Contents lists available at ScienceDirect

Forensic Science International: Digital Investigation

journal homepage: www.elsevier.com/locate/fsidi

DFRWS USA 2025 - Selected Papers from the 25th Annual Digital Forensics Research Conference USA

# Bridging knowledge gaps in digital forensics using unsupervised explainable AI

Zainab Khalid [a], Farkhund Iqbal [a,*], Mohd Saqib [b]

[a] College of Technological Innovation, Zayed University, Dubai, United Arab Emirates
[b] School of Information Studies, McGill University, Montreal, Canada

ARTICLE INFO

*Keywords:*
Digital forensics
Explainable artificial intelligence
XAI
Unsupervised learning
Isolation forest
Autoencoder
K-means
DBSCAN
GMM
SHAP
Principal component analysis
CIC MalMemAnalysis-2022

ABSTRACT

Artificial Intelligence (AI) has found multi-faceted applications in critical sectors including Digital Forensics (DF) which also require eXplainability (XAI) as a non-negotiable for its applicability, such as admissibility of expert evidence in the court of law. The state-of-the-art XAI workflows focus more on utilizing XAI tools for supervised learning. This is in contrast to the fact that unsupervised learning may be practically more relevant in DF and other sectors that largely produce complex and unlabeled data continuously, in considerable volumes. This research study explores the challenges and utility of unsupervised learning-based XAI for DF's complex datasets. A memory forensics-based case scenario is implemented to detect anomalies and cluster obfuscated malware using the Isolation Forest, Autoencoder, K-means, DBSCAN, and Gaussian Mixture Model (GMM) unsupervised algorithms on three categorical levels. The CIC MalMemAnalysis-2022 dataset's binary, and multivariate (4, 16) categories are used as a reference to perform clustering. The anomaly detection and clustering results are evaluated using accuracy, confusion matrices and Adjusted Rand Index (ARI) and explained through Shapley Additive Explanations (SHAP), using force, waterfall, scatter, summary, and bar plots' local and global explanations. We also explore how some SHAP explanations may be used for dimensionality reduction.

## 1. Introduction

A key challenge in Digital Forensics (DF) workflows is bridging the communication gap between technical forensic experts and non-technical professionals, such as police officers and judges. In reference to Federal Rule of Evidence (FRE)703,[1] expert witnesses may base their opinion on facts or data that would normally be inadmissible in court. Artificial Intelligence (AI) offers significant advantages in DF by automating analysis, improving detection accuracy, and helping investigators handle large volumes of case material (digital evidence datasets), but more so, it ensures expert evidence (EE) is clearly conveyed and easily understood. In this context, eXplainability[2] (XAI) is crucial; black box[3] models must be interpretable to be trusted and applied in high-stakes areas like DF. XAI develops trust in AI models by explaining how they reached certain predictions/classifications. The clarity gained through explanations develops trust among users and stakeholders and helps tackle the challenges of accountability and interpretability linked to advanced AI systems.

Explainability has generally been explored for supervised learning algorithms (that use labeled data to train models which then make label predictions) with XAI tools more catered for these algorithms as well Khalid et al. (2024); Hall et al. (2022, 2021); Solanke and Biasiotti (2022); Dunsin et al. (2022). Data labeling done manually is an expensive and time-consuming process Wickramasinghe et al. (2021). In DF in particular, as volumes of forensic images grow from GBs to TBs range, it becomes increasingly difficult to have and maintain labeled datasets. It also requires field experts like forensic analysts or investigators in DF. In contrast, Unsupervised Learning (UL) which uses unlabeled data to identify unknown patterns based on the structure of the data provides a wider scope of applicability. In fact, in real-case DF scenarios, UL algorithms can effectively detect unknown threats by uncovering previously unseen patterns and relationships within datasets Öztürk and

* Corresponding author.
*E-mail address:* Farkhund.Iqbal@zu.ac.ae (F. Iqbal).

[1] https://www.law.cornell.edu/rules/fre/rule_703.

[2] The terms explainability and interpretability are used interchangeably in the paper within the context of XAI.

[3] A black box AI model's internal processes are opaque and not easily interpretable, making it difficult to understand how it arrives at its decisions/predictions Khalid et al. (2024).

https://doi.org/10.1016/j.fsidi.2025.301924

Hızal (2024). UL algorithms are more suitable also because data does not need labeling. This also eliminates biases associated with labeled datasets used in supervised learning Wickramasinghe et al. (2021).

However, UL has comparatively been under-researched with respect to explainability. This may be attributed to a few challenges presented with UL algorithms. On the surface level, the missing labels/*ground truth* (due to the structural complexity of the data, limits of human knowledge, and significant volumes that complicate the categorization process) obscure the analysis and evaluation of results of clustering or anomaly detection, etc. Morichetta et al. (2019). Since data is not labeled, judging/evaluating the accuracy of the results is challenging such as whether detected anomalies truly represent malicious activity. Even though evaluation metrics like silhouette coefficient or rank index provide structural insights about the results, they do not explain why a datapoint was categorized into a particular cluster by the model Morichetta et al. (2019). Also, the results of UL algorithms can sometimes especially be harder to interpret (requiring manual expert analysis) in complex datasets like those in DF which contain noise (such as network traffic, memory dumps, multimedia, disk files, logs, etc.).

In the research study that follows, we explore the challenges and applicability of explainable UL for DF. The major contributions of this study are as follows:

- We explore the applicability of UL in a memory forensics-based DF case scenario, utilizing Isolation Forest and Autoencoder to detect anomalies and K-means, DBSCAN, and Gaussian Mixture Model (GMM) to cluster obfuscated malware in memory (using the CIC MalMemAnalysis-2022 dataset).
- We present explanations/interpretations of anomalies and clusters made by UL models, testing Shapley Additive Explanations (SHAP), typically used for supervised learning.

The rest of this paper is structured as follows. Section II discusses previous research and other related contributions. Section III details the methodology and experimental setup. Section IV discusses the results of the XAI-DF experiments utilizing unsupervised algorithms. Section V discusses the final comments, conclusion, and possible future directions in the domain.

## 2. Related work

### 2.1. State-of-the-art in XAI-DF

The utilization of AI and explainability in DF has been explored within the supervised learning domain. We previously proposed a holistic yet exhaustive XAI-DF framework detailing the workflow of DF investigations that use XAI (Khalid et al. (2024)). Our demonstrations of the framework include supervised learning-based network and memory forensics case studies with Local Interpretable Model-Agnostic Explanations (LIME) and SHAP explanations.

