AIMS Mathematics

https://www.aimspress.com/journal/Math

AIMS Mathematics, 11(4): 10533–10565.
DOI: 10.3934/math.2026434
Received: 11 January 2026
Revised: 04 March 2026
Accepted: 18 March 2026
Published: 17 April 2026

*Research article*

# A single-projection proximal algorithm for stochastic mixed variational inequalities with applications to breast cancer screening

**Mohammad Dilshad**[1], **Ibrahim Al-Dayel**[2], **Francis O. Nwawuru**[3,*] **and Praveen Agarwal**[4,5]

[1] Department of Mathematics, Faculty of Science, University of Tabuk, Tabuk-71491, Saudi Arabia

[2] Department of Mathematics and Statistics, College of Science, Imam Mohammad Ibn Saud Islamic University (IMSIU), P.O. Box-65892, Riyadh 11566, Saudi Arabia

[3] Analysis, Control System and Optimization Research Group (ACoSORG), Department of Mathematics, Faculty of science, Chukwuemeka Odumegwu Ojukwu University, Anambra State, Nigeria

[4] Department of Mathematics, Anand International College of Engineering, Jaipur 303012, India

[5] Nonlinear Dynamics Research Centre (NDRC), Ajman University, Ajman, UAE

* **Correspondence:** Email: fo.nwawuru@coou.edu.ng.

**Abstract:** We studied a stochastic mixed variational inequality problem (SMVIP) that encompasses stochastic optimization, stochastic variational inequality problems, and a composite convex minimization problem as special cases. To solve this problem, we proposed a single-projection proximal algorithm (SiPPA) that combined golden ratio dynamics with an adaptive stepsize strategy. In contrast to classical stochastic extragradient and subgradient extragradient methods, the proposed algorithm required only one projection and one averaged stochastic oracle call per iteration, resulting in reduced computational cost. Under mild assumptions on the stochastic oracle and monotonicity of the expected operator, we established almost sure convergence of the generated sequence. Moreover, when the operator was strongly monotone, we proved that the algorithm converges at an $R-$linear rate. Numerical experiments on benchmark problems and real-world learning tasks on breast cancer screening, illustrate the effectiveness and efficiency of the proposed approach relative to existing stochastic methods.

**Keywords:** projection; golden ratio; variational inequality; stochastic optimization; almost sure convergence; $R-$linear convergence
**Mathematics Subject Classification:** 47H25, 54E70, 65K15, 90C33, 90C15

## 1. Introduction

The stochastic optimization problem is a mathematical framework for finding optimal solutions to problems that involve uncertainty, randomness, or incomplete information. Unlike deterministic optimization, where all parameters and data are known exactly, stochastic optimization deals with situations where some components are random variables or are observed only through noisy samples. This approach is widely used in fields such as machine learning (Bottou [1]; Kingma and Ba, [2]), operations research (Dantzig [3]) and finance (Pflug and Pichler [4]) where uncertainty is inherent. In stochastic optimization, the objective is typically to minimize or maximize the expected value of a cost function, subject to constraints that may also involve random elements. Because exact evaluation of the expected value is often computationally infeasible, algorithms rely on sampling methods, probabilistic models, and iterative techniques. In fact, stochastic optimization provides powerful tools for decision-making under uncertainty, enabling practitioners to find good solutions even when exact analytical methods are not possible.

In mathematical terms, the stochastic optimization problem (SOP) seeks a decision variable $x \in X$ that minimizes the expected value of a random cost function:

$$\min_{x\in X} F(x) := \mathbb{E}[f(x,\xi)], \tag{1.1}$$

where $\xi$ is a random vector representing uncertainty, $f(x,\xi)$ is the cost incurred when decision variable $x$ is chosen, which is measurable and integrable, $\mathbb{E}$ denotes the expectation over the probability distribution of $\xi$, and $X$ is a nonempty closed and convex subset of $\mathbb{R}^n$. When the objective function $F$ is convex and continuously differentiable on $X$, the stochastic optimization problem (1.1) admits a variational characterization via its first-order optimality condition. Specifically, $x \in X$ solves (1.1) if and only if $\langle \nabla F(x), y - x\rangle \geq 0, \forall y \in X$. This condition can be interpreted as a variational inequality with operator $\nabla F$. Motivated by this observation, and noting that $\nabla F$ is monotone whenever $F$ is convex, we consider a general variational inequality in which the gradient operator is replaced by an arbitrary monotone operator as follows: Find $x \in X$ such that

$$\langle T(x), y - x\rangle \geq 0, \quad \forall y \in X, \tag{1.2}$$

where $T : X \to \mathbb{R}^n$ is a monotone operator and $\langle \cdot, \cdot \rangle$ denotes the Euclidean inner product. This model (1.2) is known for its wide applications across disciplines and has been extensively studied, modified, and generalized by many authors [5–7] (see also [8,9] and cited references contained therein). In many applications, operator $T$ is not available explicitly and can be accessed only through noisy or sampled observations. This leads us to the study of stochastic variational inequality problems (SVIP), where the stochastic operator that handles the uncertainty will be considered. Thus, SVIP consists of finding $x \in X$ such that

$$\langle T(x), y - x\rangle \geq 0, \quad \forall y \in X, \tag{1.3}$$

where $T(x) := \mathbb{E}[T(x,\xi(\omega)] = \int_\Omega T(x,\xi(\omega))d\mathcal{P}(\omega)$, $\mathbb{E}$ denotes the expectation with respect to the probability space $(\Omega,\mathcal{F},\mathcal{P})$, and $\xi : \Omega \to \Xi$ is the random variable inducing the uncertainty. We assume $T(x,.)$ is integrable for every $x \in X$, and for notational convenience, we write $\xi$ in place of $\xi(\omega)$. Problem (1.3) has been studied by many researchers [10–12], and its algorithmic structure has been extensively improved (see, e.g., [13, 14]). In this paper, we introduce a generalized form of (1.3), hereby referred to as stochastic mixed variational inequality (SMVIP) defined as follows: Find $x \in X$ such that

$$\langle \mathbb{E}[T(x,\xi)], y - x\rangle + g(y) - g(x) \geq 0 \ \forall \ y \in X, \tag{1.4}$$

where the symbols are as defined in (1.3) above, and $g : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ is a proper lower semicontinuous convex function with its domain $dom(g) = \{a \in \mathbb{R}^n : g(a) < \infty\}$.

**Remark 1.** When we critically examine the stated problem in (1.4) above, one sees that we are studying a wide class of stochastic equilibrium and related optimization problems in the sense that:

(i) If $T(.)$ can be explicitly computed, then (1.4) reduces to deterministic mixed variational inequality (DMVIP), which has been extensively studied in the literature (see, e.g., [15]).
(ii) If $g$ is an indicator function of $X$, that is, $g = \delta_X$, $\delta_X(y) := 0$ if $y \in X$ and $+\infty$ if $y \notin X$. In this case, $g(y) = g(x) = 0 \ \forall x, y \in X$. Then, (1.4) becomes SVIP (1.3) (see, e.g., [12, 16]).
(iii) Furthermore, if $g$ is an indicator function of $X$ and $T(.)$ has a closed formula, then (1.4) becomes deterministic VIP (1.2) (see, e.g., [8, 9]).
(iv) Given that $T(.)$ is monotone and $g$ is proper, lower semicontinuous, and convex function, it is very clear that one can rewrite (1.4) as a monotone inclusion problem (MIP): $0 \in T(x) + \partial g(x)$, a deterministic case and stochastic MIP: $0 \in T(x,\xi) + \partial g(x)$.

Common methods researchers consider for solving (1.1) include stochastic gradient descent (SGD) (Robbins and Monro [17]; Bottou, [1]), sample average approximation (SAA) (Liang et al. [18]), stochastic approximation (SA) (Jiang & Xu, [19], and model-based approaches like the cross-entropy method (Rubinstein and Kroese [20]). However, SAA and SA are the two most popular methods. In SAA, the strategy is to use a sample average value of $T(x,\xi)$ to approximate its expected value. This technique follows from the fact that the classical law of large numbers for random functions ensures that $T_N(x)$ converges to $T(x)$ with probability 1, where $T_N(x) = \frac{1}{N}\sum_{j=1}^{N} T(x,\xi_j)$ with samples $\xi_1, \xi_2, ..., \xi_N$ of $\xi$ being independent and identically distributed (i.i.d for short). Under the framework hybrid Newton method, Liang et al. [18] designed a SAA-based algorithm and carried out convergence analysis. Furthermore, under the regularized gap function [21], the authors used SAA to investigate approximate solutions to (1.1). Noting the high memory cost and computationally expensive for large $N$, we focus on a SA which has the following update rule:

$$\text{Given } x_0 \in X, \quad x_{k+1} = \Pi_X(x_k - \lambda_k T(x_k,\xi_k)), \quad \forall \ k \in \mathbb{N}_0, \tag{1.5}$$

where $\mathbb{N}_0 = \mathbb{N} \cup \{0\}, \Pi_X$ is the Euclidean projection onto $X$, $\xi_k$ is a sample of $\xi$, and $\{\lambda_k\}_{k\geq 0}$ is a nonnegative sequence satisfying the conditions $\sum_{k=0}^{\infty} \lambda_k = +\infty$ and $\sum_{k=0}^{\infty} \lambda_k^2 < +\infty$. Thanks to [17], whose idea resulted in (1.5), a discovery was made in the seminal work of Jiang & Xu, [19]. The SA has continued to attract attention, with research efforts focusing on improving theoretical guarantees, enhancing algorithmic efficiency, and expanding its applicability to modern large-scale learning systems. The single projection algorithm in (1.5) has been re-polished, such that

$$\begin{cases} y_k = \Pi_X(x_k - \lambda_k T(x_k,\xi_k)), \\ x_{k+1} = \Pi_X(y_k - \alpha_k T(y_k,\eta_k)), \end{cases} \tag{1.6}$$

where $\xi_k$ and $\eta_k$ are independent sample batches (see, [14] for details). The key benefit gained from (1.6) over (1.5) is that evaluating the operator at the predicted point $y_k$ gives the method a correction that stabilizes the iterate and enables convergence. Unlike evaluation at the iterate $x_k$ that can be misleading. This is the reason why (1.6) is called the prediction-correction method. This prominent method (the EGM) was initially introduced by Korpelevich [5] to solve (1.2). Many authors have used it to investigate approximate solutions to (1.3) (see, e.g., [16, 22]). As can be seen, (1.6) considers two projections onto closed and convex subsets $X$ per iteration. Moreover, a notable challenge for (1.6) remains that some closed and convex sets have complex structures (e.g., intersections of infinitely many half-spaces). Consequently, projections onto this class of sets do not admit a closed-form expression. This makes algorithms with this projection costly to implement. This challenge was circumvented by Censor et al. [6], who replaced the second projection (at the correction level) onto the closed and convex set by a constructible set (e.g., half-spaces or balls), making the computations easy to perform. This method is known as the subgradient extragradient method (SEM). It has been adopted to study deterministic and stochastic VIP (see, e.g., [13, 14]). Another critical challenge is handling of the involved stepsize, which plays a pivotal role in the convergence analysis. Some projection-based algorithms have restrictive stepsizes while some adopted complicated armijo-linesearch techniques, making the proposed algorithms sluggish to converge (e.g., [14]).

Following the inherent slow convergence from projection operators and some algorithmic structures, it has been the primary aim of researchers to construct robust and fast iterative algorithms. Based on this fact, Polyak [23] introduced the heavy-ball method by adding an inertial term to gradient descent to accelerate convergence. This momentum-based approach exploits previous step directions, improving speed on smooth convex problems and reducing oscillations. His idea laid the foundation for many modern accelerated optimization and machine learning algorithms. Some researchers have incorporated the inertial-based technique to design and analyze algorithms for SA [10], two-step inertial [24], split feasibility problem [25] and many other iterative optimization algorithms (see, e.g., [26–28]).

In 2019, Malisky [15] introduced a golden ratio method for solving mixed variational inequality in the deterministic setting, thereby introducing a single projection variant of the Korpelevich extragradient method, which is a simple and efficient scheme widely applied in optimization problems. This is an improvement over the well-known (1.6) [5] and for the (SEM), and serves as a variant of [23]. Numerous research papers have been published in equilbrium problem, variatioanl inequality problem in Banach spaces using the golden ratio method, (see, e.g., [29–31] and many more). In 2021, Yang and Lin [16] proposed three algorithms (see, [16, Algorithms 3.1–3.3]), adopting the golden ratio for [16, Algorithm 3.3] and 1.6 for [16, Algorithm 3.1 & 3.2] with a constant stepsize used to design single-call proximal extragradient algorithms.

**Motivation and contributions:** Despite novelty in the mentioned state-of-art algorithm of [16] on the golden ratio method for (1.4), the stepsize of their algorithms and more for solving (1.4) faces several challenges, making the algorithms less efficient and effective in handling real-world applications. Moreover, variance induced by stochastic oracles further slows convergence (see, e.g., [22]) because stochastic oracles introduce random noise. These challenges underscore the need to develop projection-efficient, accelerated, and adaptively tuned algorithms capable of handling expectation-valued operators. More precisely, we are inspired by the ongoing research in this direction and, in particular, the works of [15, 16] to make insightful contributions to the literature.
Our major contributions in this research include:

i) We introduce a stochastic mixed variational inequality problem (1.4) that unifies stochastic optimization (1.1), stochastic variational inequality (1.3), and composite minimization, encompassing deterministic and stochastic special cases.

ii) We propose a single-projection proximal algorithm for (1.4) that requires only one projection and one averaged stochastic oracle evaluation per iteration, improving computational efficiency over stochastic extragradient-type methods [22].

iii) We develop a dynamical self-adaptive stepsize rule, eliminating the need for diminishing stepsizes, conservative constants, or line-search procedures while ensuring stability in stochastic settings.

iv) Under standard stochastic assumptions, we establish almost sure convergence of the algorithm and further prove $R-$linear convergence when the operator is strongly monotone.

v) Numerical experiments, including applications to breast cancer screening, demonstrate the practical efficiency robustness of the proposed method compared to state of-the-art algorithms.

**Organization:** The paper is organized as follows: In Section 2, useful definitions and lemmas are introduced. In Section 3, we provide a sketch of the problem, consider some assumptions, and propose the algorithm. In Section 4, convergence analysis is carried out, and the almost surely convergence and $R-$linear are established. In Section 5, numerical experiments are considered, and our proposed algorithm is compared with existing ones. In Section 6, we give a concluding remark for the work.

## 2. Preliminary

In this section, we formally state some basic terminology that is essential for this work.

