Pattern Recognition 129 (2022) 108702

Contents lists available at ScienceDirect

Pattern Recognition

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

An adaptive estimation method with exploration and exploitation modes for non-stationary environments

Kutalmış Coşkun*, Borahan Tümer

Faculty of Engineering, Marmara University, İstanbul, Turkey

A R T I C L E   I N F O

Article history:
Received 23 December 2020
Revised 3 April 2022
Accepted 7 April 2022
Available online 12 April 2022

Keywords:
Stochastic learning
Concept drift
Change detection
Parameter estimation
Dynamic learning rate

A B S T R A C T

Dynamic systems are highly complex and hard to deal with due to their subject- and time-varying nature. The fact that most of the real world systems/events are of dynamic character makes modeling and analysis of such systems inevitable and charmingly useful. One promising estimation method that is capable of *unlearning* past information to deal with non-stationarity is Stochastic Learning Weak Estimator (SLWE) by Oommen and Rueda (2006). However, due to using a constant learning rate, it faces a trade-off between plasticity and stability. In this paper, we model SLWE as a random walk and provide rigorous theoretical analysis of asymptotic behavior of estimates to obtain a statistical model. Utilizing this model, we detect changes in stationarity to switch between exploratory and exploitative learning modes. Experimental evaluations on both synthetic and real world data show that the proposed method outperforms related algorithms in different types of drifts.

1. Introduction

Optimizing decision-making under uncertainty has always been an important task throughout human history. In order to achieve this, it is inevitable to study and understand how complex systems work, which often involves concepts like behavior modeling and estimation, recognizing patterns, detecting and adapting to changes, predicting future behavior, comparison of multiple systems and running simulations. A complex system is often said to be non-stationary, meaning that its behavior changes with time, generally in an unknown way. Also, depending on the characteristics or complexity of the target system (i.e., system under consideration), it might be hard, infeasible or even impossible to obtain ground truths that can be used to evaluate how well our understanding is. Considering the prevalence of mentioned systems in modern era, these properties make the analysis of such systems a beneficial but challenging task.

Behavior of a complex system is often represented with statistical models that are built considering the data collected from the system. This approach assumes that there is a true probability distribution that characterizes a specific behavior, hence determines the generated data. An important problem is to estimate the parameters of the underlying statistical model only from the observations of the target system. This, namely parameter estimation, is an important task in statistical Pattern Recognition (PR) [1,2], in both (a) training phase of a PR system, where mathematical models are being learned to represent various system behaviors, and (b) recognition/diagnosis phase, where the test inputs are processed to be classified into one of several categories.

In this respect, we would like to emphasize that in most cases, this is an *online learning* task, where the learner obtains and processes observations one-by-one, without having the whole sequence available from the beginning. The non-stationarity of the target system makes it necessary for the learning algorithm to be able to adapt to the changes that might occur at unknown time periods and keep the estimate close to the true value accordingly. Such algorithms generally employ a method to disregard obsolete information, either by actively tracking changes or maintaining an always up-to-date estimate.

Furthermore, the non-stationarity also introduces a fundamental trade-off that is studied in many branches of science, namely the *exploration* versus *exploitation* dilemma. Learners dealing with data from non-stationary systems often try to strike a balance between being adaptive and being stable. Being too exploratory results in greater adaptation capability to changes, with the drawback of instability of estimates and the risk of forgetting important information too quickly. Conversely, prioritizing exploitation too much makes estimates more stable, which however significantly reduces how well the estimates track the true value of the target parameter in cases of changes. A common approach is to prefer exploration until a sufficient amount of learning is achieved,

* 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.patcog.2022.108702
0031-3203/

then focus on exploitation. This priority shift is repeated as long as the non-stationarity is present.

SLWE, introduced by Oommen and Rueda in 2006 [3], is an efficient estimation method that updates the value of the estimate at each time instant based on the current observation. The update rule is adapted from linear reward-inaction (L_{R-I}) class of reinforcement schemes in Learning Automata (LA) context [4] and the convergence attained is said to be *weak*, which means rather than the estimate itself, statistical moments of the estimate converge asymptotically.

SLWE employs a learning coefficient parameter, λ, which is fixed throughout learning and controls the amplitude of updates to the estimate. Hence, it determines how much the recent observation affects the running estimate, making the method more adaptive or more stable. In this regard, determination of λ is connected to the exploration–exploitation trade-off mentioned before, which is the main motivation of this study.

With this study, we model SLWE as a random walk and analyze the evolution of estimates during learning. We provide a thorough analysis of position, displacement and expected jump sizes of SLWE random walk. Further, we study the evolution of positional probability density and show that the estimation mechanism of SLWE is in fact consistent with *diffusion*, a concept that is used to explain various phenomena in nature. We show that the drift velocity of estimates diminishes over time, which is consistent with the weak convergence property of SLWE. Also, we prove that the diffusion coefficient is asymptotically constant, which significantly simplifies the analysis. As a result, we obtain a statistical model for the asymptotic behavior of estimates. By estimating the drift velocity and utilizing the derived model, with an approach adapted from Time Series Analysis (TSA), we treat estimates as a time series and decompose it into different stationarity segments, which are (i) the trend-stationary phase, which is the result of non-zero drift velocity and (ii) the stationary phase, which is reached after the trend is diminished.

We provide algorithms for detecting (i) non-stationary to stationary changes, which indicate that a saturation point in learning is reached and it is preferable to prioritize exploitation and (ii) stationary to non-stationary changes, which indicate that a change has occurred and exploration should be the focus thereafter.

In this regard, we propose Stochastic Learning Weak Estimator with Adaptive Learning Mode (SLWE-ALM), which is capable of efficiently ($O(1)$ for each observation) switching between exploratory and exploitative *learning modes* with different learning coefficients, based on the stationarity of estimates. We experimentally test the proposed method on (i) synthetically generated binomial and multinomial distributions with different types of concept drift and (ii) real world data that are collected from (a) accelerometers attached to humans while performing daily activities and (b) customer reviews on products with different domain and sentiment types. We show that the proposed method successfully performs switches between mentioned learning modes and tracks the target parameter better compared to similar methods in literature, on different types of changes.

The paper is organized as follows: In Section 3, some preliminary concepts are introduced. Related work is discussed in Section 2. In Section 4, random walk modeling of SLWE and adaptive learning modes are described. In Section 5, the provided analysis is verified with simulations, proposed method is tested on synthetic and real world data and the results are discussed. In Section 6, the study is finalized with conclusions and future work.

2. Related work

Learning in non-stationary environments is often expressed with the term *concept drift*, where the target variable (or distribution) that is to be learned changes over time in an unknown way. The term is coined by Schlimmer and Granger in 1986 [5]. Due to being fundamental yet complicated with a plethora of real life applications, this problem attracted many researchers throughout the years. Especially in recent times, with the increased availability of large amounts of data, it has been rather uncommon for Machine Learning (ML) and PR applications to work on concepts that remain stationary over extended periods of time. An emerging challenge is to learn/detect/classify patterns in streaming (i.e., requiring real-time processing) and evolving (i.e., involving concept drift) data, which necessitates capability of noticing and reacting to changes, and various approaches have been proposed tackling different fundamental aspects of the problem [6–9]. Applications where concept drift is relevant are abundant, including intrusion detection [10], fraud detection and prevention [11], malware classification, [12], spam filtering [13] and financial prediction [14].

A common way to handle concept drift is to consider only the data that is relevant to the ongoing concept, which is generally referred to as *instance selection*. A widely used method for selecting representative data is to use a *sliding window* over the obtained instances and utilize the local information for near future predictions [15]. This way, temporal nature of the data is taken into account by discarding obsolete information that is not related to the current context. The size of the window may be fixed or adjusted adaptively according to the span of current drift [16,17], which is an attempt to overcome the problem of tuning window size. There are also techniques referred to as *instance weighting*, where observations have a weighted effect on the estimation depending on their properties. One prominent approach is using recency information to assign weights to instances, which is also referred to as gradual forgetting [18]. There are variations of this idea which utilize linear [18] or exponential [16] functions for forgetting irrelevant information.

Learners that are capable of handling concept drift can further be categorized into two main groups [19], which are (a) trigger based methods which utilize a signal that specifies when to change or update the current model and (b) evolving methods which try to maintain model accuracy at all times. Trigger based methods often employ a change detection technique to decide whether to start learning a new model or, if the method is capable of identifying and saving different concept descriptors, to switch to another pre-learned one. A widely used approach is to compare previous and recent portions of data by using statistical tests that indicate changes in the underlying distribution. Change detection can be done directly on the observed data [17], model parameters [20] or if the problem is supervised, which unfortunately is not always the case, using error metrics [5]. Evolving methods, on the other hand, do not try to detect changes since whether the concept has drifted or not, they always tweak the model to fit to the most recent situation. One particular example of such learners is SLWE [3], which is adapted from the LA domain by employing the L_{R-I} update scheme. This method is said to be a weak estimation method, since the estimates do not converge with probability of 1. Instead, statistical moments of estimates are shown to converge asymptotically. Such learners are capable of unlearning obsolete information and maintain a running estimate in a very space and time efficient way. Weak estimators are widely used for various applications that require dealing with non-stationary systems, which include anomaly detection [21], compression [22], online data stream classification [23] and hand gesture classification [24].

Considering the learning mode switching mechanism we propose in this study, our method can be categorized as both trigger based, as it actively tracks for changes in stationarity of estimates, and evolving since the estimate is always up-to-date even there is no change detected. We utilize the change information to control a parameter (λ) of the evolving method (SLWE).

A famous dilemma that almost all learners face regarding non-stationary environments is described in various ways in the literature. One of them is *exploration* versus *exploitation* in Adaptive Control and Reinforcement Learning (RL) [25] contexts. The same dilemma is also referred to as *stability-plasticity* [26]. In essence, it is the trade-off between integration of novel information versus taking advantage of information that worked so far. There are numerous studies and approaches tackling this problem in many branches of science. Considering learning in non-stationary environments, some common approaches are [27], (a) to prioritize exploration initially and then decrease its priority gradually as the learning proceeds based on a decreasing function of time, (b) to explore as long as the average performance is less than the estimated maximum performance, (c) to explore as long as the average prediction error, which is a measure for how well the environment is known, is higher than a predetermined level, (d) to explore considering the rate of change of average reward or prediction error. Utilization of a descending function of time as described in (i) requires the determination of decrease rate and maximum/minimum values, which generally depends on the problem and does not work well on slowly changing parameters. Also, it lacks a mechanism to re-prioritize exploration when needed. Approaches given in (ii) to (iv) are more adaptive, however they require an estimation of current performance, a measurement of model coverage or a supervising signal, which are sometimes infeasible or impossible to obtain. The method we propose does not need a performance indicator to decide whether to explore or exploit, instead it utilizes stationarity of estimates to switch between learning modes when necessary.

A particularly closely related recent study is [28], where the parameter estimation under abruptly changing environments is tackled by combining a fast (SLWE with a relatively low, fixed λ) and a slow (sample mean estimator) learner. The slow learner progressively decreases variance and is adjusted by performing a *jump* if there is enough evidence of deviation from a normal distribution (indicating an abrupt change), which is discovered empirically through stochastic simulation. The use of sample mean estimator as the slow learner makes the method perform well especially on dynamic environments with longer stationary segments with few abrupt changes. Our method, on the other hand, utilizes a single learner that, through changing a parameter considering stationarity of estimates, can be slow or fast, or as we prefer to describe as exploratory or exploitative. A particular difference between our study and [28] is that, we do not use a time-based approach and actively track changes also from non-stationarity to stationarity to determine when to switch. Also, since exploratory and exploitative modes of our method are separately adjustable for changes with different severity characteristics, we argue that SLWE-ALM is more flexible to be used with multiple types of drift and robust to changes that do not trigger the mode switch. We consider cases where the change is not detected not as false positives, but cases where exploratory mode is not necessary and the change can be dealt with in exploitative mode.

3. Preliminaries

In this section, some basic concepts used in the rest of this study are introduced.

3.1. Stochastic Learning Weak Estimator

Estimation of parameters that characterize a random variable from the observations of that random variable is an important task and has been studied for many years. In this respect, SLWE is an estimation method proposed by Oommen and Rueda in 2006 [3], in which the estimate converges *weakly*, that is, with regard to the first and second statistical moments. This property enables SLWE to outperform traditional, *strong* (i.e., estimate converges with probability 1) methods like Maximum Likelihood Estimate (MLE) and Bayesian estimates when the parameter being estimated changes with time. In such non-stationary environments, being able to quickly "unlearn" what is learned before becomes an important attribute to adapt to the new environment. SLWE achieves this by maintaining a running estimate of the target parameter by an update rule.

The estimation method is, in principle, similar to the L_{R-I} learning scheme of a learning automaton. After each observation of the random variable, the estimate is updated based on its current value and the observation, using a learning coefficient λ, $0 < \lambda < 1$. It is shown in [3] that, (a) the convergence is independent of λ, (b) the rate of convergence is determined by an explicit function of λ, and (c) variance of the estimate depends on λ. These properties make λ an important parameter to be studied.

3.1.1. Weak estimation of parameters of a binomial distribution

In the simplest case, parameters of a *binomial distribution* are estimated. A binomial distribution is the probability distribution that reflects the outcomes of n independent experiments, where each experiment is called a *Bernoulli trial* and might result in an outcome that is an element of the set of $\Omega = \{s_1, s_2\}$ of two outcomes. Generally, these two outcomes represent success/failure, true/false or yes/no cases. Thus, the parameters that characterize a binomial distribution are (a) the number of Bernoulli trials n, (b) the probability vector $p = \{p_{s_1}, p_{s_2}\}$ where p_{s_i} represents the probability of observing outcome s_i. Since the two outcomes s_1 and s_2 are mutually exclusive, the sum of probabilities of observing each outcome adds up to 1, that is $p_{s_1} + p_{s_2} = 1$.

Now, let X be a binomially distributed random variable and X_n be the realization of X at time n that can take values from the set $E = \{1, 2\}$, where each value corresponds to an outcome from the set Ω of outcomes. The problem is defined as to estimate parameters of the binomial distribution from the sequence $\{X_1, X_2, \ldots, X_n\}$. Concerning this, the number of trials is assumed to be the number of observations, reducing the goal to estimate the parameter p, which is done by maintaining an estimate $\hat{p}(n)$ that is obtained at time n by the update rule given in Eq. (1).

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

This update rule similar to L_{R-I} scheme and it adjusts the estimate based on the most recent observation. The amplitude of the update, $|\hat{p}_{s_i}(n+1) - \hat{p}_{s_i}(n)|$, depends on the learning coefficient λ.

3.1.2. Weak estimation of parameters of a multinomial distribution

A multinomial distribution is the probability distribution that reflects the outcomes of a *multinomial experiment*, which is a statistical experiment that consists of n independent trials where the outcome of each trial is an element of the set $\Omega = \{s_1, s_2, \ldots, s_k\}$ of k discrete, mutually exclusive outcomes. Thus, a multinomial distribution is characterized by the parameters n and the probability vector $P = [p_{s_1}, p_{s_2}, \ldots, p_{s_k}]^T$ where each p_{s_i}, $i = 1, \ldots, k$, is the probability of observing outcome s_i. Since each outcome is mutually exclusive, the sum of probabilities add up to 1, $\sum_{i=1}^{k} p_{s_i} = 1$.

Given a multinomially distributed random variable X, which takes values from the set $E = \{1, 2, \ldots, k\}$ where each value corresponds to an outcome from the set S described above, the realization of X at a given time n is denoted by $X_n \in E$. Considering the same problem of estimating the probability vector P from the sequence of observations of X, in this case, SLWE maintains a running estimate vector $\hat{P}(n) = [\hat{p}_{s_1}(n), \ldots, \hat{p}_{s_k}(n)]^T$ which refers to the estimate of P, where $\hat{p}_{s_i}(n)$ is the estimate of p_{s_i} at time n.

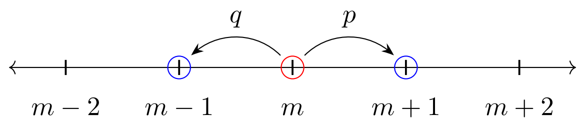


Fig. 1. A simple random walker on one-dimensional space of integers. The walker starts from $m \in \mathbb{Z}$ and jumps to the right or left with probabilities p and $q = 1 - p$, respectively.

This running estimate vector is updated after each observation by the rule given in Eq. (2).

$$\hat{p}_{s_i}(n+1) \leftarrow \begin{cases} \hat{p}_{s_i}(n) + (1-\lambda)\sum_{j \neq i} \hat{p}_{s_j}(n), & \text{if } X_n = i \\ \lambda \hat{p}_{s_i}(n), & \text{if } X_n \neq i \end{cases} \quad (2)$$

Similarly, this update rule penalizes (since $0 < \lambda < 1$) the running estimates except the one that corresponds to the most recent observation. The penalized amount is then added to the probability estimate of currently observed outcome. Since the penalized and added values are equal, this operation always obeys the second axiom of probability, that is $\sum_i^k \hat{p}_{s_i}(n) = 1$ at all times.

