Mathematics

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

AIMS Mathematics, 8(8): 18898–18918.
DOI:10.3934/math.2023962
Received: 09 January 2023
Revised: 01 May 2023
Accepted: 05 May 2023
Published: 05 June 2023

Research article

Heart disease detection using inertial Mann relaxed CQ algorithms for split feasibility problems

Suthep Suantai[1], Pronpat Peeyada[2], Andreea Fulga[3] and Watcharaporn Cholamjiak[2,*]

[1] Department of Mathematics, Faculty of Science, Chiang Mai University, Chiang Mai 50200, Thailand

[2] School of Science, University of Phayao, Phayao 56000, Thailand

[3] Department of Mathematics and Computer Sciences, Universitatea Transilvania Brasov, 500036 Brasov, Romania

* **Correspondence:** Email: watcharaporn.ch@up.ac.th.

Abstract: This study investigates the weak convergence of the sequences generated by the inertial relaxed CQ algorithm with Mann's iteration for solving the split feasibility problem in real Hilbert spaces. Moreover, we present the advantage of our algorithm by choosing a wider range of parameters than the recent methods. Finally, we apply our algorithm to solve the classification problem using the heart disease dataset collected from the UCI machine learning repository as a training set. The result shows that our algorithm performs better than many machine learning methods and also extreme learning machine with fast iterative shrinkage-thresholding algorithm (FISTA) and inertial relaxed CQ algorithm (IRCQA) under consideration according to accuracy, precision, recall, and F1-score.

Keywords: weak convergence; inertial technique; split feasibility problem; data classification; heart disease data

Mathematics Subject Classification: 46E20, 46N10, 47H04, 65Z05

1. Introduction

In this paper, we study the split feasibility problem (SFP) which is defined on two nonempty closed and convex subsets C and Q of real Hilbert space H_1 and H_2, respectively when $\mathcal{A} : H_1 \to H_2$ is a bounded linear operator. The problem SFP is to

$$\text{find } \mu^* \in C \text{ such that } \mathcal{A}\mu^* \in Q, \tag{1.1}$$

if such μ^* exists. The set $\Omega := \{\mu^* \in C : \mathcal{A}\mu^* \in Q\}$ is denoted for the solution set of the problem SFP (1.1).

In 1994, Censor and Elfving [8] first introduced the algorithm for solving the problem SFP (1.1). The existence of the inverse of the operator $\mathcal{A}^{-1}$ need to be required for computing of each iteration. After that many mathematicians (see in [3,9,10,14,34,37]) applied the problem SFP (1.1) to solve real world problems such as signal and image processing, automatic control systems, machine learning, and many more.

Byrne [7] was the first to propose a popular CQ algorithm solving SFP (1.1) which generates a sequence $\{\mu_n\}$ by the recursive procedure,

$$\mu_{n+1} = P_C(\mu_n - \lambda\mathcal{A}^T(I - P_Q)\mathcal{A}\mu_n),\ \forall n \geq 1, \tag{1.2}$$

where λ belongs in the open interval $(0, \frac{2}{\|\mathcal{A}\|^2})$ with P_C and P_Q are the projections matric onto C and Q, respectively. Another one of the famous algorithms in convex minimization problems is known that the gradient projection algorithm (GPA), this algorithm was generated as follow:

$$\mu_{n+1} = P_C(\mu_n - \lambda_n \triangledown f(\mu_n)),\ \ \forall n \geq 1, \tag{1.3}$$

where $f : H_1 \to (-\infty, +\infty]$ is a lower semicontinuous convex function, λ_n the stepsize at iteration n is chosen in the interval $(0, \frac{2}{L})$, where L is the Lipschitz constant of $\triangledown f$. It is well known that the algorithm GPA (1.3) can be reduced to solve the problem SFP (1.1) when setting $f(\mu) := \frac{1}{2}\|(I - P_Q)\mathcal{A}\mu\|^2$ with $\nabla f(\mu) = \mathcal{A}^T(I - P_Q)\mathcal{A}\mu$. The Lipschitz condition was required for the step size λ_n of the algorithms (1.2) and (1.3), that is $\lambda_n \in (0, \frac{2}{\|\mathcal{A}\|^2})$. This means that to compute the CQ algorithm, the matrix norm of $\mathcal{A}$ needs to be found, which is generally not easy work in practice.

Later on, Byrne [7] presented a different step size $\{\lambda_n\}$ without matrix norms computing. Also, Yang [41] was interested in using a step size $\{\lambda_n\}$ that has no connection with matrix norms, the algorithm GPA (1.3) was considered for variational inequality problem. After that, many different stepsizes $\{\lambda_n\}$ have been presented by many mathematicians, see in [22,35,36,41]

Another one of the different stepsizes was presented in 2018 by Pham et al. [2], this stepsize is generated as follow:

$$\lambda_n = \frac{\beta_n}{\eta_n},\ \forall n \geq 1, \tag{1.4}$$

where

$$\eta_n = \max\{1, \| \triangledown f_n(\mu_n)\|\},\ \lim_{n\to\infty}\beta_n = 0,\ \sum_{n=1}^{\infty}\beta_n = \infty.$$

The algorithm (1.2) with the stepsize (1.4) was used to solve the problem SFP (1.1). For recent results on the problem SFP with the stepsize (1.4), see [13,19,23,38,43].

Finding a way to make algorithms converge faster is another approach many authors are interested in studying. The inertial technique is one way of solving the smooth convex minimization problem, which was first proposed by Polyak [27] in 1964. Polyak's algorithm was called the heavy ball method, modified from the two-step iterative method. The next iterate is defined by making use of the previous two iterates. Later on, the heavy ball method was improved by Nesterov [25] to speed up the rate of convergence. It is denotable that the inertial terminology dramatically improves the algorithm's performance and has nice convergence properties (see [10]). Since that, the heavy ball method has been widely used to solve a wide variety of problems in the optimization field, as seen in [12,24,30,33].

In 2020, Sahu et al. [28] proposed an inertial relaxed CQ algorithm $\{\mu_n\}$ for solving the problem SFP (1.1) in a real Hilbert space by combining the inertial technique of Alvarez and Attouch [1] with the Byrne algorithm (1.2). This algorithm was generated as follows:

$$\begin{cases} \nu_n = \mu_n + \sigma_n(\mu_n - \mu_{n-1}), \\ \mu_{n+1} = P_{C_n}(\nu_n - \lambda \mathcal{A}^T(I - P_{Q_n})\mathcal{A}(\nu_n)), \ \forall n \geq 1, \end{cases} \tag{1.5}$$

where the stepsize parameter λ is still in the interval involving the norm of operator $\mathcal{A}$ and the extrapolation factor $\sigma_n \in [0, \bar{\sigma}_n]$ and $\sigma \in [0, 1)$ such that

$$\bar{\sigma}_n = \min\left\{\sigma, \frac{1}{\max\{n^2\|\mu_n - \mu_{n-1}\|^2,\ n^2\|\mu_n - \mu_{n-1}\|\}}\right\},\ \forall n \geq 1. \tag{1.6}$$

The weakly convergence of sequence $\{\mu_n\}$ generated by (1.5) was proved under the conditions of the extrapolation factor (1.6) and the stepsize parameter λ.

The study of the development of inertial techniques received significant attention. Subsequently, Beck and Teboulle [5] introduced the well-known fast iterative shrinkage-thresholding algorithm (FISTA). The algorithm is designed by choosing $t_1 = 1, \lambda > 0$ and compute

$$\begin{cases} \nu_n = P_{C_n}(\mu_n - \lambda \mathcal{A}^T(I - P_Q)\mathcal{A}\mu_n), \\ t_{n+1} = \frac{1+\sqrt{1+4t_n^2}}{2}, \sigma_n = \frac{t_n - 1}{t_{n+1}}, \\ \mu_{n+1} = \nu_n + \sigma_n(\nu_n - \nu_{n-1}). \end{cases} \tag{1.7}$$

FISTA has received a lot of attention because of its excellent computational results. Many mathematicians have used its implementation in many problem applications (see [21] and reference therein). This inertial technique is limited in the computation of the $\{\sigma_n\}$ sequence.

With the limit of choosing parameter σ_n of Beck and Teboulle [5], Gibali et al. [17] modified the following the inertial relaxed CQ algorithm (IRCQA) in a real Hilbert space. This algorithm is generated as follows:

$$\begin{cases} \nu_n = \mu_n + \sigma_n(\mu_n - \mu_{n-1}), \\ \mu_{n+1} = P_{C_n}(\nu_n - \lambda_n \mathcal{A}^T(I - P_{Q_n})\mathcal{A}(\nu_n)), \ \forall n \geq 1. \end{cases} \tag{1.8}$$

They proved that, if $\lambda_n = \tau_n \frac{f_n(\mu_n)}{\eta_n^2}$, where $\eta_n = \max\{1, \| \nabla f_n(\mu_n)\|\}$ and $\sigma_n \subset [0, \bar{\sigma}_n]$, where

$$\bar{\sigma}_n = \begin{cases} \min\left\{\sigma, \frac{\epsilon_n}{\|\mu_n - \mu_{n-1}\|^2}\right\}, & \text{if } \mu_n \neq \mu_{n-1}, \\ \sigma, & \text{otherwise}, \end{cases} \tag{1.9}$$

such that $\sum_{n=0}^{\infty} \sigma_n\|\mu_n - \mu_{n-1}\|^2 < \infty$, then the sequence $\{\mu_n\}$ generated by (1.8) converges weakly to an element in a solution set of the problem SFP (1.1). The advantage of the IRCQA (1.8) is the extrapolation factor $\{\sigma_n\}$ can be chosen in many ways under the control condition (1.9), and the stepsize parameter $\{\lambda_n\}$ was built without the matrix norm.

