

Anomaly Detection in Financial Data Using GA-Optimized MLP Models

1st Given Name Surname
dept. name of organization (of Aff.)
name of organization (of Aff.)
City, Country
email address or ORCID

2nd Given Name Surname
dept. name of organization (of Aff.)
name of organization (of Aff.)
City, Country
email address or ORCID

3rd Given Name Surname
dept. name of organization (of Aff.)
name of organization (of Aff.)
City, Country
email address or ORCID

4th Given Name Surname
dept. name of organization (of Aff.)
name of organization (of Aff.)
City, Country
email address or ORCID

5th Given Name Surname
dept. name of organization (of Aff.)
name of organization (of Aff.)
City, Country
email address or ORCID

6th Given Name Surname
dept. name of organization (of Aff.)
name of organization (of Aff.)
City, Country
email address or ORCID

Abstract—This study explores the application of genetic algorithm (GA)-optimized multi-layer perceptron (MLP) models for accurately classifying binary outcomes in anomaly detection within financial datasets, which are prone to poor target feature separability. By leveraging GA, we synthesize MLP architectures tailored to maximize precision while maintaining recall on a validation set that was not part of the training data. Experimental results demonstrate that GA-driven optimization improves the precision of anomaly detection compared to traditional regression-based methods. This approach offers a promising framework for enhancing the security and reliability of financial systems against sophisticated fraudulent behaviors and anomalies.

Index Terms—binary classification, genetic algorithm, anomaly detection, imbalanced data

I. INTRODUCTION

The detection of anomalies in financial data has become a pressing concern, necessitating the development of effective and accurate solutions. This study specifically targets the detection of the need for tax inspections among legal entities based on a dataset that includes financial statements, financial reporting attributes, and details about the executives of these organizations. Importantly, this dataset presents inherent challenges due to its initial poor separability and the imbalance of its target feature. Correctly classifying the target feature in such data is indeed the search for anomalies.

Traditional approaches, such as regression analysis, have proven insufficient [1] in handling high-dimensional and complex data, particularly when the target feature lacks clear distinguishing characteristics [2]. Neural network (NN)-based approaches, specifically, multi-layer perceptrons (MLPs), have shown effectiveness [3] in classification. However, optimizing the architecture and parameters of MLPs can be a complex task, particularly without employing optimization techniques. One powerful tool for finding global optima in complex parameter spaces is the Genetic Algorithm (GA) [4]. Inspired

by natural selection and evolution, GA have proven their efficacy across various scientific and engineering domains.

This study explores various regression models for binary classification to identify the most suitable approach for detecting anomalies in the target feature, specifically the need for tax authority audits, within a financial dataset of legal entities. The dataset is characterized by poor separability and class imbalance of the target feature. After determining the best-performing regression model, namely the MLP, GA were applied to optimize MLP’s hyperparameters to maximize model precision while maintaining recall on a validation set not used during training. Experimental results demonstrate that GA-driven optimization yields substantial improvements compared to traditional regression-based methods.

Put simply, the paper is structured as follows: Section II reviews related work in the field, providing context. Section III details our dataset and employed methodology. Section IV presents experimental results and their implications. In Section V discusses future research directions.. Finally, Section VI, we draw conclusions from our findings.

II. RELATED WORKS

Various approaches have been proposed for anomaly detection in finance, each with its strengths and limitations. To provide a comprehensive overview, we summarize the key works in the field in the Table I.

Anomaly detection is crucial in finance for identifying fraudulent activities. Ahmed et al. [6] survey reviews clustering-based unsupervised anomaly detection techniques, comparing them from different perspectives. A key challenge is the lack of real-world data, with synthetic data often used for validation. However, on the other hand, anomaly detection in financial data has been largely overlooked. In study [7] authors apply standard anomaly detection techniques (nearest-neighbours, clustering, and statistical approaches) to historical daily trading data to detect rare anomalies. The results show

TABLE I
SUMMARY OF WORKS ON ANOMALY DETECTION IN FINANCE

Reference	Focus	Limitations
Guo et al. (2015) [5]	Combining activity and density for time series anomaly detection	Limited to financial time series data, neglects individual outliers
Ahmed et al. (2016) [6]	Clustering-based unsupervised anomaly detection	Lack of real-world data, reliance on synthetic data
Ahmed et al. (2017) [7]	Standard anomaly detection techniques (nearest-neighbours, clustering, statistical approaches)	Limited to historical daily trading data
Huang et al. (2018) [8]	CoDetect framework (network and feature information)	Limited to financial fraud detection
Zhang et al. (2022) [9]	Random forest algorithm for anomaly detection	Requires manual updates for IDS to function properly
Chen et al. (2022) [10]	AntiBenford subgraph framework (statistical principles)	Limited to cryptocurrency transaction networks
Crepey et al. (2022) [11]	PCA and feedforward neural network for anomaly detection	Limited to financial time series data

that LOF (Local Outlier Factor) [12] and CMGOS (Clustering-based Multivariate Gaussian Outlier Score) [13] are the best-performing algorithms.

