IPAS: An Adaptive Sample Size Method for Weighted Finite Sum Problems with Linear Equality Constraints

Nataša Krejić[*], Nataša Krklec Jerinkić [†], Sanja Rapajić[‡], Luka Rutešić [§¶]

Abstract

Optimization problems with the objective function in the form of weighted sum and linear equality constraints are considered. Given that the number of local cost functions can be large as well as the number of constraints, a stochastic optimization method is proposed. The method belongs to the class of variable sample size first order methods, where the sample size is adaptive and governed by the additional sampling technique earlier proposed in the unconstrained optimization framework. The resulting algorithm may be a mini-batch method, increasing sample size method, or even deterministic in a sense that it eventually reaches the full sample size, depending on the problem and similarity of the local cost functions. Regarding the constraints, the method uses controlled, but inexact projections on the feasible set, yielding possibly infeasible iterates. Almost sure convergence is proved under some standard assumptions for the stochastic framework, without imposing the convexity. Numerical results on relevant machine learning experiments, i.e., real-world data sets for logistic regression problems, show that the proposed algorithm is competitive with the state-of-the-art methods.

Key words: Constrained Optimization, Projected Gradient Methods, Sample Average Approximation, Adaptive Variable Sample Size Strategies, Nonmonotone Line Search, Additional Sampling.

[*]Department of Mathematics and Informatics, Faculty of Sciences, University of Novi Sad, Trg Dositeja Obradovića 4, 21000 Novi Sad, Serbia. e-mail: natasak@uns.ac.rs

[†]Department of Mathematics and Informatics, Faculty of Sciences, University of Novi Sad, Trg Dositeja Obradovića 4, 21000 Novi Sad, Serbia. e-mail: natasa.krklec@dmi.uns.ac.rs

[‡]Department of Mathematics and Informatics, Faculty of Sciences, University of Novi Sad, Trg Dositeja Obradovića 4, 21000 Novi Sad, Serbia. e-mail: sanja@dmi.uns.ac.rs

[§]Department of Mathematics and Informatics, Faculty of Sciences, University of Novi Sad, Trg Dositeja Obradovića 4, 21000 Novi Sad, Serbia. e-mail: luka.rutesic@dmi.uns.ac.rs

[¶]Corresponding author

1 Introduction

We consider constrained optimization problems with the objective function in the form of weighted finite sum and linear equality constraints, i.e.,

$$\min_{Ax=b} f(x) := \sum_{i=1}^{N} w_i f_i(x), \tag{1.1}$$

where $f_i : \mathbb{R}^n \to \mathbb{R}, i = 1, ..., N$ are continuously-differentiable functions, $w_1, ..., w_N$ represent the weights such that

$$\sum_{i=1}^{N} w_i = 1, \quad w_i \geq 0,\ i = 1, .., N, \tag{1.2}$$

$b \in \mathbb{R}^m$ and $A \in \mathbb{R}^{m \times n}$ is assumed to be a full-rank matrix, $rank(A) = m \leq n$.

The considered problems come from different fields, mainly including Big Data problems commonly present in Machine Learning (ML) [15]. Many of ML problems are in the form of finite sum optimization problems, so a large class of stochastic methods for constrained finite sum optimization problems are proposed in many papers ([31,34] to name just a few). Notice that the choice of $w_i = 1/N$ for all $i = 1, ..., N$ yields the standard form of the finite sum objective function $f(x) = \frac{1}{N} \sum_{i=1}^{N} f_i(x)$. Optimization problems with this objection function and linear constraints have gained significant attention in practical applications (see [41]). Introduction of possibly different weights is motivated by the so called local regression models (see [18] for instance) in ML where the model parameters are recalculated for any new data point depending on its position in the space of attributes. In such approach, the objective function usually takes into account the distance between the new point and data points from the training data set. The aforementioned distances represent the weights in problem (1.1). On the other hand, the weights w_i can be viewed as probabilities of choosing the corresponding functions f_i and, in that case, the objective function represents mathematical expectation. The motivation for emphasizing weights and observing them separately from the functions f_i comes from stochastic framework. Namely, this allows stochastic algorithms to favor the functions that are more important by giving them better chances to be chosen as explained in the sequel of the paper. Moreover, some of the ML problems also include linear constraints. One of the examples is a Ridge regression which can be stated in the form of constrained optimization problem [18]. Another example would be Markowitz utility function minimization, i.e., the problems of finding optimal portfolio that minimizes the risk and maximizes the return, while the sum of unknowns must be equal to one. The two mentioned examples include only modest number of constraints and the projection on the feasible

