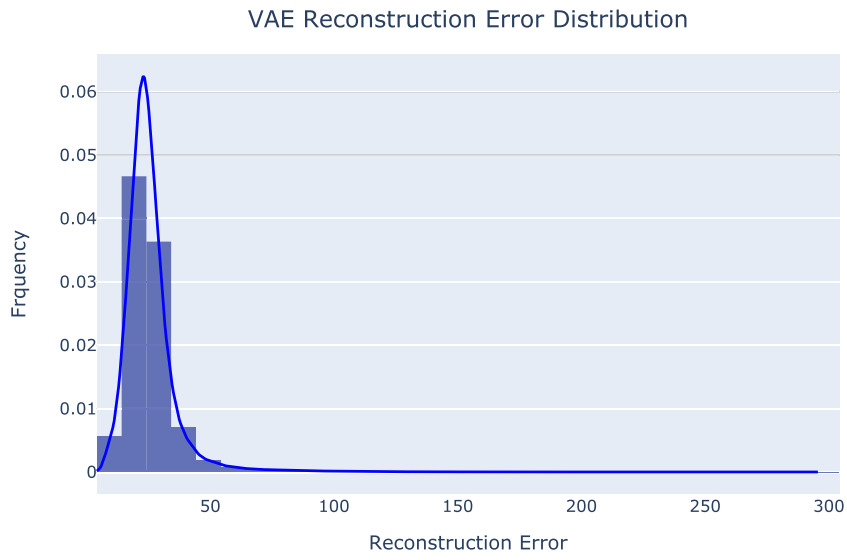

FIGURE 6 $\mathcal{D}_2$ data set: Prior validation. VAE, variational autoencoder [Color figure can be viewed at wileyonlinelibrary.com]

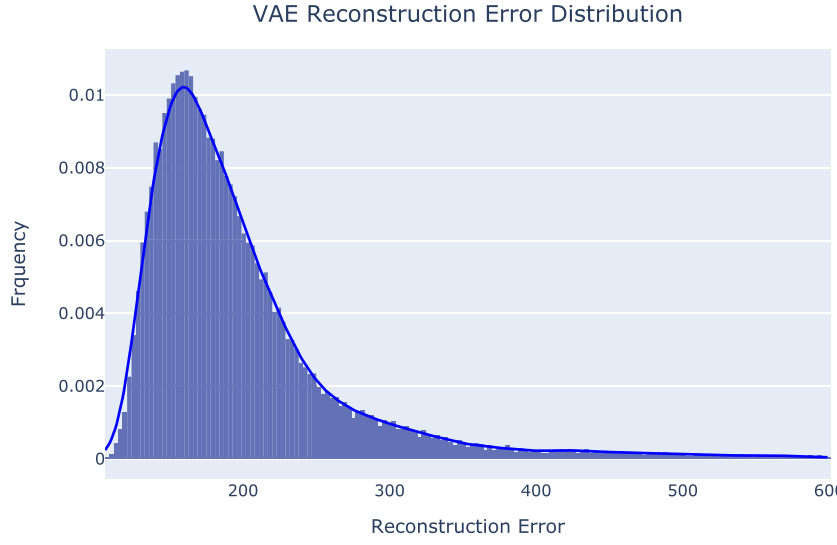

FIGURE 7 $\mathcal{D}_3$ data set: Prior validation. VAE, variational autoencoder [Color figure can be viewed at wileyonlinelibrary.com]

Unlike the AE, the VAE offers a wider scope of methodologies for fraud classification. For example, we can use reconstruction probability (An & Cho, 2015) to analyze the fraudulent behavior. Reconstruction probability involves the calculation of expected posterior log probability for a single data point $\mathrm{E}_{q_\phi(\mathbf{z}|\mathbf{x})}[\log p_\theta(\mathbf{x}|z)]$ using Markov chain Monte Carlo. The reconstruction probability distribution produced by the VAE on the testing data follows a similar trend to that of the A-RE from the AE. High reconstruction probability relates to nonfraud instances, whereas low reconstruction probability relates to fraudulent anomalies. A close

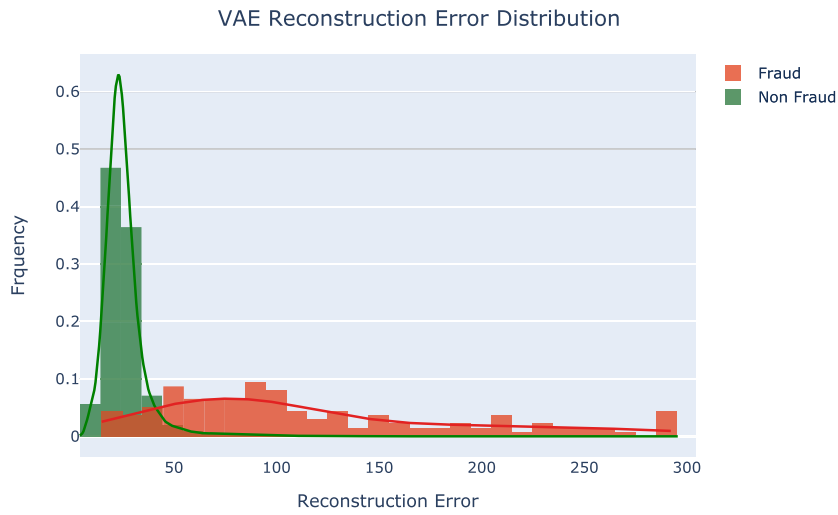

FIGURE 8 $\mathcal{D}_2$ data set: Postvalidation. VAE, variational autoencoder [Color figure can be viewed at wileyonlinelibrary.com]

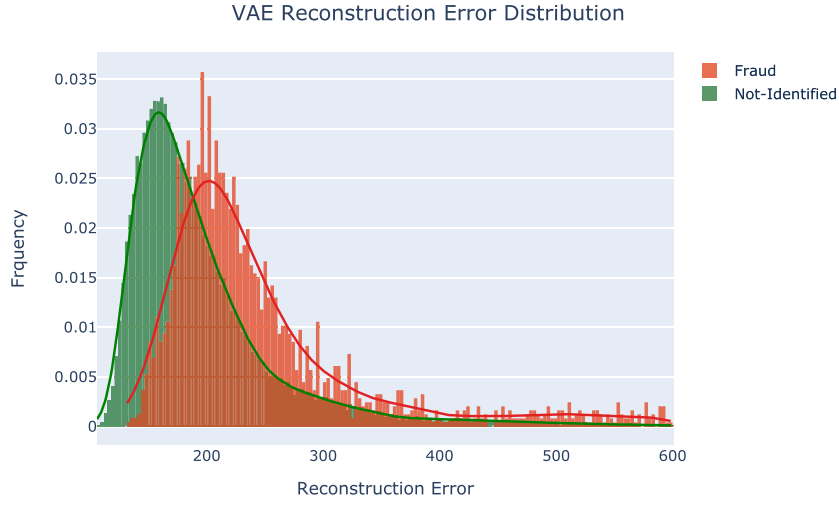

FIGURE 9 $\mathcal{D}_3$ data set: Postvalidation. VAE, variational autoencoder [Color figure can be viewed at wileyonlinelibrary.com]

observation of the VAE dynamics allows the user to use reconstruction probability to analyze criminal behavior instead of using A-RE produced by the VAE. One may notice that A-RE is incommensurable when used to compare model performances with one another, whereas reconstruction probability provides a uniform unit of measure. Figure 6 shows the reconstruction probability distribution of the testing data set. Once again, we observe that a major proportion indicates high reconstruction probability, while a minority comprises low reconstruction probability. Therefore, we can use a reconstruction probability bound for making classifications, as was the case with the A-RE bound, which will be discussed later. However, reconstruction probability does not allow variable importance, which is presented in this paper. The following section elucidates the reasoning behind this.

5 | PROPOSED METHODOLOGY FOR VARIABLE IMPORTANCE

In this section, we propose a methodology that enables AEs and VAEs to be used in semisupervised/unsupervised variable importance analysis for identifying drivers of fraud. The proposed methodology can be considered as an extension of A-RE analysis in the above section. The underlying concept of this methodology is to consider node level reconstruction error (RE) defined as $\sum_{i=1}^{N}(\mathbf{x}_j^i - \hat{\mathbf{x}}_j^i)$ (denoted by NL-RE) for each j feature across all test vectors, as opposed to solely contemplating on the A-RE discussed in the previous section ($\sum_{j=1}^{m}(\mathbf{x}_j^i - \hat{\mathbf{x}}_j^i)$) for each test vector i).