The rapid growth of computer networks brings convenience, but also security concerns due to abnormal flows. Current detection systems, like intrusion detection system (IDS) [14], have limitations, requiring manual updates to function properly. Zhang et al. [15] suggests an approach using random forest algorithm to detect abnormal samples in financial data, measuring their degree of abnormality based on similarity. Simulation results show that this method outperforms other distance-based techniques in terms of accuracy and computing time.

Financial fraud, such as money laundering, is a serious crime that involves complex networks of transactions. Existing methods focus on either network or feature information, but not both. Huang et al. [8] proposes CoDetect, a framework that leverages both network and feature information for financial fraud detection and can detect fraud activities and identify associated feature patterns. Experiments on synthetic and real-world data demonstrate its efficiency and effectiveness in combating financial fraud.

Chen et al. [10] introduce the AntiBenford subgraph framework, based on statistical principles, to detect anomalies in cryptocurrency transaction networks. This algorithm finds AntiBenford subgraphs [16] in near-linear time. Evaluations on real and synthetic data show that our framework outperforms state-of-the-art methods, detecting previously undetected anomalous subgraphs and providing new insights into financial transaction data.

Anomalies in financial time series can lead to miscalibrated risk models and erroneous risk measures. Crepey et al. [11] proposes an approach that extracts valuable features using principal component analysis (PCA) and detects anomalies using a feedforward neural network. The anomaly score is calibrated through a customized loss function, and the approach is shown to outperform existing algorithms on synthetic and real data sets. By using this approach with a basic imputation method, value-at-risk estimation errors are significantly

reduced.

Financial data is increasingly variable and unpredictable, with abnormal fluctuations containing important information. Traditional time series anomaly detection methods focus on individual outliers, neglecting the time sequence and sub-sequences. Guo et al. [5] proposes a method that combines activity and density to effectively utilize time sequence and sub-sequences features, discovering anomalies in financial time series data.

A knowledge gap exists in the literature on using optimization techniques, like GA to tune classification models for anomaly detection in financial data. This gap is particularly significant for tax authority audits, where efficient anomaly detection can reduce auditors' workload. Our article aims to bridge this gap

III. MATERIALS AND METHODS

A. Dataset Description

The dataset consisted of 10,000 records, each corresponding to an organization previously scrutinized by tax authorities. It encompassed 37 columns, including 10 binary features and 26 quantitative features related to these binaries. The target variable indicated the necessity of an audit. Notably, 36 features were indirectly known to tax authorities and were not outcomes of prior audits. Hence, the goal was to construct models that effectively minimized unnecessary audits based on these indirect indicators. Figure 1 presents histograms illustrating the distributions of binary features, essential for characterizing various aspects of legal entities

The dataset's binary features contains the Account Status feature (1b) which indicates the presence or absence of accounts associated with these entities. The High-Risk Flag (1b) identifies entities categorized as high-risk, potentially due to financial or operational vulnerabilities. Additionally, the Tax Compliance Flag (1c) flags entities known for tax avoidance or non-compliance issues. Moreover, it includes features that capture disparities in leadership and ownership structures. The Regional Leadership Inequality Flag (1d) indicates discrepancies in executive positions across regions within the

