RoseCDL: Robust and Scalable Convolutional Dictionary Learning for Rare-event and Anomaly Detection

Jad Yehya[1,*] **Mansour Benbakoura**[1,*] **Cédric Allain**[1] **Benoît Malézieux**[1]

Matthieu Kowalski[2] **Thomas Moreau**[1]

[1]Université Paris-Saclay, Inria, CEA, Palaiseau, 91120, France.
[2]Université Paris-Saclay, Inria, CNRS, Laboratoire Interdisciplinaire des Sciences du Numériques, Gif-sur-Yvette, France.
*Equal Contributions.

Abstract

Detecting rare events and anomalies in large-scale signals is essential in fields such as astronomy, physical simulations, and biomedical science. In many cases, this problem naturally decomposes into identifying common local patterns and detecting deviations that correspond to anomalies. Convolutional Dictionary Learning (CDL) is a powerful tool for modeling local structures, but its adoption for this task has been limited by computational demands and sensitivity to outliers. We introduce RoseCDL, a novel CDL algorithm designed for robust and scalable modeling of signal pattern distribution. RoseCDL leverages stochastic windowing for efficient training and incorporates inline outlier detection to enhance robustness. This enables unsupervised identification of anomalous and rare patterns in long signals based on the local reconstruction loss. Experiments on real-world datasets show that RoseCDL delivers improved detection accuracy and computational efficiency, making CDL practical for challenging detection tasks in large-scale signal analysis.

1 INTRODUCTION

Identifying recurring patterns and anomalies in a signal is a crucial task in many scientific fields, from finding QRS complex – *a.k.a.* heartbeats – in ECG (Luz et al., 2016) to detecting blood-cells in biological images (Yellin et al., 2017) or particular celestial objects in astronomical images (Giavalisco and the GOODS Teams, 2004). In large-scale settings, this process must be automated. Supervised methods have been developed to address these tasks, often relying on large annotated datasets and deep learning models (Murat et al., 2020; Choudhary et al., 2024; Cornu et al., 2024). However, obtaining labeled data can be costly, time-consuming, or even infeasible, especially when the patterns of interest are rare or hard to characterize. This motivates the development of unsupervised methods capable of automatically uncovering patterns distribution in large datasets.

Convolutional Dictionary Learning (CDL; Grosse et al., 2007) is a powerful method for modeling local structures in signals, with applications in audio, neuroscience, and image processing (Dupré la Tour et al., 2018; Papyan et al., 2017). Its core principle is to represent an observed signal as the convolution of a learned dictionary of patterns with a sparse activation vector. By aggregating over many signals, CDL learns a dictionary that captures recurring local structures, making it particularly suitable for unsupervised pattern discovery. Despite its promise, the use of CDL in practice remains limited. Two challenges, in particular, hinder its applicability to large, noisy datasets.

Scalability. Learning with CDL is computationally expensive, primarily due to the sparse coding phase. Considerable effort has been devoted to improving scalability, including deterministic optimizations (Wohlberg, 2015), local-window methods that partition the signal (Grosse et al., 2007; Papyan et al., 2017; Moreau and Gramfort, 2020; Dragoni et al., 2022), and approximate

Proceedings of the 29th International Conference on Artificial Intelligence and Statistics (AISTATS) 2026, Tangier, Morocco. PMLR: Volume 300.

techniques such as learned sparse coding and algorithm unrolling (Gregor and LeCun, 2010; Tang et al., 2021; Tolooshams et al., 2018; Tolooshams and Ba, 2022; Malézieux et al., 2022). Online algorithms (Mairal et al., 2010; Mensch et al., 2016; Liu et al., 2017; Zeng et al., 2019) further reduce computational cost by updating the dictionary on subsets of the data. However, existing approaches either struggle to generalize across large datasets or achieve only limited computational gains.

Robustness to anomalies and rare events. CDL is highly sensitive to anomalies: because outliers are sparse but often high-magnitude, the algorithm may mistakenly interpret them as meaningful patterns. Theoretical work in classical dictionary learning has formalized this risk (Gribonval et al., 2015), and a few practical remedies have been proposed, such as Elastic Net regularization (Mairal et al., 2010) or heavy-tailed fitting terms (Jas et al., 2017). Yet these methods either increase computational cost or only partially address robustness. Related research in outlier detection (OD) has emphasized reconstruction-based frameworks, where models are trained to reconstruct normal data while failing on anomalies (Ruff et al., 2021; Schmidl et al., 2022). While the majority of reconstruction-based OD methods are semi-supervised, fully unsupervised methods perform competitively (Darban et al., 2024), and similar principles are used for rare-event detection (Shyalika et al., 2024). This idea also resonates with classical robust regression, where corrupted observations are trimmed based on residual magnitude. The Least Trimmed Squares (LTS) estimator (Rousseeuw, 1984; Rousseeuw and Leroy, 2005) and its extensions, including Sparse LTS (Alfons et al., 2013) demonstrate the effectiveness of trimming in high-dimensional and structured settings.

Our contributions. This paper introduces a novel CDL algorithm called RObust and ScalablE CDL (RoseCDL) that addresses these limitations. To ensure scalability, we propose a **stochastic windowing approach**, where the sparse coding problem is solved approximately on a small, randomly sampled data windows, and the dictionary is updated using the results of local computations. To ensure robustness, we integrate an **inline outlier detection** mechanism based on local reconstruction error, which discards patches poorly explained by the learned dictionary. The same reconstruction-error criterion can be applied at test time to identify anomalies and rare events, providing a practical unsupervised detection tool alongside representation learning. The resulting algorithm learns dictionaries that capture the local structure of data, remains robust to artifacts, scales to large datasets, and naturally doubles as an unsupervised anomaly detection method. More broadly, it **reframes CDL from a reconstruction problem to the estimation of the patch distribution underlying a signal**, enabling both robust representation and principled rare-event and anomaly detection. We demonstrate its effectiveness on both synthetic and real-world data.

2 FINDING COMMON AND RARE PATTERNS IN SIGNALS: THE ROSECDL ALGORITHM

Let $\mathbf{x} \in \mathbb{R}^T$ be a univariate signal with length T. Convolutional Dictionary Learning (CDL) consists of finding a dictionary: $D = (\mathbf{d}_k)_{k\in[\![1,K]\!]} \in \mathbb{R}^{K\times L}$ of K patterns of length $L \ll T$, and corresponding activation vectors $Z = (\mathbf{z}_k)_{k\in[\![1,K]\!]} \in \mathbb{R}^{K\times(T-L+1)}$, that minimize the distance between $\mathbf{x}$ and $\hat{\mathbf{x}} = D * Z = \sum_k \mathbf{d}_k * \mathbf{z}_k$, where $*$ denotes the convolution.

This is achieved by solving the following optimization problem:

$$\begin{aligned} \min_{D,Z} \quad & F(D,Z;\mathbf{x}) = \underbrace{\tfrac{1}{2}\|\mathbf{x} - D*Z\|_2^2}_{f_{\mathbf{x}}(Z)} + \lambda\|Z\|_1, \\ \text{s.t.} \quad & \|\mathbf{d}_k\|_2^2 \le 1, \quad \forall k \in [\![1,K]\!]. \end{aligned} \tag{1}$$

where $\|Z\|_1 = \sum_k \|\mathbf{z}_k\|_1$. Fig. 1 illustrates the CDL model for an univariate signal. This problem can easily be extended to multivariate and multidimensional signals such as images by adapting the convolution operator to work with tensors.

When working with large signal databases, the primary interest is not finding the best reconstruction of the signal used to learn the dictionary but instead finding the patterns that best model the population. Therefore, this paper focuses on **characterizing the distribution** of the signals $\mathbf{x}$, with the following optimization problem:

$$\begin{aligned} \min_{D} \quad & \mathbb{E}_{\mathbf{x}}\left[\min_{Z} F(D,Z;\mathbf{x})\right], \\ \text{s.t.} \quad & \|\mathbf{d}_k\|_2^2 \le 1, \ \forall k \in [\![1,K]\!]. \end{aligned} \tag{2}$$

While this formulation departs from classical optimization-based literature for CDL (Wohlberg, 2015), it is often used with deep CDL approaches to learn denoisers that generalize to unseen images using super learning losses (Scetbon et al., 2021; Zheng et al., 2021; Deng et al., 2023). Here, we consider the unsupervised loss F to learn the dictionary of event patterns directly from the signals.

For both formulation (1) and (2), classical solvers rely on an iterative optimization algorithm to solve the

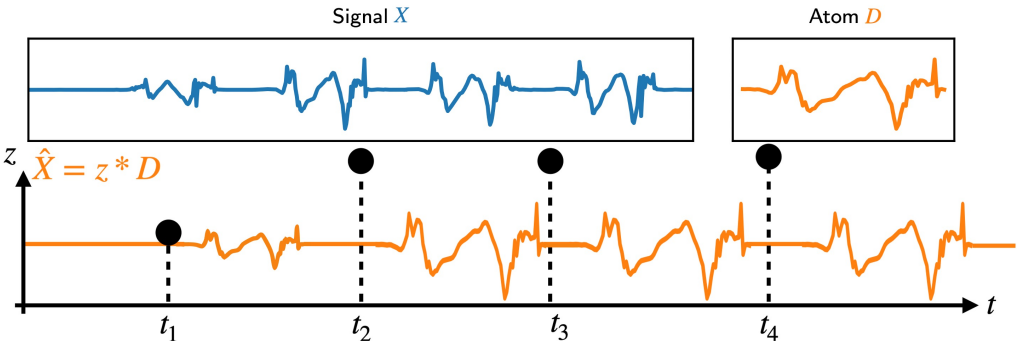

Figure 1: Schematic operation of the CDL 1D univariate signal, adapated from the `dicodile` package example[1]. The output of the CDL is composed of a set of atoms (here 1) alongside their respective activations.

sparse coding problem in Z for fixed D, and alternate it with a dictionary update step. The sparse coding step is typically solved using iterated soft-thresholding iterations (e.g., FISTA, Beck and Teboulle, 2009; Chambolle and Dossal, 2015), which reads:

$$\begin{aligned} t_0 &= 1 \\ Z_0 = Y_0 &\in \mathbb{R}^{K \times T - L + 1};\ t_0 = 1 \\ Z_{k+1} &= \mathrm{St}(Y_k - \nu \nabla_Z f_{\mathbf{x}}(Z), \nu\lambda) \\ t_{k+1} &= \frac{1}{2}(1 + \sqrt{1 + 4t_k^2}) \\ Y_{k+1} &= Z_k + \frac{t_k - 1}{t_{k+1}}(Z_{k+1} - Z_k) \end{aligned} \tag{3}$$

with St the soft-thresholding operator $\mathrm{St}(Z) = \mathrm{sign}(Z)(|Z| - \lambda)_+$ and ν the step size. The dictionary update may rely on projected gradient descent constrained to the unit ℓ_2-ball. However, these methods become computationally expensive on long signals or large datasets, as both the sparse codes and gradients must be computed over all the signals. Online methods mitigate the need to consider all signals, but at the expense of increased memory usage and still require solving the sparse coding problem on full signals. This motivates using stochastic or localized approximations, especially in settings where robustness to artifacts or outliers is required.

