# Sparse Non-Linear Vector Autoregressive Networks for Multivariate Time Series Anomaly Detection

Mohammed Ayalew Belay, *Graduate Student Member, IEEE*, Adil Rasheed, and Pierluigi Salvo Rossi, *Senior Member, IEEE*

***Abstract*—Anomaly detection in multivariate time series (MTS) is crucial in domains such as industrial monitoring, cybersecurity, healthcare, and autonomous driving. Deep learning approaches have improved anomaly detection but lack interpretability. We propose an explainable anomaly detection (XAD) framework using a sparse non-linear vector autoregressive network (SNL-VAR-Net). This framework combines neural networks with vector autoregression for non-linear representation learning and interpretable models. We employ regularization to enforce sparsity, enabling efficient handling of long-range dependencies. Additionally, augmented Lagrange multiplier-based techniques for low-rank and sparse decomposition reduce the impact of noise. Evaluation on publicly available datasets shows that SNL-VAR-Net offers comparable performance to deep learning methods with better interpretability.**

***Index Terms*—Explainable anomaly detection (XAD), low-rank decomposition, multivariate time series, vector autoregression.**

## I. Introduction

MULTIVARIATE time series (MTS) capture the temporal evolution of various physical parameters and are vital for system monitoring, such as anomaly detection and prediction. MTS anomaly detection is crucial in industrial monitoring [1], [2], [3], network intrusion detection [4], [5], wireless sensor networks [6], medical applications [7], and autonomous vehicles [8]. In industrial contexts like manufacturing and power plants, MTS anomaly detection helps identify system faults, predict equipment failures, and improve operational efficiency. In network management, focusing on unusual traffic patterns indicating potential cyberattacks is essential for deploying digital technologies in safety-critical applications. Recent advancements in MTS anomaly detection leverage machine learning, with a shift towards unsupervised methods due to challenges in supervised and semi-supervised approaches [9], [10]. In addition, several self-supervised methods, such as autoregression and contrastive learning, are proposed for MTS anomaly detection and sequential modeling [11], [12], [13].

Unsupervised deep learning models efficiently capture complex temporal patterns and are categorized into reconstruction, prediction, and generative models based on their anomaly detection strategies [14]. Despite their performance, methods based on deep neural networks (DNN) lack interpretability, which is crucial for safety-critical applications. Explainable AI (XAI) aims to bridge this gap, focusing on performance and interpretability [15]: methods for explainable anomaly detection (XAD) are categorized into pre-model, in-model, and post-model techniques [16]. Pre-model techniques rank feature importance, in-model techniques incorporate interpretability directly, and post-model techniques explain model decisions post-prediction, with LIME [17] and SHAP [18] being prominent examples. These methods enhance understanding of complex models, making them suitable for critical applications.

Several recent interpretable models are tailored for supervised XAD and do not handle effectively pattern modeling in complex MTS. They often overlook the presence of noise and outliers in real-world datasets, which can impact the training procedure and system performance during operation. In this letter, we propose a framework for unsupervised XAD in MTS based on a sparse non-linear vector autoregressive network (SNL-VAR-Net). The SNL-VAR-Net architecture integrates low-rank and sparse decomposition for noise reduction and vector autoregression based on neural networks (NNs), yielding an interpretable model with parameters representing time lags. The contributions of this work are the following:

- We introduce a novel unsupervised XAD method for MTS utilizing a SNL-VAR-Net;
- We employ low-rank and sparse (LRS) decomposition techniques to minimize the impact of noisy data during the training process, solved with the augmented Lagrange multiplier (ALM) method;
- We include sparse regularization to enforce sparsity and handle efficiently MTS with long-range dependencies;
- We evaluate the performance using two publicly-available datasets related to real-world multi-sensor systems and compare it with relevant baselines.

The rest of the letter is organized as follows: the problem of interest and the related mathematical notation are introduced in

Received 6 August 2024; revised 15 December 2024; accepted 16 December 2024. Date of publication 18 December 2024; date of current version 27 December 2024. This work was supported by the Research Council of Norway under the project DIGITAL TWIN within the PETROMAKS2 framework through Project under Grant 318899. The associate editor coordinating the review of this article and approving it for publication was Dr. Yuanfang Guo. *(Corresponding author: Mohammed Ayalew Belay.)*

Mohammed Ayalew Belay is with the Department of Electronic Systems, Norwegian University of Science and Technology, 7034 Trondheim, Norway (e-mail: mohammed.a.belay@ntnu.no).