Hall et al. (2022, 2021) perform proof-of-concept implementations of XAI for IT forensics utilizing a curated database of 23 VHD images to source multimedia (images and videos) and file metadata for training and testing AI models. They use LIME to get explanations.

Solanke (2022) examines the limitations of black-box AI models and investigates approaches to enhance the interpretability of AI-driven DF. This effort addresses the skepticism of courts, legal practitioners, and the public regarding the use of AI in digital evidence extraction, driven by concerns about transparency and comprehensibility.

Dunsin et al. (2022) propose an agent-based MADIK framework that adopts a modular approach, training and testing AI models for distinct forensic tasks such as a registry agent that deals with registry data only. Models trained for specific DF tasks like this may perform more efficiently. However, they do not explore explainability.

### 2.2. Explainability for unsupervised learning algorithms

Most of the existing literature on techniques and methodologies for UL-based XAI use some degree of supervised learning in their workflows Wickramasinghe et al. (2021). Montavon et al. (2022) propose a 'neuralization-propagation' (NEON) approach which first converts the UL model into a *functionally equivalent neural network* (a supervised algorithm) followed by the usage of supervised XAI techniques such as Layer-wise Relevance Propagation (LRP). They use this method to explain Kernel Density Estimation and K-means clustering-based case studies.

Morichetta et al. (2019) propose the EXPLAIN-IT methodology which also uses supervised learning to explain clusters. The clustering results obtained from UL models are input as labels to train a classification (supervised) model and the results are then explained using standard XAI tools like LIME. The proposed methodology is demonstrated using a YouTube QoE dataset. The authors acknowledge that using supervised models to explain UL algorithms introduces a bias.

Won Oh et al. (2022) use Autoencoder, a (neural network) UL model, for anomaly detection of nuclear power plants as part of accident mitigation measures. In addition, SHAP is used to explain the anomalies.

Brito et al. (2022) perform detection and diagnosis of faults in rotating machinery using UL anomaly detection models and use SHAP for explanations.

### 2.3. Research and testing with CIC MalMemAnalysis-2022 dataset

Carrier et al. (2022) propose an updated VolMemLyzer-V2 that extracts features from memory dumps to detect obfuscated and hidden malware. A memory dataset, CIC MalMemAnalysis-2022, is created using the VolMemLyzer-V2 to train and test learning systems for obfuscated malware detection. Their testing with the dataset entails using a *stacking* ensemble learning approach that has two layers of classifiers. Various supervised learning algorithms are used for binary classification of the dataset (benign vs. malicious) achieving an accuracy score of 99 %. This is done by using Naive Bayes (NB), Decision Tree (DT), and Random Forest (RF) as base learners in the first layer and Logistic Regression (LR) as the meta-learner in the second layer.

Dener et al. (2022) perform binary classification on the dataset using nine different supervised Machine Learning (ML) and Deep Learning (DL) models (RF, DT, Gradient Augmented Tree (GBT), LR, NB, Linear Support Vector Machine (Linear SVC), Multilayer Perceptron (MLP), Deep Neural Network (DNN), and Long Short-Term Memory (LSTM)). They mostly use default parameters during model training, getting the highest accuracy for Logistic Regression, i.e. 99.97 %.

Mezina and Burget (2022) perform both binary and multiclass (4) classification on the dataset using common supervised models (such as RF, LR, DT, MLP, Support Vector Machine (SVM), K-Nearest Neighbors (KNN)) and a proposed dilated Convolution Neural Network (CNN). They use the *random search* method to find the optimal hyperparameters for the common models. RF achieves 99.99 % accuracy, while their proposed CNN model achieves 83.53 % accuracy for multi-class classification.

Öztürk and Hızal (2024) perform binary and multiclass (4, 16) classification of the dataset using supervised learning. Their experiments prove the most effective model to be XGBoost under various experimental conditions such as percentage splits and 10-fold cross-validation. Accuracy scores of 99.99 %, 87.79 %, and 75.49 % are achieved for binary and multiclass (4, 16) classification, respectively.

## 3. Methodology and experimental setup

To explore the implementation of the explainable unsupervised learning for DF (particularly memory forensics) workflow, we adapt our XAI-DF framework detailed in Khalid et al. (2024) to unsupervised learning and experiment with the CIC MalMemAnalysis-2022 dataset to