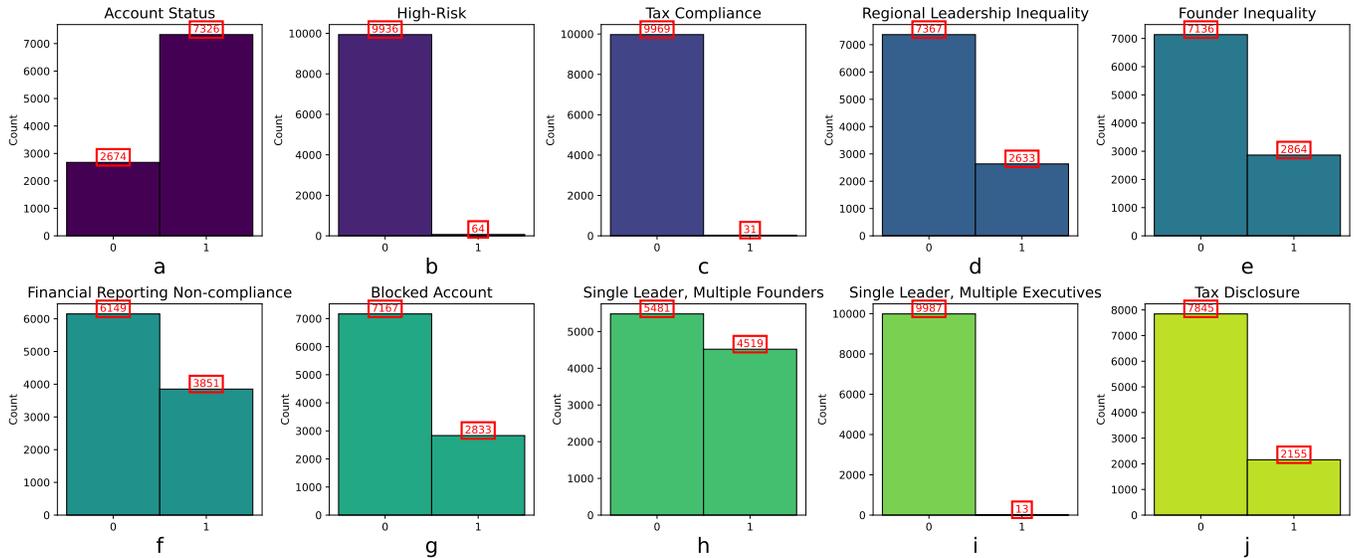


Fig. 1. Histograms of features from financial dataset with 10,000 entities correspond to: (a) Account Status, (b) High-Risk Flag, (c) Tax Compliance Flag, (d) Regional Leadership Inequality Flag, (e) Founder Inequality Flag, (f) Financial Reporting Non-compliance Flag, (g) Blocked Account Flag, (h) Single Leader, Multiple Founders Flag, (i) Single Leader, Multiple Executives Flag, (j) Tax Disclosure Flag.

organization, while the Founder Inequality Flag (1e) highlights variations in ownership among founders across regions. Furthermore, features related to financial reporting and compliance are present in the dataset. The Financial Reporting Non-compliance Flag (1f) identifies entities with inadequate or absent financial reporting practices, and the Blocked Account Flag (1g) indicates entities with currently blocked accounts. Additionally, the dataset includes features that depict the leadership and ownership structures of entities. The Single Leader, Multiple Founders Flag (1h) identifies entities with a single leader and multiple founders, while the Single Leader, Multiple Executives Flag (1i) highlights organizations led by a single leader with multiple executives. Finally, the Tax Disclosure Flag (1j) indicates entities providing tax disclosures, potentially influencing financial transparency.

The t-Distributed Stochastic Neighbor Embedding (t-SNE) and Principal Component Analysis (PCA) visualizations (Figure 2) depict the challenges posed by the poor separability of data in our dataset. After reducing the dataset, which excludes the target feature, to 2 and 3 dimensions using t-SNE and PCA, respectively, and labeling it according to the target feature, we observe significant overlap among data points. This overlap illustrates the intricate nature of the dataset, where distinguishing between classes becomes a non-trivial task for binary classification. The visual representation underscores the complexities involved in achieving effective separation and classification of entities based on their characteristics

B. Regression-based binary classification

Figure 3 illustrates our approach to selecting the optimal classifier architecture and therefore hyperparameters for a

given task, leveraging a combination of classifier evaluation metrics and GA-based optimization.

The first goal of the study was to employ various classifiers to maximize precision for the target class while maintaining a recall below 0.3 [17]. This strategy aims to identify risky legal entities with minimal false positives.

Using diverse classification methods [18] enhances modeling and prediction. Logistic Regression [19] handles linear dependencies, while Random Forest [20] manages nonlinear relationships and avoids overfitting. Adjusting classification thresholds balances precision and recall. Oversampling or undersampling [21] addresses class imbalances, and feature engineering [22] improves model interpretability. Algorithm selection [23] and ensemble methods combine multiple models for better performance.

Cost-sensitive learning [24] and customizing class weights [25] are crucial when misclassifications have distinct consequences. Advanced techniques like Meta-learning [26], XGBoost [27], LightGBM [28], CatBoost [29], and AdaBoost [30] improve classification performance. Voting ensembles [31] leverage diverse models for robust results.

Advanced methods such as Bayesian Optimization [32] and Deep Neural Networks (DNN) [33] further enhance performance. Bayesian Optimization refines hyperparameters for optimal settings, and DNNs capture complex patterns, making them powerful for binary classification. Integrating these techniques offers a sophisticated approach to improving classification accuracy.

We applied 50 different classifiers to our dataset to identify the most optimal one for our task. Following standard methodology, we used built-in classification modules in Python. Metrics were evaluated on a test set, which comprised 20

