Software Impacts 26 (2025) 100786

Contents lists available at ScienceDirect

Software Impacts

journal homepage: www.journals.elsevier.com/software-impacts

Original software publication

Screw Process Anomaly Visualization (SPAV): A Python module for local and global machine learning visualizations for screw tightening anomaly detection

Marta Moreno [a], Hugo Rocha [a], André Pilastri [b], Guilherme Moreira [c], Luís Miguel Matos [a,b], Paulo Cortez [a,b,*]

[a] ALGORITMI Centre, Minho University, Guimarães, Portugal
[b] EPMQ, CCG ZGDV Institute, Guimarães, Portugal
[c] Bosch Car Multimedia, Braga, Portugal

ARTICLE INFO

Keywords:
Anomaly detection
eXplainable Artificial Intelligence (XAI)
Industry 4.0
Screw tightening
Visual analytics

ABSTRACT

Modern screwdriver systems generate real-time angle-torque data that form tightening curves that are valuable for quality inspection issues (e.g., detect faulty processes). This work describes the Screw Process Anomaly Visualization (SPAV) Python module, which provides several eXplainable AI (XAI) graphs for Machine Learning (ML) screw tightening results, namely global and local errors, with identification of most probable anomaly angle-torque locations. SPAV integrates seamlessly with the scientific Python ecosystem and is compatible with several ML implementations, including H2O and Keras deep AutoEncoders (AE).

Code metadata

Permanent link to code/repository used for this code version	https://github.com/SoftwareImpacts/SIMPAC-2025-217
Permanent link to Reproducible Capsule	https://codeocean.com/capsule/7905696/tree/v1
Legal Code License	MIT
Code versioning system used	git
Software code languages, tools, and services used	Python
Compilation requirements, operating environments & dependencies	Specified in Codeocean platform
If available Link to developer documentation/manual	https://codeocean.com/capsule/7905696/tree
Support email for questions	luis.matos@dsi.uminho.pt

1. Screw Process Anomaly Visualization (SPAV)

Following the recent industrial revolutions (Industry 4.0 and 5.0), several assembly companies are currently under pressure to lower costs and increasing operational efficiency through Artificial Intelligence (AI) tools [1,2]. In particular, screwdriver systems are often used in the assembly industry (e.g., electronic components), generating hundreds of angle-torque real-time pair values for a single screw tightening. Then, the full tightening curve can be analyzed for quality assessment purposes (e.g., anomaly detection). The screw tightening process is inherently dynamic, requiring frequent updates to defect catalogs and recalibration of screwing zones through human evaluation of tightening curves and quality estimates. Although defects are rare, detailed and automated inspection is essential to prevent faulty assemblies from progressing along the production line.

Currently, screwdriver systems provide a "Good or Fail" (GoF) status based on predefined normal screw historical catalogs, making it susceptible of generating false positives and false negatives. As an alternative, a successful Machine Learning (ML) approach to model screw tightening GoF was proposed in [3,4], assuming only two inputs (angle and torque values) and an unsupervised one-class classification

The code (and data) in this article has been certified as Reproducible by Code Ocean: (https://codeocean.com/). More information on the Reproducibility Badge Initiative is available at https://www.elsevier.com/physical-sciences-and-engineering/computer-science/journals.

* Corresponding author at: ALGORITMI Centre, Minho University, Guimarães, Portugal.
E-mail address: pcortez@dsi.uminho.pt (P. Cortez).

https://doi.org/10.1016/j.simpa.2025.100786
Received 17 August 2025; Received in revised form 15 September 2025; Accepted 17 September 2025

learning, such as Isolation Forests (IF) and deep AutoEncoders (AE). These ML models are trained only with normal examples and when tested with new screw tightening instances, the models are capable of outputting anomaly scores for a real-time angle-torque pair, allowing also to compute an overall abnormal score for the complete tightening process (full screwing cycle). In [4], we have shown how these anomaly scores can be used to generate local eXplainable Artificial Intelligence (XAI) graphs, namely to help in identifying the most probable anomaly angle-torque locations of faulty screw processes (Fig. 4(a) is an example of the XAI graph proposed in [4]).

