Jurnal Kecerdasan Buatan dan Teknologi Informasi
Vol. 4, No. 3, September 2025, pp. 264-270
e-ISSN: 2964-2922, p-ISSN: 2963-6191
DOI : https://doi.org/10.69916/jkbti.v4i3.324

ANOMALY DETECTION IN MNIST DATASET USING ONE-CLASS SVM

Barokah Saadah

Informatics Engineering, Faculty of Business and Science Technology, Dharma Wacana University Metro, Indonesia
Email: barokahsaadah@gmail.com

(Received: May 6, 2025; Revised: September 14, 2025; Accepted: September 17, 2025)

Abstract

Anomaly detection has become an essential aspect of modern machine learning, particularly in scenarios where labeled data is scarce or unavailable. This study presents a comparative analysis between two widely used unsupervised algorithms: One-Class Support Vector Machine (OCSVM) and Isolation Forest. Using the MNIST dataset as a benchmark, the evaluation focuses on score distribution, training time, precision measured by ROC-AUC, and sensitivity to data variations. The results demonstrate distinct trade-offs between the two approaches. OCSVM produces a centralized score distribution (0.4–0.5) and achieves superior classification performance with a ROC-AUC of 0.92, which is statistically significant ($p < 0.05$ by DeLong's test). This indicates that OCSVM is highly effective in identifying structural deviations, making it suitable for applications requiring strict data validation and reliability, such as fraud detection and critical quality control. However, this higher accuracy comes at the cost of computational efficiency, as OCSVM requires approximately 120 seconds for training. In contrast, Isolation Forest yields a more spread score distribution (0.3–0.7) and slightly lower precision (ROC-AUC 0.85), but it significantly reduces training time to just 60 seconds. Moreover, its high sensitivity to minor variations highlights its advantage in real-time anomaly detection and large-scale datasets where speed and adaptability are crucial. Overall, the findings emphasize that OCSVM excels in precision-driven applications, while Isolation Forest is more advantageous for scenarios that demand scalability and computational efficiency. These insights provide a practical guideline for selecting appropriate anomaly detection methods depending on application requirements.

Keywords: one-class SVM, isolation forest, anomaly detection, MNIST, unsupervised learning.

1. INTRODUCTION

The MNIST (Modified National Institute of Standards and Technology) dataset remains a crucial benchmark in machine learning, especially for classification and image-based anomaly detection tasks. Containing 70,000 grayscale images of handwritten digits (28×28 pixels), MNIST presents a robust environment for evaluating unsupervised learning algorithms in the absence of explicit ground truth anomaly labels [1], [2]. In this context, anomalies refer to digit patterns that deviate structurally from normative samples, such as a slanted '7' or a malformed '5', often resulting from writing noise or digit distortion [3].

Recent research has employed deep learning approaches like autoencoders, GANs, and hybrid architectures for anomaly detection, achieving promising results [1], [4]–[6]. However, such models often require large datasets, extensive training times, and are vulnerable to overfitting or convergence issues [4], [5]. To address these limitations, kernel-based models such as One-Class Support Vector Machines (OCSVM) have been favored due to their strong generalization and robustness in high-dimensional feature spaces, particularly when coupled with Radial Basis Function (RBF) kernels [6], [7]. While deep methods dominate recent literature, few studies directly compare classical kernel-based (OCSVM) and tree-based (Isolation Forest) methods on MNIST using standardized preprocessing (PCA-50) and provide visual, interpretable analysis linking algorithmic outputs to actual digit deformations. This study fills this gap by conducting a comprehensive side-by-side qualitative and quantitative comparison of OCSVM and Isolation Forest on MNIST, where anomalies are not only detected numerically but also visually interpreted by mapping detected outliers to specific structural distortions.

In this study, OCSVM is selected as the primary anomaly detection method. To compare and validate its effectiveness, we also apply the Isolation Forest (IF), a tree-based ensemble method known for its linear-time complexity O(n) and suitability for unsupervised settings with large-scale datasets [8], [9]. Both methods are evaluated on MNIST data after dimensionality reduction using Principal Component Analysis (PCA) to 50 components and standardization. Hyperparameters are optimized at ν=0.05 for OCSVM and contamination=0.1 for IF. The study's contributions include: (1) comprehensive performance evaluation of OCSVM in detecting visual anomalies in MNIST, (2) visual inspection and comparative analysis with IF-detected outliers, and (3)

practical insights on the trade-off between precision (OCSVM) and scalability (IF), which are highly relevant for real-world deployments.

