ÉCOLE NORMALE SUPÉRIEURE PARIS-SACLAY

MASTER 2 - MATHÉMATIQUES, VISION, APPRENTISSAGE

MACHINE LEARNING FOR TIME SERIES

FINAL PROJECT REPORT

Anomaly Detection in Time Series: Deep Dive into Distance Methods

Oliver JACK - olijacklu@gmail.com
Paulo SILVA - paulohenriquecrs@hotmail.com
2024/2025

January 8, 2025

1 Introduction and contributions

The detection of anomalies in time series data is a crucial task in fields as diverse as healthcare, finance, and manufacturing. Time series, defined as ordered sequences of data points collected over time, often contain anomalies that indicate critical events such as health irregularities, fraudulent financial transactions, or equipment malfunctions. Anomaly detection algorithms aim to identify these situations to enable intervention and posterior analysis.

In their comprehensive study, [Schmidl et al., 2022] provided a significant contribution to the anomaly detection field by systematically evaluating 71 state-of-the-art anomaly detection algorithms across 976 time series datasets, both univariate and multivariate. Their work aimed to address the fragmented nature of the field, where algorithms have been developed independently by different communities and tested on limited, cherry-picked datasets, therefore providing flawed benchmarks. By re-implementing algorithms from six core algorithmic families (forecasting, reconstruction, distance, encoding, distribution, and tree methods), the authors delivered a broad overview of the known approaches at that time.

The study's main contributions included an extensive survey of anomaly detection techniques, systematic benchmarking of their effectiveness, and practical insights to ease the task of algorithm selection for new users. However, the wide range of the study came at the expense of the depth of the analysis. Due to the large number of algorithms and datasets considered, the study offered only high-level comparisons, with each family excelling in specific scenarios but none emerging as a clear winner across all cases. This limitation underscores the need for deeper, more focused evaluations within specific families of algorithms.

The workload division for this study was balanced, with both authors contributing to reviewing the original paper, coding the new implementation, and preparing the report and presentation. For instance, in the coding phase, one author focused on grid search configuration while the other developed the synthetic dataset generator. Our implementation used the anomaly detection algorithms provided by the original authors via Docker containers and incorporated elements from the GutenTAG tutorial, but these were integrated into entirely new code to explore scenarios beyond the original study, including hyperparameter tuning with grid search and the development of ensemble anomaly detectors. To evaluate performance, we used 80 datasets from the original paper and generated 20 new synthetic datasets to address scenarios where distance-based algorithms previously struggled. Our work seeks to complement the efforts of [Schmidl et al., 2022] by narrowing the scope to a single family of anomaly detection methods: distance-based algorithms. By focusing exclusively on this family, we aim to conduct a more detailed analysis of their performance. Specifically, we apply hyperparameters fine-tuning techniques, such as grid search, and explore ensemble methods to evaluate whether the combination of different algorithms, either from the same family or different families, can improve the performance and robustness of the anomaly detectors. Our work is available at GitHub[1] for transparency and reproducibility.

2 Method

Anomalies in time series data are points or sequences that deviate from the expected patterns of the series based on a defined measure, model, or embedding. These deviations can occur in individual channels (univariate data) or the correlation between channels (multivariate data).

[1] https://github.com/olijacklu/MVA/tree/main/Machine%20Learning%20for%20Time%20Series/Project

Distance-based methods for anomaly detection identify these deviations by measuring the distances between time series subsequences. Anomalous subsequences are expected to have significantly larger distances from others compared to normal subsequences. These methods, typically unsupervised, use approaches like nearest neighbor comparisons, cluster densities, or mappings into multidimensional spaces to evaluate the separation of subsequences and highlight irregularities. In our study, we evaluate the performance of 8 distance-based methods (CBLOF, COF, *k*-Means, KNN, LOF, PS-SVM, Sub-LOF, VALMOD), 4 of which are briefly presented in Appendix A.