Recently, there has been a growing research interest of the AI community in the XAI topic. Indeed, there are popular methods (e.g., LIME and SHAP) that provide visualizations that can help in local XAI, the understanding of particular prediction made by a supervised ML model [5]. However, our XAI approach is targeted to a more specific ML goal: the identification of most probable anomaly regions of a screw tightening process that was detected as faulty. Thus, methods such as LIME or SHAP are not directly applicable to this relevant industrial task. Within our knowledge, there is only one other work that performs a similar XAI. In [6], the authors proposed a modified version of LIME applied to Convolutional Neural Networks (CNN) trained on images related with sequential vibration data from rotary machines, aiming to detect anomalies. The visualizations consisted of spectral representations of the CNN and LIME responses, helping to identify anomaly regions in the spectral space. In contrast, our XAI screw tightening approach operates directly on the real-time angle-torque pairs using the one-class ML responses, which are more easily understood by human screw operators. We further note that we perform a pure unsupervised ML and XAI approach. Indeed, when a screw fastening process is considered faulty we have the GoF label ("Fail") but there is no ground truth regarding the specific angle-torque regions that produced such anomalous tightening. As such, it is not possible to objectively benchmark our visualizations with other XAI methods. As an alternative, the proposed visualizations were validated by using human screw operator feedback, as detailed in Section 2.

In this work, we describe the Screw Process Anomaly Visualization (SPAV) Python module, which implements the screw tightening XAI graph used in [4] and also other novel XAI plots. The proposed visualizations can be applied to any screw tightening ML model that has two inputs (angle and toque instant values) and that outputs an anomaly score (the higher the value, the more probable is the anomaly), including both unsupervised (e.g., IF and AE) and supervised learning algorithms (e.g., Random Forests, Neural Networks). For demonstration purposes, the SPAV tool is exemplified with the deep AE algorithm, under two Python ML modeling frameworks: Keras [7] and H2O [8].

Keras offers a greater modeling flexibility by allowing users to customize deep learning algorithms, such as individual layers with different activation functions, data distribution assumptions and layer types, while H2O is an Automated ML (AutoML) tool that focuses on facilitating the model selection and hyperparameter tuning process for non-ML experts. For Keras, the selected AE incorporates a bottleneck layer consisting of a single hidden unit to define the latent space and it is trained using the Adam optimization algorithm with the Mean Absolute Error (MAE) as the loss function, as detailed in [3,4].

To demonstrate the SPAV XAI graphs, we used the open-source Pyscrew dataset [9], focusing specifically on the scenario titled "Assembly Conditions 2" (S04). The AEs were trained by using 500 normal screw tightening processes, with each process being related with hundreds of angle-torque values. Then, the AEs were tested using 256 unseen screw tightening cycles, which contain 64 abnormal processes (a full anomaly tightening example is shown in Fig. 3, which includes around 1,700 angle-torque values).

In terms of the AE screw tightening test predictive results, we report (as an example) the Keras AE Receiver Operating Characteristic (ROC) curve [10] in the left of Fig. 1. The obtained 97% value of the Area Under the Curve (AUC) suggests an excellent AE discrimination. The right of Fig. 1 shows the respective confusion matrix when adopting a selected threshold value of 1.13 (as signaled in the left of Fig. 1, the threshold was optimized in terms of getting the highest True Positive Rate for a low False Positive Rate value). The classification results demonstrate strong performance across both classes. The model achieved an overall accuracy of 98% on the test set, comprising 256 screw tightening processes. For the Normal class, precision and recall were 98% and 99%, respectively, indicating that the model correctly identified nearly all normal instances with minimal false positives. For the Anomaly class, precision was 97% and recall 94%, reflecting a high ability to detect faulty processes while maintaining low misclassification rates. Moreover, the F1-score for the Anomaly class reached 95%, confirming a high quality trade-off between precision and recall. These results suggest that the predictive model is highly reliable for distinguishing between normal and abnormal screw tightening cycles, even in the presence of class imbalance.

The first XAI example is shown in Fig. 2. When using the H2O trained AE, the plot shows all 256 test screw cycles colored in terms of a normalized ($\in [0,1]$) AE anomaly scores. This visualization clearly facilitates the identification of global faulty screw signatures (colored more light green to yellow).

A local XAI example is provided in Fig. 3. In this case, the Keras trained AE was tested using a single (thus local) screw tightening that was abnormal (process ID number #815 with a "Fail" GoF status). The local XAI curve is colored using normalized ($\in [0,1]$) abnormal scores. This graph highlights the initial screwing phase (yellow colored points) as the most probable faulty region, suggesting a deviation from the expected behavior at the initial stage. Two additional local XAI plots are provided in Fig. 4.The left plot (Fig. 4(a)) highlights all points (colored in orange) with an anomaly score higher than a user selected threshold (as an example, the threshold value was set to 5.5), while the right plot (Fig. 4(b)) uses a Kernel Density Estimation (KDE) to estimate the highest abnormal density regions (for the same threshold of 5.5). All the three exemplified local XAI graphs can be instrumental in diagnosing specific faulty deviations within individual executions.

