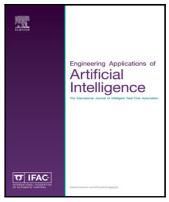
Engineering Applications of Artificial Intelligence 123 (2023) 106363

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

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



Learning under concept drift and non-stationary noise: Introduction of the concept of persistence

Kutalmış Coşkun *, Borahan Tümer

Computer Engineering Department, Faculty of Engineering, Marmara University, İstanbul, Turkey

ARTICLE INFO

Keywords:
Stochastic learning
Concept drift
Non-stationary noise
Parameter estimation
Dynamic learning rate

ABSTRACT

Learning from noisy data is a challenging task especially when the system under consideration has a non-stationary nature. The source of the noise is often assumed to be stationary, however the severity or characteristics of noise may also be time-varying, which causes multiple sources of drift in the collected data. This study introduces a novel adaptive learning rate approach to improve learning when the observations from a non-stationary system is altered by an also non-stationary source of noise. As an example to this approach, we propose Persistence Aware Robust Learner (PeARL), an online learning method that utilizes a novel concept called *persistence*, which is a local noisiness estimation metric to measure the correspondence of a signal to discrete white noise. Making use of this metric, PeARL is able to adaptively adjust the learning rate for each observation during learning to reduce the effect of noise. With this level of control on the learning rate, noisy instances have less disruptive effect on the maintained estimate. We experimentally evaluate PeARL on (a) systematically generated synthetic data and (b) real-world data, including accelerometer readings collected from people (`HASC2010corpus`) and current measurements from electric motors collected within the scope of EU-funded research project iRel40. The experiments reveal a *favorable region* of noise rate, in which the proposed method achieves up to 40% reduction in mean absolute error (MAE).

1. Introduction

Data collected from real-world systems rarely convey isolated information regarding an attribute of the source of the measurement, rather they are inevitably contaminated with other signals in a way that separation is not evidently possible. The other signals that are irrelevant to the system under consideration are named *noise*, and they also carry information about the state of the sources of noise (Vaseghi, 2008). From this point of view, learning from noisy data is essentially extracting useful information from an intermingled set of sources of data, making noise a fundamental cause of the limitation of the accuracy of decision-making in pattern recognition applications (Gupta and Gupta, 2019).

Another challenging aspect of real-world systems is their time-varying behavior. The dynamic (or evolving) nature of a system is often reflected in the data collected from it. Such a signal is said to be *non-stationary* and involve *concept drift* (Ditzler et al., 2015). Dealing with concept drift when there is noise in the collected data makes decision-making even more challenging, since a perceived change in data might be an indication of drift, or it might simply be caused by noise. Moreover, non-stationarity is not necessarily exclusive to the system under consideration, that is, the source of noise can also have a time-varying nature, which is also referred to as *heterogeneous* noise. In that case, there could be multiple sources of drift in the data, that, in principle, are not directly apparent to the learner, which shall distinguish these to maintain accuracy by *adapting* to the drifting behavior of the target system.

Generally, there is a trade-off between sensitivity to noise and adaptability to drift (Widmer and Kubat, 1996; Gama et al., 2014). Namely, being robust to noise causes learners to be more selective on detecting or adapting to non-stationarity. The effect of this trade-off is often manifested as a delayed or false detection of, or adaptation to changes. Depending on the application, if the amount and the characteristics of noise affecting the observations are known a-priori, one of these effects could be preferred. Being rarely the case (i.e., this could be infeasible or even impossible in some cases), complete awareness of this information is an important assumption to break.

This study focuses on online learning under concept drift and non-stationary noise. Let us start by describing a simple estimation problem, which will be extended later by introducing non-stationarity and noise. Consider a black-box system that outputs a value at each time step from a set of possible values. The output of the system is determined by an unknown probability distribution that assigns a probability to each possible value. The learner obtains observations about the output of the system as they occur, and the goal of the learner is to estimate

* Corresponding author.
E-mail addresses: kutalmis.coskun@marmara.edu.tr (K. Coşkun), borahan.tumer@marmara.edu.tr (B. Tümer).

https://doi.org/10.1016/j.engappai.2023.106363
Received 13 September 2022; Received in revised form 16 April 2023; Accepted 21 April 2023
Available online xxxx
0952-1976/

the probability distribution as accurate as possible. A straightforward yet effective approach to this problem is to count each outcome as they appear, then derive a probability distribution from the frequencies of each outcome. Now, let us introduce the first challenging component to the problem by stating that the underlying probability distribution that determines the output is non-stationary, in other words, it changes at unknown times. An important aspect of the non-stationarity in this setting is that there is no direct indication of a change other than the reflection of the change in the oncoming outputs of the system. Therefore, the straightforward approach of estimation by counting suffers from the fact that the observations before a change become irrelevant and can negatively affect the accuracy of the estimate. This form of the estimation problem, which is described more mathematically in Definition 1 in Section 3.1, necessitates an approach that is capable of discarding irrelevant information. A common strategy is to use a sliding window to limit the information used to calculate the estimate (Lu et al., 2018). Another approach is to update the estimate after each observation in a way that the probabilities of unobserved outcomes approach zero over time (Oommen and Rueda, 2006).

Now, let us extend the problem further by introducing imperfect observations. Consider a noise source that alters each observation with an unknown probability ω. In this case, an observation could be reliable (i.e., it represents what the target system actually produced) or it could be useless as it might have been changed in an unknown way that is unrelated to the behavior of the target system. The mechanism of the noise source that determines how the original value is changed is also unknown, and therefore we assume that the altered observation could be any possible value with equal likelihood. In addition, the probability ω is not necessarily fixed for all observations, which means that similar to the target system, the effect of noise also has a non-stationary nature. In this case, the learner should also estimate the *noisiness* of the observation before using it.

An important aspect of the problems defined here is that the estimator does not receive any feedback, neither about the estimation performance nor about the amount and characteristics of noise existing in observations. Similarly, the estimate calculated at each time step by the learner does not have any effect on the output of the target system. Also, the complete sequence is not presented to the learner at once, that is, observations are obtained in an *online* manner.

The type of noise described in the estimation problem is likely to occur in applications where streaming multidimensional continuous data are subjected to discretization by clustering before being utilized to train mathematical models or make a decision using the pre-trained models. Any distortion occurring in the data, which may be due to inherent measurement errors or incorrect transfer/transcription, has a chance to change the cluster that the observation is assigned to, resulting in a sequence of noisy assignments. An example would be training a Markov chain from sensor readings obtained from a real-world system. Since the state space of a Markov chain is defined to be finite or countably infinite (Chung, 1967), continuous measurements are often mapped to a set of discrete states, which are then used to construct the chain. Noise in raw data could (a) alter the cluster models in an inconsistent way and lead to suboptimal representation, or (b) change the mapping of a multidimensional continuous instance to a state, even if the model itself is trained on clean data (or obtained analytically). Such incorrect models or assignments would negatively impact the performance of system behavior identification and/or decision-making.

With this study, we propose an adaptive approach, which utilizes a noisiness estimation metric to measure how much an observation is affected by noise. Using this metric, the learning rate parameter is adaptively changed after each observation. As an example to this approach, we introduce PeARL, which utilizes *persistence*, a novel metric to measure to what extent the clean signal is intermingled with white noise.

The contribution that this study provides can be listed as follows:

- An adaptive learning rate approach is introduced to deal with concept drift and non-stationary noise.
- A novel and computationally efficient measure called *persistence* is introduced in order to evaluate the local similarity of a signal to discrete white noise.
- Utilizing persistence, a robust and online learning method called PeARL is proposed with two mapping schemes from persistence to learning rate.
- PeARL is experimentally evaluated on (a) synthetic data to test the performance on various cases (e.g. different drift and noise characteristics) and (b) real-world data, including accelerometer readings from humans and electric current readings from electric motors.

The paper is organized as follows. Section 2 discusses relevant studies from the literature. Section 3 provides the preliminary information regarding the proposed approach. Section 4 introduces the proposed adaptive learning rate approach and PeARL. Section 5 presents and discusses experiment results on synthetic and real-world data sets. Finally, in Section 6, the study is concluded by discussing the limitations and listing possible future work.

2. Related work

Due to the nature of this work that combines two major challenges in online learning, related studies in the literature are from two main clusters, namely research on (a) learning under concept drift, and (b) learning from noisy data.

Concept drift research addresses the practical need for Machine Learning (ML) models to adapt to changes in the data they are trained on. In many real-world applications, data distributions are non-stationary, and models that do not account for concept drift will inevitably degrade in performance over time. Methods for learning in presence of concept drift can be categorized into two major groups (Ditzler et al., 2015), namely (a) active approaches, which aim at detecting the drift to trigger an adaptation mechanism and (b) passive approaches, which always update the model regardless of the presence of drift. The learning method proposed in this paper, PeARL, is based on SLWE (Oommen and Rueda, 2006), which is a foundational algorithm from the passive approach group. It has previously been extended to find changes in Markov dependencies (Aslancı et al., 2017), better adapt to abruptly changing environments (Hammer and Yazidi, 2018; Coşkun and Tümer, 2022) and online classification of data streams (Tavasoli et al., 2019). However, to the best of our knowledge, SLWE has not been used before to learn from noisy data.