Adil Rasheed is with the Department of Engineering Cybernetics, Norwegian University of Science and Technology, 7034 Trondheim, Norway (e-mail: adil.rasheed@ntnu.no).

Pierluigi Salvo Rossi is with the Department of Electronic Systems, Norwegian University of Science and Technology, 7034 Trondheim, Norway, and also with the Department of Gas Technology, SINTEF Energy Research, 7491 Trondheim, Norway (e-mail: salvorossi@ieee.org).

Digital Object Identifier 10.1109/LSP.2024.3520019

Section II; Section III provides the mathematical details of the proposed methodology; the experimental setup and the datasets considered for model validation are presented in Section IV; Section V shows the numerical results for performance analysis and related discussion; finally, conclusions and future research directions are given in Section VI.[1]

## II. Problem Statement

We consider a MTS with $K$ components, with $x_k[n] \in \mathbb{R}$ denoting the value of the $k$th univariate component at discrete time $n$. The *data vector* $\boldsymbol{x}[n] = (x_1[n], x_2[n], \ldots, x_K[n])^T \in \mathbb{R}^K$ collects all the components at time $n$, and the collection of data vectors related to $N$ discrete times steps is arranged into the *data matrix* $\boldsymbol{X} = (\boldsymbol{x}[1], \boldsymbol{x}[2], \ldots, \boldsymbol{x}[N]) \in \mathbb{R}^{K \times N}$.

A training data matrix ($\boldsymbol{X}_{\text{train}}$) represents the MTS observations under normal conditions. The proposed framework aims at characterizing normal MTS behavior and detecting deviations. A test data matrix ($\boldsymbol{X}_{\text{test}} \in \mathbb{R}^{K \times M}$) including MTS from both normal and anomalous conditions paired with a *label vector* $\boldsymbol{y} = (y_1, y_2, \ldots, y_M)^T$, with $y_m = 1$ (resp. $y_m = 0$) denoting the presence (resp. absence) of an anomaly at discrete time $m$, are used for performance evaluation.

## III. Anomaly Detection

### A. Low-Rank and Sparse Decomposition

We employ LRS decomposition to mitigate the impact of noisy training data. The data matrix is decomposed into a structured low-rank component ($\boldsymbol{L} \in \mathbb{R}^{K \times N}$) representing the normal behavior[2] and a sparse component ($\boldsymbol{S} \in \mathbb{R}^{K \times N}$) representing anomalies and/or noise:

$$\boldsymbol{X} = \boldsymbol{L} + \boldsymbol{S} \, . \tag{1}$$

The non-convex formulation of the LRS decomposition minimizes the rank of the low-rank matrix ($\boldsymbol{L}$) and the number of non-zero elements of the sparse matrix ($\boldsymbol{S}$) [19], [20]:

$$\min_{\boldsymbol{L},\boldsymbol{S}} \ \text{rank}(\boldsymbol{L}) + \|\boldsymbol{S}\|_0 \, , \quad \text{s.t.} \quad \boldsymbol{L} + \boldsymbol{S} = \boldsymbol{X} \, . \tag{2}$$

However, we consider a surrogate convex relaxation approach

$$\min_{\boldsymbol{L},\boldsymbol{S}} \ \|\boldsymbol{L}\|_* + \lambda \|\boldsymbol{S}\|_1, \quad \text{s.t.} \quad \boldsymbol{L} + \boldsymbol{S} = \boldsymbol{X} \, , \tag{3}$$

where $\lambda$ is a regularization parameter that trades-off the low-rank and sparse components. We use the ALM method as:

$$\mathcal{J}(\boldsymbol{L}, \boldsymbol{S}, \boldsymbol{Y}, \rho) = \|\boldsymbol{L}\|_* + \lambda \|\boldsymbol{S}\|_1 + \langle \boldsymbol{Y}, \boldsymbol{X} - \boldsymbol{L} - \boldsymbol{S} \rangle + \frac{\rho}{2} \|\boldsymbol{X} - \boldsymbol{L} - \boldsymbol{S}\|_F^2 \, , \tag{4}$$

where $\boldsymbol{Y} \in \mathbb{R}^{K \times N}$ is the matrix of Lagrange multipliers and $\rho > 0$ is a penalty parameter.