In terms of the computational effort, the proposed visualizations are fast and scale linearly with the number of angle-torque pairs per cycle N: $\mathcal{O}(N)$. For example, on a personal computer (Intel i9 2.3 GHz processor), the Global XAI plot requires a total of 3 s when handling 128 screw cycles and 6 s when processing the full 256 test set (Fig. 2). As for the local XAI visualizations, when using the demonstrated AE model, an average time elapsed of 0,42 s is required to generate each plot (e.g., Fig. 3 or Fig. 4(a)). Thus, the proposed visualizations can be adopted in a real-time industrial analyses.

2. Impacts and potential outcomes

While demonstrated here with the open-source Pyscrew dataset [9], the SPAV Python module is being tested and validated in a automotive assembly company (Bosch Car Multimedia). In effect, research and software is being deployed on the Project n° 179826 with the funding reference: SIFN-01-9999-FN-179826. Using real-world data collected from the company, high quality predictive results were achieved by the adopted ML approaches, as shown in [3,4]. Moreover, using more recently collected data (in the year of 2025) and the H2O AE model, we obtained a high quality predictive discrimination for unseen data (e.g., above 95%). Furthermore, we have shown the global and local XAI screw tightening AE graphs to the company experts (including manufacturing engineers and operators), which provided a positive feedback. In effect, it was mentioned that by analyzing the concentration of anomalous points, the XAI visualizations can help in adjusting and calibrating the screwing systems, improving both their precision and reliability. The proposed XAI graphs were considered particularly valuable in manufacturing environments where screwing processes are extensively used, involving distinct types of products and suppliers. By incorporating the proposed XAI graphs into a continuous screw tightening process optimization, there is potential to reduce the occurrence of anomalies, leading to increased productivity, higher revenue and a lower environmental impact.

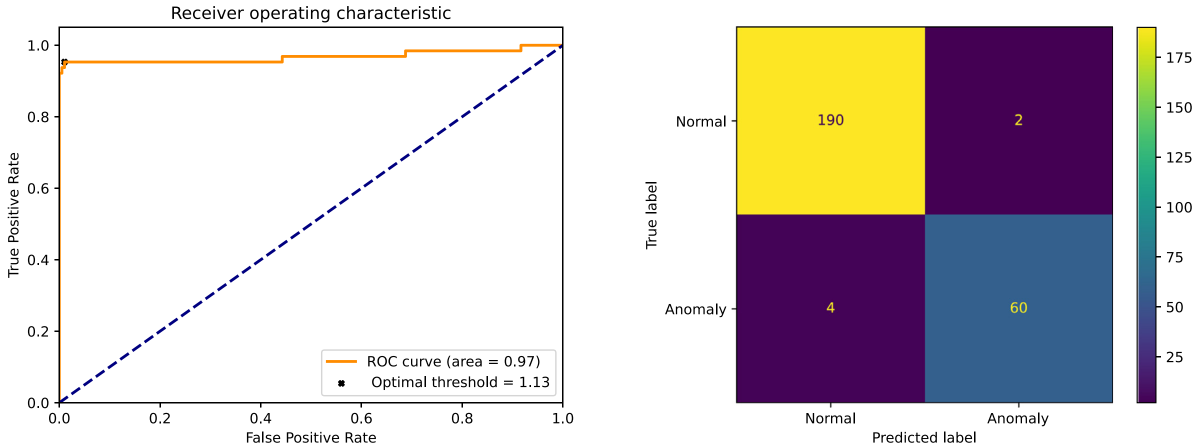


Fig. 1. AE ROC curve (left graph) and confusion matrix using the selected threshold (right plot).