In this study, we adopt the scoring approach used in the original paper, where the algorithms assign a continuous anomaly score to each data point in the time series, which we decided to normalize to the interval [0,1] for consistency and comparability reasons. A higher score indicates a higher likelihood of an anomaly being present, adhering to the principle that $s_i > s_j$ implies T_i is more anomalous than T_j. While applying a threshold would allow for a binary classification of anomalies, this is not the goal of the original paper, nor of our study. For the evaluation, we also rely on two threshold-agnostic metrics from the original work: the *Area Under the Receiver Operating Characteristic curve* (AUC-ROC) and the *Area Under the Precision-Recall curve* (AUC-PR). The AUC-ROC measures the ability of the model to distinguish between positive and negative classes, while the AUC-PR measures the model's ability to balance precision and recall. These metrics allow us to come up with a fair assessment of the algorithms' performance, while maintaining reproducibility with the original results.

To replicate the experiments and extend them to new datasets, we used a Docker-based environment to run the algorithms implemented by the authors. This framework ensures consistent execution of the algorithms, independent of language or library versions. By following these steps, we were able to guarantee the reproducibility of our results.

3 Data

In order to evaluate the performance of the distance-based algorithms, we required reliable, anomaly-annotated data. While numerous related datasets were available, we encountered significant challenges, since these datasets often presented one or more of the following issues: a) The dataset's source was not trustworthy. b) The anomalies were not labeled, making it impossible to evaluate the algorithms' performance. c) The dataset had already been used by the authors in their evaluation (a substantial overlap, as the original study considered 976 time series).

To address these challenges, we adopted the following approach: instead of evaluating our algorithms on all datasets, we selected the 80 univariate datasets with the lowest average AUC-ROC scores across the eight distance-based algorithms considered. This decision was motivated by the fact that not all algorithms are capable of handling multivariate data, and we aimed to focus on scenarios where distance-based methods were most challenged.

Additionally, we used the authors' GutenTAG synthetic data generator [Wenig et al., 2022] to create time series datasets specifically designed to challenge distance-based methods. Based on insights from the original paper, all six families of anomaly detection methods (including distance-based methods) performed poorly on time series with a *cylinder-bell-funnel* (CBF) base oscillation. This oscillation involves a pattern where the time series transitions between three distinct phases: a cylindrical baseline, a bell-shaped peak, and a funnel-shaped trough. Using this knowledge, we generated 20 synthetic datasets with a CBF base oscillation, random numbers of anomalies (ranging from 1-5), and random types of anomalies (related to the amplitude, pattern, platform, and

trend). This brought the total number of datasets to 100, all theoretically representing scenarios where distance-based methods underperform.

Our objective was to extend the analysis of the original paper, by exploring whether the performance of these methods could be improved through three approaches: a) Adjusting the algorithm hyperparameters via a grid search across the 100 datasets. b) Following the idea of ensemble learning by linearly combining algorithms within the distance-based family. c) Extending the ensemble technique by linearly combining methods from different algorithm families. The details of these approaches, along with the results and their implications, will be discussed in the next section.

4 Experiments & Results

In the original paper, the authors demonstrated that distance-based methods excel in detecting specific types of anomalies, such as frequency, variance, and pattern-shift anomalies, while also effectively accounting for trends. Among these methods, algorithms like Sub-LOF emerged as top performers, achieving the highest average AUC-ROC scores across various characteristics. However, the authors also acknowledged that there are many datasets with anomaly types where distance-based methods generally struggle.

When it comes to parameter selection, the authors employed a systematic hyperparameter tuning approach, based on certain assumptions such as parameter independence and the ability to jointly optimize "shared" parameters (parameters with identical functions across multiple algorithms). While this is a valid approach for a parameter space of size 192 over 976 datasets and serves as a great starting point for our analysis, we decided to run a full grid search for most algorithms to optimize their AUC scores over the 100 chosen datasets. For numerical values, we generally opted for a range approach, while testing all possible settings for categorical values. It is worth noting that for those algorithms having more than 20 possible parameter combinations, we opted for a randomized search instead, as the prolonged run times made a complete grid search impractical.