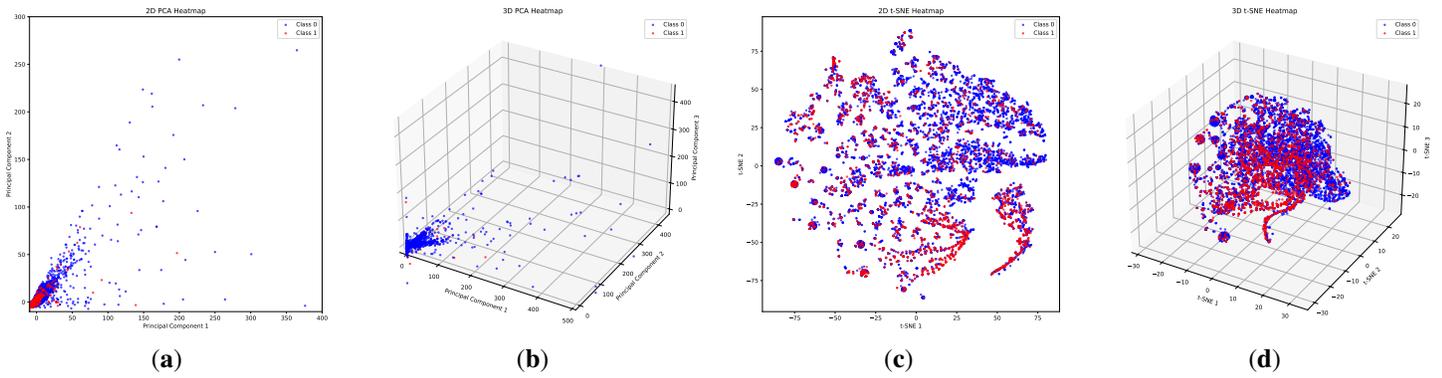


Fig. 2. (a) 2D PCA Heatmap and (b) 3D PCA Heatmaps; (c) 2D t-SNE Heatmap and (d) 3D t-SNE Heatmaps: PCA and t-SNE were used for dimensionality reduction to two and three dimensions. Both PCA and t-SNE visualizations illustrate the poor separability of the data points based on the target labels. The data points are colored according to the binary target labels, where blue represents Class 0 and red represents Class 1.

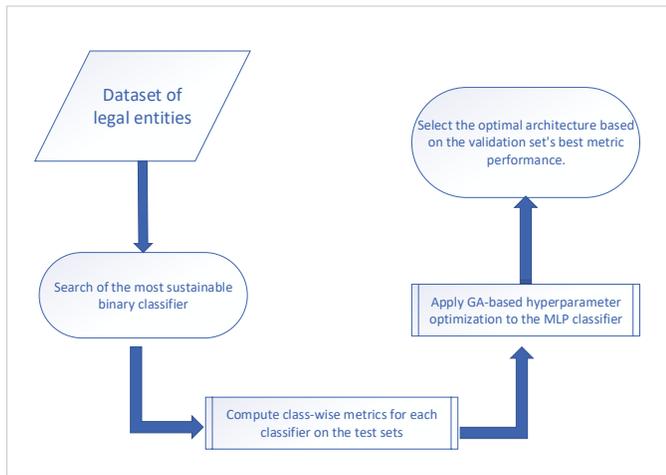


Fig. 3. Schematic representation of the proposed approach, illustrating the steps involved in classifier selection, hyperparameter tuning, and evaluation on the holdout set

C. GA-based optimizing MLP models

The next aim of this research was to utilize GA optimization to synthesize MLP models aimed at maximizing precision while maintaining recall for binary classification of a dataset concerning legal entities. The primary objective was to reduce unnecessary tax audits conducted by tax authorities. The selection of MLP was based on its performance as the model that achieved the best metrics.

The dataset was divided into training, testing, and validation sets using a stratified split with respective sizes of 70%, 20%, and 10%, and a fixed random state of 42. Features were standardized using StandardScaler from scikit-learn.

The hyperparameter space included the following parameters: activation function (identity, logistic, tanh, or relu), solver (adam or sgd), alpha (0.0001, 0.001, 0.01, or 0.1), and hidden layer sizes (ranging from 2 to 10 layers with 2 to 128 neurons per layer).

GA implementation was carried out using the DEAP library in Python. Each individual in the GA was defined as a

list of four elements: hidden layer sizes, activation function, solver, and alpha. The fitness function was defined as the validation accuracy of the neural network. The GA operated with a population size of 20 over 10 generations, employing a crossover probability of 0.5 and a mutation probability of 0.2.

IV. RESULTS

Fig. 4 illustrates the precision, recall, and accuracy values across different experiments. The accuracy remains below 0.8, underscoring the challenging the classification task of imbalanced data with poor separability of target feature. The recall for the target feature (1) fluctuates between 0.2 and 0.4, indicating the model's ability to capture a portion of positive instances. Precision for the target feature (1) hovers around 0.6, reflecting the balance between correctly identified positive instances and false positives. These trends provide insights into the model's performance and highlight areas for potential improvement in achieving a more balanced classification outcome.

Over several generations, GA consistently demonstrated an incremental improvement in precision on the validation dataset (Fig. 5). Initial generations exhibited diversity in the models produced; however, as the number of generations increased, models achieving high precision became predominant. The application of GA facilitated a substantial enhancement in precision compared to baseline models, thereby confirming the effectiveness of the approach.

In the ordered precision growth chart of the validation dataset, we observe an increase in precision from approximately 0.4 to 0.54. However, test precision displays fluctuations ranging between 0.45 and 0.62. Concurrently, accuracy metrics on both test and validation datasets exhibit similar patterns, hovering around 0.8. Meanwhile, the recall metric remains consistently low, typically averaging below 0.18.

The systematic rise in validation precision suggests that models generated by the GA are becoming more adept at correctly identifying positive cases (true positives) relative to all positive predictions (true positives + false positives). This underscores the GA's efficacy in optimizing model parameters for enhanced performance on validation data.