Learning from noisy data is a subset of learning from data with *suboptimal quality*, and it attracts attention due to the ever-increasing popularity of ML methods and their applications to real-world problems. Characterization and negative effects of noise have been previously shown on supervised learning (Nettleton et al., 2010; Twala, 2013), unsupervised learning (Dave, 1991) and reinforcement learning (Fox et al., 2017). From the ML point of view, noise might be *attribute noise*, where some values of a data instance are erroneous or *class noise*, where the assigned label is wrong (Gupta and Gupta, 2019). Due to the unsupervised nature of this study, we specifically consider attribute noise rather than class noise.[1] Besides possible measurement or environment related sources, noise is also sometimes intentionally added for privacy-related reasons (Kumar et al., 2019; Wang and Hegde, 2019; Mireshghallah et al., 2020), which is an important attempt to make ML models usable in cases where user privacy is a critical concern, further increasing the importance of methods that can manage noisy data.

Due to the adverse effect of noise on the performance (e.g. classification accuracy or estimation/prediction capability), algorithms usually deal with noisy data by either (Gupta and Gupta, 2019),

[1] However, considering the discretization of continuous data by clustering, incorrect assignment of an input instance to a cluster due to noise has a similar effect to class noise from the point of view of the estimator.

(a) filtering, which is identifying and disregarding noisy instances (Kubica and Moore, 2003; Fefilatyev et al., 2012),
(b) polishing, which aims to extract clean values by making adjustments on the noisy data (Yan Zhang et al., 2005; Liebchen et al., 2007), or
(c) ignoring, which requires a robust algorithm that adaptively adjusts the model and learn directly from the noisy data (Cesa-Bianchi et al., 2011; Kang et al., 2020; Xu and Chen, 2022; Song et al., 2022).

In our case, since the data are not available to the learner at once, a preliminary analysis of data for filtering or polishing is not applicable. Thus, our approach fits the group (c) the most, and the most related studies to the problem defined previously are from this group.

A notable study (Song et al., 2015) from the robustness group (c) proposes a method to change the learning rate of stochastic gradient descent (SGD) as a function of heterogeneity, which occurs due to using multiple sources of data with different noise levels. This, in principle, corresponds to learning from data with *varying quality*, which is an important aspect of many ML applications (Crammer et al., 2005; Fazelpour et al., 2015; Kläs and Vollmer, 2018). The motivation of research on learning from data with varying levels of noise is especially evident in applications which require learning from sensitive data collected from people with different privacy preferences. Similar to Song et al. (2015), our approach also utilizes the idea of controlling the learning rate according to an estimation about noise. However, the problem that Song et al. (2015) focuses on is stationary, that is, there is no inherent drift in the data. Although similar approaches have previously been proposed to achieve SGD with noise-adaptive learning rate (Sasaki et al., 2011; Van Hulle, 1995), to our knowledge, there has not been a study that combines the concept drift and heterogeneous noise.

Another noteworthy study about learning with noisy data is (Hao et al., 2022), which introduces a model-agnostic approach to deal with noisy labels. The method utilizes a noise setting detection module to select one of the noise-robust loss functions for different noise distributions. Likewise, Hao et al. (2022) also does not consider non-stationary data, however, our persistence metric shares a similar intention to be used in a model-agnostic way, since it can be easily mapped to any attribute of the learning method that controls the effect of individual samples.

In addition to learning under concept drift, the challenge brought by non-stationary noise lies in the fact that its severity can vary across different parts of the data, making it difficult to understand and adapt. Since information regarding the non-stationarity of noise is not directly available to the learner, a common approach is to track some local characteristics of the noisy signal to recognize possible changes in noise and adapt accordingly. A particular study that follows this approach is Kimura and Shigeta (2013), which utilizes a coil-sensitivity and/or noise map to prioritize signal parts with less noise. However, unlike ours, their goal is to correct *spatially* inhomogeneous noise occurring in magnetic resonance imaging. The mechanism we propose, in principle, also works similarly in the temporal sense, by measuring the persistence of a local window of observations to control the learning rate.

3. Preliminaries

This section provides information about preliminary concepts and studies that are referred to throughout the paper.

3.1. Estimation without noise

Let us consider a simple estimation problem, the goal of which is to estimate the underlying probability distribution for a random variable. At each time step, the estimator receives an observation about the realization of the random variable. The random variable is effectively a black-box for the estimator, besides the set of possible values and a stream of observations. Also, the underlying probability distribution is non-stationary, that is, it changes at unknown times without any observable indication apart from the effect on the new realizations. This problem is given in Definition 1.

Definition 1 (*Estimation without Noise*). Let $X : \Omega_X \rightarrow E_X$ be a discrete random variable with non-stationary multinomial distribution $P_X : E_X \rightarrow [0, 1]$ and $X[n] \in E_X$ be the realization of X at time n. Here, we define the problem of estimating P_X from the streaming clean sequence $S_{\text{clean}} = \langle X[n], X[n+1], \dots, X[n+K] \rangle$ of realizations of X.

The challenging aspect of the problem in Definition 1 is that, P_X is non-stationary, hence the goal is to maintain the estimate $\hat{P}_X$ as close to P_X as possible while P_X changes at unknown times. This problem has received attention before (Oommen and Rueda, 2006), however the realizations were assumed to be perfect. With this study, we extend this problem by introducing noisy observations.

3.2. Stochastic Learning Weak Estimator (SLWE)

SLWE (Oommen and Rueda, 2006) is an efficient method that estimates the underlying probability distribution of a discrete random variable from the realizations of that random variable. This is achieved by utilizing an update rule that is based on the linear reward-inaction (L_{R-I}) reinforcement scheme from the Learning Automata (LA) context. The estimate is maintained throughout learning by updating it after each observation with the rule given in Eq. (1), where $\hat{p}_{s_i}(n+1)$ is the probability of observing the outcome s_i as $(n+1)$th observation, λ is the learning coefficient and X_n is the realization of random variable X at n.

$$\hat{p}_{s_i}(n+1) \leftarrow \begin{cases} \hat{p}_{s_i}(n) + (1-\lambda) \sum_{j \neq i} \hat{p}_{s_j}(n), & if\ X_n = i \\ \lambda \hat{p}_{s_i}(n), & if\ X_n \neq i \end{cases} \tag{1}$$

The advantage of such an approach shows up when the underlying distribution is non-stationary. While other statistically powerful methods such as maximum likelihood estimation (MLE) struggle to adapt to time-varying behavior in the sequence, SLWE is able to quickly *unlearn* unrelated information and maintain accurate estimates.

The learning coefficient λ plays an important role at the convergence of the estimate. Being fixed during learning, λ determines how fast the estimate approaches the new value after a change. However, variance of the estimate is directly proportional to the λ, introducing a trade-off between adaptiveness and stability (Coşkun and Tümer, 2022).

The problem that SLWE is proposed for by default does not consider that some observations might be affected by noise. Such instances are treated the same by the update rule, which manifests as perturbations in estimate during learning. Depending on the presence of noise, the estimate might never get as close to the correct value as desired. Focusing on this extension of the problem, the method we propose controls the amount of updates via controlling λ during learning, depending on the likeliness of an observation being altered by noise.

3.3. White noise

Due to being a widely observed phenomenon, many signal processing applications involve modeling, reducing or filtering noise (Vaseghi, 2008). These operations exploit as much knowledge as possible about the nature of noise being dealt with in that particular application. The more precisely known the characteristic of noise, the better the reduction of adverse effects usually is. When no information about noise is available, or it is not possible to make an assumption; white noise, which is the most random signal possible, can be considered.

White noise is uncorrelated random signal with equal intensity at different frequencies (Vaseghi, 2008; Zhang et al., 2023). For discrete

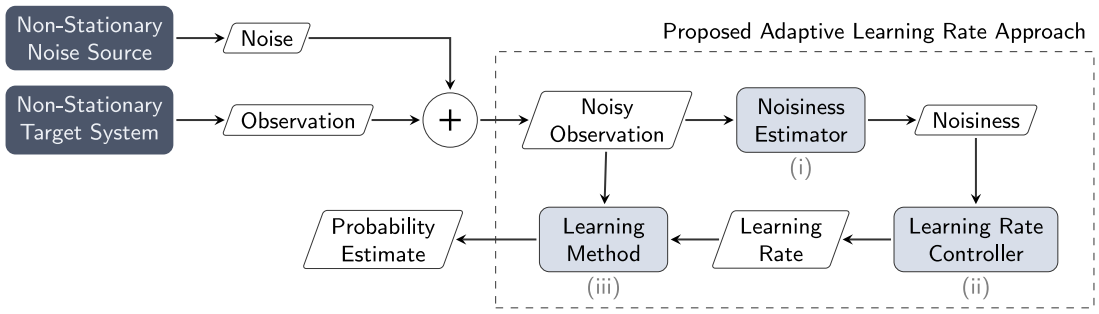


Fig. 1. The flow diagram of the proposed adaptive learning rate approach. This flow is repeated for each observation obtained from the system.

signals, this means that all possible outcomes are equally probable, which can be modeled by a uniformly distributed discrete random variable. This study focuses on estimating the underlying probability distribution of a random variable from realizations that are contaminated by discrete white noise.

4. Proposed adaptive learning rate approach and PeARL

As previously described, the problem we focus on in this study is to estimate the underlying probability distribution of a black-box non-stationary system through a sequence of observations (realizations of a discrete random variable) that are altered by a non-stationary noise source. Now, let us start by defining the extended form of Definition 1, so that the observations are not perfect.