2.1 Stochastic windowing

Due to the convolutional structure of the CDL model, points in the far-apart signal are only weakly dependent Moreau and Gramfort (2020). Indeed, the value of the sparse code at time t is seldom impacted by the values at time $t + s$, with s larger than the size of the dictionary L. We will therefore perform windowing spanning across all channels synchronously.

[1] https://tommoral.github.io/dicodile/auto_examples/plot_gait.html

By considering long enough windows of the signal, the problem (2) can be approximated by

$$\begin{aligned} \min_D \quad & \mathbb{E}_\tau \left[\min_Z F(D, Z; \mathbf{x}_\tau) \right], \\ \text{s.t.} \quad & \|\mathbf{d}_k\|_2^2 \leq 1,\ \forall k \in [\![1, K]\!], \end{aligned} \tag{4}$$

where $\mathbf{x}_\tau$ are windows of the signal, starting at $\tau \in [\![1, T - W_{\text{win}} + 1]\!]$ and of size W_{win} such that $L \leq W_{\text{win}} \ll T$. This formulation is only approximate as it produces inexact sparse code for the windows, due to border effect. But with the weak spatial dependence of the model, if the window is large and the activation are sparse, these effects are negligible. Moreover, by considering all windows with overlap, we limit the impact of specific border effects in the algorithm. To minimize (4), we propose RoseCDL, a stochastic gradient descent algorithm aimed to characterize the distribution of patches in the signal.

We process as follows: At each step of the CDL outer problem, we begin by sampling N_W windows $(\mathbf{x}_w)_{1 \leq w \leq N_W}$ from a uniform distribution. The windows are selected with overlap to limit the bias due to border effects. We then compute an approximate sparse code $Z_w^\star$ for each subproblem by minimizing the sparse coding over the window. Following Malézieux et al. (2022); Tolooshams and Ba (2022), whose results suggested that optimizing over D did not require precise sparse coding at each step, we use an approximation $Z^{N_{\text{FISTA}}}$ of $Z^\star(D; \mathbf{x})$, where $Z^{N_{\text{FISTA}}}$ is given by N_{FISTA} iterations of the FISTA algorithm (3).

Using these per-window approximated activation vectors, we can then update the dictionary D. Due to the stochastic nature of the algorithm, we depart from traditional alternate minimization strategies and perform only one gradient step. This single-step update strategy is further justified by the inherent noise in the gradient estimate, which arises from both the stochastic sampling of windows and the use of approximate sparse codes. In such settings, performing multiple

Algorithm 1 CDL with stochastic windowing.

input X, N_{iter}, N_W, N_{FISTA}
  Initialize $D^{(0)}$
  for $0 \leq i \leq N_{\text{iter}} - 1$ **do**
    Sample N_W windows in the dataset: $(X_w)_{w \in [\![1, N_w]\!]}$
    for $1 \leq w \leq N_W$ **do**
      Compute the approximate sparse code $Z_w^{N_{\text{FISTA}}} \approx Z_w^\star(D^{(i)}; X_w)$
      Compute an outlier mask (cf. Sec. 2.2)
      Compute the loss F and its gradient $\nabla_D F$ outside the outlier mask
    end for
    Compute best step size α_i with SLS
    $D^{(i+1)} \leftarrow D^{(i)} - \alpha_i \nabla_D \sum_w F_w(D^{(i)}, Z_w^{N_{\text{FISTA}}}; X_w)$
  end for
output $D^{(N_{\text{iter}})}$

gradient steps per batch does not significantly reduce the update variance and may even amplify the impact of the approximation error in the sparse coding stage. This strategy is in-line with the deep CDL algorithm, but we do not backpropagate the gradient through the sparse code approximation $Z^{N_{\text{FISTA}}}$, as it leads to unstable Jacobian estimation for the original problem (4), as described in Malézieux et al. (2022). To stabilize the learning and reduce the number of parameters, we compute the optimal stepsize of the dictionary update with the Stochastic Line Search algorithm (SLS; Vaswani et al., 2019). This algorithm can be efficiently implemented using deep learning frameworks and can leverage GPU acceleration. The complete procedure is summarized in Alg. 1. We note that when trimming is ignored and the sparse codes are solved exactly, RoseCDL fits within the SGD setting, with stochasticity arising from the sampled windows. In this setting, convergence properties are well established (Bottou et al., 2018)

2.2 Inline outlier detection

The second component of RoseCDL is its inline outlier detection framework. Let $\mathbf{x}$ be a signal of the form:

$$\mathbf{x} = \mathbf{d}_a * \mathbf{z}_a + \mathbf{d}_b * \mathbf{z}_b + \mathbf{n}, \tag{5}$$

where $\mathbf{d}_a$ is a common pattern in the signal, $\mathbf{d}_b$ is a rare pattern, and $\mathbf{n}$ is an abnormal pollution (e.g., artifacts, sudden spikes in the signal). Traditional CDL algorithms are likely to struggle to recover $\mathbf{d}_a$ for two main reasons. First, artifacts in $\mathbf{n}$ often have a significant variance, which can distract the CDL from the significant signal. Second, even when the anomalies in $\mathbf{n}$ are well discarded, the rare pattern $\mathbf{d}_b$ acts as a pollution that prevents the algorithm from learning the atom $\mathbf{d}_a$. While effects of $\mathbf{n}$ and $\mathbf{d}_b$ are often discarded through preprocessing Dupré la Tour et al. (2018), it is often very difficult to use this reliably on large population datasets.

The intuition behind our approach is that if $\mathbf{d}_a$ is sufficiently represented in $\mathbf{x}$, then most of the patches of $\mathbf{x}$ should contain information relative to this pattern. Consequently, these patches should be the best reconstructed ones. In comparison, the patches containing non-zero values of $\mathbf{n}$ are expected to have a high reconstruction error due to the unpredictable nature of anomalies and artifacts. Finally, given a dictionary $\mathbf{d}$ that is more correlated with $\mathbf{d}_a$ than $\mathbf{d}_b$, the patches containing chunks of $\mathbf{d}_b$ are expected to have a higher reconstruction error than those with $\mathbf{d}_a$.

To summarize, the distribution of patch reconstruction errors $F(D, Z; \mathbf{x}_\tau)_\tau$ is expected to have three modes:

1. The main, low-value mode corresponding to chunks of $\mathbf{x}_a = \mathbf{d}_a * \mathbf{z}_a$,
2. A secondary mode with a slightly higher value corresponding to chunks of $\mathbf{x}_b = \mathbf{d}_b * \mathbf{z}_b$,
3. A high-value mode corresponding to chunks of $\mathbf{n}$.

Consequently, the reconstruction error of a patch can be used as an indicator to determine whether it contains information about the relevant signal. We leverage this intuition to inflect the learning trajectory of $\mathbf{d}$ towards $\mathbf{d}_a$. We define the set $\mathcal{P}_\beta = \{\tau | F(D, Z; \mathbf{x}_\tau) < \beta\}$, where $\beta \in [0, 1]$ is a threshold selecting the proportion of outliers. With a well chosen β, $\mathcal{P}_\beta$ indicates the patches that coincide with realizations of $\mathbf{x}_a$. Therefore, to make the CDL converge towards $\mathbf{d}_a$, we change the objective value for the dictionary updates for its trimmed version:

$$\widetilde{F}(Z, D; \mathbf{x}) = \frac{1}{W_{\text{patch}}} \sum_{\tau \in \mathcal{P}_\beta} F(Z, D; \mathbf{x}_\tau). \tag{6}$$

Using this trimmed loss, we can show that in simple settings, the RoseCDL algorithm is more robust to the presence of outliers.

Proposition 2.1 (Stability of the common pattern)**.** *Consider a population of signals X composed of two patterns $\mathbf{d}_a$ and $\mathbf{d}_b$, with activations such that the patterns do not overlap in the signal, and corrupted by an additive Gaussian noise $\epsilon \sim \mathcal{N}(0, \sigma Id)$. Introduce $c = \mathbf{d}_a^\top \mathbf{d}_b$ and ρ the proportion of rare-event pattern $\mathbf{d}_b$ activations in the population. Then, in the noiseless setting,*

i. $\mathbf{d}_a$ is a fixed point of the classical CDL algorithm for $K = 1$ if $c \leq \lambda$

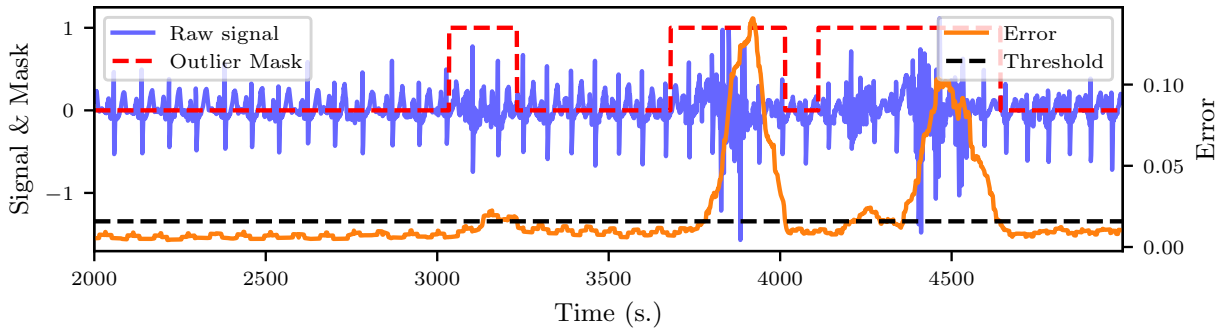

Figure 2: Raw signal X, reconstruction error, threshold and learned outlier mask on subject a02 (minute 56) of Physionet Apnea-ECG data set. Detection method is based on modified z-score (MAD), with $\alpha = 3.5$. The method correctly identifies outliers blocks.

ii. $\mathbf{d}_b$ *is a fixed point of* RoseCDL *algorithm for* $K = 1$ *if* $c \leq \lambda$ *or the* RoseCDL *algorithm is used with an outlier threshold trimming a proportion of windows greater than* ρ.