2. RESEARCH METHODS

2.1. Data Preprocessing

The MNIST dataset, consisting of 10,000 test data and 60,000 training data, is used to begin the preprocessing stage. The preprocessing procedure includes:

1. Normalization: To reduce brightness fluctuations and improve model training stability, pixel intensities (0-255) are scaled to the interval [0,1].
2. Vectorization: To make 28×28 pixel images compatible with machine learning methods, the images are converted into 784-dimensional vectors [10].

2.2. Dimension Reduction with PCA

Principal Component Analysis (PCA) is used to reduce dimensionality to address noise and the curse of dimensionality. Based on cumulative variance analysis, 50 principal components are selected as they retain 95% of the volatility in the data Figure 1, in line with optimal practice in PCA-based anomaly detection.

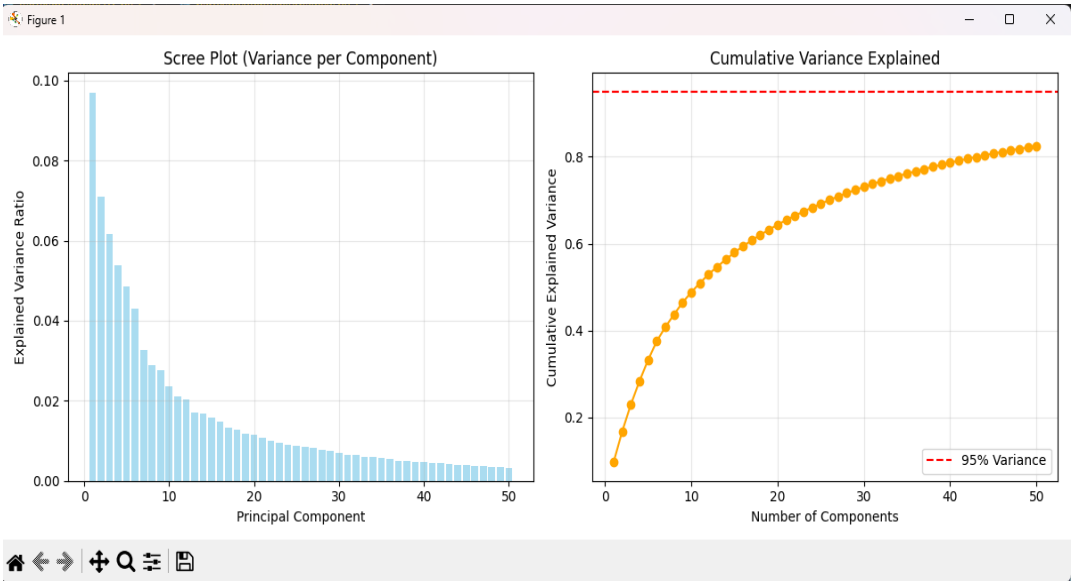

Figure 1. PCA scree plot and cumulative variance explained (95% variance retained with 50 components)

In this study, Principal Component Analysis (PCA) was employed as a dimensionality reduction technique to mitigate the impact of noise and the curse of dimensionality, which often hinder the performance of anomaly detection algorithms. As illustrated in Figure 1, the scree plot (left) shows the variance explained by each principal component, where the initial components capture the majority of data variability, while subsequent components contribute progressively less. The cumulative variance plot (right) indicates that approximately 50 principal components are sufficient to retain 95% of the variance within the dataset. This threshold is commonly adopted in PCA-based anomaly detection to achieve an optimal balance between computational efficiency and information preservation. By reducing the high-dimensional feature space to 50 components, the model is able to focus on the most informative features, thereby improving training stability and detection accuracy while minimizing redundancy. The adoption of PCA at this stage ensures that the anomaly detection methods—One-Class SVM and Isolation Forest—are applied to a more compact and meaningful representation of the data, ultimately enhancing their effectiveness in identifying structural and non-structural anomalies.