Our results showed that optimizing the algorithm parameters specifically for the 100 datasets led to significant deviations from the default values provided by the authors (as can be seen in Table 2). Figure 1 highlights the effectiveness of parameter fine-tuning in improving the performance of most algorithms, particularly Sub-LOF and k-Means. Sub-LOF exhibits the highest post-tuning mean AUC-ROC of 0.77 with a relatively low variance, indicating consistent performance across datasets. Meanwhile, VALMOD, COF, and PS-SVM only exhibit minimal gains in terms of both AUC-ROC and AUC-PR. This analysis underscores the value of fine-tuning while highlighting the need to consider both mean performance and variability when evaluating algorithm effectiveness.

While parameter tuning resulted in a subtle improvement in terms of detection performance, we wanted to take things a step further by considering ensemble learning. More specifically, we investigated whether a linear combination of distance-based methods could outperform individual algorithms across the 100 datasets. For this purpose, we selected the three top-performing distance algorithms based on their average AUC-ROC and AUC-PR scores: Sub-LOF, *k*-means, and PS-SVM. Although all three algorithms adopt a distance-based approach to anomaly detection, they are still very different in nature (see Appendix A) and could potentially improve the overall performance by complementing one another. In our scoring system, a linear combination of algorithms corresponds to linearly combining their output score arrays, with values constrained to the interval [0,1]. In that sense, it is fairly computationally efficient to test for many different possible linear combinations, whose weights are positive and add up to 1, which ensures that the

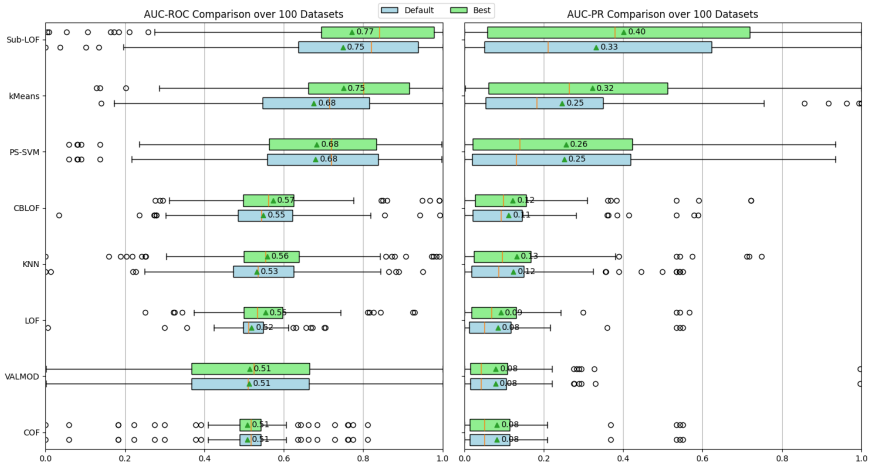

Figure 1: AUC-ROC and AUC-PR scores for both pre- and post-parameter fine-tuning.

final score is kept in the range [0,1].

The ensemble approach within the distance family yielded the best results when combining Sub-LOF ($w_1 = 0.5$), k-means ($w_2 = 0.3$), and PS-SVM ($w_3 = 0.2$). This configuration achieved a mean AUC-ROC of 0.798 (see Figure 2) across the 100 datasets, representing a notable improvement over the individual performance of these algorithms. As previously mentioned, Sub-LOF consistently outperformed others post-tuning, and its inclusion in the ensemble capitalized on its robustness. The clustering-based approach of k-means and the margin-based scoring of PS-SVM helped counteracted Sub-LOF's tendency to struggle with boundary and overlapping anomalies, resulting in a more balanced and robust detection capability.