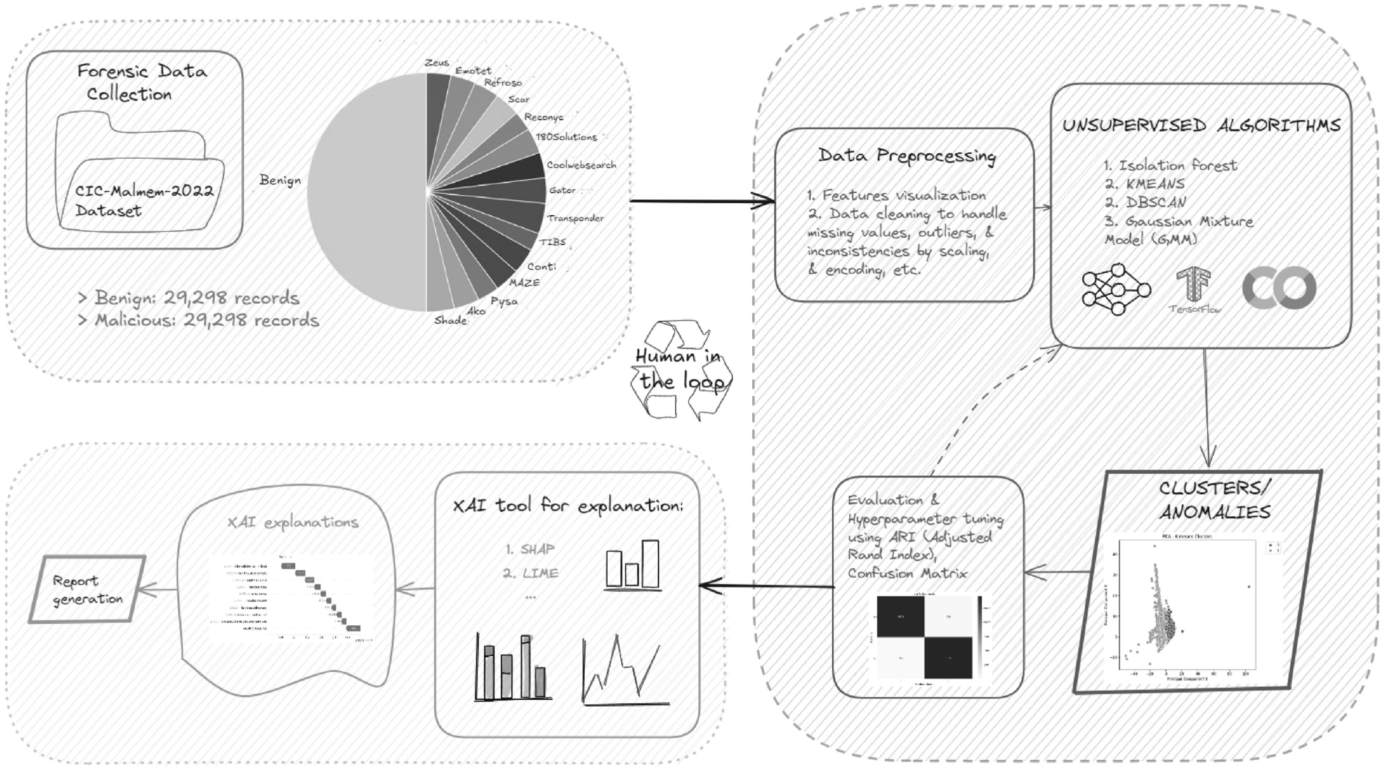


**Fig. 1.** Unsupervised learning methodology.

detect anomalies (i.e. obfuscated malware) using Isolation forest and Autoencoder, and also cluster various malware categories in memory using K-means, DBSCAN, and Gaussian Mixture Model (GMM). Shapley Additive Explanations (SHAP), a state-of-the-art post-hoc XAI tool based on game theory, is used to interpret/explain the anomalies and clusters based on feature importance Lundberg and Lee (2017). The methodology is detailed in Fig. 1.

### 3.1. CIC MalMemAnalysis-2022 dataset for learning systems

Researchers at the Canadian Institute for Cybersecurity created the CIC MalMemAnalysis-2022 dataset using real malware to train and test learning systems for obfuscated malware detection Carrier et al. (2022).

The dataset was created using malicious and benign memory dumps. For malicious dumps, 2916 samples of malware categorized into Ransomware (Conti, MAZE, Pysa, Ako, Shade), Spyware (180Solutions, Coolwebsearch, Gator, Transponder, TIBS), and Trojan Horse (Zeus, Emotet, Refroso, scar, Reconyc) were executed. Benign dumps were captured after benign Windows activity. The VolMemLyzer-V2 tool was then used to extract 55 features from the memory dumps, creating a CSV 'CIC MalMemAnalysis-2022'. The dataset, balanced using the SMOTE algorithm, consists of 58,596 records with 29,298 benign and 29,298 malicious instances. The *'Class'* feature can be used as a reference for binary clustering/classification and anomaly detection, and the *'Category'* feature which has the types (4) and sub-types (16) of malware can be referenced for multivariate clustering/classification.

For our implementation, we use the dataset in the context of a memory forensics investigation to perform (a) anomaly detection and (b) binary and multivariate (4, 16) clustering-based analysis.

### 3.2. Unsupervised learning models for clustering

The Isolation Forest and Autoencoder models were used for anomaly detection, whereas K-means, DBSCAN, and GMM models were used to obtain binary and multivariate clusters of the dataset. We used the scikit-learn (Python) implementations of these algorithms in Google Colaboratory,[4] which may be referenced from Github at https://github.com/znbkhld/Unupervised-XAI-DF.

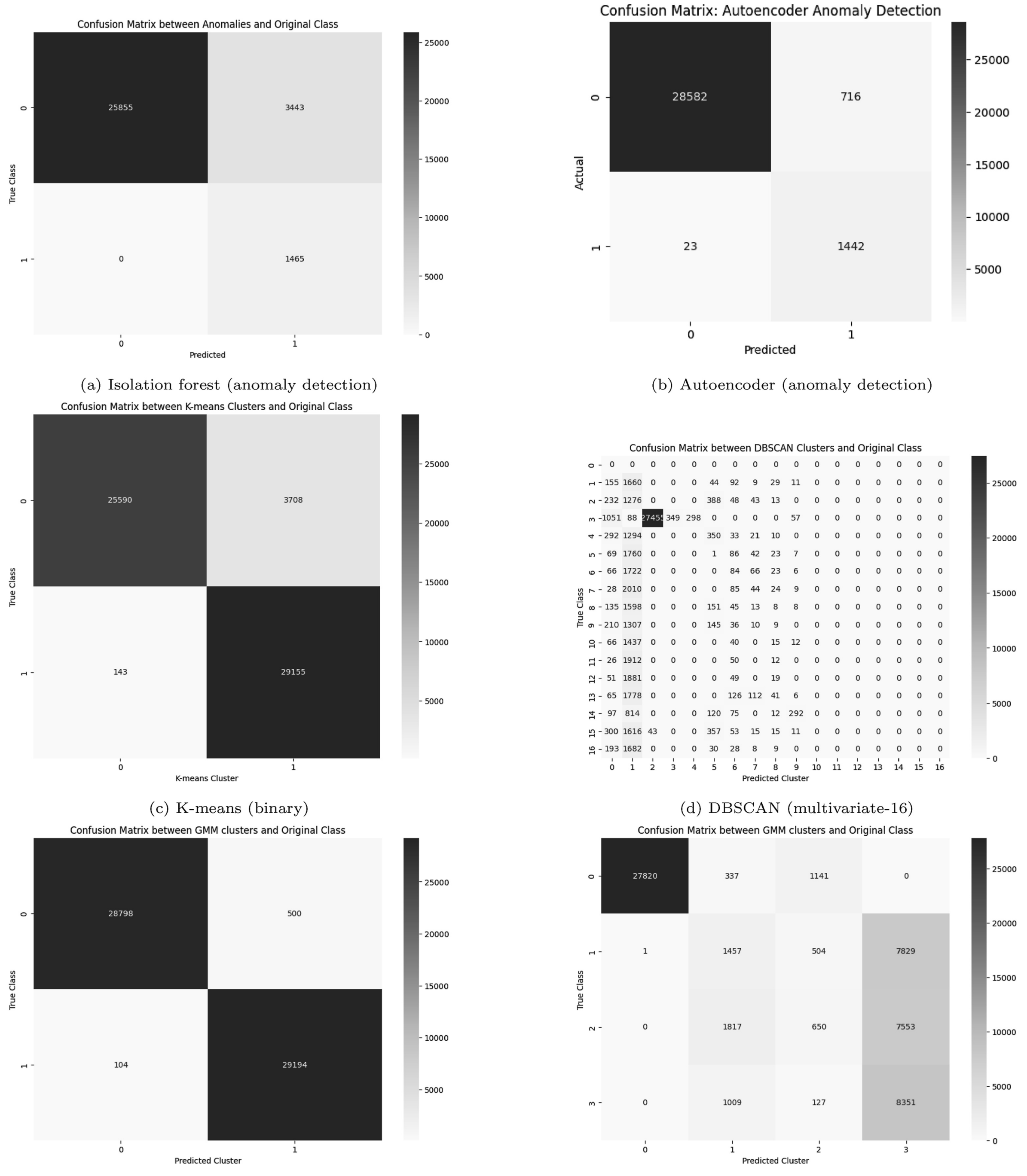
For anomaly detection, since the dataset contains an equal number of malicious and benign instances, only initial 5 % of malicious instances were retained while purging the rest to create an imbalance in benign vs. malicious instances and, in turn, the *anomalous* nature of malicious instances in the dataset.