set is not a big challenge. However, data fitting problems in general (e.g. least squares) may also include a large number of (linear) constraints (see e.g. [37] for further references). In that case, projecting on the feasible set is too expensive and inexact projections may be a better option.

A variety of line search and trust region methods have been proposed to solve nonlinear constrained problems. Various algorithms have been designed to solve deterministic equality-constrained optimization problems (see [4, 9] for further references), while recent research has focused on developing stochastic optimization algorithms. There has been a growing interest in adapting line search and trust region methods in stochastic framework for unconstrained optimization problems [1–3, 7, 8, 10, 12, 19, 21, 24–26], but significantly fewer algorithms have been proposed to solve stochastic equality-constrained optimization problems (see [4] for further references and [5, 11, 14, 35, 38, 40]).

Projected gradient methods can be used to solve constrained optimization problems, (see [6], [13]). Their generalization to stochastic framework have been investigated in [22, 33]. A novel class of projected gradient methods for constrained minimization both in deterministic and stochastic settings is proposed in [27]. Large scale problems require inexact projections because of computational cost.

Numerous first order and second order algorithms have been developed for solving unconstrained finite sum minimization [1–3, 12, 24, 26]. These methods are based on nonmonotone line search or trust region technique. The usefulness of line search nonmonotonicity in deterministic case [17, 28] was also demonstrated in a stochastic case. A class of algorithms which uses nonmonotone line search rule fitting a variable sample size scheme at each iteration was proposed in [21].

In large scale problems, the computation of the objective function and its gradient (and additionally Hessian if needed) is expensive, so their approximations are generally used in order to reduce the computational cost. Sampling-based approaches have long played an important role in stochastic optimization and stochastic programming [32]. Subsampling is a natural way of computing these approximations, and adaptive subsampling appears to be particularly suited to large dimensional problem in the form of finite sums.

Adaptive sample size strategies for finite sum problems are presented in [1, 2, 12, 24, 26] and some other types of adaptive subsample approach can also be found in [3, 7, 8, 10, 22].

Aditional sampling (two step sampling in each iteration) presented in [12, 24, 26] plays an important role in the sample size scheduling. It is used as a control for accepting the step and increasing the sample size if necessary. The additional sampling can be arbitrarily cheap, i.e., even the sample size 1 is sufficient, and hence it does not make the process more expensive. Other additional line search and trust region sample size strategies can be found

in [19, 25].

The method we propose here belongs to the family of projected gradient methods, with the step size determined by a nonmonotone line search rule. The specific form of the objective function, in particular the case of large N and n motivates the adaptive subsample approach. Thus we work with approximate objective function (and the corresponding gradient), in other words with random linear models. The sample size that defines the model in each iteration is computed according to the estimated progress towards minimizer in each iteration. The approximate gradient direction is then projected inexactly to the feasible set, yielding a new iteration that might be infeasible in a controlled way. Inexact projection in fact means that we will solve the corresponding system of linear equations only approximately, using any linear solver. The progress is measured by an additional approximation of the objective function, which can be very rough, in fact even the sample of size 1 will be suitable for this additional objective function approximation. Besides the measure of progress the additional sampling done in each iteration allows us to overcome theoretical difficulties that arise from the fact that the direction and step size are not independent random variables and hence one cannot apply the martingale theory for theoretical analysis. Such difficulties can be overcame with the predetermined step size, for instance $1/k$, which yields a.s. convergence in stochastic gradient descent framework (see e.g. [1] for further references). But in that case the step sizes become very small very fast and hence the method is very slow. Furthermore, the method we propose here is adaptive in the sense that sample size increase is problem dependent. Roughly speaking the similarity of functions f_i governs the process and the iterative procedure might end without reaching the full sample, i.e., with very cheap iterations, or the exact objective function and the gradient are used at the final stages of the iterative procedure.