Definition 2 (*Estimation with Noise*). Let $W : \Omega_W \rightarrow E_W$ be a discrete random variable with the uniform distribution $P_W : E_W \rightarrow 1/|E_W|$, where $|E_W|$ is the cardinality of the set of all possible outcomes (sometimes also denoted by $|\Sigma|$). Here, W represents a white noise process, since all possible outcomes are equiprobable and uncorrelated (Vaseghi, 2008). Now, let Z be another random variable so that,

$$Z = \begin{cases} X, & \text{with probability } 1-\omega \\ (X+W) \bmod |E_X|, & \text{with probability } \omega \end{cases} \tag{2}$$

where ω is the *noise rate* and X is the non-stationary random variable as given in Definition 1. In this case, the problem becomes estimating P_X from the noisy sequence $S_{\text{noisy}} = \langle Z[n], Z[n+1], \dots, Z[n+K] \rangle$.

In the problem described in Definition 2, ω is not necessarily constant for all parts of S_{noisy}, pointing to the concept of non-stationary noise. That is, similar to P_X, it might change at unknown times not necessarily in a synchronous way with P_X. This, in principle, results in two sources of drift occurring in observed data, one from the changes in noise behavior (ω) and one from the changes (concept drift) in system behavior (P_X). Since the learner has no direct access to this information, it shall process the observation sequence and respond accordingly.

The approach we propose involves three components, namely (i) an online metric that measures the *noisiness* of the observation, (ii) a controller that calculates the best learning rate for the update to be performed on the estimate, and (iii) an online learning method that maintains a probability estimate using the noisy observations and the learning rate. The purpose of the structure we propose is to control the amount of information that the noisy observation contributes to the maintained probability estimate, depending on the estimated noisiness level. This approach is depicted as a flow diagram in Fig. 1.

PeARL is an example to the approach shown in Fig. 1, and uses (i) persistence for noisiness estimation, (ii) proposed linear mapping schemes as the learning rate controller, and (iii) SLWE as the learning method. Now, we discuss the components of PeARL in Sections 4.1 and 4.2.

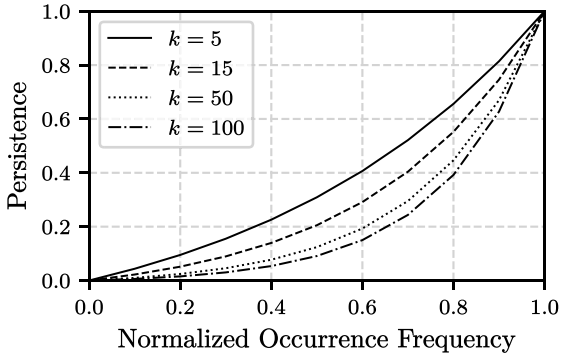


Fig. 2. Persistence function.

4.1. The concept of persistence

As a metric to estimate the noisiness of a given observation (component (i) in Fig. 1), we propose persistence. A key distinction between perceived changes due to white noise and true changes in the behavior of the observed system, which we exploit in this study, is based on the difference in time-dependent behavior. By definition, white noise affects each observation from the system in an unknown way with uniform probability distribution. On the contrary, a true concept drift (especially sudden drift) has a *persistent* effect on observations, which was not present before the drift. Therefore, the main idea behind our approach is that, repetitive occurrence of unlikely events is a sign of drift.

In order to measure the occurrence frequency of an event in a given time frame, we define the concept of persistence in Definition 3, which is essentially used to differentiate noisy observations from clean observations.

Definition 3. Persistence of occurrence of event $x_i \in E$, namely ρ_{x_i}, is defined as,

$$\rho_{x_i} = \frac{k^{l_{x_i}/L} - 1}{k - 1} \tag{3}$$

where $k > 1$ is the growth factor (approaches linear growth as $k \rightarrow 1$), l_{x_i} is the number of occurrences of event $X_n = x_i$ observed in memory and L is the look-back distance (memory size) in terms of events.

Persistence function with different k values is visualized in Fig. 2.

As seen in Eq. (3), persistence ρ_{x_i} can take values from $[0, 1]$. If all of the observations in the memory are the same event, then the normalized frequency l_{x_i}/L of that event becomes 1, therefore the persistence of that event ρ_{x_i} becomes 1. On the opposite, if an event is never observed before, we get $\rho_{x_i} = 0$. Notice that we link the noisiness of an observation to how less its persistence is, which is the key idea to deal with discrete white noise as described in Eq. (2).

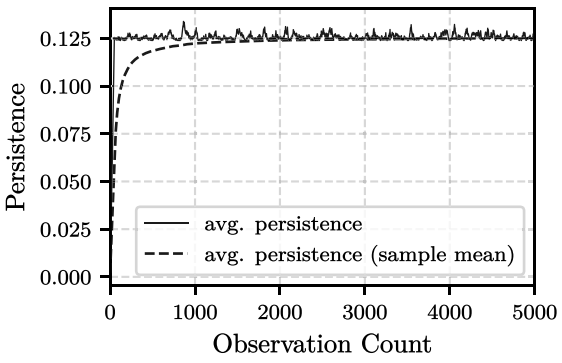


Fig. 3. Average persistence of white noise is plotted for 5000 observations with $k = 5$ and $L = 50$. The theoretical value calculated as given in Theorem 1 is 0.12384 while the final empirically obtained value is 0.12506.

Naturally, since l_{x_i} is the frequency, sum of all normalized frequencies add up to 1 as shown in Eq. (4), where $|\Sigma|$ is the cardinality of the set of possible values that x_i can take.

$$\sum_{i=1}^{|\Sigma|} l_{x_i}/L = 1 \tag{4}$$

The vector $\rho_X = \{\rho_{x_1}, \rho_{x_2}, \ldots, \rho_{x_{|\Sigma|}}\}$ involves persistence of all possible events. Also, the average persistence $\bar{\rho}_X$ can be defined as shown in Eq. (5), which is simply the average of persistence values in ρ_X.

$$\bar{\rho}_X = \frac{1}{|\Sigma|} \sum_{i=1}^{|\Sigma|} \rho_{x_i} \tag{5}$$

Now, we discuss why persistence is a useful metric for noisiness estimation in Section 4.1.1 and then we discuss the computational complexity of persistence calculation in Section 4.1.2.

4.1.1. Persistence of white noise

Persistence is a measure of how repetitive an outcome is. White noise, considering the equiprobability of each outcome, is expected to have the minimum possible persistence for each and every event possible. This also causes the average persistence to be minimum for white noise. This lower bound is studied in Theorem 1 and empirically validated in Fig. 3.

Theorem 1. *Let X be an independent and identically distributed (IID) uniform multinomial random variable, then $\mathbb{E}[\rho_X] = \frac{k^{1/|\Sigma|}-1}{k-1}$.*

Proof. Due to the law of large numbers and X being uniformly distributed, the frequency of each possible outcome is expected to be equal after a large number of realizations. Considering the memory size (L) in Eq. (3), at any selected time frame, an equal number of realizations of each event is expected to be found in the memory, which is:

$$\mathbb{E}\left[l_{x_i}\right] = \frac{L}{|\Sigma|} \tag{6}$$

By plugging Eq. (6) to Eq. (3), we get the expected persistence of each event:

$$\mathbb{E}\left[\rho_{x_i}\right] = \frac{k^{1/|\Sigma|}-1}{k-1} \tag{7}$$

Since this is the case for all events, expected persistence of the random variable X is also $(k^{1/|\Sigma|}-1)/(k-1)$. □

Now, we show in Theorem 2 that the average persistence is minimum only for white noise.

Theorem 2. *Let $\bar{\rho}_X$ be the average persistence of random variable X. Then, $\bar{\rho}_X$ is minimum if and only if X is uniformly distributed.*

Proof. Let A be the statement that $\bar{\rho}_X$ is minimum and B be the statement that X is uniformly distributed. We prove $A \iff B$ by first showing that $A \implies B$ by writing $\bar{\rho}_X$ as in Eq. (8) by Eqs. (3) and (5).

$$\bar{\rho}_X = \frac{1}{|\Sigma|} \sum_{i=1}^{|\Sigma|} \frac{k^{l_{x_i}/L} - 1}{k-1} \tag{8}$$

For $\bar{\rho}_X$ to be minimum $\nabla\bar{\rho}_X$ should be zero and $\nabla^2\bar{\rho}_X$ should be greater than zero. $\nabla\bar{\rho}_X$ is written as:

$$\nabla\bar{\rho}_X = \nabla\left[\frac{1}{|\Sigma|}\left(\frac{k^{l_{x_1}/L} - 1 + k^{l_{x_2}/L} - 1 + \cdots + k^{S/L} - 1}{k-1}\right)\right] \tag{9}$$

where (from Eq. (4)) $S = L - \sum_{i=1}^{|\Sigma|-1} l_{x_i}$. Then, we have,

$$\nabla\bar{\rho}_X = \frac{\ln k}{(k-1)|\Sigma|L}\left[(k^{l_{x_1}/L} - k^{S/L})d_1 + (k^{l_{x_2}/L} - k^{S/L})d_2 + \cdots + (k^{l_{x_{|\Sigma|-1}}/L} - k^{S/L})d_{|\Sigma|-1}\right] \tag{10}$$

for which to be zero, the following must hold:

$$(k^{l_{x_1}/L} - k^{S/L})d_1 + (k^{l_{x_2}/L} - k^{S/L})d_2 + \cdots + (k^{l_{x_{|\Sigma|-1}}/L} - k^{S/L})d_{|\Sigma|-1} = 0 \tag{11}$$

This requires each independent component of $\nabla\bar{\rho}_X$ be equal to zero, resulting in $|\Sigma| - 1$ independent equations:

$$k^{l_{x_i}/L} = k^{S/L} \text{ for } i \in \{1, 2, \ldots, n-1\} \tag{12}$$

Therefore we get:

$$\begin{aligned} l_{x_i} &= S = L - \sum_{j=1}^{|\Sigma|-1} l_{x_j} \\ 2l_{x_i} &= L - (L - l_{x_i} - l_{x_{|\Sigma|}}) \\ l_{x_i} &= l_{x_{|\Sigma|}} \quad \forall i \in \{1, 2, \ldots, |\Sigma| - 1\} \end{aligned} \tag{13}$$

which indicates uniform distribution.