3.1.3. Non-stationary distributions

The main advantage of SLWE shows up when the distribution is non-stationary. A non-stationary distribution is characterized by a parameter (or a parameter vector) that changes with time. These changes can be (i) gradual where the transition from one value to another happens over some time or (ii) abrupt where the parameter "switches" from one value to another suddenly. For abrupt changes, distribution is assumed to be stationary between change points. These stationary segments are called *regimes* which enables us to describe a non-stationary distribution as a linear combination of stationary regimes.

It should be noted that, each regime can be arbitrarily long, that is, changes occur in random times. However, very frequent changes would result in a chaotic distribution where learning is not possible. Hence, we assume each regime continues long enough for the parameter to be learned.

Considering the general definition of a random variable as a function that maps the set of possible outcomes Ω to a measurable space E, that is$X : \Omega \to E$, the sequence of observations $S = \{X_1, X_2, \ldots, X_k\}$ also involves the same non-stationary structure defined earlier.

In problems that involve non-stationarity, a learner is expected to adapt to the new value of the parameter as soon as possible while maintaining accuracy throughout learning. SLWE is shown to outperform sliding window based techniques in non-stationary distributions that involve sudden changes [3].

3.2. Random walks

A random walk is generally defined as a stochastic process which specifies a trajectory that consists of successive random steps. In order to describe a random walk, it is necessary to specify (i) a state space (i.e., positions that the random walker can be at), (ii) a set of valid jumps, (iii) a probability distribution that indicates probabilities of each jump, (iv) an initial state probability distribution of the walker. A particularly simple form of random walk would be defined on the set of integers, $\mathbb{Z}$. Initially, the random walker is placed at position $s_0 = m$, where s_t indicates the position of the walker at time t and $m \in \mathbb{Z}$. After fixed time intervals Δt, the walker either jumps to the right (to the least integer greater than s_t) with probability p or to the left (to the greatest integer less than s_t) with probability $q = 1 - p$. A representation of this simple random walk is given in Fig. 1.

Random walks have been studied extensively are shown to be useful mathematical objects for modeling and quantifying behavior of stochastic activities [29].

4. Methodology

In this section, the proposed method is described in multiple subsections. The learning mode switching method we propose in this paper utilizes the asymptotic behavior of estimates obtained from SLWE. We study the evolution of estimates in dynamic environments by modeling SLWE as a random walk in Section 4.1. Starting from the simple walk on one-dimensional space of integers introduced earlier, we relax two assumptions considering state space and jump sizes to model SLWE. In Section 4.1.1, position and displacement concepts are defined according to the update rule of SLWE. The asymptotic analysis of expected position and displacement is studied through multiple lemmas. Additionally, the relation between position and displacement is explained in a form of feedback loop which is considered to be the reason behind convergence. In Section 4.1.2, the evolution of positional probability density is studied. With the help of analysis techniques often used in statistical mechanics, the behavior of probability density as the number of jumps grows large is examined and a model for asymptotic distribution is obtained.

In Section 4.2, the proposed adaptive switching mechanism is explained. Utilizing the derived model in Section 4.1, Section 4.2.1 discusses how learning progress is noticed by tracking stationarity of estimates. Complexity of the proposed method is discussed in Section 4.3.

4.1. Stochastic learning weak estimator as a random walk

The consecutive update approach of SLWE on estimates makes it possible to model and study it as a random walk on a one-dimensional space. It is plausible to consider that the estimate obtained at time n using SLWE update rule corresponds to the position of a random walker at time n. Furthermore, the probabilities of applying penalty or rewarding in the update rule for the binomial case given in Eq. (1), correspond to probabilities of the random walker jumping to a position that is on the left or right, respectively. In this study, such a random walk is referred to as an "SLWE random walk."

In spite of the correspondence to the simple random walk explained in Section 3.2, modeling SLWE as a random walk requires relaxing some restrictive assumptions introduced earlier. The first assumption that needs to be relaxed is that the state space of the simple random walk is discrete, that is to say, the consecutive positions that the walker can be at are a fixed distance apart (i.e., $\Delta x = const.$). This is the reason why positions along the line are labeled using integers in Fig. 1. However, this is not the case in SLWE random walk because of the fact that the position of the walker is analogous to the probability estimate obtained by SLWE. Thus, SLWE random walker can be at any real-valued position between 0 and 1.

Moreover, the assumption of step sizes of all jumps to either of the two directions being fixed and equal (i.e., $|\Delta x| = const.$) also needs to be relaxed. Due to the fact that in SLWE, each update makes use of the most recent estimate to calculate the new estimate; the location reached by the SLWE random walker after a jump depends on its previous location and the learning coefficient, λ. This characteristic makes jumps initiated from different locations along the one-dimensional space result in different displacements of the walker.

The assumption of time intervals between jumps (τ) being fixed and equal does not need to be relaxed to model SLWE since each estimate is also obtained with fixed time intervals. For the sake of

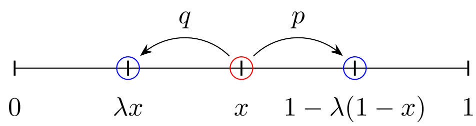


Fig. 2. SLWE is modeled as a random walk on a 1-dimensional continuous interval $[0, 1]$. The walker starts from position x, $0 < x < 1$, and jumps to position $1 - \lambda(1 - x)$ with probability p and to position λx with probability $q = 1 - p$.

simplicity, we omit τ in our notation by setting $\tau = 1$. Considering these, modeling of SLWE as a random walker is given in Fig. 2.

4.1.1. Position and displacement

Let us start the analysis of SLWE random walk by introducing some concepts that will be useful for the following formulations. The position of the walker after n jumps is denoted by x_n and from the update rule of SLWE for the binomial case given in Eq. (1), it can be written as:

$$x_n \leftarrow \begin{cases} 1 - \lambda(1 - x_{n-1}), & \text{with probability } p \\ \lambda x_{n-1}, & \text{with probability } q = 1 - p \end{cases} \tag{3}$$

The walker can not jump to a position that is beyond the limits of the one-dimensional space. For instance, if $x_{n-1} = 1$ and the walker jumps to the right, it reaches to $x_n = 1 - \lambda(1 - 1) = 1$, remaining at the position where the jump is initiated. The same is valid for the case where $x_{n-1} = 0$ and the walker jumps to the left.

An important characteristic of SLWE random walk is that the expected position of the walker converges after a large number of jumps. It turns out that the converged value is the probability of jumping to the right, p. This is shown in Lemma 1 by a utilizing a technique that is generally used in LA context. A similar proof is given in [3].

Lemma 1. *Let x_n be the position of SLWE random walker after n jumps and p be the probability that the walker jumps to the right. Then, $\mathbb{E}[x_\infty] = p$.*

Proof. Let us write the conditional expected value of the position after $n+1$ jumps in terms of x_n by using Eq. (3).

$$\begin{aligned} \mathbb{E}[\,x_{n+1}\,|\,x_n] &= p[1 - \lambda(1 - x_n)] + q[\lambda x_n] \\ &= p(1 - \lambda + \lambda x_n) + (1 - p)(\lambda x_n) \\ &= p - p\lambda + p\lambda x_n + \lambda x_n - p\lambda x_n \\ &= p(1 - \lambda) + \lambda x_n \end{aligned} \tag{4}$$

Taking the expectation second time, by the law of total expectation we obtain:

$$\mathbb{E}[x_{n+1}] = p(1 - \lambda) + \lambda\mathbb{E}[x_n] \tag{5}$$

As $n \to \infty$, both $\mathbb{E}[x_{n+1}]$ and $\mathbb{E}[x_n]$ converge to $\mathbb{E}[x_\infty]$, therefore we get:

$$\begin{aligned} \mathbb{E}[x_\infty] &= p(1 - \lambda) + \lambda\mathbb{E}[x_\infty] \\ \mathbb{E}[x_\infty] - \lambda\mathbb{E}[x_\infty] &= p(1 - \lambda) \\ \mathbb{E}[x_\infty](1 - \lambda) &= p(1 - \lambda) \\ \mathbb{E}[x_\infty] &= p \end{aligned} \tag{6}$$

which completes the proof. □

Another important concept that is used in the analysis of random walks is the *displacement*, which is the amount of change in position due to a jump. The displacement of the walker caused by the nth jump is denoted by s_n. From Eq. (3), s_n can be written as:

$$s_n \leftarrow \begin{cases} (1 - \lambda)(1 - x_{n-1}), & \text{with probability } p \\ x_{n-1}(\lambda - 1), & \text{with probability } q = 1 - p \end{cases} \tag{7}$$

As a result of Lemma 1, the expected displacement of SLWE walker converges to zero over time. This behavior is shown in Lemma 2.

Lemma 2. *Let s_n be the displacement of SLWE random walker caused by the nth jump. Then, $\mathbb{E}[s_\infty] = 0$.*

Proof. Displacement is the difference between final and initial positions, therefore:

$$s_n = x_n - x_{n-1} \tag{8}$$

Taking the expectation of both sides:

$$\mathbb{E}[s_n] = \mathbb{E}[x_n] - \mathbb{E}[x_{n-1}] \tag{9}$$

As $n \to \infty$, both $\mathbb{E}[x_n]$ and $\mathbb{E}[x_{n-1}]$ converge to $\mathbb{E}[x_\infty]$. Hence we get:

$$\mathbb{E}[s_\infty] = \mathbb{E}[x_\infty] - \mathbb{E}[x_\infty] \tag{10}$$

which is zero. □

As defined earlier, displacement is the amount of change in the position of the walker resulting from a single jump. In addition to this, *total displacement* is the overall change in position in multiple jumps, compared to the initial position. It is possible to formulate the position of the walker after n jumps by:

$$\begin{aligned} x_1 &= x_0 + s_1 \\ x_2 &= x_1 + s_2 = x_0 + s_1 + s_2 \\ x_3 &= x_2 + s_3 = x_1 + s_2 + s_3 = x_0 + s_1 + s_2 + s_3 \\ &\vdots \\ x_n &= x_0 + \textstyle\sum_{i=1}^{n} s_i \end{aligned} \tag{11}$$

where the term $\sum_{i=1}^{n} s_i$ the total displacement of the walker after n jumps and is denoted by s_n^*.

In accordance to Lemmas 1 and 2, expected total displacement after a large number of jumps is equal to the difference between p and the initial position, which is shown in Lemma 3.

Lemma 3. *Let s_n^* be the total displacement of SLWE random walker after n jumps. Then, $\mathbb{E}[s_\infty^*] = p - x_0$.*

Proof. From Eq. (11), s_n^* is written as:

$$s_n^* = x_n - x_0 \tag{12}$$

Taking the expectation of both sides, we get:

$$\mathbb{E}[s_n^*] = \mathbb{E}[x_n] - x_0 \tag{13}$$

As $n \to \infty$, we get:

$$\mathbb{E}[s_\infty^*] = \mathbb{E}[x_\infty] - x_0 \tag{14}$$

From Lemma 1, we know that $\mathbb{E}[x_n]$ converges to p as $n \to \infty$, hence we get:

$$\mathbb{E}[s_\infty^*] = p - x_0 \tag{15}$$

which completes the proof. □

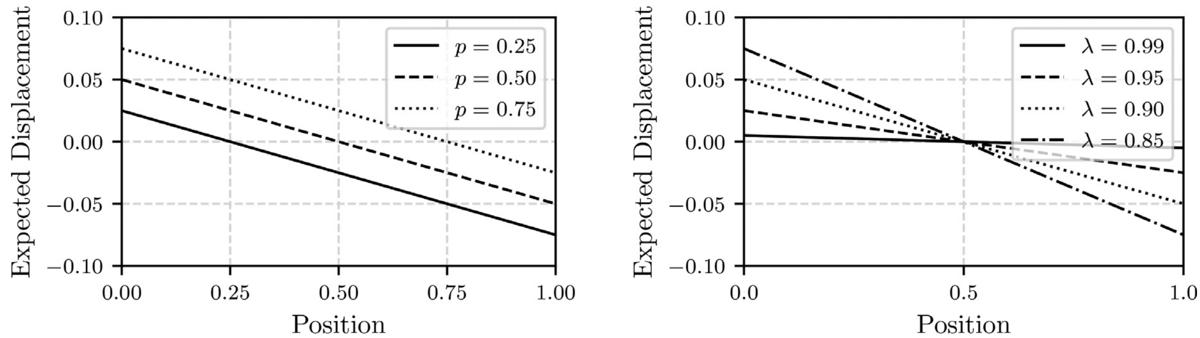
Now, let us study how the expected displacement of SLWE random walker caused by the nth jump changes with the position of the walker. It turns out that the expected displacement always moves the agent closer to p, which is shown in Lemma 4.

Lemma 4. *Let $\mathbb{E}[s_n]$ be the expected displacement of SLWE random walker caused by the nth jump. Then the following is true:*

$$\operatorname{sgn}(\mathbb{E}[s_n]) = \begin{cases} -1, & \text{if } x_{n-1} > p \\ 0, & \text{if } x_{n-1} = p \\ 1, & \text{if } x_{n-1} < p \end{cases} \tag{16}$$

Proof. From Eq. (7), $\mathbb{E}[s_n]$ is calculated as:

$$\begin{aligned} \mathbb{E}[s_n] &= q[x_{n-1}(\lambda - 1)] + p[(1 - \lambda)(1 - x_{n-1})] \\ &= (1 - \lambda)(-x_{n-1} + px_{n-1} + p - px_{n-1}) \\ &= (1 - \lambda)(p - x_{n-1}) \end{aligned} \tag{17}$$



(a) $\lambda = 0.9$ (b) $p = 0.5$

Fig. 3. The relation between expected displacement, position where the jump is initiated and λ is depicted for different p and λ values. The point where the expected displacement becomes zero is the equilibrium point. As seen in (a), p determines the position of equilibrium point, whereas as seen in (b), λ determines the amount of expected displacement.

where $(1-\lambda)$ and p are fixed and always non-negative since $0 < \lambda < 1$ and $0 \le p \le 1$. Therefore, the sign of $\mathbb{E}[s_n]$ is determined by the value of x_{n-1}.

The sign of $\mathbb{E}[s_n]$ for different values of x_{n-1} can be calculated by dividing the 1-dimensional space into three segments, which are $x_{n-1} < p$, $x_{n-1} = p$ and $x_{n-1} > p$. These segments are represented with points x_L, x_P and x_R where $x_L < x_P = p < x_R$.

Then, for $x_{n-1} = x_L$:

$$\mathbb{E}[s_n] = (1-\lambda)(p - x_L) \tag{18}$$

where the term $(p - x_L)$ is positive since $x_L < p$. Therefore $\mathrm{sgn}(\mathbb{E}[s_n]) = 1$.

For $x_{n-1} = x_P$:

$$\mathbb{E}[s_n] = (1-\lambda)(p - x_P) \tag{19}$$

which is zero. Therefore $\mathrm{sgn}(\mathbb{E}[s_n]) = 0$.

Finally, for $x_{n-1} = x_R$:

$$\mathbb{E}[s_n] = (1-\lambda)(p - x_R) \tag{20}$$

where the term $(p - x_R)$ is negative since $x_R > p$. Therefore $\mathrm{sgn}(\mathbb{E}[s_n]) = -1$, which completes the proof. □

Considering this relation between the expected displacement and the position of the walker, it is possible to say that each jump is expected to bring the walker closer to p. We call this point the *equilibrium point* and denote it by x_{eq}. Expected displacements of all jumps that are initiated from non-equilibrium points are non-zero and tend to move the walker gradually to the equilibrium point. Independent of the initial position, the walker eventually reaches to and tends to stay around this point. This relation between position and displacement can be considered as a *feedback loop* and is shown in Fig. 3.

Now, let us examine how sizes of jumps (i.e., absolute displacement) in either direction change with the position of the walker. Absolute displacement to the left and right are denoted by $\left|s_n^L\right|$ and $\left|s_n^R\right|$, respectively. As seen in Eq. (7), displacement depends on the position of the walker, λ and the direction of the jump. It turns out that, location p is the point where the expected absolute displacements caused by jumps in both directions become equal. In addition to this, if the walker is located at the left of p (i.e., $x_{n-1} < p$), the size of a right jump is greater than that of a left jump, and the opposite is true for $x_{n-1} > p$. This relation is shown in Lemma 5.