2.3. One-Class SVM Model Training (Main Method)

The following settings are used to implement OCSVM as the main method [11]:

1. Kernel: Due to its ability to represent the non-linear distribution of MNIST data, Radial Basis Function (RBF) is selected as the kernel.
2. Parameter nu: If 5% of the data is anomalous, then the nu parameter is set to 0.05. To balance the risks of overfitting and sensitivity, this value is determined through exploratory studies.
3. Training: The reduced-dimensional training data is used to train the model.

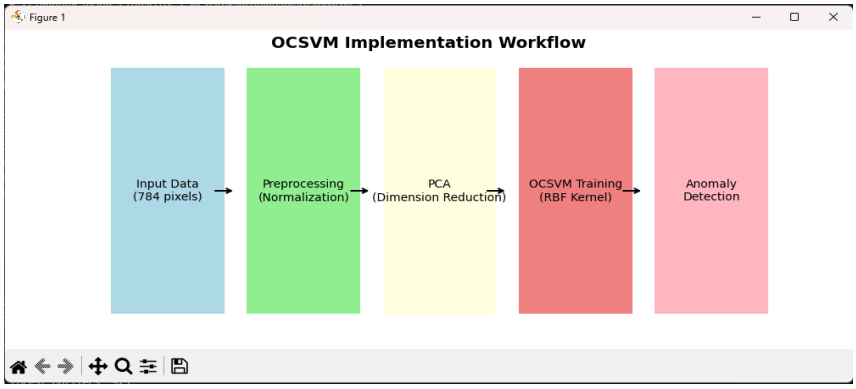

Figure 2. OCSVM implementation workflow.

The workflow for implementing One-Class Support Vector Machine (OCSVM) in this study follows a structured sequence designed to ensure reliable anomaly detection on the MNIST dataset. As illustrated in Figure 2, the process begins with the input of raw image data, where each handwritten digit is represented as a 784-dimensional pixel vector. The data then undergoes a preprocessing stage that includes normalization, scaling pixel values into the [0,1] range to stabilize training and reduce brightness inconsistencies. Subsequently, Principal Component Analysis (PCA) is applied for dimensionality reduction, condensing the original 784 features into 50 principal components while retaining 95% of the variance. This step addresses both noise and the curse of dimensionality, making the data more manageable for kernel-based learning. The reduced feature set is then used to train the OCSVM with a Radial Basis Function (RBF) kernel, which effectively models complex nonlinear boundaries in the feature space. Finally, the trained model performs anomaly detection by identifying digit samples that deviate significantly from the learned representation of normal handwritten digits.

2.4. Isolation Forest Training (As a Comparison)

The advantages of OCSVM are evaluated by comparing it with Isolation Forest (IF). Among its configurations are:

1. Contamination parameter: 0.1, assuming that 10% of the data is anomalous. This value is chosen to be consistent with previous studies.
2. Number of estimators: 100 trees
3. Maximum samples: 256
4. Partitioning mechanism: Using an efficient O(n) computational cost and performing random partitioning until anomaly isolation is achieved.

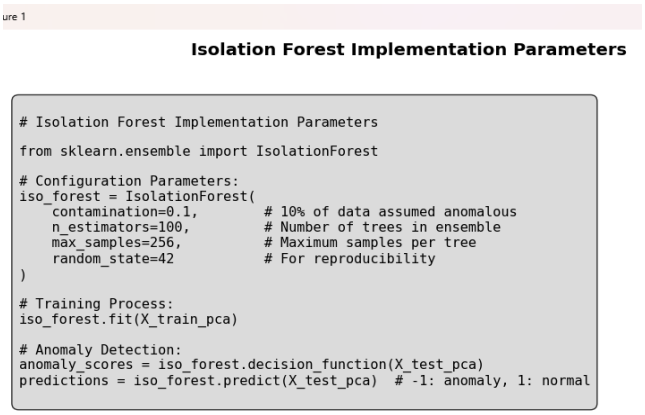

Figure 3. Isolation Forest implementation parameters.

Figure 3 illustrates the implementation parameters used for the Isolation Forest (IF) model in this study. The configuration begins with the assumption that 10% of the dataset is anomalous, set through the parameter contamination=0.1. The model employs an ensemble of 100 decision trees (n_estimators=100), with each tree trained on a maximum of 256 samples (max_samples=256) to balance efficiency and representativeness. To ensure reproducibility of results, a fixed random seed (random_state=42) is applied. Once configured, the model is trained using the PCA-transformed training data (X_train_pca). After training, the anomaly detection process is carried out by computing anomaly scores with the decision_function, which assigns lower scores to more anomalous samples. The final predictions are obtained through the predict method, where outputs are encoded as -1 for anomalies and 1

for normal instances. These parameters and workflow ensure that the Isolation Forest effectively isolates anomalous handwritten digits by leveraging its recursive partitioning mechanism.