Random linear models that we use can be more or less similar to the original function f and thus the decrease of the model might not be a decrease for the true objective function. Therefore we use a nonmonomotone line search procedure [17,28], that is more relaxed in accepting the step than the classical Armijo rule. The same approach is exploited in several methods with random models, [2, 12, 22, 26]. Putting together the random linear models, nonmonotone line search procedure, additional sampling and inexact projections we end up with an efficient and theoretically sound method that is almost surely (a.s.) converging to a stationary point of (1.1).

To summarize, our main contributions may be listed as follows:

- The additional sampling concept for solving unconstrained finite sum problems [12,24,26] is incorporated into the stochastic method to solve constrained optimization problems with weighted sum objective functions.

- The proposed method relaxes the common assumption of feasible iter-

ates in stochastic projected gradient framework (e.g. [22]) and allows controlled, but inexact projections which can be of great significance for problems with large number of constraints.

- Almost sure convergence is proved under rather standard assumptions for stochastic optimization framework.
- The efficiency of the proposed method is confirmed through a number of tests performed on both academic and real-world data problems.

The paper is organized as follows. Section 2 contains basic definitions and statements known from the literature. It also provides some basic concepts and preliminary results needed for further analysis. The proposed method - IPAS is stated and explained in Section 3, while the convergence analysis is delegated to Section 4. Section 5 is devoted to numerical results, while the main conclusions are derived in the final section.

2 Preliminaries

Let us start with explaining the inexact projections we will use in the algorithm. Under the assumption of fully ranked A, one can show that the orthogonal projection $\pi_S(y)$ of a point y on the feasible set $S := \{x \in \mathbb{R}^n \mid Ax = b\}$ is given by

$$\pi_S(y) = y - A^T(AA^T)^{-1}(Ay - b). \tag{2.1}$$

The above equality comes from the fact that $\pi_S(y) = argmin_{Ax=b}\frac{1}{2}\|y - x\|^2$ and, since the orthogonal projection problem is convex, the solution is determined by the KKT conditions

$$A\,\pi_S(y) = b, \quad A^T\lambda = y - \pi_S(y),$$

where λ represents the vector of Lagrange multipliers. Multiplying the second equation with A from the left and using the first one, we obtain

$$AA^T\lambda = Ay - b. \tag{2.2}$$

This together with $\pi_S(y) = y - A^T\lambda$ yields (2.1). The important feature for our analysis lies in the fact that the projection operator to the set S has the following property.

Lemma 2.1. *Let $A \in \mathbf{R}^{m\times n}$ with $rank(A) = m$ and $w_i \geq 0, i = 1, \ldots, N$, $\sum_{i=1}^N w_i = 1$. For any set of points $y^i \in \mathbf{R}^n, i = 1, \ldots, N$ there holds*

$$\pi_S(\sum_{i=1}^N w_i y^i) = \sum_{i=1}^N w_i \pi_S(y^i). \tag{2.3}$$

Proof. Using (2.1), there follows

$$\begin{aligned}\pi_S(\sum_{i=1}^{N} w_i y^i) &= \sum_{i=1}^{N} w_i y^i - A^T(AA^T)^{-1}(A\sum_{i=1}^{N} w_i y^i - b) \\ &= \sum_{i=1}^{N} w_i y^i - A^T(AA^T)^{-1}(A\sum_{i=1}^{N} w_i y^i - \sum_{i=1}^{N} w_i b) \\ &= \sum_{i=1}^{N} w_i (y^i - A^T(AA^T)^{-1}(Ay^i - b)) = \sum_{i=1}^{N} w_i \pi_S(y^i).\end{aligned}$$