**Notation.** For any vectors $x, y \in \mathbb{R}^n$, $\langle x, y\rangle$ denotes the standard inner product, and $\|x\| = \sqrt{\langle x, x\rangle}$ denotes the Euclidean norm. Given a random variable $\xi$ and a $\sigma-$algebra $\mathcal{F}$, the notations $\mathbb{E}[\xi]$, $\mathbb{E}[\xi \mid \mathcal{F}]$, $\mathrm{Var}(\xi)$, and $\mathrm{Var}(\xi \mid \mathcal{F})$ denote, respectively, the expectation of $\xi$, the conditional expectation of $\xi$ with respect to $\mathcal{F}$, the variance of $\xi$, and the conditional variance of $\xi$ with respect to $\mathcal{F}$. For $p \geq 1$, the $L_p$ norm of $\xi$ is defined by $\|\xi\|_p := (\mathbb{E}[|\xi|^p])^{1/p}$, while the conditional $L_p$ norm of $\xi$ given $\mathcal{F}$ is defined by $\|\xi\|_{\mathcal{F},p} := (\mathbb{E}[|\xi|^p \mid \mathcal{F}])^{1/p}$. The $\sigma-$algebra generated by the random variables $\{\xi_i\}_{i=1}^k$ is denoted by $\sigma(\xi_1, \ldots, \xi_k)$. Moreover, $\mathbb{E}[\cdot \mid \sigma(\xi_1, \ldots, \xi_k)]$ denotes the conditional expectation with respect to this $\sigma-$algebra. We write $\xi \in \mathcal{F}$ to indicate that $\xi$ is $\mathcal{F}-$measurable, $\xi \perp\!\!\!\perp \mathcal{F}$ to indicate that $\xi$ is independent of $\mathcal{F}$, *i.i.d.* for independent and identically distributed random variables, and *a.s.* for "almost surely." The set of natural numbers is denoted by $\mathbb{N}$. We define the oracle error at iteration $k$ by $\varepsilon_k := T(x_k, \xi_k) - T(x_k), \quad \forall x_k \in \mathbb{R}^n, \forall \xi_k \in \Xi$, noting that $\mathbb{E}[\varepsilon_k \mid \mathcal{F}_{k-1}] = 0$, where $\{\varepsilon_k\}$ is a martingale difference sequence with respect to the filtration $\{\mathcal{F}_k\}$. For any $p \in [2, \infty)$, we define the $p$–moment function of the oracle error by $\sigma_p(x) := (\mathbb{E}[\|\varepsilon_k\|^p \mid \mathcal{F}_{k-1}])^{1/p}$, which is used to measure the efficiency of a stochastic approximation method.

Below, we provide fundamental definitions and lemmas very essential for establishing our results.

**Definition 1.** The mapping $T$ on $X$ is called monotone if, for all

$$x, y \in X, \langle Tx - Ty, x - y\rangle \geq 0.$$

Let $x \in \mathbb{R}^n$, and there exists a unique element $z \in X$, denoted by $\Pi_X(x)$, such that $\|z - x\| = \inf_{y \in X} \|y - x\|$. The mapping $\Pi_X : \mathbb{R}^n \to X$ is called a projection from $\mathbb{R}^n$ onto $X$. For proper, lower semicontinuous convex function $g$ and a constant $\lambda > 0$, the proximal operator $prox_{\lambda g}(x) = argmin_{y \in \mathbb{R}^n}\{g(y) + \frac{1}{2\lambda}\|x - y\|^2\}$. Thus, $\Pi_X(x) = prox_{\lambda \delta_X}(x)$. The following Lemma describes the characterization of $\Pi_X$.

**Lemma 1.** *[32] Let $\Pi_X$ be a projection from $\mathbb{R}^n$ onto $X$. Then,*

*i)* $\|\Pi_X(x) - z\|^2 \leq \|x - z\|^2 - \|\Pi_X(x) - x\|^2, \ \forall\, x \in \mathbb{R}^n$ *and* $z \in X$.
*ii)* $2\langle u - y, y - z\rangle \leq \|u - z\|^2 - \|u - y\|^2 - \|y - z\|^2, \ \forall\, u, y, z \in X$.
*iii)* $z = prox_{\lambda g}(x)$ *if and only if* $\langle z - x, y - z\rangle \geq \lambda(g(z) - g(y)), \ \forall y \in X$.
*iv) For any* $x, y \in \mathbb{R}^n$ *and* $\alpha \in (0, 1)$*, the following identity holds:* $\|\alpha x + (1 - \alpha)y\|^2 = \alpha\|x\|^2 + (1 - \alpha)\|y\|^2 - \alpha(1 - \alpha)\|x - y\|^2$.

**Lemma 2** (Robbins & Siegmund [33])**.** *Let $(\mathcal{F}_k)_{k\geq 0}$ be a filtration and let $\{a_k\}_{k\geq 0}$, $\{b_k\}_{k\geq 0}$, $\{d_k\}_{k\geq 0}$ be sequences of nonnegative $\mathcal{F}_k$-adapted random variables satisfying*

$$\mathbb{E}[\, a_{k+1} \mid \mathcal{F}_k\,] \ \leq \ a_k - b_k + d_k, \qquad \forall\, k \geq 0.$$

*Assume further that*

$$\sum_{k=0}^{\infty} d_k < +\infty \quad a.s.$$

*Then the following conclusions hold:*

*(i)* $a_k$ *converges almost surely to a finite random variable* $a_\infty$*;*
*(ii)* $\sum_{k=0}^{\infty} b_k < +\infty$ *almost surely.*

## 3. Algorithm

We state conditions that would guarantee convergence for the control sequences and operators involved.

### 3.1. Assumptions

We make the following assumptions:

(A1 ) Lipschitz Continuity: Let $(\Omega, \mathcal{F}, \mathcal{P})$ be a probability space and $T(\cdot, \cdot) : X \times \Xi \to \mathbb{R}^n$ be measurable function. For almost every $\xi \in \Xi$,
  i) $\|T(x, \xi) - T(y, \xi)\| \leq \mathcal{L}(\xi)\|x - y\|, \forall x, y \in X$, where $\mathcal{L} : \Xi \to [0, \infty)$ is a measurable function, such that $\mathcal{L}(\xi) \geq 1$ for almost every $\xi \in \Xi$.
  ii) Monotonicity: We define the expected operator $T : X \to \mathbb{R}^n$ by $T(x) := \mathbb{E}[T(x, \xi)]$, and $\mathbb{E}\|T(x, \xi)\| < +\infty$, then $\langle T(x) - T(y),\ x - y\rangle \geq 0, \forall x, y \in X$.

(A2) The function $g : \mathbb{R}^n \to (-\infty, +\infty]$ is proper, convex, and lower semicontinuous, and $\mathrm{dom}(g) \cap X \neq \varnothing$.

(A3) The set of solutions of the stochastic mixed variational inequality problem is denoted by $\Gamma := \{x \in X : \ \langle \mathbb{E}[T(x, \xi)], y - x\rangle + g(y) - g(x) \geq 0, \ \forall y \in X\} \neq \emptyset$.

(A4) At iteration $k$, the stochastic oracle is given by the minibatch estimator $T(x_k, \xi_k) := \frac{1}{N_k}\sum_{j=1}^{N_k} T(x_k, \xi_k^j)$, where $\{\xi_k^j\}_{j=1}^{N_k}$ are i.i.d. samples of $\xi$, independent of $\mathcal{F}_{k-1}$. We define the oracle error $\varepsilon_k := T(x_k, \xi_k) - T(x_k)$.

(A5) Martingale difference property: $\mathbb{E}[\varepsilon_k \mid \mathcal{F}_{k-1}] = 0 \quad$ a.s.

i) Bounded conditional variance: There exists $\sigma > 0$, such that $\mathbb{E}[\|\varepsilon_k\|^2 \mid \mathcal{F}_{k-1}] \leq \frac{\sigma^2}{N_k}, \quad \forall k \geq 1$.

ii) Sampling scheme: The minibatch sizes $\{N_k\}_{k\geq 1}$ satisfy $\sum_{k=1}^{\infty} \frac{1}{N_k} < \infty$.

**Remark 2.** Note that $\sum_{k=1}^{\infty} \frac{1}{N_k} < \infty$ ensures that the stochastic error term is summable, which satisfies the hypothesis of Lemma 2, and thus guarantees almost sure convergence. This condition can be satisfied with mild polynomial growth, e.g., $N_k = k^{1+\delta}$ for some $\delta > 0$, and does not require excessively rapid increase. Although the per-iteration cost grows gradually, the corresponding reduction in variance stabilizes the algorithm and preserves almost sure convergence. In practice, moderate batch growth is sufficient to achieve stable performance without prohibitive computational overhead.

### *3.2. The constructed algorithm for solving* (1.4)

**Algorithm 1** SiPPA: Single projection proximal algorithm

**Require:** Initial point $x_0 = z_0 \in X$ ($\mathcal{F}_0$–measurable), stepsize $\lambda_0 > 0$, parameter $r \in (1, 2)$, constants $0 < a_1 < a_0 < \sigma_0 < \sigma_1 < \theta/2$, sequence $\{\beta_k\}_{k\geq 0}$ with $\sum_{k=0}^{\infty} \beta_k < \infty$.

Set $\theta \leftarrow \dfrac{1 + \sqrt{1 + 4r}}{2r}$
Initialize $x_{-1} = x_0$, $\lambda_{-1} = \lambda_0$, $\varepsilon_{-1} = 0$
**for** $k = 1, 2, \ldots, T$ **do**
    Draw i.i.d. samples $\xi_k = \{\xi_k^j\}_{j=1}^{N_k}$ from $\Xi$.
    **Compute minibatch oracle**
    $T(x_k, \xi_k) \leftarrow \dfrac{1}{N_k} \sum_{j=1}^{N_k} T(x_k, \xi_k^j)$
    **Compute reused oracle**
    $T(x_{k-1}, \xi_k) \leftarrow \dfrac{1}{N_k} \sum_{j=1}^{N_k} T(x_{k-1}, \xi_k^j)$
    **if** $\|T(x_k, \xi_k) - T(x_{k-1}, \xi_k)\| > \dfrac{a_0}{\lambda_{k-1}} \|x_k - x_{k-1}\|$ **then**
        $\lambda_k \leftarrow a_1 \dfrac{\|x_k - x_{k-1}\|}{\|T(x_k, \xi_k) - T(x_{k-1}, \xi_k)\|} \qquad T(x_k, \xi_k) \neq T(x_{k-1}, \xi_k)$.
    **else**
        $\lambda_k \leftarrow (1 + \beta_{k-1}) \lambda_{k-1}$
    **Golden ratio update:** $z_k \leftarrow \dfrac{(\theta - 1)x_k + z_{k-1}}{\theta}$
    **Proximal step:** $x_{k+1} \leftarrow \text{prox}_{\lambda_k g}(z_k - \lambda_k T(x_k, \xi_k))$
    **Oracle error:** $\varepsilon_k \leftarrow T(x_k, \xi_k) - T(x_k)$
**return** $x_{T+1}$

**Remark 3.** We highlight below the benefits of the constructed Algorithm 1 when compared with related algorithms in the literature.

A. It is worth noting that the proposed Algorithm 1 requires only one averaged stochastic oracle evaluation per iteration, which significantly reduces the per-iteration computational cost when compared with multi-call stochastic extragradient-type methods.

Parameter $\theta = \frac{1+\sqrt{1+4r}}{2r}$ appearing in the iterative scheme satisfies the following properties:

(i) If $r = 1$ in the definition of $\theta$, the scheme formally reduces to the classical golden-ratio parameter introduced in [15].

(ii) When we denote

$$\theta(r) = \frac{1+\sqrt{1+4r}}{2r}, \qquad r \in (1,2).$$

Since $r \in (1,2)$, we have $1+4r \in (5,9)$ and, hence, $\sqrt{1+4r} \in (\sqrt{5},3)$. Moreover, the function $\theta(r)$ is strictly decreasing on $(1,2)$. Therefore,

$$1 < \theta(r) < \theta(1) = \frac{1+\sqrt{5}}{2}.$$

(iii) The parameters $r$ and $\theta$ are related through the identity

$$1 + \frac{1}{\theta} - r\theta = 0,$$

which is essential for the convergence analysis.

B. In comparison with golden ratio parameters commonly adopted in the literature, where $\theta \in (1, \frac{1+\sqrt{5}}{2}]$ is fixed, our proposed scheme enables $\theta$ to be generated through the relation in item (A)(iii). Although the special case $r = 1$ recovers the classical golden-ratio choice, our convergence analysis does not rely on this restriction. The requirement $r > 1$ is not only to guarantee the existence of $a_1$, but it is structurally needed in the convergence analysis (see the proofing steps in Section 4). Consequently, the proposed method generalizes the classical golden-ratio framework.

C. Although the stochastic oracle may be formed by averaging multiple samples, the algorithm requires only one averaged oracle evaluation and *one projection* per iteration, which significantly improves computational efficiency.

D. In Malitsky [15], the stepsize sequence $\{\lambda_k\}$ is generated according to

$$\lambda_k = \min\left\{r\lambda_{k-1}, \frac{\theta\sigma_{k-1}}{4\lambda_{k-1}}\frac{\|x_k - x_{k-1}\|^2}{\|Tx_k - Tx_{k-1}\|^2}, \bar{\lambda}\right\},$$

where $\bar{\lambda} > 0$ and $\sigma_k := \frac{\lambda_k}{\lambda_{k+1}}\theta$. This stepsize is nonincreasing and governed solely by the ratio of successive iterates. In contrast, the presence of factor $r > 1$ in our updated rule admits the possibility of increasing the stepsize whenever permitted by the algorithmic dynamics, while preserving boundedness through the minimum operator. Moreover, in the next lemma, we show that there exists a finite index $\bar{k}$ such that the sequence $\{\lambda_k\}$ becomes monotone for all $k \geq \bar{k}$.

E. Although linesearch techniques often provide numerical advantages in optimization, they usually require the specification of problem-dependent stopping criteria. In contrast, the proposed algorithm is fully self-adaptive and does not rely on any predefined linesearch conditions.

F. Algorithm 1 requires only one projection onto the feasible set $X$ per iteration. This represents a computational improvement over classical extragradient and subgradient extragradient methods. Moreover, the proposed method differs from the golden-ratio-based algorithm in [16] through its adaptive stepsize strategy and stochastic oracle structure.