For clustering, to assess the dataset's clustering tendency (and verify the non-random nature of the data), the Hopkins statistic was calculated, yielding a value of 0.9, which suggests that the dataset is suitable for clustering.

The dataset CSV was loaded using Pandas after being imported onto Google Colab. Since this dataset does contain the ground truth, i.e. the *'Class'* and *'Category'* features, they were removed for experiments with the unsupervised models during preprocessing. The anomalies detected and clusters formed by all the models are purely based on the raw data patterns without labels. Before clustering, StandardScaler was used to standardize the features. The model hyperparameters, specifically for DBSCAN, were identified based on *'random search'*. Since DBSCAN discovers clusters without an input parameter (that specifies the number of clusters from the get-go), it was used to perform multivariate (16) type clustering (further details in "Results and Discussion"). K-means was used to perform binary (2) clustering while GMM was used for binary and multivariate (4) type clustering. Note that these specific models with the specific types of clustering gave the best results compared to other models (and types) which is why they are included in the study for a detailed discussion.

Since ground truth labels were available, they were used (only) to evaluate the performance of the unsupervised models. Isolation Forest

[4] https://colab.research.google.com/.



(a) Isolation forest (anomaly detection)

(b) Autoencoder (anomaly detection)

(c) K-means (binary)

(d) DBSCAN (multivariate-16)

(e) GMM (binary)

(f) GMM (multivariate-4)

**Fig. 2.** Confusion matrices for IF, Autoencoder, K-means, DBSCAN, and GMM models.

and Autoencoder were evaluated using Accuracy score and a confusion matrix.[5] Clustering algorithms were evaluated using two metrics: (1) confusion matrices, and (2) Adjusted Rand Index (ARI).[6] Finally, Principal Component Analysis (PCA) was used to visualize the original class and obtained clusters across two dimensions for comparison.

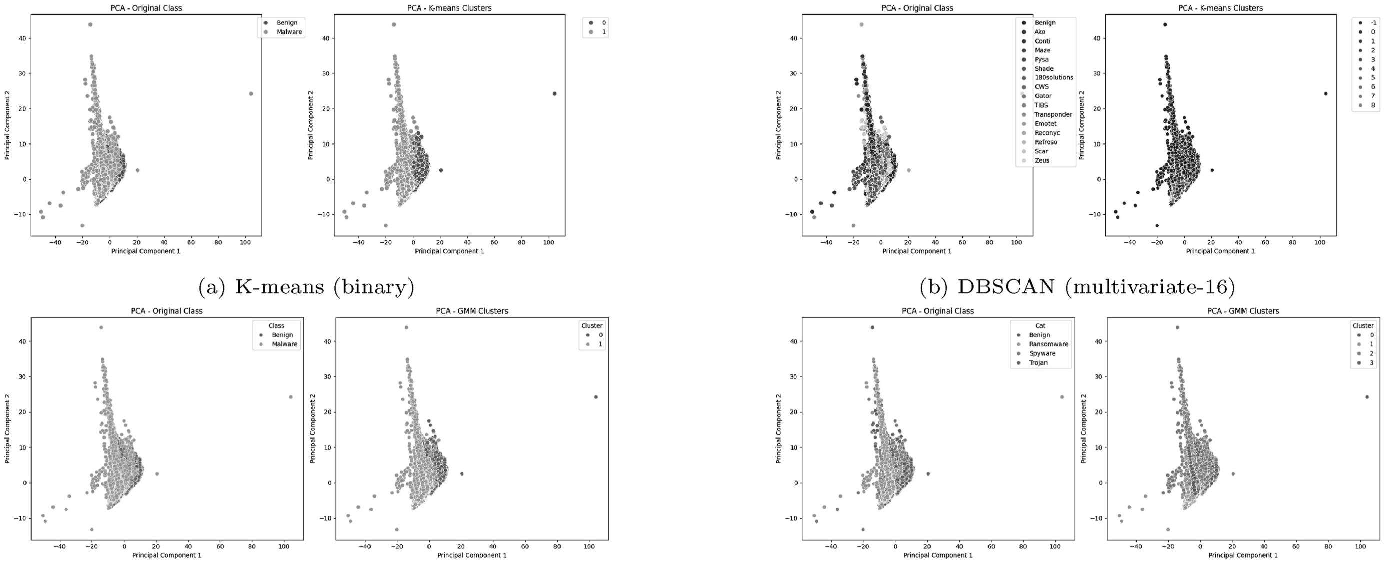
### 3.3. *SHAP explanations for interpretability*

Evaluation metrics like rank indexes can quantify the efficiency of particular algorithms, but detailed information about the anomalies/clusters, and the most relevant features contributing to the assignment of instances to them may be obtained via XAI tools like SHAP Lundberg and Lee (2017). Unlike most tools for explainability, SHAP can be used with both supervised and unsupervised algorithms Brito et al. (2022).