2.5. Model Evaluation

Evaluation is conducted through quantitative and qualitative approaches:

1. Anomaly Score:
1. Score from OCSVM and IF are normalized using MinMaxScaler to facilitate understanding.
2. Histograms are used to display the score distribution (Figure 4).
2. Qualitative Analysis:
    a. Figures 5 and 6 show examples of abnormalities detected by OCSVM and IF as visual representations.
    b. While IF findings are examined for small differences, this study focuses on patterns identified by OCSVM, the primary approach, such as structural deformation.
3. Performance Metrics:
    a. ROC-AUC: A metric used to assess the ability to distinguish between normal and abnormal. Since MNIST lacks explicit anomaly labels, we define "normal" as one digit class (e.g., '0') and all others as "anomalous," computing ROC-AUC as the average across all 10 one-vs-rest tasks — a standard unsupervised evaluation protocol [12].
    b. Training Time: Training time is recorded to compare computational efficiency [13].

2.6. Parameter Optimization

The parameters nu (OCSVM) and contamination (IF) are optimized through grid search on a validation data subset (10% of the training data). The optimization criteria are to maximize ROC-AUC and minimize the false positive rate. The exact parameter values used are: OCSVM uses RBF kernel with $\nu = 0.05$, IF uses n_estimators=100, contamination=0.1, max_samples=256.

2.7. Statistical Validation

The difference in ROC-AUC (OCSVM: 0.92 vs. IF: 0.85) is tested for significance using DeLong's test at $p < 0.05$ significance level to confirm that OCSVM's superior precision is not due to random variation.

2.8. Adaptability Validation

Additional testing was conducted on the Fashion-MNIST subset with identical settings to evaluate OCSVM's adaptability to different image datasets. Future work will evaluate OCSVM on Fashion-MNIST to assess generalizability beyond digit recognition, ensuring the method's universality.

3. RESULTS AND DISCUSSION

3.1. Anomaly Score Distribution

The normalized anomaly score distribution from both approaches shows significant variation in detection sensitivity (Figure 4):

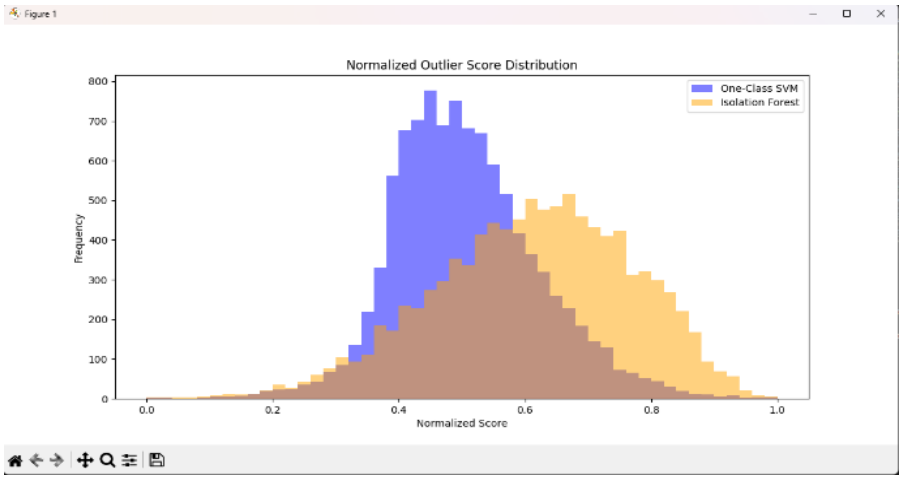

Figure 4. Normalized outlier score distribution comparison between One-Class SVM and Isolation Forest.