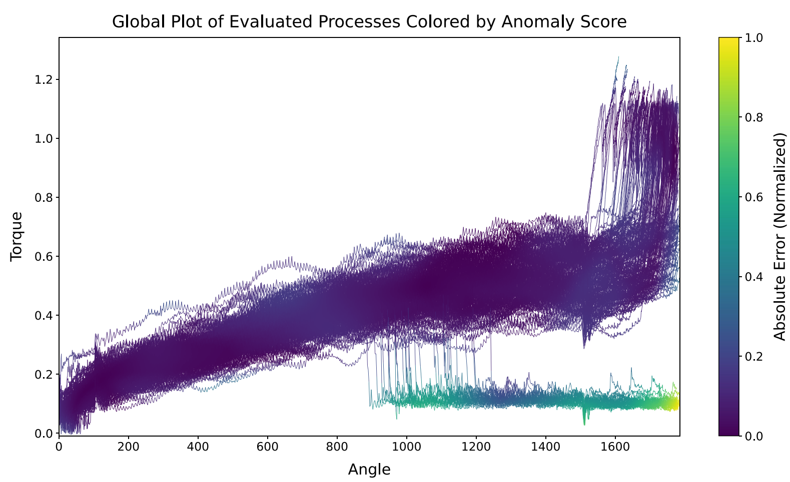


Fig. 2. Global XAI plot with all test screw processes colored by their normalized anomaly score values.

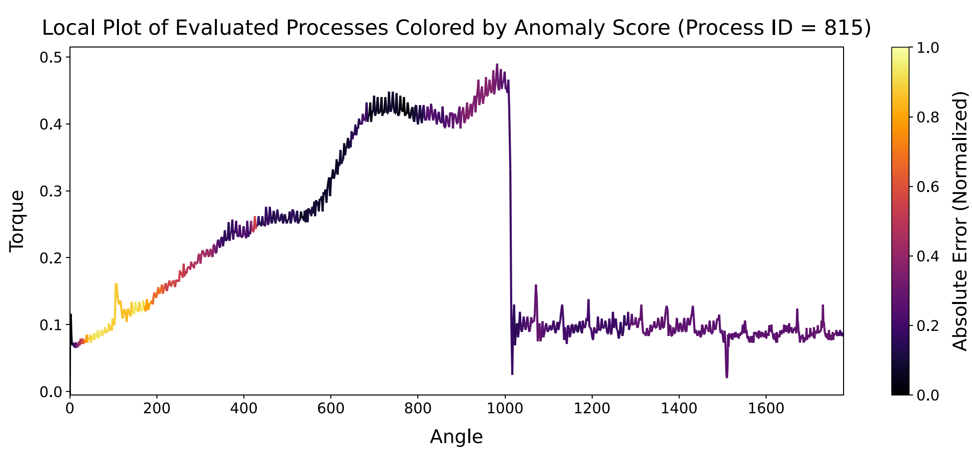
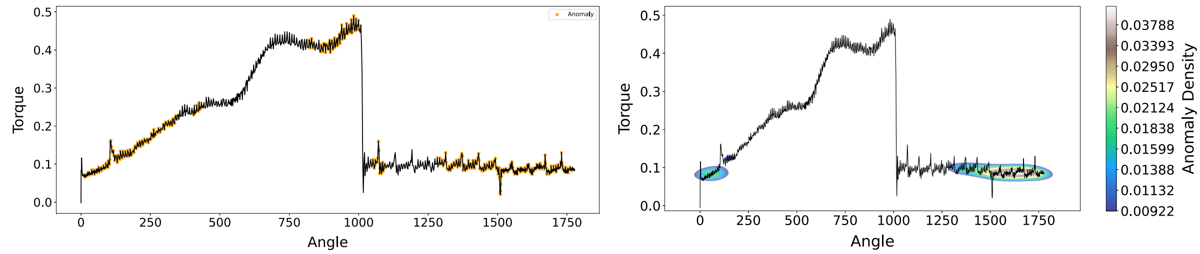


Fig. 3. Local XAI for a single faulty screw process colored by normalized anomaly score values.

3. Limitations and future work

Screw Process Anomaly Visualization (SPAV) is a Python software module that was recently developed to address the specific challenge of performing a data-driven support to the calibration of screwing machines in manufacturing environments, which is a frequent issue in real-world industrial manufacturing settings. The primary goal of SPAV is to provide a suite of XAI visualization techniques (e.g., density plots) that enable non-ML expert users to easily identify critical anomalous points from predictive ML models and that arise during the screwing process.