The proposed methodology for fraud variable importance analysis consists of three steps. In Step 1, we perform an A-RE analysis and obtain a proportion of test instances having an A-RE greater than a set threshold. In other words, we limit the scope of analysis to the upper tail of the A-RE distribution discussed in Section 4. In Step 2, we perform aggregate NL-RE analysis for the limited scope of test instances. Finally, in Step 3, we perform NL-RE analysis on the lower tail of the RE distribution and proceed to disregard common variables found in Step 2. Because of the priori assumption, having verified earlier that a high A-RE denotes a high likelihood of the test instance being fraud, Step 1 makes intuitive sense. It merely attempts to set aside fraud instances for variable importance analysis, so that the variable importance denotes a key driver of fraud. Variables with the highest aggregate NL-RE indicate leading drivers of fraud, while variables with the least aggregate NL-RE are of least importance to fraud analysis. The findings of this methodology are discussed separately for each of the three data sets used in the study.

5.1 | Variable importance: $\mathcal{D}_1$ data set

Figure 10 shows the NL-RE for the upper tail of the A-RE distribution produced by the AE, which indicates that the input variables location code, marital status, and education of a policy holder are the three drivers of fraud. Figure 11 depicts the NL-RE for the upper tail of the A-RE distribution produced by the VAE. The VAE identifies claim reason, location code, and marital status as the three drivers of fraud.

It is possible that analysis can produce misleading results if NL-RE is not conducted on the upper tail of the A-RE distribution. Hence, Step 1 of the variable importance algorithm plays a crucial role in capturing the drivers of fraud. Figures 12 and 13 depict NL-RE conducted on the A-RE distribution produced by the AE and VAE. As we might expect, aggregate NL-RE across the entire distribution does not narrow the scope to obtain the desired drivers of fraud. Instead, it shows the overall model performance for a mixture of fraud and nonfraud instances, which does not offer any valuable insights to the user. If we were to disregard this important step, we would be misled to choose education, months since policy inception, and number of open complaints, in that order, as the drivers of fraud based on the A-RE distribution produced by AE, and sales channel, marital status, and employment status based on the A-RE distribution produced by VAE. However, the true drivers of fraud based on AE and VAE as shown by Figures 10 and 11, respectively, are location code, marital status, education, and claim reason.

As can be seen from the following section, there can be situations with overlapping important variables between analyses that included Step 1 and did not include Step 2, that is, the AE includes

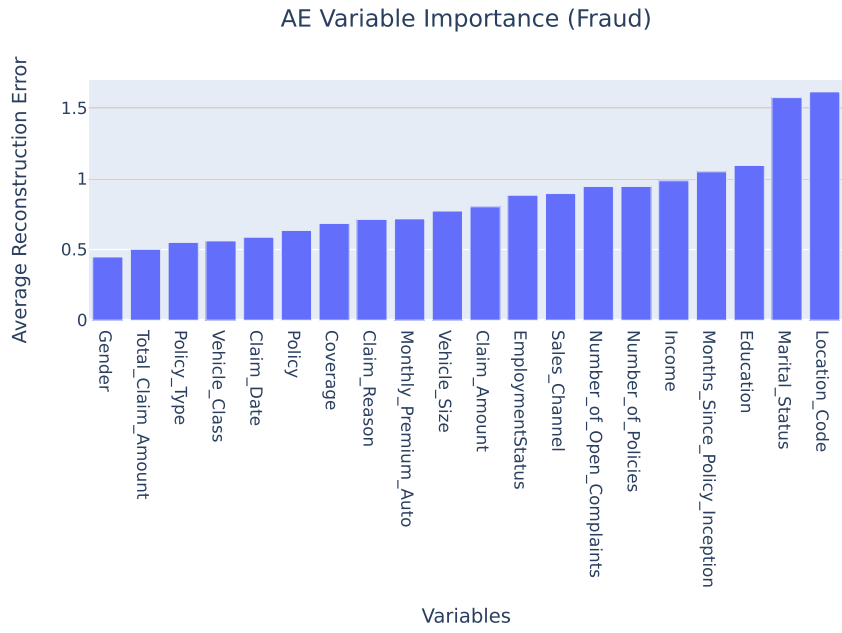

FIGURE 10 $\mathcal{D}_1$ data set (upper-tail): AE. AE, autoencoder [Color figure can be viewed at wileyonlinelibrary.com]

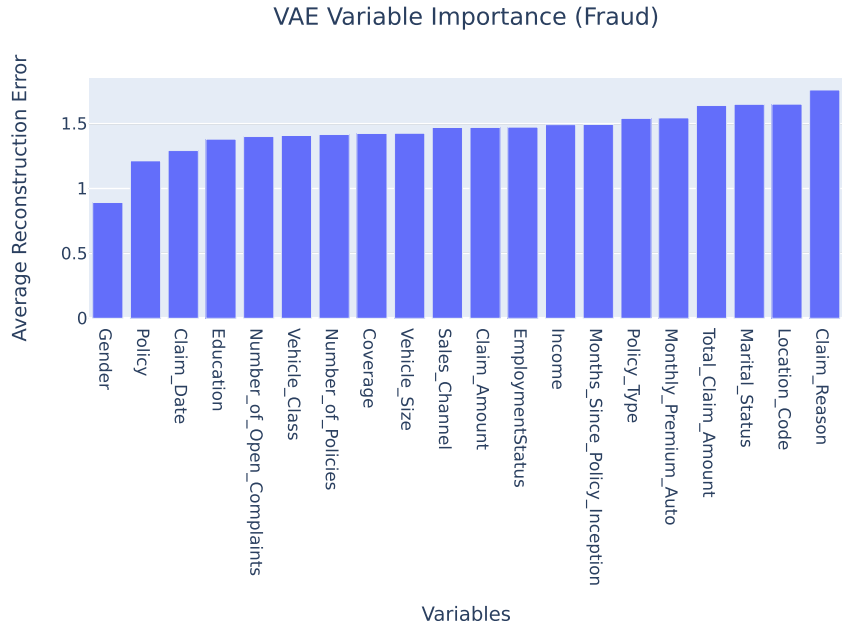

FIGURE 11 $\mathcal{D}_1$ data set (upper-tail): VAE. VAE, variational autoencoder [Color figure can be viewed at wileyonlinelibrary.com]

education as an important driver of fraud in its top three fraud drivers, whereas the VAE includes employment status as an important driver in its top three fraud drivers in both situations (despite a change in the order of importance). This could possibly be due to features with complex internal representations, causing the model to have high RE when mapping. However, this would then

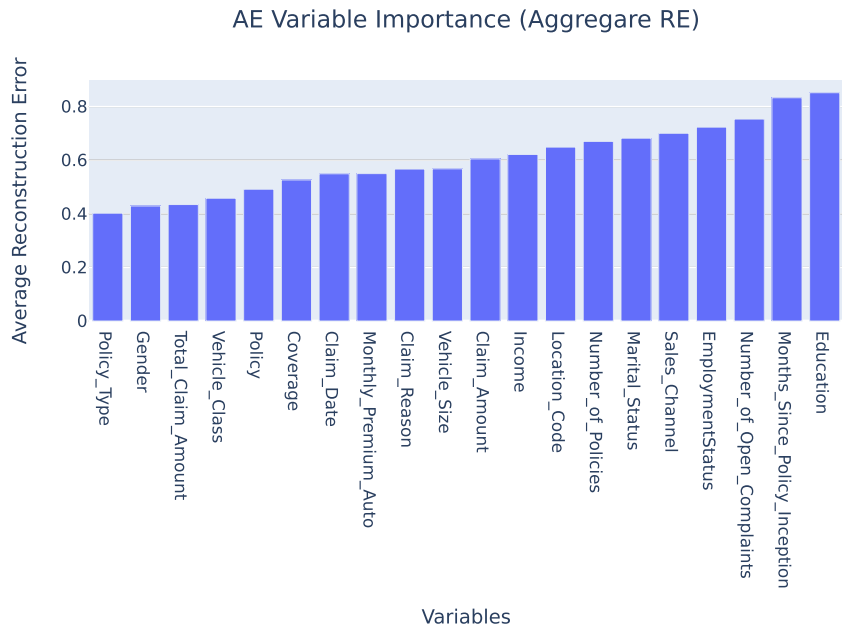

FIGURE 12 $\mathcal{D}_1$ data set NL-RE (aggregate): AE. AE, autoencoder; NL, node level; RE, reconstruction error [Color figure can be viewed at wileyonlinelibrary.com]