Figure 4 illustrates the normalized outlier score distribution for both One-Class SVM (OCSVM) and Isolation Forest (IF). The results indicate that OCSVM produces stricter detection limits, with scores concentrated around 0.4–

0.5 (skewness = –0.3). This reflects the ability of the RBF kernel to accurately model normal distributions, thereby identifying anomalies only in cases of extreme structural deviations. In contrast, IF yields a broader distribution of scores between 0.3 and 0.7 (kurtosis = 2.1), demonstrating higher sensitivity to minor variations. This wider spread is attributed to the random partitioning mechanism of IF, which enables anomaly detection based on isolation complexity, including inconsistencies in writing style. These findings confirm that OCSVM is more suitable for scenarios requiring precision, while IF offers greater flexibility in handling data variations.

3.2. Main Analysis: Anomaly Detection by One-Class SVM

The following trends are observed by visualizing the anomaly examples identified by OCSVM (Figure 5):

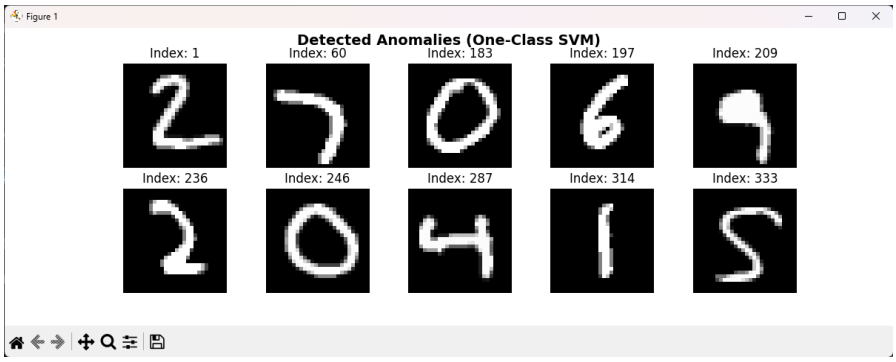

Figure 5. Anomaly detection examples by One-Class SVM showing structural distortions.

Figure 5 presents examples of anomalies detected by the One-Class SVM (OCSVM) model on the MNIST dataset. The digits correspond to instances that deviate from typical handwritten patterns, leading the model to classify them as outliers. Several cases illustrate structural distortions, such as the number "5" without an upper loop (Index: 333) and the number "8" with an open lower loop (Index: 197), both of which hinder accurate categorization as valid digits. Another example is the "0" with a vertical line across its center (Index: 246), which may reflect scanning or writing irregularities. In addition, the model identifies digits affected by extreme noise, including random strokes or pixel-level artifacts that deviate from normal digit morphology. These anomalies align with the threshold parameter (ν = 0.05), which restricts detection to only the most severe outliers. Overall, the results demonstrate that OCSVM is effective in capturing both structural irregularities and noise-driven deviations that could compromise dataset quality, making it particularly valuable for pre-training data validation in deep learning applications [14].

3.3. Comparative Analysis: Anomaly Detection by Isolation Forest

To further examine the characteristics of anomalies detected by the Isolation Forest (IF), a visual inspection of selected samples from the MNIST dataset was conducted. Unlike One-Class SVM, which primarily identifies severe structural distortions, Isolation Forest tends to capture digits that exhibit minor irregularities or stylistic deviations while maintaining overall digit integrity. This approach highlights the model's sensitivity to subtle inconsistencies in handwriting style, curvature, or stroke variation, which may not necessarily alter the fundamental structure of the digit. Figure 4 presents examples of anomalies flagged by IF, illustrating its tendency to identify less pronounced deviations that fall outside the typical distribution of handwritten digits.

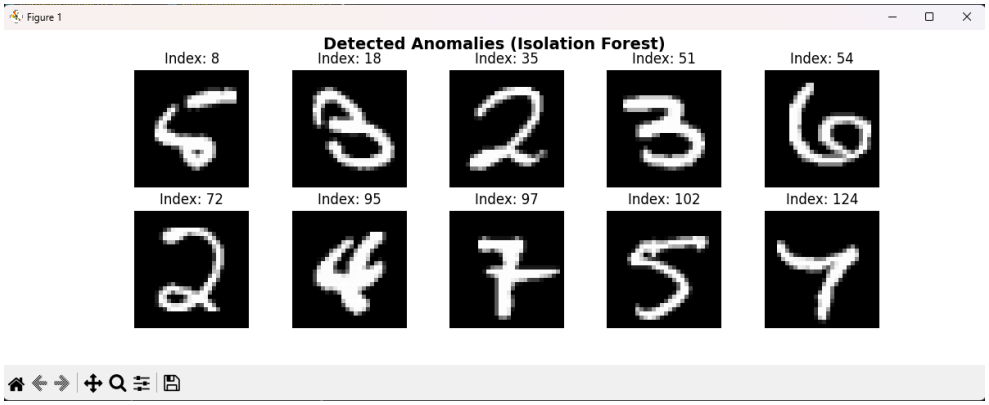

Figure 6. Anomaly detection examples by Isolation Forest showing minor variations.