Lemma 5. *Let $\mathbb{E}[|s_n^L|]$ and $\mathbb{E}[|s_n^R|]$ be the expected absolute displacement of SLWE random walker to the left and right, respectively. Then,*

$$\mathrm{sgn}\left(\mathbb{E}\left[\left|s_n^R\right|\right] - \mathbb{E}\left[\left|s_n^L\right|\right]\right) = \begin{cases} -1, & \text{if } x_{n-1} > p \\ 0, & \text{if } x_{n-1} = p \\ 1, & \text{if } x_{n-1} < p \end{cases} \tag{21}$$

Proof. From Eq. (7), we can write:

$$\begin{aligned} \mathbb{E}\left[\left|s_n^R\right|\right] - \mathbb{E}\left[\left|s_n^L\right|\right] &= p(1-\lambda)(1-x_{n-1}) - qx_{n-1}(1-\lambda) \\ &= (1-\lambda)[p(1-x_{n-1}) - (1-p)x_{n-1}] \\ &= (1-\lambda)(p - px_{n-1} - x_{n-1} + px_{n-1}) \\ &= (1-\lambda)(p - x_{n-1}) \end{aligned} \tag{22}$$

Let us find the sign this function for all possible values of x_{n-1} by the same technique used in Lemma 4. The 1-dimensional space is divided into three segments, which are $x_{n-1} < p$, $x_{n-1} = p$ and $x_{n-1} > p$ and represented with points x_L, x_P and x_R respectively where $x_L < x_P = p < x_R$.

Then, for $x_{n-1} = x_L$:

$$\mathbb{E}\left[\left|s_n^R\right|\right] - \mathbb{E}\left[\left|s_n^L\right|\right] = (1-\lambda)(p - x_L) \tag{23}$$

where both terms $(p - x_L)$ and $(1-\lambda)$ are positive since $x_L < p$ and $0 < \lambda < 1$. Hence, $\mathrm{sgn}(\mathbb{E}[|s_n^R|] - \mathbb{E}[|s_n^L|]) = 1$.

For $x_{n-1} = x_P$:

$$\begin{aligned} \mathbb{E}\left[\left|s_n^R\right|\right] - \mathbb{E}\left[\left|s_n^L\right|\right] &= (1-\lambda)(p - x_P) \\ &= 0 \end{aligned} \tag{24}$$

which implies $\mathrm{sgn}(\mathbb{E}[|s_n^R|] - \mathbb{E}[|s_n^L|]) = 0$.

Finally, for $x_{n-1} = x_R$:

$$\mathbb{E}\left[\left|s_n^R\right|\right] - \mathbb{E}\left[\left|s_n^L\right|\right] = (1-\lambda)(p - x_R) \tag{25}$$

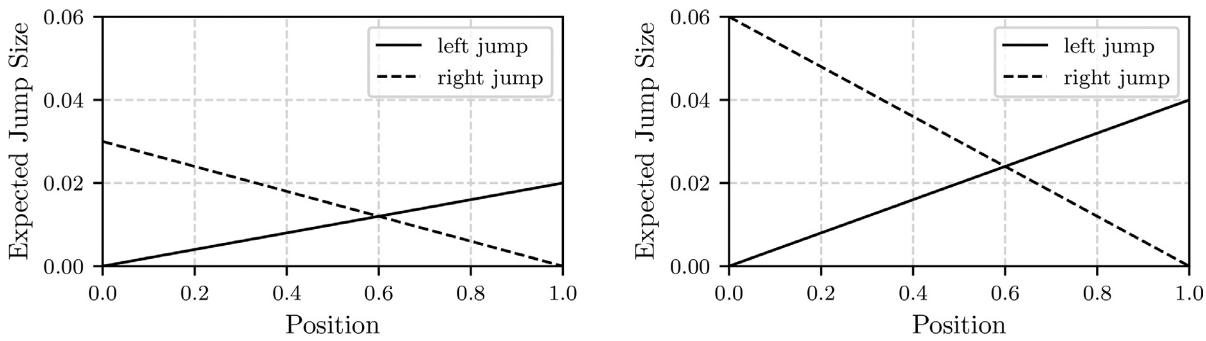
where the term $(p - x_R)$ is negative. Therefore, $\mathrm{sgn}(\mathbb{E}[|s_n^R|] - \mathbb{E}[|s_n^L|]) = -1$, which completes the proof. □

Considering this relation between the position and the absolute displacement, it is possible to say that it is harder for the walker (i.e., more jumps needed) to move away from p than getting close to p.

Expected jump sizes to either direction for different positions are shown in Fig. 4.

4.1.2. Evolution of positional probability density

Now, let us consider another aspect of SLWE random walk, which is how the positional probability density that determines the probability of the walker being at a given position, changes as the number of jumps grows large. We are particularly interested in the asymptotic behavior of the mentioned probability density, since it will form a base for the following sections. We will be utilizing the analysis techniques that are often used in the context of *statistical mechanics*.



(a) $p = 0.6$, $\lambda = 0.95$ (b) $p = 0.6$, $\lambda = 0.9$

Fig. 4. Expected jump sizes of SLWE random walk to either direction for different positions where the jump is initiated. In this case, $p = 0.6$. As previously mentioned in Lemma 5, left and right jumps sizes are equal only at position p. As seen from the difference between (a) and (b), λ effects the expected jump sizes but not the equilibrium point.

Up to now, we have used the number of jumps, n, as an argument to time-dependent terms in our notation. As mentioned earlier in the definition of SLWE random walk, we assume that jumps occur in regular time intervals, τ. That being the case, the first jump occurs at time τ, the second at time 2τ and the nth jump at time $n\tau$. We switch to time index notation as we will be dealing with partial derivatives shortly.

Let $W(x,t)$ be the probability density to find the walker at position interval $(x, x+dx)$ at time t. As previously stated, we are interested in the behavior of this probability density as $t \to \infty$. Also, it is useful to define the transition probability $P(x,t|x',t')$, which is the probability of the walker jumping to x at time t given that it is located at x' at time t'. Depending on the relation between t and t', this probability could indicate multi-jump transitions. However, since SLWE random walk satisfies the Markov property, which can be deduced from the fact that only the $(n-1)$th estimate is used for calculating the nth estimate, we mostly use $P(x,t|x',t-\tau)$, which is the single jump transition probability.

It is plausible to start with studying the change that happens in $W(x,t)$ after a single jump. This could be done by writing $W(x,t)$ in terms of probabilities of all possible positions that the walker could have been at time $t-\tau$ and are single jump away from x. Additionally, probabilities of such jumps should also be taken into account. This recursive approach is known as the Chapman-Kolmogorov equation [30] and could be shown as:

$$W(x,t+\tau) = \sum_{x'} W\left(x',t\right)P\left(x,t+\tau|x',t\right) \tag{26}$$

This equation, in essence, specifies that in order to reach x at time $t+\tau$, the walker must first reach an arbitrary intermediate position x', then make the transition from x' to x. By taking into account all possible intermediate positions from which the walker can reach x with a single jump and the probabilities of these jumps, it is possible to determine $W(x,t+\tau)$. This way of representing the evolution of a system that is modeled with a probability distribution is an example of a *master equation*.

As shown previously in Eq. (7), for SLWE random walk, displacement depends on the position of the walker. Thus, Eq. (26) could be written as:

$$W(x,t+\tau) = \sum_{s} W(x-s,t)\Pi(s|x-s) \tag{27}$$

where $\Pi(s|x)$ is the probability that the displacement of the walker caused by the jump initiated from position x is s. This probability can be obtained for SLWE random walk from Eq. (7) as:

$$\Pi(s|x) \leftarrow \begin{cases} p, & \text{if } s = (1-\lambda)(1-x) \\ q = 1-p, & \text{if } s = x(\lambda-1) \\ 0, & \text{otherwise} \end{cases} \tag{28}$$

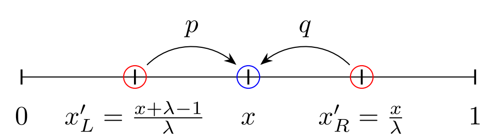


Fig. 5. There are only two positions from which an SLWE random walker can reach x with a single jump.

The generalized master equation given in Eq. (27) can be written for SLWE random walk easily. If the walker is at position x at time $t+\tau$, there are only two possible positions where the walker could have been at time t, which are x'_L (left) and x'_R (right). These specific positions are calculated using Eq. (3) and shown in Fig. 5.

$$\begin{aligned} 1-\lambda\left(1-x'_L\right) &= x \\ x'_L &= \tfrac{x+\lambda-1}{\lambda} \end{aligned} \tag{29}$$

$$\begin{aligned} \lambda x'_R &= x \\ x'_R &= \tfrac{x}{\lambda} \end{aligned} \tag{30}$$

Considering Eqs. (27), (28), (29) and (30) and the arbitrary intermediate point $x' = \{x'_L, x'_R\}$, it is possible to write $W(x,t+\tau)$ for SLWE random walk as:

$$\begin{aligned} W(x,t+\tau) &= W\left(x'_L,t\right)\Pi\left(x-x'_L|x'_L\right) + W\left(x'_R,t\right)\Pi\left(x-x'_R|x'_R\right) \\ &= W\left(x'_L,t\right)p + W\left(x'_R,t\right)q \\ &= W\left(\tfrac{x+\lambda-1}{\lambda},t\right)p + W\left(\tfrac{x}{\lambda},t\right)q \end{aligned} \tag{31}$$

It is now useful to introduce *Kramers-Moyal expansion*, which is a method of applying *Taylor series* expansion to a master equation in order to obtain a partial differential equation, and given as [31,32]:

$$\frac{\partial}{\partial t}W(x,t) = \sum_{n=1}^{\infty} \frac{(-1)^n}{n!}\frac{\partial^n}{\partial x^n}\left[\alpha_n(x)W(x,t)\right] \tag{32}$$

where $\alpha_n(x)$ is the nth jump moment and given as:

$$\alpha_n(x) = \int_{-\infty}^{\infty} \left(x'-x\right)^n P\left(x'|x\right)dx' \tag{33}$$

where $P(x'|x)$ is the probability of jumping from x to x'. For applying Kramers-Moyal expansion, the number of terms to include in the series is an important decision. *Pawula theorem* [33] specifies that the expansion might stop after the first or the second term. If the expansion does not stop after the second term, then it must contain infinite number of terms for the solution to be considered as a probability density function [34].

Applying Kramers-Moyal expansion to Eq. (31) with the assumption of τ being small and keeping only the first two terms, we obtain the *Fokker-Planck equation* [35,36]:

$$\frac{\partial}{\partial t}W(x,t) = -\frac{\partial}{\partial x}[\mu(x)W(x,t)] + \frac{\partial^2}{\partial x^2}[D(x)W(x,t)] \tag{34}$$

where,

$$\mu(x) = \alpha_1(x) \tag{35}$$

is the *drift velocity* and,

$$D(x) = \frac{\alpha_2(x)}{2} \tag{36}$$

is the *diffusion coefficient*.

Let us examine the asymptotic behavior of drift velocity and diffusion coefficient of SLWE random walk.

Theorem 6. *Let $\mu(x_t)$ be the drift velocity of SLWE random walk at time t. Then, $\mu(x_\infty) = 0$.*

Proof. Drift velocity at time t can be calculated as:

$$\begin{aligned}\mu(x_t) &= \alpha_1(x_t)\\ &= \int_{-\infty}^{\infty}\left(x'_{t+\tau} - x_t\right)P\left(x'_{t+\tau}|x_t\right)dx'_t\end{aligned} \tag{37}$$

For SLWE random walk, the first jump moment can be written as:

$$\begin{aligned}\alpha_1(x_t) &= \textstyle\sum_s s\Pi(s|x_t)\\ &= \mathbb{E}[s_t]\end{aligned} \tag{38}$$

As $t \to \infty$, we get:

$$\mu(x_\infty) = \mathbb{E}[s_\infty] \tag{39}$$

From Lemma 2, we plug the value of $\mathbb{E}[s_\infty]$:

$$\mu(x_\infty) = 0 \tag{40}$$

which completes the proof. □

This means that, the drift term in Eq. (34) tends to approach zero for SLWE random walk with the growing number of jumps. This is not the case for simple random walk defined earlier, where an asymmetry in the jump probabilities (i.e., $p \neq q$) causes a nonzero drift velocity of the walker. Thus, as long as the asymmetry is present, the walker constantly gets drifted towards the dominant side. For SLWE random walk, the feedback loop between the position and the displacement forces the walker to approach the equilibrium point $x_{eq} = p$. This does not affect the asymmetry in transition probabilities, but it produces another asymmetry in the sizes of left and right jumps. These two asymmetries are exactly the opposite, that is:

$$\frac{p}{q} = \frac{\left|s_n^L\right|}{\left|s_n^R\right|} \tag{41}$$

Consequently, since these two asymmetries cancel out the effect of each other, the expected displacement and the net drift both approach zero.

In order to examine the diffusion coefficient, we first investigate the asymptotic behavior of the expected squared position of SLWE random walker and show that it is constant in Lemma 7.

Lemma 7. *Let $\mathbb{E}[x_t^2]$ be the expected value of squared position of SLWE random walk. Then, $\mathbb{E}[x_\infty^2]$ is constant.*

Proof. From Eq. (3), x_t^2 is written as:

$$x_t^2 \leftarrow \begin{cases}[1-\lambda(1-x_{t-\tau})]^2, & \text{with probability } p\\ [\lambda x_{t-\tau}]^2, & \text{with probability } q = 1-p\end{cases} \tag{42}$$

From Eq. (42), the conditional expectation is written as:

$$\begin{aligned}\mathbb{E}\left[x_t^2|x_{t-\tau}\right] &= p[1-\lambda(1-x_{t-\tau})]^2 + q[\lambda x_{t-\tau}]^2\\ &= p\left[1-2\lambda(1-x_{t-\tau}) + \lambda^2(1-x_{t-\tau})^2\right] + q\lambda^2x_{t-\tau}^2\\ &= p\left(1-2\lambda-2\lambda x_{t-\tau} + \lambda^2 - 2\lambda^2x_{t-\tau} + \lambda^2x_{t-\tau}^2\right) + (1-p)\lambda^2x_{t-\tau}^2\\ &= \lambda^2x_{t-\tau}^2 + 2\lambda p x_{t-\tau}(1-\lambda) + p(1-\lambda)^2\end{aligned} \tag{43}$$

Taking the expected value of the conditional expectation yields:

$$\mathbb{E}\left[\mathbb{E}\left[x_t^2|x_{t-\tau}\right]\right] = \mathbb{E}\left[x_t^2\right] = \lambda^2\mathbb{E}\left[x_{t-\tau}^2\right] + 2\lambda p(1-\lambda)\mathbb{E}[x_{t-\tau}] + p(1-\lambda)^2 \tag{44}$$

As $t \to \infty$, $\mathbb{E}[x_t^2]$ and $\mathbb{E}[x_{t-\tau}^2]$ converge to $\mathbb{E}[x_\infty^2]$. Therefore we get:

$$\mathbb{E}\left[x_\infty^2\right] = \lambda^2\mathbb{E}\left[x_\infty^2\right] + 2\lambda p(1-\lambda)\mathbb{E}[x_\infty] + p(1-\lambda)^2 \tag{45}$$

Since $\mathbb{E}[x_\infty] = p$ (Lemma 1),

$$\begin{aligned}\mathbb{E}\left[x_\infty^2\right] &= \lambda^2\mathbb{E}\left[x_\infty^2\right] + 2\lambda p^2(1-\lambda) + p(1-\lambda)^2\\ \mathbb{E}\left[x_\infty^2\right]\left(1-\lambda^2\right) &= 2\lambda p^2(1-\lambda) + p(1-\lambda)^2\\ \mathbb{E}\left[x_\infty^2\right](1+\lambda) &= 2\lambda p^2 + p(1-\lambda)\\ \mathbb{E}\left[x_\infty^2\right] &= \frac{2\lambda p^2 + p(1-\lambda)}{1+\lambda}\end{aligned} \tag{46}$$

which is constant. □

Using Lemma 7, we show in Theorem 8 that the diffusion coefficient is also asymptotically constant.