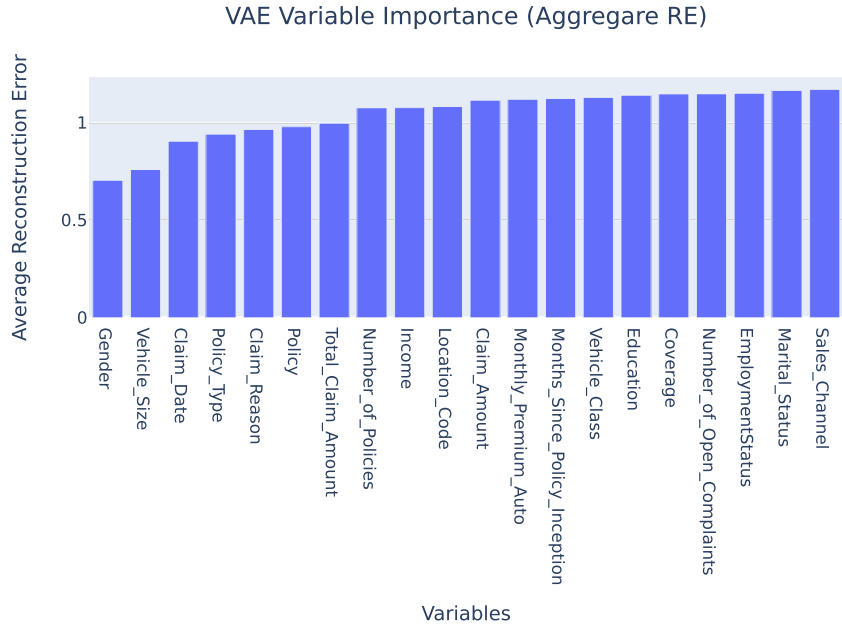

FIGURE 13 $\mathcal{D}_1$ data set NL-RE (aggregate): VAE. NL, node level; RE, reconstruction error; VAE, variational autoencoder [Color figure can be viewed at wileyonlinelibrary.com]

violate the driver of fraud concept, which was discussed above, as it could be an important variable even if one were to examine the NL-RE of the lower tail of the A-RE distribution. Hence, to overcome this limitation, the third step is introduced, which involves a comparison of the variable importance for the upper tail of the A-RE distribution with that for the lower tail. Subsequently, we proceed to disregard any common variables allowing us to obtain distinct drivers of fraud.

Applying the third step of the methodology to $\mathcal{D}_1$ data set may not yield the expected results, as both the upper and lower tails consist of common variables in its ranking. This could be because of the nature of data and model performance. As discussed earlier, the $\mathcal{D}_2$ data set is far more skewed than the $\mathcal{D}_1$ data set, and thus, it is a more suitable candidate for the set unsupervised variable importance methodology. At Step 1 of the methodology, the $\mathcal{D}_1$ data set proved to be a poor candidate for this methodology, as it failed to display a clearly "skewed" RE distribution. This makes it ineligible for this unsupervised methodology, let alone Steps 2 and 3. If we were to proceed, we would obtain misleading results as shown. However, before coming to this conclusion, we should make reasonable effort to modify the model architecture and perform necessary hyper parameter tuning to achieve better model performance or, in this case, a skewed A-RE distribution. Several regularization techniques were used over the course of training the models, including but not limited to batch-normalization (Ioffe & Szegedy, 2015), early stopping, and dropout (Srivastava et al., 2014).

5.2 | Variable importance: $\mathcal{D}_2$ data set

When Step 1 of the algorithm was applied to the $\mathcal{D}_2$ data set, the following variable importance distribution, depicted by Figures 14 and 15, was produced. Based on this, the AE identified V1, V8, and V2 as the top three fraud drivers, whereas the VAE identified V2, V8, and V7 as the fraud drivers.

A comparison of the top three variables in Steps 1 and 2 reveals that both steps share common variables such as V2 and V8 using the AE. It could be that V2 and V8 contain a larger weight of RE, and to confirm this, we must proceed to Step 3 (Figures 16 and 17).

For Step 3, Figure 18 indicates NL-RE for the lower tail of the A-RE distribution produced by the AE on the $\mathcal{D}_2$ data set, where the key NL-RE drivers are V13, V1, and V9. Because the drivers of fraud at Step 2 were identified as V1, V8, and V2, where V1 is a common feature, it is recommended to disregard V1 as a driver and consider V8 and V2 as the drivers of fraud. Applying the same thinking to the A-RE distribution produced by the VAE (Figure 19), we found that the true drivers of fraud remained the same as in the initial Step 2 examination, that is, V2, V8, and V7.

5.3 | Variable importance: $\mathcal{D}_3$ data set

In applying the proposed algorithm to the $\mathcal{D}_3$ data set, we should be mindful of the nature of the data. Unlike the previous two fraud versus nonfraud data sets, this data set compares the non-identified with the fraud. The AE identifies V239, V248, and V68 as the drivers of fraud, whereas the VAE identifies V74, V28, and V36 as the fraud drivers (Figures 20 and 21).

To verify that these drivers are indeed actual fraud drivers, Step 2 is performed as a comparison using Figures 22 and 23. As none of the top three variables is common among the A-RE distributions, we can proceed to consider them as drivers of fraud and disregard Step 3.

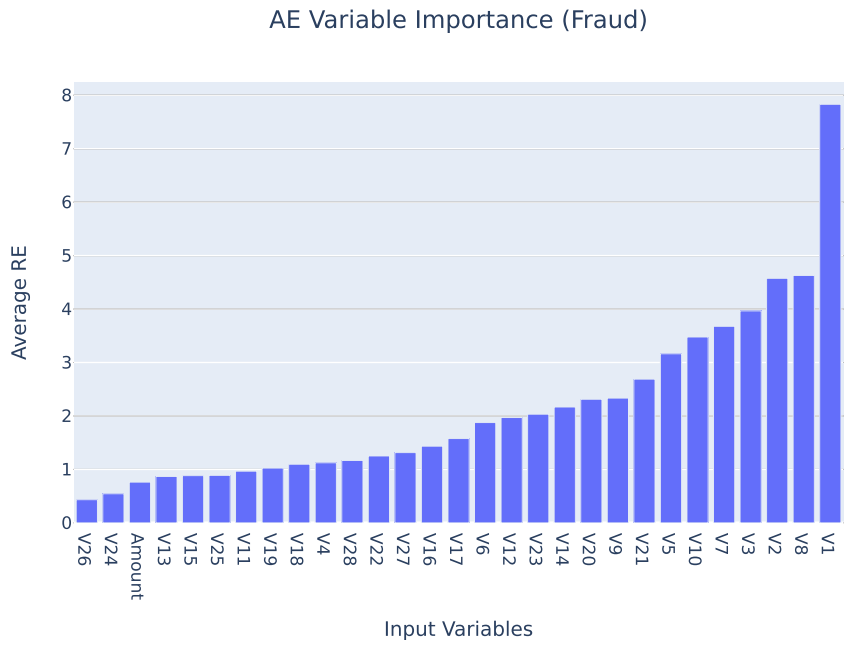

FIGURE 14 $\mathcal{D}_2$ data set (upper-tail): AE. AE, autoencoder [Color figure can be viewed at wileyonlinelibrary.com]

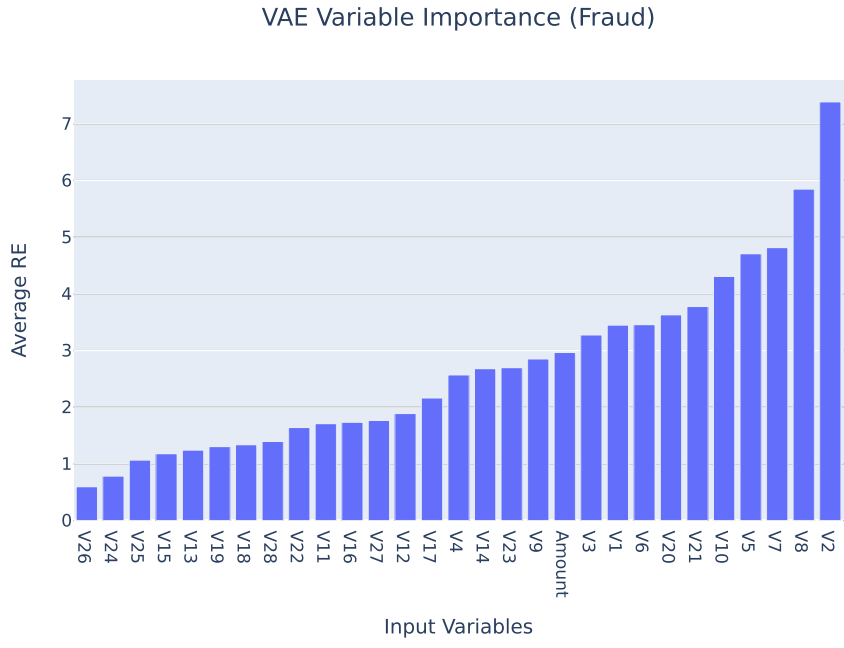

FIGURE 15 $\mathcal{D}_2$ data set (upper-tail): VAE. VAE, variational autoencoder [Color figure can be viewed at wileyonlinelibrary.com]