We utilize SHAP to extract a range of explanations for anomalies and cluster analysis based on feature importance. In particular, force, waterfall, scatter, summary, and bar plots' local and global explanations are extracted. To do so, we use the *Explainer* implementation/function of SHAP which can be applied to any model and is not limited to tree-based

[5] Confusion matrix, also known as error matrix, is a visual table that is used to evaluate the performance of a model.

[6] Adjusted Rand Index (ARI) computes the similarity between predicted clusters and ground truth labels. It ranges from +1 to −1 where +1 indicates perfect similarity between *two clusterings*, 0 indicates random, and −1 indicates that the clusterings are completely different.



(a) K-means (binary)

(b) DBSCAN (multivariate-16)

(c) GMM (binary)

(d) GMM (multivariate-4)

**Fig. 3.** Principal Components' comparison (original class vs. obtained clusters) for K-means, DBSCAN, and GMM models.

**Table 1**
Anomaly detection and clustering results for CIC MalMemAnalysis-2022 dataset.

| Unsupervised algorithm | Hyperparameters | Evaluation metrics |
|---|---|---|
| Isolation Forest (anomaly detection) | n_estimators = 100, max_samples = 'auto', max_features = 55, contamination = 'auto', random_state = 42 | 0.8595 (accuracy) |
| Autoencoder (anomaly detection) | epochs = 50, batch_size = 32, validation_split = 0.1, Activation = ReLU, Optimizer = Adam, Loss = Mean squared error | 0.9759 (accuracy) |
| K-means (binary clustering) | n_clusters = 2, random_state = 42 | 0.7544 (ARI) |
| DBSCAN (multivariate-16 clustering) | eps = 2.3, min_samples = 190 | 0.5690 (ARI) |
| GMM (binary clustering) | n_components = 2, random_state = 42 | 0.9592 (ARI) |
| GMM (multivariate-4 clustering) | n_components = 4, random_state = 42 | 0.6449 (ARI) |

models or neural networks.

## 4. Results and Discussion

### 4.1. Evaluation metrics

For anomaly detection of the unbalanced CIC MalMemAnalysis-2022 dataset, Isolation Forest hyperparameters were set to n_estimators = 100, max_samples = 'auto', max_features = 55, contamination = 'auto', random_state = 42. The overall accuracy was quantified to 0.8595 using all 55 features. The confusion matrix for IF is illustrated in Fig. 2a.

Autoencoder hyperparameters for anomaly detection were set to epochs = 50, batch_size = 32, validation_split = 0.1, Activation = ReLU, Optimizer = Adam, Loss = Mean squared error. The overall accuracy was quantified to 0.9759 using all 55 features. The confusion matrix for Autoencoder is illustrated in Fig. 2b.

Applying K-means clustering to the CIC dataset with n_clusters = 2 (benign vs. malware 'Class' label) and random_state = 42 gave ARI of 0.7544 which is considered good with respect to a complex dataset. The confusion matrix for K-means binary clustering is illustrated in Fig. 2c. A visual comparison of the original class vs. obtained K-means (binary) clusters can be made using the PCA plots illustrated in Fig. 3a.

DBSCAN hyperparameters were set to eps = 2.3, min_samples = 190 through *random search*. Since DBSCAN does not input the number of clusters, it is evaluated based on the maximum possible clusters in the dataset, i.e. the multivariate-16 category. DBSCAN discovers **9** clusters from the dataset (being inaccurate) and achieving an ARI of 0.5690. The confusion matrix and PCA plots for DBSCAN are illustrated in Figs. 2d and 3b, respectively.

GMM was used for binary (n_components = 2, random_state = 42) and multivariate-4 (n_components = 4, random_state = 42) clustering giving ARIs 0.9592 and 0.6449, respectively. The confusion matrices and PCA plots for GMM are illustrated in Fig. 2e and f and Fig. 3c, d respectively.

GMM achieves excellent results for binary clustering. K-means and DBSCAN results are considerably good, reinforcing that unsupervised models may be effectively used in complex datasets like DF.

Table 1 lists the hyperparameters of the unsupervised models and the final ARIs achieved.