The proof is deferred in App. A. This theoretical analysis is intentionally illustrative: its purpose is to provide intuition on why trimming improves robustness, rather than to serve as a full convergence guarantee for the algorithm. Nevertheless, this demonstrates that even in ideal conditions (no overlap and starting with the right pattern), classical CDL fails to recover the common pattern due to interference from rare events, while RoseCDL corrects this via trimming. However, a critical design choice is the selection of the statistic β used as the threshold, discussed below.

Threshold selection. From the distribution of patch reconstruction errors $(\boldsymbol{\varepsilon}_\tau)_\tau = (F(D, Z; \mathbf{x}_\tau))_\tau$, one needs to compute a threshold β which separates the outlier patches in $\mathbf{x}$ from the normal ones. The goal is to detect extreme points in the distribution, which are too large compared to the population of patch errors. The outlier detection literature provides three main methods relevant in this case:

1. The **method of quantiles**, where $\beta = Q_{\boldsymbol{\varepsilon},(1-\alpha)}$ is the quantile of order $(1-\alpha)$ of the set $\boldsymbol{\varepsilon}$,
2. The z**-score method** (Iglewicz and Hoaglin, 1993), where each error $\boldsymbol{\varepsilon}_\tau$ has an associated score $z_\tau = (\boldsymbol{\varepsilon}_\tau - \mu_{\boldsymbol{\varepsilon}})/\sigma_{\boldsymbol{\varepsilon}}$, with $\mu_{\boldsymbol{\varepsilon}}$ and $\sigma_{\boldsymbol{\varepsilon}}$ denoting respectively the mean and standard deviation of the distribution of errors, and where outliers are defined as observations such that $|z_\tau| > \alpha$, generally $\alpha = 2$ or 3, thus having $\beta = \mu_{\boldsymbol{\varepsilon}} + \alpha\ \sigma_{\boldsymbol{\varepsilon}}$,
3. The **modified** z**-score** (MAD; Iglewicz and Hoaglin 1993), where, similarly as the z-score, each error point is associated with a score based on the median, and the resulting threshold is $\beta = \text{Med}_{\boldsymbol{\varepsilon}} + (\alpha\ \text{Mad}_{\boldsymbol{\varepsilon}})/0.6745$, with $\text{Mad}_{\boldsymbol{\varepsilon}} = \text{Med}\,(|\boldsymbol{\varepsilon} - \text{Med}_{\boldsymbol{\varepsilon}}|)$, where Med denotes the median operator, and generally $\alpha = 3.5$. Here, $\alpha = 3.5$ is not a tunable hyperparameter but a standard choice in robust statistics (Iglewicz and Hoaglin, 1993). The threshold is computed adaptively from the empirical distribution of reconstruction errors, making it data-driven, statistically grounded, and independent of any contamination-rate assumptions. It is not inflated by extreme reconstruction errors caused by outliers, which prevents it from being excessively high, which would reduce detection sensitivity.

These methods are initially bilateral, but only the upper bound is considered in this work because we aim to detect outliers with large reconstruction errors. Fig. 2 illustrates the outlier detection method on real ECG data, with the outliers mask computed with the reconstruction error and the threshold.

Role of the outlier mask for rare-event detection. The inline outlier detection module is a central component of our algorithm. It enables the computation of an outlier mask during training, serving two key purposes: (1) it excludes identified outliers from the loss function, thereby improving the robustness of dictionary learning; (2) it enables the unsupervised detection of rare events in the signal by interpreting the outlier mask as a detection map. Indeed, provided a signal $\mathbf{x}$ in the form of Eq. (5), we have shown that RoseCDL is able to extract the contribution of the common pattern $\mathbf{d}_a$. Consequently, the corresponding outlier mask can be used to select the residual signal $\mathbf{x}' = \mathbf{d}_b * \mathbf{z}_b + \mathbf{n}$. Running another instance of RoseCDL on $\mathbf{x}'$ then allows one to recover the pattern $\mathbf{d}_b$ and localize its occurrences $\mathbf{z}_b$. In multivariate time series, RoseCDL computes reconstruction errors on multi-

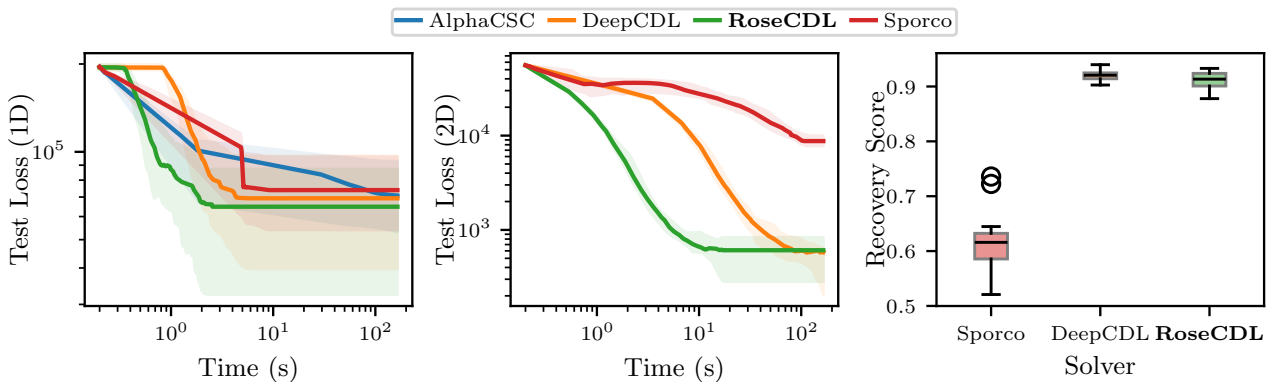

Figure 3: Comparison of optimization runtime for **RoseCDL**, `AlphaCSC`, Sporco, and DeepCDL in 1D and 2D settings, highlighting the superior scalability and convergence speed of **RoseCDL**. The runtime plots show the evolution of test loss over time. The third subplot reports the dictionary recovery score at convergence for 2D data. As `AlphaCSC` cannot be applied to 2D data, only results for Sporco, DeepCDL, and **RoseCDL** are shown.

channel patches, so even small deviations on individual channels can accumulate into a global deviation that exceeds the computed threshold. In practice, multivariate anomalies are detected whenever their combined effect sufficiently alters the local motif.

3 NUMERICAL EXPERIMENTS

In this section, we present numerical experiments that demonstrate the scalability and robustness of RoseCDL, as well as its ability to detect anomalies in temporal (1D) and spatial (2D) signals, with an arbitrary number of channels, on simulated and real-world data. We implemented the algorithm using the `Pytorch` framework (Paszke et al., 2017). [2] The experiments fall into three categories, each targeting a different aspect: scalability, robustness to outliers, and anomaly detection.

3.1 Scalability

We performed a comprehensive comparison between RoseCDL and three state-of-the-art CDL methods: `AlphaCSC` (Dupré la Tour et al., 2018), Sporco (Wohlberg, 2017), and DeepCDL, a variant of Tolooshams and Ba (2022). DeepCDL represents the unrolled variant of RoseCDL where windows are not stochastically sampled and gradients account for the Jacobian of the sparse code computed through backpropagation. In contrast, RoseCDL employs an alternating minimization scheme as explained in Sec. 2.

We evaluate each method's computational cost and runtime on two large-scale datasets. The one-dimensional (1D) experiment is performed on 20 synthetic multivariate signals containing 50,000 time samples with two channels (see App. C for details). The two-dimensional (2D) experiment is conducted on a semi-synthetic dataset of images of 2000×2000 pixels. The cost is evaluated as the value of the objective function $F(D, Z^\star(D); \mathbf{x})$ at each iteration on a separate test set, with $Z^\star$ computed to convergence, assessing the capacity of the solver to minimize Eq. (2). To evaluate dictionary recovery, we use the metric proposed by Moreau and Gramfort (2020), which aligns true and estimated atoms by convolutional cosine similarity (see App. D).

The empirical results shown in the left and middle panels of Fig. 3 demonstrate that RoseCDL exhibits superior scalability attributable to two architectural advantages shared with our implementation of DeepCDL: **(i)** GPU-optimized training yields substantial speedups when properly leveraged **(ii)** Fast Fourier Transform-based convolutions significantly reduce the computation time for large kernels compared to standard spatial convolutions. The distinction from DeepCDL arises primarily from optimization strategy differences: DeepCDL's unrolled architecture requires gradient computation over substantially larger parameter sets, increasing computational overhead compared to RoseCDL's alternating minimization approach.

The right panel of Fig. 3 demonstrates that RoseCDL matches the dictionary recovery performance of DeepCDL and substantially better than Sporco, validating both computational efficiency and solution quality.

It is important to note that `AlphaCSC` does not support two-dimensional data and was excluded from image-based experiments. Despite this limitation, RoseCDL achieves comparable or superior test costs across all evaluated configurations.

[2] The code is available at `https://github.com/tomMoral/RoseCDL`.