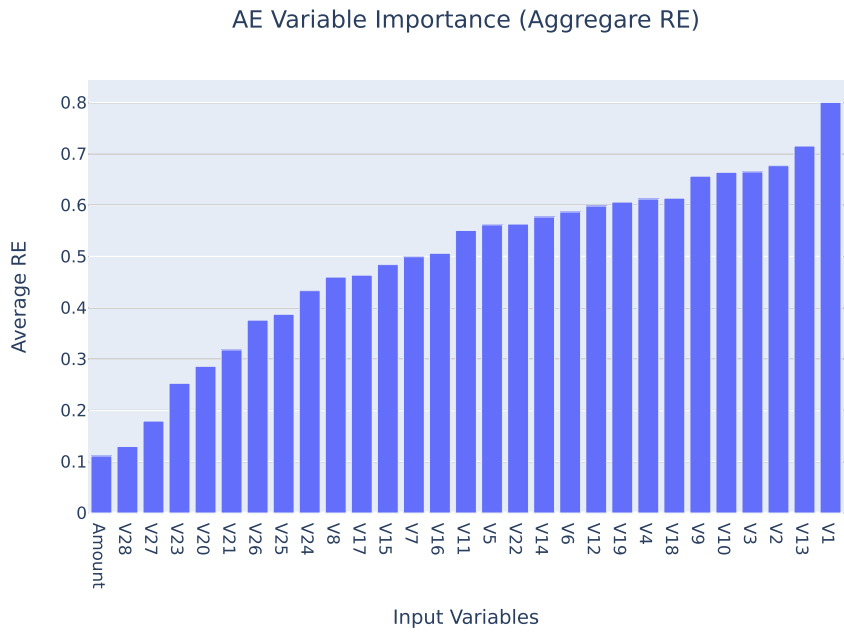

FIGURE 16 $\mathcal{D}_2$ data set NL-RE (aggregate): AE. AE, autoencoder; NL, node level; RE, reconstruction error [Color figure can be viewed at wileyonlinelibrary.com]

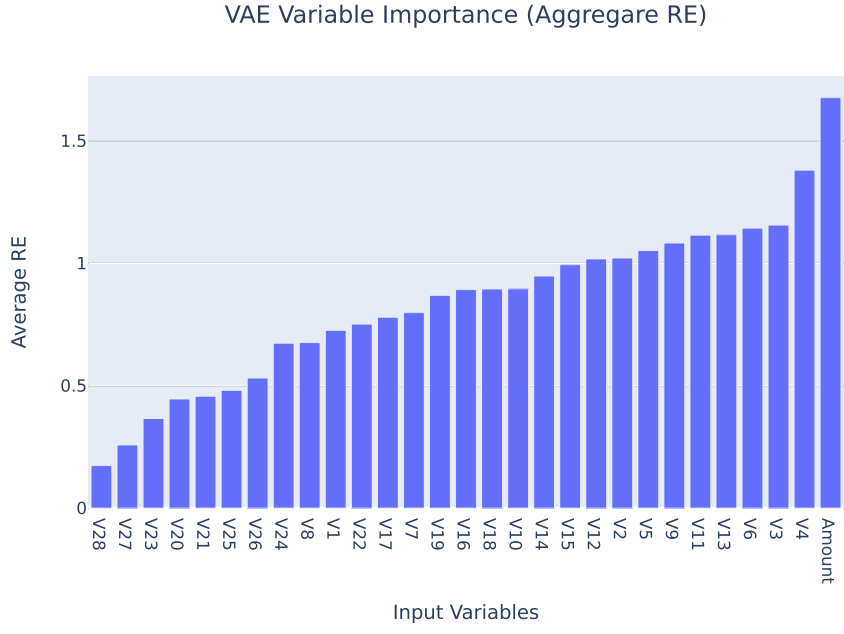

FIGURE 17 $\mathcal{D}_2$ data set NL-RE (aggregate): VAE. NL, node level; RE, reconstruction error; VAE, variational autoencoder [Color figure can be viewed at wileyonlinelibrary.com]

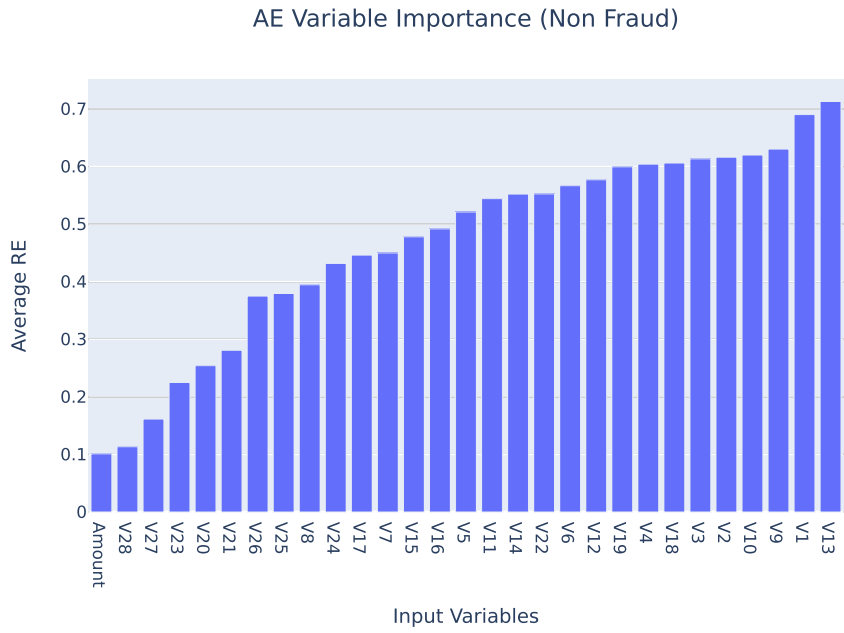

FIGURE 18 $\mathcal{D}_2$ data set NL-RE (lower-tail): AE. AE, autoencoder; NL, node level; RE, reconstruction error [Color figure can be viewed at wileyonlinelibrary.com]

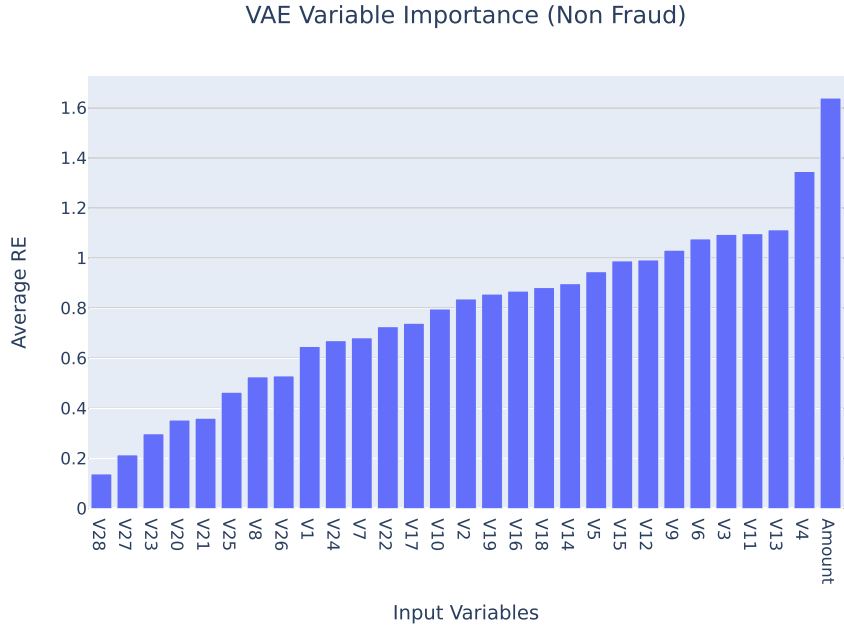

FIGURE 19 $\mathcal{D}_2$ data set NL-RE (lower-tail): VAE. NL, node level; RE, reconstruction error; VAE, variational autoencoder [Color figure can be viewed at wileyonlinelibrary.com]