Initially, the SPAV module was tested using the H2O tool and AE model and then demonstrated also with the Keras framework. Nevertheless, as described in Section 1, the SPAV module can be executed



(a) (b)

Fig. 4. Local XAI plots for a faulty screw process (#815), Keras AE and fixed threshold (value of 5.5): left – with orange colored values above the threshold; and right – with abnormal density colored regions. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

with other ML algorithms (e.g., IF, Random Forests) and frameworks (e.g., scikit-learn [11]). In terms of limitations, it is worth mentioning that a pure unsupervised learning XAI approach was adopted, since we did not have access to labels regarding specific anomaly regions for all faulty screw tightenings. Moreover, we only approached anomaly detection ML and not an ahead-of-time prediction of anomalies. In addition, the SPAV software was validated offline, where the results for a batch (previously collected) dataset were shown to the company operators.

In future work, we aim to handle the mentioned limitations by integrating the proposed SPAV module into the real screw tightening assembly environment. This will allow us to gather valuable feedback, such as a supervised (human) identification of the true anomaly angle-torque regions. Furthermore, we aim to apply predictive ML models (e.g., by employing a time series approach), allowing to gather other interesting performance measures (e.g., detection delay time).

CRediT authorship contribution statement

Marta Moreno: Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Data curation. **Hugo Rocha:** Visualization, Validation, Software, Methodology, Investigation, Data curation. **André Pilastri:** Visualization, Validation, Supervision, Project administration, Methodology, Investigation, Formal analysis, Conceptualization. **Guilherme Moreira:** Validation, Resources, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Luís Miguel Matos:** Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Software, Resources, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Paulo Cortez:** Writing – review & editing, Visualization, Validation, Supervision, Resources, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Data curation, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work is supported by national funds, through the Operational Competitiveness and Internationalization Programme (COMPETE 2020) [Project n° 179826; Funding Reference: SIFN-01-9999-FN-179826]. The authors also wish to thank the anonymous reviewers for their helpful comments.

References

[1] Xun Xu, Yuqian Lu, Birgit Vogel-Heuser, Lihui Wang, Industry 4.0 and industry 5.0—Inception, conception and perception, J. Manuf. Syst. 61 (2021) 530–535.
[2] Mahboob Elahi, Samuel Olaiya Afolaranmi, Jose Luis Martinez Lastra, Jose Antonio Perez Garcia, A comprehensive literature review of the applications of AI techniques through the lifecycle of industrial equipment, Discov. Artif. Intell. 3 (1) (2023) 43.
[3] Diogo Ribeiro, Luís Miguel Matos, Paulo Cortez, Guilherme Moreira, André Luiz Pilastri, A comparison of anomaly detection methods for industrial screw tightening, in: Osvaldo Gervasi, Beniamino Murgante, Sanjay Misra, Chiara Garau, Ivan Blecic, David Taniar, Bernady O. Apduhan, Ana Maria A.C. Rocha, Eufemia Tarantino, Carmelo Maria Torre (Eds.), Computational Science and Its Applications - ICCSA 2021 - 21st International Conference, Cagliari, Italy, September 13-16, 2021, Proceedings, Part II, in: Lecture Notes in Computer Science, vol. 12950, Springer, 2021, pp. 485–500.
[4] Diogo Ribeiro, Luís Miguel Matos, Guilherme Moreira, André Pilastri, Paulo Cortez, Isolation forests and deep autoencoders for industrial screw tightening anomaly detection, Computers 11 (4) (2022) 54.
[5] Ahmed M.A. Salih, Zahra Raisi-Estabragh, Ilaria Boscolo Galazzo, Petia Radeva, Steffen E. Petersen, Karim Lekadir, Gloria Menegaz, A perspective on explainable artificial intelligence methods: SHAP and LIME, Adv. Intell. Syst. 7 (1) (2025).
[6] Oliver Mey, Deniz Neufeld, Explainable AI algorithms for vibration data-based fault detection: Use case-adadpted methods and critical evaluation, Sensors 22 (23) (2022).
[7] François Chollet, et al., Keras, 2015, https://keras.io.
[8] Erin LeDell, Sebastien Poirier, H2O autoML: Scalable automatic machine learning, in: 7th ICML Workshop on Automated Machine Learning (AutoML), 2020.
[9] Nikolai West, Jochen Deuse, Pyscrew: A comprehensive dataset collection from industrial screw driving experiments, 2025, arXiv preprint arXiv:2505.11925.
[10] Tom Fawcett, An introduction to ROC analysis, Pattern Recognit. Lett. 27 (8) (2006) 861–874.
[11] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, E. Duchesnay, Scikit-learn: Machine learning in python, J. Mach. Learn. Res. 12 (2011) 2825–2830.