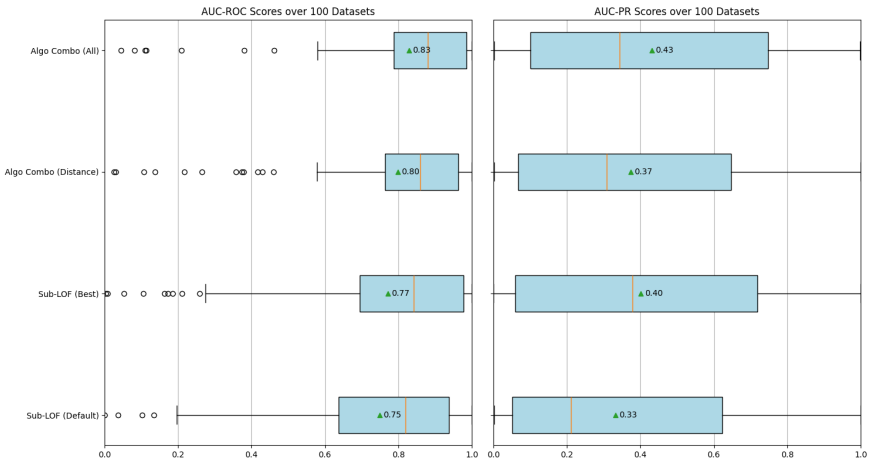

Figure 2: Comparison of AUC-ROC & AUC-PR scores across 100 datasets, highlighting the performance gains of ensemble methods (distance-based and cross-family) over individual Sub-LOF configurations (default and tuned).

While the ensemble approach within the same family of methods led to improved results, we aimed to push the boundaries even further by exploring an ensemble approach that combined methods from different families, while still including the distance-based family. To achieve this, we selected Sub-LOF, the best-performing distance-based method post-parameter tuning in terms

of average AUC-ROC and AUC-PR. Additionally, we included GrammarViz3 from the encoding family and DWT-MLEAD from the distribution family, both identified in the original paper as strong performers, ranking second and third in terms of average AUC-ROC and demonstrating robust capabilities on most time series.

Once again, by evaluating a linear space of possible weight combinations, we achieved an impressive average AUC-ROC of 0.829 (see Figure 2) for the combination Sub-LOF ($w_1 = 0.5$), GrammarViz3 ($w_2 = 0.1$), and DWT-MLEAD ($w_3 = 0.4$). The inclusion of GrammarViz3 and DWT-MLEAD allowed the ensemble to take advantage of diverse algorithmic approaches, with GrammarViz3 excelling at encoding temporal patterns and DWT-MLEAD being adept at handling distributional anomalies. In this sense, the diversity of algorithms complemented each other, as illustrated in Figure 3, where GrammarViz3 and DWT-MLEAD counteract Sub-LOF's tendency to overreact to periodic patterns and enhance its detection of true anomalies

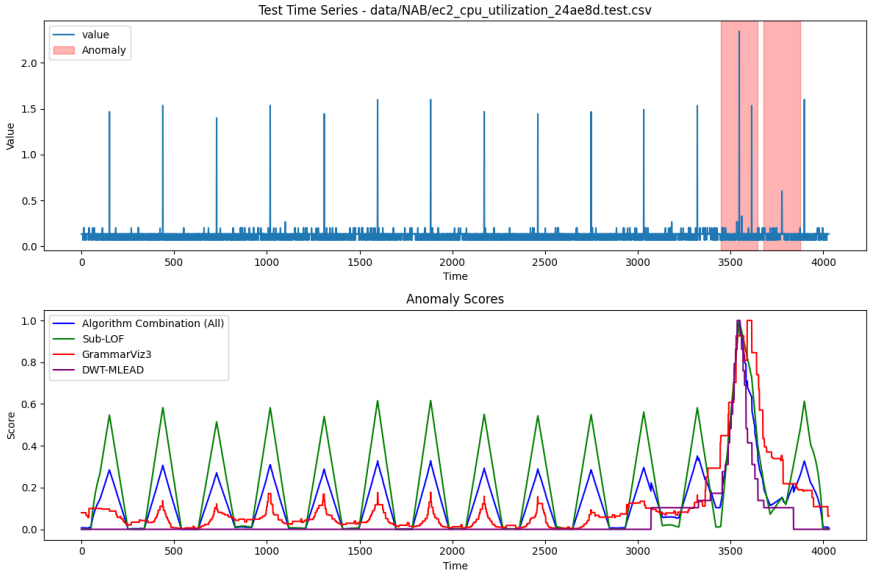

Figure 3: Example of anomalous time series & the associated anomaly scores.