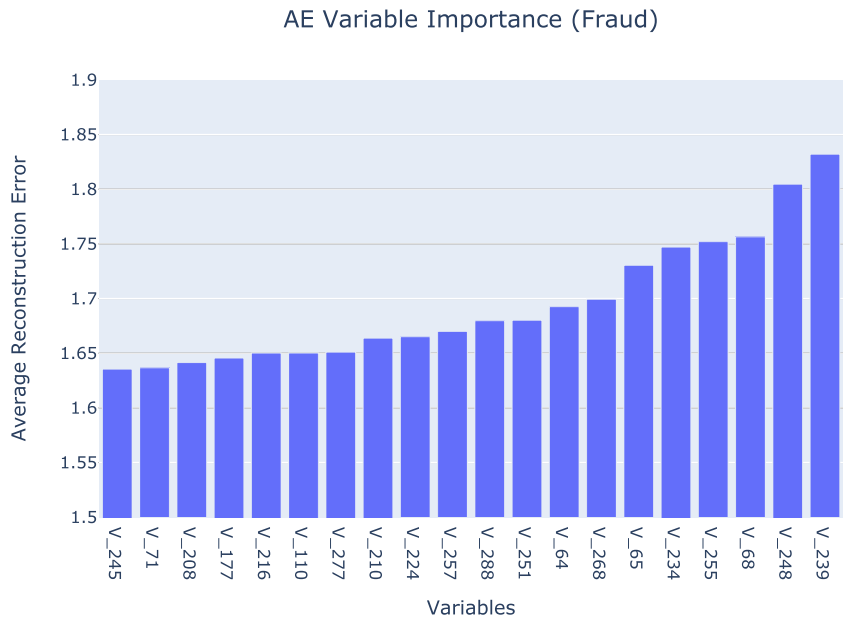

FIGURE 20 $\mathcal{D}_3$ data set NL-RE (upper-tail): AE. AE, autoencoder; NL, node level; RE, reconstruction error [Color figure can be viewed at wileyonlinelibrary.com]

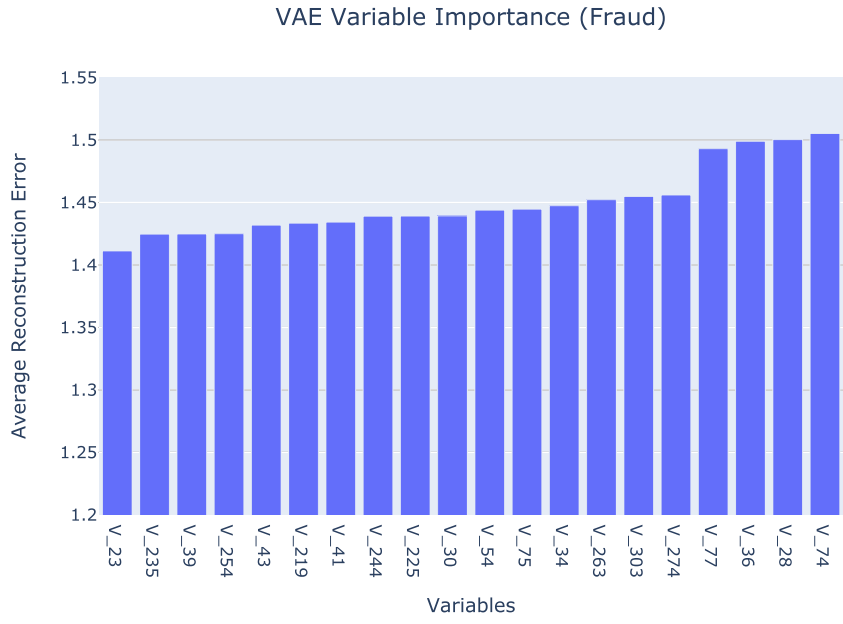

FIGURE 21 $\mathcal{D}_3$ data set NL-RE (upper-tail): VAE. NL, node level; RE, reconstruction error; VAE, variational autoencoder [Color figure can be viewed at wileyonlinelibrary.com]

After justifying the steps in the variable importance algorithm, the question on how to discern drivers of fraud between the VAE and AE results remains. For example, for the $\mathcal{D}_2$ data set, while V2 and V7 are accepted as two of the fraud drivers by both the AE and VAE, a greater differentiation exists among the rest of the input variables. One key reason is the different

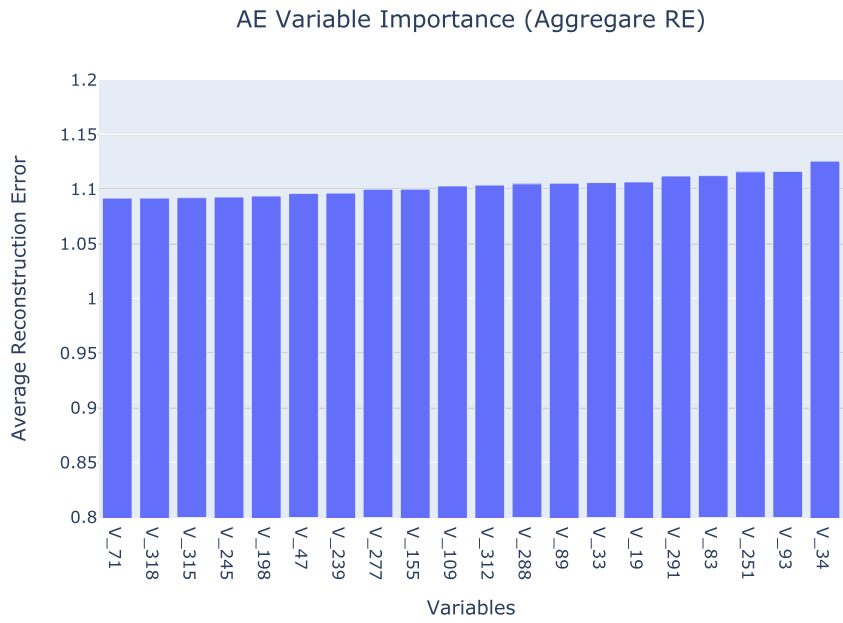

FIGURE 22 $\mathcal{D}_3$ data set NL-RE (aggregate): AE. AE, autoencoder; NL, node level; RE, reconstruction error [Color figure can be viewed at wileyonlinelibrary.com]

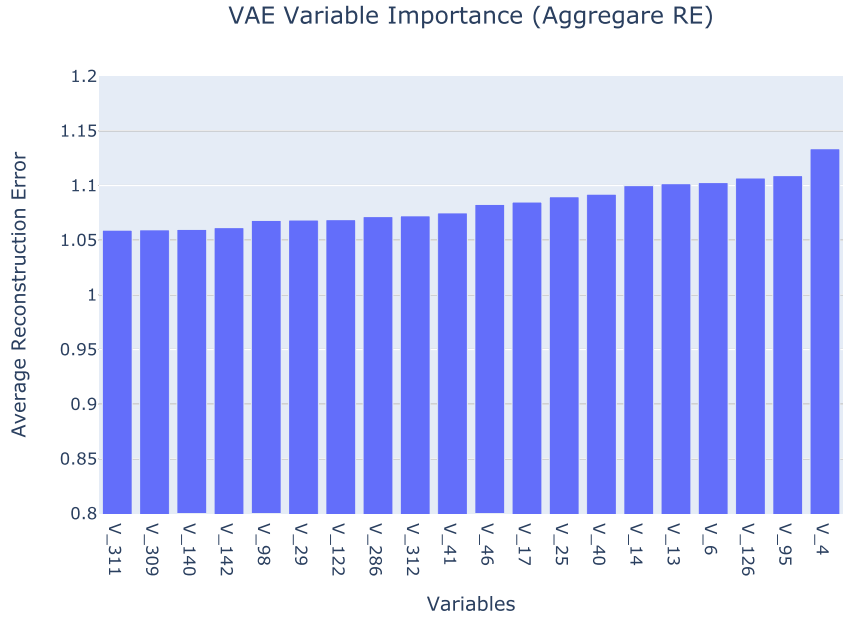

FIGURE 23 $\mathcal{D}_3$ data set NL-RE (aggregate): VAE. NL, node level; RE, reconstruction error; VAE, variational autoencoder [Color figure can be viewed at wileyonlinelibrary.com]

model architectures and dynamics between the AE and VAE as discussed in Section 3. This implies that one of the models could be better at reconstructing the inputs than the other. An attempt can be made to assess which model is better based on output labels, given the presence of output labels, or ongoing user feedback, given it is completely unsupervised.