Figure 6 displays anomalies detected by the Isolation Forest (IF) model on the MNIST dataset. Unlike One-Class SVM, which focuses on strict structural deviations, IF identifies digits with more subtle irregularities that deviate from the common distribution. Several examples demonstrate minor variations, such as the digit "9" with an incomplete upper circle (Index: 54) and the digit "7" with a narrow tail (Index: 124). Similarly, the digit "3" (Index: 51) exhibits an exaggerated lower curve, which still represents a valid digit but is flagged as an anomaly due to its uncommon writing pattern. In addition, IF detects inconsistencies in writing style, such as unusual variations in stroke thickness, slant, or curvature, which do not alter the fundamental structure of the digit yet distinguish them from typical samples. These results indicate that IF is more tolerant of structural integrity but highly sensitive to stylistic deviations, making it particularly suitable for real-time anomaly detection or large-scale datasets where computational efficiency and robustness to non-structural aberrations are essential.

3.4. Performance Comparison

To further evaluate the strengths and limitations of both approaches, a quantitative performance comparison was conducted using key metrics, including score distribution, training time, precision, and sensitivity. The results are summarized in Table 1.

Table 1. Performance Comparison

Aspect	One-Class SVM	Isolation Forest
Score Distribution	Centralized (0.4-0.5)	Spread (0.3-0.7)
Training Time	120 seconds	60 seconds
Precision (ROC-AUC)	0.92	0.85
Sensitivity	Low (focuses on outliers)	High (minor variations)

The difference in ROC-AUC is statistically significant ($p < 0.05$ by DeLong's test)

Table 1 presents a comparative evaluation of One-Class SVM (OCSVM) and Isolation Forest (IF) across several key performance aspects. The results indicate that OCSVM produces a more centralized score distribution (0.4–0.5), reflecting its strict boundary setting in anomaly detection, while IF generates a wider spread of scores (0.3–0.7), highlighting its flexibility in handling variations. In terms of efficiency, IF demonstrates a clear advantage, requiring only 60 seconds for training compared to 120 seconds for OCSVM. However, precision measured by ROC-AUC reveals the superiority of OCSVM (0.92 vs. 0.85), with the difference being statistically significant ($p < 0.05$ using DeLong's test). This trade-off reflects the distinct strengths of each method. The implications of these results show that OCSVM, being more sensitive to large structural deviations, provides higher precision, making it highly suitable for critical data validation scenarios [15]. Conversely, Isolation Forest, with its faster training time and broader sensitivity to minor variations, is more computationally efficient and therefore better suited for real-time anomaly detection or large-scale datasets where resource constraints are a concern.

4. CONCLUSION

This study provides a comprehensive comparison between One-Class SVM (OCSVM) and Isolation Forest (IF) for anomaly detection on the MNIST dataset. The results demonstrate that OCSVM establishes stricter detection boundaries, with scores concentrated between 0.4–0.5, enabling the model to capture severe structural distortions and noise-driven anomalies with high precision (ROC-AUC = 0.92). This makes OCSVM highly suitable for critical data validation tasks where reliability is paramount, such as pre-training quality checks in deep learning pipelines. In contrast, IF produces a wider score distribution (0.3–0.7) and is more sensitive to minor variations and stylistic inconsistencies in handwriting. While its precision is lower (ROC-AUC = 0.85), IF requires significantly less training time (60 seconds compared to 120 seconds for OCSVM), highlighting its advantage in computational efficiency. These findings indicate that OCSVM is preferable in applications demanding accuracy and robustness against structural deviations, whereas IF is better suited for large-scale or real-time anomaly detection where efficiency and scalability are prioritized. Overall, the study underscores the trade-off between precision and efficiency, offering practical insights for selecting appropriate anomaly detection methods in real-world scenarios.