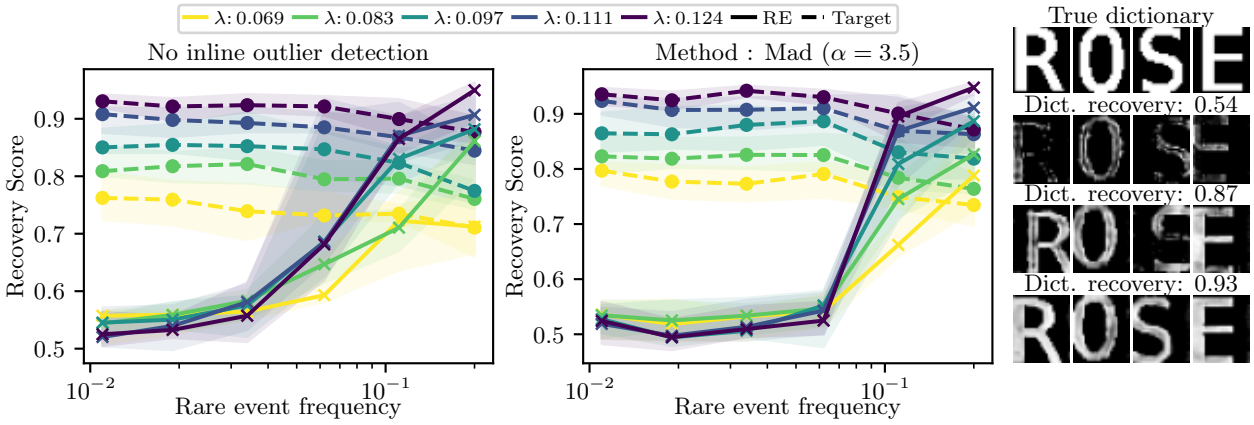

Figure 4: Comparison between RoseCDL without inline outlier detection (left panel) and with it (middle panel). Evolution of median recovery scores for RoseCDL on the `ROSE+Z` data set across 20 independent trials. Shaded regions represent the range between 25th and 75th percentiles. Right panel: illustration of recovered atoms for different recovery scores.

To ensure that the results above are not artifacts of parameter choices, we evaluated RoseCDL under different setups and compared it to `AlphaCSC`, assessing both computation acceleration and potential trade-offs in solution quality. Varying the window size from 10 to 100 times the atom length produced only minor differences, with performance more strongly influenced by the choice of optimizer. Using Adam resulted in a 7–8× speedup while keeping the error within +4% of `AlphaCSC`'s, whereas SLS achieved smaller speedups (4.5–5.5×) but reduced the error compared to `AlphaCSC`. When varying the signal length, RoseCDL exhibited sublinear scaling and remained usable on signal lengths where `AlphaCSC` became prohibitive. These experiments are detailed in Appendix B.1.

3.2 Robustness to outliers

In this subsection, we study the role of the inline outlier detection module in enabling RoseCDL to reconstruct common patterns in the presence of anomalies and outlier patterns. First, an example of RoseCDL mechanism on the Physionet Apnea-ECG (Penzel et al., 2000) is shown in Fig. 2 highlighting parts of the data that are discarded during training when using the inline outlier module as well as the outlier mask produced. This shows that the reconstruction error highlights part of the signal that significantly differ from the rest, allowing to identify anomalies in the data. This also stabilizes the learning, as shown in App. B.3 which presents a side-by-side visualization of atoms learned with and without inline outlier detection, demonstrating the clear benefits of this module.

To better quantify this effect, we constructed the `ROSE+Z` dataset, a semi-synthetic corpus of images generated from 5,000 characters drawn from four letters (`R`, `O`, `S`, and `E`) together with spaces, mimicking text-like documents. A small proportion of the letter `Z` was added to introduce rare events. Using this dataset, we assess how RoseCDL recovers the underlying patterns under varying levels of contamination. For this experiment, we implemented the inline outlier detection module with the MAD method described in Sec. 2.2, as preliminary comparisons against the quantile and z-score methods consistently favored MAD (see App. B.2). We then evaluated RoseCDL with and without inline outlier detection across 20 independent trials on the `ROSE+Z` dataset. Fig. 4 shows the recovery score for both the common letters `ROSE` and the outlier letter `Z` depending on the outlier frequency.

The inline outlier detection stabilizes training, improves the quality of the learned dictionary, and reduces variability across trials by preventing rare events from contaminating the learning process. While without inline outlier detection, the recovery slowly decreases, as the outlier gets mixed in with the common patterns, the inline mechanism allows for a sharper transition once the outlier frequency becomes too large and cannot be discarded anymore. A key parameter of RoseCDL is the regularization coefficient λ, introduced in Sec. 2, which controls the sparsity of the learned activation vector. We express λ as a fraction of the maximum regularization value, λ_{max}, corresponding to the smallest regularization value for which the activation vector is entirely zero. Our insights indicate that the best dictionary recovery is achieved when setting $\lambda = 0.1\lambda_{max}$, effectively balancing sparsity and reconstruction accu-

	Dim.	Dataset	RoseCDL	AB	DAGMM	MP	AT	TimesFM	TimesNet
Pattern	U	ECG	**0.534**	0.099	0.077	0.163	0.095	0.150	0.151
	U	SVDB	**0.431**	0.217	–	0.224	0.106	0.119	0.121
	M	Simulated	**0.986**	0.704	0.644	–	0.624	0.652	0.590
No Pattern	U	Daphnet	0.111	0.186	–	0.101	0.060	0.048	**0.289**
	U	MITDB	0.155	0.186	–	**0.250**	0.135	0.136	0.152
	U	Dodgers	**0.238**	0.078	0.102	0.199	0.140	0.087	0.126
	M	MSL	0.136	**0.171**	0.088	–	0.053	0.136	0.111
	M	SMAP	0.111	0.136	–	–	**0.564**	0.107	0.140
Avg. Runtime (s)			**23.47**	2594.71	23221.20	687.12	11715.00	3986.07	1916.37
Avg. Pattern			**0.65 ± 0.30**	0.34 ± 0.32	0.36 ± 0.40	0.19 ± 0.04	0.28 ± 0.30	0.31 ± 0.30	0.29 ± 0.26
Avg. No Pattern			0.15 ± 0.05	0.15 ± 0.05	0.09 ± 0.01	0.18 ± 0.08	**0.19 ± 0.21**	0.10 ± 0.04	0.16 ± 0.07

Table 1: Comparison of AUC-PR of RoseCDL with state-of-the-art anomaly detection methods on both pattern (P) and non-pattern (NP) datasets, spanning univariate (U) and multivariate (M) settings. RoseCDL achieves the strongest performance on pattern anomaly datasets, its target regime, while remaining competitive on non-pattern data. Notably, RoseCDL delivers these results with a runtime more than two orders of magnitude faster than all deep baselines, and unlike Matrix Profile (MP), it scales seamlessly to multivariate inputs. Bold values indicate the best score in each row, underlined the second best. (AB : AnomalyBERT, AT : AnomalyTransformer).

racy. Qualitative and quantitative results are given in App. B.4. The inline outlier detection also improves the algorithm's robustness with respect to this choice of the regularization parameter λ.

3.3 Anomaly detection

We evaluated the anomaly detection capability of RoseCDL on diverse datasets from the TSB-UAD and TSB-AD benchmarks (Paparrizos et al., 2022; Liu and Paparrizos, 2024), covering both univariate and multivariate cases. Since RoseCDL targets *pattern anomalies*, we included datasets both well- and poorly-aligned with this inductive bias to assess its strengths and limitations.

For baselines, we compared against representative unsupervised methods across two major families identified by Darban et al. (2024): forecasting-based and reconstruction-based. Specifically, we selected competitive examples such as the Anomaly Transformer (AT; Xu et al., 2022) and TimesNet (Wu et al., 2023), which respectively exemplify reconstruction-based and forecasting-based approaches. To further broaden the comparison, we included the classical Matrix Profile method (MP; Yeh et al., 2016), representing a strong shallow baseline, as well as the recently proposed foundation model TimesFM (Das et al., 2024), which extends beyond the scope of recent surveys such as Ruff et al. (2021) and Darban et al. (2024).

As summarized in Tab. 1, RoseCDL consistently achieves competitive or superior performance in terms of AUC-PR on datasets with pattern anomalies. This validates its central design principle of leveraging local structures. On datasets lacking strong patterns (e.g., Daphnet, Dodgers, MSL, SMAP) or where anomalies arise outside the pattern dimension (e.g., MITDB), performance drops but remains comparable to baselines, highlighting both the scope and the boundaries of the proposed method. More precisely, anomalies spanning several atom lengths can still be detected, but anomalies affecting long-range temporal dynamics or global drifts without disrupting local shape are less aligned with CDL assumptions, which explains the performance gap on datasets like MITDB where anomalies primarily affect heartbeat timing rather than morphology.

Examples of both missed and detected anomalies on various datasets are given in App. B.5. Beyond accuracy, RoseCDL is over two orders of magnitude faster than competing methods, enabling scalability to large collections of time series.

These results demonstrate that RoseCDL can extract meaningful patterns from corrupted real-world data without explicit preprocessing or prior knowledge of the outlier proportion. This is made possible by the inline outlier detection module, which enhances both training stability and robustness to hyperparameter choices.

4 CONCLUSION

In this study, we introduce RoseCDL, a robust and scalable approach to Convolutional Dictionary Learning (CDL). While CDL has traditionally focused on reconstructing signals, we reframe it as estimating the distribution of local patches. This approach supports the two main features of RoseCDL: stochastic win-

dowing for scalability and inline outlier detection for robustness to anomalies. Together, these components make RoseCDL suitable not only for large-scale pattern discovery but also for unsupervised anomaly and rare-event detection. Concretely, the method first models the common patterns in a signal, removes them, and then learns a dictionary on the residuals. This yields a robust representation and an outlier mask, while also uncovering the distinctive patterns that occur specifically within anomalous regions.

A central concept of this paper is reframing CDL as a tool to characterize the distribution of patterns in the signal. Beyond improving scalability and robustness, this perspective also provides a principled foundation for anomaly and rare-event detection: local reconstruction errors naturally reflect deviations from the learned patch distribution.

Limitations. Like all CDL-based methods, our approach relies on a non-convex objective. Nevertheless, the optimization scheme builds on well-established principles, for which convergence to meaningful local minima has been observed and studied in prior work (Gribonval et al., 2015; Malézieux et al., 2022; Tolooshams and Ba, 2022). Our trimming strategy intentionally excludes poorly reconstructed patches during training, which introduces bias but is consistent with our design goal of separating atypical regions rather than fitting them. While this mechanism confers robustness to outliers, it is conservative by nature: many segments may be flagged as atypical. Consequently, RoseCDL is better suited for robust representation learning and rare event detection than for high-precision anomaly scoring.

Acknowledgement

This project was supported by the French National Research Agency (ANR) through the BenchArk project (ANR-24-IAS2-0003) and EBUL project (ANR- 23-CE23-0001). Mansour Benbakoura was supported from a national grant attributed to the ExaDoST project of the NumPEx PEPR program, under the reference ANR-22-EXNU-0004. This work was performed using HPC resources from GENCI–IDRIS (Grant 2025-AD011015308R1).