Many supervised models (e.g., random forest and gradient boosting machines) allow the user to analyze variable importance with ease. However, they possess an inherent bias based on the algorithm used to compute the variable importance. Moreover, the supervised models tend to inflate the importance of continuous features or high-cardinality categorical variables. Thus, the result varies with the type of loss function and the algorithm used for variable importance. Therefore, the variable importance produced by many supervised models is not optimized for identifying fraud-driving variables but is instead useful in identifying variables that are important for model optimization given a set criterion. The proposed methodology above allows variable importance analysis in an unsupervised fashion through both the AE and VAE in the absence of any output labels and inherent algorithmic bias.

6 | RECONSTRUCTION ERROR THRESHOLD

It is important to understand how the RE threshold was set for variable importance analysis. In the presence of reliable output labels for testing data (as was with $\mathcal{D}_1$ and $\mathcal{D}_2$ data sets), we can optimize the A-RE classification threshold through several performance metrics. Owing to the class imbalanced nature of the data, the performance metrics/measures used in this study were recall, precision, and F1 score. Because the two unsupervised deep learning methods were utilized, the metrics were used in a proactive nature, unlike in supervised models where the metrics are used in a retrospective nature. The performance metrics are proactive, as one should alter the threshold to obtain the desired results predicated on a scale of importance; once the scale of importance is achieved, comparison is justifiable—scale of importance is defined as the degree of importance laid between recall and precision by the insurer. Meanwhile, supervised models do not provide such optionality to impose a scale of importance to obtain the insurer's desired results. They only provide one set of results; thus, they are retrospective. This will be intuitive when analyzing results obtained from the study in the following section. Hence, before using these metrics, it is important to understand their functionality.

The first metric, recall, measures the proportion of actual fraud cases correctly identified by the model. It is calculated as follows:

$$\text{Recall} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}}, \tag{16}$$

where true-positive denotes the true fraud instances correctly identified as fraud, and false-negative denotes the fraud instances incorrectly identified as nonfraud. The second measure, precision, is calculated as follows:

$$\text{Precision} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}}, \tag{17}$$

where false-positive denotes nonfraud cases incorrectly predicted as fraud.

Figure 24 shows the sensitivity of precision against RE thresholds. Based on the cost associated with fraud instances for the insurer, a lower RE threshold could be set to allow for greater capture of fraud instances at the cost of false-positive cases. Figure 25 illustrates the sensitivity of recall against RE thresholds.

An inverse relationship exists between recall and precision owing to the denominator of the respective formulas for recall Equation (16) and precision Equation (17); thus, we should expect a trade off when improving one relative to another. Trying to achieve both high precision and

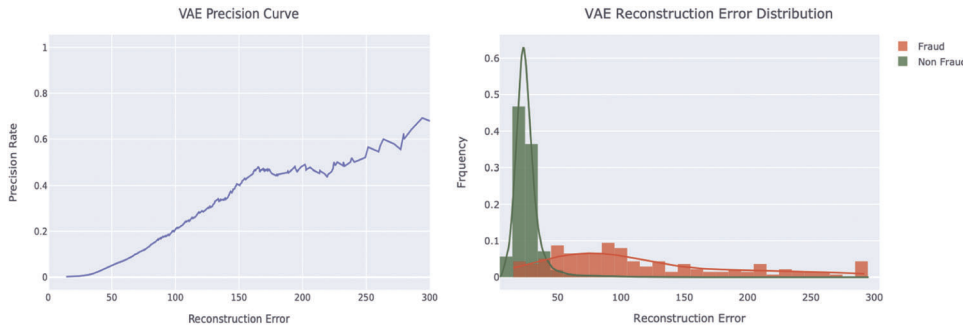

FIGURE 24 Precision vs. reconstruction error [Color figure can be viewed at wileyonlinelibrary.com]

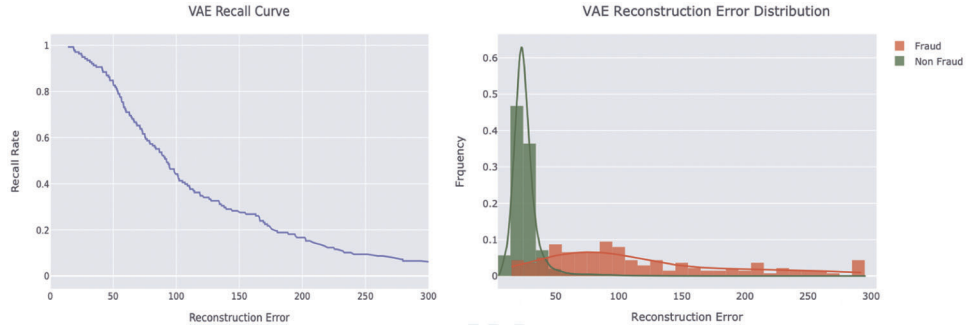

FIGURE 25 Recall vs. reconstruction error [Color figure can be viewed at wileyonlinelibrary.com]

recall closer to one is an impossible task. The third measure, F1, conflates recall and precision together by taking the harmonic mean of the two, forming a single metric. The F1 score is defined as follows:

$$\text{F1} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}. \tag{18}$$

Owing to the high imbalanced nature of the data, considering the recall, precision, and F1 score alone would mislead the user. Many measures are available to overcome this. One such approach is the weighted recall, precision, and F1 measures (W.Recall, W.Precision, and W.F1) expressed as follows:

$$\frac{1}{\sum_{l \in 1,0} |c_l|} \left[\sum_{l \in 1,0} \left| c_l \right| \psi(y_l, \hat{y}_l) \right], \tag{19}$$

where $|c_l|$ is the number of instances in class l, $|\hat{y}_l|$, and y_l denote the predicted and true outcome label in class l, respectively. Let ψ be any function, in this case it is recall, precision, and F1 score at each instance. The numerator weighs each class by its metric score, whereas the denominator acts as the normalizing factor. See Pedregosa et al. (2011), for more details.

A keen observation of these measures will help the user to understand that performance of infrequent classes is given less weight, thus hiding the performance of infrequent classes (fraud

cases). However, this is not true in reality, where the scale of importance given to each class may vary based on business operations. The scale of importance is defined as the degree of importance laid between recall and precision by the insurer. Typically for insurers, it is most costly to have FNs (fraudulent cases being identified as nonfraudulent) than having low FPs (nonfraudulent cases being identified as fraud). An ill-informed understanding of the scale of importance ("imprudent") connotes catastrophic consequences. It can create a situation where the insurer is coaxed into believing that the model is performing well and is intransigent to possible model defections. However, at the model re-evaluation phase, disenchantment is bound to emerge as the illusions appear in the form of loss of market share and increased customer complaints. Thus, the scale of importance should be considered strong priori for good model performance, and it is vital to apply sound judgment when developing it.

Tables 3 and 4 show the performance of the $\mathcal{D}_1$ data set against varying RE thresholds for the VAE and AE, respectively. Because $\mathcal{D}_3$ data set does not contain output labels, such an analysis, as was possible with $\mathcal{D}_1$ and $\mathcal{D}_2$ data sets, could not be conducted. Tables 5 and 6 show the performance of the $\mathcal{D}_2$ data set against varying RE thresholds. To measure performance,

TABLE 3 VAE on $\mathcal{D}_1$ data set

RE-T	TP	FN	FP	TN	Support	Precision	Recall	F1	W.Precision	W.Recall	W.F1
15	331	19	1842	92	2284	0.1523	0.9457	0.2624	0.7252	0.1852	0.1164
18	266	84	1502	432	2284	0.1505	0.7600	0.2512	0.7320	0.3056	0.3371
20	199	151	1111	823	2284	0.1519	0.5686	0.2398	0.7388	0.4475	0.5160
22	129	221	696	1238	2284	0.1564	0.3686	0.2196	0.7425	0.5985	0.6516

Abbreviations: FN, false-negative; FP, false-positive; RE, reconstruction error; TN, true-negative; TP, true-positive.