Theorem 8. *Let $D(x_t)$ be the diffusion coefficient of SLWE random walk at time t. Then, $D(x_\infty)$ is constant.*

Proof. The diffusion coefficient at time t is:

$$\begin{aligned}D(x_t) &= \frac{\alpha_2(x_t)}{2}\\ &= \frac{1}{2}\int_{-\infty}^{\infty}\left(x'_{t+\tau} - x_t\right)^2P\left(x'_{t+\tau}|x_t\right)dx'_t\end{aligned} \tag{47}$$

For SLWE random walk, the second jump moment can be written as:

$$\begin{aligned}\alpha_2(x_t) &= \textstyle\sum_s s^2\Pi(s|x_t)\\ &= \mathbb{E}\left[s_t^2\right]\end{aligned} \tag{48}$$

Hence, as $t \to \infty$, diffusion coefficient becomes:

$$D(x_\infty) = \frac{\mathbb{E}\left[s_\infty^2\right]}{2} \tag{49}$$

In order to calculate $\mathbb{E}[s_\infty^2]$, we first derive $\mathbb{E}[s_t^2]$. From Eq. (7) s_t^2 is written as:

$$s_t^2 \leftarrow \begin{cases}(1-\lambda)^2(1-x_{t-\tau})^2, & \text{with probability } p\\ x_{t-\tau}^2(\lambda-1)^2, & \text{with probability } q = 1-p\end{cases} \tag{50}$$

From Eq. (50), we write the conditional expectation as:

$$\begin{aligned}\mathbb{E}\left[s_t^2|x_{t-\tau}\right] &= p\left[(1-\lambda)^2(1-x_{t-\tau})^2\right] + q\left[x_{t-\tau}^2(\lambda-1)^2\right]\\ &= (1-\lambda)^2\left[p(1-x_{t-\tau})^2 + (1-p)x_{t-\tau}^2\right]\\ &= (1-\lambda)^2\left[p\left(1-2x_{t-\tau} + x_{t-\tau}^2\right) + x_{t-\tau}^2 - px_{t-\tau}^2\right]\\ &= (1-\lambda)^2\left(x_{t-\tau}^2 - 2px_{t-\tau} + p\right)\end{aligned} \tag{51}$$

Taking the expectation a second time, we get:

$$\mathbb{E}\left[s_t^2\right] = (1-\lambda)^2\left(\mathbb{E}\left[x_{t-\tau}^2\right] - 2p\mathbb{E}[x_{t-\tau}] + p\right) \tag{52}$$

As $t \to \infty$, we obtain:

$$\mathbb{E}\left[s_\infty^2\right] = (1-\lambda)^2\left(\mathbb{E}\left[x_\infty^2\right] - 2p\mathbb{E}[x_\infty] + p\right) \tag{53}$$

From the proofs of Lemma 1 and Lemma 7, we plug values of $\mathbb{E}[x_\infty]$ and $\mathbb{E}[x_\infty^2]$:

$$\begin{aligned}\mathbb{E}\left[s_\infty^2\right] &= (1-\lambda)^2\left(\frac{2\lambda p^2 + p(1-\lambda)}{1+\lambda} - 2p^2 + p\right)\\ &= \frac{2(1-\lambda)^2\left(p-p^2\right)}{1+\lambda}\end{aligned} \tag{54}$$

Finally we get:

$$D(x_\infty) = \frac{(1-\lambda)^2(p-p^2)}{1+\lambda} \tag{55}$$

where p and λ are constants and there is no positional term. Therefore, the theorem is proven. □

This result, diffusion coefficient being asymptotically constant, is important since it significantly simplifies the solution of the Fokker-Planck equation.

Now, let us examine the statistical variance of positions of SLWE random walk as $t \to \infty$. A similar proof is given in [3].

Theorem 9. *Statistical variance of the positions of SLWE random walker is only determined by p and λ asymptotically.*

Proof. Statistical variance is calculated as:

$$\mathrm{Var}[x_\infty] = \mathbb{E}\left[x_\infty^2\right] - \mathbb{E}[x_\infty]^2 \tag{56}$$

From Lemma 1 and Lemma 7, we plug the values of $\mathbb{E}[x_\infty^2]$ and $\mathbb{E}[x_\infty]^2$ and obtain:

$$\begin{aligned} \mathrm{Var}[x_\infty] &= \tfrac{2\lambda p^2 + p(1-\lambda)}{1+\lambda} - p^2 \\ &= \tfrac{p(1-p) - p\lambda(1-p)}{1+\lambda} \\ &= \tfrac{(1-\lambda)\left(p-p^2\right)}{1+\lambda} \end{aligned} \tag{57}$$

which is consistent with [3]. □

With Theorems 6 and 8 we have shown that the net drift velocity for SLWE random walk diminishes over time and the diffusion coefficient is asymptotically constant. As a result, as $t \to \infty$, the Fokker-Planck equation is reduced to:

$$\frac{\partial}{\partial t} W(x,t) = D \frac{\partial^2}{\partial x^2} W(x,t) \tag{58}$$

which is the well-known diffusion equation (sometimes called Fick's second law of diffusion) and explains how concentration changes with time due to diffusion.

Eq. (58) has the same mathematical form with the famous *heat equation*, which describes the time-dependent distribution of heat in a solid medium. Therefore, a *fundamental solution* of heat equation, which is also called the *heat kernel*, is also valid for Eq. (58) with the only difference of using diffusion coefficient D instead of thermal conductivity k. This solution is given as [37]:

$$W(x,t) = \frac{1}{\sqrt{4\pi Dt}} \exp\left(-\frac{x^2}{4Dt}\right) \tag{59}$$

which is a normal distribution with $\mu = 0$ and $\sigma^2 = 2Dt$.

One important characteristic of the solution given in Eq. (59) is that, given enough time, it transforms into a uniform distribution. This also explains the fact that the heat exchange stops when the thermal equilibrium is reached. However, this is not the case in SLWE random walk. Since the statistical variance of the positions of the walker is constant as shown in Theorem 9, the positional probability density does not become a uniform distribution over time. Instead, it converges to a normal distribution and stays the same as long as p does not change.

Another important point regarding the solution given in Eq. (59) is that there is no limitation on the particle positions, $x \in (-\infty, \infty)$, i.e., a particle can be located anywhere. However, this is also not the case in SLWE random walk. Since positions of the walker correspond to probability values, it must be limited to a range of values between 0 and 1, that is $x \in [0,1]$. As previously mentioned, update rules of SLWE for both binomial and multinomial cases already ensure that positions always remain in this range due to the fact that jump size approaches 0 as position approaches 0 or 1. Therefore, in its unbounded form, solution given in Eq. (59) does not perfectly model the positional probability density of SLWE random walk.

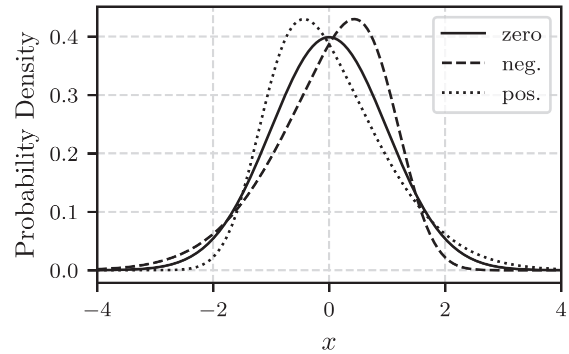


Fig. 6. Standard normal densities with different skewness values. Zero, negative and positive skew densities have the same mean and variance.

In order to satisfy $x \in [0,1]$, probabilities of positions beyond this range could be set to zero. However, since the support of normal distribution is defined as $x \in \mathbb{R}$, clipping these probability values to zero without any scaling operation would violate the second axiom of probability, in other words, the following equation would not be satisfied:

$$\int_0^1 W(x,t)dx = 1 \tag{60}$$

A mathematically justifiable way of applying truncation to a probability density function is to scale it up uniformly so that the integral over the resulting range equals to 1. The resulting distribution is an example of *doubly truncated normal distribution*, where a normal distribution is clipped from both a lower and an upper bound, and defined as $\psi(\bar{\mu}, \bar{\sigma}, a, b; x)$ [38] where $\bar{\mu}$ and $\bar{\sigma}$ are the mean and variance of the original (sometimes called the parent) distribution; a and b are the lower and upper bounds of truncation.

For SLWE random walk, the effect of truncation depends on the values of p and λ, which determine the mean and the variance of the positional probability density. The difference between the original and the truncated distribution gets smaller as the amount of area being clipped out gets smaller. Considering this, it is expected to observe less difference between the original and the truncated distribution as the mean of original distribution gets farther from both lower and upper bounds (i.e., p being close to 0.5) and the variance of original distribution gets smaller (λ being close to 1).

Another important effect of introducing bounds to possible position values is that it alters the third moment of the distribution, which is a measure for *skewness* (i.e., the lack of symmetry). A distribution with a negative skew (also called left-skewed or skewed to the left) has a longer left tail and is concentrated on the right side, whereas the opposite is valid for a distribution with a positive skew. The effect of skewness is depicted in Fig. 6.

The reason behind having bounds in possible values of a random variable introducing skewness is further analyzed in Theorem 10.

Theorem 10. *Let Y be a real random variable, a a real number and $X_a =: \max(a, Y)$. Then, the skewness of X is a non-decreasing function of a.*

Proof. Let $\gamma(a)$ be the skewness of X_a, then $\gamma(a)$ must be a non-decreasing function. The nth moment of X_a is,

$$M_n(a) = \int_a^\infty (x-a)^n f(x)dx \tag{61}$$

where $f(x)$ is the probability density function of Y.

Leibniz integral rule states that,

$$\frac{d}{dt}\int_a^b f(x,t)dx = \int_a^b \frac{\partial}{\partial t} f(x,t)dx \tag{62}$$

Applying Eq. (62) to moment equation, we get:

$$\begin{aligned} M'_n(a) &= \tfrac{d}{da}\textstyle\int_a^\infty (x-a)^n f(x)dx \\ &= \textstyle\int_a^\infty \tfrac{\partial}{\partial a}(x-a)^n f(x)dx \end{aligned} \tag{63}$$

where, with the help of chain rule for integration, $\frac{\partial}{\partial a}(x-a)^n$ is written as:

$$\begin{aligned} \tfrac{\partial}{\partial a}\left((x-a)^n\right) &= \tfrac{\partial u^n}{\partial u}\tfrac{\partial u}{\partial a} \\ &= -n(x-a)^{n-1} \end{aligned} \tag{64}$$

where $u = x - a$. Therefore, we get:

$$\begin{aligned} M'_n(a) &= \textstyle\int_a^\infty -n(x-a)^{n-1} f(x)dx \\ &= -nM_{n-1}(a) \end{aligned} \tag{65}$$

As a special case, the derivative of the first moment is,

$$M'_1(a) = \int_a^\infty -f(x)dx = -F(a) \tag{66}$$

where $F(a)$ is the cumulative distribution function of Y.

Skewness, namely the third standardized moment, is defined as:

$$\begin{aligned} \overline{\mu}_3 &= \tfrac{\mu_3}{\sigma^3} \\ &= \frac{\mathbb{E}\left[(x-\mu)^3\right]}{\left(\mathbb{E}\left[(x-\mu)^2\right]\right)^{3/2}} \\ &= \frac{\mathbb{E}\left[x^3\right]-3\mu\mathbb{E}\left[x^2\right]+3\mu^2\mathbb{E}[x]-\mu^3}{\left(\mathbb{E}\left[x^2\right]-2\mu\mathbb{E}[x]+\mu^2\right)^{3/2}} \end{aligned} \tag{67}$$

By substituting μ and $\mathbb{E}[x]$ with $M_1(a)$, $\mathbb{E}[x^2]$ with $M_2(a)$ and $\mathbb{E}[x^3]$ with $M_3(a)$, we obtain $\gamma(a)$. Note that, we omit indices while writing moments for reasons of simplicity.

$$\gamma(a) = \frac{M_3-3M_1M_2+2M_1^3}{(M_2-M_1^2)^{3/2}} \tag{68}$$

Taking the derivative of skewness as shown in Eq. (65), we get:

$$\begin{aligned} \gamma'(a) &= \left(M'_3 - 3M'_1M_2 - 3M_1M'_2 + 6M_1^2M'_1\right)\left(M_2-M_1^2\right)^{-3/2} \\ &\quad +\left(M_3-3M_1M_2+2M_1^3\right)(-3/2)\left(M_2-M_1^2\right)^{-5/2}\left(M'_2-2M_1M'_1\right) \\ &= \left(-3M_2+3M_2F+6M_1^2-6M_1^2F\right)\left(M_2-M_1^2\right)^{-3/2} \\ &\quad +\left(M_3-3M_1M_2+2M_1^3\right)(-3/2)\left(M_2-M_1^2\right)^{-5/2}(-2M_1+2M_1F) \\ &= \frac{(1-F)(-3)\left(M_2-2M_1^2\right)}{\left(M_2-M_1^2\right)^{3/2}} + \frac{(1-F)(-2M_1)(-3/2)\left(M_3-3M_1M_2+2M_1^3\right)}{\left(M_2-M_1^2\right)^{5/2}} \\ &= \frac{(1-F)\left(-3M_2^2+3M_1^2M_2+6M_1^2M_2-6M_1^4+3M_1M_3-9M_1^2M_2+6M_1^4\right)}{\left(M_2-M_1^2\right)^{5/2}} \\ &= \frac{3(1-F)\left(M_1M_3-M_2^2\right)}{\left(M_2-M_1^2\right)^{5/2}} \end{aligned} \tag{69}$$

where 3 and $(1-F)$ are obviously non-negative. The component in the denominator, $(M_2 - M_1^2)$, is essentially the variance of X, namely,

$$M_2 - M_1^2 = \mathbb{E}\left[X^2\right] - (\mathbb{E}[X])^2 = \mathrm{Var}[X] \tag{70}$$

which is always non-negative. Finally, $(M_1M_3 - M_2^2)$ can be shown to be non-negative by applying Cauchy-Schwarz inequality, which states that,

$$(\mathbb{E}[AB])^2 \le \mathbb{E}\left[A^2\right]\mathbb{E}\left[B^2\right] \tag{71}$$

for any two random variables A and B. $(M_1M_3 - M_2^2)$ can be rewritten as,

$$M_1M_3 - M_2^2 = \mathbb{E}[X]\mathbb{E}\left[X^3\right] - \left(\mathbb{E}\left[X^2\right]\right)^2 \tag{72}$$

where the relation $\mathbb{E}[X]\mathbb{E}[X^3] \ge (\mathbb{E}[X^2])^2$ is obtained for $A = X^{1/2}$ and $B = X^{3/2}$ in Eq. (71).

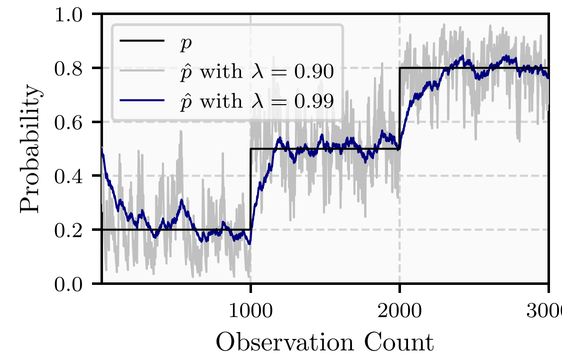


Fig. 7. Estimates obtained from two SLWE instances with different learning rates, run on a non-stationary sequence composed of three regimes, which are sequences sampled from different binomial distributions with parameters 0.2, 0.5 and 0.8, respectively.

Thus, $\gamma'(a)$ is always non-negative, which implies that the skewness $\gamma(a)$ is a non-decreasing function of the lower bound introduced to the random variable. □

As a result, we can deduce that if a random variable is bounded from below, depending on the bound itself, the probability distribution will have non-negative skewness. Proof of Theorem 10 can be modified to show that an upper bound resulting non-positive skewness. In the case of asymmetrical double bounds, net skewness will always be non-zero. Therefore, we conclude that in most cases, positional probability distribution converges to a normal distribution; however if p is close to either bound and/or variance is so high that the distribution is bounded from either end, we observe a truncated skew-normal distribution.

4.2. Adaptive learning mode

The main focus of this study is motivated by the fact that SLWE with a fixed learning coefficient λ, faces a trade-off situation between *adaptation speed*, i.e., how fast the estimates adapt to a new regime and *tracking stability*, i.e., how stable the estimates are. This trade-off is also described as the *Stability-Plasticity Dilemma* in [26]. This situation occurs due to the nature of the multiplicative updates performed on estimates after each observation. As seen in the update rule of SLWE given in Eq. (1), learning coefficient λ determines the effect of the most recent observation over the running estimate. For simplicity, sometimes we prefer to use the term *learning rate*, denoted by η, and $\eta = 1 - \lambda$.

Let us elaborate the effect of learning rate on the running estimate. A larger learning rate results in updates with larger magnitudes, which is useful especially when a new regime is being learned. However, this makes estimates less stable, that is to say, the variance of estimates becomes larger. This can also be deduced from Theorem 9, where the statistical variance of positions of SLWE random walk is directly proportional to η. Resulting intense fluctuations of estimates negatively impact the overall accuracy of estimator. This drawback has a higher impact in case noise is involved in observations, for the reason that a larger learning rate would amplify the disruptive effect of such incorrect instances.

A smaller learning rate, on the other hand, reduces the plasticity and positively affects the stability of the estimates. Since each subsequent update would occur within a relatively smaller amplitude, the variance of resulting values becomes smaller. Further, a smaller learning rate reduces the negative effect of noisy instances. The effect of different learning rates is shown in Fig. 7, where an SLWE instance with higher learning rate tracks new values of the target parameter much faster, with the drawback of a larger variance of estimates.