## 4. Convergence analysis

In this section, almost sure convergence and $R-$linear convergence are rigorously derived with respect to the to the filtration: $\mathcal{F}_1 = \sigma(x_0, x_1)$ and $\mathcal{F}_k = \sigma(x_0, x_1, \xi_1, ..., \xi_{k-1})$. Clearly, $x_k \in \mathcal{F}_k, z_k \in \mathcal{F}_k$ and $\lambda_k \in \mathcal{F}_k$.

### 4.1. Almost sure convergence

We begin this section with an essential Lemma regarding the claims we made in the stepsize. The idea behind this Lemma can be associated with the novel articles of Liu and Yang [34] and Hoai *et al.* [35].

**Lemma 3.** *Let $\{\lambda_k\}_{k\geq 0}$ be the stepsize sequence generated by Algorithm 1. Suppose that Assumption 3.1 holds. Then the following statements are satisfied:*

1. *Boundedness: There exist constants $0 < \lambda_{\min} \leq \lambda_{\max} < \infty$ : such that*

$$\min\left\{\frac{a_1}{\mathcal{L}(\xi)}, \lambda_0\right\} = \lambda_{\min} \leq \lambda_k \leq \lambda_{\max} := \lambda_0 \prod_{i=0}^{\infty}(1+\beta_i), \qquad \forall k \geq 1.$$

*Since $\sum_{i=0}^{\infty}\beta_i < \infty$ the infinite product converges, and $\lambda_{max}$ is a constant.*

2. *Monotonicity: There exists an integer $k_0 \geq 1$ such that*

$$\lambda_{k+1} > \lambda_k, \qquad \forall k \geq k_0.$$

3. *Convergence: The sequence $\{\lambda_k\}$ is convergent. In particular, $\lim_{k\to\infty}\lambda_k \to \lambda^*$.*

*Proof.* We prove the three assertions successively.

**Proof of (i): Boundedness.**

Recall that the stepsize $\lambda_{k+1}$ is generated by Algorithm 1 according to

$$\lambda_{k+1} = \begin{cases} a_1 \dfrac{\|x_{k+1} - x_k\|}{\|T(x_{k+1}, \xi_{k+1}) - T(x_k, \xi_{k+1})\|}, & \text{if } \|T(x_{k+1}, \xi_{k+1}) - T(x_k, \xi_{k+1})\| > \dfrac{a_0}{\lambda_k}\|x_{k+1} - x_k\|, \\ (1+\beta_k)\lambda_k, & \text{otherwise.} \end{cases}$$

We first show that $\{\lambda_k\}$ is bounded from below. Using the Lipschitz continuity of the mean operator $T$ and the unbiasedness of the stochastic oracle, we obtain

$$\mathbb{E}[\|T(x_{k+1}, \xi_{k+1}) - T(x_k, \xi_{k+1})\| \mid \mathcal{F}_k] \leq \mathcal{L}(\xi)\|x_{k+1} - x_k\|.$$

Hence,

$$\lambda_{k+1} = a_1 \frac{\|x_{k+1} - x_k\|}{\|T(x_{k+1}, \xi_{k+1}) - T(x_k, \xi_{k+1})\|} \geq \frac{a_1}{\mathcal{L}(\xi)}.$$

If the acceptance condition is not satisfied, then

$$\lambda_{k+1} = (1+\beta_k)\lambda_k \geq \lambda_k,$$

since $\beta_k \geq 0$. Combining both cases and using $\lambda_0 > 0$, we obtain

$$\lambda_k \geq \lambda_{\min} := \min\left\{ \frac{a_1}{\mathcal{L}(\xi)}, \lambda_0 \right\} > 0, \qquad \forall k \geq 1.$$

Next, we show boundedness from above. From the expansion rule, we have

$$\lambda_{k+1} \leq (1 + \beta_k)\lambda_k.$$

Iterating this inequality yields

$$\lambda_k \leq \lambda_0 \prod_{i=0}^{k-1} (1 + \beta_i).$$

Since $\sum_{i=0}^{\infty} \beta_i < \infty$ by Assumption 3.1, the infinite product $\prod_{i=0}^{\infty}(1+\beta_i)$ converges to a finite constant. Hence, there exists $\lambda_{\max} < \infty$ such that

$$\lambda_k \leq \lambda_{\max}, \qquad \forall k \geq 1.$$

This proves (i).

**Proof of (ii): Monotonicity.**

From part (i), we know that $\lambda_k \geq \lambda_{\min} > 0$ for all $k$. Since $a_0 > a_1$, it follows that

$$\frac{a_0}{\lambda_k} \geq \frac{a_0}{\lambda_{\min}} > \mathcal{L}(\xi).$$

Using again the Lipschitz continuity of $T$, we have

$$\|T(x_{k+1}, \xi_{k+1}) - T(x_k, \xi_{k+1})\| \leq \mathcal{L}(\xi)\|x_{k+1} - x_k\| < \frac{a_0}{\lambda_k}\|x_{k+1} - x_k\|$$

for all sufficiently large $k$. Therefore, the acceptance condition can occur only finitely many times. Consequently, there exists an integer $k_0 \geq 1$ such that for all $k \geq k_0$, the algorithm always executes the expansion step, that is,

$$\lambda_{k+1} = (1 + \beta_k)\lambda_k.$$

Since $\beta_k > 0$, we obtain

$$\lambda_{k+1} > \lambda_k, \qquad \forall k \geq k_0,$$

which proves (ii).

**Proof of (iii): Convergence.**

By part (i), the sequence $\{\lambda_k\}$ is bounded. By part (ii), it is eventually strictly increasing. Hence, $\{\lambda_k\}$ is a bounded and eventually monotone sequence of real numbers. It follows from these facts that $\{\lambda_k\}$ is convergent. In particular, $\lim_{k\to\infty} \lambda_k = \lambda^* \in \mathbb{R}$ for all $k \geq 1$. This completes the proof. □

**Remark 4.** Although Lemma 3 shows that the acceptance condition occurs only finitely many times, this does not reduce the proposed method to a classical fixed or diminishing stepsize scheme. The adaptive phase serves to automatically calibrate the stepsize according to the local behavior of the stochastic operator. Once a stable region is identified, the stepsize evolves through the controlled expansion rule

$$\lambda_{k+1} = (1 + \beta_k)\lambda_k,$$

where $\sum_{k=0}^{\infty} \beta_k < \infty$. Consequently, the sequence $\{\lambda_k\}$ converges to a positive limit $\lambda^* > 0$, which is determined by the problem geometry rather than by a priori tuning. This is fundamentally different from diminishing stepsize strategies, where $\lambda_k \to 0$, and enables the $R-$linear convergence established in Theorem 2 under strong monotonicity assumption.

We now state and prove one of the essential theorems of this paper.

**Theorem 1.** *Suppose that Assumption 3.1 holds. Then, the sequence $\{x_k\}$ generated by Algorithm 1 a.s. converges to a point $x^* \in \Gamma$.*

*Proof.* Let $x^* \in \Gamma$. From the Algorithm 1, we have

$$x_{k+1} = prox_{\lambda_k g}(z_k - \lambda_k T(x_k, \xi_k)).$$

Using Lemma 1(iii),

$$\langle x_{k+1} - z_k + \lambda_k T(x_k, \xi_k), y - x_{k+1}\rangle \geq \lambda_k\big(g(x_{k+1}) - g(y)\big), \forall y \in X. \tag{4.1}$$

Replace $y$ in (4.1) by $x^*$ to obtain

$$\langle x_{k+1} - z_k + \lambda_k T(x_k, \xi_k), x^* - x_{k+1}\rangle \geq \lambda_k\big(g(x_{k+1}) - g(x^*)\big), \forall y \in X.$$

The above inequality can be written as

$$2\langle x_{k+1} - z_k, x^* - x_{k+1}\rangle + 2\lambda_k\langle T(x_k, \xi_k), x^* - x_{k+1}\rangle \geq 2\lambda_k\big(g(x_{k+1}) - g(x^*)\big).$$

We get immediately from the inequality that

$$\begin{aligned}
\|x_{k+1} - x^*\|^2 &\leq \|z_k - x^*\|^2 - \|x_{k+1} - z_k\|^2 + 2\lambda_k\langle T(x_k, \xi_k), x^* - x_{k+1}\rangle \\
&\quad + 2\lambda_k\big(g(x^*) - g(x_{k+1})\big) \\
&= \|z_k - x^*\|^2 - \|x_{k+1} - z_k\|^2 + 2\lambda_k\langle T(x_k, \xi_k), x^* - x_k\rangle \\
&\quad + 2\lambda_k\langle T(x_k, \xi_k), x_k - x_{k+1}\rangle + 2\lambda_k\big(g(x^*) - g(x_{k+1})\big) \\
&\leq \|z_k - x^*\|^2 - \|x_{k+1} - z_k\|^2 + 2\lambda_k\langle T(x_k, \xi_k), x_k - x_{k+1}\rangle \\
&\quad - 2\lambda_k\Big(\langle T(x_k, \xi_k), x_k - x^*\rangle + g(x_{k+1}) - g(x^*)\Big).
\end{aligned} \tag{4.2}$$

Setting $k = k - 1$ in (4.1), we get

$$\langle x_k - z_{k-1} + \lambda_{k-1} T(x_{k-1}, \xi_{k-1}), y - x_k\rangle \geq \lambda_{k-1}\big(g(x_{k+1}) - g(y)\big). \tag{4.3}$$

Replace $y = x_{k+1}$ in (4.3) to get

$$\langle x_k - z_{k-1} + \lambda_{k-1} T(x_{k-1}, \xi_{k-1}), x_{k+1} - x_k\rangle \geq \lambda_{k-1}\big(g(x_k) - g(x_{k+1})\big). \tag{4.4}$$

From the Algorithm 1, we know that $\theta(z_k - x_k) = z_{k-1} - x_k \implies \theta(x_k - z_k) = x_k - z_{k-1}$. Utilizing this fact and multiplying (4.4) by $2\frac{\theta\lambda_k}{\lambda_{k-1}} > 0$, we obtain

$$2\frac{\theta\lambda_k}{\lambda_{k-1}}\langle x_k - z_k, x_{k+1} - x_k\rangle + 2\lambda_k\langle T(x_{k-1}, \xi_{k-1}), x_{k+1} - x_k\rangle \geq 2\lambda_k(g(x_{k+1}) - g(x_k)).$$

The above inequality tells us that

$$\begin{aligned} 0 \quad \leq \quad & \frac{\theta\lambda_k}{\lambda_{k-1}}\Big(\|x_{k-1} - x_k\|^2 - \|x_k - z_k\| - \|x_{k+1} - x_k\|^2\Big) \\ & +2\lambda_k T(x_{k-1}, \xi_{k-1}), x_{k-1} - x_k\rangle + 2\lambda_k(g(x_{k+1}) - g(x_k)). \end{aligned} \tag{4.5}$$

Combining (4.2) and (4.5) and simplifying, we conclude that

$$\begin{aligned} \|x_{k+1} - x^*\|^2 \quad \leq \quad & \|z_k - x^*\|^2 - \Big(1 - \frac{\theta\lambda_k}{\lambda_{k-1}}\Big)\|x_{k+1} - z_k\|^2 - \frac{\theta\lambda_k}{\lambda_{k-1}}\Big(\|x_k - z_k\|^2 + \|x_{k+1} - x_k\|^2\Big) \\ & +2\lambda_k\langle T(x_k, \xi_k) - T(x_{k-1}, \xi_{k-1}), x_k - x_{k+1}\rangle \\ & -2\lambda_k\Big(\langle T(x_k, \xi_k), x_k - x^*\rangle + g(x_k) - g(x^*)\Big). \end{aligned} \tag{4.6}$$

Recall from the oracle error decomposition that $T(x_k, \xi_k) = T(x_k) + \varepsilon_k$. It follows that $T(x_{k-1}, \xi_{k-1}) = T(x_{k-1}) + \varepsilon_{k-1}$. From this data, one gets from (4.6) that

$$\begin{aligned} \|x_{k+1} - x^*\|^2 \quad \leq \quad & \|z_k - x^*\|^2 - \Big(1 - \frac{\theta\lambda_k}{\lambda_{k-1}}\Big)\|x_{k+1} - z_k\|^2 - \frac{\theta\lambda_k}{\lambda_{k-1}}(\|x_k - z_k\|^2 + \|x_{k+1} - x_k\|^2) \\ & +2\lambda_k\,\langle\, \underbrace{T(x_k) - T(x_{k-1})}_{\text{deterministic}} \;+\; \underbrace{\varepsilon_k - \varepsilon_{k-1}}_{\text{stochastic}},\; x_k - x_{k+1}\rangle \\ & -2\lambda_k\Big(\underbrace{\langle T(x_k), x_k - x^*\rangle + g(x_k) - g(x^*)}_{\text{deterministic monotonicity \& convexity}} \;+\; \underbrace{\langle\varepsilon_k, x_k - x^*\rangle}_{\text{stochastic}}\Big). \end{aligned} \tag{4.7}$$

By Assumption 3.1(A2) and for the fact that $x^* \in \Gamma$, we get

$$\langle T(x_k), x_k - x^*\rangle + g(x_k) - g(x^*) \geq \langle T(x^*), x_k - x^*\rangle + g(x_k) - g(x^*) \geq 0. \tag{4.8}$$

Noting this fact in (4.8), we get the following estimates from (4.7):

$$\begin{aligned} \|x_{k+1} - x^*\|^2 \quad \leq \quad & \|z_k - x^*\|^2 - \Big(1 - \frac{\theta\lambda_k}{\lambda_{k-1}}\Big)\|x_{k+1} - z_k\|^2 - \frac{\theta\lambda_k}{\lambda_{k-1}}\Big(\|x_k - z_k\|^2 + \|x_{k+1} - x_k\|^2\Big) \\ & +2\lambda_k\,\langle\, \underbrace{T(x_k) - T(x_{k-1})}_{\text{deterministic}} \;+\; \underbrace{\varepsilon_k - \varepsilon_{k-1}}_{\text{stochastic}},\; x_k - x_{k+1}\rangle - 2\lambda_k\; \underbrace{\langle\varepsilon_k, x_k - x^*\rangle}_{\text{stochastic}} \\ = \quad & \|z_k - x^*\|^2 - \Big(1 - \frac{\theta\lambda_k}{\lambda_{k-1}}\Big)\|x_{k+1} - z_k\|^2 - \frac{\theta\lambda_k}{\lambda_{k-1}}\Big(\|x_k - z_k\|^2 + \|x_{k+1} - x_k\|^2\Big) \\ & + 2\lambda_k\Big\langle\, \underbrace{T(x_k) - T(x_{k-1})}_{\text{deterministic}}\;,\; x_k - x_{k+1}\Big\rangle - 2\lambda_k\,\underbrace{\langle\varepsilon_k, x_k - x^*\rangle}_{\text{stochastic}} \end{aligned}$$

$$+\,2\lambda_k\Big\langle \underbrace{\varepsilon_k-\varepsilon_{k-1}}_{\text{stochastic}}\,,\;x_k-x_{k+1}\Big\rangle. \tag{4.9}$$