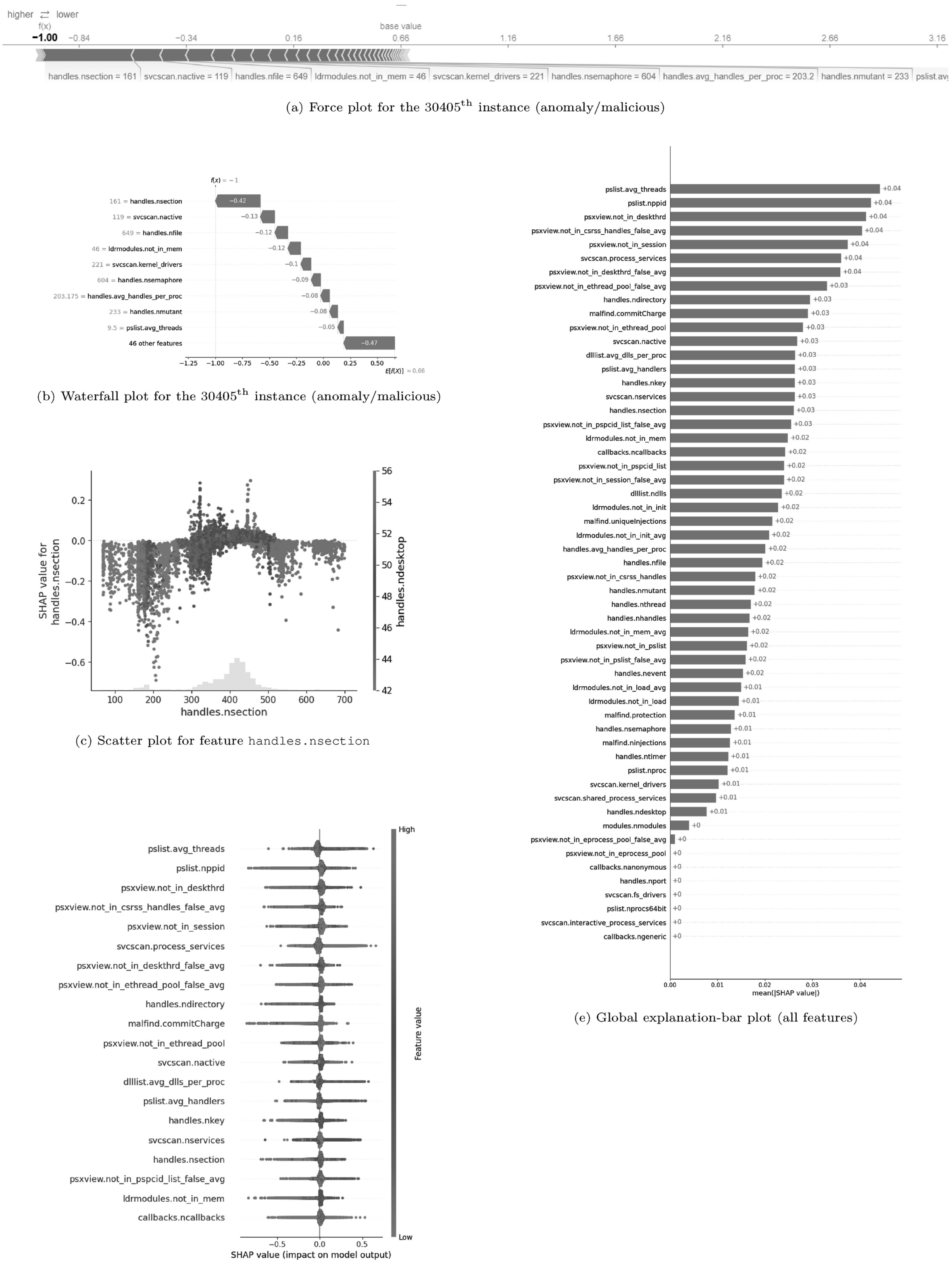
### 4.2. SHAP explanations

SHAP explanations for IF anomaly detection based on feature importance are illustrated in Fig. 4. A force plot[7] (Fig. 4a) for the 30405th instance in the dataset which was flagged as an anomaly/malicious (i.e. f(x)[8] = -1) illustrates features that contribute towards its categorization; this is a local explanation.[9] A waterfall plot (Fig. 4b) illustrates the same explanation. These plots suggest that top features for the subject instance were handles.nsection, svcscan.nactive, and

[7] In a force plot, features highlighted in red indicate their contribution to increasing the predicted value, while those in blue represent their influence in decreasing the prediction.

[8] In anomaly detection, f(x) is 1 in case of a benign instance and −1 in case of a malicious/anomaly instance.

[9] *Local explanations* interpret categorization of one instance in a dataset while *global explanations*, accounting for all instances, provide a holistic interpretation.



(a) Force plot for the 30405th instance (anomaly/malicious)

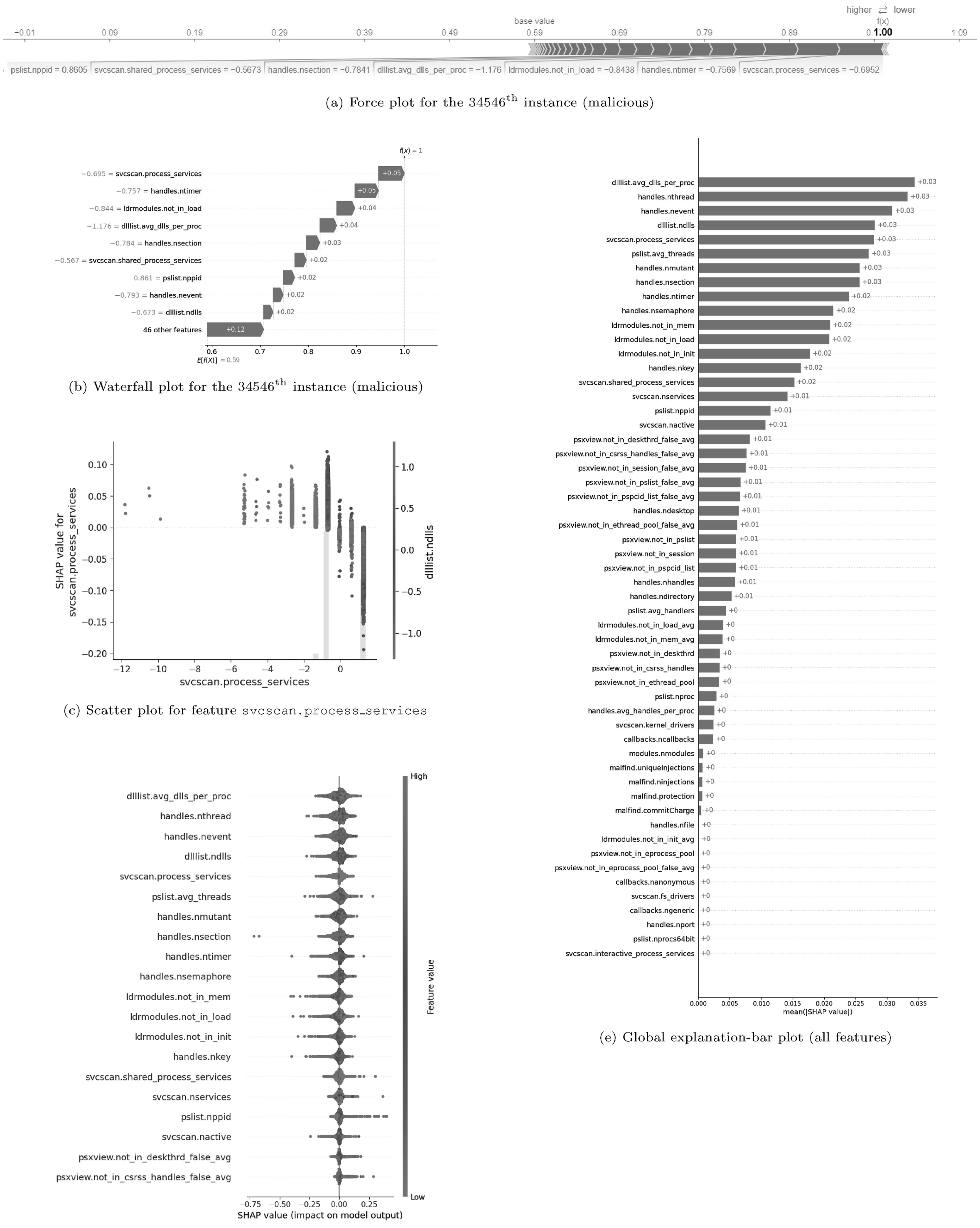
(b) Waterfall plot for the 30405th instance (anomaly/malicious)