[1] *Notation* – vectors and matrices are denoted with bold lower-case and bold upper-case letters, respectively; rank, and transpose operators are represented as $\text{rank}(\cdot)$ and $(\cdot)^{\mathrm{T}}$; $\|\cdot\|_j$, $\|\cdot\|_*$, $\|\cdot\|_{\mathrm{F}}$, and $\langle \cdot, \cdot \rangle$ denote the $\ell_j$-norm, the nuclear norm, the Frobenius norm, and the Frobenius inner product.

[2] Analogously to $\boldsymbol{X}$, we denote $\boldsymbol{l}[n] \in \mathbb{R}^K$ the column of the low-rank matrix $\boldsymbol{L}$ representing the low-rank representation of $\boldsymbol{x}[n]$.

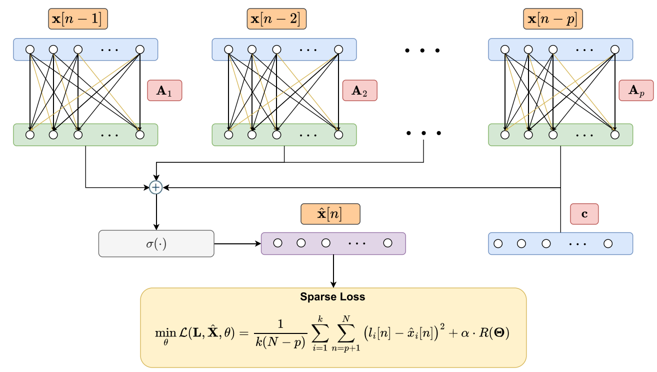
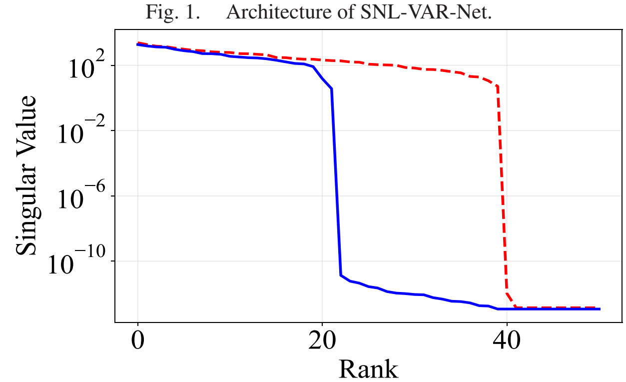
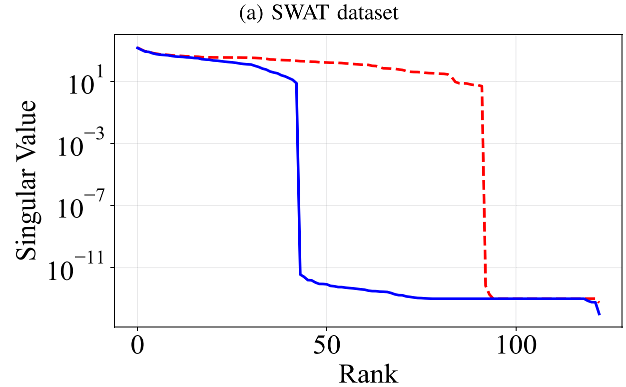



Fig. 1. Architecture of SNL-VAR-Net.




(a) SWAT dataset




(b) WADI dataset

Fig. 2. Scree plot of the original (dashed lines) and low-rank (solid lines) data representations.

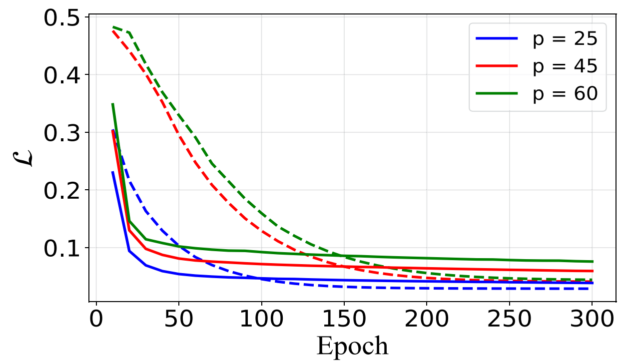
### B. Snl-Var-Net

The proposed architecture is shown in Fig. 1. To compute data-vector estimate ($\hat{\boldsymbol{x}}[n]$), SNL-VAR-Net combines vector autoregression with neural networks:

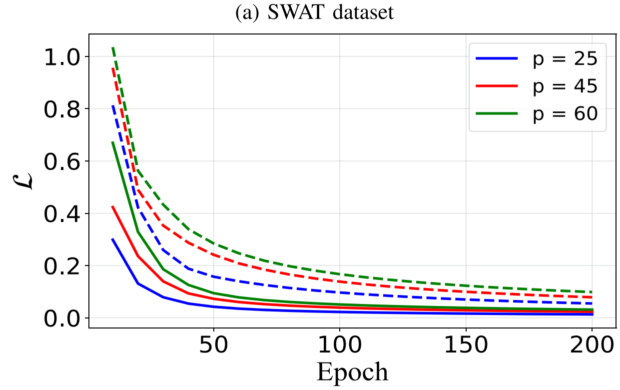
$$\hat{\boldsymbol{x}}[n] = \sigma \left( \sum_{\ell=1}^{p} \boldsymbol{A}_\ell \boldsymbol{x}[n - \ell] + \boldsymbol{c} \right) \, , \tag{5}$$

where $\sigma(\cdot)$ is a nonlinear activation function,[3] $p$ is the order of the vector autoregression, $\boldsymbol{A}_\ell \in \mathbb{R}^{K \times K}$ is the coefficients matrix for the $\ell$th lag, and $\boldsymbol{c} \in \mathbb{R}^K$ is a vector of constants. Equation (5) provides model interpretability since each coefficient matrix

[3] We use a sigmoid activation function.

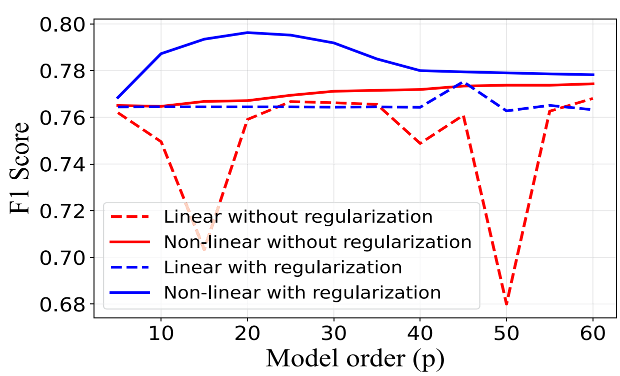



(a) SWAT dataset

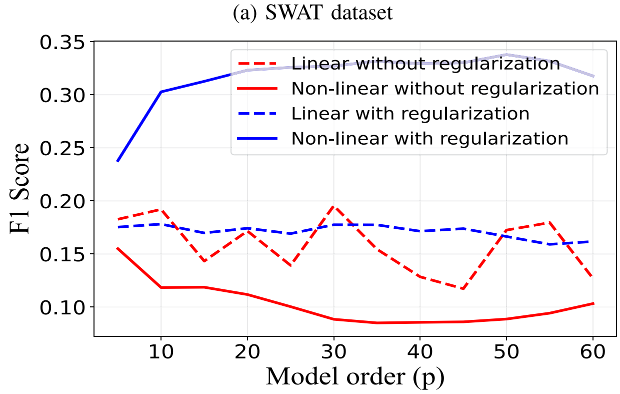



(b) WADI dataset

Fig. 3. Training loss without regularization (dashed lines) and with regularization (solid lines) for various model orders.