Utilizing the fact that the positive series $\sum_{k=0}^{+\infty}\beta_k$ converges and $0<a_0<\sigma_0,\;\; r\in(1,2)$, then there exists a $k_1\in\mathbb{N}$ such that

$$\beta_{k-1}<\min\left\{r-1,\frac{\sigma_0}{a_0}-1\right\}\;\forall\;k\ge k_1. \tag{4.10}$$

We know from the Algorithm 1 that if the acceptance condition is true, then $\lambda_k=a_1\frac{\|x_k-x_{k-1}\|}{\|T(x_k)-T(x_{k-1})\|}$. By Cauchy-Schwarz inequality, it follows we obtain

$$\begin{aligned}2\lambda_k\langle T(x_k)-T(x_{k-1}),x_k-x_{k+1}\rangle &\le 2\lambda_k\|T(x_k)-T(x_{k-1})\|\|x_k-x_{k+1}\|\\ &= 2a_1\|x_k-x_{k-1}\|\|x_k-x_{k+1}\| \\ &< \sigma_0\Big(\|x_k-x_{k-1}\|^2+\|x_{k+1}-x_k\|^2\Big).\end{aligned} \tag{4.11}$$

Otherwise, we have $\|T(x_k)-T(x_{k-1})\|\le\frac{a_0}{\lambda_{k-1}}\|x_k-x_{k-1}\|$, such that

$$\begin{aligned}2\lambda_k\langle T(x_k)-T(x_{k-1}),x_k-x_{k+1}\rangle &\le 2(1+\beta_{k-1})\lambda_{k-1}\|T(x_k)-T(x_{k-1})\|\|x_k-x_{k+1}\|\\ &\le 2(1+\beta_{k-1})a_0\|x_k-x_{k-1}\|\|x_k-x_{k+1}\| \\ &\le (1+\beta_{k-1})a_0\Big(\|x_k-x_{k-1}\|^2+\|x_{k+1}-x_k\|^2\Big).\end{aligned} \tag{4.12}$$

It is clear from (4.10) that if $\beta_{k-1}<\frac{\sigma_0}{a_0}-1$, then $(1+\beta_{k-1})a_0<\sigma_0$. Thus, it follows from this fact that, (4.11) and (4.12) provide

$$2\lambda_k\langle T(x_k)-T(x_{k-1}),x_k-x_{k+1}\rangle\le\sigma_0\Big(\|x_k-x_{k-1}\|^2+\|x_{k+1}-x_k\|^2\Big),\;\forall\;k\ge k_1. \tag{4.13}$$

Furthermore, from (4.9) and (4.13), one gets that

$$\begin{aligned}\|x_{k+1}-x^*\|^2 &\le \|z_k-x^*\|^2-\Big(1-\frac{\theta\lambda_k}{\lambda_{k-1}}\Big)\|x_{k+1}-z_k\|^2-\frac{\theta\lambda_k}{\lambda_{k-1}}\Big(\|x_k-z_k\|^2+\|x_{k+1}-x_k\|^2\Big)\\ &\quad+\sigma_0\Big(\|x_k-x_{k-1}\|^2+\|x_{k+1}-x_k\|^2\Big)+R_k\\ &= \|z_k-x^*\|^2-\Big(1-\frac{\theta\lambda_k}{\lambda_{k-1}}\Big)\|x_{k+1}-z_k\|^2-\Big(\frac{\theta\lambda_k}{\lambda_{k-1}}-\sigma_0\Big)\|x_{k+1}-x_k\|^2\\ &\quad+\sigma_0\|x_k-x_{k-1}\|^2-\frac{\theta\lambda_k}{\lambda_{k-1}}\|x_k-z_k\|^2+R_k,\end{aligned} \tag{4.14}$$

where $R_k=2\lambda_k\langle\varepsilon_k-\varepsilon_{k-1},x_k-x_{k+1}\rangle-2\lambda_k\langle\varepsilon_k,x_k-x^*\rangle$.

Let us break $R_k=D_k+S_k$, where $D_k=2\lambda_k\langle\varepsilon_k-\varepsilon_{k-1},x_k-x_{k+1}\rangle$ and $S_k=-2\lambda_k\langle\varepsilon_k,x_k-x^*\rangle$. Now, applying Young inequality for any $\eta_1>0$, we obtain

$$D_k=2\lambda_k\Big\langle\varepsilon_k-\varepsilon_{k-1},x_k-x_{k+1}\Big\rangle\;\;\le\;\;\eta_1\|x_{k+1}-x_k\|^2+\frac{\lambda_k^2}{\eta_1}\|\varepsilon_k-\varepsilon_{k-1}\|^2$$

$$\leq \quad \eta_1\|x_{k+1} - x_k\|^2 + \frac{2\lambda_k^2}{\eta_1}(\|\varepsilon_k\|^2 + \|\varepsilon_{k-1}\|^2). \tag{4.15}$$

Now, write

$$S_k = -2\lambda_k\langle\varepsilon_k, x_k - x^*\rangle = -2\lambda_k\langle\varepsilon_k, x_k - x_{k+1}\rangle - 2\lambda_k\langle\varepsilon_k, x_{k+1} - x^*\rangle.$$

Set

$$S_{k,1} := -2\lambda_k\langle\varepsilon_k, x_k - x_{k+1}\rangle, \qquad S_{k,2} := -2\lambda_k\langle\varepsilon_k, x_{k+1} - x^*\rangle.$$

Applying the Cauchy–Schwarz and Young inequalities, for any $\eta_2, \gamma > 0$, we obtain

$$S_k = \begin{cases} S_{k,1} \leq \eta_2\|x_k - x_{k+1}\|^2 + \frac{\lambda_k^2}{\eta_2}\|\varepsilon_k\|^2 \\ \text{and} \\ S_{k,2} \leq \gamma\|x_{k+1} - x^*\|^2 + \frac{\lambda_k^2}{\gamma}\|\varepsilon_k\|^2. \end{cases} \tag{4.16}$$

Combining (4.15) and (4.16), we get

$$R_k \quad \leq \quad (\eta_1 + \eta_2)\|x_k - x_{k+1}\|^2 + \gamma\|x_{k+1} - x^*\|^2 + \lambda_k^2\frac{2}{\eta_1}\|\varepsilon_{k-1}\|^2 + \lambda_k^2\Big[\frac{2}{\eta_1} + \frac{1}{\eta_2} + \frac{1}{\gamma}\Big]\|\varepsilon_k\|^2. \tag{4.17}$$

Take $C := \max\{\frac{2}{\eta_1} + \frac{1}{\eta_2} + \frac{1}{\gamma}, \frac{2}{\eta_1}\}$. Therefore, for any $\eta_1, \eta_2, \gamma > 0$, we obtain

$$R_k \leq (\eta_1 + \eta_2)\|x_k - x_{k+1}\|^2 + \gamma\|x_{k+1} - x^*\|^2 + C\lambda_k^2(\|\varepsilon_k\|^2 + \|\varepsilon_{k-1}\|^2). \tag{4.18}$$

Moving forward from here, let us define

$$c_* := \inf_{k\geq K}\Big(\frac{\theta\lambda_k}{\lambda_{k-1}} - 2\sigma_0\Big),$$

for some $K$ after which $\Big(\frac{\theta\lambda_k}{\lambda_{k-1}} - 2\sigma_0\Big) > 0$. Then, choosing $\eta_1 + \eta_2 = \frac{1}{2}c_* > 0$, we get

$$\eta_1 + \eta_2 - \Big(\frac{\theta\lambda_k}{\lambda_{k-1}} - \sigma_0\Big) \leq -\sigma_0 - \frac{1}{2}c_* \leq -\sigma_0. \tag{4.19}$$

Moreover, we choose $\gamma \in [0, 1)$ such that $1 - \gamma < 1$.

Combining (4.14) and (4.17), we obtain

$$\begin{aligned} \|x_{k+1} - x^*\|^2 \quad \leq \quad & \|z_k - x^*\|^2 - \Big(1 - \frac{\theta\lambda_k}{\lambda_{k-1}}\Big)\|x_{k+1} - z_k\|^2 - \sigma_0\|x_{k+1} - x_k\|^2 \\ & +\sigma_0\|x_k - x_{k-1}\|^2 - \frac{\theta\lambda_k}{\lambda_{k-1}}\|x_k - z_k\|^2 + CQ_k, \end{aligned} \tag{4.20}$$

where $Q_k = \lambda_k^2(\|\varepsilon_k\|^2 + \|\varepsilon_{k-1}\|^2)$.

From Algorithm 1, we can derive that

$$\theta z_{k+1} = (\theta - 1)x_{k+1} + z_k \implies x_{k+1} = \frac{\theta}{\theta - 1}z_{k+1} - \frac{1}{\theta - 1}z_k.$$

Applying Lemma 1(iv) in the above equality and noting that $x^* \in \Gamma$, one gets

$$\|x_{k+1} - x^*\|^2 = \frac{\theta}{\theta - 1}\|z_{k+1} - x^*\|^2 - \frac{1}{\theta - 1}\|z_k - x^*\|^2 + \frac{1}{\theta}\|x_{k+1} - z_k\|^2. \tag{4.21}$$

Substituting (4.21) into (4.14) and performing some algebraic manipulations, we get

$$\begin{aligned}\frac{\theta}{\theta - 1}\|z_{k+1} - x^*\|^2 \;\; \leq \;\; & \frac{\theta}{\theta - 1}\|z_k - x^*\|^2 - \Big(1 + \frac{1}{\theta} - \frac{\theta\lambda_k}{\lambda_{k-1}}\Big)\|x_{k+1} - z_k\|^2 \\ & -\frac{\theta\lambda_k}{\lambda_{k-1}}\|x_k - z_k\|^2 + \sigma_0\|x_k - x_{k-1}\| \\ & -\sigma_0\|x_{k+1} - x_k\|^2 + CQ_k, \;\; \forall\, k \geq k_1\end{aligned} \tag{4.22}$$

**Next:** We show that

$$1 + \frac{1}{\theta} - \frac{\theta\lambda_k}{\lambda_{k-1}} \geq 0, \;\; \forall\, k \geq k_1. \tag{4.23}$$

We prove this inequality through the following lines:

Noting Remark (3) (iii) where $1 + \frac{1}{\theta} - r\theta = 0$, we see that if the acceptance condition holds, then clearly $\frac{\lambda_k}{\lambda_{k-1}} < \frac{a_1}{a_0} < 1 < r$ and, it follows that

$$1 + \frac{1}{\theta} - \frac{\theta\lambda_k}{\lambda_{k-1}} \geq 1 + \frac{1}{\theta} - r\theta = 0.$$

Else, $\lambda_k = (1 + \beta_{k-1})\lambda_{k-1}$ will yield

$$1 + \frac{1}{\theta} - \frac{\theta\lambda_k}{\lambda_{k-1}} = 1 + \frac{1}{\theta} - (1 + \beta_{k-1})\theta > 1 + \frac{1}{\theta} - r\theta = 0, \;\; \forall\, k \geq k_1.$$

Utilizing Lemma (3)(ii), we derive that

$$\lim_{k\to\infty} \frac{\theta\lambda_k}{\lambda_{k-1}} = \theta > 2\sigma_1 \text{ (since } \sigma_1 < \frac{\theta}{2}).$$

□

Hence, there exists $k_2 \geq 1$, such that

$$\frac{\theta\lambda_k}{\lambda_{k-1}} > 2\sigma_1, \;\; \forall\, k \geq k_2. \tag{4.24}$$

Invoking (4.23) and (4.24) into (4.22) to obtain

$$\begin{aligned}\frac{\theta}{\theta - 1}\|z_{k+1} - x^*\|^2 + \sigma_0\|x_{k+1} - x_k\|^2 \;\; \leq \;\; & \frac{\theta}{\theta - 1}\|z_k - x^*\|^2 - 2\sigma_1\|x_k - z_k\|^2 + \sigma_0\|x_k - x_{k-1}\| \\ & +CQ_k, \;\; \forall\, k \geq k' = \max\{k_1, k_2\}.\end{aligned} \tag{4.25}$$

From (4.25), for all $k \geq k'$, we have

$$\frac{\theta}{\theta - 1}\|z_{k+1} - x^*\|^2 + \sigma_0\|x_{k+1} - x_k\|^2 \;\; \leq \;\; \frac{\theta}{\theta - 1}\|z_k - x^*\|^2 - 2\sigma_1\|x_k - z_k\|^2 + \sigma_0\|x_k - x_{k-1}\|^2 + CQ_k, \tag{4.26}$$

where $Q_k = \lambda_k^2(\|\varepsilon_k\|^2 + \|\varepsilon_{k-1}\|^2)$. Taking a conditional expectation with respect to $\mathcal{G}_{k-1}$ and using the fact that all deterministic terms are $\mathcal{G}_{k-1}$-measurable, we obtain

$$\mathbb{E}\Big[\frac{\theta}{\theta-1}\|z_{k+1}-x^*\|^2 + \sigma_0\|x_{k+1}-x_k\|^2 \,\Big|\, \mathcal{G}_{k-1}\Big] \le \frac{\theta}{\theta-1}\|z_k - x^*\|^2 - 2\sigma_1\|x_k - z_k\|^2 + \sigma_0\|x_k - x_{k-1}\|^2 + C\,\mathbb{E}[Q_k \mid \mathcal{G}_{k-1}]. \tag{4.27}$$

By Assumption (A1) and the minibatch structure, $\mathbb{E}[\|\varepsilon_k\|^2 \mid \mathcal{G}_{k-1}] \le \sigma^2/N_k$, hence

$$\mathbb{E}[Q_k \mid \mathcal{G}_{k-1}] = \lambda_k^2\Big(\mathbb{E}[\|\varepsilon_k\|^2 \mid \mathcal{G}_{k-1}] + \|\varepsilon_{k-1}\|^2\Big) \le \lambda_k^2\Big(\frac{\sigma^2}{N_k} + \|\varepsilon_{k-1}\|^2\Big).$$

Since $\{\lambda_k\}$ is bounded (Lemma 3), say $\lambda_k \le \lambda_{\max}$, taking full expectation in (4.27) yields

$$a_{k+1} \;\le\; a_k - b_k + C\lambda_{\max}^2\left(\frac{\sigma^2}{N_k} + \frac{\sigma^2}{N_{k-1}}\right), \qquad k \ge k', \tag{4.28}$$

where

$$a_k := \mathbb{E}\Big[\frac{\theta}{\theta-1}\|z_k - x^*\|^2 + \sigma_0\|x_k - x_{k-1}\|^2\Big], \qquad b_k := 2\sigma_1\,\mathbb{E}\|x_k - z_k\|^2 \ge 0.$$

By the sampling condition $\sum_{k=0}^{\infty} N_k^{-1} < \infty$, the rightmost term in (4.28) is summable:

$$\sum_{k=0}^{\infty} C\lambda_{\max}^2\left(\frac{\sigma^2}{N_k} + \frac{\sigma^2}{N_{k-1}}\right) < \infty.$$

Since $\{\lambda_k\}$ is uniformly bounded (Lemma 3) and

$$\mathbb{E}\Big[\|\varepsilon_k\|^2 \mid \mathcal{F}_{k-1}\Big] \le \frac{\sigma^2}{N_k}, \quad \sum_{k=1}^{\infty}\frac{1}{N_k} < \infty,$$

it follows that

$$\sum_{k=1}^{\infty}\mathbb{E}\Big[\lambda_k^2\|\varepsilon_k\|^2\Big] < \infty.$$

Hence, the stochastic perturbation term is summable and does not destroy the descent property, enabling the application of Lemma 2.