(c) Scatter plot for feature `handles.nsection`

(d) Global explanation-summary plot (top 20 features)

(e) Global explanation-bar plot (all features)

**Fig. 4.** SHAP explanations for Isolation Forest anomaly detection.



(a) Force plot for the 34546$^{\mathrm{th}}$ instance (malicious)

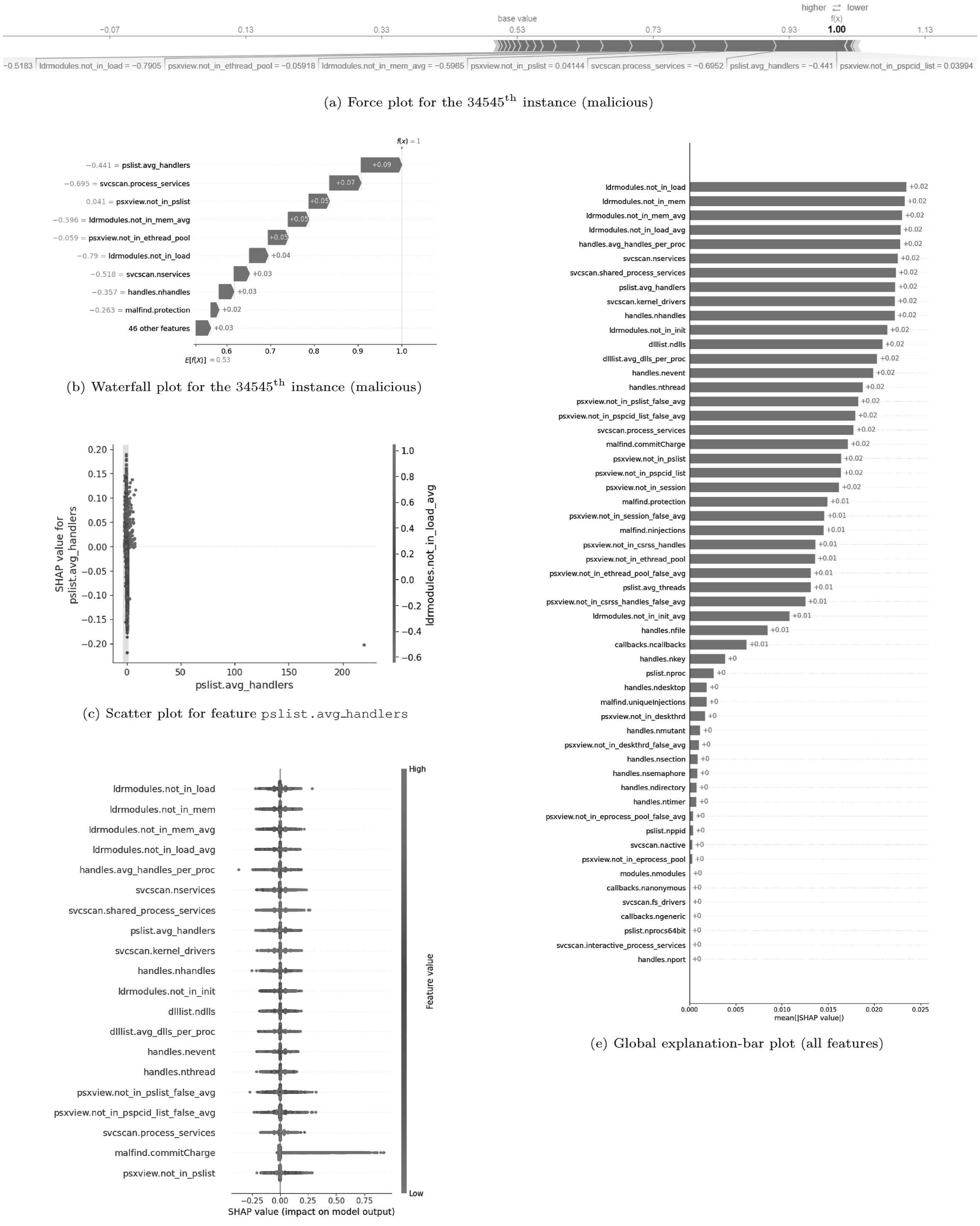
(b) Waterfall plot for the 34546$^{\mathrm{th}}$ instance (malicious)

(c) Scatter plot for feature `svcscan.process_services`

(d) Global explanation-summary plot (top 20 features)

(e) Global explanation-bar plot (all features)

**Fig. 5.** SHAP explanations for K-means (binary) clustering.



(a) Force plot for the 34545th instance (malicious)

(b) Waterfall plot for the 34545th instance (malicious)

(c) Scatter plot for feature `pslist.avg_handlers`

(d) Global explanation-summary plot (top 20 features)

(e) Global explanation-bar plot (all features)

**Fig. 6.** SHAP explanations for GMM (binary) clustering.

handles.nfile. A scatter(/dependence) plot of the top feature for the instance can be used to gauge how fluctuation in the feature's value relates to SHAP values and, by extension, impacts its categorization as an anomaly. The vertical spread of SHAP values reflects the relative influence of each feature on the categorization. Fig. 4c illustrates the scatter plot of handles.nsection. The summary and bar plots illustrated in Fig. 4d and e are global explanations providing insight into features that are top determinants in anomaly detection holistically i.e. pslist. avg_threads, pslist.nppid, and psxiew.not_in_deskthrd (these have the highest mean absolute SHAP values). The bar plot also highlights around 7 features with close to no contribution; these may be removed for further analysis as part of dimensionality reduction to get better results.

SHAP explanations for K-means (binary) clusters are illustrated in Fig. 5. A force plot (Fig. 5a) for the 34546th instance in the dataset which was accurately clustered to the malware category is denoted by the value 1 for f(x)[10]) Fig. 5b illustrates the same explanation as a waterfall plot. According to these explanations, the svcscan.process_services, handles.ntimer, ldrmodules.not_in_load features are top determinants for the 34546th instance. The svcscan.process_services feature's contribution, in particular, can be viewed via a scatter plot (Fig. 5c). The distribution is fragmented, showing a clear separation between feature values. The summary and bar plots illustrated in Fig. 5d and e highlight top features globally: dlllist.av_dlls_per_proc, handles.nthread, handles. nevent. The bar plot indicates around 10 features with close to no contribution which may be removed.