□

To compute the exact projection on the set S one needs to solve the system (2.2) exactly. In some applications, if the number of equalities is modest one can use the closed form (2.1) for any given point y. However, if the dimension of the problem is very large and/or the number of equality constraints is large, finding the exact solution to (2.2) can be impractical. Thus, we will assume that the linear system is solved only approximately. The quality of inexact projection in each iteration will be controlled by the norm of the residual vector defined as follows. Let us denote by $\tilde{\pi}_S(y)$ the inexact projection of point y on feasible set S, more precisely,

$$\tilde{\pi}_S(y) = y - A^T\tilde{\lambda}(y), \tag{2.4}$$

where $\tilde{\lambda}(y)$ is an approximate solution of (2.2). The residual is denoted by

$$r(y) := AA^T\tilde{\lambda}(y) - Ay + b, \tag{2.5}$$

while the feasibility measure of point y is defined as

$$e(y) = \|Ay - b\|. \tag{2.6}$$

We will state the condition for the residual vector in each iteration a bit ahead. The following simple lemma will be used later on.

Lemma 2.2. *Let $z \in \mathbb{R}^n$ be an arbitrary point and $\tilde{\lambda}(z)$ be an approximate solution of (2.2) such that $\|r(z)\| \leq M$ with $M > 0$. Then $e(\tilde{\pi}_S(z)) \leq M$.*

Proof. The condition

$$\|r(z)\| = \|AA^T\tilde{\lambda}(z) - Az + b\| \leq M$$

implies

$$\begin{aligned}e(\tilde{\pi}_S(z)) &= \|A\tilde{\pi}_S(z) - b\| = \|A(z - A^T\tilde{\lambda}(z)) - b\| \\ &= \|Az - b - AA^T((AA^T)^{-1}(Az - b + r(z)))\| \\ &= \|Az - b - Az + b - r(z)\| = \|r(z)\| \leq M,\end{aligned}$$

which completes the proof. □

One can show that the projected gradient direction of the form

$$d(x) = \pi_S(x - \nabla f(x)) - x \tag{2.7}$$

is a descent direction for function f at point $x \in S$ unless x is a stationary point for problem (1.1). More precisely, the following result is known.

Theorem 2.3. *[6] Assume that $f \in C^1(S)$ and $x \in S$. Then the projected gradient direction* (2.7) *satisfies:*

a) $d(x)^T \nabla f(x) \leq -\|d(x)\|^2$,

b) $d(x) = 0$ *if and only if x is a stationary point for problem* (1.1).

The method we propose will be based on approximate objective function and the gradient. Let us denote by $f_{\mathcal{N}_k}$ the approximation of function f used in iteration k, i.e.,

$$f_{\mathcal{N}_k}(x) := \frac{1}{N_k} \sum_{i \in \mathcal{N}_k} f_i(x), \tag{2.8}$$

where $N_k := |\mathcal{N}_k|$, $\mathcal{N}_k = \{i_1^k, ..., i_{N_k}^k\}$, and each $i_j^k \in \mathcal{N}_k$ takes the value $s \in \mathcal{N} := \{1, ..., N\}$ with probability w_s, i.e.,

$$P(i_j^k = s) = w_s, s = 1, ..., N, j \in \mathcal{N}_k, k \in \mathbb{N}. \tag{2.9}$$

This way we have an unbiased estimate of f, i.e.,

$$E(f_{\mathcal{N}_k}(x)|x) = \frac{1}{N_k} \sum_{j=1}^{N_k} E(f_{i_j^k}(x)|x) = \frac{1}{N_k} \sum_{j=1}^{N_k} f(x) = f(x).$$

As already stated, in the proposed algorithm we use inexact projections of the approximate gradient $\nabla f_{\mathcal{N}_k}$ which can yield nondescent directions p_k and infeasible points x_k. Moreover, we deal with approximate functions and thus imposing monotone line search is not beneficial in general. Thus, we use nonmonotone Armijo-type line search [28], to determine the step size t_k

$$f_{\mathcal{N}_k}(x_k + t_k p_k) \leq f_{\mathcal{N}_k}(x_k) + c_1 t_k (\nabla f_{\mathcal{N}_k}(x_k))^T p_k + \varepsilon_k, \tag{2.10}$$

with some ε_k which satisfies

$$\sum_{k=0}^{\infty} \varepsilon_k \leq \bar{\varepsilon} < \infty, \quad \varepsilon_k > 0. \tag{2.11}$$