A classical approach to this problem is to prefer exploration at the beginning in order to build a general idea about the environment and gradually prioritize exploitation of current knowledge over time. One possible way to achieve this is to use a decreasing function to lower the corresponding exploration parameter gradually with time. However, this approach lacks a mechanism for re-prioritizing exploration after changes and still suffers from the problem of finding best values for parameters like decrease rate, type of decay, maximum and minimum values for each application. Since there is no set of parameters that performs well on every type of problem (no free lunch theorem), an adaptive method is preferable.

One possible way to make learning rate adaptive is to control it according to the *learning progress*. This way, exploitation could be gradually prioritized when learning approaches to a level of saturation. However, measuring and detecting this level is a challenging task. With this study, we propose a technique to measure the learning progress of SLWE method by utilizing the asymptotic behavior derived in Section 4.1.

4.2.1. Measuring learning progress

A learner that uses the $L_{R\text{-}I}$ reinforcement scheme, updates estimates in a multiplicative manner after each observation. Being an evolving method that utilizes $L_{R\text{-}I}$ scheme, SLWE only maintains the latest estimate for the target parameter to be estimated. However, throughout learning, the history of values of estimates forms a *time series*. By analyzing this series of data, it is possible to split it into a sequence of segments each with different characteristics. One useful property to consider for segmentation is *stationarity*.

A stationary process is said to remain in a statistical equilibrium, that is to say, it preserves its statistical properties throughout time. In particular, a stationary time series varies about a constant mean with a constant variance [39]. Considering the characteristics of SLWE, there is a time period when the drift velocity is non-zero, hence the mean is not constant. Therefore, the time series of estimates in this period is non-stationary. However, after some time, drift velocity approaches zero and estimates converge to a normal distribution with a specific mean and variance. Time series of estimates after this point is stationary, that is, its mean and variance are preserved as long as the regime continues. This transition from non-stationarity to stationarity is an indication of saturation and a checkpoint in learning progress. In this regard, prioritizing exploitation after the mentioned transition is preferable.

Moreover, changes in the target parameter due to the non-stationary nature, results in previously weakly converged estimates being not correct anymore (assuming that the true value of the target parameter is changed). This change causes the previously diminished drift velocity to become non-zero again. Therefore, time series becomes non-stationary again and the learning progress is reset, making exploration preferable once more.

In view of these, our proposed method is built upon detecting the stationarity transition of estimates. Specifically, we detect non-stationary to stationary transition as it indicates saturation; and stationary to non-stationary transition as it indicates a regime change. These two types of transitions are shown in Fig. 8. Based on these changes, the *learning mode* is changed, which could be (i) an *exploratory* mode, which is activated after a stationary to non-stationary transition and utilizes a higher learning rate or (ii) an *exploitative* mode, which is activated after a non-stationary to stationary transition and utilizes a lower learning rate.

Now, let us explain the mentioned stationarity transitions and how we detect them.

Non-stationary to Stationary Transition

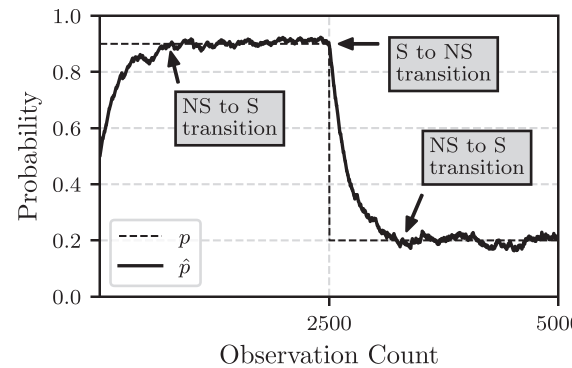


Fig. 8. Estimate series is a sequence of two stationarity regions. In this example, first and second regimes are composed of points sampled from binomial distributions with parameters 0.9 and 0.2, respectively.

In the early stages of a regime, consecutive updates on estimates gradually move the mean to the correct value of the target parameter. A series of estimates during this phase forms a signal that can be considered as a result of a non-stationary process. Specifically, since nonzero drift velocity in this phase causes the mean to change progressively, the underlying process could be categorized as *trend-stationary*. A trend-stationary process is defined as:

$$y_t = \mu_t + \epsilon_t \quad (73)$$

where μ_t is a deterministic mean trend and ϵ_t is a stationary stochastic process. As the learning proceeds, since drift velocity diminishes over time, this trend tends to disappear. After it disappears, estimates start to form a type of signal that is a result of a stationary process.

Note that, analytically calculating drift velocity is not possible since the true value of the target parameter, p, is not known. Hence, we estimate drift velocity considering the changes in expected value of estimates.

Now, let us introduce some concepts that we use in order to detect this transition.

A series of estimates ranging within the nth and mth estimates is denoted by $S(n, m)$ and shown as:

$$S(n, m) = \left\{ \hat{p}(n), \hat{p}(n+1), \ldots, \hat{p}(m) \right\} \quad (74)$$

where $m \geq n \geq 0$ and $\hat{p}(n)$ is the nth estimate.

The mean of series $S(n, m)$ is denoted by $\mu_{S(n,m)}$ and calculated as:

$$\mu_{S(n,m)} = \frac{1}{m-n+1} \sum_{i=n}^{m} \hat{p}(i) \quad (75)$$

The variance of series $S(n, m)$ is denoted by $\sigma^2_{S(n,m)}$ and calculated as:

$$\sigma^2_{S(n,m)} = \frac{1}{m-n+1} \sum_{i=n}^{m} (\hat{p}(i) - \mu_{S(n,m)})^2 \quad (76)$$

From Lemma 1, we know that $\mu_{S(n,m)}$ converges to p as $m \to \infty$ and there is no regime change. In this regard, it is useful to track the difference between consecutive values of expected estimates as the learning proceeds. An indication of drift velocity approaching zero is expected estimates getting arbitrarily closer for a period of time. This indication could be detected by tracing the *moving variance* of expected estimates from the starting point of the last stationary to non-stationary transition. For this reason, we define the series of expected estimates ranging within the nth and mth expected estimates, where each expected estimate is calculated considering estimates starting from the n_{ns}th one. This series is denoted by $S_E(n_{ns}, n, m)$ and shown as:

$$S_E(n_{ns}, n, m) = \left\{\mu_{S(n_{ns},n)}, \mu_{S(n_{ns},n+1)}, \ldots, \mu_{S(n_{ns},m)}\right\} \tag{77}$$

where $m \geq n \geq n_{ns} \geq 0$. This series definition is useful for calculating the moving variance for nth estimate, which is denoted by $s^2(n_{ns}, n)$ and calculated as:

$$s^2(n_{ns}, n) = \frac{1}{w_s} \sum_{i=n-w_s+1}^{n} \left(\mu_{S(n_{ns},i)} - \mu_{S_E(n_{ns},n-w_s+1,n)}\right)^2 \tag{78}$$

where w_s is the window size, $\mu_{S_E(n_{ns},n,m)}$ is the mean of expected estimates starting from the nth to the mth one and calculated as:

$$\mu_{S_E(n_{ns},n,m)} = \frac{1}{m-n+1} \sum_{i=n}^{m} S_E(n_{ns}, i, i) \tag{79}$$

As the drift velocity approaches zero, the value of $s^2(n_{ns}, n)$ also approaches zero. Hence, convergence is detected at time n if:

$$s^2(n_{ns}, n) < \tau_s \tag{80}$$

where τ_s is the *stationarity threshold*.

Suppose convergence is detected at time $n = n_s$; then the estimates for the first and second moments of the distribution are calculated as:

$$\bar{x} = \mu_{S(n_{ns},n_s)} \tag{81}$$

$$s^2 = \sigma^2_{S(n_{ns},n_s)} \tag{82}$$

These estimates will be used for detecting the stationary to non-stationary transition.

Notice that in some of the equations, a series of estimates (or expected estimates) that is starting from the n_{ns}th estimate is used, which is the estimate where the last stationary to non-stationary transition happened. This is to consider only the estimates for the current regime. If there are no such transitions happened before, since there is nothing known about the distribution at the beginning, we assume that it is non-stationary and calculate from the first estimate.

Stationary to Non-stationary Transition

Due to the non-stationary nature of the sequence, value of the target parameter p switches to a new random value in unknown intervals. After a switch, the estimate of p becomes obsolete, which causes the previously diminished drift velocity to become nonzero again. As a result, a stationary to non-stationary transition occurs in the series of estimates.

This transition can be detected by evaluating how good the previously learned distribution is at representing new estimates. As estimates from a different regime are obtained, this measure tends to decrease with a rate determined by the drift velocity, hence by the learning rate.

Considering estimates of the first and second moments, $\bar{x}$ and s^2, which are calculated in the previous stationary phase, it is possible to measure the deviation from the current model by:

$$d = \frac{\left|\hat{p}(n) - \bar{x}\right|}{s} \tag{83}$$

and a regime change is detected when the following condition,

$$d > \tau_{ds} \tag{84}$$

where τ_{ds} is called the *deviation severity threshold*, is observed τ_{dc} times, which is called the *deviation count threshold*. For $\tau_{ds} = 1$, $\tau_{ds} = 2$ and $\tau_{ds} = 3$, the estimate must be different than $\approx 68.27\%$, $\approx 95.45\%$ and $\approx 99.73\%$ of the distribution respectively to be counted as deviative.

The adaptive learning mode switching method is given in Algorithm 1 and the single-pass calculation of utilized statistical properties are given in Algorithms 2 and 3.

4.3. Complexity analysis

With each observation, sample mean and variance of the estimates obtained from SLWE is calculated with a single-pass approach (Welford's online algorithm [40]), where the new values are calculated in $O(1)$ time utilizing a recurrence relation.

The non-stationary to stationary detection is based on the moving variance of sample means, which is also calculated in $O(1)$ time with a similar approach. The first w_s calculations are simply done by the single-pass method; then with each observation, oldest element is replaced with the new one. A similar recurrence relation of already-known values is obtained for mean and variance.

The update functions are given in Algorithms 2 and 3 and the derivations are provided in Appendix A.

Consequently, the mechanism for adaptively switching between exploratory and exploitative modes works in $O(1)$ time for each observation, making the time complexity of SLWE-ALM $O(n)$ where n is the number of observations.

5. Experimental evaluation

In this section, theoretical results obtained in previous sections are simulated and experimentally tested on both synthetic and real-world data and results are reported.

5.1. Simulation of provided analysis

The provided analysis of SLWE random walk is verified with a stochastic simulation of extended number of jumps. Our goal is to show that (a) the drift velocity diminishes with growing number of jumps, (b) the positional probability density turns into a normal distribution with $\mu = p$ and $\sigma^2 = (p - p^2)(1 - \lambda)/(1 + \lambda)$, and (c) when the distribution is clipped in an asymmetric way, we observe non-zero skewness.

In Fig. 9, the evolution of positional probability density for SLWE random walk is shown. As seen in four consecutive periods of 250 jumps each in Fig. 9(a) to 9(d), as the number of jumps increases, the histogram of positions of the walker "drifts" towards p. After some time, as the sample mean gets closer to p, the drift diminishes and estimate distribution approaches a normal distribution. Also, during this time period, the sample variance tends to decrease towards the theoretical value.

The large time behavior of the same SLWE instance in Fig. 9 is given in Fig. 10. In Fig. 10(a), histogram of 2×10^5 jumps is shown, where the sample mean and variance are equal to the theoretical values up to 3 and 4 decimal places, respectively. In Fig. 10(b), the quantile-quantile(QQ) plot of 1×10^4 positions is given, where the linearity of the points indicate that the data is normal distributed.

In Fig. 11, the large time behavior of estimates obtained from SLWE with $\lambda = 0.95$ and $p = \{0.08, 0.92\}$ is given. As seen in Fig. 11(a) and 11(c), sample mean and variance are again very close to theoretical values. However, as predicted, the histogram of positions is left-skewed in Fig. 11(a) and right-skewed in Fig. 11(c).

5.2. Experiments on synthetic data

In this set of experiments, proposed estimator with learning mode switching capability, SLWE-ALM, is compared to various estimators in the literature on synthetic data, which are systematically produced by generating random variates from distributions of each regime. The parameter p that characterizes a distribution does not change in the course of the active regime. At the regime switching points however, p switches to the parameter of the new regime.

In each run, algorithms are run on the identical non-stationary sequence and estimates ($\hat{p}$'s) are obtained. The success of tracking

Algorithm 1 Stochastic Learning Weak Estimator with Adaptive Learning Mode (SLWE-ALM)

Initialize $\lambda_{\text{explore}}, \lambda_{\text{exploit}}, w_s, \tau_s, \tau_{ds}, \tau_{dc}$
isStationary $\leftarrow$ FALSE, $i_{dc} \leftarrow 0$, $n_{\text{reg}} \leftarrow 1$, $\lambda \leftarrow \lambda_{\text{explore}}$, $\bar{x}_{\hat{p}}^{\text{old}} \leftarrow 0$, $\bar{x}_{\mathbb{E}\hat{p}}^{\text{old}} \leftarrow 0$
for each observation x **do**
  $\hat{p} \leftarrow SLWE(x, \lambda)$
  $n_{\text{reg}} \leftarrow n_{\text{reg}} + 1$
  $\bar{x}_{\hat{p}}^{\text{new}} \leftarrow (\hat{p}, \bar{x}_{\hat{p}}^{\text{old}}, n_{\text{reg}})$
  $s_{\hat{p}}^2 \leftarrow (\hat{p}, \bar{x}_{\hat{p}}^{\text{old}}, s_{\hat{p}}^2, n_{\text{reg}})$
  if $n_{\text{reg}} < w_s$ **then**
    $\bar{x}_{\mathbb{E}\hat{p}}^{\text{new}} \leftarrow (\bar{x}_{\hat{p}}^{\text{new}}, \bar{x}_{\mathbb{E}\hat{p}}^{\text{old}}, n_{\text{reg}})$
    $s_{\mathbb{E}\hat{p}}^2 \leftarrow (\bar{x}_{\hat{p}}^{\text{new}}, \bar{x}_{\mathbb{E}\hat{p}}^{\text{old}}, s_{\mathbb{E}\hat{p}}^2, n_{\text{reg}})$
  else
    $\bar{x}_{\mathbb{E}\hat{p}}^{\text{new}} \leftarrow (\bar{x}_{\hat{p}}^{\text{new}}, \bar{x}_{\hat{p}}^{\text{replace}}, \bar{x}_{\mathbb{E}\hat{p}}^{\text{old}}, n_{\text{reg}})$
    $s_{\mathbb{E}\hat{p}}^2 \leftarrow (\bar{x}_{\hat{p}}^{\text{new}}, \bar{x}_{\hat{p}}^{\text{replace}}, \bar{x}_{\mathbb{E}\hat{p}}^{\text{old}}, s_{\mathbb{E}\hat{p}}^2, n_{\text{reg}})$
  if isStationary = FALSE **then**
    if $s_{\mathbb{E}\hat{p}}^2 < \tau_{ds}$ **then** ▷ stationarity detected
      $\lambda \leftarrow \lambda_{\text{exploit}}$
      $m_1, m_2 \leftarrow \bar{x}_{\hat{p}}^{\text{new}}, s_{\hat{p}}^2$
      isStationary $\leftarrow$ TRUE
  if isStationary = TRUE **then**
    $m_1, m_2 \leftarrow \bar{x}_{\hat{p}}^{\text{new}}, s_{\hat{p}}^2$
    if $\frac{\hat{p}-m_1}{\sqrt{m_2}} > \tau_{ds}$ **then**
      $i_{dc} = i_{dc} + 1$
    else if $i_{dc} > 0$ **then**
      $i_{dc} = i_{dc} - 1$
    if $i_{dc} \geq \tau_{dc}$ **then** ▷ regime change detected
      $\lambda \leftarrow \lambda_{\text{explore}}$
      reset n_{reg}, i_{dc}, isStationary

Algorithm 2 Growing Window Statistics

function GrowingMean($x_{\text{new}}, \bar{x}_{\text{old}}, N$)
  $\bar{x}_{\text{new}} \leftarrow \bar{x}_{\text{old}} + (x_{\text{new}} - \bar{x}_{\text{old}})/N$
  return $\bar{x}_{\text{new}}$
function GrowingVar($x_{\text{new}}, \bar{x}_{\text{old}}, s_{\text{old}}^2, N$)
  $\bar{x}_{\text{new}} \leftarrow$ GrowingMean$(x_{\text{new}}, \bar{x}_{\text{old}}, N)$
  $s_{\text{new}}^2 \leftarrow [s_{\text{old}}^2(N-1) + (x_{\text{new}} - \bar{x}_{\text{new}})(x_{\text{new}} - \bar{x}_{\text{old}})]/N$
  return s_{new}^2

Algorithm 3 Moving Window Statistics

function MovingMean($x_{\text{new}}, x_{\text{old}}, \bar{x}_{\text{old}}, N$)
  $\bar{x}_{\text{new}} \leftarrow \bar{x}_{\text{old}} + (x_{\text{new}} - x_{\text{old}})/N$
  return $\bar{x}_{\text{new}}$
function MovingVar($x_{\text{new}}, x_{\text{old}}, \bar{x}_{\text{old}}, s_{\text{old}}^2, N$)
  $\bar{x}_{\text{new}} \leftarrow$ MovingMean$(x_{\text{new}}, x_{\text{old}}, \bar{x}_{\text{old}}, N)$
  $s_{\text{new}}^2 \leftarrow s_{\text{old}}^2 + [(x_{\text{new}} - x_{\text{old}})(x_{\text{new}} - \bar{x}_{\text{new}} + x_{\text{old}} - \bar{x}_{\text{old}})]/N$
  return s_{new}^2

p is measured by the mean absolute error(MAE) between p and $\hat{p}$ as follows:

$$\text{MAE} = \frac{1}{N}\sum_{n=1}^{N}\left|p_m - \hat{p}_n\right| \tag{85}$$

where N is the total number of random variates in the non-stationary sequence, p_m is the parameter of the ongoing regime and $\hat{p}_n$ is the nth estimate. This error metric is equal to zero in the case of perfect estimates.