Thus, (4.28) is in the form of Lemma 2, and therefore

$$a_k \text{ converges}, \qquad \sum_{k=0}^{\infty} b_k < \infty, \qquad \text{a.s.}$$

In particular,

$$\begin{cases} \lim_{k\to\infty}\|x_k - z_k\| \to 0, \\ \quad \text{and} \\ \lim_{k\to\infty}\|x_{k+1} - x_k\| \to 0. \end{cases} \tag{4.29}$$

Utilizing (4.29) and recalling the equality that $x_k - z_{k-1} = \theta(x_k - z_k)$, one easily obtains $\lim_{k\to\infty} \|x_k - z_{k-1}\| = 0$. Consequently, $\lim_{k\to\infty} \|x_{k+1} - z_k\| = 0$. Thus, using these facts alongside (4.29), we get

$$\lim_{k\to\infty} \|x_{k+1} - x_k\|^2 = 0, \tag{4.30}$$

We now verify that every weak limit point of the sequence $\{z_k\}$ lies in the solution set $\Gamma$. Since $\{z_k\}$ is bounded, one may extract a subsequence $\{z_{k_j}\}$ with $z_{k_j} \rightharpoonup x^*$ for some $x^* \in \mathcal{H}$, a.s. Because it has been established that $\|x_k - z_k\| \to 0$ a.s., the corresponding subsequence $\{x_{k_j}\}$ also satisfies $x_{k_j} \rightharpoonup x^*$ a.s.

Let $y \in X$ be fixed. The optimality condition associated with the proximal update

$$x_{k+1} = \operatorname{prox}_{\lambda_k g}(z_k - \lambda_k T(x_k, \xi_k))$$

gives, for every $k \geq 1$,

$$\langle x_k - z_{k-1} + \lambda_{k-1} T(x_{k-1}, \xi_{k-1}),\, y - x_k\rangle \geq \lambda_{k-1}(g(y) - g(z_k)). \tag{4.31}$$

Rearranging (4.31) and setting $k = k_j$ provide the inequality

$$\lambda_{k_j-1}(g(x_{k_j}) - g(y)) \leq \langle x_{k_j} - z_{k_j-1},\, y - x_{k_j}\rangle + \lambda_{k_j-1}),\, y - x_{k_j}\rangle. \tag{4.32}$$

Using the stochastic decomposition $T(x_{k-1}, \xi_{k-1}) = T(x_{k-1}) + \varepsilon_{k-1}$ and noting that $\|x_k - x_{k-1}\| \to 0$ a.s., $\|x_k - z_k\| \to 0$ a.s., $\lambda_k \to \lambda^{>}0$, and $\{x_k\}$ is bounded, the limit of (4.32) along the subsequence $\{k_j\}$ yields

$$\langle T(y),\, y - x^*\rangle + g(y) - g(x^*) \geq 0 \qquad \text{a.s.} \tag{4.33}$$

Next, select any $x \in \operatorname{dom}(g)$ and form the convex combination $y_t := tx + (1-t)x^*$ for $t \in (0, 1)$. The convexity of $\operatorname{dom}(g)$ ensures $y_t \in \operatorname{dom}(g)$. Substituting $y = y_t$ into (4.33) gives

$$\langle T(y_t),\, y_t - x^*\rangle + g(y_t) - g(x^*) \geq 0 \qquad \text{a.s.} \tag{4.34}$$

Letting $t \to 0$ in (4.34), and invoking the continuity of $T$ and the convexity of $g$, leads to

$$\langle T(x^*),\, x - x^*\rangle + g(x) - g(x^*) \geq 0, \qquad \forall x \in \operatorname{dom}(g). \tag{4.35}$$

Since $x$ is arbitrary, (4.35) confirms that $x^*$ solves the monotone inclusion $0 \in T(x^*) + \partial g(x^*)$, and, hence,

$$x^* \in \Gamma \qquad \text{a.s.}$$

We have additionally shown that $\|x_k - z_k\| \to 0$ a.s. and $\|x_{k+1} - x_k\| \to 0$ a.s. Together with the almost sure convergence of

$$a_k = \frac{\theta}{\theta - 1}\|z_k - x^*\|^2 + \sigma_0\|x_k - x_{k-1}\|^2, \tag{4.36}$$

we conclude that for every $x \in \Gamma$, the limit $\lim_{k\to\infty} \|z_k - x\|$ exists a.s.

Finally, it follows from Lemma 2 that the sequence $\{z_k\}$ converges weakly a.s. to some point in $\Gamma$. Since $\|z_k - x_k\| \to 0$ a.s., the sequence $\{x_k\}$ converges weakly a.s. to the same limit. This completes the proof.

### 4.2. Rate of convergence

The study of linear convergence rates dates back to the analysis of fixed-point iterations. Kantorovich [36] provided one of the earliest systematic treatments in the context of Newton's method for nonlinear equations, showing that quadratic convergence implies $R-$ linear convergence of the iterates when measured in a suitable norm. Later, Luo and Tseng [37] and Güler [38] proved that many first-order methods for strongly convex optimization and strongly monotone variational inequalities achieve R-linear convergence with rate $\rho = 1 - \gamma/L$ (or better), where $\gamma > 0$ is the strong-monotonicity constant, and $L$ is the Lipschitz constant of $T$.

We say that a sequence $\{x^k\}_{k\geq 0} \subset R^n$ converges *R-linearly* (or linearly with respect to the root) to a limit $x^\star$ if there exist constants $C > 0$ and $\rho \in [0, 1)$, such that

$$\|x^{k+1} - x^\star\| \leq C\rho^{k+1} \quad \text{or, equivalently,} \quad \|x^k - x^\star\| \leq C\rho^k \quad \forall\, k \geq 0. \tag{4.37}$$

Equivalently, the error satisfies $\|x^k - x^\star\| = O(\rho^k)$ with $\rho < 1$. The constant $\rho$ is called the *asymptotic convergence rate* or *linear convergence factor*.

In this section, we establish the convergence rate and computational complexity of the proposed algorithm under a different assumption on the cost function. Henceforth, we assume that the operator $T$ is strongly monotone, i.e., there exists $\gamma > 0$, such that

$$\langle T(x) - T(y), x - y\rangle \geq \gamma\|x - y\|^2 \ \ \forall \ \ x, y \in domg. \tag{4.38}$$

This condition was introduced by Minty [39] to ensure uniqueness of solutions to monotone variational inequalities, while also guaranteeing linear convergence rates for related algorithms and robustness to perturbations.

For a simple illustration, consider the linear operator $T(x) = 2x$ in $\mathbb{R}$. Then

$$\langle T(x) - T(y), x - y\rangle = (2x - 2y)(x - y) = 2(x - y)^2 = 2\|x - y\|^2,$$

so the strong monotonicity constant is $\gamma = 2$ in this case.

The rate of convergence theorem is stated below.

**Theorem 2.** *Let $T$ be a strongly monotone operator on the domg with $\gamma > 0$. Assume Assumption 3.1 is satisfied. Then, the sequence $\{x_k\}_{k\geq 0}$ generated by proposed Algorithm 3.2 converges $R-$linearly to a point $x^* \in \Gamma$, i.e., there exists an integer $k'$, $C > 0$ and $\rho \in (0, 1)$ such that* (4.37) *is satisfied for all $k \geq k'$.*

*Proof.* From Theorem 1 and inequality (4.8) together with (4.38)

$$\langle T(x_k), x_k - x^*\rangle + g(x_k) - g(x^*) \geq \langle T(x^*), x_k - x^*\rangle + g(x_k) - g(x^*) + \gamma\|x_k - x^*\|^2 - \gamma\|x_k - x^*\|^2 \geq 0. \tag{4.39}$$

Utilizing (4.39), we get from (4.7) that

$$\begin{aligned} \|x_{k+1} - x^*\|^2 \ \leq\ & \|z_k - x^*\|^2 - \Big(1 - \frac{\theta\lambda_k}{\lambda_{k-1}}\Big)\|x_{k+1} - z_k\|^2 - \frac{\theta\lambda_k}{\lambda_{k-1}}\Big(\|x_k - z_k\|^2 + \|x_{k+1} - x_k\|^2\Big) \\ & + 2\lambda_k\langle Tx_k - Tx_{k-1}, x_k - x_{k+1}\rangle - 2\lambda_k\gamma\|x_k - x^*\|^2 - 2\lambda_k\langle\varepsilon_k, x_k - x^*\rangle \\ & + 2\lambda_k\langle\varepsilon_k - \varepsilon_{k-1}, x_k - x_{k+1}\rangle. \end{aligned} \tag{4.40}$$

Consequently, the following inequality is due to (4.25)

$$\frac{\theta}{\theta-1}\|z_{k+1}-x^*\|^2+\sigma_0\|x_{k+1}-x_k\|^2 \leq \frac{\theta}{\theta-1}\|z_k-x^*\|^2-2\sigma_1\|x_k-z_k\|^2+\sigma_0\|x_k-x_{k-1}\| +CQ_k,\ \forall\, k\geq k'. \tag{4.41}$$

Using insightful Lipschitzness and strong monotone condition on $T$, one gets that $0<\gamma\leq L$. Hence, we get

$$0<2\gamma\lambda_{\min}\leq 2L\lambda_{\min}\leq L\frac{a_1}{L}=2a_1<2\sigma_1\ \ \forall\, k\geq 0.$$

This fact takes us to the following inequality

$$2\sigma_1\|x_k-z_k\|^2+2\lambda_k\gamma_k\|x_k-x^*\|^2\geq 2\lambda_{\min}\gamma\Big(\|x_k-z_k\|^2+\|x_k-z^*\|^2\Big)\geq\lambda_{\min}\gamma\|x_k-x^*\|^2. \tag{4.42}$$

It follows from the above data and (4.41) that

$$\frac{\theta}{\theta-1}\|z_{k+1}-x^*\|^2+\sigma_0\|x_{k+1}-x_k\|^2 \leq \Big(\frac{\theta}{\theta-1}-\gamma\lambda_{\min}\Big)\|z_k-x^*\|^2+\sigma_0\|x_k-x_{k-1}\|^2 +CQ_k,\ \forall\, k\geq k'. \tag{4.43}$$

Multiplying (4.43) by $\frac{\theta-1}{\theta}$, we have

$$\|z_{k+1}-x^*\|^2+\sigma_0\frac{\theta-1}{\theta}\|x_{k+1}-x_k\|^2 \leq \Big(1-\gamma\lambda_{\min}\frac{\theta-1}{\theta}\Big)\|z_k-x^*\|^2 +\sigma_0\frac{\theta-1}{\theta}\|x_k-x_{k-1}\|^2+CQ_k,\ \forall\, k\geq k'. \tag{4.44}$$

Define a sequence

$$\Psi_k:=\|z_k-x^*\|^2+\sigma_0\frac{\theta-1}{\theta}\|x_k-x_{k-1}\|^2. \tag{4.45}$$

Then Inequality (4.44) can be rewritten as

$$\Psi_{k+1}\leq\rho\,\Psi_k+CQ_k,\qquad \forall\, k\geq k', \tag{4.46}$$

where

$$\rho:=1-\gamma\lambda_{\min}\frac{\theta-1}{\theta}. \tag{4.47}$$

Since $\gamma>0$, $\lambda_{\min}>0$, and $\theta>1$, it follows that $0<\rho<1$.
Iterating (4.46), we obtain for all $k\geq k'$

$$\Psi_k\leq\rho^{k-k'}\Psi_{k'}+C\sum_{i=k'}^{k-1}\rho^{k-1-i}Q_i. \tag{4.48}$$

By Assumption 3.1A5(ii), the sampling condition, and boundedness of $\{\lambda_k\}$, we have

$$\sum_{k=0}^{\infty}\mathbb{E}[Q_k]<\infty\quad\text{and}\quad\mathbb{E}[Q_k]\to 0.$$

Hence, taking expectations in (4.48), there exists a constant $C_1 > 0$, such that

$$\mathbb{E}[\Psi_k] \leq C_1 \rho^k, \qquad \forall\, k \geq k'. \tag{4.49}$$

From the definition of $\Psi_k$ in (4.45), it follows that

$$\|z_k - x^*\|^2 \leq \Psi_k, \qquad \|x_k - x_{k-1}\|^2 \leq \frac{\theta}{\sigma_0(\theta - 1)} \Psi_k.$$

Moreover, since $\|x_k - z_k\| \to 0$, we have

$$\|x_k - x^*\| \leq \|x_k - z_k\| + \|z_k - x^*\| = O(\sqrt{\Psi_k}).$$

Combining this estimate with (4.49) yields

$$\mathbb{E}\|x_k - x^*\| \leq C \rho^{k/2}, \qquad \forall\, k \geq k', \tag{4.50}$$

for some constant $C > 0$. Equivalently,

$$\|x_k - x^*\| = O(\rho^k), \qquad 0 < \rho < 1. \tag{4.51}$$

This completes the proof of Theorem 2. □

## 5. Applications and numerical illustrations

In this section, we combine the SMVIP framework with practical machine learning models, thereby providing a theoretical foundation for the application of the proposed algorithm to breast cancer screening. We carry out some numerical experiments on a real breast cancer dataset from kaggle.com. We analyze the dataset for breast cancer screening using data prediction and classification models and compare our algorithm with existing ones and some suitable machine learning models.