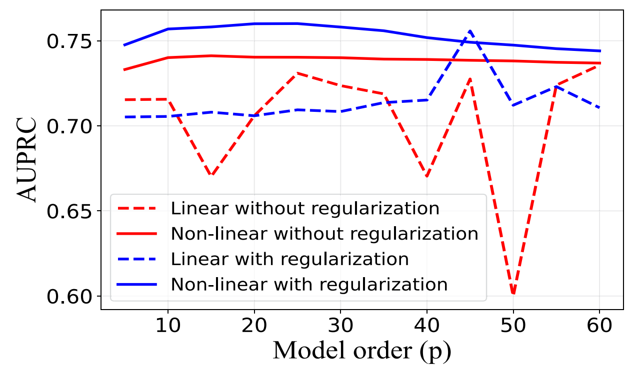



(a) SWAT dataset

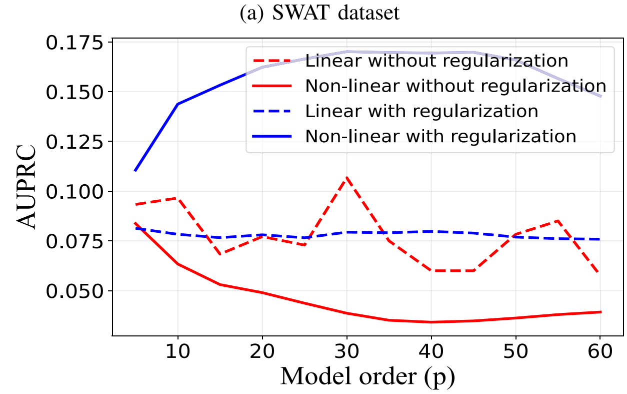



(b) WADI dataset

Fig. 4. Impact on F1-score of the non-linear activation function and of the regularization term for various model orders.




(a) SWAT dataset




(b) WADI dataset

Fig. 5. Impact on AUPRC of the non-linear activation function and of the regularization term for various model orders.

($\boldsymbol{A}_\ell$) is responsible for the signal dynamic related to a specific lag ($\ell$).

The model parameters ($\boldsymbol{\Theta} \equiv \{\boldsymbol{A}_1, \boldsymbol{A}_2, \ldots, \boldsymbol{A}_p, \boldsymbol{c}\}$) are learned by replacing the data vector ($\boldsymbol{x}[n]$) with its low-rank representation ($\boldsymbol{l}[n]$) in the loss function, which combines mean square error (MSE) with sparse regularization:

$$\mathcal{L}(\boldsymbol{\Theta}) = \mathcal{L}_{MSE}(\boldsymbol{\Theta}) + \alpha \cdot R(\boldsymbol{\Theta}) \ , \tag{6}$$

where $\mathcal{L}_{MSE}(\boldsymbol{\Theta}) = \frac{1}{K(N-p)} \sum_{k=1}^{K} \sum_{n=p+1}^{N} (l_k[n] - \hat{x}_k[n])^2$, $\alpha$ is a regularization parameter, and $R(\cdot)$ is a regularization term computed as the sum of all the costs ($c_i$) associated with each parameter $\theta_i \in \boldsymbol{\Theta}$ according to a predefined function.[4]

SNL-VAR-Net computes the anomaly score as the mean prediction error using the $\ell_2$-norm. The anomaly score for a given test data point ($\boldsymbol{x}[n]$) is given by $s_n = \|\boldsymbol{x}[n] - \hat{\boldsymbol{x}}[n]\|_2$. Anomaly detection ($\hat{y}_m$) is performed based on a threshold-based rule[5] applied to the anomaly score ($s_m$), i.e.

$$\hat{y}_m = \begin{cases} 1 & s_m > \tau^* \\ 0 & s_m \leq \tau^* \end{cases} \ . \tag{7}$$

[4]We use $c_i = \log(1 + \gamma|\theta_i|)$ with $\gamma$ being a tunable hyperparameter.

[5]We assumed $\tau^* = \hat{\mu}_s + 3\hat{\sigma}_s$, where $\hat{\mu}_s$ and $\hat{\sigma}_s^2$ represent the maximum-likelihood estimates of mean and variance of the training anomaly score.

## IV. Experimental Setup

We conducted our experiments using two publicly available MTS datasets: Secure Water Treatment (SWAT) [21], [22] and Water Distribution (WADI) [23]. SWAT was collected from a testbed that simulates the physical process and control system of a real-world water treatment system, and includes readings from 51 sensors over 11 days of operation, with a sampling interval of 1 s. WADI was collected from a testbed that expands SWAT to a comprehensive network for water treatment, storage, and distribution, and includes readings from 123 sensors over approximately 16 days, with a sampling interval of 1 s. We assumed stationarity in the data, which is critical for the autoregressive components of the model. For nonstationary datasets, preprocessing techniques (e.g., differencing, detrending, seasonal decomposition) must be applied.