Clearly, the direction p_k and the step size t_k obtained in (2.10) both depend on $\mathcal{N}_k$. Therefore we can not rely on the martingale theory commonly used in stochastic gradient methods such as SGD [36] where the predetermined step size sequence is used. To overcome this difficulty and avoid

predefined step sizes we employ the idea of additional sampling. This simple and computationally cheap remedy is successfully used in [12,26]. Other possibilities are given in [19, 25]. The method proposed in [24] also uses additional sampling but in trust-region framework.

Similarly as for $f_{\mathcal{N}_k}$, we form an additional sampling approximation $f_{\mathcal{D}_k}$ by

$$f_{\mathcal{D}_k}(x) := \frac{1}{D_k} \sum_{i \in \mathcal{D}_k} f_i(x), \tag{2.12}$$

where $D_k := |\mathcal{D}_k|$, $\mathcal{D}_k = \{l_1^k, ..., l_{D_k}^k\}$, and each $l_j^k \in \mathcal{D}_k$ takes the value $s \in \mathcal{N} := \{1, ..., N\}$ with probability w_s, i.e.,

$$P(l_j^k = s) = w_s, s = 1, ..., N, j \in \mathcal{D}_k, k \in \mathbb{N}. \tag{2.13}$$

The key point for the efficiency of this approach lies in the fact that the cardinality of $\mathcal{D}_k$ is arbitrary, with only requirement being $D_k \leq N - 1$. Thus one can even take $D_k = 1$ in each iteration. The numerical experiments presented in Section 4 are performed with $D_k = 1$.

The following technical lemmas will be used further on.

Lemma 2.4. *Let $e_{k+1} \leq \theta e_k + \eta_k$ for all $k \geq 0$ with $e_k \geq 0, k = 1, 2, ..., \theta \in [0, 1)$ and $\{\eta_k\}$ satisfying $\lim_{k\to\infty} \eta_k = 0$. Then*

$$\lim_{k\to\infty} e_k = 0.$$

Proof. Applying the induction argument we can show that

$$e_k \leq \theta^k e_0 + s_k,$$

where $s_k = \sum_{j=1}^{k} \theta^{j-1} \eta_{k-j}$. Under the stated conditions there holds $\lim_{k\to\infty} s_k = 0$ (see [39, Lemma 3.1, (a)]) and we conclude that $\lim_{k\to\infty} e_k = 0$.

□

3 The Method

The method we consider will generate an infinite sequence of iterations x_k. As already stated in each iteration we will use a subsample $\mathcal{N}_k$ and the corresponding function $f_{\mathcal{N}_k}(x_k)$ and the gradient $\nabla f_{\mathcal{N}_k}(x_k)$. The inexact projection $\tilde{\pi}_S$ will be used to generate the search direction with the approximation error controlled by a nonincreasing sequence of positive numbers η_k such that

$$\sum_{k=0}^{\infty} \eta_k^2 \leq \bar{\eta} < \infty. \tag{3.1}$$

Thus, for $y_k = x_k - \nabla f_{\mathcal{N}_k}(x_k)$ we will require that the projection residuals satisfy

$$\|r(y_k)\| \leq \eta_k. \tag{3.2}$$

Clearly, the inexactness of the projection will decrease as k increases and eventually we will approach the feasible set. The search direction p_k will be computed as usual, as the difference of inexact projection $\tilde{\pi}_S(y_k)$ and the current approximation x_k. After the computation of p_k we distinguish two cases. If $N_k < N$ we proceed to determine the step size by the nonmonotone rule (2.10) and define $\bar{x}_k = x_k + t_k p_k$. Adding ε_k in the decrease condition allow us to compute a suitable step size even if the direction is nondecreasing and hence the line search rule (3.5) is always well defined, even without the lower bound t_{min} posed in the mini-batch case of Step S3.