Similarly, for $\nabla^2\bar{\rho}_X$ we have,

$$\nabla^2\bar{\rho}_X = \frac{\ln^2 k}{(k-1)|\Sigma|L^2}(k^{l_{x_1}/L} + k^{l_{x_2}/L} + \cdots + k^{l_{x_{|\Sigma|}}/L}) \tag{14}$$

which is always positive since $k > 1$, $L > 0$ and $l_{x_i} \geq 0$ by definition.

Then, we show that $B \implies A$ by using the fact that a uniformly distributed X leads l_{x_i}/L to be equal to $1/|\Sigma|$ as shown in Theorem 1. We need to show that $\nabla\bar{\rho}_X = 0$ for $l_{x_i}/L = 1/|\Sigma|\ \forall i \in \{1, 2, \ldots, |\Sigma|\}$. By plugging $1/|\Sigma|$ for all l_{x_i}/L in Eq. (10), we get zero. Similarly, $\nabla^2\bar{\rho}_X$ is always positive.

Since we showed that both $A \implies B$ and $B \implies A$, we can state that $A \iff B$. □

4.1.2. Complexity of persistence calculation

Persistence of event x can be calculated in $O(1)$ time when the l_x/L ratio is known. One way to achieve this is to keep $|\Sigma|$ occurrence frequencies in memory and update them after each observation. Maintaining these frequencies include increasing and decreasing the occurrence count of the newest and the oldest observations, respectively. Increasing the corresponding frequency is trivial, however decreasing the oldest one requires also keeping all L observations in memory to correctly calculate l_x/L ratio. As a result, this way of persistence calculation requires $O(1)$ time for each observation, with the total memory complexity of $O(L + |\Sigma|)$.

An alternative way would be to only keep L observations in memory, with the cost of $O(L)$ time complexity to calculate l_x/L after each observation. This means calculating l_x from scratch every time. With this approach, persistence calculation costs $O(L)$ time and $O(L)$ memory.

Considering these, it would be reasonable to choose the first way, since additional $O(|\Sigma|)$ memory is negligible compared to additional $O(L)$ time.

4.2. Adaptive learning rate

In order to mitigate the disruptive effect of noise, learning rate is adaptively changed after each observation considering the persistence calculated. We propose two mapping schemes from persistence to learning rate.

4.2.1. Event-based linear mapping from persistence to learning rate

One way to change learning rate according to persistence is to use an event-based linear mapping from $[0, 1]$ to $[\eta_{min}, \eta_{max}]$ as shown in Eq. (15), where $\rho_{\max} = 1$ and $\rho_{\min} = 0$.

$$\begin{aligned} \frac{\eta - \eta_{\min}}{\eta_{\max} - \eta_{\min}} &= \frac{\rho - \rho_{\min}}{\rho_{\max} - \rho_{\min}} \\ (\eta - \eta_{\min})(\rho_{\max} - \rho_{\min}) &= (\eta_{\max} - \eta_{\min})(\rho - \rho_{\min}) \\ \eta &= \eta_{\min} + \rho(\eta_{\max} - \eta_{\min}) \end{aligned} \tag{15}$$

This mapping, which we call Event Persistence to Learning Rate (EP2LR), enables to calculate a new learning rate based on the persistence value obtained after each observation. Minimum and maximum ρ values correspond to minimum and maximum values for η.

4.2.2. Linear mapping from average persistence to learning rate

Another possible mapping scheme, which we call AP2LR, utilizes the characteristics of average persistence. As shown in Theorem 1, the minimum possible average persistence value, which is obtained from white noise, is $(k^{1/|\Sigma|} - 1)/(k - 1)$. Similarly, the maximum possible average persistence is also bounded and can be shown to be equal to $1/|\Sigma|$ with a similar approach given in Theorem 1. Using these bounds, a new learning rate value can be calculated from the average persistence as shown in Eq. (16).

$$\begin{aligned} \frac{\eta - \eta_{\min}}{\eta_{\max} - \eta_{\min}} &= \frac{\bar{\rho} - \bar{\rho}_{\min}}{\bar{\rho}_{\max} - \bar{\rho}_{\min}} \\ (\eta - \eta_{\min})(\bar{\rho}_{\max} - \bar{\rho}_{\min}) &= (\eta_{\max} - \eta_{\min})(\bar{\rho} - \bar{\rho}_{\min}) \\ \eta &= \eta_{\min} + \frac{(\eta_{\max} - \eta_{\min})(\bar{\rho} - \frac{k^{1/|\Sigma|}-1}{k-1})}{1/|\Sigma| - \frac{k^{1/|\Sigma|}-1}{k-1}} \end{aligned} \tag{16}$$

4.3. Persistence Aware Robust Learner (PeARL)

Utilizing the proposed persistence metric, here we provide the pseudocode for PeARL with EP2LR in Algorithm 1. The other mapping scheme, AP2LR, can also be used in Algorithm 1 using Eq. (16).

Algorithm 1 Persistence Aware Robust Learner (PeARL)

Require: $|\Sigma|$, $\eta_{\min}$, $\eta_{\max}$, k, L
1: initialize queue Q of observations, empty vector l, empty stack m, uniform estimate $\hat{P}$
2: **while** Q is not empty **do**
3: get observation o from Q and push to m
4: $l_o \leftarrow l_o + 1$
5: **if** $|m| > L$ **then**
6: $x \leftarrow m.\text{pop}()$ ▷ *remove oldest observation from memory*
7: $l_x \leftarrow l_x - 1$
8: **end if**
9: $\rho \leftarrow (k^{l_o/L} - 1)/(k - 1)$ ▷ *persistence calculation*
10: $\eta \leftarrow \eta_{\min} + \rho(\eta_{\max} - \eta_{\min})$ ▷ *EP2LR mapping*
11: $\hat{P} \leftarrow (1 - \eta)\hat{P}$ ▷ *SLWE update*
12: $\hat{P}_o \leftarrow \hat{P}_o + \eta$
13: **end while**

5. Experimental evaluation

In this section, we report and discuss experiment results obtained on synthetic and real-world data.

5.1. Experiments with synthetic data

The proposed method, PeARL, is tested on systematically generated data to measure the effect of persistence based learning rate control in different cases. We perform two kinds of experiments to showcase what PeARL is capable of, which are:

(a) Stationary Noise: ω is kept constant during each run, however we perform multiple runs with increasing values of ω to test the robustness. With this approach, we examine the region in $\omega \in [0, 1]$ where the proposed method becomes beneficial.
(b) Non-stationary Noise: ω is changed during runs. The changes in P and ω might overlap or occur at different times so that the following cases are covered: (i) P and ω are constant, (ii) P changes, but ω is constant, (iii) P is constant, but ω changes, and (iv) P and ω both change.

While generating data, the following parameters are controlled:

- $|\Sigma|$ is the cardinality of the alphabet that is used to generate the sequence (i.e., the number of unique values that random variable X can take).
- For non-stationary P:
  - d_p is the duration of each regime in terms of observations.
  - n_p is the number of regimes in the sequence.
  - $\Delta P_{\min}$ and $\Delta P_{\max}$ are the minimum and maximum difference of consecutive regimes, respectively.
- For non-stationary noise:
  - d_n is the duration of each noise regime.
  - $\Delta\omega_{\min}$ and $\Delta\omega_{\max}$ are similarly the minimum and maximum difference in ω for consecutive regimes.