TABLE 4 AE on $\mathcal{D}_1$ data set

RE-T	TP	FN	FP	TN	Support	Precision	Recall	F1	W.Precision	W.Recall	W.F1
10	280	70	1589	345	2284	0.1498	0.8000	0.2524	0.7269	0.2736	0.2874
12	172	178	973	961	2284	0.1502	0.4914	0.2301	0.7375	0.4961	0.5649
14	54	296	368	1566	2284	0.1280	0.1543	0.1399	0.7318	0.7093	0.7201
15	33	317	207	1727	2284	0.1375	0.0943	0.1119	0.7365	0.7706	0.7524

Abbreviations: FN, false-negative; FP, false-positive; RE, reconstruction error; TN, true-negative; TP, true-positive.

TABLE 5 VAE on $\mathcal{D}_2$ data set

RE-T	TP	FN	FP	TN	Support	Precision	Recall	F1	W.Precision	W.Recall	W.F1
15	137	1	58,792	12,272	71,202	0.0023	0.9928	0.0046	0.9980	0.1743	0.2940
20	134	4	42,156	28,908	71,202	0.0032	0.9710	0.0063	0.9979	0.4079	0.5772
30	130	8	14,972	56,092	71,202	0.0086	0.9420	0.0171	0.9979	0.7896	0.8805
40	123	15	4758	66,306	71,202	0.0252	0.8913	0.0490	0.9979	0.9330	0.9635
60	111	27	1229	69,835	71,202	0.0828	0.8043	0.1502	0.9978	0.9824	0.9895

Abbreviations: FN, false-negative; FP, false-positive; RE, reconstruction error; TN, true-negative; TP, true-positive.

TABLE 6 AE on $\mathcal{D}_2$ data set

RE-T	TP	FN	FP	TN	Support	Precision	Recall	F1	W.Precision	W.Recall	W.F1
10	137	1	54,376	16,688	71,202	0.0025	0.9928	0.00500	0.9980	0.2363	0.3796
15	131	7	24,414	46,650	71,202	0.0053	0.9493	0.0106	0.9979	0.6570	0.7910
20	124	14	7924	63,140	71,202	0.0154	0.8986	0.0303	0.9979	0.8885	0.9391
30	120	18	1965	69,099	71,202	0.0576	0.8696	0.1080	0.9979	0.9721	0.9842
40	106	32	901	70,163	71,202	0.1053	0.7681	0.1852	0.9978	0.9869	0.9918

Abbreviations: FN, false-negative; FP, false-positive; RE, reconstruction error; TN, true-negative; TP, true-positive.

recall, precision, F1 score, and the weighted version of the metrics (W.precision, W.recall, and W.F1 score) were used. Moreover, each table shows a summary confusion matrix, which depicts the true-positives (TP), true-negatives (TN), false-positives (FP), false-negatives (FN), and the total support used. The results based on $\mathcal{D}_1$ and $\mathcal{D}_2$ data sets displayed no significant distinction between the performance of the two unsupervised deep learning models; thus, it does not allow any relative comparison between the two models.

One can observe the sensitivity of the RE thresholds based on $\mathcal{D}_2$ data set, against each metric from Tables 5 and 6. As the RE threshold increases, there is a decrease in the recall and an increase in the precision, which conforms with the earlier analysis. However, an ideal threshold would have to be chosen at the discretion of the insurer based on the cost involved to achieve a recall and precision trade off. The respective RE thresholds for $\mathcal{D}_1$ data set were chosen with no specific criteria; it was chosen so that it would depict a clear image of the impact of thresholds on the metrics to the reader.

It can be claimed that the thresholds for RE are overly sensitive to model performance based on $\mathcal{D}_1$ data set (Tables 3 and 4). Such over sensitivity highlights the RE distribution, consisting of a major concentration on the left tail of the distribution. A plausible explanation would be the smaller data proportion available for both training and testing data. It could be that either the data were of a very poor quality or that data contained a small proportion of fraud data points, thus leading to suboptimal weights, which are not capable of identifying key feature representations. Therefore, $\mathcal{D}_1$ data set tends to perform poorly compared to $\mathcal{D}_2$ data set.

Moreover, the RE threshold analysis shown above can only be conducted if output labels exist. However, one of the primary limitations faced in fraud detection modeling is the lack of output labels. In such a scenario, unlike the supervised models that cannot function whatsoever, unsupervised deep learning models offer an alternative. An insurance company can use the proposed methodology to produce an A-RE distribution, set a threshold toward the tail end of the A-RE distribution, and conduct testing over certain trial periods. For example, they can use ongoing feedback, such as customer complaints and fraud cases uncovered over the course of the trial period, to determine the optimal threshold. Moreover, if the company possesses a small proportion of historical fraudulent cases, they can use it to validate the threshold.

Finally, we present the performance of supervised models on $\mathcal{D}_2$ data set. Compared with $\mathcal{D}_1$ and $\mathcal{D}_3$ data sets, which have limited data volume and unknown output results, respectively, $\mathcal{D}_2$ data set is the most suitable data set to compare the fraud detection performance using different approaches. Before analyzing the results, we should be remindful that, when training the unsupervised deep-learning models, we never make use of the output labels. However, following the definition of supervised models, it is impossible to train a model without having output labels. Thus, supervised models had the benefit of superior quality output labels of $\mathcal{D}_2$ data set.

Analysis of the supervised models on $\mathcal{D}_2$ data set in Table 7 indicates that Random Forest performs the best with the highest F1 score, while Naive Bayes performs the best in capturing recall with only 19 FNs. Putting these results into insurance fraud context would help us understand FNs to be most costly in comparison to FPs, due to the high cost involved, that is, fraudulent cases not identified as fraud are more costly than the auditing cost of nonfraudulent cases being identified as fraud. Unsupervised deep learning models, without having the need for any output labels, manage to achieve excellent results that are most appropriate for an insurer (e.g., AE on $\mathcal{D}_2$ data set achieved only 18 FNs and 1965 FPs out of the total 71,202 test sample). Thus, we find that in realistic settings of having no output labels (supervised models cannot be utilized), unsupervised deep learning models achieve excellent results which yield

TABLE 7 Supervised models on $\mathcal{D}_2$ data set

Model	TP	FN	FP	TN	Support	Precision	Recall	F1	W.Precision	W.Recall	W.F1
K-NN	115	23	7	71,057	71,202	0.9426	0.8333	0.8846	0.9996	0.9996	0.9996
SVM	108	30	6	71,508	71,202	0.9474	0.7826	0.8571	0.9995	0.9995	0.9995
Random Forest	117	21	8	71,056	71,202	0.9360	0.8478	0.8897	0.9996	0.9996	0.9996
Naïve Bayes	119	19	1578	69,486	71,202	0.0701	0.8623	0.1297	0.9979	0.9976	0.9870

Abbreviations: FN, false-negative; FP, false-positive; TN, true-negative; TP, true-positive.

low cost to the insurer. This in turn would yield high quality insights when accompanied with the variable importance methodology proposed in this paper earlier, where supervised variable importance methods are not capable of functioning.

7 | CONCLUDING REMARKS

In conclusion, it appears that the proposed unsupervised deep learning variable importance methodology, relative to supervised variable importance, offers pragmatic insights into the data while also providing exceptional performance in the absence of training output labels. Given a situation where no output labels exist, it provides an excellent base model framework to begin fraud detection until further output label data are collated over the years. Moreover, the ability to capture the dynamic modus operandi of criminals, which precludes insurance fraud in advance, is one of the key advantages of this methodology. Furthermore, the cost effectiveness in doing so at a business setting makes it feasible and appealing from the insurers' perspective. These reasons make the implementation of unsupervised deep learning models for fraud detection more efficient, feasible, and practicable than that of many other models.

The approach proposed in this paper can be extended to other areas of the insurance industry, such as identifying consumer behavior and climate impact on claims, leading to possible future research opportunities. One such potential research area exists in workers compensation insurance in which we can attempt to understand whether an employee has made a claim or not. There are many different factors that affect the ability of an employee to make a total permanent disability claim and stay at home or return to work after a few years. Even with the abundant availability of input data and features, modeling such an event can be problematic, and the ultimate solution to such a problem might be unsupervised deep learning. Similarly, in insurance fraud detection, the output label that the insurer is trying to model is not available in nature because of limited availability of reliable data. In workers' compensation, there is an extended durational impact of the reliability because of the lifetime of human beings. An employee injured in his or her 30's can decide to return to work at age 50, which is unknown until it takes place. This provides the ideal conditions not only to model the claims, but also to gain pragmatic insights from the data lying idle in the database. Using the variable importance procedure proposed in this paper, one is able to study what drives an employee to stay at home after making a claim or return to work after a while. It could even be argued that the vast availability of data and features can hinder one from attempting to model even the simplest of relationships.