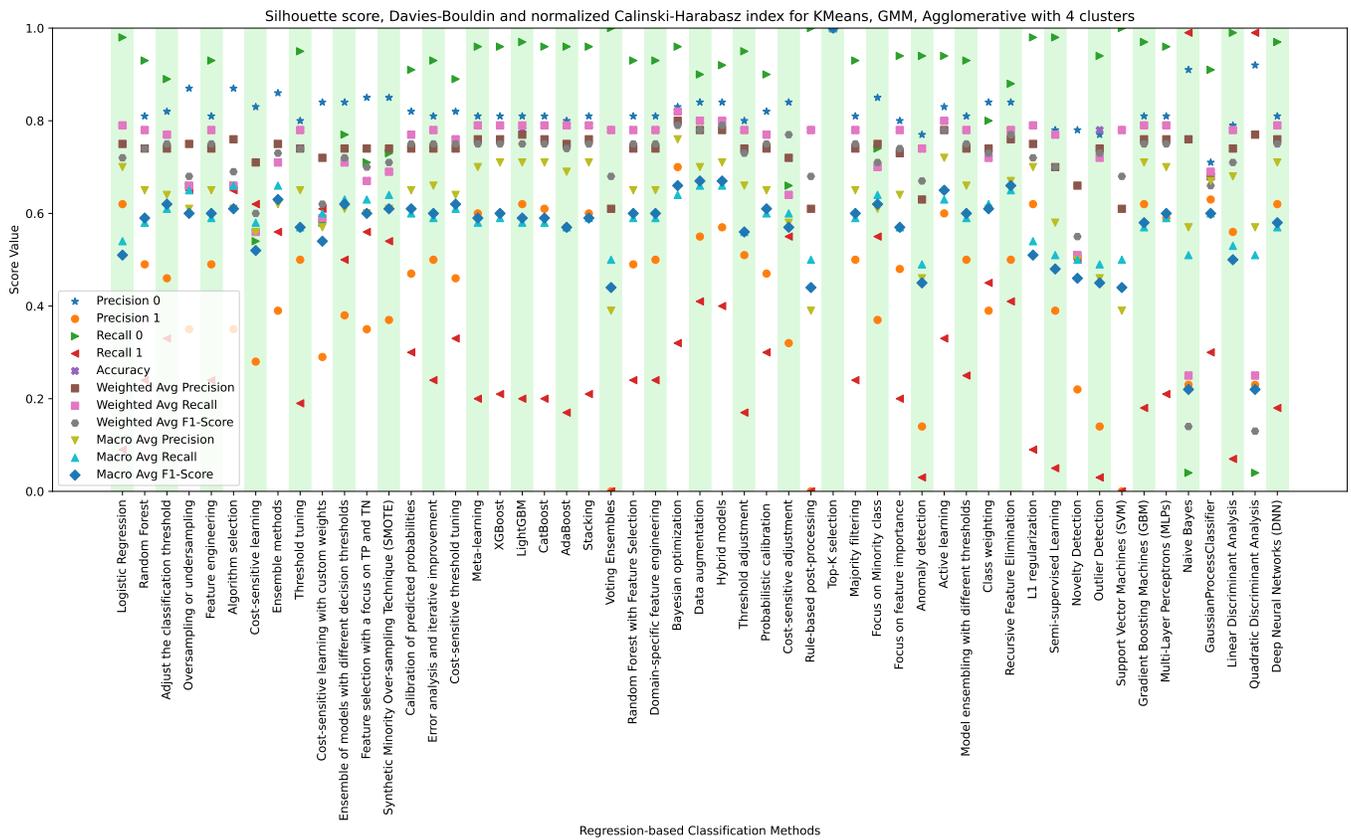


Fig. 4. Class-wise metrics for target feature (0 - no need for inspection, 1 - inspection required.) for various binary classifiers