5.2.1. Experiments on binomial random variables

Experiment Setup

In this set of experiments, estimators are tested on binomial random variables. The non-stationarity of the sequence is established by considering sudden, incremental and mixed drift types. For sudden drift, the parameters n, d, $\Delta_p^{\min}$ and $\Delta_p^{\max}$ are used to control the number of regimes, regime duration, minimum and maximum difference in p for each consecutive regime, respectively. This way we can generate sequences with different change frequency and severity characteristics. For incremental drift, we consider the following two versions, (a) linear incremental drift, where p changes to p' over a period of time in a linear fashion which is controlled by the same parameters as in the sudden drift case with the difference of Δ_p's representing the overall change in p, and (b)

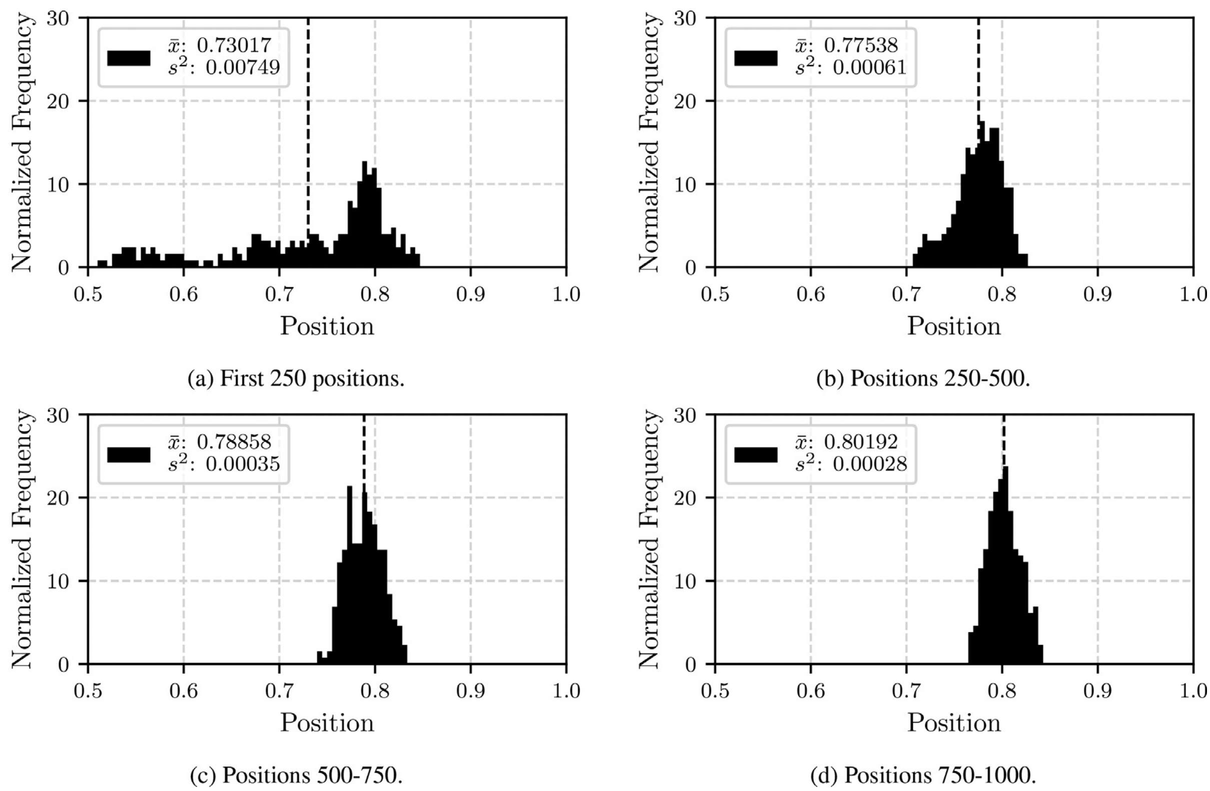


Fig. 9. Histograms of locations of an SLWE random walker for different periods of jumps. In each histogram, horizontal axis is zoomed in for better visibility and vertical dashed lines indicate the sample mean. In this case, $p = 0.8$ and $\lambda = 0.98$, therefore $\mu = 0.8$ and $p = 0.00161$. Initial position of the walker is $x_0 = 0.5$.

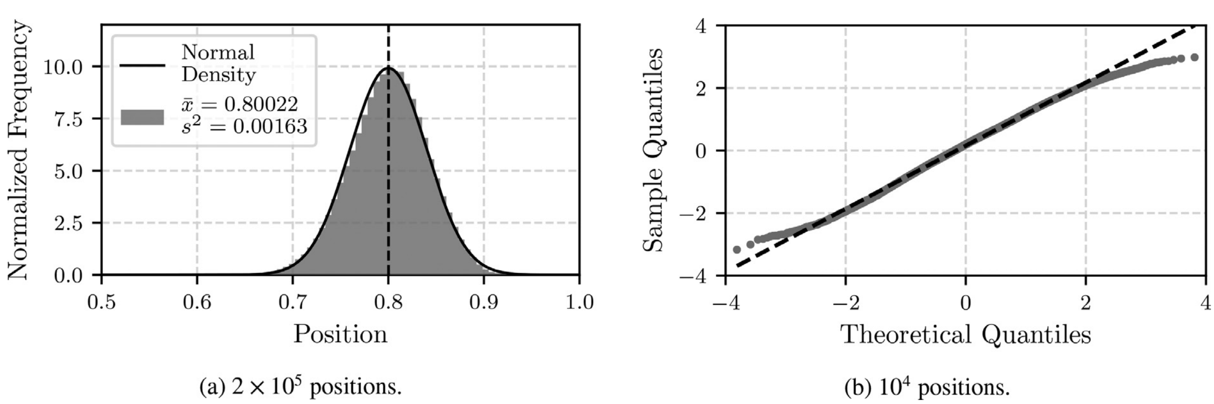


Fig. 10. Large time behavior of estimates shown in Fig. 9. Since $p = 0.8$ and $\lambda = 0.98$, $\mu = 0.8$ and $p = 0.00161$. In (b), Q-Q plot of positions is shown, where $R^2 = 0.997$.

sinusoidal incremental drift, where p values are obtained by quantizing $(\sin(r\pi) + 1)/2$ function, where $r \in [r_0, r_1]$, so that there are n regimes each with a duration of d.

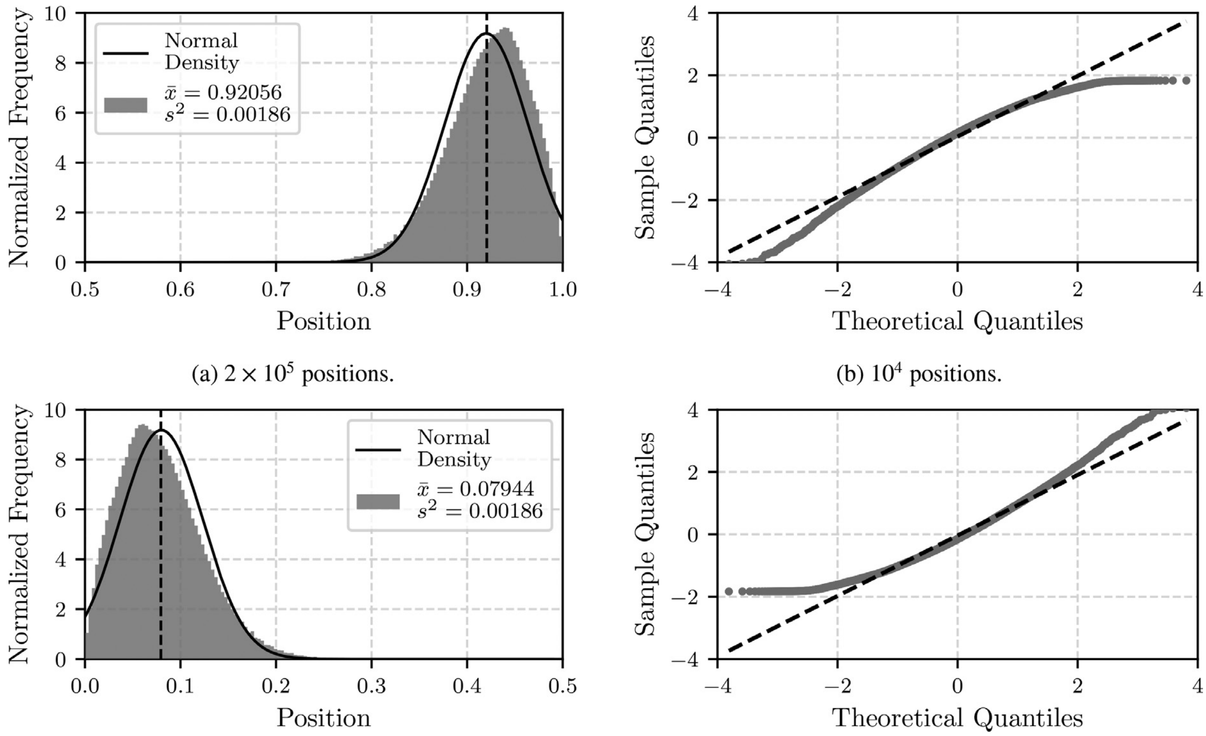
For the mixed case, the aforementioned drift types are combined to obtain a single non-stationary sequence. These drift types are depicted in Fig. 12.

Methods

The methods we compare the proposed method to are SLWE, Algorithm 1 in [28] (which is referred to as HY1 for clarity), ADaptive WINdowing 2 (ADWIN2) [17], HDDM-A (uses average as estimator) and HDDM-W (uses exponentially weighted moving average (EWMA) statistic as estimator) from [41], Page-Hinkley [42] and as a baseline, sample mean estimator (SME).

Parameters

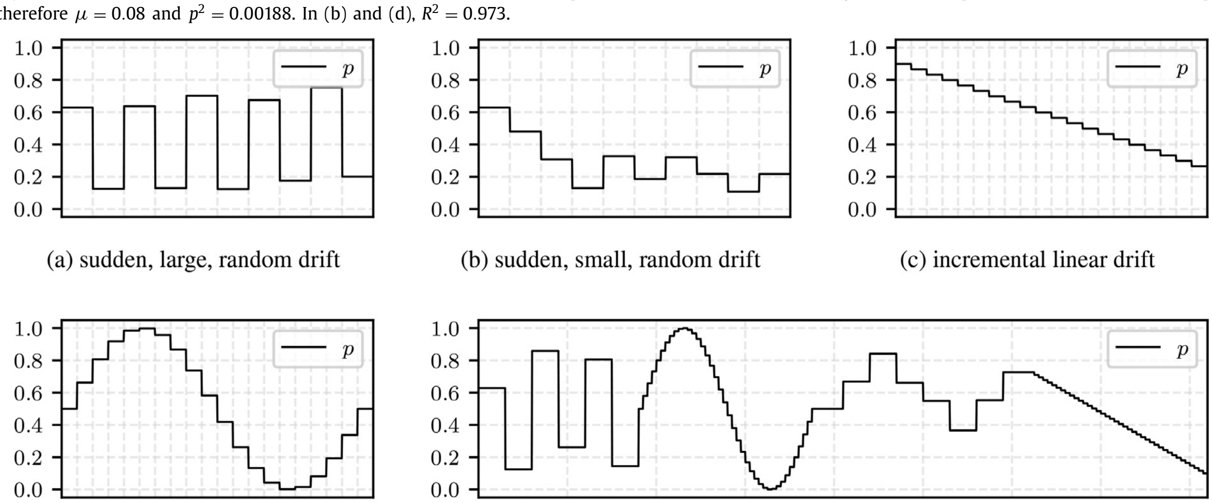
In [3], it is stated that for SLWE $\lambda \approx 0.9$ is empirically found to give good results on binomial random variables. For the non-stationary sequences we run the experiments on, in order to measure the effect of learning mode switches of the proposed method, two instances of SLWE with $\lambda = \{0.96, 0.98\}$, which correspond to exploratory and exploitative modes of the proposed method, are considered. The values for λ for these instances are selected empirically considering performance on sequences with different characteristics. For HY1, in [28] it is pointed out that $\lambda = 0.96$, $\alpha = 10^{-3}$ and $\tilde{n} = 25$ perform well for binomial random variables with both large and small changes. For more smooth changes, authors selected $\lambda = 0.96$, $\alpha = 10^{-2}$ and $\tilde{n} = 1$. For the proposed method, in order to align with HY1 as much as possible, $\lambda_{\text{explore}} = 0.96$, $\lambda_{\text{exploit}} = 0.98$ (same values are used with corresponding SLWE instances), $w_s = 10$, $\tau_s = 5 \times 10^{-3}$, $\tau_{ds} = 3.291$ (which corresponds to $\alpha = 10^{-3}$ of HY1) and $\tau_{dc} = 1$ are used for all types of changes. Setting τ_s to lower values makes the method more selective for the exploratory to exploitative mode switch, which causes more time to be spent in exploration. This could be useful for sequences that might undergo very rapid changes time to time. τ_{ds} and τ_{dc}, on the other hand, control how much and how persistent changes must be to trigger the exploitative to exploratory switch. Despite



(a) 2×10^5 positions. (b) 10^4 positions.

(c) 2×10^5 positions. (d) 10^4 positions.

Fig. 11. Large time behavior of estimates obtained from SLWE. In (a) and (b), $p = 0.92$ and $\lambda = 0.95$, therefore $\mu = 0.92$ and $p^2 = 0.00188$. In (c) and (d), $p = 0.08$ and $\lambda = 0.95$, therefore $\mu = 0.08$ and $p^2 = 0.00188$. In (b) and (d), $R^2 = 0.973$.



(a) sudden, large, random drift (b) sudden, small, random drift (c) incremental linear drift

(d) incremental sinusoidal drift (e) mixed

Fig. 12. Drift types considered in binomial experiments. In all sub-plots, horizontal axis represents observation count.

not being focused on in this study, we expect higher values of τ_{dc} to make the method more robust to observations that are affected by noise. For ADWIN2, δ is set to 0.002 based on empirical results. For the other estimators, parameters are tuned to maximize performance. To eliminate the effect of randomness, each non-stationary sequence is generated 100 times with different seeds for the Pseudorandom Number Generator (PRNG) and average results are reported.

Results

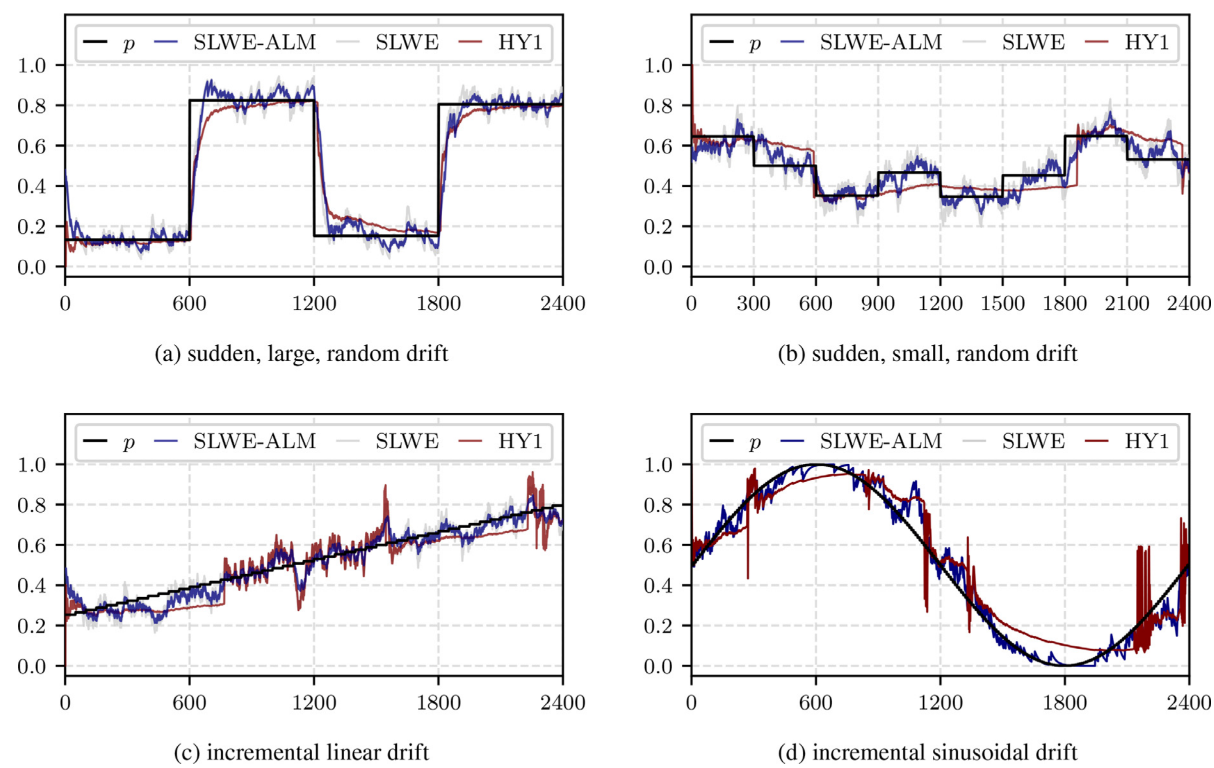
The results are given in Table 1 and an example experiment run for each drift type is shown in Fig. 13. As seen in Table 1, proposed method achieves less (or equal) error than all compared approaches in all drift types.