5 Conclusion

Based on the conducted experiments, one can conclude that while parameter tuning is critical for optimizing individual algorithm performance, ensemble learning offers a broader, more robust improvement by combining complementary strengths. The best ensemble configurations, both within the distance family and across families, demonstrate that leveraging algorithm diversity can help reduce the weaknesses of any single method, resulting in superior detection performance.

Interestingly, the ensemble incorporating methods from different families showed a clear advantage over the distance-family ensemble, underscoring the value of algorithmic diversity in anomaly detection. This finding highlights the potential for further research into hybrid ensembles that combine methods from multiple families to tackle increasingly complex anomaly detection challenges. Another possible aspect to consider in future work is runtime efficiency. While the current study focused on the overall detection performance, evaluating and optimizing the computational cost of ensembles would be crucial for real-world applications.

References

[Breunig et al., 2000] Breunig, M., Kröger, P., Ng, R., and Sander, J. (2000). Lof: Identifying density-based local outliers. volume 29, pages 93–104.

[Ma and Perkins, 2003] Ma, J. and Perkins, S. (2003). Time-series novelty detection using one-class support vector machines. volume 3, pages 1741 – 1745 vol.3.

[Ramaswamy et al., 2000] Ramaswamy, S., Rastogi, R., and Shim, K. (2000). Efficient algorithms for mining outliers from large data sets. volume 29, pages 427–438.

[Schmidl et al., 2022] Schmidl, S., Wenig, P., and Papenbrock, T. (2022). Anomaly detection in time series: a comprehensive evaluation. *Proc. VLDB Endow.*, 15(9):1779–1797.

[Wenig et al., 2022] Wenig, P., Schmidl, S., and Papenbrock, T. (2022). Timeeval: a benchmarking toolkit for time series anomaly detection algorithms. *Proc. VLDB Endow.*, 15(12):3678–3681.

[Yairi et al., 2001] Yairi, T., Kato, Y., and Hori, K. (2001). Fault detection by mining association rules from housekeeping data.

A Mathematical Overview of Selected Distance-Based Methods

This appendix presents a brief description and the mathematical aspects of four particular, yet distinct distance-based methods we explored in our study: k-Means [Yairi et al., 2001], KNN [Ramaswamy et al., 2000], PS-SVM [Ma and Perkins, 2003], and Sub-LOF [Breunig et al., 2000].

A.1 k-Means

k-Means is a clustering algorithm that groups similar subsequences of a time series into clusters, where the cluster centroids represent typical patterns. For anomaly detection, the algorithm assigns each subsequence to the closest cluster and calculates the distance to the assigned centroid. Subsequences with larger distances are considered to be potential anomalies, as they deviate significantly from the cluster's representative pattern.

Mathematically, given a set of n subsequences $\{x_1, x_2, \ldots, x_n\}$ and k clusters, k-means minimizes the within-cluster sum of squared distances (WCSSD):

$$\text{WCSSD} = \sum_{j=1}^{k} \sum_{x_i \in C_j} \|x_i - \mu_j\|^2 \tag{1}$$

where C_j is the set of subsequences in the j-th cluster, μ_j is the centroid of cluster C_j, and $\|x_i - \mu_j\|$ is the Euclidean distance.

A.2 KNN

K-Nearest Neighbors (KNN) identifies anomalies in time series by measuring the distances between each subsequence and its k-nearest neighbors. Anomalies are expected to have larger average distances to their nearest neighbors compared to normal subsequences, as they deviate from typical patterns.

Mathematically, for a given subsequence x_i, the anomaly score is computed as the average distance to its k-nearest neighbors:

$$\text{Score}(x_i) = \frac{1}{k} \sum_{j=1}^{k} \|x_i - x_{(j)}\| \tag{2}$$

where $x_{(j)}$ represents the j-th nearest neighbor of x_i, and $\|x_i - x_{(j)}\|$ is the Euclidean distance. A higher score indicates a higher likelihood of anomaly.

A.3 PS-SVM

PhaseSpace-SVM (PS-SVM) adapts one-class Support Vector Machines (SVMs) for time series data by projecting it into a phase space. Time series, being sequential data, cannot be directly used in traditional SVMs, which operate on independent vector sets. To address this, the time series $x(t)$ is transformed into a set of vectors $T_E(N)$ in a phase space Q using a time-delay embedding process. This embedding introduces the embedding dimension E, creating vectors $\mathbf{x}_E(t)$ from overlapping subsequences of the time series.