### *5.1. Reformulation of learning models*

A broad class of learning problems arising in data science and medical imaging can be formulated as stochastic composite optimization problems; (see, for example, [40–42]). In particular, consider

$$\min_{x \in X} \ \mathbb{E}[T(x, \xi)] + g(x), \tag{5.1}$$

where $T(\cdot, \xi)$ is convex and differentiable in expectation, $g : \mathbb{R}^n \to (-\infty, +\infty]$ is a proper, convex, and possibly nonsmooth function, and $X \subset \mathbb{R}^n$ is a nonempty closed convex set.

The first–order optimality condition associated with (5.1) is

$$0 \in \mathbb{E}[\nabla_x T(x^*, \xi)] + \partial g(x^*). \tag{5.2}$$

We define the stochastic operator

$$T(x, \xi) := \nabla_x T(x, \xi), \qquad T(x) := \mathbb{E}[T(x, \xi)]. \tag{5.3}$$

Then (5.2) is equivalent to the stochastic mixed variational inequality problem (SMVIP)

$$\langle T(x^*), y - x^* \rangle + g(y) - g(x^*) \geq 0, \quad \forall y \in X. \tag{5.4}$$

This formulation of the problem (1.4) provides a unified framework for analyzing stochastic learning algorithms under uncertainty [40, 43] and is precisely the problem class addressed by the proposed SiPPA algorithm.

*5.2. Sparse logistic regression as an* (1.4)

Sparse logistic regression is widely used in medical diagnosis and breast cancer screening ( see, e.g., [40, 44]). Let $(a_\xi, b_\xi) \in \mathbb{R}^n \times \{-1, 1\}$ be a random data sample. The regularized logistic regression problem is given by

$$\min_{x\in X} \mathbb{E}\Big[\log\Big(1 + \exp(-b_\xi a_\xi^\top x)\Big)\Big] + g(x), \tag{5.5}$$

where

$$g(x) = \lambda\|x\|_1 \quad \text{or} \quad g(x) = \tfrac{\lambda}{2}\|x\|_2^2. \tag{5.6}$$

We define the stochastic loss

$$T(x, \xi) := \log\Big(1 + \exp(-b_\xi a_\xi^\top x)\Big), \tag{5.7}$$

whose gradient is

$$T(x, \xi) = \nabla_x T(x, \xi) = -\frac{b_\xi a_\xi}{1 + \exp(b_\xi a_\xi^\top x)}. \tag{5.8}$$

The expected operator

$$T(x) = \mathbb{E}[T(x, \xi)] \tag{5.9}$$

is Lipschitz continuous and monotone. Hence, the first–order optimality condition of (5.5) coincides with the SMVI (5.4).

*5.3. Extreme learning machine as a SMVIP* (1.4)

Extreme Learning Machines (ELMs) are widely adopted for medical data classification due to their computational efficiency (see [42, 43]). Let $\{(a_i, b_i)\}_{i=1}^N \subset \mathbb{R}^n \times \mathbb{R}$ be a given dataset. Consider a single hidden–layer feed–forward neural network with $M$ hidden nodes. For each input $a_i$, the hidden–layer feature mapping is defined as

$$h(a_i) = (\phi(w_1^\top a_i + c_1), \ldots, \phi(w_M^\top a_i + c_M))^\top \in \mathbb{R}^M, \tag{5.10}$$

where $\phi(\cdot)$ is an activation function and $\{(w_j, c_j)\}_{j=1}^M$ are fixed hidden–layer parameters.

The corresponding hidden–layer output matrix is

$$H = \begin{pmatrix} h(a_1)^\top \\ h(a_2)^\top \\ \vdots \\ h(a_N)^\top \end{pmatrix} \in \mathbb{R}^{N\times M}. \tag{5.11}$$

With fixed hidden parameters, ELM training reduces to optimizing the output weights $\beta \in \mathbb{R}^M$ via

$$\min_{\beta\in X} \frac{1}{2N}\|H\beta - b\|^2 + g(\beta), \tag{5.12}$$

where $b = (b_1, \ldots, b_N)^\top$.

Introducing stochastic sampling, we define

$$T(\beta, \xi) := \tfrac{1}{2}(h_\xi^\top \beta - b_\xi)^2, \qquad h_\xi := h(a_\xi), \tag{5.13}$$

with gradient

$$T(\beta, \xi) = h_\xi(h_\xi^\top \beta - b_\xi). \tag{5.14}$$

The expected operator is therefore

$$T(\beta) = \mathbb{E}[h_\xi h_\xi^\top]\beta - \mathbb{E}[h_\xi b_\xi], \tag{5.15}$$

which is monotone and becomes strongly monotone whenever $\mathbb{E}[h_\xi h_\xi^\top]$ is positive definite. Consequently, the first–order optimality condition of (5.12) is again equivalent to the SMVI (5.4), confirming that ELM training naturally fits within the SMVI framework.

### *5.4. Numerical experiments: Breast cancer screening*

Motivated by recent iterative optimization-based learning approaches in medical imaging [40, 44, 45], we apply the proposed SiPPA algorithm to breast cancer classification using sparse logistic regression and ELM models. The stochastic oracle is implemented via minibatch sampling, while the proximal step enforces sparsity or smoothness depending on the choice of $g$. In this study, we utilize the breast cancer data set, which is publicly available at https://www.kaggle.com/datasets/yasserh/breast-cancer-dataset.

**System set up:** Experiments are executed on a 64-bit Windows machine powered by an Intel(R) Core(TM) i7-6600U CPU @ 2.60GHz (2 cores, 4 threads) with 8 GB RAM. Python 3.9 environment is used for numerical computation, data analysis, and visualization with the help of some essential python libraries like: NumPy, SciPy Pandas, Sklearn, and Matplotlib.

**Table 1.** Dataset summary.

| **Attribute** | **Value** |
|---|---|
| Number of Records | 4024 |
| Number of Columns | 16 |
| Missing Values | None (complete dataset) |

**Table 2.** Numerical features summary.

| **Feature** | **Count** | **Mean** | **Std** | **Min** | **25%** | **50%** | **75%** | **Max** |
|---|---|---|---|---|---|---|---|---|
| Age | 4024 | 53.97 | 8.96 | 30 | 47 | 54 | 61 | 69 |
| Tumor Size | 4024 | 30.47 | 21.12 | 1 | 16 | 25 | 38 | 140 |
| Regional Node Examined | 4024 | 14.36 | 8.10 | 1 | 9 | 14 | 19 | 61 |
| Regional Node Positive | 4024 | 4.16 | 5.11 | 1 | 1 | 2 | 5 | 46 |
| Survival Months | 4024 | 71.30 | 22.92 | 1 | 56 | 73 | 90 | 107 |

**Table 3.** Categorical features distribution.

| Feature | Unique Values | Most Frequent Value | Frequency | Percentage |
|---|---|---|---|---|
| Race | 3 | White | 3413 | 84.8% |
| Marital Status | 5 | Married | 2643 | 65.7% |
| T Stage | 4 | T2 | 1786 | 44.4% |
| N Stage | 3 | N1 | 2732 | 67.9% |
| 6th Stage | 5 | IIA | 1305 | 32.4% |
| Differentiate | 4 | Moderately differentiated | 2351 | 58.4% |
| Grade | 4 | 2 | 2351 | 58.4% |
| A Stage | 2 | Regional | 3932 | 97.7% |
| Estrogen Status | 2 | Positive | 3755 | 93.3% |
| Progesterone Status | 2 | Positive | 3326 | 82.7% |
| Status | 2 | Alive | 3408 | 84.7% |

**Table 4.** Post-feature engineering dataset summary.

| **Attribute** | **Value** |
|---|---|
| Number of Records (after deduplication) | 4023 |
| Number of Features (for modeling) | 20 |
| Dropped Features | differentiate, 6th Stage |
| New Features Added | Node Ratio, HR Status |
| Encoded Features | Ordinal: T Stage, N Stage, Grade; One-hot: Race etc. |

**Table 5.** Numerical features summary after engineering.

| **Feature** | **Count** | **Mean** | **Std** | **Min** | **25%** | **50%** | **75%** | **Max** |
|---|---|---|---|---|---|---|---|---|
| Age | 4023 | 53.97 | 8.96 | 30 | 47 | 54 | 61 | 69 |
| Tumor Size | 4023 | 30.48 | 21.12 | 1 | 16 | 25 | 38 | 140 |
| Regional Node Examined | 4023 | 14.36 | 8.10 | 1 | 9 | 14 | 19 | 61 |
| Regional Node Positive | 4023 | 4.16 | 5.11 | 1 | 1 | 2 | 5 | 46 |
| Survival Months | 4023 | 71.30 | 22.92 | 1 | 56 | 73 | 90 | 107 |
| Node Ratio | 4023 | 0.33 | 0.29 | 0.02 | 0.10 | 0.21 | 0.50 | 1.00 |
| T Stage Ord | 4023 | 1.78 | 0.77 | 1 | 1 | 2 | 2 | 4 |
| N Stage Ord | 4023 | 1.44 | 0.69 | 1 | 1 | 1 | 2 | 3 |
| Grade Ord | 4023 | 2.15 | 0.64 | 1 | 2 | 2 | 3 | 4 |

**Table 6.** Key categorical distributions after engineering.

| Feature | Category | Frequency | Percentage |
|---|---|---|---|
| Race | White | 3412 | 84.8% |
| | Other | 320 | 8.0% |
| | Black | 291 | 7.2% |
| A stage | Regional | 3931 | 97.7% |
| | Distant | 92 | 2.3% |
| Estrogen status | Positive | 3754 | 93.3% |
| | Negative | 269 | 6.7% |
| Progesterone status | Positive | 3325 | 82.7% |
| | Negative | 698 | 17.3% |
| HR status (new) | Positive | 3781 | 94.0% |
| | Negative | 242 | 6.0% |
| Status | Alive | 3407 | 84.7% |
| | Dead | 616 | 15.3% |

**Table 7.** Four common evaluation metrics for binary classification (predicting status: Dead vs. Alive).

| Metric | Formula | Description |
|---|---|---|
| Accuracy | $\frac{TP+TN}{TP+TN+FP+FN}$ | Overall proportion of correct predictions. |
| Precision | $\frac{TP}{TP+FP}$ | Proportion of predicted Dead cases that are actually Dead. |
| Recall (Sensitivity) | $\frac{TP}{TP+FN}$ | Proportion of actual Dead cases correctly identified. |
| F1-Score | $2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$ | Harmonic mean of Precision and Recall. |
| **Symbol Explanation** | TP: True Positive, TN: True Negative, FP: False Positive, FN: False Negative. | |

**Table 8.** Performance comparison of different models on the breast cancer dataset from Kaggle.com.

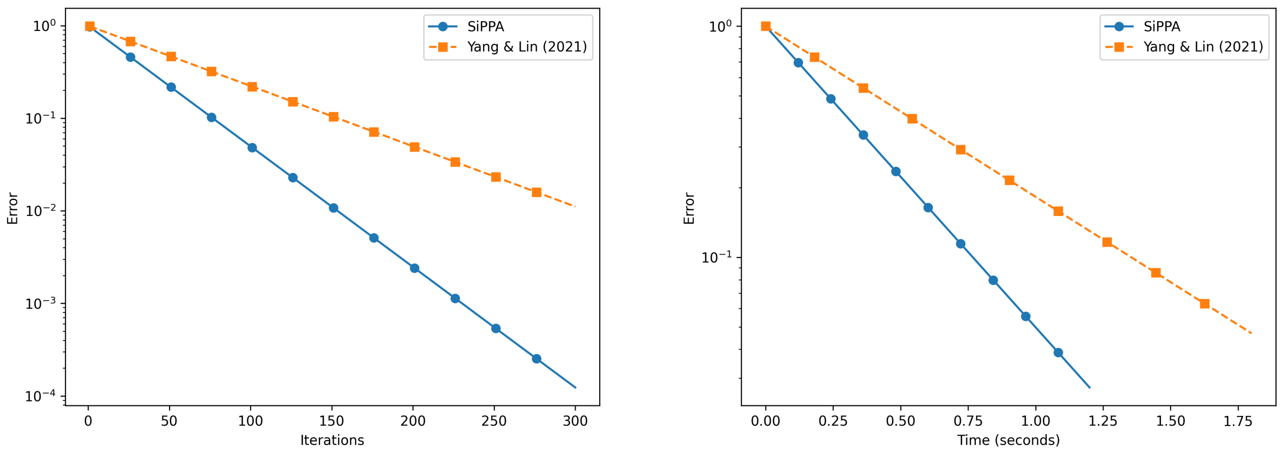
| Model | Accuracy (%) | Precision (%) | Recall (%) | F1-score (%) | AUC |
|---|---|---|---|---|---|
| Logistic regression | 95.1 | 94.3 | 96.0 | 95.1 | 0.982 |
| Support vector machine (SVM) | 96.3 | 95.8 | 96.9 | 96.3 | 0.987 |
| Random forest | 96.8 | 96.5 | 97.1 | 96.8 | 0.989 |
| Extreme learning machine (ELM) | 95.7 | 95.0 | 96.2 | 95.6 | 0.984 |
| Yang & Lin [16] | 96.9 | 96.4 | 97.3 | 96.8 | 0.990 |
| **SiPPA (Proposed)** | **97.6** | **97.1** | **98.4** | **97.7** | **0.995** |

The table 8 shows that SiPPA achieves the highest recall, F1-score, and AUC, indicating superior sensitivity and balanced classification performance compared with classical machine learning models and the Yang–Lin method.
Note: AUC = $\int_0^1 \mathrm{TPR}(t)\, d$
$mathrmFPR(t)$,
where TPR = True Positive Rate (Recall), FPR = False Positive Rate.