We use binary-classification metrics suitable for scenarios where anomalous samples are less frequent than normal ones: $F_1$ score and area under the precision-recall curve (AUPRC) [14]. We perform a comprehensive comparison with state-of-the-art unsupervised anomaly detection algorithms: Isolation Forest (IF) [24]; Gaussian Mixture Model (GMM) [25]; Empirical Cumulative Distribution Functions (ECOD) [26]; Copula-Based Outlier Detection (COPOD) [27]; Multi-layer Perceptron Autoencoders (AE) [28]; Variational Autoencoders (VAE) [29]; Deep Support Vector Data Description (DeepSVDD) [30]; Adversarially Learned Anomaly Detection (ALAD) [31]. UnSupervised Anomaly Detection (USAD) [32]; Deep Autoencoding Gaussian Mixture Model (DAGMM) [33]. Data preprocessing involved downsampling and min-max normalization. For the downsampled test data, labels are determined such that a sample is an anomaly if the window contains at least a single anomalous measurement. We utilized PyTorch deep learning to implement SNL-VAR-Net.

TABLE I
Performance Comparison With Baselines

| Models | SWaT | | WADI | |
|---|---|---|---|---|
| | $F_1$ | $AUPR$ | $F_1$ | $AUPR$ |
| ECOD | 0.7409 | 0.7571 | 0.2967 | 0.1615 |
| COPOD | 0.7475 | 0.7585 | 0.2986 | 0.1588 |
| IF | 0.6880 | 0.5033 | 0.2026 | 0.0987 |
| GMM | 0.5152 | 0.3757 | - | 0.0987 |
| AE | 0.7660 | 0.7267 | 0.1091 | 0.0488 |
| VAE | 0.7660 | 0.7267 | 0.1091 | 0.0488 |
| DeepSVDD | 0.6053 | 0.6132 | 0.0840 | 0.0387 |
| ALAD | 0.2176 | 0.0884 | 0.0887 | 0.0328 |
| USAD | 0.8046 | 0.7031 | 0.6763 | 0.1196 |
| DAGMM | 0.8017 | 0.6917 | 0.7033 | 0.1359 |
| SNL-VAR-Net | 0.7963 | 0.7599 | 0.3379 | 0.1659 |

## V. Results and Discussions

The performance of the ALM-based LRS decomposition for both SWAT and WADI datasets is shown in Fig. 2. When applying LRS decomposition, the singular values for both datasets exhibit lower magnitude and a smoother descent, indicating a reduction of noise and fewer significant components. Furthermore, the flattened tail in the decomposed singular values suggests that LRS decomposition suppresses less informative variations attributed to outliers and/or noise.

The training loss convergence for the SNL-VAR-Net framework is presented in Fig. 3. We compared convergence across various model order selections ($p = 15, 30, 45$) in the case in which the regularization term is not included and in the case in which regularization with logarithmic regularization function. The regularization improves the convergence rate of the training loss, with a bigger impact for higher model orders.

The impact of the non-linear activation function and regularization term introduced in the SNL-VAR-Net framework is shown in Figs. 4 and 5 via the metrics F1-score and AUPRC, respectively. The results show how the non-linear activation function stabilizes the performance with respect to the model order ($p$), thus making its proper selection not necessary. Also, the regularization term provides relevant gains, thus making more efficient use of the sparse coefficients. The proposed SNL-VAR-Net consistently outperforms other variants over both SWAT and WADI datasets by exploiting both the non-linearity and the sparse regularization.

The comparison of the SNL-VAR-Net model against recently proposed baseline anomaly detection methods on both the SWAT and WADI datasets is presented in Table I. SNL-VAR-Net outperforms alternative state-of-the-art solutions. Although this result cannot be considered a proof of superiority in terms of performance, it is apparent how the proposed method is a valid solution with the additional desirable characteristic of being easily interpretable via the autoregressive architecture which it relies on.

## VI. Conclusion

We presented a novel and interpretable unsupervised anomaly detection approach for MTS data. The proposed framework employs NN-based vector autoregression with SLR decomposition regularization. The enforced sparsity enables efficient handling of datasets with long-range dependencies and avoids model order selection. In addition, to reduce the impact of noise on anomaly detection, augmented Lagrange multiplier-based techniques were integrated for low-rank and sparse decomposition. We conducted a performance evaluation using two publicly available datasets and several baseline methods. The results demonstrated that the proposed SNL-VAR-Net achieves effective anomaly detection while providing interpretability. The importance of employing both non-linearity in the vector autoregressive model and LRS decomposition was shown. Future work will focus on extending SNL-VAR-Net to real-time systems, exploring different low-rank approximation methods and applying them to specific domain applications such as network intrusion detection and industrial monitoring, additional interprebility analysis, and investigate non-stationary issues.