We start data generation by generating P, which is a matrix whose rows are the probability distribution for each regime and columns are the probabilities of each possible outcome, as shown in Eq. (17).

$$P = \begin{bmatrix} P_{1,1} & P_{1,2} & \dots & P_{1,|\Sigma|} \\ P_{2,1} & P_{2,2} & \dots & P_{2,|\Sigma|} \\ \vdots & \vdots & \ddots & \vdots \\ P_{n,1} & P_{n,2} & \dots & P_{n,|\Sigma|} \end{bmatrix} \tag{17}$$

The probability distribution of the first regime, $P_1 = \{P_{1,1}, P_{1,2}, \dots, P_{1,|\Sigma|}\}$, is obtained by generating a Dirichlet distribution with $\alpha = \{\alpha_i = 1\}$ where $i \in \{1, \dots, |\Sigma|\}$. Remaining distributions are generated similarly so that the following condition, which ensures that the difference between each consecutive row is between desired bounds, holds:

$$\Delta P_{\min} \leq \frac{1}{|\Sigma|} \sum_{j=1}^{|\Sigma|} |P_{i,j} - P_{i-1,j}| \leq \Delta P_{\max} \tag{18}$$

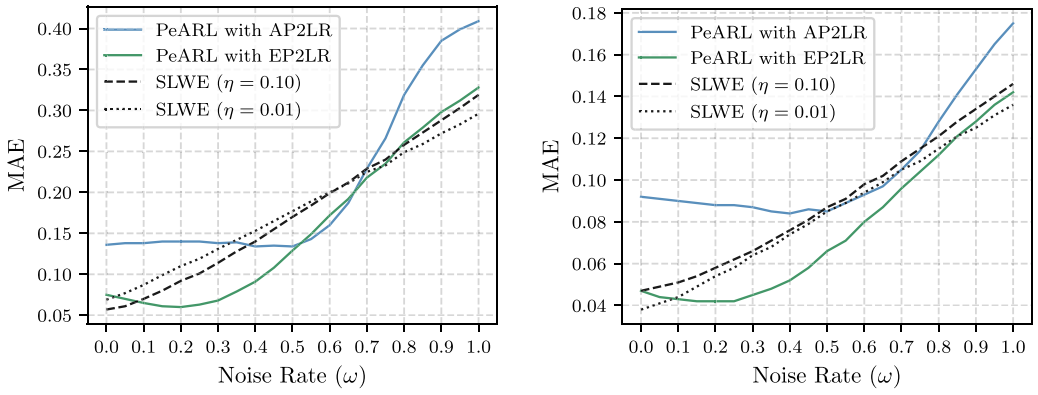
After generating P, d_p tokens are sampled for each regime according to the probability distribution P_i. As a result, we get a clean sequence (i.e., without any noise) of $N = n_p d_p$ observations. Then, we inject white noise by altering each token in the sequence with probability ω. Thus, when ω is constant, ωN incorrect instances are expected to occur in a sequence of N tokens.

The performance of estimators are measured by comparing the correct probability distribution P_i of the active regime and the estimate $\hat{P}_i$ after each observation. This comparison is done by calculating the MAE as shown in Eq. (19),

$$\text{MAE} = \frac{1}{N|\Sigma|} \sum_{j=1}^{N} \sum_{k=1}^{|\Sigma|} |P_{i,k} - \hat{P}_{i,k}[j]| \tag{19}$$

where N is the total number of observations and $\hat{P}[n]$ is the estimate after nth observation.

We run two instances of SLWE with $\lambda = \{1 - \eta_{\min}, 1 - \eta_{\max}\}$ to show the proposed mapping from ρ (persistence) to η (learning rate) is effective. We are not aware of a method that can deal with heterogeneous



(a) $|\Sigma| = 4$ (b) $|\Sigma| = 8$

Fig. 4. Performance of estimators on constant noise rate (ω). In this set of experiments, we set $d_p = 600$ and $n_p = 5$. $\Delta P_{\min}$ is set to $\{0.4, 0.2\}$ for $|\Sigma| = \{4, 8\}$, respectively. Since the bounds of ΔP are $\{0, \frac{2}{|\Sigma|}\}$, a single value might result in impossible cases to generate. Hence $\Delta P_{\min}$ is selected so that for all $|\Sigma|$ values we generate significant and sudden changes in P. In all cases, $\Delta P_{\max}$ is set to 1.

noise with concept drift on online learning tasks. Therefore, by comparing PeARL with two SLWE instances that use the learning rate bounds of PeARL, our goal is to show that the proposed adaptive learning rate approach that utilizes persistence is useful. In this experiment set, we empirically selected $\eta_{\min} = 0.01$, $\eta_{\max} = 0.1$, $k = 5$ and $L = 15$ for PeARL and used the mapping schemes described in Section 4.2.

Fig. 4 shows the results obtained from experiments with fixed ω and $|\Sigma| \in \{4, 8\}$.

In the experiment results shown in Fig. 4, we observe that PeARL, especially with the EP2LR mapping, achieves significantly less MAE than both SLWE instances in a region of ω, which we call the *favorable region*. This region involves noise rate values that the proposed adaptive learning rate approach is worth applying, and we argue that these values are likely to be encountered in real-world applications. Based on the experiments conducted, we can say that the start/end points (which we call the *turning points*) of the favorable region can change depending on the parameters of persistence and the nature of the sequence. However, generally they are observed at around $\omega \approx 0.15$ and $\omega \approx 0.75$.

Before the first turning point where the noise rate is relatively small, applying the proposed learning rate control seems to negatively affect MAE. This can be linked to the unnecessary reduction of learning rate due to persistence being small locally and is a good example of the downside of being unnecessarily selective during learning. Likewise, after the second turning point where the noise rate is large, it is better to be more exploitative since most of the observations are noise, which results in exploitative SLWE performing the best.

Comparing the mappings proposed in Section 4.2, we observe that the event-based mapping significantly outperforms the average persistence based mapping in most of the cases. This could be linked to average persistence based mapping being less responsive to a noisy observation, since the average value is more robust to outliers. Although there is a region in Fig. 4(a) where this is not the case and the average persistence based mapping achieved the best results among all estimators, it is not replicated in Fig. 4(b) where $|\Sigma| = 8$. Also, although the error of PeARL with AP2LR is relatively high, it is observed to be quite robust to the increasing noise rate up to $\omega \approx 0.5$.

Fig. 5 shows the results obtained from experiments with changing ω. In contrast to the experiments in Fig. 4, here we change the noise rate during each run to test the adaptation capabilities of the methods. The changes in $\hat{P}$ are indicated with vertical dashed lines whereas the changes in the noise rate is shown with the ω signal with a separate vertical axis in addition to the error axis. The changes in $\hat{P}$ and ω are not necessarily synchronized, however such a case is not excluded either.

As visible in absolute and cumulative error curves in Figs. 5(a) and 5(b), when ω is also non-stationary, PeARL achieves notably less error compared to fixed-learning-rate estimators. The effect of the proposed method is manifested to the CAE curve as more slowly increasing error when the instantaneous noise rate is within the favorable region. In experiments with changing ω, PeARL is run with EP2LR mapping as it outperformed AP2LR.

5.1.1. Custom scenarios

Here we simulate scenarios for some special non-stationary noise characteristics.

In Fig. 6(a), ω switches to a relatively high value (0.5) for some time and then switches back to normal, which resembles an interference occurred in observation sequence. There are many possible reasons for such a case happening in real-world applications, which could be temporary failures in sensors or intentional/unintentional jamming affecting the communication. As visible in Fig. 6(a), the cumulative error of the proposed method increases slower than the others when the interference occurs. As a result, when the noise rate drops back to normal, PeARL has significantly less cumulative error compared to fixed-η estimators.

In Fig. 6(b), ω is smoothly decreased from 1 to 0 over the entire observation period. This scenario shows the performance of estimators when observations transition from being completely chaotic to perfect. As visible in Fig. 6(b), PeARL manages to flatten the error curve and achieve less overshoots in regime changes.

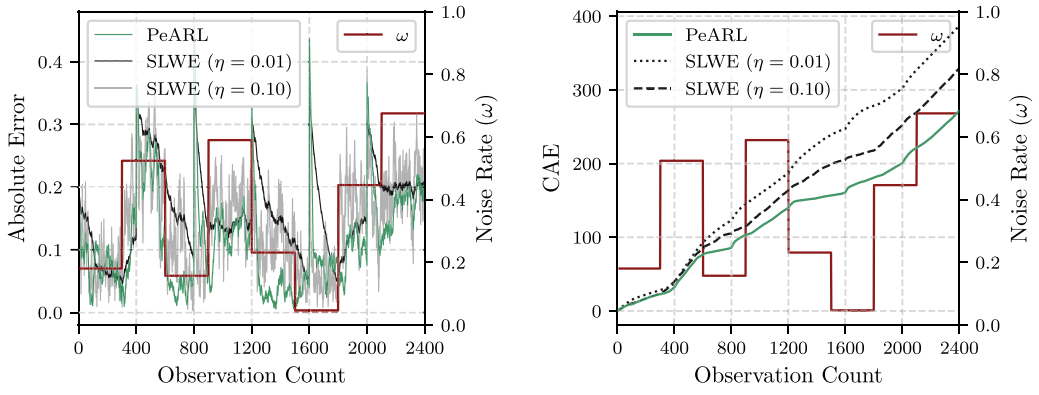
In both cases, PeARL is run with EP2LR mapping since it significantly outperformed AP2LR.

5.2. Experiments with real world data

This set of experiments aim to demonstrate the effect of the proposed learning rate control mechanism on data collected from real-world systems.

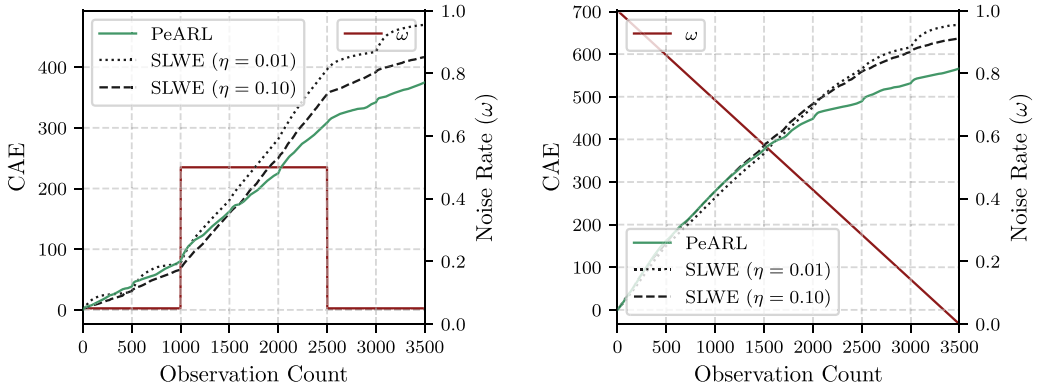
5.2.1. Human activity data

Human Activity Sensing Consortium (HASC) provides `HASC2010corpus` (Kawaguchi et al., 2011), which involves 3-dimensional acceleration data collected from people while they are performing their daily activities like walking, jogging, skipping (happy walking), taking the stairs up or down. The dataset involves both (a) measurements for separate activities and (b) sequences of activities performed by different subjects. The experiments are conducted with the sample data project described in the corpus website (HASC, 2011), which includes 18 activity sequences performed by 7 people.