In this paper, we propose an inertial Mann relaxed CQ algorithms to solve the split feasibility problems in Hilbert spaces. Our work is inspired by iterative methods developed Dang et al. [10], and Gibali et al. [17]. We apply our main result to solve a data classification problem in machine learning and then compare the performance of our algorithm with FISTA and IRCQA.

2. Preliminaries

Let H_1 and H_2 be real Hilbert spaces. The strong (weak) convergence of a sequence $\{\mu_n\}$ to μ is denoted by $\mu_n \to \mu$ ($\mu_n \rightharpoonup \mu$), respectively. Given a bounded linear operator $\mathcal{A} : H_1 \to H_2$, $\mathcal{A}^T$ denotes the adjoint of $\mathcal{A}$. For any sequence $\{\mu_n\} \subset H_1$, $\omega_n(\mu_n)$ denotes the weak w-limit set of $\{\mu_n\}$, that is,

$$\omega_\omega(\mu_n) := \{\mu \in H_1 : \mu_{n_j} \rightharpoonup \mu \text{ for some subsequence } \{n_j\} \text{ of } \{n\}\}.$$

Let C be a nonempty closed and convex subset of a real Hilbert space H_1. The metric projection from H_1 onto C is defined by for each $\mu \in H_1$, there exists a unique $x^* \in C$ such that

$$\|\mu - x^*\| \le \|\mu - \nu\|, \ \forall \nu \in C.$$

x^* is called the metric projection from H_1 onto C and denoted by $P_C\mu$.

Lemma 2.1. [35] *Let $f : H_1 \to \mathbb{R}$ be a function defined by*

$$f(\mu) := \frac{1}{2}\|\mathcal{A}\mu - P_Q\mathcal{A}\mu\|^2, \ \forall \mu \in H_1.$$

Then following assertions hold:
(i) f is convex and differentiable;
(ii) f is weakly lower semicontinuous on H_1;
(iii) $\nabla f(\mu) = \mathcal{A}^T(I - P_Q)\mathcal{A}\mu$ for all $\mu \in H_1$;
(iv) ∇f is $\frac{1}{\|\mathcal{A}\|^2}$ inverse strongly monotone, that is,

$$\langle \nabla f\mu - \nabla f y, \mu - \nu\rangle \ge \frac{1}{\|\mathcal{A}\|^2}\|\nabla f\mu - \nabla f\nu\|^2, \ \forall \mu, \nu \in H_1.$$

Lemma 2.2. [1] *Let $\{\kappa_n\}$, $\{\delta_n\}$ and $\{\alpha_n\}$ be the sequences in $[0, +\infty)$ such that $\kappa_{n+1} \le \kappa_n + \alpha_n(\kappa_n - \kappa_{n-1}) + \delta_n$ for all $n \ge 1$, $\sum_{n=1}^{\infty} \delta_n < +\infty$ and there exists a real number α with $0 \le \alpha_n \le \alpha < 1$ for all $n \ge 1$. Then the followings hold:*
(i) $\sum_{n\ge1}[\kappa_n - \kappa_{n-1}]_+ < +\infty$, where $[t]_+ = \max\{t, 0\}$;
(ii) There exists $\kappa^* \in [0, +\infty)$ such that $\lim_{n\to+\infty} \kappa_n = \kappa^*$.

Lemma 2.3. [40] *Consider the problem SFP (1.1) with the function f as in Lemma 2.1 and let $\lambda > 0$ and $\mu^* \in H_1$. The point μ^* solve the problem SFP (1.1) if and only if the point μ^* solve the fixed point equation:*

$$\mu^* = P_C(\mu^* - \lambda\nabla f(\mu^*)) = P_C(\mu^* - \lambda\mathcal{A}^T(I - P_Q)\mathcal{A}\mu^*). \tag{2.1}$$

Lemma 2.4. [26] *Let $\{\mu_n\}$ be a sequence in a real Hilbert H_1 such that there exists a nonempty closed and convex subset Ω of H_1 satisfying:*
$\lim_{n\to\infty} \|\mu_n - \mu\|$ exists for all $\mu \in \Omega$ and any weak cluster point of $\{\mu_n\}$ belongs to Ω.
Then there exists $\mu^* \in \Omega$ such that $\mu_n \rightharpoonup \mu^*$.

Lemma 2.5. [32] *Let X be a Banach space satisfying Opial's condition and let $\{\mu_n\}$ be a sequence in X. Let $u, v \in X$ be such that*

$$\lim_{n\to\infty} \|\mu_n - u\| \ \text{and} \ \lim_{n\to\infty} \|\mu_n - v\| \ \text{exists}.$$

If $\{\mu_{n_k}\}$ and $\{\mu_{m_k}\}$ are subsequences of $\{\mu_n\}$ which converge weakly to u and v, respectively, then $u = v$.

3. Main results

In this section, we introduce an inertial Mann relaxed CQ algorithm for solving the SFP (1.1). Let C and Q be a nonempty closed and convex subsets of a real Hilbert spaces H_1 and H_2, respectively, such that

$$C = \{\mu \in H_1 : c(\mu) \leq 0\}, \quad Q = \{\nu \in H_2 : q(\nu) \leq 0\}, \tag{3.1}$$

where $c : H_1 \to \mathbb{R}$ and $q : H_2 \to \mathbb{R}$ are lower semi-continuous convex functions. We also assume that ∂c and ∂q are bounded operators. For a sequence $\{\nu_n\}$ in H_1, we define the half-spaces C_n and Q_n as follow:

$$C_n = \{\mu \in H_1 : c(\nu_n) \leq \langle u_n, \nu_n - \mu\rangle\}, \tag{3.2}$$

where $u_n \in \ \partial c(\nu_n)$, and

$$Q_n = \{\nu \in H_2 : q(\mathcal{A}\nu_n) \leq \langle v_n, \mathcal{A}\nu_n - \nu\rangle\}, \tag{3.3}$$

where $v_n \in \partial q(\mathcal{A}\nu_n)$ and $\mathcal{A} : H_1 \to H_2$ is bounded linear operator. We see that $C \subseteq C_n$ and $Q \subseteq Q_n$ for each $n \geq 1$. Define

$$f_n(\mu) := \frac{1}{2}\|(I - P_{Q_n})\mathcal{A}\mu\|^2, \ \forall \mu \in H_1 \text{ and } n \geq 1. \tag{3.4}$$

Hence, we have

$$\nabla f_n(\mu) = \mathcal{A}^T(I - P_{Q_n})\mathcal{A}\mu.$$

Our algorithm is defined as follows:

Algorithm 3.1. : Inertial Mann relaxed CQ algorithm

Initialization: Take $\mu_0, \mu_1 \in C$ and set $n = 1$.
Iterative Steps: Generate $\{\mu_n\}$ by computing the following step:
Step 1. Compute

$$\nu_n = \mu_n + \sigma_n(\mu_n - \mu_{n-1}), \tag{3.5}$$

where $\sigma_n \in [0, \sigma)$ for each $n \geq 1$ such that for some $\sigma \in [0, 1)$.
Step 2. Compute

$$z_n = P_{C_n}(\nu_n - \lambda_n \nabla f_n(\nu_n)),$$

where $\lambda_n \in (0, \frac{2}{\|\mathcal{A}\|^2})$.
Step 3. Compute

$$\mu_{n+1} = (1 - \alpha_n)\nu_n + \alpha_n z_n, \tag{3.6}$$

where $\alpha_n \in (0, 1)$.
Update n to $n + 1$ and go to **Step 1**.

Assume that the following conditions hold:

$$\sum_{n=1}^{\infty} \sigma_n \max\{\|\mu_n - \mu_{n-1}\|^2, \|\mu_n - \mu_{n-1}\|\} < \infty. \tag{3.7}$$

$$0 < \liminf_{n\to\infty} \lambda_n \le \limsup_{n\to\infty} \lambda_n < \frac{2}{\|\mathcal{A}\|^2} \tag{3.8}$$

$$0 < \liminf_{n\to\infty} \alpha_n \le \limsup_{n\to\infty} \alpha_n < 1 \tag{3.9}$$

Lemma 3.1. *Let $\{\mu_n\}$ be the sequence generated by Algorithm 3.1. Assume that the conditions (3.7)–(3.9) hold. Then we have the following conclusions:*
(i) $\langle \nabla f_n(\nu_n), \nu_n - \mu^* \rangle \ge 2f_n(\nu_n)$ for all $\mu^* \in \Omega$ and $n \in \mathbb{N}$.
(ii) $\|\mu_{n+1} - \mu^*\|^2 \le \|\nu_n - \mu^*\|^2 - 4\lambda_n\alpha_n(1 - \frac{1}{2}\lambda_n\|\mathcal{A}\|^2)f_n(\nu_n))$ for all $\mu^* \in \Omega$.
(iii) If $\lim_{n\to\infty} \|\mu_n - \mu^*\|$ exists and $\sum_{n=1}^{\infty}[\|\mu_n - \mu^*\|^2 - \|\mu_{n-1} - \mu^*\|^2]_+ < \infty$ for all $\mu^* \in \Omega$ then we have
(a) $\{\mu_n\}$, $\{\nu_n\}$ and $\{\nabla f_n(\nu_n)\}$ are bounded,
(b) $\|\mu_{n+1} - \mu_n\| \to 0$.