However, low-frequency components in the time series can cause bias in this transformation, where vectors align along the diagonal vector $\mathbf{1} = [1\ 1\ ...\ 1]^T$. To mitigate this issue, the projected phase space Q' is introduced, where vectors are orthogonally projected away from $\mathbf{1}$. This step reduces bias and enhances the method's ability to detect anomalies.

The transformation is mathematically defined as:

$$\mathbf{x}_E(t) = [x(t-E+1), x(t-E+2), \ldots, x(t)], \tag{3}$$

where $t \in E, \ldots, N$, and $\mathbf{x}_E(t) \in Q$.

To address diagonal bias, the projection is computed using:

$$\mathbf{x}'_E(t) = \left(I - \frac{1}{E}\mathbf{1}\mathbf{1}^T\right)\mathbf{x}_E(t), \tag{4}$$

where I is the identity matrix, $\mathbf{1}$ is the diagonal vector, and $\mathbf{x}'_E(t)$ represents the adjusted vectors in the projected phase space Q'.

A.4 Sub-LOF

Subsequence LOF (Sub-LOF) is an adaptation of the Local Outlier Factor (LOF) algorithm designed for time series data. Instead of analyzing fixed points in a spatial dataset, Sub-LOF evaluates subsequences within a time series to identify anomalies. By converting the time series into overlapping subsequences, the algorithm assigns an outlier score to each subsequence, quantifying its deviation relative to its nearest neighbors.

The core of Sub-LOF is the computation of the Local Outlier Factor for a subsequence p, based on its $MinPts$-nearest neighbor subsequences $N_{MinPts}(p)$. The LOF value of p is defined as:

$$LOF_{MinPts}(p) = \frac{\sum_{o \in N_{MinPts}(p)} \frac{lrd_{MinPts}(o)}{lrd_{MinPts}(p)}}{|N_{MinPts}(p)|},$$

where the local reachability density (lrd) of a subsequence p is the inverse of the average reachability distance between p and its $MinPts$-nearest neighbors. A high LOF value indicates that p is an outlier with respect to its neighbors, while a low LOF value suggests homogeneity.

B Tables

B.1 Summary of chosen Datasets

Collection	Size	Avg. AUC-ROC	Avg. AUC-PR	Avg. Length	Avg. # Anomalies
NAB	29	0.56	0.23	3,333.76	2.00
GutenTAG	24	0.62	0.17	10,000.00	5.38
Self-Generated	20	0.63	0.10	10,000.00	2.70
KDD-TSAD	8	0.58	0.11	9,651.75	1.00
NASA-MSL	8	0.58	0.23	3,898.75	1.38
NASA-SMAP	7	0.55	0.27	8,177.57	1.43
NormA	4	0.61	0.12	5,375.00	1.50

Table 1: Breakdown of 100 chosen datasets by collection.

B.2 Optimal Hyperparameters after Grid Search

Parameter	CBLOF	COF	kMeans	KNN	LOF	PS-SVM	Sub-LOF	VALMOD
n_clusters	25 (50)	/	50 (50)	/	/	/	/	/
alpha	0.8 (0.9)	/	/	/	/	/	/	/
beta	7 (5)	/	/	/	/	/	/	/
use_weights	`False` (`False`)	/	/	/	/	/	/	/
n_neighbors	/	25 (50)	/	100 (50)	100 (50)	/	100 (50)	/
anomaly_window_size	/	/	50 (20)	/	/	/	/	/
leaf_size	/	/	/	10 (20)	10 (20)	/	10 (20)	/
method	/	/	/	largest (largest)	/	/	/	/
distance_metric_order	/	/	/	3 (2)	3 (1)	/	/	/
kernel	/	/	/	/	/	rbf	/	/
heap_size	/	/	/	/	/	/	/	25 (50)

Table 2: Optimal Hyperparameters after Grid Search over 100 chosen Datasets. Values in parentheses represent default values provided by the authors.