References

A. Alfons, C. Croux, and S. Gelper. Sparse least trimmed squares regression for analyzing high-dimensional large data sets. *The Annals of Applied Statistics*, pages 226–248, 2013. 2

A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202, 2009. 3

L. Bottou, F. E. Curtis, and J. Nocedal. Optimization methods for large-scale machine learning. *SIAM Review*, 60(2):223–311, 2018. 4

A. Chambolle and C. Dossal. On the convergence of the iterates of the "fast iterative shrinkage/thresholding algorithm". *Journal of Optimization Theory and Applications*, 166:968–982, 2015. 3

S. Choudhary, S. Kumar, P. S. Siddhaarth, and G. Charitasri. Transforming blood cell detection and classification with advanced deep learning models: A comparative study. *arXiv:2410.15670*, 2024. 1

D. Cornu, P. Salomé, B. Semelin, A. Marchal, J. Freundlich, S. Aicardi, X. Lu, G. Sainton, F. Mertens, F. Combes, and C. Tasse. YOLO-CIANNA: Galaxy detection with deep learning in radio data - I. A new YOLO-inspired source detection method applied to the SKAO SDC1. *Astronomy & Astrophysics*, 690: A211, 2024. 1

D. F. Crouse. On implementing 2D rectangular assignment algorithms. *IEEE Transactions on Aerospace and Electronic Systems*, 52(4):1679–1696, 2016. 20

Z. Z. Darban, G. I. Webb, S. Pan, C. Aggarwal, and M. Salehi. Deep learning for time series anomaly detection: A survey. *ACM Computing Surveys*, 57(1): 1–42, 2024. 2, 8

A. Das, W. Kong, R. Sen, and Y. Zhou. A decoder-only foundation model for time-series forecasting. In *Proceedings of the 41st International Conference on International Conference on Machine Learning (ICML)*, 2024. 8

X. Deng, J. Xu, F. Gao, X. Sun, and M. Xu. DeepM^{2}CDL: Deep multi-scale multi-modal convolutional dictionary learning network. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(5):2770–2787, 2023. 2

L. Dragoni, R. Flamary, K. Lounici, and P. Reynaud-Bouret. Sliding window strategy for convolutional spike sorting with LASSO: Algorithm, theoretical guarantees and complexity. *Acta Applicandae Mathematicae*, 179(1), 2022. 1

T. Dupré la Tour, T. Moreau, M. Jas, and A. Gramfort. Multivariate convolutional sparse coding for electromagnetic brain signals. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 31, pages 3292–3302, 2018. 1, 4, 6

M. Giavalisco and the GOODS Teams. The great observatories origins deep survey: Initial results from optical and near-infrared imaging. *The Astrophysical Journal Letters*, 600(2):L93, 2004. 1

K. Gregor and Y. LeCun. Learning fast approximations of sparse coding. In *Proceedings of the 27th International Conference on International Conference on Machine Learning (ICML)*, page 399–406. Omnipress, 2010. 2

R. Gribonval, R. Jenatton, and F. Bach. Sparse and spurious: Dictionary learning with noise and outliers. *IEEE Transactions on Information Theory*, 61(11): 6298–6319, 2015. 2, 9

R. Grosse, R. Raina, H. Kwong, and A. Y. Ng. Shift-invariant sparse coding for audio classification. *Cortex*, 8:9, 2007. 1

B. Iglewicz and D. C. Hoaglin. Chapter 3: Outlier labeling. In *How to Detect and Handle Outliers*, volume 16 of *The ASQC Basic References in Quality Control: Statistical Techniques*. ASQC Quality Press, 1993. 5

M. Jas, T. D. La Tour, U. Şimşekli, and A. Gramfort. Learning the morphology of brain signals using alpha-stable convolutional sparse coding. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 1099–1108, 2017. 2

J. Liu, C. Garcia-Cardona, B. Wohlberg, and W. Yin. Online convolutional dictionary learning. In *Proceedings of the International Conference on Image Processing (ICIP)*, pages 1707–1711, 2017. 2

Q. Liu and J. Paparrizos. The elephant in the room: Towards a reliable time-series anomaly detection benchmark. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 37, 2024. 8

E. J. d. S. Luz, W. R. Schwartz, G. Cámara-Chávez, and D. Menotti. ECG-based heartbeat classification for arrhythmia detection: A survey. *Computer Methods and Programs in Biomedicine*, 127:144–164, 2016. 1

J. Mairal, F. Bach, J. Ponce, and G. Sapiro. Online learning for matrix factorization and sparse coding. *Journal of Machine Learning Research*, 11(1), 2010. 2

B. Malézieux, T. Moreau, and M. Kowalski. Understanding approximate and unrolled dictionary learning for pattern recovery. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2022. 2, 3, 4, 9

A. Mensch, J. Mairal, B. Thirion, and G. Varoquaux. Dictionary learning for massive matrix factorization. In *Proceedings of the 33rd International Conference on Machine Learning (ICML)*, volume 48, pages 1737–1746. PMLR, 2016. 2

T. Moreau and A. Gramfort. DiCoDiLe: Distributed convolutional dictionary learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020. 1, 3, 6, 19

F. Murat, O. Yildirim, M. Talo, U. B. Baloglu, Y. Demir, and U. R. Acharya. Application of deep learning techniques for heartbeats detection using ECG signals-analysis and review. *Computers in Biology and Medicine*, 120:103726, 2020. 1

J. Paparrizos, Y. Kang, P. Boniol, R. S. Tsay, T. Palpanas, and M. J. Franklin. Tsb-uad: an end-to-end benchmark suite for univariate time-series anomaly detection. *Proceedings of the VLDB Endowment*, 15 (8):1697–1711, 2022. 8

V. Papyan, Y. Romano, J. Sulam, and M. Elad. Convolutional dictionary learning via local processing. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017. 1

A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer. Automatic differentiation in PyTorch. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017. 6

T. Penzel, G. B. Moody, R. G. Mark, A. L. Goldberger, and J. H. Peter. The Apnea-ECG database. *Computers in Cardiology*, 27:255–258, 2000. 7, 16

P. J. Rousseeuw. Least median of squares regression. *Journal of the American Statistical Association*, 79 (388):871–880, 1984. 2

P. J. Rousseeuw and A. M. Leroy. *Robust Regression and Outlier Detection*. John Wiley & Sons, 2005. 2

L. Ruff, J. R. Kauffmann, R. A. Vandermeulen, G. Montavon, W. Samek, M. Kloft, T. G. Dietterich, and K.-R. Muller. A unifying review of deep and shallow anomaly detection. *Proceedings of the IEEE*, 109(5):756–795, 2021. 2, 8

M. Scetbon, M. Elad, and P. Milanfar. Deep K-SVD denoising. *IEEE Transactions on Image Processing*, 30:5944–5955, 2021. 2

S. Schmidl, P. Wenig, and T. Papenbrock. Anomaly detection in time series: a comprehensive evaluation. *Proceedings of the VLDB Endowment*, 15(9):1779–1797, 2022. 2

C. Shyalika, R. Wickramarachchi, and A. P. Sheth. A comprehensive survey on rare event prediction. *ACM Computing Surveys*, 57(3), 2024. 2

H. Tang, H. Liu, W. Xiao, and N. Sebe. When dictionary learning meets deep learning: Deep dictionary learning and coding network for image recognition with limited data. *IEEE Transactions on Neural Networks and Learning Systems*, 32(5):2129–2141, 2021. 2

B. Tolooshams and D. E. Ba. Stable and interpretable unrolled dictionary learning. *Transactions on Machine Learning Research (TMLR)*, 2022. 2, 3, 6, 9

B. Tolooshams, S. Dey, and D. Ba. Scalable convolutional dictionary learning with constrained recurrent sparse auto-encoders. In *Proceedings of the 28th IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*, pages 1–6, 2018. 2

S. Vaswani, A. Mishkin, I. Laradji, M. Schmidt, G. Gidel, and S. Lacoste-Julien. Painless stochastic gradient: Interpolation, line-search, and convergence rates. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 32, pages 3732–3745, 2019. 4

B. Wohlberg. Efficient algorithms for convolutional sparse representations. *IEEE Transactions on Image Processing*, 25(1):301–315, 2015. 1, 2

B. Wohlberg. SPORCO: A Python package for standard and convolutional sparse representations. In *Proceedings of the 15th Python in Science Conference*, pages 1–8, 2017. 6

H. Wu, T. Hu, Y. Liu, H. Zhou, J. Wang, and M. Long. Timesnet: Temporal 2d-variation modeling for general time series analysis. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2023. 8

J. Xu, H. Wu, J. Wang, and M. Long. Anomaly transformer: Time series anomaly detection with association discrepancy. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2022. 8

C.-C. M. Yeh, Y. Zhu, L. Ulanova, N. Begum, Y. Ding, H. A. Dau, D. F. Silva, A. Mueen, and E. Keogh. Matrix profile i: All pairs similarity joins for time series: A unifying view that includes motifs, discords and shapelets. In *Proceedings of the 16th International Conference on Data Mining (ICDM)*, pages 1317–1322, 2016. 8

F. Yellin, B. D. Haeffele, and R. Vidal. Blood cell detection and counting in holographic lens-free imaging by convolutional sparse dictionary learning and coding. In *Proceedings of the IEEE International Symposium on Biomedical Imaging (ISBI)*, pages 650–653, 2017. 1

Y. Zeng, J. Chen, and G.-B. Huang. Slice-based online convolutional dictionary learning. *IEEE Transactions on Cybernetics*, pages 1–14, 2019. 2

H. Zheng, H. Yong, and L. Zhang. Deep convolutional dictionary learning for image denoising. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 630–641, 2021. 2

Checklist

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. Yes
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. Yes
   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. Yes

2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. Yes
   (b) Complete proofs of all theoretical results. Yes
   (c) Clear explanations of any assumptions. Yes

3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). Yes
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). Yes
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). Yes
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). Yes

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets. Yes

(b) The license information of the assets, if applicable. Not Applicable

(c) New assets either in the supplemental material or as a URL, if applicable. Yes

(d) Information about consent from data providers/curators. Yes

(e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. Not Applicable

5. If you used crowdsourcing or conducted research with human subjects, check if you include:

(a) The full text of instructions given to participants and screenshots. Not Applicable