(a) Absolute error (b) Cumulative absolute error (CAE)

Fig. 5. Performance of algorithms when ω is changing. Vertical dashed lines indicate changes in $\hat{P}$.



(a) Interference (b) Chaos to order

Fig. 6. CAE of estimators in two different scenarios. Vertical dashed lines indicate changes in $\hat{P}$.

Each measurement is a 3-dimensional continuous (limited by the precision of the measurement equipment) vector, forming an $N \times 3$ matrix, where N is the total number of measurements in the sequence. Using a clustering method (breathing k-means (Fritzke, 2020)), the rows of this matrix are clustered, resulting in an N-long discrete sequence of k elements, where k is the number of clusters, which is obtained by optimizing number of clusters using Davies–Bouldin (DB) score (Davies and Bouldin, 1979). This sequence is called the *discrete clean sequence* and the goal of estimators is to estimate the probability distribution of this sequence (computed using MLE) as close as possible by processing observations one by one. In order to measure the effect of noise, a 3-dimensional continuous white noise signal is added to each original measurement with probability ω. Similar to synthetic multinomial experiments, we test two scenarios where ω is fixed and randomly changing during the experiment. Then, the resulting noisy signal is discretized using the clustering information (cluster exemplars) obtained while processing the clean signal. Resulting sequence is called the *discrete noisy sequence*, and is fed to the methods. This setup is depicted in Fig. 7 as a flow diagram.

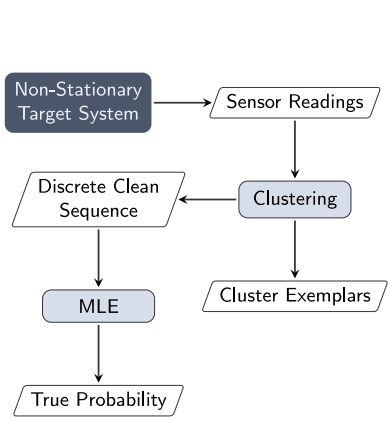
Fig. 7(a) describes how we get the true probability distributions of activity sequences, which are used in evaluation of the estimators. As shown in Fig. 7(b), we measure the performance of estimators via MAE between true probability distribution, which is obtained from the clean sequence; and the estimated probability distribution, which is obtained by processing the noisy sequence. Since the learning methods only process the noisy signals this way, we evaluate how much the methods are affected by the noise. Since activities are multinomially distributed, we use the same set of parameters described in synthetic experiments.

Table 1
Performance (MAE) of estimators on human activity data with changing ω. Each value is the average of 50 runs. In this experiment, $d_p = 1200$, $\Delta \boldsymbol{P}_{\min} = 0.1$, $\Delta \boldsymbol{P}_{\max} = 0.5$.

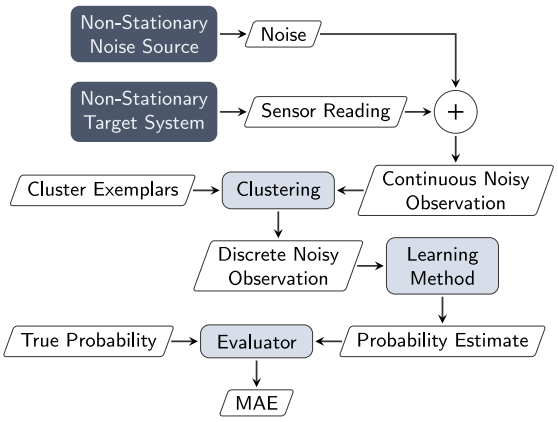
Sequence ID	$\lvert\Sigma\rvert$	SLWE ($\eta = \eta_{\max}$)	SLWE ($\eta = \eta_{\min}$)	PeARL with AP2LR	PeARL with EP2LR
1001	4	0.162	0.129	0.191	**0.116**
1002	5	0.143	0.122	0.150	**0.104**
1003	4	0.150	0.127	0.175	**0.108**
1004	16	0.060	**0.039**	0.072	**0.039**
1005	16	0.061	**0.039**	0.075	0.040
1006	8	0.103	0.079	0.110	**0.072**
1007	7	0.115	0.086	0.125	**0.078**
1008	5	0.146	0.118	0.159	**0.104**
1009	4	0.172	0.137	0.212	**0.134**
1010	4	0.178	0.155	0.168	**0.135**
1011	10	0.083	0.061	0.084	**0.052**
1012	8	0.101	0.075	0.105	**0.066**
1013	4	0.163	0.144	0.166	**0.120**
1014	5	0.139	0.110	0.148	**0.093**
1015	5	0.142	0.111	0.159	**0.100**
1016	4	0.179	0.153	0.175	**0.134**
1017	5	0.141	0.112	0.163	**0.100**
1018	4	0.167	0.140	0.177	**0.121**
Avg.	7	0.134	0.108	0.145	**0.095**

Results on fixed (Fig. 8(a)) and changing (Fig. 8(b)) ω are summarized in Fig. 8. Also, in Table 1, the MAE of estimators on all sequences are given for the changing ω case.

In Fig. 8(a), we observe again a favorable region between $\omega \approx 0.25$ and $\omega \approx 0.75$, indicating that PeARL with EP2LR is able to reduce the effect of noise and achieve lower MAE. AP2LR mapping achieved a

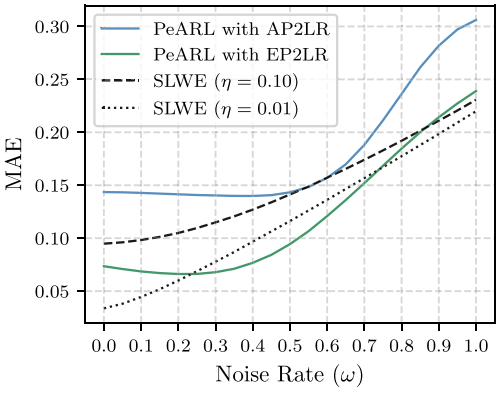


(a) True probability distribution and cluster exemplars are obtained by clustering the continuous sensor readings from the non-stationary system.

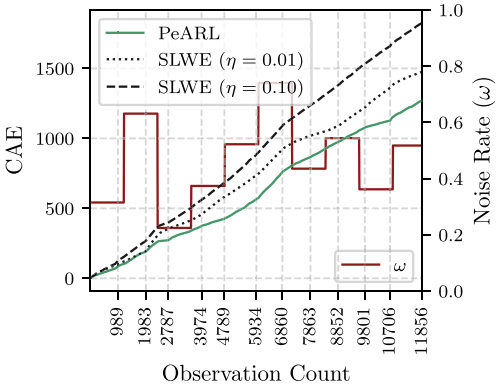


(b) Using the cluster exemplars, noisy observations are clustered and fed to the learning methods. The probability estimates are then compared with the true probability distribution.

Fig. 7. Flow diagrams describing the experiment setup in human activity experiments.



(a) Average (of 20 runs) MAE on 18 activity sequences.



(b) An example CAE result from sequence HASC1001.

Fig. 8. Performance of estimators on human activity data on fixed (a) and changing (b) ω.

similar error curve to the synthetic experiments and is outperformed by EP2LR.

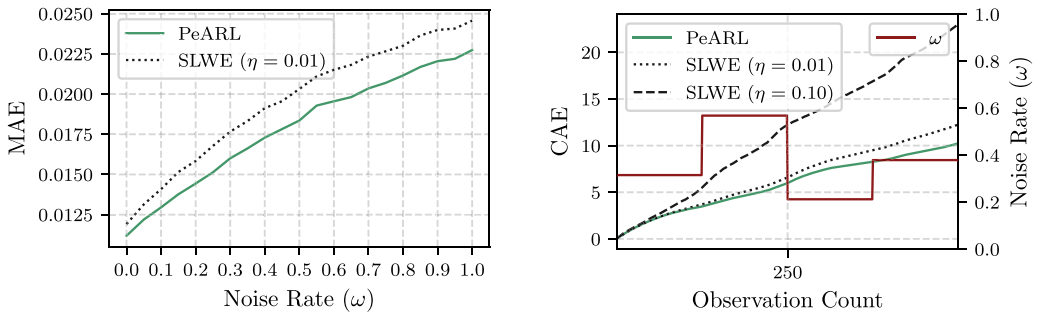
In Fig. 8(b), an example run from the changing ω experiments is shown. Consistent with the synthetic experiments, we observe that the error curve of the proposed method increases more slowly while the noise rate changes, resulting in a significant difference in CAE by the end of the sequence. Also, in Table 1, overall (average of 50 runs) results for each sequence in the dataset are given for the same set of experiments. In all sequences except one (1005), proposed method PeARL with EP2LR mapping performs either better or equally well compared to other estimators. Considering the values of $|\Sigma|$ in sequences 1004 and 1005, this could be linked to activity changes being subtle in terms of $\Delta \boldsymbol{P}$, which explains the performance of exploitative SLWE getting close to that of PeARL.