GMM (binary) explanations are illustrated in Fig. 6. Fig. 6a and b shows force and waterfall plots for the 34545th instance in the dataset categorized malicious. A scatter plot for the top feature (pslist. avg_handlers) in these local explanations (Fig. 6c) exhibits a linear trend across its range of SHAP values and a quite wide vertical spread. According to the summary and bar plots illustrated in Fig. 6d and e, the top features globally are ldrmodules.not_in_load, ldrmodules.not_in_mem, and ldrmodules.not_in_mem_avg. The 7 features with no contribution according to the bar plot can be removed to increase ARIs in further testing.

These visualization techniques for interpretable explanations provide clear insights and transparency into why specific memory instances (running processes) are classified as malicious or benign based on behavioral patterns, API calls, and memory usage anomalies. As malware constantly evolves, periodic model updates and validation with recent forensic datasets are recommended. These explanations may not only bridge the knowledge gap between technical and non-technical personnel but also assist forensic experts in refining detection strategies. Forensic analysts can validate model predictions, while decision-makers without deep technical expertise can better understand the reasoning behind AI predictions. Integrating explainability into DF workflows ultimately improves model reliability, facilitates more informed responses, and strengthens proactive threat mitigation.

## 5. Conclusion and future work

This study demonstrates the potential of unsupervised learning-based Explainable AI (XAI) methodologies in Digital Forensics (DF), focusing on memory forensics to detect and cluster obfuscated malware. By employing Isolation Forest, Autoencoder, K-means, DBSCAN, and Gaussian Mixture Models (GMM) on the CIC MalMemAnalysis-2022 dataset, we effectively illustrated anomaly detection and clustering outcomes across binary and multivariate levels (4 and 16 categories). The integration of SHAP explanations provided both local and global interpretability through visualization techniques, aiming for algorithmic transparency and practical applicability in DF workflows. Despite these promising results, several challenges remain. Future work can expand this research to other types of forensic datasets, such as network traffic or disk images, to validate the generalizability of the methodology. Also, automated pipelines to integrate unsupervised XAI workflows into end-to-end DF tools may be developed, reducing reliance on expert intervention. These contributions will ensure that DF continues to benefit from AI-driven insights without compromising on interpretability or reliability.

## Acknowledgement

This study is supported by Research Incentive Funds (activity codes: R23064, R21096), Research Office, Zayed University, Dubai, United Arab Emirates.

## References

Brito, L.C., Susto, G.A., Brito, J.N., Duarte, M.A.V., 2022. An explainable artificial intelligence approach for unsupervised fault detection and diagnosis in rotating machinery. Mech. Syst. Signal Process. 163, 108105. https://doi.org/10.1016/j.ymssp.2021.108105.

Carrier, T., Victor, P., Tekeoglu, A., Lashkari, A., 2022. Detecting obfuscated malware using memory feature engineering. In: Proceedings of the 8th International Conference on Information Systems Security and Privacy. https://doi.org/10.5220/0010908200003120.

Dener, M., Ok, G., Orman, A., 2022. Malware detection using memory analysis data in big data environment. Appl. Sci. 12, 8604. https://doi.org/10.3390/app12178604.

Dunsin, D., Ghanem, M.C., Ouazzane, K., 2022. The use of artificial intelligence in digital forensics and incident response in a constrained environment. International Journal of Information and Communication Engineering 16, 280–285.

Hall, S.W., Sakzad, A., Choo, K.R., 2021. Explainable artificial intelligence for digital forensics. WIREs Forensic Science 4. https://doi.org/10.1002/wfs2.1434.

Hall, S.W., Amin, Sakzad, Minagar, Sepehr, 2022. A proof of concept implementation of explainable artificial intelligence (XAI) in digital forensics. Lect. Notes Comput. Sci. 66–85. https://doi.org/10.1007/978-3-031-23020-2_4.

Khalid, Z., Iqbal, F., Fung, B.C.M., 2024. Towards a unified XAI-based framework for digital forensic investigations. Forensic Sci. Int.: Digit. Invest. 50, 301806. https://doi.org/10.1016/j.fsidi.2024.301806.

Lundberg, S., Lee, S.-I., 2017. A Unified Approach to Interpreting Model Predictions arXiv:1705.07874 [cs, stat].

Mezina, A., Burget, R., 2022. Obfuscated malware detection using dilated convolutional network. In: 14th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT). https://doi.org/10.1109/ICUMT57764.2022.9943443.

Montavon, G., Kauffmann, J., Samek, W., Müller, Klaus-Robert, 2022. Explaining the predictions of unsupervised learning models. Lect. Notes Comput. Sci. 117–138. https://doi.org/10.1007/978-3-031-04083-2_7.

Morichetta, A., Casas, P., Mellia, M., 2019. EXPLAIN-IT. arXiv (Cornell University). https://doi.org/10.1145/3359992.3366639.

Öztürk, A., Hızal, S., 2024. Detection and Analysis of Malicious Software Using Machine Learning Models. Sakarya University Journal of Computer and Information Sciences. …1489237. https://doi.org/10.35377/saucis…1489237.

Solanke, A.A., 2022. Explainable digital forensics AI: towards mitigating distrust in AI-based digital forensics analysis using interpretable models. Forensic Sci. Int.: Digit. Invest. 42, 301403. https://doi.org/10.1016/j.fsidi.2022.301403.

Solanke, A.A., Biasiotti, M.A., 2022. Digital forensics AI: evaluating, standardizing and optimizing digital evidence mining techniques. KI - Künstliche Intelligenz 36. https://doi.org/10.1007/s13218-022-00763-9.

Wickramasinghe, C.S., Amarasinghe, K., Marino, D.L., Rieger, C., Manic, M., 2021. Explainable unsupervised machine learning for cyber-physical systems. IEEE Access 9, 131824–131843. https://doi.org/10.1109/access.2021.3112397.

Won Oh, S., Seon Jo, H., Jun Lee, H., Gyun Na, M., Oh, S.W., Jo, H.S., Lee, H.J., Na, M.G., 2022. Anomalies detection by unsupervised learning using explainable artificial intelligence in nuclear power plants. In: Transactions of the Korean Nuclear Society Spring Meeting Jeju, Korea.

[10] In cluster analysis, f(x) is 0 in case of a benign instance and 1 in case of a malicious instance.