(b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. Not Applicable

(c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. Not Applicable

A ANALYTICAL STUDY

Proposition 2.1 (Stability of the common pattern)**.** *Consider a population of signals X composed of two patterns $\mathbf{d}_a$ and $\mathbf{d}_b$, with activations such that the patterns do not overlap in the signal, and corrupted by an additive Gaussian noise $\epsilon \sim \mathcal{N}(0, \sigma Id)$. Introduce $c = \mathbf{d}_a^\top \mathbf{d}_b$ and ρ the proportion of rare-event pattern $\mathbf{d}_b$ activations in the population. Then, in the noiseless setting,*

i. *$\mathbf{d}_a$ is a fixed point of the classical CDL algorithm for $K = 1$ if $c \leq \lambda$*

ii. *$\mathbf{d}_b$ is a fixed point of RoseCDL algorithm for $K = 1$ if $c \leq \lambda$ or the RoseCDL algorithm is used with an outlier threshold trimming a proportion of windows greater than ρ.*

Proof. We consider a dictionary $D \in \mathbb{R}^{1\times L}$ with a single atom $\mathbf{d}$. As we consider signals composed of patterns with no overlap, we can separate each segment and we have a population of signals $X = zd_i + \epsilon$, with $z \in \mathbb{R}$, $\epsilon \sim \mathcal{N}(0, \sigma^2 I)$ and $d_i = \mathbf{d}_a$ with probability $1 - \rho$ and $\mathbf{d}_b$ with probability ρ, with $\|\mathbf{d}_a\|_2 = \|\mathbf{d}_b\|_2 = 1$. We consider all atoms $\mathbf{d}, \mathbf{d}_a, \mathbf{d}_b$ to be unit norm. Wlog, we can consider $z = 1$, as this amounts to rescaling the value of λ_{max}, and we consider that $c_a = \mathbf{d}^\top \mathbf{d}_a$ and $c_b = \mathbf{d}^\top \mathbf{d}_b$ are positive, as we can consider $-\mathbf{d}$ otherwise. We also consider that the noise level is small enough such that $\sigma^2 < c_j$.

This model is a simplified model in which we have a population of signals where we want to identify the pattern of an event $\mathbf{d}_a$ from the pattern of a rare event $\mathbf{d}_b$.

In this setting, if we further have that the auto-correlation of $\mathbf{d}$ with $\mathbf{d}_a$ and $\mathbf{d}_b$ is maximal when they are aligned, then the sparse coding of a signal X can be computed with the following formula:

$$z^*(X, \mathbf{d}) = \begin{cases} 0 & \text{if} \quad c + \epsilon^\top \mathbf{d} \leq \lambda \\ c + \epsilon^\top \mathbf{d} - \lambda & \text{otherwise} \end{cases} \tag{7}$$

with $c = \mathbf{d}^\top d_i$, which has value c_a with probability $1 - \rho$ and c_b otherwise.

We can compute the loss value for this $z^*(X, \mathbf{d})$ for X where z^* is non-zero:

$$\begin{aligned} F(\mathbf{d}, z^*; X) &= \frac{1}{2}\|X - z^*\mathbf{d}\|_2^2 + \lambda\|z^*\|_1 && (8)\\ &= \frac{1}{2}\left(\|X\|_2^2 - 2(c + \epsilon^\top \mathbf{d} - \lambda)(c + \epsilon^\top \mathbf{d}) + \|(c + \epsilon^\top \mathbf{d} - \lambda)\mathbf{d}\|_2^2\right) + \lambda|c + \epsilon^\top \mathbf{d} - \lambda| && (9)\\ &= \frac{1}{2}\left(\|X\|_2^2 - 2(c + \epsilon^\top \mathbf{d} - \lambda)(c + \epsilon^\top \mathbf{d}) + (c + \epsilon^\top \mathbf{d} - \lambda)^2 + 2\lambda(c + \epsilon^\top \mathbf{d} - \lambda)\right) && (10)\\ &= \frac{1}{2}\left(\|X\|_2^2 - 2(c + \epsilon^\top \mathbf{d} - \lambda)(c + \epsilon^\top \mathbf{d} - \lambda) + (c + \epsilon^\top \mathbf{d} - \lambda)^2\right) && (11)\\ &= \frac{1}{2}\left(\|X\|_2^2 - (c + \epsilon^\top \mathbf{d} - \lambda)^2\right) && (12)\\ &= \frac{1}{2}\left(\|d_i\|_2^2 - (c - \lambda)^2 + \|\epsilon\|_2^2 - (\epsilon^\top \mathbf{d})^2 - 2(1 - (c - \lambda))\epsilon^\top \mathbf{d}\right) && (13)\\ & && (14) \end{aligned}$$

Taking the expectation over the noise yields:

$$\mathbb{E}_\epsilon\left[F(\mathbf{d}, z^*; X)\right] = \frac{1}{2}\left(1 - (c - \lambda)^2 + (L - 1)\sigma^2\right) \tag{15}$$

For c between λ and 1, this function is decreasing in c, meaning that for two samples constructed with $\mathbf{d}_a$ and $\mathbf{d}_b$, if the correlation $c_0 = \mathbf{d}^\top \mathbf{d}_a$ is larger than the correlation $c_b = \mathbf{d}^\top \mathbf{d}_b$, then the reconstruction loss for sample 0 is smaller in expectation than the reconstruction loss for a sample 1.

We can also compute the gradient of this function with respect to $\mathbf{d}$. Note that with the KKT condition defining z^*, we have that the $\nabla_z F(\mathbf{d}, z*; X) = 0$, and thus we do not need to compute the Jacobian of z^* when computing

the derivative of F with respect to $\mathbf{d}$. The gradient reads:

$$\nabla_{\mathbf{d}} F(\mathbf{d}, z^*; X) = z^*(z^*\mathbf{d} - X) \tag{16}$$
$$= (z^*)^2\mathbf{d} - z^*X \tag{17}$$
$$= (c + \epsilon^\top \mathbf{d} - \lambda)^2\mathbf{d} - (c + \epsilon^\top \mathbf{d} - \lambda)(d_i + \epsilon) \tag{18}$$

Taking the expectation over the noise yields:

$$\mathbb{E}_\epsilon \left[\nabla_{\mathbf{d}} F(\mathbf{d}, z^*; X)\right] = ((c-\lambda)^2 + \sigma^2)\mathbf{d} - (c-\lambda)d_i + \underbrace{\mathbb{E}_\epsilon \left[\epsilon^\top \mathbf{d}\epsilon\right]}_{\sigma^2\mathbf{d}} \tag{19}$$
$$= ((c-\lambda)^2 + 2\sigma^2)\mathbf{d} - (c-\lambda)d_i \tag{20}$$

This yields

$$\mathbb{E}[\nabla_{\mathbf{d}} F(\mathbf{d}, z^*; X)] = \left((1-\rho)(c_a-\lambda)^2 + \rho(c_b-\lambda)^2 + 2\sigma^2\right)\mathbf{d} - (1-\rho)(c_a-\lambda)\mathbf{d}_a - \rho(c_b-\lambda)\mathbf{d}_b$$

In the noiseless case, if $\mathbf{d} = \mathbf{d}_a$, and $\lambda \leq c_b = (\mathbf{d}_b)^\top \mathbf{d}_a < 1$, with the classical algorithm, the expected gradient reads

$$\mathbb{E}_X \left[\nabla_{\mathbf{d}} F(\mathbf{d}_a, z^*; X)\right] = -(1-\rho)\lambda(1-\lambda)\mathbf{d}_a + \rho((c_a-\lambda)^2\mathbf{d}_a - (c_b-\lambda)\mathbf{d}_b) \tag{21}$$
$$= \left(\rho(c_a-\lambda)^2 - (1-\rho)\lambda(1-\lambda)\right)\mathbf{d}_a - \rho(c_b-\lambda)\mathbf{d}_b \tag{22}$$

This gradient is not colinear with $\mathbf{d}_a$, showing that $\mathbf{d}_a$ is not a fixed point of the projected gradient descent algorithm in this context. Even in a noiseless and very simple setting, the $\mathbf{d}_a$ is not a solution of the Classical CDL algorithm.

In contrast, when using the least trimmed square procedure with a trimming threshold rejecting a proportion ρ of the samples, we can show that $\mathbf{d}_a$ is a fixed point in the noiseless setting. As seen in (15), the loss for samples X associated with $\mathbf{d}_b$ is smaller than the loss for samples associated with $\mathbf{d}_a$, and therefore rejecting ρ samples from the gradient computation leads to:

$$\mathbb{E}_X \left[\nabla_{\mathbf{d}} F(\mathbf{d}_a, z^*; X)\right] = -(1-\rho)\lambda(1-\lambda)\mathbf{d}_a \tag{23}$$

as the gradient is colinear with $\mathbf{d}_a$, thus $\mathbf{d}_a$ is a fixed point of the projected gradient descent and of the learning procedure. □

B ADDITIONAL EXPERIMENTS

In this section, we present two numerical experiments intended to characterize the behavior of RoseCDL. We first discuss the influence of the regularization parameter λ on the ability of RoseCDL to learn the atoms. Then, we highlight the importance of the inline outlier detection module in making the CDL robust to outliers.

B.1 Scalability

To thoroughly assess scalability, we conduct additional experiments varying both window sizes and signal lengths.

B.1.1 Window size

For window size analysis on 1D signals with $T = 100{,}000$ and $\lambda = 0.8$, we evaluate both Adam and SLS optimizers across window sizes ranging from $10L$ to $100L$. To ensure full GPU utilization, we maintain a constant product window size $\times$ batch size. As shown in Tab. B.1, RoseCDL consistently achieves validation losses within 4% of `AlphaCSC` performance across all configurations, while maintaining runtimes of 12–22% relative to `AlphaCSC`, corresponding to approximately 5$\times$ speedup.

Optimizer	**Adam**				**SLS**			
Window size	$10L$	$20L$	$50L$	$100L$	$10L$	$20L$	$50L$	$100L$
Validation loss	+3.0%	+3.8%	+2.7%	+2.7%	−0.2%	−0.2%	−0.2%	**-0.4%**
Runtime	15.4%	**12.3%**	12.5%	12.7%	21.7%	21.2%	16.3%	17.6%

Table B.1: Comparison of Adam and SLS with different window sizes. Validation loss expressed relatively to `AlphaCSC`'s. Runtime expressed in proportion to `AlphaCSC`'s.