Proof. (i) Let $\mu^* \in \Omega$ and $\mathcal{A}^*$ is adjoint operator of $\mathcal{A}$. Since $C \subseteq C_n$ and $Q \subseteq Q_n$, $\mu^* = P_C(\mu^*) = P_{C_n}(\mu^*)$ and $(I - P_Q)(\mathcal{A}\mu^*) = (I - P_{Q_n})(\mathcal{A}\mu^*) = 0$. From $(I - P_{Q_n})$ is firmly nonexpansive, for each $n \in \mathbb{N}$, we have

$$\begin{aligned}
2f_n(\nu_n) &= \|(I - P_{Q_n})\mathcal{A}\nu_n\|^2 \\
&= \|(I - P_{Q_n})\mathcal{A}\nu_n - (I - P_{Q_n})\mathcal{A}\mu^*\|^2 \\
&\le \langle (I - P_{Q_n})\mathcal{A}\nu_n - (I - P_{Q_n})\mathcal{A}\mu^*, \mathcal{A}\nu_n - \mathcal{A}\mu^* \rangle \\
&= \langle (I - P_{Q_n})\mathcal{A}\nu_n, \mathcal{A}\nu_n - \mathcal{A}\mu^* \rangle \\
&= \langle \mathcal{A}^*(I - P_{Q_n})\mathcal{A}\nu_n, \nu_n - \mu^* \rangle \\
&= \langle \nabla f_n(\nu_n), \nu_n - \mu^* \rangle.
\end{aligned}$$

(ii) Let $\mu^* \in \Omega$. Set $t_n = \nu_n - \lambda_n \nabla f_n(\nu_n)$, we have

$$\begin{aligned}
\|\mu_{n+1} - \mu^*\|^2 &= \|(1 - \alpha_n)\nu_n + \alpha_n P_{C_n}((I - \lambda_n \nabla f_n)\nu_n) - \mu^*\|^2 \\
&\le (1 - \alpha_n)\|\nu_n - \mu^*\|^2 + \alpha_n\|P_{C_n}(t_n) - \mu^*\|^2 \\
&\le (1 - \alpha_n)\|\nu_n - \mu^*\|^2 + \alpha_n(\|t_n - \mu^*\|^2 - \|t_n - P_{C_n}(t_n)\|^2) \\
&= \|\nu_n - \mu^*\|^2 - \alpha_n(\|\nu_n - \mu^*\|^2 + \|\nu_n - \lambda_n\nabla f_n(\nu_n) - \mu^*\|^2 - \|\nu_n - \lambda_n\nabla f_n(\nu_n) - \mu_{n+1}\|^2) \\
&= \|\nu_n - \mu^*\|^2 - \alpha_n(\|\nu_n - \mu_{n+1}\|^2 + 2\lambda_n\langle \nabla f_n(\nu_n), \nu_n - \mu^* \rangle - 2\lambda_n\langle \nabla f_n(\nu_n), \nu_n - \mu_{n+1} \rangle).
\end{aligned}$$

From part (i), we get

$$\begin{aligned}
\|\mu_{n+1} - \mu^*\|^2 &\le \|\nu_n - \mu^*\|^2 - \alpha_n(\|\nu_n - \mu_{n+1}\|^2 + 2\lambda_n\|\nabla f_n(\nu_n)\|\|\nu_n - \mu_{n+1}\| - 4\lambda_n f_n(\nu_n)) \\
&\le \|\nu_n - \mu^*\|^2 - \alpha_n(\|\nu_n - \mu_{n+1}\|^2 + (\lambda_n\|\nabla f_n(\nu_n)\|)^2 + \|\nu_n - \mu_{n+1}\|^2 - 4\lambda_n f_n(\nu_n)) \\
&= \|\nu_n - \mu^*\|^2 + \lambda_n^2\alpha_n\|\nabla f_n(\nu_n)\|^2 - 4\alpha_n\lambda_n f_n(\nu_n) \\
&\le \|\nu_n - \mu^*\|^2 + 2\lambda_n^2\alpha_n\|\mathcal{A}\|^2 f_n(\nu_n) - 4\alpha_n\lambda_n f_n(\nu_n) \\
&= \|\nu_n - \mu^*\|^2 - 4\lambda_n\alpha_n(1 - \frac{1}{2}\lambda_n\|\mathcal{A}\|^2)f_n(\nu_n).
\end{aligned} \tag{3.10}$$

(iii) Let $\mu^* \in \Omega$. Suppose that $\lim_{n\to\infty} \|\mu_n - \mu^*\|$ exists, (3.7) holds and $\sum_{n=1}^{\infty}[\|\mu_n - \mu^*\|^2 - \|\mu_{n-1} - \mu^*\|^2]_+ < \infty$, we have

$$\|\mu_{n+1} - \nu_n\|^2 + \|\mu_{n+1} - \mu^*\|^2 = \|\nu_n - \mu^*\|^2 + 2\langle \mu_{n+1} - \nu_n, \mu_{n+1} - \mu^* \rangle. \tag{3.11}$$

On the other hand, for each $n \in \mathbb{N}$,

$$
\begin{aligned}
\|\nu_n - \mu^*\|^2 &= (1+\sigma_n)\|\mu_n - \mu^*\|^2 - \sigma_n\|\mu_{n-1} - \mu^*\|^2 + \sigma_n(1+\sigma_n)\|\,\mu_n - \mu_{n-1}\|^2 \\
&\le (1+\sigma_n)\|\mu_n - \mu^*\|^2 - \sigma_n\|\mu_{n-1} - \mu^*\|^2 + 2\sigma_n\|\mu_n - \mu_{n-1}\|^2. \qquad (3.12)
\end{aligned}
$$

From (3.11) and (3.12), we have

$$
\begin{aligned}
\|\mu_{n+1} - \nu_n\|^2 + \|\mu_{n+1} - \mu^*\|^2 &\le \|\mu_n - \mu^*\|^2 + \sigma_n(\|\mu_n - \mu^*\|^2 - \|\mu_{n-1} - \mu^*\|^2) \\
&\quad + 2\sigma_n\|\,\mu_n - \mu_{n-1}\|^2 + 2\langle \mu_{n+1} - \nu_n, \mu_{n+1} - \mu^*\rangle. \qquad (3.13)
\end{aligned}
$$

Since $\{\mu_n\}$ is bounded, it follows from (3.12) that $\{\nu_n\}$ is also bounded. Since ∇f_n is $\|\mathcal{A}\|^2$-Lipschitz, we have

$$
\|\nabla f_n(\nu_n)\| = \|\nabla f_n(\nu_n) - \nabla f_n(\mu^*)\| \le \|\mathcal{A}\|^2\|\nu_n - \mu^*\|.
$$

Hence $\{\nabla f_n(\nu_n)\}$ is also bounded.

Since $\lambda \in (0, \frac{2}{\|\mathcal{A}\|^2})$, we have

$$
\begin{aligned}
\|\mu_{n+1} - \mu^*\|^2 &\le (1-\alpha_n)\|\nu_n - \mu^*\|^2 + \alpha_n\|z_n - \mu^*\|^2 - (1-\alpha_n)\alpha_n\|\nu_n - z_n\|^2 \\
&\le \|\nu_n - \mu^*\|^2 - (1-\alpha_n)\alpha_n\|\nu_n - z_n\|^2 \\
&= \|\mu_n - \mu^* + \sigma_n(\mu_n - \mu_{n+1})\|^2 - (1-\alpha_n)\alpha_n\|\nu_n - z_n\|^2 \\
&= \|\mu_n - \mu^*\|^2 + 2\sigma_n\langle \mu_n - \mu_{n+1}, \nu_n - \mu^*\rangle - (1-\alpha_n)\alpha_n\|\nu_n - z_n\|^2.
\end{aligned}
$$

This implies that

$$
(1-\alpha_n)\alpha_n\|\nu_n - z_n\|^2 \le \|\mu_n - \mu^*\|^2 - \|\mu_{n+1} - \mu^*\|^2 + 2\sigma_n\langle \mu_n - \mu_{n+1}, \nu_n - \mu^*\rangle.
$$

If follows from $\lim_{n\to\infty}\|\mu_n - \mu^*\|$ exists and (3.7) that

$$
\lim_{n\to\infty}\|\nu_n - z_n\| = 0. \qquad (3.14)
$$

We next show that $\|\mu_{n+1} - \mu_n\| \to 0$. It follows from (3.14), we have

$$
\begin{aligned}
\langle \mu_{n+1} - \nu_n, \mu_{n+1} - \mu^*\rangle &= \langle (1-\alpha_n)\nu_n + \alpha_n z_n - \nu_n, (1-\alpha_n)\nu_n + \alpha_n z_n - \mu^*\rangle \\
&= \alpha_n\langle z_n - \nu_n, (1-\alpha_n)(\nu_n - \mu^*) + \alpha_n(z_n - \mu^*)\rangle \\
&= \alpha_n\langle z_n - \nu_n, (1-\alpha_n)(\nu_n - \mu^*) + \alpha_n\langle z_n - \nu_n, \alpha_n(z_n - \mu^*)\rangle \\
&= \alpha_n(1-\alpha_n)\langle z_n - \nu_n, \nu_n - \mu^*\rangle + \alpha_n^2\langle z_n - \nu_n, z_n - \mu^*\rangle \to 0, \text{ as } n\to\infty. \qquad (3.15)
\end{aligned}
$$