5.2.2. Electric motor data

End-of-line (EoL) tests are an important step in manufacturing to reduce possible early failures that might occur in the beginning of a product's life-cycle. Electric motors, being one of the major components of many important higher-level systems (e.g., electric vehicles, household appliances), are subjected to a test after production to detect and discard faulty ones. Some common properties to be measured during tests are electrical signals such as current and voltage, and sometimes vibration and sound. This dataset, including current signals of electric motors collected during EoL tests, is provided by Arçelik, a household appliance manufacturer that utilizes the produced electric motors in washing machines, for research within the scope of European Union (EU) funded project intelligent Reliability 4.0 (iRel40). The dataset contains healthy (labeled as `PASS`) and faulty (labeled as `FAIL`) instances of electric motors. Each motor is tested for approximately 7 seconds and classified by a semi-automated system, where human operators are involved with an authority to override the system's decision regarding the final `PASS`/`FAIL` label.

Due to the processes in manufacturing, the behavior of a motor might change during the test, resulting in a `PASS` → `FAIL` or `FAIL` → `PASS` transition. The non-stationary sequence is obtained by combining steady-state signals of motors with alternating labels. This signal is then subjected to *decomposition*, where a moving window is used to extract short time-series of length w_d, forming an $N \times w_d$ matrix whose rows are overlapping signal segments. The goal of this step is to maximize the variety of signal segments that will be used to train the discretization model. This resulting matrix is then fed to a clustering algorithm. Up to this point, the signal is considered clean, and in order to test the effect of noise, continuous white noise is added to each measurement in the original non-stationary signal with probability ω. Then, using the same model trained on the clean signal, the noisy signal is discretized and the noisy discrete sequence is obtained. Similar to previous experiments, performance of estimators are measured with both fixed and changing ω.

Similar to experiments on human activity data, performance of estimators are measured by calculating how close (in terms of MAE) $\hat{\boldsymbol{P}}$, which is obtained from noisy sequence, to $\boldsymbol{P}$ which is obtained from clean sequence. This way, we measure how robust estimators are to increasing amounts of noise in the target signal.



(a) Average (of 20 runs) MAE of 10 motor sequences.

(b) An example CAE result from sequence 01-363.

Fig. 9. Performance of estimators on electric motor data with fixed (a) and changing (b) ω. Exploratory SLWE is not shown in Fig. 9(a) for visibility reasons as it performed the worst. Similarly, AP2LR mapping is omitted in both subfigures as it is overperformed by others.

Table 2
Performance (MAE) of estimators on electric motor dataset with changing ω. Each value is the average of 50 runs with different non-stationary characteristics of noise. In this experiment, $d_p = 1250$, $\Delta \boldsymbol{P}_{\min} = 0.1$, $\Delta \boldsymbol{P}_{\max} = 0.5$ and $|\Sigma|$=8.

Sequence ID	SLWE ($\eta = \eta_{\max}$)	SLWE ($\eta = \eta_{\min}$)	PeARL with AP2LR	PeARL with EP2LR
01-363	0.046	0.022	0.041	**0.019**
02-364	0.047	0.022	0.041	**0.020**
03-365	0.048	0.023	0.039	**0.022**
04-366	0.047	0.022	0.039	**0.020**
05-367	0.048	0.025	0.040	**0.022**
06-370	0.047	0.023	0.039	**0.021**
07-373	0.049	0.026	0.040	**0.023**
08-375	0.048	0.025	0.040	**0.022**
09-376	0.047	0.023	0.042	**0.021**
10-378	0.048	0.025	0.039	**0.022**
11-379	0.048	0.024	0.038	**0.021**
12-380	0.048	0.023	0.040	**0.021**
13-381	0.048	0.025	0.041	**0.023**
14-382	0.049	0.025	0.039	**0.023**
15-383	0.048	0.025	0.039	**0.022**
Avg.	0.048	0.024	0.040	**0.022**

Two instances of SLWE with $\lambda = \{1 - \eta_{\min}, 1 - \eta_{\max}\}$ and PeARL with $\eta_{\min} = 0.01$, $\eta_{\max} = 0.1$, $k = 50$ and $L = 100$ are used with EP2LR and AP2LR mappings. Signals are decomposed with $w_d = 10$ and clustered using breathing k-means (Fritzke, 2020) into 8 clusters. For the stationary noise case, estimators are tested on increasing values of ω from $[0, 1]$ with 0.05 steps, each in 20 runs with different seeds. For the non-stationary noise case, $d_p = 1250$, $\Delta \boldsymbol{P}_{\min} = 0.1$, $\Delta \boldsymbol{P}_{\max} = 0.5$ are used.

Fig. 9(a) shows the MAE of algorithms on fixed ω and Fig. 9(b) shows CAE from an example sequence with changing ω. Also, results from non-stationary noise experiments for each sequence are given in Table 2.

As visible in Fig. 9(a), PeARL achieved less MAE in *all* noise rate levels. This result is different than previous synthetic and real world experiments as we do not observe any turning points. Instead, the favorable region is the entire ω space ($[0, 1]$). The reason for this could be that the sensor measurements already include some amount of noise so that the proposed method managed to perform better even without additional noise.

Fig. 9(b) is consistent with the results shown before. When ω is also changing, CAE of PeARL is more slowly increasing than both exploratory and exploitative estimators. Also, as shown in Table 2, it outperforms other estimators in all sequences.

6. Conclusions and future work

With this study, we introduced an adaptive learning rate approach to improve learning when a non-stationary target system is affected by an also non-stationary noise source. As an example to this approach, we proposed PeARL, which uses persistence metric to estimate the noisiness of observations and change the learning rate accordingly. Persistence turned out to be a simple yet helpful metric for measuring closeness of a discrete sequence to white noise. The computational efficiency, being a local metric that can be used in online learning and being easily mapped to a learning rate space makes it an appealing measure with a potential to be used for optimizing estimation performance in applications where noise is affecting observations in an unknown way. The favorable region observed in experiments involves noise amounts that are possible to occur in real world applications. The real effectiveness of persistence based learning rate control emerges when the non-stationarity in the signal is rather large, i.e., changes in the underlying probability distribution (ΔP) are significant.

PeARL uses parameters $\eta_{\min}$, $\eta_{\max}$, k and L to adjust the learning rate in an online manner. Although empirically finding good values for these was not too difficult, this may not be the case in every application. Therefore, an adaptive method to set these *without* introducing more parameters would be a good extension to this study.

Another limitation of PeARL, and thus the persistence metric, is the assumption of the additive white noise, which is not the most common type of noise in the physical world. However, since we intended to keep the unsupervised nature of the study, we presented the adaptive-η approach with an example that makes the least number of assumptions on the noise model. Following studies may implement the proposed architecture to deal with other types of noise.

The research on learning rate control is likely to proceed with combining other metrics (like the stationarity measure in Coşkun and Tümer (2022) to also deal with exploration/exploitation dilemma) to get a more comprehensive control mechanism. Another possible future work would be to extend persistence definition to continuous signals, which would enable many other possible uses regarding learning rate control.

CRediT authorship contribution statement

Kutalmış Coşkun: Conceptualization, Methodology, Software, Writing – original draft. **Borahan Tümer:** Supervision, Formal analysis, Writing – review & editing.

Declaration of competing interest

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

Data availability

The authors do not have permission to share data.

Acknowledgment

This study is funded under project iRel40. iRel40 is a European co-funded innovation project that has been granted by the ECSEL Joint Undertaking (JU) under grant agreement No. 876659. The funding of the project comes from the Horizon 2020 research program and participating countries. National funding is provided by Germany, including the Free States of Saxony and Thuringia, Austria, Belgium, Finland, France, Italy, the Netherlands, Slovakia, Spain, Sweden, and Turkey.

We would like to thank all MInD research group members, especially Zeynep Kumralbaş and Hazel Çavuş for their precious comments that certainly improved the quality of this study.

References

Aslancı, E., Coşkun, K., Schüller, P., Tümer, B., 2017. Detection of regime switching points in non-stationary sequences using stochastic learning based weak estimation method. In: 2017 IEEE 15th International Conference on Industrial Informatics. INDIN, pp. 787–792. http://dx.doi.org/10.1109/INDIN.2017.8104873.

Cesa-Bianchi, N., Shalev-Shwartz, S., Shamir, O., 2011. Online learning of noisy data. IEEE Trans. Inform. Theory 57 (12), 7907–7931.

Chung, K.L., 1967. Markov Chains. Springer-Verlag, New York.

Coşkun, K., Tümer, B., 2022. An adaptive estimation method with exploration and exploitation modes for non-stationary environments. Pattern Recognit. 129, 108702.

Crammer, K., Kearns, M., Wortman, J., 2005. Learning from data of variable quality. In: Weiss, Y., Schölkopf, B., Platt, J. (Eds.), Advances in Neural Information Processing Systems, Vol. 18. MIT Press, URL: https://proceedings.neurips.cc/paper/2005/file/465636eb4a7ff4b267f3b765d07a02da-Paper.pdf.

Dave, R.N., 1991. Characterization and detection of noise in clustering. Pattern Recognit. Lett. 12 (11), 657–664.