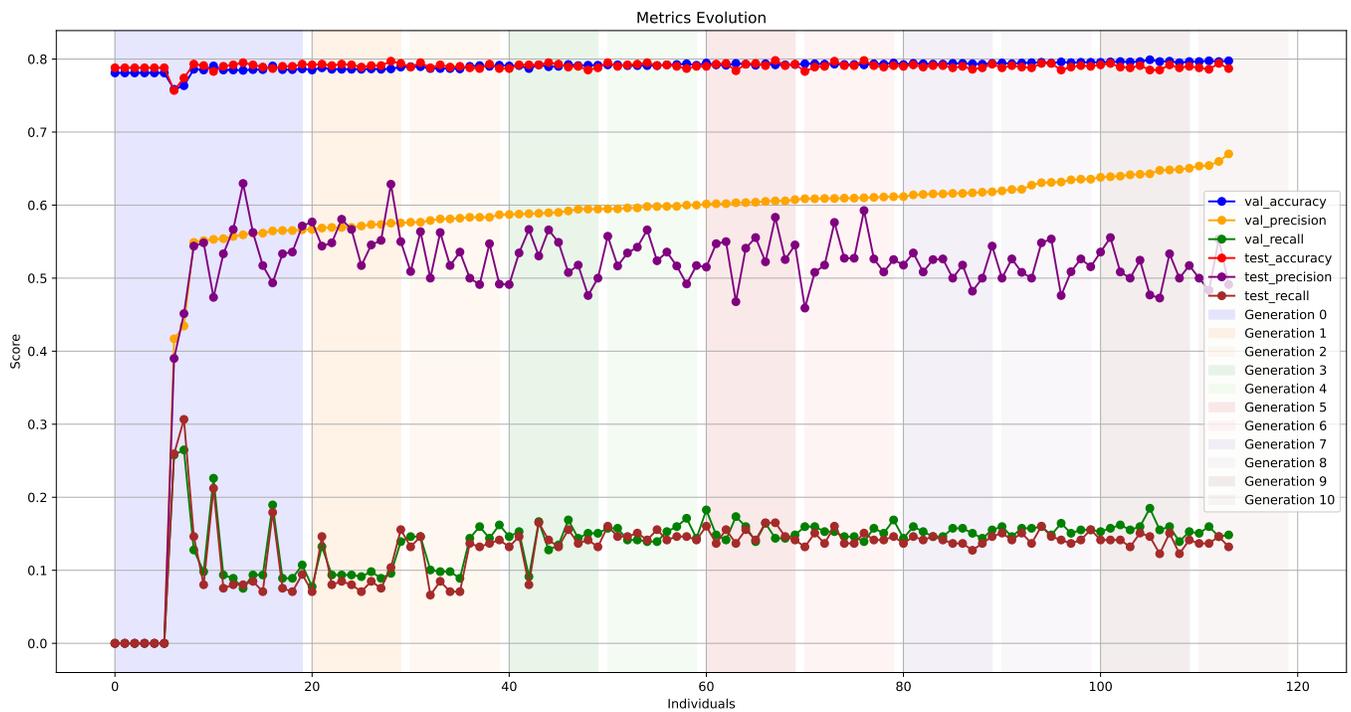


Fig. 5. Evolution of Precision Metrics on Validation and Test Datasets for GA-optimized MLP models.

Fluctuations in both test and validation precision can be attributed to several factors. Overfitting may occur when models tailored to the validation set struggle to generalize to unseen test data, leading to varying performance levels. Furthermore, disparities in the distribution or characteristics of the validation and test datasets can impact model performance differentially. Additionally, the complexity of GA-evolved models may contribute to performance variability across datasets.

The consistent behavior of accuracy metrics, averaging around 0.8 on both datasets, indicates reliable performance in terms of correct predictions (true positives + true negatives) relative to all predictions.

However, the relatively low values of recall, averaging less than 0.18, suggest that the models are less effective in capturing all positive cases, particularly true positives, compared to all actual positive instances. This imbalance may reflect a prioritization of precision over recall, potentially influenced by class imbalance or specific characteristics of the problem domain.

The optimal architectures with neurons per each layer, activation function (AF) for all layer, and hyperparameters for metrics validation accuracy (VA) and validation precision (VP) and are presented in Table II.

TABLE II
TOP-PERFORMING MODELS

Layers	AF	Solver	α	VA	VP
(22, 55, 124, 82, 121, 79, 41)	tanh	sgd	0.001	0.799	0.6429
(62, 82, 87, 54)	tanh	sgd	0.1	0.7975	0.6701
(62, 82, 87, 54)	tanh	sgd	0.001	0.7975	0.6542
(109, 20, 95, 43, 56)	tanh	sgd	0.1	0.7945	0.6195
(45, 115, 53)	relu	sgd	0.0001	0.796	0.6415

V. DISCUSSION

The results of this study open up several avenues for future research in the area of anomaly detection. For instance, future work could explore the application of GA-based optimization to multi-class anomaly detection tasks [34]. This approach could be particularly useful when, besides determining the need for a tax audit, it is necessary to identify other unknown indicators before an audit, such as potential money laundering activities and offshore transfers based on indirect signs.

Comparison with other optimization algorithms, such as particle swarm optimization [35] or Bayesian optimization [36], is also warranted to evaluate their effectiveness relative to GA. Additionally, applying the proposed approach to other anomaly detection tasks, such as fraud detection, network intrusion detection, or medical diagnosis, could further validate its versatility and robustness.

The integration of GA with other machine learning techniques, such as ensemble methods [37] or transfer learning [38], should not be overlooked. As anomaly detection datasets continue to grow in size and complexity, future research could focus on developing scalable and parallelized versions of the proposed approach to handle large-scale datasets [39]. Moreover, research could also aim at developing

techniques to provide deeper insights into the decision-making process of the optimized models, enhancing their interpretability and trustworthiness.

VI. CONCLUSIONS

In this study, we explored the application of various classifiers for binary classification tasks burned with lack separability and imbalanced in target feature. Our results demonstrate that the choice of classifier can significantly impact the performance of the model. Specifically, we found that the MLP classifier outperformed other classifiers, such as logistic regression and decision trees, in terms of accuracy and precision.