## REFERENCES


[1] S. Yin, S. X. Ding, X. Xie, and H. Luo, "A review on basic data-driven approaches for industrial process monitoring," *IEEE Trans. Ind. Electron.*, vol. 61, no. 11, pp. 6418–6428, Nov. 2014.

[2] H. Nizam, S. Zafar, Z. Lv, F. Wang, and X. Hu, "Real-time deep anomaly detection framework for multivariate time-series data in industrial IoT," *IEEE Sensors J.*, vol. 22, no. 23, pp. 22836–22849, Dec. 2022.

[3] M. A. Belay, A. Rasheed, and P. Salvo Rossi, "MTAD: Multi-objective transformer network for unsupervised multisensor anomaly detection," *IEEE Sensors J.*, vol. 24, no. 12, pp. 20254–20265, Jun. 2024.

[4] P. García-Teodoro, J. Díaz-Verdejo, G. Maciá-Fernández, and E. Vázquez, "Anomaly-based network intrusion detection: Techniques, systems and challenges," *Comput. Secur.*, vol. 28, no. 1/2, pp. 18–28, 2009.

[5] M. H. Bhuyan, D. K. Bhattacharyya, and J. K. Kalita, "Network anomaly detection: Methods, systems and tools," *IEEE Commun. Surveys Tuts.*, vol. 16, no. 1, pp. 303–336, Firstquarter 2014.

[6] S. Rajasegarar, C. Leckie, and M. Palaniswami, "Anomaly detection in wireless sensor networks," *IEEE Wireless Commun.*, vol. 15, no. 4, pp. 34–40, Aug. 2008.

[7] A. Ukil, S. Bandyoapdhyay, C. Puri, and A. Pal, "IoT healthcare analytics: The importance of anomaly detection," in *Proc. Int. Conf. Adv. Inf. Netw. Appl.*, 2016, pp. 994–997.

[8] F. V. Wyk, Y. Wang, A. Khojandi, and N. Masoud, "Real-time sensor anomaly detection and identification in automated vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 3, pp. 1264–1276, Mar. 2020.

[9] N. Larosa, J. Farber, P. Venkitasubramaniam, R. Blum, and A. A. Rashdan, "Separating sensor anomalies from process anomalies in data-driven anomaly detection," *IEEE Signal Process. Lett.*, vol. 29, pp. 1704–1708, 2022.

[10] J. Chen, X. Tan, S. Rahardja, J. Yang, and S. Rahardja, "Joint selective state space model and detrending for robust time series anomaly detection," *IEEE Signal Process. Lett.*, vol. 31, pp. 2050–2054, 2024.

[11] M. A. Belay, A. Rasheed, and P. Salvo Rossi, "Self-supervised modular architecture for multi-sensor anomaly detection and localization," in *Proc. IEEE Conf. Artif. Intell.*, 2024, pp. 1278–1283.

[12] Y. Yang, C. Huang, L. Xia, C. Huang, D. Luo, and K. Lin, "Debiased contrastive learning for sequential recommendation," in *Proc. ACM Web Conf. 2023 - Proc. World Wide Web Conf.*, 2023, pp. 1063–1073.

[13] Z. Wang et al., "Multi-level contrastive learning framework for sequential recommendation," in *Proc. Int. Conf. Inf. Knowl. Manage.*, 2022, pp. 2098–2107.

[14] M. A. Belay, S. S. Blakseth, A. Rasheed, and P. Salvo Rossi, "Unsupervised anomaly detection for IoT-Based multivariate time series: Existing solutions, performance analysis and future directions," *Sensors*, vol. 23, no. 5, 2023, Art. no. 2844.

[15] C. Rudin, "Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead," *Nat. Mach. Intell.*, vol. 1, no. 5, pp. 206–215, 2018.