Notably, the SLS optimizer demonstrates superior convergence properties with validation losses within 0.4% of `AlphaCSC`, though with slightly increased runtime compared to Adam. The results validate that border effects do not hinder convergence with our stochastic windowing approach, even for small window sizes (as low as $10L$).

B.1.2 Signal length

For signal length scalability, we evaluate performance on signals ranging from 10k to 1M time samples. As detailed in Tab. B.2, RoseCDL demonstrates sublinear scaling, with runtimes growing from 15.2 s (10k samples) to 202.8 s (1M samples), while `AlphaCSC` becomes computationally prohibitive beyond 100k samples. This superior scalability enables RoseCDL to process signals substantially larger than existing full-signal sparse coding methods.

T	10k	30k	100k	300k	1M
Runtime RoseCDL (s)	15.2	16.4	32.6	68.0	202.8
Runtime `AlphaCSC` (s)	67.5	102.8	198.5	N/A	N/A

Table B.2: Signal length scaling: runtime comparison between RoseCDL and `AlphaCSC` for varying signal sizes T.

B.2 Choice of the outlier detection method

To evaluate the performance of RoseCDL on 2D data, we constructed a semi-synthetic dataset of images by generating 5000 characters sampled from a set of four letters (`R`, `O`, `S`, and `E`) along with spaces. These images emulate text-like documents composed of words formed from the selected characters. We added the letter `Z` in a small proportion to introduce rare events. Experiments were conducted with a 10% contamination rate. Consequently, for the inline outlier detection methods, we implemented the quantile detection method with α value: 10 %, alongside with MAD and z-score methods as described in Sec. 2.2. To assess the efficiency of the inline outlier module on this dataset, we compared it to RoseCDL without the inline outlier module, where we computed the threshold by computing the reconstruction error after the learning part. We show the detection over the epochs using different outlier detection algorithms by computing the F1 score between the inline model mask

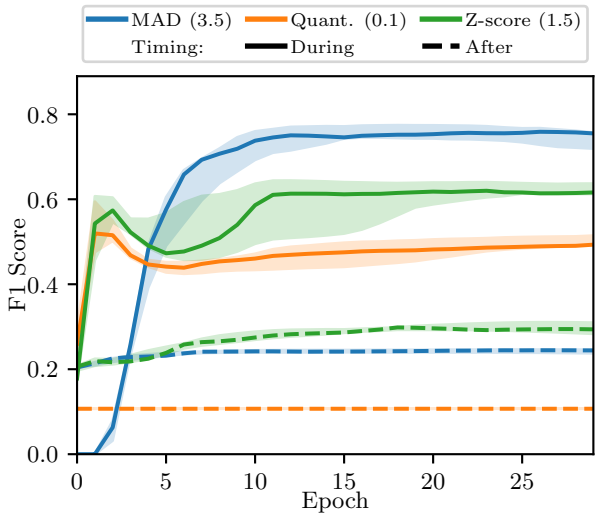

Figure B.1: Comparison of the F1 score evolution of different methods over the epochs on rare event detection task on the (`R`, `O`, `S`, `E`, and `Z`).

for the `during` procedure and the mask after the reconstruction by a RoseCDL model without an inline outlier method for the `after` procedure in Fig. B.1. The inline outlier detection module, particularly the MAD method, works by discarding the rare events from the dictionary learning part. These rare events are reconstructed with lower fidelity compared to the more prevalent patterns. This selective degradation results in higher reconstruction errors that are sharply localized at the rare events' positions and improves their detection. In opposition to that, in the absence of this module, rare events are still reconstructed less accurately than common patterns; however, the contrast in reconstruction quality is less distinct. This diminishes the precision of rare event detection and ultimately leads to a lower F1 score.

B.3 Inline outlier detection on real-world data

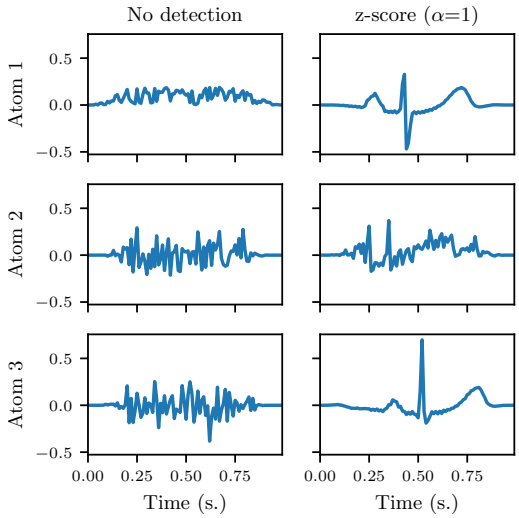

Figure B.2: Learned atoms with and without outliers detection method, on 10 bad trials of subject a02 of dataset Physionet Apnea-ECG.

We evaluate our approach through two complementary experiments. First, we consider the Physionet Apnea-ECG dataset (Penzel et al., 2000) without any preprocessing, thereby preserving the raw variability of the signals. From a 10 min ECG segment containing blocks of anomalies, we learned a three-atom dictionary (1 s each) to assess robustness against corrupted data (Fig. 2). Without outlier detection, the model converges to noise-like patterns,

whereas the inline anomaly detection mechanism suppresses high-variance anomalies and enables recovery of meaningful ECG atoms (Fig. B.2). While careful parameter tuning can sometimes yield similar results, it remains unreliable and computationally costly.

B.4 Influence of the regularization parameter

A key parameter of RoseCDL is the regularization coefficient λ, introduced in Sec. 2, which controls the sparsity of the learned activation vector. We express λ as a fraction of the maximum regularization value, $\lambda_{\max}$, corresponding to the smallest regularization value for which the activation vector is entirely zero. Note that when using the trimmed objective, $\lambda_{\max}$ computation is adapted to account for the modified loss. To analyze the effect of regularization on dictionary recovery, we compare recovery scores across different values of λ using synthetic data. The results, presented in Fig. B.3, demonstrate how varying λ influences the quality of the recovered dictionary. Our findings indicate that the best dictionary recovery is achieved when setting $\lambda = 0.1\lambda_{\max}$, effectively balancing sparsity and reconstruction accuracy.

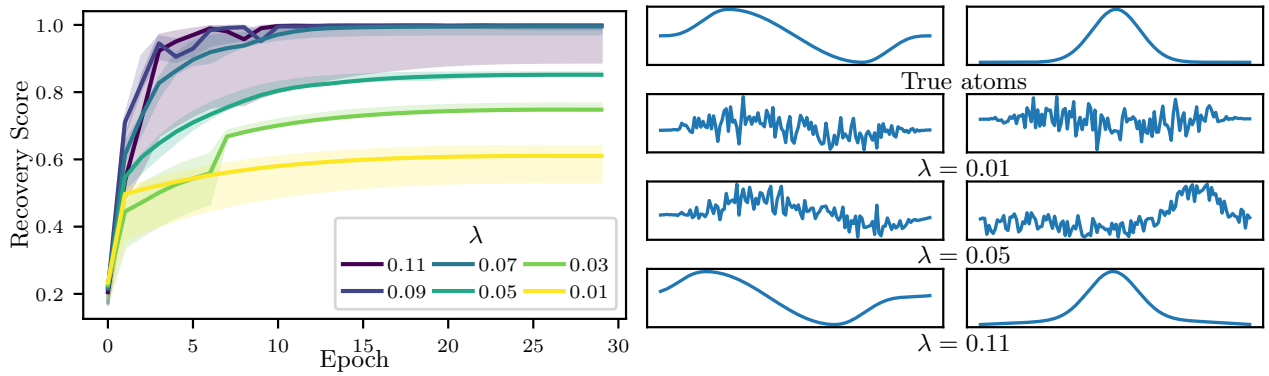

Figure B.3: (a) Impact of the regularization on the recovery score over the epochs. (b) Atoms recovered at the end of training for different regularization values.

B.5 Anomalies : Missed and detected examples

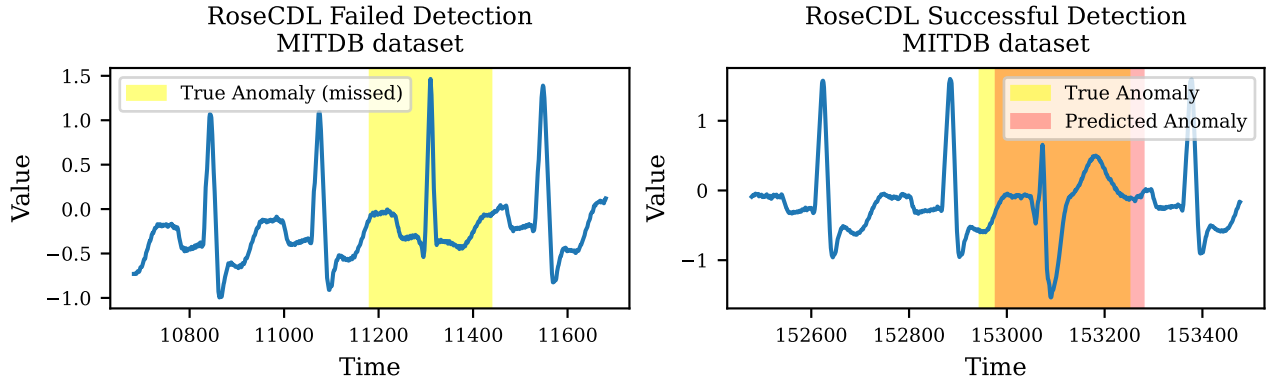

Figure B.4: Examples of missed and detected anomalies on the MITDB Arythmia dataset.