GA has been shown to be a powerful tool for solving complex optimization problems, and our study highlights its potential for optimizing hyperparameters in anomaly detection tasks. By leveraging the GA, we can identify optimal hyperparameters that result in improved performance and accuracy, leading to more effective anomaly detection.

Overall, our study highlights the importance of considering multiple classifiers, optimizing hyperparameters, and leveraging powerful optimization algorithms, such as the GA, for solving anomaly detection problems. By combining these approaches, we can develop more accurate and reliable models that can be applied to a wide range of anomaly detection tasks, particularly from fraud detection in financial data.

APPENDIX

Appendix A]Data Availability All code, datasets, and images referenced in this article are publicly available in the following GitHub repository: catauggie/AnomalyData (accessed on 2024-06-15). Researchers are encouraged to refer to this repository for access to the complete set of resources used in the study.

REFERENCES

- [1] S. Thudumu, P. Branch, J. Jin, and J. Singh, "A comprehensive survey of anomaly detection techniques for high dimensional big data," *Journal of Big Data*, vol. 7, pp. 1–30, 2020.
- [2] S. S. Noureen, S. B. Bayne, E. Shaffer, D. Porschet, and M. Berman, "Anomaly detection in cyber-physical system using logistic regression analysis," in *2019 IEEE Texas Power and Energy Conference (TPEC)*. IEEE, 2019, pp. 1–6.
- [3] M. Egmont-Petersen, J. L. Talmon, A. Hasman, and A. W. Ambergen, "Assessing the importance of features for multi-layer perceptrons," *Neural networks*, vol. 11, no. 4, pp. 623–635, 1998.
- [4] A. Lambora, K. Gupta, and K. Chopra, "Genetic algorithm-a literature review," in *2019 international conference on machine learning, big data, cloud and parallel computing (COMITCon)*. IEEE, 2019, pp. 380–384.
- [5] C.-M. Guo, L.-Y. Xu, H.-F. Liu, L. Wang, X. Yu, and B. Han, "The financial data of anomaly detection research based on time series," in *2015 International Conference on Computer Science and Applications (CSA)*. IEEE, 2015, pp. 86–89.
- [6] M. Ahmed, A. N. Mahmood, and M. R. Islam, "A survey of anomaly detection techniques in financial domain," *Future Generation Computer Systems*, vol. 55, pp. 278–288, 2016.
- [7] M. Ahmed, N. Choudhury, and S. Uddin, "Anomaly detection on big data in financial markets," in *Proceedings of the 2017 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2017*, 2017, pp. 998–1001.
- [8] D. Huang, D. Mu, L. Yang, and X. Cai, "Codetect: Financial fraud detection with anomaly feature detection," *IEEE Access*, vol. 6, pp. 19 161–19 174, 2018.