Hence, using the VAE or a further variation of the AE/VAE, with the sound technical applications discussed in the paper, the task can be simplified to a matter of running several graphics processing units for training the model. It is shown that the AE and VAE that are initially designed for image recognition can be extended and used in claim modeling and fraud detection. These unsupervised deep learning models can provide new research opportunities for those who aspire to further their studies in actuarial science. A notable window for research would be to unlock the potential of VAEs (Doersch, 2016) and hyperspherical VAEs (Davidson et al., 2018). These models possess some of the most fascinating properties with colossal potential.

Research extensions in claims monitoring in workers compensation schemes and other areas of actuarial application provide a glimpse at the potential of unsupervised deep learning models at a wider scope. Existing applications of these unsupervised deep learning models include artificial image generation and 3D image generation. The breakthrough of these models

would be in exploiting their financial applications and emerging risk management in actuarial work context. The coalescence of actuarial science and data science is instrumental for insurers' profitability and survival in the modern era. Unsupervised deep learning models ameliorate the relationship between policyholders and insurers with efficient claims processing and settlement procedures and better customer tailored products owing to enhanced fraud detection procedures in place. The possible application of unsupervised deep learning models within the financial and insurance industry will be advantageous for the future actuary.

ACKNOWLEDGMENTS

This study was supported in part by the Research Grants Council of the Hong Kong Special Administrative Region (project no. 17304921) and by a Faculty Research Grant from The University of Melbourne.

REFERENCES

An, J., & Cho, S. (2015). Variational autoencoder based anomaly detection using reconstruction probability. In *Special Lecture on IE* (pp. 1–18).

Arenz, O., Zhong, M., & Neumann, G. (2020). Trust-region variational inference with Gaussian mixture models. *Journal of Machine Learning Research*, *21*(163), 1–60.

Artís, M., Ayuso, M., & Guillén, M. (2002). Detection of automobile insurance fraud with discrete choice models and misclassified claims. *Journal of Risk and Insurance*, *69*(3), 325–340.

Artís, M., Ayuso, M., & Guillén, M. (1999). Modelling different types of automobile insurance fraud behaviour in the Spanish market. *Insurance: Mathematics and Economics*, *24*(1-2), 67–81.

Baldi, P. (2012). Autoencoders, unsupervised learning, and deep architectures. In *Proceedings of ICML workshop on unsupervised and transfer learning* (pp. 37–29).

Chalapathy, R., & Chawla, S. (2019). Deep learning for anomaly detection: A survey. *arXiv preprint. arXiv:1901.03407*.

Davidson, T. R., Falorsi, L., De Cao, N., Kipf, T., & Tomczak, J. M. (2018). Hyperspherical variational auto-encoders. *arXiv preprint. arXiv:1804.00891*.

Derrig, R. A. (2002). Insurance fraud. *Journal of Risk and Insurance*, *69*(3), 271–287.

Dionne, G., Giuliano, F., & Picard, P. (2009). Optimal auditing with scoring: Theory and application to insurance fraud. *Management Science*, *55*(1), 58–70.

Doersch, C. (2016). Tutorial on variational autoencoders. *arXiv preprint. arXiv:1606.05908*.

Ekin, T., Ieva, F., Ruggeri, F., & Soyer, R. (2018). Statistical medical fraud assessment: Exposition to an emerging field. *International Statistical Review*, *86*(3), 379–402.

Ekin, T., Lakomski, G., & Musal, R. M. (2019). An unsupervised Bayesian hierarchical method for medical fraud assessment. *Statistical Analysis and Data Mining: The ASA Data Science Journal*, *12*(2), 116–124.

Fawcett, T., & Provost, F. (1997). Adaptive fraud detection. *Data Mining and Knowledge Discovery*, *1*(3), 291–316.

Gill, K. M., Woolley, A., & Gill, M. (2005). Insurance fraud: The business as a victim? In *Crime at work* (pp. 73–82). Palgrave MacMillan.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT Press.

Hastie, T., Tibshirani, R., & Friedman, J. (2009). Unsupervised learning. In *The elements of statistical learning* (pp. 485–585). Springer.

Helmy, A. A., Omar, Y. M., & Hodhod, R. (2018). An innovative word encoding method for text classification using convolutional neural network. In *2018 14th international computer engineering conference (ICENCO)* (pp. 42–47). IEEE.

Ioffe, S., & Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning* (pp. 448–456). PMLR.

Jain, A. K., Murty, M. N., & Flynn, P. J. (1999). Data clustering: A review. *ACM Computing Surveys (CSUR)*, *31*(3), 264–323.

Khatri, S., Arora, A., & Agrawal, A. (2020). Supervised machine learning algorithms for credit card fraud detection: A comparison. *IEEE, 10th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*, 680–683.

Kemp, G. (2010). Fighting public sector fraud in the 21st century. *Computer Fraud & Security*, *2010*(11), 16–28.

Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint. arXiv:1412.6980*.

Kingma, D. P., & Welling, M. (2013). Auto-encoding variational bayes. *arXiv preprint. arXiv:1312.6114*.

LeCun, Y., Touresky, D., Hinton, G., & Sejnowski, T. (1988). A theoretical framework for back-propagation. In *Proceedings of the 1988 connectionist models summer school* (Vol. 1, pp. 21–28). Morgan Kaufmann.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*(7553), 436–444.

Li, J., Huang, K. Y., Jin, J., & Shi, J. (2008). A survey on statistical methods for health care fraud detection. *Health Care Management Science*, *11*(3), 275–287.

Nguyen, T., & Perez, V. (2020). Privatizing plaintiffs: How medicaid, the false claims act, and decentralized fraud detection affect public fraud enforcement efforts. *Journal of Risk and Insurance*, *87*(4), 1063–1091.

Nian, K., Zhang, H., Tayal, A., Coleman, T., & Li, Y. (2016). Auto insurance fraud detection using unsupervised spectral ranking for anomaly. *The Journal of Finance and Data Science*, *2*(1), 58–75.

Niu, X., Wang, L., & Yang, X. (2019). A comparison study of credit card fraud detection: Supervised versus unsupervised. *arXiv preprint. arXiv:1904.10604*.

Noble, C. C., & Cook, D. J. (2003). Graph-based anomaly detection. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 631–636).

Palacio, S. (2018). "Outlier Detection," Mendeley Data, V2, *doi: 10.17632/g3vxppc8k4.2*.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ...Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, *12*, 2825–2830.

Phua, C., Lee, V., Smith, K., & Gayler, R. (2010). A comprehensive survey of data mining-based fraud detection research. *arXiv preprint. arXiv:1009.6119*.

Rezende, D. J., Mohamed, S., & Wierstra, D. (2014). Stochastic backpropagation and approximate inference in deep generative models. In *International conference on machine learning* (pp. 1278–2286). PMLR.

Sharma, A., & Panigrahi, P. K. (2013). A review of financial accounting fraud detection based on data mining techniques. *arXiv preprint. arXiv:1309.3944*.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, *15*(1), 1929–1958.

Srivastava, N., & Salakhutdinov, R. (2014). Multimodal learning with deep Boltzmann machines. *Journal of Machine Learning Research*, *15*(1), 2949–2980.

Van Engelen, J. E., & Hoos, H. H. (2020). A survey on semi-supervised learning. *Machine Learning*, *109*(2), 373–440.

Wilhelm, W. K. (2004). The fraud management lifecycle theory: A holistic approach to fraud management. *Journal of Economic Crime Management*, *2*(2), 1–38.

Zafari, B., & Ekin, T. (2019). Topic modelling for medical prescription fraud and abuse detection. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, *68*(3), 751–769.

Zheng, J., & Peng, L. (2018). An autoencoder-based image reconstruction for electrical capacitance tomography. *IEEE Sensors Journal*, *18*(13), 5464–5474.

How to cite this article: Gomes, C., Jin, Z., & Yang, H. (2021). Insurance fraud detection with unsupervised deep learning. *J Risk Insur*. 88, 591–624. https://doi.org/10.1111/jori.12359