Davies, D.L., Bouldin, D.W., 1979. A cluster separation measure. IEEE Trans. Pattern Anal. Mach. Intell. PAMI-1 (2), 224–227.

Ditzler, G., Roveri, M., Alippi, C., Polikar, R., 2015. Learning in nonstationary environments: A survey. IEEE Comput. Intell. Mag. 10 (4), 12–25.

Fazelpour, A., Khoshgoftaar, T.M., Dittman, D.J., Shanab, A.A., 2015. Observing the effect of the choice of classifier on bioinformatics data with varying levels of data quality and class balance. In: 2015 IEEE International Conference on Information Reuse and Integration. IEEE, San Francisco, CA, USA, pp. 372–379. http://dx.doi.org/10.1109/IRI.2015.63.

Fefilatyev, S., Shreve, M., Kramer, K., Hall, L., Goldgof, D., Kasturi, R., Daly, K., Remsen, A., Bunke, H., 2012. Label-noise reduction with support vector machines. In: Proceedings of the 21st International Conference on Pattern Recognition. ICPR2012, IEEE, pp. 3504–3508.

Fox, R., Pakman, A., Tishby, N., 2017. Taming the noise in reinforcement learning via soft updates. arXiv:1512.08562[Cs, Math].

Fritzke, B., 2020. Breathing K-means. http://dx.doi.org/10.48550/ARXIV.2006.15666, URL: https://arxiv.org/abs/2006.15666.

Gama, J., Ž.liobaitė, I., Bifet, A., Pechenizkiy, M., Bouchachia, A., 2014. A survey on concept drift adaptation. ACM Comput. Surv. 46 (4), 1–37.

Gupta, S., Gupta, A., 2019. Dealing with noise problem in machine learning data-sets: A systematic review. Procedia Comput. Sci. 161, 466–474.

Hammer, H.L., Yazidi, A., 2018. Parameter estimation in abruptly changing dynamic environments using stochastic learning weak estimator. Appl. Intell. 48 (11), 4096–4112.

Hao, S., Li, P., Wu, R., Chu, X., 2022. A model-agnostic approach for learning with noisy labels of arbitrary distributions. In: 2022 IEEE 38th International Conference on Data Engineering. ICDE, IEEE, Kuala Lumpur, Malaysia, pp. 1219–1231. http://dx.doi.org/10.1109/ICDE53745.2022.00096.

HASC, 2011. HASC2010corpus. Online. URL: http://hasc.jp/hc2010/HASC2010corpus/hasc2010corpus-en.html. (Accessed 1 November 2021).

Kang, Z., Pan, H., Hoi, S.C.H., Xu, Z., 2020. Robust graph learning from noisy data. IEEE Trans. Cybern. 50 (5), 1833–1843.

Kawaguchi, N., Yang, Y., Yang, T., Ogawa, N., Iwasaki, Y., Kaji, K., Terada, T., Murao, K., Inoue, S., Kawahara, Y., et al., 2011. HASC2011corpus: towards the common ground of human activity recognition. In: Proceedings of the 13th International Conference on Ubiquitous Computing. pp. 571–572.

Kimura, T., Shigeta, T., 2013. Inhomogeneous noise correction combined with uniform filter and sensitivity map (INCUS) for multi-coil imaging including parallel imaging. Magn. Reson. Med. Sci. 12 (1), 21–30.

Kläs, M., Vollmer, A.M., 2018. Uncertainty in machine learning applications: A practice-driven classification of uncertainty. In: Hoshi, M., Seki, S. (Eds.), Developments in Language Theory, vol. 11088. Springer International Publishing, Cham, pp. 431–438. http://dx.doi.org/10.1007/978-3-319-99229-7_36.

Kubica, J., Moore, A., 2003. Probabilistic noise identification and data cleaning. In: Third IEEE International Conference on Data Mining. IEEE Comput. Soc., Melbourne, FL, USA, pp. 131–138. http://dx.doi.org/10.1109/ICDM.2003.1250912.

Kumar, M., Rossbory, M., Moser, B.A., Freudenthaler, B., 2019. Deriving an optimal noise adding mechanism for privacy-preserving machine learning. In: Anderst-Kotsis, G., Tjoa, A.M., Khalil, I., Elloumi, M., Mashkoor, A., Sametinger, J., Larrucea, X., Fensel, A., Martinez-Gil, J., Moser, B., Seifert, C., Stein, B., Granitzer, M. (Eds.), Database and Expert Systems Applications. Springer International Publishing, Cham, pp. 108–118.

Liebchen, G., Twala, B., Shepperd, M., Cartwright, M., Stephens, M., 2007. Filtering, robust filtering, polishing: techniques for addressing quality in software data. In: First International Symposium on Empirical Software Engineering and Measurement. ESEM 2007, IEEE, Madrid, Spain, pp. 99–106. http://dx.doi.org/10.1109/ESEM.2007.70.

Lu, J., Liu, A., Dong, F., Gu, F., Gama, J., Zhang, G., 2018. Learning under concept drift: A review. IEEE Trans. Knowl. Data Eng. 1.

Mireshghallah, F., Taram, M., Ramrakhyani, P., Jalali, A., Tullsen, D., Esmaeilzadeh, H., 2020. Shredder: learning noise distributions to protect inference privacy. In: Proceedings of the Twenty-Fifth International Conference on Architectural Support for Programming Languages and Operating Systems. ACM, Lausanne Switzerland, pp. 3–18. http://dx.doi.org/10.1145/3373376.3378522.

Nettleton, D.F., Orriols-Puig, A., Fornells, A., 2010. A study of the effect of different types of noise on the precision of supervised learning techniques. Artif. Intell. Rev. 33 (4), 275–306.

Oommen, B.J., Rueda, L., 2006. Stochastic learning-based weak estimation of multinomial random variables and its applications to pattern recognition in non-stationary environments. Pattern Recognit. 39 (3), 328–341.

Sasaki, M., Kuribayashi, T., Ito, S., Inoue, Y., 2011. Active random noise control using adaptive learning rate neural networks with an immune feedback law. Int. J. Appl. Electromagn. Mech. 36 (1–2), 29–39.

Song, S., Chaudhuri, K., Sarwate, A., 2015. Learning from data with heterogeneous noise using SGD. In: Lebanon, G., Vishwanathan, S.V.N. (Eds.), Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics. In: Proceedings of Machine Learning Research, vol. 38, PMLR, San Diego, California, USA, pp. 894–902, URL: https://proceedings.mlr.press/v38/song15.html.

Song, H., Kim, M., Park, D., Shin, Y., Lee, J.-G., 2022. Learning from noisy labels with deep neural networks: A survey. IEEE Trans. Neural Netw. Learn. Syst. 1–19.

Tavasoli, H., Oommen, B.J., Yazidi, A., 2019. On utilizing weak estimators to achieve the online classification of data streams. Eng. Appl. Artif. Intell. 86, 11–31.

Twala, B., 2013. Impact of noise on credit risk prediction: Does data quality really matter? Intell. Data Anal. 17 (6), 1115–1134.

Van Hulle, M., 1995. Learning rate adaptation achieved in unsupervised competitive learning: An application to noise cancelling. In: Proceedings of ICNN'95 - International Conference on Neural Networks. 2, pp. 860–864 vol.2. http://dx.doi.org/10.1109/ICNN.1995.487531.

Vaseghi, S.V., 2008. Advanced digital signal processing and noise reduction. John Wiley & Sons.

Wang, B., Hegde, N., 2019. Privacy-preserving Q-learning with functional noise in continuous spaces. In: Wallach, H., Larochelle, H., Beygelzimer, A., d' Alché-Buc, F., Fox, E., Garnett, R. (Eds.), Advances in Neural Information Processing Systems, Vol. 32. Curran Associates, Inc., URL: https://proceedings.neurips.cc/paper/2019/file/6646b06b90bd13dabc11ddba01270d23-Paper.pdf.

Widmer, G., Kubat, M., 1996. Learning in the presence of concept drift and hidden contexts. Mach. Learn. 23 (1), 69–101.

Xu, X., Chen, M., 2022. Discovery of subdiffusion problem with noisy data via deep learning. J. Sci. Comput. 92 (1), 23.

Yan Zhang, Xingquan Zhu, Xindong Wu, Bond, J., 2005. ACE: An aggressive classifier ensemble with error detection, correction and cleansing. In: 17th IEEE International Conference on Tools with Artificial Intelligence. ICTAI'05, IEEE, Hong Kong, China, http://dx.doi.org/10.1109/ICTAI.2005.23, 8 pp.–317.

Zhang, X.-D., Wang, X., Chang, D., Chang, L., Zhang, D., 2023. Modern Signal Processing. DE GRUYTER, Boston.

Kutalmış Coşkun received his B.Sc. and M.Sc. degrees from Computer Science & Engineering department at the Faculty of Engineering, Marmara University. He is currently a researcher at MInD research group at the same department. His research interests include syntactic pattern recognition, reinforcement learning, stochastic processes, random walks, Markov models and learning in non-stationary environments.

Borahan Tümer received his B.Sc. and M.S. degrees both in Computer Engineering from Boğaziçi University (İstanbul, Turkey) in 1987, and from Istanbul Technical University in 1990, respectively and Ph.D. degree in Electrical and Computer Engineering from Marquette University (Milwaukee, WI) in 1998. Prof. Tümer currently works at the Computer Engineering Department at Marmara University and leads MInD research group and lab. His current research interests are in learning systems, syntactic pattern recognition, reinforcement learning and sequential data analysis.