- [9] C. Zhang, C. Liu, X. Zhang, and G. Almpandis, "An up-to-date comparison of state-of-the-art classification algorithms," *Expert Systems with Applications*, vol. 82, pp. 128–150, 2017.
- [10] T. Chen and C. Tsourakakis, "Antibenford subgraphs: Unsupervised anomaly detection in financial networks," in *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2022, pp. 2762–2770.
- [11] S. Crépey, N. Lehdili, N. Madhar, and M. Thomas, "Anomaly detection in financial time series by principal component analysis and neural networks," *Algorithms*, vol. 15, no. 10, p. 385, 2022.
- [12] O. Alghushairy, R. Alsini, T. Soule, and X. Ma, "A review of local outlier factor algorithms for outlier detection in big data streams," *Big Data and Cognitive Computing*, vol. 5, no. 1, p. 1, 2020.
- [13] M. Muhammad, U. Daniel Ani, A. A. Abdullahi, and P. Radanliev, "Device-type profiling for network access control systems using clustering-based multivariate gaussian outlier score," in *The 5th International Conference on Future Networks & Distributed Systems*, 2021, pp. 270–279.
- [14] M. Pradhan, C. K. Nayak, and S. K. Pradhan, "Intrusion detection system (ids) and their types," in *Securing the internet of things: Concepts, methodologies, tools, and applications*. IGI Global, 2020, pp. 481–497.
- [15] Q. Zhang, "Financial data anomaly detection method based on decision tree and random forest algorithm," *Journal of Mathematics*, vol. 2022, no. 1, p. 9135117, 2022.
- [16] R. K. Somkunwar, M. P. Pimpalkar, K. M. Katakound, A. S. Bhide, S. P. Chinchalkar, and Y. M. Patil, "A fraud detection system in financial networks using antibenford subgraphs and machine learning algorithms," in *2023 International Conference on Ambient Intelligence, Knowledge Informatics and Industrial Electronics (AIKIIE)*. IEEE, 2023, pp. 1–6.
- [17] T. Saito and M. Rehmsmeier, "The precision-recall plot is more informative than the roc plot when evaluating binary classifiers on imbalanced datasets," *PLoS one*, vol. 10, no. 3, p. e0118432, 2015.
- [18] A. Elatawneh, C. Kalaitzidis, G. P. Petropoulos, and T. Schneider, "Evaluation of diverse classification approaches for land use/cover mapping in a mediterranean region utilizing hyperion data," *International Journal of Digital Earth*, vol. 7, no. 3, pp. 194–216, 2014.
- [19] M. P. LaValley, "Logistic regression," *Circulation*, vol. 117, no. 18, pp. 2395–2399, 2008.
- [20] M. Pal, "Random forest classifier for remote sensing classification," *International journal of remote sensing*, vol. 26, no. 1, pp. 217–222, 2005.
- [21] R. Mohammed, J. Rawashdeh, and M. Abdullah, "Machine learning with oversampling and undersampling techniques: overview study and experimental results," in *2020 11th international conference on information and communication systems (ICICS)*. IEEE, 2020, pp. 243–248.
- [22] C. R. Turner, A. Fuggetta, L. Lavazza, and A. L. Wolf, "A conceptual basis for feature engineering," *Journal of Systems and Software*, vol. 49, no. 1, pp. 3–15, 1999.
- [23] J. R. Rice, "The algorithm selection problem," in *Advances in computers*. Elsevier, 1976, vol. 15, pp. 65–118.
- [24] C. Elkan, "The foundations of cost-sensitive learning," in *International joint conference on artificial intelligence*, vol. 17, no. 1. Lawrence Erlbaum Associates Ltd, 2001, pp. 973–978.
- [25] A. Aue and M. Gamon, "Customizing sentiment classifiers to new domains: A case study," in *Proceedings of recent advances in natural language processing (RANLP)*, vol. 1, no. 3.1, 2005, pp. 2–1.
- [26] J. Vanschoren, "Meta-learning," *Automated machine learning: methods, systems, challenges*, pp. 35–61, 2019.
- [27] Z. E. Aydin and Z. K. Ozturk, "Performance analysis of xgboost classifier with missing data," *Manchester Journal of Artificial Intelligence and Applied Sciences (MJAIAS)*, vol. 2, no. 02, p. 2021, 2021.
- [28] B. Wang, Y. Wang, K. Qin, and Q. Xia, "Detecting transportation modes based on lightgbm classifier from gps trajectory data," in *2018 26th International Conference on Geoinformatics*. IEEE, 2018, pp. 1–7.
- [29] A. A. Ibrahim, R. L. Ridwan, M. M. Muhammed, R. O. Abdulaziz, and G. A. Saheed, "Comparison of the catboost classifier with other machine learning methods," *International Journal of Advanced Computer Science and Applications*, vol. 11, no. 11, 2020.
- [30] T.-K. An and M.-H. Kim, "A new diverse adaboost classifier," in *2010 International conference on artificial intelligence and computational intelligence*, vol. 1. IEEE, 2010, pp. 359–363.
- [31] L. I. Kuncheva and J. J. Rodríguez, "A weighted voting framework for classifiers ensembles," *Knowledge and information systems*, vol. 38, pp. 259–275, 2014.
- [32] P. I. Frazier, "Bayesian optimization," in *Recent advances in optimization and modeling of contemporary problems*. Informs, 2018, pp. 255–278.
- [33] Y. Geifman and R. El-Yaniv, "Selective classification for deep neural networks," *Advances in neural information processing systems*, vol. 30, 2017.
- [34] I. S. Thaseen, A. K. Chitturi, F. Al-Turjman, A. Shankar, M. R. Ghalib, and K. Abhishek, "An intelligent ensemble of long-short-term memory with genetic algorithm for network anomaly identification," *Transactions on Emerging Telecommunications Technologies*, vol. 33, no. 10, p. e4149, 2022.
- [35] D. Wang, D. Tan, and L. Liu, "Particle swarm optimization algorithm: an overview," *Soft computing*, vol. 22, pp. 387–408, 2018.
- [36] B. Shahriari, K. Swersky, Z. Wang, R. P. Adams, and N. De Freitas, "Taking the human out of the loop: A review of bayesian optimization," *Proceedings of the IEEE*, vol. 104, no. 1, pp. 148–175, 2015.
- [37] M. Dostmohammadi, M. Z. Pedram, S. Hoseinzadeh, and D. A. Garcia, "A ga-stacking ensemble approach for forecasting energy consumption in a smart household: A comparative study of ensemble methods," *Journal of Environmental Management*, vol. 364, p. 121264, 2024.
- [38] A. Farahani, B. Pourshojae, K. Rasheed, and H. R. Arabnia, "A concise review of transfer learning," in *2020 international conference on computational science and computational intelligence (CSCI)*. IEEE, 2020, pp. 344–351.
- [39] J. Chen, K. Li, K. Bilal, K. Li, S. Y. Philip *et al.*, "A bi-layered parallel training architecture for large-scale convolutional neural networks," *IEEE transactions on parallel and distributed systems*, vol. 30, no. 5, pp. 965–976, 2018.