By $\sum_{n=1}^{\infty}[\|\mu_n - z\|^2 - \|\mu_{n-1} - z\|^2]_+ < \infty$ and $\sum_{n=1}^{\infty}\sigma_n\|\mu_n - \mu_{n-1}\|^2 < \infty$, it follows from (3.13) and (3.15) that

$$
\lim_{n\to\infty}\|\mu_{n+1} - \nu_n\|^2 = 0. \qquad (3.16)
$$

On the other hand, from (3.7), we have

$$
\|\nu_n - \mu_n\| = \sigma_n\|\mu_n - \mu_{n-1}\| \to 0. \qquad (3.17)
$$

From (3.16) and (3.17), we get

$$
\|\mu_{n+1} - \mu_n\| \le \|\mu_{n+1} - \nu_n\| + \|\nu_n - \mu_n\| \to 0 \text{ as } n\to\infty.
$$

This completes the proof. □

Let $\{\mu_n\}$ be sequence which defined by Algorithm 3.1. We next prove that there exists a subsequence $\{\mu_{n_j}\}$ of the sequence $\{\mu_n\}$ converges weakly to a solution of the problem SFP (1.1).

Theorem 3.2. *Let H_1 and H_2 be two real Hilbert spaces, and let C and Q be nonempty closed convex subsets of H_1 and H_2, respectively. Let $\mathcal{A} : H_1 \to H_2$ be a bounded linear operator. Assume that the solution set Ω of the problem SFP (1.1) is nonempty, the condition (3.7) holds, and $\{\lambda_n\}$, $\{\alpha_n\}$ satisfies the condition (3.8). Let $\{\mu_n\}$ be a sequence generated by Algorithm 3.1. Then we have the following:*
(i) $\{\mu_n\}, \{\nu_n\}$ and $\nabla f_n(\nu_n)$ are bounded;
(ii) There exists a subsequence $\{\mu_{n_j}\}$ of $\{\mu_n\}$ converging weakly to a point $\mu^* \in \Omega$;
(iii) The sequence $\{\mu_n\}$ converges weakly to a point $\mu^* \in \Omega$.

Proof. Let $\mu^* \in \Omega$. From Lemma 3.1 (ii), there exists $m \in \mathbb{N}$ such that

$$\|\mu_{n+1} - \mu^*\|^2 + 4\lambda_n\alpha_n(1 - \frac{1}{2}\lambda_n\|\mathcal{A}\|^2)f_n(\nu_n) \le \|\nu_n - \mu^*\|^2, \ \forall n \ge m. \tag{3.18}$$

From (3.12) and (3.18), we have

$$\begin{aligned}\|\mu_{n+1} - \mu^*\|^2 + \sigma_n\|\mu_{n-1} - \mu^*\|^2 + 4\lambda_n\alpha_n(1 - \frac{1}{2}\lambda_n\|\mathcal{A}\|^2)f_n(\nu_n) &\le (1 + \sigma_n)\|\mu_n - \mu^*\|^2 \\ &+ 2\sigma_n\|\mu_n - \mu_{n-1}\|^2, \ \forall n \ge m.\end{aligned} \tag{3.19}$$

Since $4\lambda_n\alpha_n(1 - \frac{1}{2}\lambda_n\|\mathcal{A}\|^2)f_n(\nu_n) \ge 0$, from (3.19), we have

$$\|\mu_{n+1} - \mu^*\|^2 + \sigma_n\|\mu_{n-1} - \mu^*\|^2 \le (1 + \sigma_n)\|\mu_n - \mu^*\|^2 + 2\sigma_n\|\mu_n - \mu_{n-1}\|^2, \tag{3.20}$$

which implies that, for each $n \ge m$,

$$\|\mu_{n+1} - \mu^*\|^2 - \|\mu_n - \mu^*\|^2 \le \sigma_n(\|\mu_n - \mu^*\|^2 - \|\mu_{n-1} - \mu^*\|^2) + 2\sigma_n\|\mu_n - \mu_{n-1}\|^2. \tag{3.21}$$

From (3.10), we have

$$\begin{aligned}4\lambda_n\alpha_n(1 - \frac{1}{2}\lambda_n\|\mathcal{A}\|^2)f_n(\nu_n) &\le \|\mu_n - \mu^*\|^2 - \|\mu_{n+1} - \mu^*\|^2 \\ &+2\sigma_n\langle\mu_n - \mu_{n-1}, \nu_n - \mu^*\rangle.\end{aligned} \tag{3.22}$$

Applying Lemma 2.2 of [1] in (3.21) with the data $\psi_n = \|\mu_n - \mu^*\|^2$, $\delta_n = 2\sigma_n\|\mu_n - \mu_{n-1}\|^2$, we obtain that $\lim_{n\to\infty} \|\mu_n - \mu^*\|$ exists and $\sum_{n\ge m}^{\infty}[\|\mu_n - \mu^*\|^2 - \|\mu_{n-1} - \mu^*\|^2]_+ < \infty$. This leads, from Lemma 3.1 (iii) that $\{\mu_n\}, \{\nu_n\}$ and $\{\nabla f_n(\nu_n)\}$ are bounded. Since $\{\nabla f_n(\nu_n)\}$ is bounded. It follows from (3.22) and conditions (3.7)–(3.9) that

$$\lim_{n\to\infty} f_n(\nu_n) = 0. \tag{3.23}$$

Since $\{\nu_{n_j}\}$ is bounded, there exists a subsequence $\{\nu_{n_{j_m}}\}$ of $\{\nu_{n_j}\}$ which converges weakly to μ^*. Since $P_{Q_{n_j}}\mathcal{A}\nu_{n_j} \in Q_{n_j}$, we have

$$q(\mathcal{A}\nu_{n_j}) \le \langle v_{n_j}, \mathcal{A}\nu_{n_j} - P_{Q_{n_j}}\mathcal{A}\nu_{n_j}\rangle, \tag{3.24}$$

where $v_{n_j} \in \partial q(\mathcal{A}\nu_{n_j})$. Since ∂q is bounded, then $\{v_{n_j}\}$ is also bounded. From (3.24), we have

$$q(A\nu_{n_j}) \le \|v_{n_j}\|\|\mathcal{A}\nu_{n_j} - P_{Q_{n_j}}\mathcal{A}\nu_{n_j}\| \to 0 \ \text{ as } \ j \to \infty.$$

It follow from the assumption of q that

$$q(\mathcal{A}\mu^*) \leq 0,$$

which means that $\mathcal{A}\mu^* \in Q$. By Lemma 3.1 (iii), we have

$$\lim_{n\to\infty} \|\mu_n - \mu_{n+1}\| = 0.$$

Note that $z_{n_j} \in C_{n_j}$. By the definition of C_{n_j}, we get

$$c(\nu_{n_j}) \leq \langle u_{n_j}, \nu_{n_j} - z_{n_j}\rangle,$$

where $u_{n_j} \in \partial c(\nu_{n_j})$. Since ∂c is bounded, we see that $\{u_{n_j}\}$ is bounded. From (3.14), we have

$$c(\nu_{n_j}) \leq \|u_{n_j}\|\|\nu_{n_j} - z_{n_j}\| \to 0 \ \text{ as } \ j \to \infty.$$

Similarly, we obtain that $c(\mu^*) \leq 0$, i.e., $\mu^* \in C$. From 3.17. Therefore, $\mu_{n_j} \rightharpoonup \mu^* \in \Omega$.

Since $\{\mu_n\}$ is bounded and H is reflexive, $\omega_\omega(\mu_n)$ is nonempty. Let $p \in \omega_\omega(\mu_n)$ be an arbitrary element. Then there exists a subsequence $\{\mu_{n_k}\}$ of $\{\mu_n\}$ such that $\mu_{n_k} \rightharpoonup p$. Let $q \in \omega_\omega(\mu_n)$ and $\{\mu_{n_i}\} \subseteq \{\mu_n\}$ be such that $\mu_{n_i} \rightharpoonup q$. From (ii), we have $p, q \in \Omega$. By Lemma 2.5, $p = q$. Applying Lemma 2.4 and Lemma 3.1 (iii), there exists $\mu^* \in \Omega$ such that $\mu_n \rightharpoonup \mu^*$. □

For the convergence of Algorithm 3.1, we see that the parameter $\{\lambda_n\}$ needs to satisfy the Lipschitz condition that is $\lambda_n \in (0, \frac{2}{\|\mathcal{A}\|^2})$. So, Algorithm 3.1 is flexible to use by choosing the parameter $\{\lambda_n\}$. For example, applying the stepsize (1.6) and (1.9) of Dang et al. [10] and Gibali et al. [17], respectively, we present a new update step size in the following Algorithm 3.3 and Algorithm 3.4:

Algorithm 3.3. Initialization: Take $\{\lambda_1\} \in (0, \frac{2}{\|\mathcal{A}\|^2})$, $\{\alpha_n\} \in (0, 1)$, and $\rho_1, \rho_2 \in (0, 2)$ and $N \in \mathbb{N}$. Select arbitrary points $\mu_0, \mu_1 \in C$ and $\sigma_n \in [0, \sigma)$ for some $\sigma \in [0, 1)$. Set $n = 1$.