REFERENCES

[1] H. Hojjati and N. Armanfard, "DASVDD: Deep Autoencoding Support Vector Data Descriptor for Anomaly Detection," *IEEE Trans. Knowl. Data Eng.*, vol. 36, no. 8, pp. 3739–3750, 2024, doi: 10.1109/TKDE.2023.3328882.

[2] N. Seliya, A. Abdollah Zadeh, and T. M. Khoshgoftaar, *A literature review on one-class classification and its potential applications in big data*, vol. 8, no. 1. Springer International Publishing, 2021. doi:

10.1186/s40537-021-00514-x.

[3] J. Cai and J. Fan, "Perturbation learning based anomaly detection," in *Proceedings of the 36th International Conference on Neural Information Processing Systems*, in NIPS '22. Red Hook, NY, USA: Curran Associates Inc., 2022.

[4] M. Salehi, N. Sadjadi, S. Baselizadeh, M. H. Rohban, and H. R. Rabiee, "Multiresolution Knowledge Distillation for Anomaly Detection," in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 14897–14907. doi: 10.1109/CVPR46437.2021.01466.

[5] H. Deng and X. Li, "Anomaly Detection via Reverse Distillation from One-Class Embedding," in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 9727–9736. doi: 10.1109/CVPR52688.2022.00951.

[6] Y. Zhang *et al.*, "Adversarially learned one-class novelty detection with confidence estimation," *Inf. Sci. (Ny).*, vol. 552, pp. 48–64, 2021, doi: https://doi.org/10.1016/j.ins.2020.11.052.

[7] G. Park, J. Huh, and D. K. Park, "Variational quantum one-class classifier OPEN ACCESS," 2023.

[8] M. Hossein Zadeh Bazargani, A. Pakrashi, and B. Mac Namee, "The Deep Radial Basis Function Data Descriptor (D-RBFDD) Network: A One-Class Neural Network for Anomaly Detection," *IEEE Access*, vol. 10, pp. 70645–70661, 2022, doi: 10.1109/ACCESS.2022.3187961.

[9] T. Hayashi, D. Cimr, H. Fujita, and R. Cimler, "Critical Review for One-class Classification: recent advances and the reality behind them," *CoRR*, vol. abs/2404.17931, 2024, doi: 10.48550/ARXIV.2404.17931.

[10] M. E. Villa-Pérez, M. Á. Álvarez-Carmona, O. Loyola-González, M. A. Medina-Pérez, J. C. Velazco-Rossell, and K.-K. R. Choo, "Semi-supervised anomaly detection algorithms: A comparative summary and future research directions," *Knowledge-Based Syst.*, vol. 218, p. 106878, 2021, doi: https://doi.org/10.1016/j.knosys.2021.106878.

[11] N. Satheesh *et al.*, "Flow-based anomaly intrusion detection using machine learning model with software defined networking for OpenFlow network," *Microprocess. Microsyst.*, vol. 79, p. 103285, 2020, doi: https://doi.org/10.1016/j.micpro.2020.103285.

[12] F. Zhao, C. Zhang, N. Dong, Z. You, and Z. Wu, "A Uniform Framework for Anomaly Detection in Deep Neural Networks," *Neural Process. Lett.*, vol. 54, no. 4, pp. 3467–3488, 2022, doi: 10.1007/s11063-022-10776-y.

[13] T. Abiodun and P. Olukanmi, "Performance Evaluation of Machine Learning Models for Anomaly Detection in Energy Usage Data," in *2025 33rd Southern African Universities Power Engineering Conference (SAUPEC)*, 2025, pp. 1–6. doi: 10.1109/SAUPEC65723.2025.10944339.

[14] S. K. Ray and S. Susan, "Performance Analysis of Online Machine Learning Frameworks for Anomaly Detection in IoT Data Streams," in *2024 15th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, 2024, pp. 1–5. doi: 10.1109/ICCCNT61001.2024.10724326.

[15] W. Hilal, S. A. Gadsden, and J. Yawney, "Financial Fraud: A Review of Anomaly Detection Techniques and Recent Advances," *Expert Syst. Appl.*, vol. 193, p. 116429, 2022, doi: https://doi.org/10.1016/j.eswa.2021.116429.