[16] Z. Li, Y. Zhu, and M. Van Leeuwen, "A survey on explainable anomaly detection," *ACM Trans. Knowl. Discov. Data*, vol. 18, no. 1, 2023, Art. no. 23.

[17] U. Schlegel, D. L. Vo, D. A. Keim, and D. Seebacher, "TS-MULE: Local interpretable model-agnostic explanations for time series forecast models," *Commun. Comput. Inf. Sci.*, vol. 1524 CCIS, pp. 5–14, 2021. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-030-93736-2_1

[18] L. Antwarg, R. M. Miller, B. Shapira, and L. Rokach, "Explaining anomalies detected by autoencoders using Shapley additive explanations," *Expert Syst. Appl.*, vol. 186, 2021, Art. no. 115736.

[19] M. A. Belay, A. Rasheed, and P. S. Rossi, "Multivariate time series anomaly detection via low-rank and sparse decomposition," *IEEE Sensors J.*, vol. 24, no. 21, pp. 34942–34952, Nov. 2024.

[20] M. Rahmani and G. K. Atia, "High dimensional low rank plus sparse matrix decomposition," *IEEE Trans. Signal Process.*, vol. 65, no. 8, pp. 2004–2019, Apr. 2017.

[21] J. Goh, S. Adepu, K. N. Junejo, and A. Mathur, "A dataset to support research in the design of secure water treatment systems," *Crit. Inf. Infrastructures Secur.*, vol. 10242, pp. 88–99, 2017.

[22] P. M. Aditya and O. T. Nils, "SWaT: A. water treatment testbed for research and training on ICS security," in *Proc. 2016 Int. Workshop Cyber-Phys. Syst. Smart Water Netw.*, 2016, pp. 31–36.

[23] C. M. Ahmed, V. R. Palleti, and A. P. Mathur, "WADI: A water distribution testbed for research in the design of secure cyber physical systems," in *Proc. Int. Workshop Cyber- Phys. Syst. Smart Water Netw.*, 2017, pp. 25–28.

[24] F. T. Liu, K. M. Ting, and Z. H. Zhou, "Isolation-based anomaly detection," *ACM Trans. Knowl. Discov. Data*, vol. 6, no. 1, pp. 1–39, 2012.

[25] C. C. Aggarwal, *Probabilistic and Statistical Models for Outlier Detection*. Berlin, Germany: Springer, 2017, pp. 35–64.

[26] Z. Li, Y. Zhao, X. Hu, N. Botta, C. Ionescu, and G. H. Chen, "ECOD: Unsupervised outlier detection using empirical cumulative distribution functions," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 12, pp. 12181–12193, Dec. 2023.

[27] Z. Li, Y. Zhao, N. Botta, C. Ionescu, and X. Hu, "COPOD: Copula-based outlier detection," in *Proc. IEEE Int. Conf. Data Mining*, 2020, pp. 1118–1123.

[28] M. Sakurada and T. Yairi, "Anomaly detection using autoencoders with nonlinear dimensionality reduction," in *Proc. ACM Int. Conf. Proc. Ser.*, 2014, pp. 4–11, doi: 10.1145/2689746.2689747.

[29] D. P. Kingma and M. Welling, "An introduction to variational autoencoders," *Foundations Trends Mach. Learn.*, vol. 12, no. 4, pp. 307–392, 2019, doi: 10.1561/2200000056.

[30] L. Ruff et al., "Deep one-class classification," in *Proc. Int. Conf. Mach. Learn.*, J. Dy and A. Krause, Eds., 2018, vol. 80, pp. 4393–4402.

[31] H. Zenati, M. Romain, C. S. Foo, B. Lecouat, and V. Chandrasekhar, "Adversarially learned anomaly detection," in *Proc. - IEEE Int. Conf. Data Mining*, 2018, pp. 727–736.

[32] A. Julien, M. Pietro, G. Frédéric, M. Sébastien, and A. Z. Maria, "USAD: Unsupervised anomaly detection on multivariate time series," in *Proc. 26th ACM SIGKDD Int. Conf. Knowl. Discov. Data Minining*, 2020, vol. 20, pp. 3395–3404. [Online]. Available: https://dl.acm.org/doi/abs/10.1145/3394486.3403392

[33] B. Zong et al., "Deep autoencoding Gaussian mixture model for unsupervised anomaly detection," in *Proc. Int. Conf. Learn. Representations*, 2018.