In the case $N_k = N$, i.e., if the full precision in the objective function is needed, the projection could still be inaccurate and hence we might search along infeasible direction. In that case we check if the search direction is decreasing. If yes, we proceed to the line search. Else, if the direction is not sufficiently decreasing, we discard the search direction and take a new projection of the current iteration x_k to get x_{k+1} and terminate the iteration. In this case we will call the iteration unsuccessful.

For all iterations where $N_k < N$ in Step 4 of the algorithm we perform additional sampling, taking a new sample $\mathcal{D}_k$, independently of $\mathcal{N}_k$. As already mentioned, this step is meant to be a computationally cheap measure of progress as $\mathcal{D}_k$ can be arbitrary small and we will work with $D_k = 1$ in our numerical tests. For this new sample $\mathcal{D}_k$ we compute the direction $u_k = x_k - \nabla f_{\mathcal{D}_k}(x_k)$ and then project it approximately to get $s_k = \tilde{\pi}_S(u_k) - x_k = -\nabla f_{\mathcal{D}_k}(x_k) - A^T\tilde{\lambda}_k(u_k)$. Here we keep the same accuracy of the projection as in Step 2 for p_k. Notice that for $N_k = N$ this additional sampling is not needed as the line search in Step 3 is performed with the true objective function.

Finally, at Step 5 we update the iteration. So, if we have enough decrease in the objective function $f_{\mathcal{D}_k}$, according to (3.7), we update the iteration and keep the same sample size for the next iteration. Roughly speaking we are saying here that $f_{\mathcal{N}_k}$ is a good approximation of the objective function as the decrease condition holds for another (independently sampled) function $f_{\mathcal{D}_k}$. Notice that the condition (3.7) is looser than the step size rule (3.5) as ε_k is multiplied with some constant C which can be large.

Algorithm 1: IPAS (Inexact Projection with Additional Sampling)

S0 *Initialization.* Input: $x_0 \in \mathbb{R}^n, N_0 \in \mathbb{N}, \beta, c, c_1, t_{min} \in (0,1), C > 0$, $\{\eta_k\}$ satisfying (3.1), $\{\varepsilon_k\}$ satisfying (2.11), $k := 0$.

S1 *Subsampling.* If $N_k < N$, choose $\mathcal{N}_k$ via (2.9). Else, set $f_{\mathcal{N}_k} = f$.

S2 *Search direction.* Compute

$$p_k = \tilde{\pi}_S(y_k) - x_k = -\nabla f_{\mathcal{N}_k}(x_k) - A^T\tilde{\lambda}_k(y_k) \tag{3.3}$$

with $y_k = x_k - \nabla f_{\mathcal{N}_k}(x_k)$, and $\tilde{\lambda}_k(y_k)$ satisfying (3.2).
If $N_k < N$ go to step S3.

If $N_k = N$, and

$$(\nabla f(x_k))^T p_k \leq -c\|p_k\|^2 \tag{3.4}$$

go to step S3.
Else set $x_{k+1} = \tilde{\pi}_S(x_k)$ with $\tilde{\lambda}_k(x_k)$ satisfying $\|r(x_k)\| \leq \eta_k$, set $k = k+1$ and go to step S1.

S3 *Step size.* If $N_k = N$, find the smallest $j \in \mathbb{N}_0$ such that $t_k = \beta^j$ satisfies

$$f_{\mathcal{N}_k}(x_k + t_k p_k) \leq f_{\mathcal{N}_k}(x_k) + c_1 t_k (\nabla f_{\mathcal{N}_k}(x_k))^T p_k + \varepsilon_k. \tag{3.5}$$

Else, if $N_k < N$, starting with $t_k = 1$, while $t_k \geq t_{min}$ and

$$f_{\mathcal{N}_k}(x_k + t_k p_k) > f_{\mathcal{N}_k}(x_k) + c_1 t_k (\nabla f_{\mathcal{N}_k}(x_k))^T p_k + \varepsilon_k,$$

reduce t_k by factor β.

Set $\bar{x}_k = x_k + t_k p_k$.