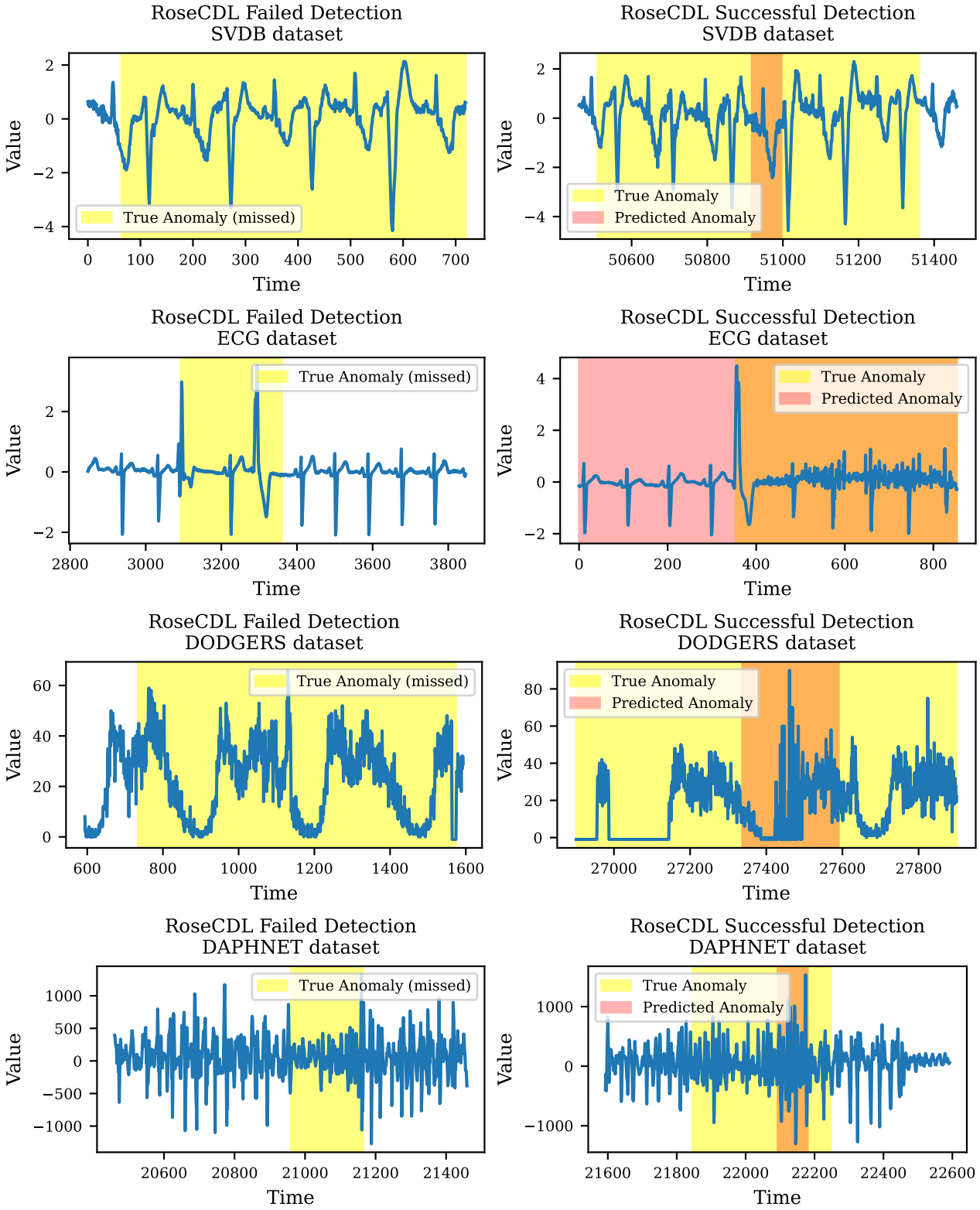

Figure B.5: Examples of missed (left) and detected (right) anomalies across datasets: MITDB, SVDB, ECG, Dodgers, and Daphnet.

C DATA SIMULATION

The synthetic multivariate 1D signals $X \in \mathbb{R}^{P \times T}$ used in Sect. 3 are generated from a dictionary $D \in \mathbb{R}^{K \times P \times L}$, a sparse activation vector $Z \in \mathbb{R}^{K \times (T-L+1)}$, and a random Gaussian noise $\varepsilon \sim \mathcal{N}\left(0, \sigma^2\right)$ as $X = D * Z + \varepsilon$. In this definition,

- P is the number of channels,
- T is the length of the signal,
- K is the number of atoms,
- L is the length of the atoms.

In the experiments conducted in Sect. 3, we generated signals of length $T = 50\ 000$ with $P = 2$ channels from dictionaries with $K = 2$ atoms of length $L = 64$. The atoms were generated from sine and gaussian waveforms, as illustrated in Fig. C.1. The activations Z were randomly generated sparse Dirac combs with sparsity 0.4 % and the noise level was set to $\sigma = 0.1$.

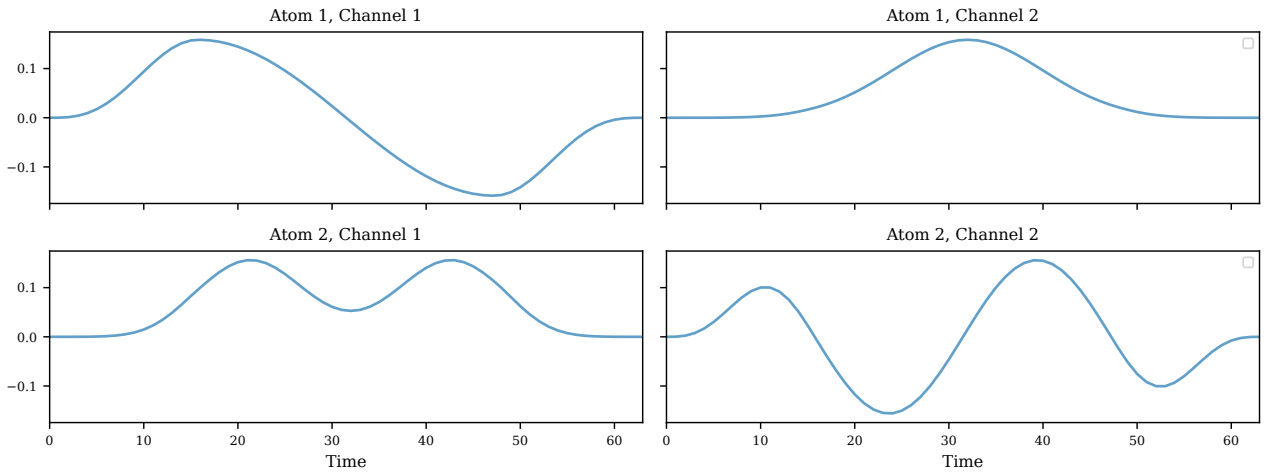

Figure C.1: True dictionary in experiments on synthetic data.

D DICTIONARY EVALUATION

In our methodology, we evaluate the effectiveness of a learned dictionary, denoted as $\widehat{\mathbf{D}} \in \mathbb{R}^{K' \times P \times L'}$, by comparing it against a set of true dictionary patterns, represented as $\mathbf{D} \in \mathbb{R}^{K \times P \times L}$ and computing a "recovery score", using the convolutional cosine similarity following optimal assignment, as defined by Moreau and Gramfort (2020). The learned dictionary and the true patterns are structured as three-dimensional arrays, where dimensions correspond to the number of atoms, channels, and atoms' duration. The learned dictionary may differ from the true dictionary in terms of the number of atoms and the length of time atoms, typically featuring more atoms and extended durations.

The evaluation process involves a computational step known as multi-channel correlation. In this step, each atom of the learned dictionary is systematically compared with each pattern in the true dictionary. This comparison is carried out channel by channel, aggregating the results to capture the overall similarity between the dictionary atom and the pattern.

After performing these comparisons for all combinations of atoms and patterns, we create a matrix that represents the correlation strengths between each pair. To objectively assess the quality of the learned dictionary, we use an optimization technique called the Hungarian algorithm. This algorithm finds the best possible "matching" between the learned dictionary atoms and the true patterns, aiming to maximize the overall correlation.

The final score, which quantifies the performance of the learned dictionary, is derived by averaging the values of these optimal matchings. This score is scaled between 0 and 1, where 1 represents the best possible performance. A higher score indicates that the learned dictionary more accurately represents the true dictionary patterns, providing a measure of its quality and effectiveness in capturing the essential features of the data.

Mathematically, the recovery score between the dictionaries $\widehat{\mathbf{D}}$ and $\mathbf{D}$ can be expressed as follow:

$$\text{score} = \frac{1}{K} \sum_{i=1}^{K} C_{i,j^*(i)} \ , \tag{24}$$

where $j^*(i), i = 1, \dots, K$ denote the results of the linear sum assignment problem (Crouse, 2016)[3] on correlation matrix $C \coloneqq \text{Corr}\left(\mathbf{D}, \widehat{\mathbf{D}}\right) \in \mathbb{R}^{K \times K'}$, with $\forall i \in [\![1\,, K]\!]$, $\forall j \in [\![1\,, K']\!]$,

$$C_{i,j} = \max_{l=1,\dots,L+L'-1} \text{Corr}_{\text{2D}}\left(D_i, \widehat{D}_j\right)[l] \in \mathbb{R} \ , \tag{25}$$

where $D_i \in \mathbb{R}^{P \times L}$ and $\widehat{D}_j \in \mathbb{R}^{P \times L'}$. The multivariate "2D" correlation between the two matrices D and $\widehat{D}$ is defined as follow:

$$\text{Corr}_{\text{2D}}\left(D, \widehat{D}\right) = \sum_{p=1}^{P} \text{Corr}_{\text{1D}}\left(d_p, \hat{d}_p\right) \in \mathbb{R}^{L+L'-1} \ , \tag{26}$$

where $d_p \in \mathbb{R}^L$ and $\hat{d}_p \in \mathbb{R}^{L'}$. The 1D "*full*" correlation between the two vectors d and $\hat{d}$ is defined as follow, $\forall t \in [\![1\,, L + L' - 1]\!]$:

$$\text{Corr}_{\text{1D}}\left(d, \hat{d}\right)[t] = \left(d * \hat{d}\right)[t - T + 1] = \sum_{l=1}^{L} d[l]\hat{d}\,[l - t + T] \in \mathbb{R} \ , \tag{27}$$

where $T \coloneqq \max(L, L')$.

E EXPERIMENTS SETUP

Experiment	Data size	Number of runs	Hardware
Runtime 1D	20 signals, 50,000 data points, 2 channels	50	GPU NVIDIA A40, 30 CPUs
Runtime 2D	2000×2000 grayscale images	20	GPU NVIDIA A40
Regularization impact 1D	10 signals, 30,000 data points, 1 channel	10	GPU NVIDIA A40
Regularization impact 2D	2000×2000 grayscale images	10	GPU NVIDIA A40
Inline vs After	2D image data	10	GPU NVIDIA A40
Physionet	10 trials of subject a02	10	GPU NVIDIA A40 and CPUs

[3] We use the `SciPy`'s implementation.