Iterative Steps: Generate $\{\mu_n\}$ by computing the following step:
Step 1. Compute

$$\nu_n = \mu_n + \sigma_n(\mu_n - \mu_{n-1}). \tag{3.25}$$

Step 2. Compute

$$z_n = P_{C_n}(\nu_n - \lambda_n \nabla f_n(\nu_n)).$$

Step 3. Compute

$$\mu_{n+1} = (1 - \alpha_n)\nu_n + \alpha_n z_n. \tag{3.26}$$

$$\lambda_{n+1} = \begin{cases} \min\left\{\lambda_n, \frac{\rho_1\|\nu_n - z_n\|}{\Xi(\nu_n)}, \frac{\rho_2\|z_n - \mu_{n+1}\|}{\Xi(\mu_{n+1})}\right\}, & if\ \Xi(\nu_n) \neq 0, \Xi(\mu_{n+1}) \neq 0,\ n \leq N, \\ \frac{2}{n\|\mathcal{A}\|^2}, & n > N, \\ \lambda_n, & otherwise, \end{cases}$$

where $\Xi(x) = \|\nabla f_n(z_n) - \nabla f_n(x)\|$.
Update n to $n + 1$ and go to **Step 1**.

Algorithm 3.4. Initialization: Take $\{\lambda_1\} \in (0, \frac{2}{\|\mathcal{A}\|^2})$, $\{\alpha_n\} \in (0,1)$, and $\ell > 0$. Select arbitrary points $\mu_0, \mu_1 \in C$ and $\sigma_n \in [0, \sigma)$ for some $\sigma \in [0,1)$. Set $n = 1$.

Iterative Steps: Generate $\{\mu_n\}$ by computing the following step:
Step 1. Compute

$$\nu_n = \mu_n + \sigma_n(\mu_n - \mu_{n-1}). \tag{3.27}$$

Step 2. Compute

$$z_n = P_{C_n}(\nu_n - \lambda_n \nabla f_n(\nu_n)).$$

Step 3. Compute

$$\mu_{n+1} = (1-\alpha_n)\nu_n + \alpha_n z_n. \tag{3.28}$$

$$\lambda_{n+1} = \begin{cases} \min\left\{\lambda_n, \ell\Theta(\nu_n), \ell\Theta(z_n), \ell\Theta(\mu_{n+1})\right\}, & if\ \Theta(\nu_n) \neq 0, \Theta(z_n) \neq 0, \Theta(\mu_{n+1}) \neq 0,\ n \leq N, \\ \frac{2}{n\|\mathcal{A}\|^2}, & n > N, \\ \lambda_n, & otherwise, \end{cases}$$

where $\Theta(x) = \|\nabla f_n(x)\|$.
Update n to $n+1$ and go to **Step 1**.

Remark 3.1. From Algorithms 3.3 and 3.4, it's easy to see that the stepsize λ_n is a nonincreasing sequence in $(0, \frac{2}{\|\mathcal{A}\|^2})$ and satisfies the condition (3.8).

4. Application to data classification problem

Currently, cardiovascular disease is the leading cause of death. World Health Organization (WHO) reported 17.9 million human deaths caused by cardiovascular diseases in the year 2019 that was estimated to be 32% the year 2019 [39]. In Thailand [11] cardiovascular disease is the number 1 cause of death for Thai people and increases in all age groups. Therefore, monitoring the heart condition at regular intervals and tracing out the problem at an earlier stage is the need to control the life-threatening situation due to heart failure. To predict heart disease, we used the UCI Machine Learning Heart Disease dataset, which is available on the Internet at [15], was used to evaluate the proposed model. The dataset comprises 76 characteristics and 303 records. However, only 14 attributes from the dataset were used for training and testing. This dataset contains the various attributes are Age, Gender, CP, Trestbps, Chol, Fbs, Restecg, Thalach, Exang, Oldpeak, Slope, Ca, Thal and Num (target variable). The dataset consists of 138 normal instances versus 165 abnormal instances. The following Table 1 shows visualization of the dataset.

Table 1. Overview of the UCI Machine Learning Heart Disease dataset.

Attribute	Description	$\bar{x}$	S.D.	Max	Min	C.V.
Age	Age of patient in years	54.37	9.07	77	29	16.68
Sex	Male and female	0.68	0.47	1	0	68.10
Cp	Chest pain type	0.97	1.03	3	0	106.55
Trestbps	Resting blood pressure (in mm Hg on admission to the hospital)	131.62	17.51	200	94	13.30
Chol	Serum cholesterol shows the amount of triglycerides present	246.26	51.75	564	126	21.01
Fbs	Fasting blood sugar larger than 120 mg/dl	0.15	0.36	1	0	239.44
Restecg	Resting electrocardiographic results	0.53	0.52	2	0	99.42
Thalach	Maximum heart rate achieved	149.65	22.87	202	71	15.28
Exang	Exercise-induced angina (1 yes)	0.33	0.47	1	0	143.55
Oldpeak	ST depression induced by exercise relative to rest	1.04	1.16	6.2	0	111.50
Slope	The slope of the peak exercise ST segment	1.40	0.62	2	0	43.96
Ca	Number of major vessels colored by fluoroscopy	0.73	1.02	4	0	139.97
Thal	No explanation provided, but probably thalassemia	2.31	0.61	3	0	26.42
Target	No disease, disease	-	-	-	-	-

S.D.: Standard deviation; C.V.: Coefficient of variation.

In 2021, Bharti et al. [6] presented the comparison of different machine learning algorithms of the UCI Machine Learning Heart Disease dataset with feature selections and normalization for getting better accuracy. In this section, we shall apply our Algorithms 3.1, 3.3, and 3.4 to optimize weight parameter in training data for machine learning by using 5-fold cross-validation [20] in extreme learning machine (ELM). Very recently, Sarnmeta et al. [29] also considered the UCI Machine Learning Heart Disease dataset using an accelerated forward backward algorithm with linesearch technique for convex minimization problems in ELM with 10-fold cross-validation. The following Table 2 shows the efficiency of our algorithm in extreme learning machine by original dataset compare with the existing machine learning methods were presented in Bharti et al. [6] and ELM algorithm in Sarnmeta et al. [29].

Table 2. Highest accuracy of different machine learning methods using the UCI Machine Learning Heart Disease dataset.

Machine learning method	Accuracy(%)
Logistic regression	83.30
K neighbors	84.80
Support vector machine	83.20
Random forest	80.30
Decision tree	82.30
Artificial neural network [4]	82.50
Learning vector quantization neural network algorithm [31]	85.55
ELM(Sarnmeta et al. [29])	83.87
ELM(our algorithm)	87.69

For our machine learning classification process, we start at letting $\mathcal{U} := \{(\mu_s, r_s) : \mu_s \in \mathbb{R}^n, r_s \in \mathbb{R}^m, s = 1, 2, ..., N\}$ be a training set of N distinct samples where μ_s is an input training data and r_s is a target data. The output function of ELM for single-hidden layer feed forward neural networks (SLFNs) [16, 42] with M hidden nodes and activation function $\mathcal{V}$ is

$$O_s = \sum_{i=1}^{M} w_i \mathcal{V}(c_i \mu_s + e_i),$$

where c_i and e_i are parameters of weight and finally the bias, respectively. To find the optimal output weight w_i at the i-th hidden node, then the hidden layer output matrix $\mathcal{A}$ is generated as follows:

$$\mathcal{A} = \begin{bmatrix} \mathcal{V}(c_1\mu_1 + e_1) & \dots & \mathcal{V}(c_M\mu_1 + e_M) \\ \vdots & \ddots & \vdots \\ \mathcal{V}(c_1\mu_N + e_1) & \dots & \mathcal{V}(c_M\mu_N + e_M) \end{bmatrix}.$$

To solve ELM is to find optimal output weight $w = [w_1^T, ..., w_M^T]^T$ such that $\mathcal{A}w = \mathcal{T}$, where $\mathcal{T} = [r_1^T, ..., r_N^T]^T$ is the training target data. The least square problem is used for finding the solution of linear equation $\mathcal{A}w = \mathcal{T}$ in the cases of the *Moore-Penrose generalized inverse* of $\mathcal{A}$ may be not easy to compute when the matrix $\mathcal{A}^\dagger$ does not exist. To reduce overfitting of the model in training, we consider constrain least square problem in closed convex subsets C of H_1 as follow:

$$\min_{\omega \in C} \frac{1}{2}\{\|\mathcal{A}\omega - \mathcal{T}\|_2^2\}, \tag{4.1}$$

where $C = \{x \in H_1 : \|x\|_1 \leq \gamma\}$ such that γ is regularization parameters. For applying our inertial Mann relaxed CQ algorithm to solve the problem (4.1), we define $f(\mu) := \frac{1}{2}\|(I - P_Q)\mathcal{A}\mu\|^2$, $\forall \mu \in H_1$, and $Q = \{\mathcal{T}\}$, and let $c(\mu) = \|\mu\|_1 - \gamma$ and $q(\mu) = \frac{1}{2}\|\mu - \mathcal{T}\|^2$.