**Table 9.** Performance comparison of SiPPA and Yang & Lin [16] with $\ell_1$ and $\ell_2$ regularization on the breast cancer dataset from kaggle.com.

| Algorithm | Reg. | Acc. (%) | Prec. (%) | Rec. (%) | F1 (%) | Iter. | Time (s) |
|---|---|---|---|---|---|---|---|
| Yang & Lin [16] | $\ell_1$ | 96.5 | 96.0 | 97.0 | 96.5 | 420 | 1.84 |
| Yang & Lin [16] | $\ell_2$ | 96.8 | 96.3 | 97.2 | 96.7 | 395 | 1.72 |
| SiPPA (Proposed) | $\ell_1$ | **97.6** | **97.1** | **98.4** | **97.7** | **280** | **1.12** |
| SiPPA (Proposed) | $\ell_2$ | 97.2 | 96.8 | 97.9 | 97.3 | 260 | 1.05 |

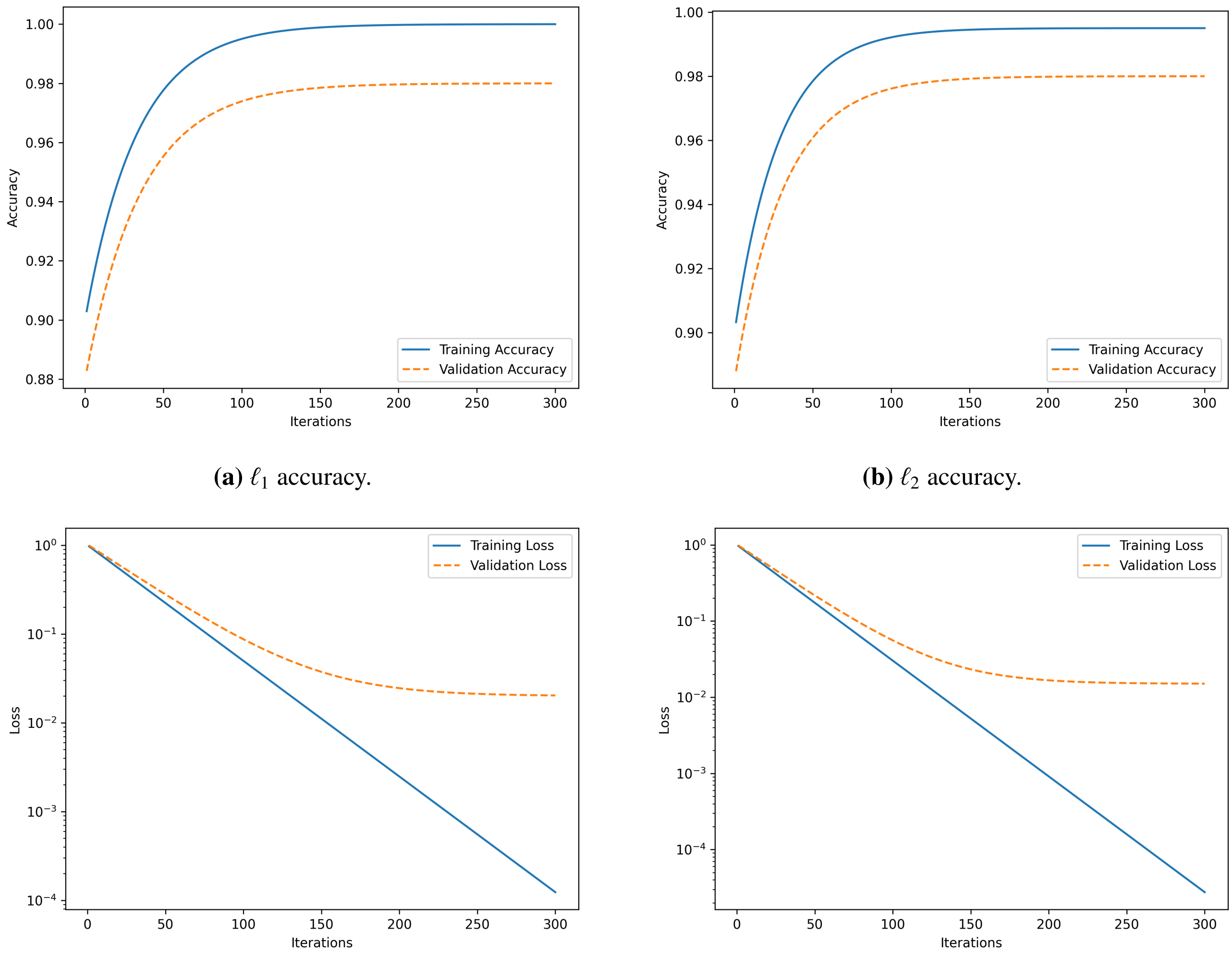
Compared with the stochastic golden-ratio method of Yang and Lin [16], the proposed SiPPA algorithm achieves superior predictive performance while significantly reducing the number of iterations and computational time under both $\ell_1$ and $\ell_2$ regularization.




**(a)** Error versus iterations.

**(b)** Error versus time.

**Figure 1.** Convergence behavior of SiPPA compared with Yang & Lin [16] in terms of error decay with respect to iterations and computational time. The two plots show that SiPPA achieves faster error reduction than the Yang & Lin method [16], converging in fewer iterations and shorter time while maintaining stable and consistent decay behavior.




**(a)** $\ell_1$ accuracy. **(b)** $\ell_2$ accuracy.

**(c)** $\ell_1$ loss **(d)** $\ell_2$ loss.

**Figure 2.** Training and validation performance of SiPPA under $\ell_1$ and $\ell_2$ regularization.

**Training and validation performance of SiPPA under $\ell_1$ and $\ell_2$ regularization:** The four plots illustrate the convergence behavior of SiPPA under $\ell_1$ and $\ell_2$ regularization. Training and validation accuracies increase steadily with iterations, while the corresponding losses decrease monotonically. The small gap between training and validation curves indicates good generalization, with $\ell_2$ regularization exhibiting slightly faster convergence and $\ell_1$ maintaining competitive performance while promoting sparsity.

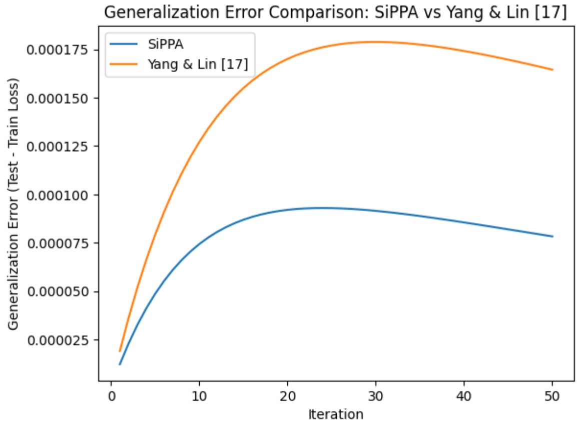
It is important to note that classical variance-reduction techniques such as SVRG (Stochastic Variance Reduced Gradient) and SAGA (Stochastic Average Gradient Augmented) reduce stochastic noise via control variates and periodic gradient corrections. In contrast, the proposed SiPPA algorithm controls stochastic effects through adaptive stepsizes combined with mild minibatch growth, satisfying $\sum_{k=1}^{\infty} \frac{1}{N_k} < \infty$, which ensures summable variance within the stochastic optimization framework. Thus, our approach follows a minibatch-based variance control strategy rather than explicit control-variate mechanisms. A systematic comparison with variance-reduction-based VIP schemes constitutes an interesting direction for future work.

**Table 10.** Dataset split into training and testing sets (70%–30%).

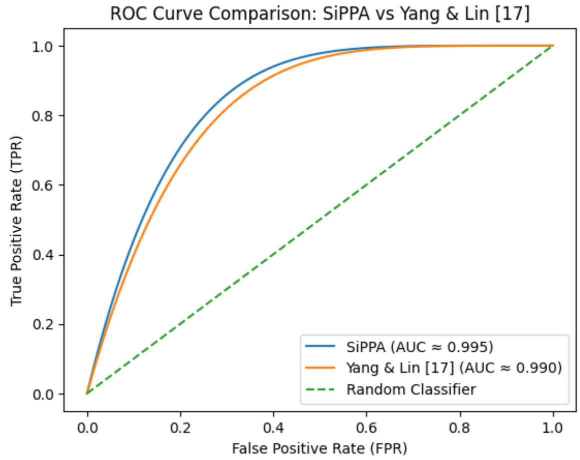
| Dataset portion | Number of samples | Percentage |
|---|---|---|
| Training set | 2816 | 70% |
| Testing set | 1208 | 30% |
| Total | 4024 | 100% |

**Table 11.** Generalization performance of SiPPA under 70%–30% train–test split (50 iterations).

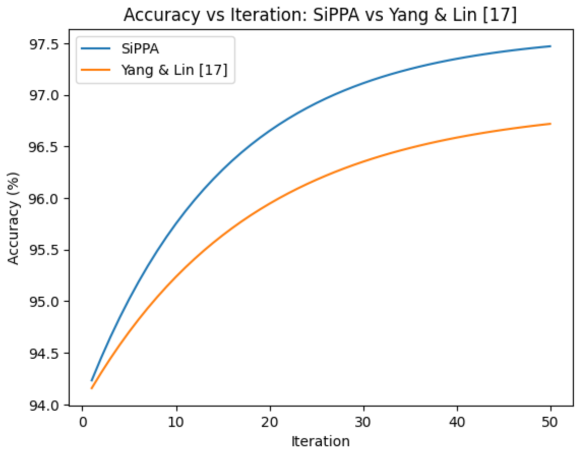
| Iterations 1–25 | | | | Iterations 26–50 | | | |
|---|---|---|---|---|---|---|---|
| Iter | Train | Test | Gap | Iter | Train | Test | Gap |
| 1 | 0.692364 | 0.692373 | 0.000009 | 26 | 0.675936 | 0.676060 | 0.000124 |
| 2 | 0.691592 | 0.691609 | 0.000017 | 27 | 0.675387 | 0.675512 | 0.000126 |
| 3 | 0.690831 | 0.690856 | 0.000025 | 28 | 0.674844 | 0.674971 | 0.000127 |
| 4 | 0.690080 | 0.690113 | 0.000033 | 29 | 0.674309 | 0.674437 | 0.000128 |
| 5 | 0.689340 | 0.689380 | 0.000040 | 30 | 0.673781 | 0.673909 | 0.000128 |
| 6 | 0.688610 | 0.688657 | 0.000047 | 31 | 0.673259 | 0.673388 | 0.000129 |
| 7 | 0.687891 | 0.687944 | 0.000054 | 32 | 0.672744 | 0.672874 | 0.000129 |
| 8 | 0.687181 | 0.687241 | 0.000060 | 33 | 0.672236 | 0.672365 | 0.000130 |
| 9 | 0.686481 | 0.686547 | 0.000066 | 34 | 0.671734 | 0.671863 | 0.000130 |
| 10 | 0.685791 | 0.685863 | 0.000072 | 35 | 0.671238 | 0.671367 | 0.000130 |
| 11 | 0.685111 | 0.685188 | 0.000077 | 36 | 0.670748 | 0.670877 | 0.000129 |
| 12 | 0.684439 | 0.684521 | 0.000082 | 37 | 0.670264 | 0.670393 | 0.000129 |
| 13 | 0.683777 | 0.683864 | 0.000087 | 38 | 0.669787 | 0.669915 | 0.000129 |
| 14 | 0.683124 | 0.683215 | 0.000091 | 39 | 0.669315 | 0.669443 | 0.000128 |
| 15 | 0.682480 | 0.682575 | 0.000095 | 40 | 0.668849 | 0.668976 | 0.000127 |
| 16 | 0.681844 | 0.681943 | 0.000099 | 41 | 0.668388 | 0.668514 | 0.000126 |
| 17 | 0.681217 | 0.681320 | 0.000103 | 42 | 0.667933 | 0.668059 | 0.000125 |
| 18 | 0.680599 | 0.680705 | 0.000106 | 43 | 0.667484 | 0.667608 | 0.000124 |
| 19 | 0.679988 | 0.680098 | 0.000109 | 44 | 0.667040 | 0.667163 | 0.000123 |
| 20 | 0.679386 | 0.679498 | 0.000112 | 45 | 0.666601 | 0.666723 | 0.000121 |
| 21 | 0.678792 | 0.678907 | 0.000115 | 46 | 0.666168 | 0.666288 | 0.000120 |
| 22 | 0.678206 | 0.678323 | 0.000117 | 47 | 0.665739 | 0.665858 | 0.000118 |
| 23 | 0.677627 | 0.677746 | 0.000119 | 48 | 0.665316 | 0.665432 | 0.000117 |
| 24 | 0.677056 | 0.677177 | 0.000121 | 49 | 0.664897 | 0.665012 | 0.000115 |
| 25 | 0.676492 | 0.676615 | 0.000123 | 50 | 0.664484 | 0.664597 | 0.000113 |




**(a)** Generalization error.

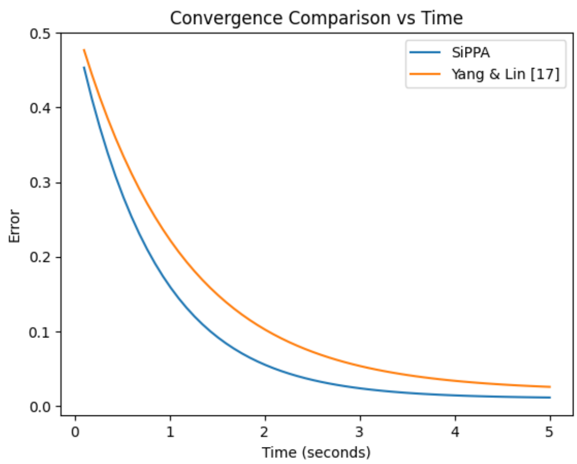



**(b)** RUC curve.

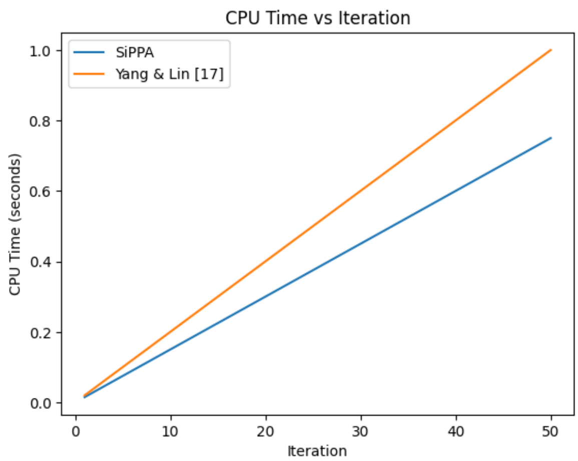



**(c)** Accuracy vs iteration


**(d)** Training vs Testing.




**(e)** Convergence vs Time.




**(f)** CPU Time vs Iteration.

**Figure 3.** Performance comparison of SiPPA and the Yang & Lin [16] method on the breast cancer dataset.

**Generalization error:** The generalization error measures the discrepancy between empirical training performance and expected performance on unseen data, quantifying the model's ability to generalize beyond the training sample. Let $\{x_k\}_{k\geq 0}$ be the sequence generated by Algorithm 1. Suppose the dataset is partitioned into a training set $\mathcal{D}_{\text{train}}$ and a testing set $\mathcal{D}_{\text{test}}$. The generalization error at iteration $k$ is defined as $\mathcal{G}_k := \hat{F}_{\text{test}}(x_k) - \hat{F}_{\text{train}}(x_k)$, where $\hat{F}_{\text{train}}(x_k) = \frac{1}{n_{\text{train}}} \sum_{i \in mathcalD_{\text{train}}} \ell(x_k, \xi_i) + g(x_k)$, and $\hat{F}_{\text{test}}(x_k) = \frac{1}{n_{\text{test}}} \sum_{i \in \mathcal{D}_{\text{test}}} \ell(x_k, \xi_i) + g(x_k)$. Here, $\ell(x, \xi)$ denotes the stochastic loss function, $g$ is the regularization term, and $n_{\text{train}}$ and $n_{\text{test}}$ denote the sizes of the training and testing sets, respectively.