As expected, exploratory SLWE instance outperforms the exploitative instance in the case of sudden and severe changes. For tracking smaller changes however, exploitative SLWE performed better due to sufficient adaptivity and better stability. Compared to both instances, SLWE-ALM manages to maintain better accuracy, which is an indication of successfully switching between modes when beneficial.

HY1, which is the most closely related approach to the proposed method, is outperformed by SLWE-ALM, with a more significant difference especially on smaller changes. We observe that HY1 is severely impacted by false negatives, as it progressively decreases variance when there is no drift detected. In cases where the target parameter changes with smaller increments, HY1 be-

Table 1
Results obtained from experiments on binomial random variables. Each value is the average MAE of 100 runs. Parameters used to generate non-stationary sequences with large sudden drift are $n = 4$, $d = 600$, $\Delta_p^{\min} = 0.5$, $\Delta_p^{\max} = 1$. For small sudden drift $n = 8$, $d = 300$, $\Delta_p^{\min} = 0.1$, $\Delta_p^{\max} = 0.1$; for linear incremental drift $n = 48$, $d = 50$, $\Delta_p^{\min} = 0.5$, $\Delta_p^{\max} = 1$ and for sinusoidal incremental drift $n = 240$, $d = 10$, $r_0 = 0$, $r_1 = 2$ are used. Mixed case is obtained by combining an instance of each drift type. SLWE (E) and SLWE (T) correspond to exploratory and exploitative SLWE instances, respectively.

Algorithm	Large Sudden	Small Sudden	Linear Incremental	Sinusoidal Incremental	Mixed
SLWE-ALM	0.054	0.046	0.040	0.040	0.052
HY1	0.057	0.066	0.052	0.066	0.070
SLWE (E)	0.055	0.054	0.053	0.040	0.054
SLWE (T)	0.063	0.047	0.042	0.043	0.053
ADWIN2	0.159	0.099	0.123	0.172	0.180
HDDM-A	0.059	0.084	0.093	0.126	0.095
HDDM-W	0.057	0.063	0.063	0.050	0.062
Page-Hinkley	0.204	0.100	0.113	0.264	0.160
Sample Mean	0.274	0.113	0.175	0.291	0.209



(a) sudden, large, random drift
(b) sudden, small, random drift
(c) incremental linear drift
(d) incremental sinusoidal drift

Fig. 13. Estimates obtained from SLWE-ALM, SLWE and HY1. In each sub-figure, vertical axis is the probability and horizontal axis is observation count.

haves like the sample mean estimator (visible in Fig. 13b and 13c), which has limited adaptivity. On the other hand, the proposed method manages to adapt to such changes even they do not trigger the mode switch. This property enables SLWE-ALM to successfully track the target value with the same set of parameters, in cases of both large and small changes.

Among the other methods, we observe that HDDM-A and HDDM-W performed better than ADWIN2, whereas Page-Hinkley and the sample mean estimator (as expected) performed the worst.

5.2.2. Experiments on multinomial random variables

Experiment Setup

In this set of experiments, estimators are tested on multinomial random variables. Non-stationary sequences are generated considering the average absolute difference between each consecutive regime, hence controlling drift severity. For these experiments, large and small sudden drift types are taken into account since more than two oppositely incremental drifting concepts are not possible to coexist in the same sequence. Sequences with sudden drift are generated with parameters n, d, $\Delta_p^{\min}$ and $\Delta_p^{\max}$ which, similar to the binomial experiments, control the number of regimes, regime duration, minimum and maximum average difference in p, respectively.

Methods and Parameters

The methods from the binomial experiments that can be extended to the multinomial case are SLWE and HY2. The values of λ considered in [3] for multinomial experiments are between 0.9 and 0.99. For HY2, authors used $\lambda = 0.95$, $\alpha = 10^{-5}$ and $\tilde{n} = 20$ for both types of changes. With the same purpose of establishing fairness as in the binomial case, we run the proposed method with $\lambda_{\text{explore}} = 0.95$, $\lambda_{\text{exploit}} = 0.98$, $w_s = 10$, $\tau_s = 5 \times 10^{-3}$, $\tau_{ds} = 3.291$ and $\tau_{dc} = 1$ in both drift types. The extension of SLWE-ALM to multinomial case is fairly simple. Since in this case there would be more than one value for the moving variance and deviation from the current model, we simply consider the maximum of these values to be compared with the thresholds. Another approach could be considering the mean of these values.

Results

Table 2
Results obtained from experiments on multinomial ($|\sum| = 4$) random variables. Each value is the average MAE of 100 runs. Parameters used to generate the large sudden drift are $n = 5, d = 600$, $\Delta_p^{\min} = 0.4$, $\Delta_p^{\max} = 1$ and small sudden drift are $n = 5, d = 600$, $\Delta_p^{\min} = 0$, $\Delta_p^{\max} = 0.1$. Mixed case is obtained by combining an instance of each drift type. SLWE (E) and SLWE (T) correspond to exploratory and exploitative SLWE instances, respectively.

Algorithm	Large Sudden	Small Sudden	Mixed
SLWE-ALM	0.038	0.029	0.036
HY2	0.040	0.035	0.038
SLWE (E)	0.046	0.050	0.049
SLWE (T)	0.068	0.031	0.049

As shown in Table 2, proposed method outperforms other estimators in all drift types. We observe that HY2 performs better on multinomial random variables than binomial, which could be caused by the fact that, as discussed in [28], changes are more evident in the multinomial setting.

5.3. Experiments on real world data

In this section, proposed method is experimentally tested on real world data.

5.3.1. Estimation of human activities from accelerometer data

Data Set

In this set of experiments, estimators are run on 3 accelerometer data from `HASC2010corpus`[3]. This data set is formed by Human Activity Sensing Consortium (HASC), and includes labels and sensor data collected from different people while they are performing the following activities: walking, jogging, skipping (happy walking), walking up/down the stairs and staying. The data set includes both (a) segmented data for each activity, performed by different subjects multiple times and (b) activity sequence data, in which different subjects perform various activities in a consecutive manner while the readings from attached accelerometer is constantly logged. Our experiments are performed on the activity sequence data.

This data set is believed to be suitable to test the proposed method considering the assumption that accelerometer readings from different activities are characterized by different distributions. This is also the reason behind the non-stationarity in the activity sequence. Each activity alone is considered to be a stationary process, which is consistent with the regime definition provided before. Similarly, the goal is to measure the effect of changing the learning rate according to stationarity.

Experiment Setup

Initially, sensor data is subjected to a preprocessing phase, which involves downsampling, normalization and discretization. Since the dimensionality of data is already low, a dimensionality reduction step was not needed. The sampling rate of accelerometer is 100Hz for all sequences, and the average activity duration is ≈ 1000 samples long, which corresponds to ≈ 10 seconds for each activity. However, activities vary in duration both between the reappearances in the same sequence and along different sequences. The sensor data for each sequence is downsampled (or decimated) by a factor of 3, that is, only every 3rd sample is kept. The normalization is applied so that each 3-dimensional vector is scaled to unit norm. Discretization is done by utilizing a clustering method and evaluating clustering performance for parameter optimization. In activity sequences, each sensor reading is clustered and represented with the cluster it is assigned to. As a result, a 1-dimensional discrete sequence of cluster labels is obtained. Specifically, in this set of experiments, hierarchical agglomerative clustering with Ward's linkage metric [43] is used. Since the correct labels are not known, clustering performance is measured with a model-based metric, Davies-Bouldin score [44]. While deciding on the number of clusters, the resulting difference between distributions of activities is also taken into consideration. This difference for consecutive regimes can be calculated as:

$$\Delta p = \frac{1}{|\Sigma|} \sum_{i=0}^{|\Sigma|} \left| p_{s_i}^{n} - p_{s_i}^{n+1} \right| \tag{86}$$

where $p_{s_i}^{n}$ is the probability of event s_i at nth regime. As expected, we observed less Δp as the number of clusters decreases. As a result, clustering parameters are empirically selected so that the Davies-Bouldin score is minimized and distributions of regimes are not indistinguishable.

Methods and Parameters

Since the resulting sequence of each activity is in essence realizations of a multinomial random variable, we compare the proposed method to approaches considered in multinomial experiments with the same set of parameters.

Evaluation

Evaluation of the estimates are done by comparing the probability estimate of each possible outcome to the true probability of that outcome. However, unlike the synthetic experiments, true probability distribution of each regime are not known in this case. Therefore, true parameters of the distribution are obtained by analyzing the clustered sequence of observations for each activity in the sequence in an offline manner. Similar to the synthetic experiments, MAE is used to measure how close estimates are to the true parameters.

Results

The results obtained are given in Table 3. Considering the average of 18 sequences from 7 subjects, SLWE-ALM outperforms compared methods.

Table 3
Results of experiments done on HASC2010corpus, which involves 18 activity sequences performed by 7 subjects. The average number of activities in activity sequences is 12. Performance of estimators are measured in terms of MAE, which is calculated considering the true distribution parameters empirically obtained from the data offline.

Sequence ID	SLWE-ALM	HY2	SLWE (E)	SLWE (T)
HASC1001	0.031	0.038	0.035	0.033
HASC1002	0.030	0.038	0.034	0.036
HASC1003	0.033	0.030	0.041	0.035
HASC1004	0.038	0.033	0.046	0.041
HASC1005	0.037	0.039	0.045	0.039
HASC1006	0.032	0.031	0.037	0.036
HASC1007	0.035	0.042	0.041	0.039
HASC1008	0.035	0.034	0.038	0.037
HASC1009	0.033	0.037	0.034	0.033
HASC1010	0.031	0.039	0.033	0.034
HASC1011	0.040	0.038	0.039	0.041
HASC1012	0.036	0.039	0.040	0.040
HASC1013	0.027	0.030	0.028	0.028
HASC1014	0.032	0.040	0.040	0.035
HASC1015	0.038	0.036	0.041	0.042
HASC1016	0.029	0.040	0.034	0.033
HASC1017	0.038	0.043	0.039	0.038
HASC1018	0.043	0.041	0.045	0.044
Avg:	0.034	0.037	0.038	0.037

[3] https://hasc.jp/hc2010/HASC2010corpus/hasc2010corpus-en.html

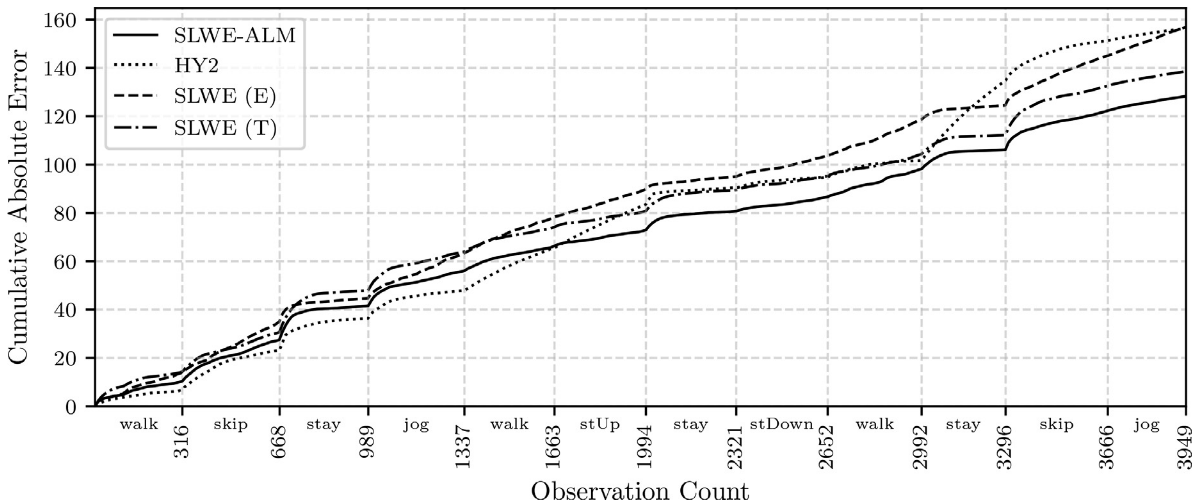


Fig. 14. Cumulative absolute error of estimators on HASC1014 sequence.

In Fig. 14 we show an example sequence where the robustness of the proposed method compared to HY2 is visible. The sudden increase in cumulative absolute error of HY2 starting around 3000 is a result of a false positive. At the same spot, proposed method also did not detect a change hence remained in exploitative mode. However, being still able to adapt to the new regime, SLWE-ALM does not suffer from such cases and achieves less cumulative error than other estimators.

5.3.2. Multi-domain sentiment dataset

Data Set

In this experiment, we consider the problem of tracking domain and sentiment information on streaming text data obtained from customer reviews [45]. The data set contains reviews on 4 types of products (books, DVDs, kitchen and electronics) with 2 sentiment categories (positive and negative) taken from `amazon.com`.

Experiment Setup

By selecting a set of keywords that are possible indicators of each domain and sentiment class, we form a vector $S = \{s_0, s_1, \ldots, s_k\}$ for each review where s_i is the indicator which takes the value of 1 if the corresponding keyword exists in the product review. The non-stationary sequence of these vectors involves product reviews from different classes, therefore there are changes such as `books/positive` → `books/negative` or `dvd/positive` → `electronics/positive`.

Evaluation

Similar to experiments on `HASC2010corpus`, reference multinomial distributions of keywords are calculated offline before the experiment. Tracking performance is again measured with MAE.

Methods and Parameters

Since there are k probability estimates to be obtained, this experiment is also similar to multinomial experiments, therefore proposed method is compared to SLWE and HY2 with recommended parameters.

Results

Results obtained are given in Table 4 and cumulative absolute errors of estimates on an example review sequence is shown in Fig. 15. Consistent with the results of previously reported experiments, SLWE-ALM achieved less MAE than compared methods.

Table 4
Results obtained from experiments on multi-domain sentiment dataset, where the goal is to track the vector of probabilities of selected keywords.

Algorithm	MAE
SLWE-ALM	0.045
HY2	0.051
SLWE (E)	0.051
SLWE (E)	0.053
SLWE (T)	0.058

6. Conclusions and future work

In this paper, we modeled SLWE, a weak estimator that maintains an estimate by a multiplicative update rule similar to $L_{R\text{-}I}$ reinforcement scheme studied in the LA context, as a random walk, and analyzed the asymptotic behavior of estimates. As a result, we have proven that the drift velocity of estimates diminishes over time as the number of estimates grows, which is consistent with the weak convergence property of SLWE. We also showed that the diffusion coefficient of estimates is asymptotically constant, which significantly simplifies the analysis. We discovered that the estimate behavior is consistent with diffusion, a concept that is used to explain many different natural phenomena. Consequently, we derived a model for large-time estimate behavior, which forms the basis of the switching mechanism we proposed for an important trade-off that many estimators face.

SLWE-ALM utilizes the obtained estimate model to track changes in stationarity and prioritizes exploration or exploitation accordingly. A non-stationary to stationary change indicates that the regime is learned, hence exploitation mode is activated. Whereas a stationary to non-stationary change indicates a regime change, which activates exploration mode. With this switching capability, SLWE-ALM manages to combine the advantages of both sides of the trade-off and obtains lower error scores on both synthetic and real world data sequences.

This study forms the mathematical background for the analysis of estimates obtained from SLWE and proposes a simple method for adaptive learning rate control based on learning progress. A possible next step is to work on continuous learning rate control instead of a binary mode switch. We believe that this approach would further improve the tracking capability with more precise control over the exploration/exploitation trade-off.