The following four evaluation metrics: Accuracy, Precision, Recall, and F1-score [18] are considered for comparing the performance of the classification algorithms:

$$Accuracy = \frac{\mathsf{TP} + \mathsf{TN}}{\mathsf{TP} + \mathsf{FP} + \mathsf{TN} + \mathsf{FN}} \times 100\%, \tag{4.2}$$

$$Precision = \frac{\mathsf{TP}}{\mathsf{TP} + \mathsf{FP}} \times 100\%, \tag{4.3}$$

$$Recall = \frac{\mathsf{TP}}{\mathsf{TN} + \mathsf{FN}} \times 100\%, \tag{4.4}$$

$$F1 - score = \frac{2 \times (Precision \times Recall)}{Precision + Recall}, \tag{4.5}$$

where TP:=True Positive, FN:=False Negative, TN:=True Negative and FP:=False Positive.

The binary cross-entropy loss function is the mean of a cross-entropy resulting from two probability distributions, the probability distribution we want versus the probability distribution estimated by the model. By computing the following average:

$$Loss = -\frac{1}{K}\sum_{i=1}^{K} y_i \log \hat{y}_i + (1 - y_i)\log(1 - \hat{y}_i),$$

where $\hat{y}_i$ is the i-th scalar value in the model output, y_i is the corresponding target value, and K is the number of scalar values in the model output.

We start computation by setting the activation function as sigmoid, hidden nodes $M = 100$, regularization parameter $\lambda = 1 \times 10^{-5}$ and $\alpha_n = \frac{1}{n+1}$ for Algorithms 3.1, 3.3, and 3.4 with $\lambda_n = \frac{0.9}{2(max(eigenvalue(\mathcal{A}^T\mathcal{A})))}$ for Algorithm 3.1, $\lambda_1 = \frac{0.9}{2(max(eigenvalue(\mathcal{A}^T\mathcal{A})))}$, $\rho_1 = \rho_2 = 1.99$ for Algorithm 3.3 and $\lambda_1 = \frac{0.9}{2(max(eigenvalue(\mathcal{A}^T\mathcal{A})))}$ for Algorithm 3.4. The stopping criteria is the number of iteration 100. We compare the performance of the algorithm with different parameters $\bar{\sigma}_n$ as seen in Table 3 when

$$\sigma_n = \begin{cases} \frac{\bar{\sigma}_n}{n^2 \max\{\|\mu_n-\mu_{n-1}\|^2, \|\mu_n-\mu_{n-1}\|\}}, & if\ n > N\ and\ \ \mu_n \neq \mu_{n-1}, \\ \bar{\sigma}_n, & otherwise, \end{cases}$$

where N is a number of iterations that we want to stop. We can see that parameters σ_n satisfies the condition in Algorithm 3.1, Algorithm 3.3, and Algorithm 3.4 all of each case of $\bar{\sigma}_n$ in Table 3.

Table 3. Numerical results of $\bar{\sigma}_n$.

			Loss	
	$\bar{\sigma}_n$	Training Time	Training	Test
Algorithm 3.1	0.3	0.0371	0.252224	0.230180
	0.5	0.0239	0.251785	0.229676
	$\frac{1}{n}$	0.0321	0.252403	0.230384
	$\frac{1}{\|\mu_n-\mu_{n-1}\|^2+n^2}$	0.0333	0.252805	0.230993
	$\frac{2^{13}}{\|\mu_n-\mu_{n-1}\|^3+n^3+2^{13}}$	0.0322	0.250660	0.228933
Algorithm 3.3	0.3	0.1511	0.252224	0.230180
	0.5	0.1681	0.251785	0.229676
	$\frac{1}{n}$	0.1804	0.252403	0.230384
	$\frac{1}{\|\mu_n-\mu_{n-1}\|^2+n^2}$	0.1750	0.252805	0.230993
	$\frac{2^{13}}{\|\mu_n-\mu_{n-1}\|^3+n^3+2^{13}}$	0.1773	0.250660	0.228933
Algorithm 3.4	0.3	0.1398	0.252224	0.230180
	0.5	0.1342	0.251785	0.229676
	$\frac{1}{n}$	0.1314	0.252403	0.230384
	$\frac{1}{\|\mu_n-\mu_{n-1}\|^2+n^2}$	0.1123	0.252805	0.230993
	$\frac{2^{13}}{\|\mu_n-\mu_{n-1}\|^3+n^3+2^{13}}$	0.1450	0.250660	0.228933

We can see that $\bar{\sigma}_n = \frac{2^{13}}{\|\mu_n-\mu_{n-1}\|^3+n^3+2^{13}}$ highly improves the performance of Algorithm 3.1, Algorithm 3.3, and Algorithm 3.4. We next choose it as the default inertial parameter for later our calculation.

By setting $\bar{\sigma}_n = \frac{2^{13}}{\|\mu_n-\mu_{n-1}\|^3+n^3+2^{13}}$, $\alpha_n = \frac{1}{n+1}$ for Algorithms 3.1, 3.3, and 3.4 with $\rho_1 = \rho_2 = 1.99$ for Algorithm 3.3. The stopping criteria is the number of iteration 100. We obtain the results of the different parameters h when $\lambda_n = \frac{h}{2(max(eigenvalue(\mathcal{A}^T\mathcal{A})))}$ for Algorithm 3.1 and different parameters λ_1 for Algorithm 3.3 and Algorithm 3.4 as seen in Table 4.

Table 4. Numerical results of λ_n of Algorithm 3.1 and λ_1 of Algorithm 3.3 and Algorithm 3.4, respectively.

			Loss	
	h, λ_1	Training Time	Training	Test
Algorithm 3.1	0.7	0.0380	0.250782	0.228759
	0.9	0.0347	0.250660	0.228933
	1	0.0331	0.250174	0.228310
	1.9	0.0256	0.247012	0.224474
	1.9999	0.0338	0.246779	0.224221
Algorithm 3.3	0.7	0.1440	0.250782	0.228759
	0.9	0.1581	0.250660	0.228933
	1	0.1533	0.250174	0.228310
	1.9	0.1735	0.247012	0.224474
	1.9999	0.1574	0.246795	0.224238
Algorithm 3.4	0.7	0.1317	0.250782	0.228759
	0.9	0.1367	0.250660	0.228933
	1	0.1313	0.250174	0.228310
	1.9	0.1280	0.247012	0.224474
	1.9999	0.1353	0.246779	0.224221

Table 5. Numerical results of α_n.

			Loss	
	α_n	Training Time	Training	Test
Algorithm 3.1	0.3	0.0364	0.242989	0.220795
	0.5	0.0376	0.240726	0.219035
	$\frac{1}{n}$	0.0372	0.244716	0.221871
	$\frac{1}{n+1}$	0.0343	0.246779	0.224221
	$\frac{1}{100n+1}$	0.0366	0.271235	0.259327
Algorithm 3.3	0.3	0.1605	0.243010	0.220813
	0.5	0.1621	0.240745	0.219048
	$\frac{1}{n}$	0.1654	0.244733	0.221890
	$\frac{1}{n+1}$	0.1820	0.246795	0.224238
	$\frac{1}{100n+1}$	0.1762	0.271299	0.259421
Algorithm 3.4	0.3	0.1396	0.242989	0.220795
	0.5	0.1281	0.240726	0.219035
	$\frac{1}{n}$	0.1367	0.244716	0.221871
	$\frac{1}{n+1}$	0.1444	0.246779	0.224221
	$\frac{1}{100n+1}$	0.1264	0.271235	0.259327

We can see that $h = \lambda_1 = 1.9999$ highly improves the performance of Algorithm 3.1, Algorithm 3.3, and Algorithm 3.4. We next choose it as the default suitable step size for later our calculation.

Setting the inertial parameters $\bar{\sigma}_n = \frac{2^{13}}{\|\mu_n-\mu_{n-1}\|^3+n^3+2^{13}}$, $\lambda_n = \frac{1.9999}{2(max(eigenvalue(\mathcal{A}^T\mathcal{A})))}$ for Algorithm 3.1 and $\bar{\sigma}_n = \frac{2^{13}}{\|\mu_n-\mu_{n-1}\|^3+n^3+2^{13}}$, $\lambda_1 = \frac{1.9999}{2(max(eigenvalue(\mathcal{A}^T\mathcal{A})))}$ for Algorithm 3.3 and Algorithm 3.4 with $\rho_1 = \rho_2 = 1.99$ for Algorithm 3.3. The comparison of all algorithms with different parameters α_n are presented in Table 5.

We can see that $\alpha_n = 0.5$ highly improves the performance of Algorithm 3.1, Algorithm 3.3, and Algorithm 3.4. Therefore, we choose it as the default parameter α_n for later our calculation. We compare the performance of FISTA, IRCQA, and our algorithm. All the parameters are chosen as seen in Table 6.

Table 6. Chosen parameters of each algorithm.