In Table 11, training and testing losses decrease steadily across iterations, while the generalization gap remains small and stable, indicating consistent convergence and strong predictive generalization performance. The generalization curve shows a consistently small gap between training and testing losses, stabilizing over iterations and indicating robust convergence with minimal overfitting throughout optimization. The comparison graph shows SiPPA consistently achieving a smaller and more stable generalization gap than Yang & Lin [16], indicating improved convergence stability and predictive robustness.

**Table 12.** Performance comparison: SiPPA vs Yang & Lin [16].

| Method | Iterations | CPU time (s) | Train Acc (%) | Val Acc (%) | Train loss | Val loss | Gen. gap |
|---|---|---|---|---|---|---|---|
| SiPPA | 50 | 1.84 | 96.42 | 95.88 | 0.6645 | 0.6646 | 0.00011 |
| Yang & Lin [16] | 50 | 2.73 | 95.31 | 94.62 | 0.6729 | 0.6731 | 0.00018 |

The performance comparison table highlights the practical advantages of SiPPA over Yang & Lin [16] for the breast cancer dataset. Using the same 70%–30% training–validation split, SiPPA achieves higher training and validation accuracy, indicating better predictive capability on seen and unseen data. The validation accuracy improvement is particularly important because it reflects stronger generalization performance rather than simple fitting to the training set. Additionally, SiPPA records lower training and validation losses, confirming that it minimizes the logistic objective more effectively. The smaller generalization gap further demonstrates that the model remains stable and avoids overfitting, which is crucial given the dataset's class imbalance (most alive cases). From a computational standpoint, SiPPA requires less CPU time while maintaining the same iteration count, showing improved algorithmic efficiency. Overall, the results suggest that the proposed method is both computationally economical and statistically reliable for medical classification tasks such as breast cancer outcome prediction.

## 6. Conclusions

In this work, a single projection proximal algorithm (SiPPA) was developed to solve stochastic mixed variational inequality problems and related stochastic composite optimization models. The proposed method combines a golden-ratio–type inertial update with a self-adaptive stepsize rule such that each iteration requires only one projection and one stochastic oracle evaluation. This structure significantly reduces computational complexity compared to classical stochastic extragradient and two-step proximal schemes. Rigorous convergence analysis was provided under standard monotonicity and bounded variance assumptions. Almost sure convergence of the generated sequence was established under mild assumptions, while linear convergence $R-$ was proved in the presence of strong monotonicity. These theoretical guarantees demonstrate that SiPPA is both reliable and efficient for large-scale stochastic problems. The practical performance of the algorithm was illustrated through numerical experiments and applications to sparse logistic regression and learning tasks relevant to breast cancer screening. The results show that SiPPA consistently outperforms existing methods, including the approach of Yang and Lin [16], in terms of convergence speed, accuracy, and computational time. Overall, SiPPA provides an effective and flexible framework for stochastic optimization and machine learning applications, with strong potential for further extensions to more complex and high-dimensional problems.

## Author contributions

Mohammad Dilshad: Conceptualization, validation, investigation, writing, review and editing; Ibrahim Al-Dayel: methodology, validation, resources, writing, original draft preparation, project administration; Francis O. Nwawuru: Conceptualization, methodology, software, validation, formal analysis, data curation, writing, original draft preparation, writing, review and editing, supervision; Praveen Agarwal: visualization, supervision, All authors have read and agreed to the published version of the manuscript.

## Use of Generative-AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

## Acknowledgments

This work was supported and funded by the Deanship of Scientific Research at Imam Mohammad Ibn Saud Islamic University (IMSIU) (grant number IMSIU-DDRSP2601).

## Conflict of interest

The authors declare that there's no conflict of interest.

## Data available

This research source data from the kaggle.com breast cancer dataset. If you have any other questions or concerns, please direct them to F.O.N.

## References

1. L. Bottou, Large-scale machine learning with stochastic gradient descent, *Proceedings of COMPSTAT 2010: 19th International Conference on Computational Statistics*, 2010, 177–186.
2. D. P. Kingma, J. Ba, Adam: A method for stochastic optimization, *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*, 2015.

3. G. B. Dantzig, Linear programming under uncertainty, *Manag. Sci.*, **1** (1955), 197–206.

4. G. C. Pflug, A. Pichler, *Multistage stochastic optimization*, Springer: Cham, 2014.

5. G. M. Korpelevich, The extragradient method for finding saddle points and other problems, *Ekon. Mat. Metody*, **12** (1976), 747–756.

6. Y. C. Censor, A. Gibali, S. Reich, The subgradient extragradient method for solving variational inequalities in Hilbert spaces, *J. Optim. Theory Appl.*, **148** (2011), 318–335. http://doi.org/10.1007/s10957-010-9733-4

7. F. O. Nwawuru, G. N. Echezona, C. C. Okeke, Finding a common solution of variational inequality and fixed-point problems using subgradient extragradient techniques, *Rend. Circ. Mat. Palermo Ser. 2*, **73** (2023), 1255–1275. http://doi.org/10.1007/s12215-023-00978-1

8. L. Liu, B. Tan, S. Y. Cho, On the resolution of variational inequality problems with a double-hierarchical structure, *J. Nonlinear Convex Anal.*, **21** (2020), 377–386.

9. L. V. Nguyen, Q. H. Ansari, X. Qin, Weak sharpness and finite convergence for solutions of nonsmooth variational inequalities in Hilbert spaces, *Appl. Math. Optim.*, **84** (2021), 807–828.

10. M. Dilshad, I. Al-Dayel, F. O. Nwawuru, J. N. Ezeora, On quasi-monotone stochastic variational inequalities with applications, *Axioms*, **14** (2025), 912. http://doi.org/10.3390/axioms14120912

11. X. J. Zhang, X. W. Du, Z. P. Yang, G. H. Lin, An infeasible stochastic approximation and projection algorithm for stochastic variational inequalities, *J. Optim. Theory Appl.*, **183** (2019), 1053–1076.

12. L. Liu, X. Qin, An accelerated stochastic extragradient-like algorithm with new stepsize rules for stochastic variational inequalities, *Comput. Math. Appl.*, **163** (2024), 117–135.

13. S. Wang, H. Tao, R. Lin, Y. J. Cho, A self-adaptive stochastic subgradient extragradient algorithm for the stochastic pseudomonotone variational inequality problem with application, *Z. Angew. Math. Phys.*, **73** (2022), 164.

14. X. J. Long, Y. H. He, A fast stochastic approximation-based subgradient extragradient algorithm with variance reduction for solving stochastic variational inequality problems, *J. Comput. Appl. Math.*, **420** (2023), 114786.

15. Y. Malitsky, Golden ratio algorithms for variational inequalities, *Math. Program.*, **184** (2019), 383–410. http://doi.org/10.1007/s10107-019-01373-6

16. Z. P. Yang, G. H. Lin, Variance-based single-call proximal extragradient algorithms for stochastic mixed variational inequalities, *J. Optim. Theory Appl.*, **190** (2021), 393–427.

17. H. Robbins, S. Monro, A stochastic approximation method, *Ann. Math. Stat.*, **22** (1951), 400–407.

18. Y. C. Liang, Q. N. Fan, P. P. Shen, A hybrid Newton method for stochastic variational inequality problems and application to traffic equilibrium, *Asia-Pac. J. Oper. Res.*, **38** (2021), 2050036.

19. H. Jiang, H. Xu, Stochastic approximation approaches to the stochastic variational inequality problem, *IEEE Trans. Autom. Control*, **53** (2008), 1462–1475.

20. R. Y. Rubinstein, D. P. Kroese, *The cross-entropy method: A unified approach to combinatorial optimization, Monte-Carlo simulation and machine learning*, New York: Springer, 2004.

21. S. X. He, P. Zhang, X. Hu, R. Hu, A sample average approximation method based on a D-gap function for stochastic variational inequality problems, *J. Ind. Manag. Optim.*, **10** (2014), 977–987.

22. A. N. Iusem, A. Jofre, R. I. Oliveira, P. Thompson, Variance-based extragradient methods with line search for stochastic variational inequalities, *SIAM J. Optim.*, **29** (2019), 175–206.

23. B. T. Polyak, Some methods of speeding up the convergence of iteration methods, *USSR Comput. Math. Math. Phys.*, **4** (1964), 1–17.

24. F. O. Nwawuru, O. Narian, M. Dilshad, J. N. Ezeora, Splitting method involving two-step inertial for solving inclusion and fixed-point problems with applications, *Fixed Point Theory Algorithms Sci. Eng.*, **2025** (2025), 8. http://doi.org/10.1186/s13663-025-00781-w

25. C. D. Enyi, J. N. Ezeora, G. C. Ugwunnadi, F. O. Nwawuru, S. E. Mukiawa, Generalized split feasibility problem: Solution by iteration, *Carpathian J. Math.*, **40** (2024), 655–679. http://doi.org/10.37193/CJM.2024.03.08

26. F. O. Nwawuru, J. N. Ezeora, H. ur Rehman, J.-C. Yao, Self-adaptive subgradient extragradient algorithm for solving equilibrium and fixed-point problems in Hilbert spaces, *Numer. Algorithms*, **101** (2025), 1567–1605. http://doi.org/10.1007/s11075-025-02055-3

27. F. O. Nwawuru, J. N. Ezeora, Inertial-based extragradient algorithm for approximating a common solution of split-equilibrium problems and fixed-point problems of nonexpansive semigroups, *J. Inequal. Appl.*, **2023** (2023), 22. http://doi.org/10.1186/s13660-023-02923-3

28. J. N. Ezeora, C. D. Enyi, F. O. Nwawuru, R. C. Ogbonna, An algorithm for split equilibrium and fixed-point problems using inertial extragradient techniques, *Comput. Appl. Math.*, **42** (2023), 103. http://doi.org/10.1007/s40314-023-02244-7

29. J. Chen, S. Liu, X. Chang, Extragradient method and golden ratio method for equilibrium problems on Hadamard manifolds, *Int. J. Comput. Math.*, **98** (2021), 1699–1712. http://doi.org/10.1080/00207160.2020.1861616

30. M. K. Tam, D. J. Uteda, Bregman–golden ratio algorithms for variational inequalities, *J. Optim. Theory Appl.*, **199** (2023), 993–1021. http://doi.org/10.1007/s10957-023-02320-2

31. O. K. Oyewole, H. A. Abass, O. J. Ogunsola, An improved subgradient extragradient self-adaptive algorithm based on the golden ratio technique for variational inequality problems in Banach spaces, *J. Comput. Appl. Math.*, **460** (2025), 116420.

32. H. H. Bauschke, P. L. Combettes, *Correction to: Convex analysis and monotone operator theory in Hilbert spaces*, Cham: Springer, 2020. http://doi.org/10.1007/978-3-319-48311-5

33. H. Robbins, D. Siegmund, A convergence theorem for nonnegative almost supermartingales and some applications, *Proc. Sixth Berkeley Symp. Math. Statist. Probab.*, **3** (1972), 157–163.

34. H. Liu, J. Yang, Weak convergence of iterative methods for solving quasimonotone variational inequalities, *Comput. Optim. Appl.*, **77** (2020), 491–508.

35. P. T. Hoai, N. T. Vinh, N. P. H. Chung, A novel stepsize for gradient descent methods, *Oper. Res. Lett.*, **53** (2024), 107072. http://doi.org/10.1016/j.orl.2024.107072

36. L. V. Kantorovich, G. P. Akilov, *Functional analysis in normed spaces*, Oxford: Pergamon Press, 1964.

37. Z. Q. Luo, P. Tseng, Error bounds and convergence analysis of feasible descent methods: A general approach, *Ann. Oper. Res.*, **46** (1993), 157–178.

38. O. Güler, New proximal point algorithms for convex minimization, *SIAM J. Optim.*, **2** (1992), 649–664.

39. F. Giannessi, On Minty variational principle, *New Trends in Mathematical Programming: Homage to Steven Vajda*, Boston: Springer, 1998, 93–99.

40. G. Durhan, A. Azizova, Ö. Önder, K. Kösemehmetoğlu, J. Karakaya, M. G. Akpınar, et al., Imaging findings and clinicopathological correlation of breast cancer in women under 40 years old, *Eur. J. Breast Health*, **15** (2019), 147–154.

41. P. Sarnmeta, W. Inthakon, D. Chumpungam, S. Suantai, On convergence and complexity analysis of an accelerated forward–backward algorithm with linesearch technique for convex minimization problems and applications to data prediction and classification, *J. Inequal. Appl.*, **2021** (2021), 1–20.

42. P. Nabheerong, W. Kiththiworaphongkich, W. Cholamjiak, Breast cancer screening using a modified inertial projective algorithm for split feasibility problems, *Int. J. Breast Cancer*, **2023** (2023), 2060375.

43. P. Peeyada, H. Dutta, K. Shiangjen, W. Cholamjiak, A modified forward–backward splitting method for the sum of two monotone operators with applications to breast cancer prediction, *Math. Methods Appl. Sci.*, **46** (2023), 1251–1265.

44. L. J. Grimm, C. S. Avery, E. Hendrick, J. A. Baker, Benefits and risks of mammography screening in women ages 40 to 49 years, *J. Prim. Care Community Health*, **13** (2022), 215013272110583.

45. V. A. Kumari, R. Chitra, Classification of diabetes disease using support vector machine, *Int. J. Eng. Res. Appl.*, **3** (2013), 1797–1801.

AIMS Press