Furthermore, the detection method described in this paper utilizes a significantly less complex model for estimates than we obtained from the analysis of asymptotic behavior. This simple model

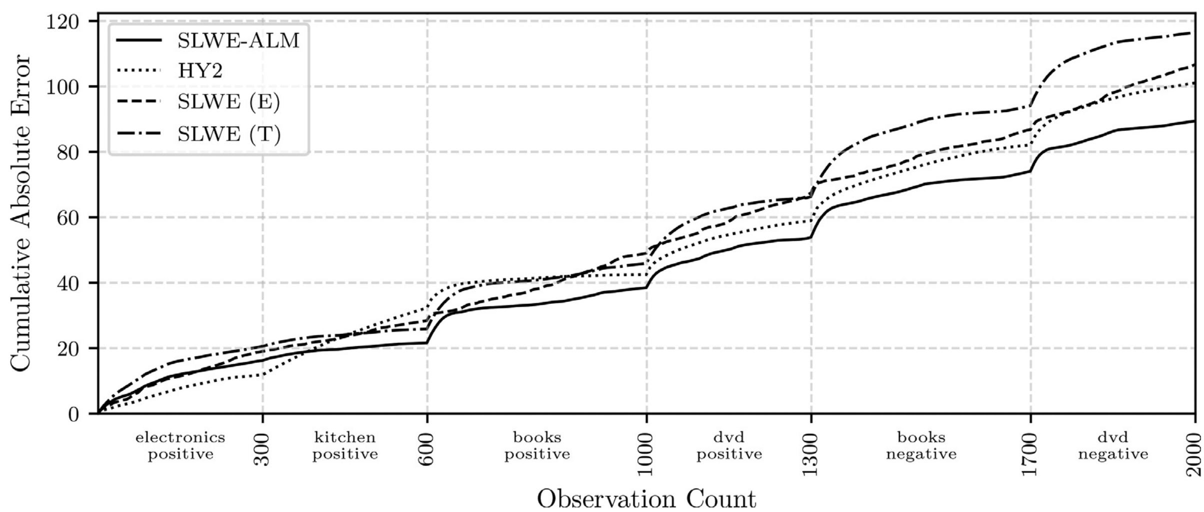


Fig. 15. Cumulative absolute error of estimators on an example sequence of customer reviews with different domain and sentiment properties.

is sufficient for most of the cases, however when the variance (which is determined by the learning rate) is too high or p is close to probability bounds, the third moment should also be taken into account. Therefore, another possible future work is to consider truncated skew-normal distributions by estimating the third moment from estimates.

An important aspect about the problem we are working on, namely estimating distribution parameters from random variables, is noise. Especially when working with real world data, noisy instances may cause estimates to be incorrectly updated, increasing the difference between p and $\hat{p}$. The amplitude of such updates depends on the learning rate. Therefore, despite not covering noise in this study, we expect that the proposed method would improve the robustness to noise due to being in an exploratory mode only when necessary. Although it is not focused on in experimental evaluation, SLWE-ALM uses a parameter (τ_{dc}) that can be adjusted to make the method more selective against the effect of noise.

SLWE-ALM makes use of a change detection method to decide on the amount of exploration needed. In this regard, another promising future extension to this study is to improve stationary to non-stationary change detection by considering the conditionality of observations. In a related previous study [20], we showed that SLWE can be used to estimate nth order Markov dependencies in linear time. By tracking these probabilities, SLWE Change Detection (SCD) algorithm effectively detects regime switching points. Hence, utilization of this approach to control learning rate is expected to further improve estimation performance.

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.

Appendix A. Single-Pass Mean and Variance Calculation

A1. Adding an element (growing window)

Adding a new element x_N to series $[x_0, x_1, x_2, \dots, x_{N-1}]$ results in new series $[x_0, x_1, x_2, \dots, x_N]$. Mean and variance of the series change from $(\bar{x}_0, s_0^2)$ to $(\bar{x}_1, s_1^2)$. The new mean, $\bar{x}_1$, can be calculated as:

$$\bar{x}_1 = \bar{x}_0 + \frac{x_N - \bar{x}_0}{N+1}$$

The new variance, s_1^2, can be calculated as:

$$\begin{aligned}
s_1^2 - s_0^2 &= \frac{1}{N+1}\sum_{i=0}^{N}(x_i - \bar{x}_1)^2 - \frac{1}{N}\sum_{i=0}^{N-1}(x_i - \bar{x}_0)^2 \\
(N+1)s_1^2 - Ns_0^2 &= \sum_{i=0}^{N}(x_i - \bar{x}_1)^2 - \sum_{i=0}^{N-1}(x_i - \bar{x}_0)^2 \\
&= (x_N - \bar{x}_1)^2 + \sum_{i=0}^{N-1}(x_i - \bar{x}_1)^2 - \sum_{i=0}^{N-1}(x_i - \bar{x}_0)^2 \\
&= (x_N - \bar{x}_1)^2 + \sum_{i=0}^{N-1}\left[(x_i - \bar{x}_1)^2 - (x_i - \bar{x}_0)^2\right] \\
&= (x_N - \bar{x}_1)^2 + \sum_{i=0}^{N-1}\left[(2x_i - \bar{x}_0 - \bar{x}_1)(\bar{x}_0 - \bar{x}_1)\right] \\
&= (x_N - \bar{x}_1)^2 + (\bar{x}_0 - \bar{x}_1)\left[\sum_{i=0}^{N-1}(x_i - \bar{x}_0) + \sum_{i=0}^{N-1}(x_i - \bar{x}_1)\right] \\
&= (x_N - \bar{x}_1)^2 + (\bar{x}_0 - \bar{x}_1)\left[\left(\sum_{i=0}^{N-1}(x_i - \bar{x}_1)\right) + (x_N - \bar{x}_1) - (x_N - \bar{x}_1)\right] \\
&= (x_N - \bar{x}_1)^2 + (\bar{x}_0 - \bar{x}_1)\left[\left(\sum_{i=0}^{N}(x_i - \bar{x}_1)\right) - (x_N - \bar{x}_1)\right] \\
&= (x_N - \bar{x}_1)^2 + (\bar{x}_0 - \bar{x}_1)(\bar{x}_1 - x_N) \\
&= (x_N - \bar{x}_1)(x_N - \bar{x}_1 - \bar{x}_0 + \bar{x}_1) \\
&= (x_N - \bar{x}_1)(x_N - \bar{x}_0) \\
s_1^2 &= \frac{Ns_0^2 + (x_N - \bar{x}_1)(x_N - \bar{x}_0)}{N+1}
\end{aligned}$$

A2. Substituting an element (moving window)

Substituting the element x_0 by x_N makes the series $[x_0, x_1, x_2, \dots, x_{N-1}]$ become $[x_1, x_2, x_3, \dots, x_N]$. Mean and variance of the series change from $(\bar{x}_0, s_0^2)$ to $(\bar{x}_1, s_1^2)$. The new mean, $\bar{x}_1$, can be calculated as:

$$\bar{x}_1 = \bar{x}_0 + \frac{x_N - x_0}{N}$$

The new variance, s_1^2, can be calculated as:

$$s_1^2 - s_0^2 = \left[\frac{1}{N}\sum_{i=1}^{N} x_i^2 - \left(\frac{1}{N}\sum_{i=1}^{N} x_i\right)^2\right] - \left[\frac{1}{N}\sum_{i=0}^{N-1} x_i^2 - \left(\frac{1}{N}\sum_{i=0}^{N-1} x_i\right)^2\right]$$

$$= \left(\frac{1}{N}\sum_{i=1}^{N} x_i^2\right) - \bar{x}_1^2 - \left(\frac{1}{N}\sum_{i=0}^{N-1} x_i^2\right) + \bar{x}_0^2$$

$$N\left(s_1^2 - s_0^2\right) = \left(\sum_{i=1}^{N} x_i^2\right) - N\bar{x}_1^2 - \left(\sum_{i=0}^{N-1} x_i^2\right) + N\bar{x}_0^2$$

$$= x_N^2 - x_0^2 - N\left(\bar{x}_1^2 - \bar{x}_0^2\right)$$

$$= (x_N - x_0)(x_N + x_0) - N(\bar{x}_1 - \bar{x}_0)(\bar{x}_1 + \bar{x}_0)$$

$$= (x_N - x_0)(x_N + x_0) - N\left(\frac{x_N - x_0}{N}\right)(\bar{x}_1 + \bar{x}_0)$$

$$= (x_N - x_0)(x_N + x_0) - (x_N - x_0)(\bar{x}_1 + \bar{x}_0)$$

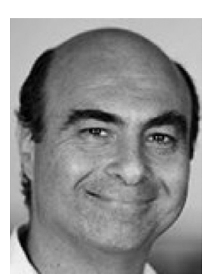
$$= (x_N - x_0)(x_N - \bar{x}_1 + x_0 - \bar{x}_0)$$

$$s_1^2 = s_0^2 + \frac{(x_N - x_0)(x_N - \bar{x}_1 + x_0 - \bar{x}_0)}{N}$$

References

[1] S. Theodoridis, K. Koutroumbas, Pattern recognition, Elsevier, 2006.
[2] R.O. Duda, P.E. Hart, et al., Pattern classification, John Wiley & Sons, 2006.
[3] B.J. Oommen, L. Rueda, Stochastic learning-based weak estimation of multinomial random variables and its applications to pattern recognition in non-stationary environments, Pattern Recognit 39 (3) (2006) 328–341.
[4] K.S. Narendra, M.A. Thathachar, Learning Automata: An Introduction, Prentice-Hall, Inc., 1989.
[5] J.C. Schlimmer, R.H. Granger, Incremental learning from noisy data, Mach Learn 1 (3) (1986) 317–354.
[6] J. Haug, G. Kasneci, Learning parameter distributions to detect concept drift in data streams, in: 2020 25th International Conference on Pattern Recognition (ICPR), IEEE, 2021, pp. 9452–9459.
[7] M. Kárnỳ, On assigning probabilities to new hypotheses, Pattern Recognit Lett 150 (2021) 170–175.
[8] C. Yang, Y.-m. Cheung, J. Ding, K.C. Tan, Concept drift-tolerant transfer learning in dynamic environments, IEEE Trans Neural Netw Learn Syst (2021).
[9] M.G. Rahman, M.Z. Islam, Adaptive decision forest: an incremental machine learning framework, Pattern Recognit 122 (2022) 108345.
[10] T. Lane, C.E. Brodley, Temporal sequence learning and data reduction for anomaly detection, ACM Transactions on Information and System Security (TISSEC) 2 (3) (1999) 295–331.
[11] C.S. Hilas, Designing an expert system for fraud detection in private telecommunications networks, Expert Syst Appl 36 (9) (2009) 11559–11569.
[12] R. Jordaney, K. Sharad, S.K. Dash, Z. Wang, D. Papini, I. Nouretdinov, L. Cavallaro, Transcend: Detecting concept drift in malware classification models, in: 26th {USENIX} Security Symposium ({USENIX} Security 17), 2017, pp. 625–642.
[13] F. Fdez-Riverola, E.L. Iglesias, F. Díaz, J.R. Méndez, J.M. Corchado, Applying lazy learning algorithms to tackle concept drift in spam filtering, Expert Syst Appl 33 (1) (2007) 36–48.
[14] J. Sun, H. Fujita, P. Chen, H. Li, Dynamic financial distress prediction with concept drift based on time weighting combined with adaboost support vector machine ensemble, Knowl Based Syst 120 (2017) 4–14.
[15] M. Salganicoff, Tolerating Concept and Sampling Shift in Lazy Learning Using Prediction Error Context Switching, in: Lazy learning, Springer, 1997, pp. 133–155.
[16] R. Klinkenberg, Learning drifting concepts: example selection vs. example weighting, Intell. Data Anal. 8 (3) (2004) 281–300.
[17] A. Bifet, R. Gavalda, Learning from time-changing data with adaptive windowing, in: Proceedings of the 2007 SIAM International Conference on Data Mining, SIAM, 2007, pp. 443–448, doi:10.1137/1.9781611972771.42.
[18] I. Koychev, in: Gradual forgetting for adaptation to concept drift, Proceedings of ECAI 2000 Workshop on Current Issues in Spatio-Temporal Reasoning, 2000.
[19] I. Žliobaitė, Learning under concept drift: an overview, arXiv preprint arXiv:1010.4784 (2010).
[20] E. Aslanci, K. Coşkun, P. Schüller, B. Tümer, 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), IEEE, 2017, pp. 787–792.
[21] J. Zhan, B.J. Oommen, J. Crisostomo, Anomaly detection in dynamic systems using weak estimators, ACM Transactions on Internet Technology (TOIT) 11 (1) (2011) 3.
[22] H. Chen, T. Su, Y. Zhang, An efficiently adaptive probability estimation method in cabac of hevc using stochastic learning weak estimator, JOURNAL OF INFORMATION 8 (5) (2017) 1149–1164.
[23] H. Tavasoli, B.J. Oommen, A. Yazidi, On utilizing weak estimators to achieve the online classification of data streams, Eng Appl Artif Intell 86 (2019) 11–31.
[24] M. Bhaduri, J. Zhan, C. Chiu, A novel weak estimator for dynamic systems, IEEE Access 5 (2017) 27354–27365.
[25] S. Ishii, W. Yoshida, J. Yoshimoto, Control of exploitation–exploration meta-parameter in reinforcement learning, Neural networks 15 (4–6) (2002) 665–687.
[26] G.A. Carpenter, S. Grossberg, A massively parallel architecture for a self-organizing neural pattern recognition machine, Computer vision, graphics, and image processing 37 (1) (1987) 54–115.
[27] S.W. Wilson, et al., Explore/exploit strategies in autonomy, in: Proc. of the Fourth International Conference on Simulation of Adaptive Behavior: From Animals to Animats, volume 4, 1996, pp. 325–332.
[28] H.L. Hammer, A. Yazidi, Parameter estimation in abruptly changing dynamic environments using stochastic learning weak estimator, Applied Intelligence 48 (11) (2018) 4096–4112, doi:10.1007/s10489-018-1205-3.
[29] F. Xia, J. Liu, H. Nie, Y. Fu, L. Wan, X. Kong, Random walks: a review of algorithms and applications, IEEE Transactions on Emerging Topics in Computational Intelligence 4 (2) (2019) 95–107.
[30] A. Papoulis, S.U. Pillai, Probability, random variables, and stochastic processes, Tata McGraw-Hill Education, 2002.
[31] H.A. Kramers, Brownian motion in a field of force and the diffusion model of chemical reactions, Physica 7 (4) (1940) 284–304.
[32] J. Moyal, Stochastic processes and statistical physics, Journal of the Royal Statistical Society. Series B (Methodological) 11 (2) (1949) 150–210.
[33] R. Pawula, Approximation of the linear boltzmann equation by the fokker–planck equation, Physical review 162 (1) (1967) 186.
[34] H. Risken, Fokker-planck Equation, in: The Fokker-Planck Equation, Springer, 1996, pp. 63–95.
[35] A.D. Fokker, Die mittlere energie rotierender elektrischer dipole im strahlungsfeld, Ann Phys 348 (5) (1914) 810–820.
[36] V. Planck, Über einen satz der statistischen dynamik und seine erweiterung in der quantentheorie, Sitzungberichte der (1917).
[37] A. Einstein, et al., On the motion of small particles suspended in liquids at rest required by the molecular-kinetic theory of heat, Ann Phys 17 (549–560) (1905) 208.
[38] J. Burkardt, The truncated normal distribution, Department of Scientific Computing Website, Florida State University (2014) 1–35.
[39] G.E. Box, G.M. Jenkins, G.C. Reinsel, G.M. Ljung, Time Series Analysis: Forecasting and Control, John Wiley & Sons, 2015.
[40] B. Welford, Note on a method for calculating corrected sums of squares and products, Technometrics 4 (3) (1962) 419–420.
[41] I. Frias-Blanco, J. del Campo-Ávila, G. Ramos-Jimenez, R. Morales-Bueno, A. Ortiz-Diaz, Y. Caballero-Mota, Online and non-parametric drift detection methods based on hoeffding's bounds, IEEE Trans Knowl Data Eng 27 (3) (2014) 810–823, doi:10.1109/TKDE.2014.2345382.
[42] E.S. Page, Continuous inspection schemes, Biometrika 41 (1/2) (1954) 100–115.
[43] J.H. Ward Jr, Hierarchical grouping to optimize an objective function, J Am Stat Assoc 58 (301) (1963) 236–244.
[44] D.L. Davies, D.W. Bouldin, A cluster separation measure, IEEE Trans Pattern Anal Mach Intell (2) (1979) 224–227.
[45] J. Blitzer, M. Dredze, F. Pereira, Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification, in: Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics, 2007, pp. 440–447.

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 and MInD-NET research labs 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.