Algorithm	$\bar{\sigma}_n$	λ_n	λ_1	α_n	ρ_1, ρ_2	τ_n
FISTA	-	$\frac{0.2}{2\|\mathcal{A}\|^2}$	-	-	-	-
IRCQA	$\frac{1}{\|\mu_n-\mu_{n-1}\|^2+n+2}$	-	-	-	-	$\frac{1}{n+1}$
Algorithm 3.1	$\frac{2^{13}}{\|\mu_n-\mu_{n-1}\|^3+n^3+2^{13}}$	$\frac{1.9999}{2(max(eigenvalue(\mathcal{A}^T\mathcal{A})))}$	-	0.5	-	-
Algorithm 3.3	$\frac{2^{13}}{\|\mu_n-\mu_{n-1}\|^3+n^3+2^{13}}$	-	$\frac{1.9999}{2(max(eigenvalue(\mathcal{A}^T\mathcal{A})))}$	0.5	1.99	-
Algorithm 3.4	$\frac{2^{13}}{\|\mu_n-\mu_{n-1}\|^3+n^3+2^{13}}$	-	$\frac{1.9999}{2(max(eigenvalue(\mathcal{A}^T\mathcal{A})))}$	0.5	-	-

For comparison, We set sigmoid as an activation function, number of hidden nodes $M = 100$ and regularization parameter $\lambda = 1 \times 10^{-5}$.

Table 7 shows that our algorithm is among those with the highest precision, recall, F1-score, and accuracy efficiency. Additionally, it has the lowest number of iterations. This means that it has the highest probability of correctly classifying heart disease compared to algorithms examinations. We next present the training and validation loss with the accuracy of training to show that our algorithm has good fit model in the training dataset.

Table 7. The performance of each algorithm.

Algorithm	**Iteration No.**	**Training Time**	**Precision**	**Recall**	**F1-score**	**Accuracy**
FISTA	72	0.0336	100.00	87.50	93.33	87.69
IRCQA	85	0.0758	100.00	87.50	93.33	87.69
Algorithm 3.1	67	0.0386	100.00	87.50	93.33	87.69
Algorithm 3.3	68	0.0975	100.00	87.50	93.33	87.69
Algorithm 3.4	67	0.0934	100.00	87.50	93.33	87.69

From Figures 1–3, we can see that the Training Loss and Validation Loss values have decreased, where the Validation Loss value is lower than Training Loss. On the contrary, when we look at the Accuracy graph, we see that Training Accuracy and Validation Accuracy increase, where the Validation Accuracy is higher than Training Accuracy.

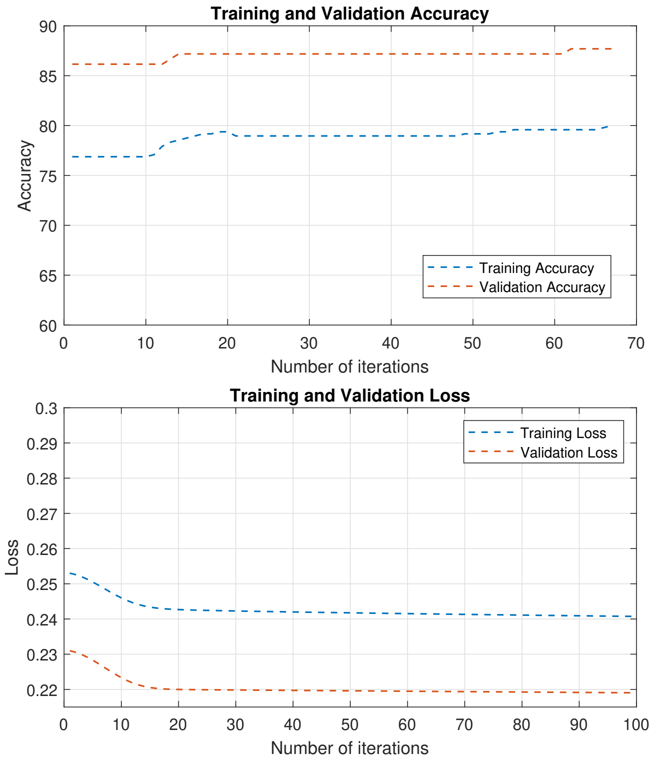

Figure 1. Accuracy and Loss plots of the iteration of Algorithm 3.1.

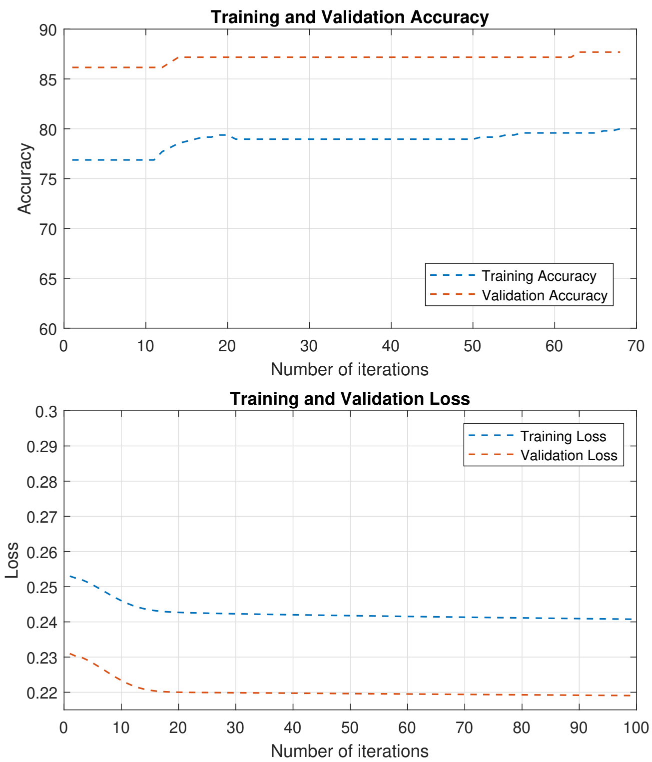

Figure 2. Accuracy and Loss plots of the iteration of Algorithm 3.3.

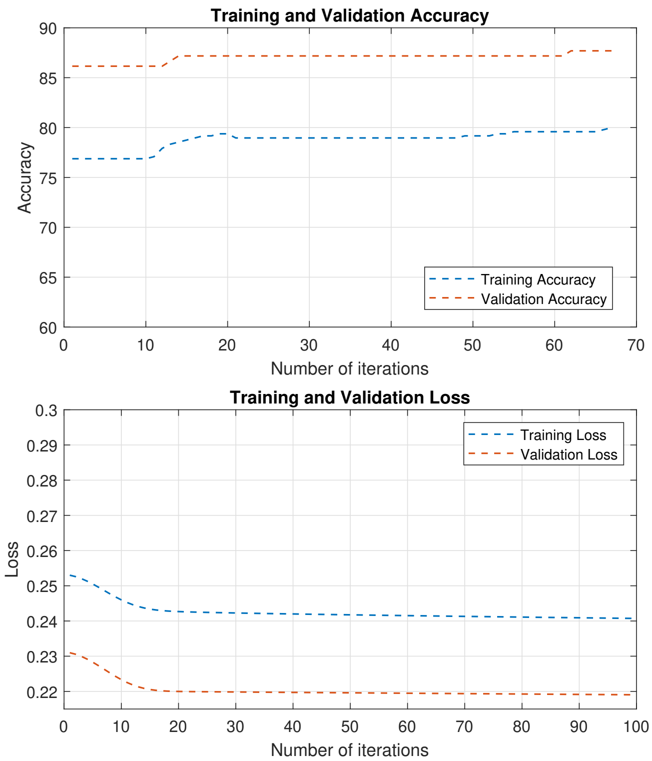

Figure 3. Accuracy and Loss plots of the iteration of Algorithm 3.4.

5. Conclusions

This paper considers solving split feasibility problems using the inertial Mann relaxed *CQ* algorithms. Under some suitable conditions imposed on parameters, we have proved the weak convergence of the algorithm. Moreover, we present choosing different stepsize modifications to achieve an efficient algorithm. We show the efficiency of our algorithm by comparing it with different machine learning methods and also extreme learning machine with FISTA and IRCQA algorithms in data classification using the UCI Machine Learning Heart Disease dataset. The results show that our algorithms are better than the other algorithms.

Use of AI tools declaration

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

Acknowledgments

This research was supported by the NSRF via the program Management Unit for Human Re-sources & Institutional Development, Research and Innovation [grant number B05F640183] and Chiang Mai University. This research was also supported by National Research Council of Thailand (N42A650334) and Thailand Science Research and Innovation, the University of Phayao (Grant No. FF66-UoE).

Data availability

The dataset used in this research is publicly available at the UCI machine learning repository on https://archive.ics.uci.edu/ml/datasets/Heart+Disease.

Conflict of interest

The authors declare no conflicts of interest.

References

1. F. Alvarez, H. Attouch, An inertial proximal method for maximal monotone operators via discretization of a nonlinear oscillator with damping, *Set-Valued Anal.*, **9** (2001), 3–11. https://doi.org/10.1023/A:1011253113155
2. P. K. Anh, N. T. Vinh, V. T. Dung, A new self-adaptive CQ algorithm with an application to the LASSO problem, *J. Fixed Point Theory Appl.*, **20** (2018), 142. https://doi.org/10.1007/s11784-018-0620-8
3. Q. H. Ansari, A. Rehan, Split feasibility and fixed point problems, In: *Nonlinear analysis*, New Delhi: Birkhäuser, 2014, 281–322. https://doi.org/10.1007/978-81-322-1883-8_9
4. K. Aravinthan, M. Vanitha, A comparative study on prediction of heart disease using cluster and rank based approach, *International Journal of Advanced Research in Computer and Communication Engineering*, **5** (2016), 421–424.
5. A. Beck, M. Teboulle, A fast iterative shrinkage-thresholding algorithm for linear inverse problems, *SIAM J. Imaging Sci.*, **2** (2009), 183–202. https://doi.org/10.1137/080716542
6. R. Bharti, A. Khamparia, M. Shabaz, G. Dhiman, S. Pande, P. Singh, Prediction of heart disease using a combination of machine learning and deep learning, *Comput. Intell. Neurosci.*, **2021** (2021), 8387680. https://doi.org/10.1155/2021/8387680
7. C. Byrne, Iterative oblique projection onto convex sets and the split feasibility problem, *Inverse probl.*, **18** (2002), 441. https://doi.org/10.1088/0266-5611/18/2/310
8. Y. Censor, T. Elfving, A multiprojection algorithm using Bregman projections in a product space, *Numer. Algor.*, **8** (1994), 221–239. https://doi.org/10.1007/BF02142692
9. Y. T. Chow, Y. Deng, Y. He, H. Liu, X. Wang, Surface-localized transmission eigenstates, super-resolution imaging, and pseudo surface plasmon modes, *SIAM J. Imaging Sci.*, **14** (2021), 946–975. https://doi.org/10.1137/20M1388498
10. Y. Dang, J. Sun, H. Xu, Inertial accelerated algorithms for solving a split feasibility problem, *J. Ind. Manag. Optim.*, **13** (2017), 1383–1394. https://doi.org/10.3934/jimo.2016078
11. Department of Disease Control, the department of disease control joins the campaign for world heart day, 2021. Available from: https://ddc.moph.go.th/brc/news.php?news=20876&deptcode=brc&fbclid=\IwAR2GSqs1NVuYuGfmO4kOsUElK0T4ZOyRFnmbty2aZ_rnQ7Xc3jmhu6DIMSk.

12. Q. L. Dong, J. Z. Huang, X. H. Li, Y. J. Cho, T. M. Rassias, MiKM: multi-step inertial Krasnosel'skiĭ–Mann algorithm and its applications, *J. Glob. Optim.*, **73** (2019), 801–824. https://doi.org/10.1007/s10898-018-0727-x

13. Q. L. Dong, X. H. Li, D. Kitkuan, Y. J. Cho, P. Kumam, Some algorithms for classes of split feasibility problems involving paramonotone equilibria and convex optimization, *J. Inequal. Appl.*, **2019** (2019), 77. https://doi.org/10.1186/s13660-019-2030-x

14. Q. L. Dong, Y. C. Tang, Y. J. Cho, T. M. Rassias, "Optimal" choice of the step length of the projection and contraction methods for solving the split feasibility problem, *J. Glob. Optim.*, **71** (2018), 341–360. https://doi.org/10.1007/s10898-018-0628-z

15. D. Dua, C. Graff, UCI Machine Learning Repository, Irvine, CA: University of California, School of Information and Computer Science, 2019. Available from: `http://archive.ics.uci.edu/ml`.

16. Y. Gao, H. Liu, X. Wang, K. Zhang, On an artificial neural network for inverse scattering problems, *J. Comput. Phys.*, **448** (2022), 110771. https://doi.org/10.1016/j.jcp.2021.110771

17. A. Gibali, D. V. Thong, Tseng type methods for solving inclusion problems and its applications, *Calcolo*, **55** (2018), 49. https://doi.org/10.1007/s10092-018-0292-1

18. J. Han, M. Kamber, J. Pei, *Data mining: concepts and techniques*, Waltham, MA: Morgan Kaufman Publishers, 2012.

19. W. Jirakitpuwapat, P. Kumam, Y. J. Cho, K. Sitthithakerngkiet, A general algorithm for the split common fixed point problem with its applications to signal processing, *Mathematics*, **7** (2019), 226. https://doi.org/10.3390/math7030226

20. V. A. Kumari, R. Chitra, Classification of diabetes disease using support vector machine, *International Journal of Engineering Research and Applications*, **3** (2013), 1797–1801.

21. J. Liang, T. Luo, C. B. Schonlieb, Improving "fast iterative Shrinkage-Thresholding algorithm": faster, smarter, and greedier, *SIAM J. Sci. Comput.*, **44** (2022), A1069–A1091. https://doi.org/10.1137/21M1395685

22. G. López, V. Martín-Márquez, F. Wang, H. K. Xu, Solving the split feasibility problem without prior knowledge of matrix norms, *Inverse Probl.*, **28** (2012), 085004. https://doi.org/10.1088/0266-5611/28/8/085004

23. Z. Ma, L. Wang, Y. J. Cho, Some results for split equality equilibrium problems in Banach spaces, *Symmetry*, **11** (2019), 194. https://doi.org/10.3390/sym11020194

24. P. E. Maingé, Inertial iterative process for fixed points of certain quasi-nonexpansive mappings, *Set-Valued Anal.*, **15** (2007), 67–79. https://doi.org/10.1007/s11228-006-0027-3

25. Y. E. E. Nesterov, A method of solving a convex programming problem with convergence rate $O(1/k^2)$, *Dokl. Akad. Nauk SSSR.*, **269** (1983), 543–547.

26. Z. Opial, Weak convergence of the sequence of successive approximations for nonexpansive mappings, *Bull. Amer. Math. Soc.*, **73** (1967), 591–597.

27. B. T. Polyak, Some methods of speeding up the convergence of iteration methods, *Comput. Math. Math. Phys.*, **4** (1964), 1–17. https://doi.org/10.1016/0041-5553(64)90137-5

28. D. R. Sahu, Y. J. Cho, Q. L. Dong, M. R. Kashyap, X. H. Li, Inertial relaxed *CQ* algorithms for solving a split feasibility problem in Hilbert spaces, *Numer. Algor.*, **87** (2021), 1075–1095. https://doi.org/10.1007/s11075-020-00999-2

29. 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), 141. https://doi.org/10.1186/s13660-021-02675-y

30. Y. Shehu, A. Gibali, New inertial relaxed method for solving split feasibilities, *Optim. Lett.*, **15** (2021), 2109–2126. https://doi.org/10.1007/s11590-020-01603-1

31. J. S. Sonawane, D. R. Patil, Prediction of heart disease using learning vector quantization algorithm, In: *2014 Conference on IT in Business, Industry and Government (CSIBIG)*, Indore, India, 2014, 1–5. https://doi.org/10.1109/CSIBIG.2014.7056973

32. S. Suantai, Weak and strong convergence criteria of Noor iterations for asymptotically nonexpansive mappings, *J. Math. Anal. Appl.*, **311** (2005), 506–517. https://doi.org/10.1016/j.jmaa.2005.03.002

33. S. Suantai, N. Pholasa, P. Cholamjiak, Relaxed *CQ* algorithms involving the inertial technique for multiple-sets split feasibility problems, *RACSAM*, **113** (2019), 1081–1099. https://doi.org/10.1007/s13398-018-0535-7

34. S. Suantai, S. Kesornprom, P. Cholamjiak, A new hybrid *CQ* algorithm for the split feasibility problem in Hilbert spaces and its applications to compressed sensing, *Mathematics*, **7** (2019), 789. https://doi.org/10.3390/math7090789

35. N. T. Vinh, P. Cholamjiak, S. Suantai, A new CQ algorithm for solving split feasibility problems in Hilbert spaces, *Bull. Malays. Math. Sci. Soc.*, **42** (2019), 2517–2534. https://doi.org/10.1007/s40840-018-0614-0

36. F. Wang, Polyak's gradient method for split feasibility problem constrained by level sets, *Numer. Algor.*, **77** (2018), 925–938. https://doi.org/10.1007/s11075-017-0347-4

37. X. Wang, Y. Guo, D. Zhang, H. Liu, Fourier method for recovering acoustic sources from multi-frequency far-field data, *Inverse Probl.*, **33** (2017), 035001. https://doi.org/10.1088/1361-6420/aa573c

38. U. Witthayarat, Y. J. Cho, P. Cholamjiak, On solving proximal split feasibility problems and applications, *Ann. Funct. Anal.*, **9** (2018), 111–122. https://doi.org/10.1215/20088752-2017-0028

39. World Health Organization, Cardiovascular diseases (CVDs), World Health Organization (WHO), 2021. Available from: `https://www.who.int/news-room/fact-sheets/detail/cardiovascular-diseases-(cvds)`.

40. H. K. Xu, Iterative methods for solving the split feasibility in infinite-dimensional Hilbert spaces, *Inverse Probl.*, **26** (2010), 105018. https://doi.org/10.1088/0266-5611/26/10/105018

41. Q. Yang, On variable-step relaxed projection algorithm for variational inequalities, *J. Math. Anal. Appl.*, **302** (2005), 166–179. https://doi.org/10.1016/j.jmaa.2004.07.048

42. W. Yin, W. Yang, H. Liu, A neural network scheme for recovering scattering obstacles with limited phaseless far-field data, *J. Comput. Phys.*, **417** (2020), 109594. https://doi.org/10.1016/j.jcp.2020.109594

43. J. Zhao, Y. Liang, Y. Liu, Y. J. Cho, Split equilibrium, variational inequality and fixed point problems for multi-valued mappings in Hilbert spaces, *Appl. Comput. Math.*, **17** (2018), 271–283.