S4 *Additional sampling.*
If $N_k = N$, set $x_{k+1} = \bar{x}_k$, $k = k+1$ and go to step S1.
Else choose $\mathcal{D}_k$ via (2.13) and compute

$$s_k = \tilde{\pi}_S(u_k) - x_k = -\nabla f_{\mathcal{D}_k}(x_k) - A^T \tilde{\lambda}_k(u_k) \tag{3.6}$$

with $u_k = x_k - \nabla f_{\mathcal{D}_k}(x_k)$ and $\tilde{\lambda}_k(u_k)$ satisfying $\|r(u_k)\| \leq \eta_k$.

S5 *The update.* If

$$f_{\mathcal{D}_k}(\bar{x}_k) \leq f_{\mathcal{D}_k}(x_k) - c\|s_k\|^2 + C\varepsilon_k, \tag{3.7}$$

set $x_{k+1} = \bar{x}_k$ and $N_{k+1} = N_k$.
Else set $x_{k+1} = x_k$, choose $N_{k+1} \in \{N_k + 1, ..., N\}$.
Set $k = k+1$ and go to step S1.

If the condition (3.7) is not satisfied we take $x_{k+1} = x_k$ and increase the sample size N_{k+1} for the new iteration. Essentially we reason here that the trial iteration should not be accepted because we need more precision and hence we increase the sample size.

Remark. An important point here is to notice that we do not need to specify how to increase the sample size if needed, i.e., we are free to choose any $N_{k+1} > N_k$. Naturally, one can choose to increase the sample size very slowly to keep the iterations computationally cheap, or to increase the sample size faster to achieve a better approximation of the objective function and the gradient, hoping to end up the process in fewer iterations. Clearly, the sample size scheduling is problem dependent.

4 Convergence analysis

Assumption A 1. *Each function $f_i, i = 1, ..., N$ is continuously differentiable with Lispchitz continuous gradient and bounded from below by a constant f_{low}.*

This assumption implies that $f(x) \geq f_{low}$ for all $x \in \mathbb{R}^n$ and it also holds for any approximate function $f_{\mathcal{N}_k}$ and $f_{\mathcal{D}_k}$. Furthermore all approximate gradients are also Lipschitz continuous and without loss of generality we may assume that $L > 0$ is a common Lispchitz constant.

The algorithm we consider generates a set of random iterations $\{x_k\}$. Nevertheless some properties hold for all iterations, independently of the sample we use to generate them. First of all, we can show using the standard arguments and Assumption A1 that the step size t_k generated in Step 3, (3.5) is bounded from below.

Lemma 4.1. *Assume that Assumption A1 holds and that step size t_k is computed in Step 3 of IPAS algorithm. Then $t_k \geq t_{\min}$ provided that $t_{\min} < \min\{1, \frac{2\beta c(1-c_1)}{L}\}$.*

Proof. If the full sample is reached, the line search is performed only if (3.4) holds and it can be proved (see [26] for example) that $t_k \geq \frac{2\beta c(1-c_1)}{L}$. On the other hand, if $N_k < N$, the line search yields $t_k \geq t_{min}$ by the construction of Step 3 of IPAS algorithm. □

Furthermore, we can prove that the feasibility of iterations $\bar{x}_k$ is eventually increasing although the projections are inexact. The following statement holds.

Lemma 4.2. *Assume that Assumption A1 holds. Then*

$$e(\bar{x}_k) \leq (1 - t_{\min})e(x_k) + \eta_k. \tag{4.1}$$

Proof. Given that

$$\begin{aligned} \bar{x}_k &= x_k + t_k p_k = x_k + t_k(\tilde{\pi}_S(y_k) - x_k) \\ &= (1 - t_k)x_k + t_k\tilde{\pi}_S(y_k), \end{aligned}$$

and

$$\begin{aligned} e(\bar{x}_k) &= \|A((1 - t_k)x_k + t_k\tilde{\pi}_S(y_k)) - b\| && (4.2) \\ &\leq (1 - t_k)\|Ax_k - b\| + t_k\|A\tilde{\pi}_S(y_k) - b\| && (4.3) \\ &= (1 - t_k)e(x_k) + t_k e(\tilde{\pi}_S(y_k)). && (4.4) \end{aligned}$$

Since $t_k \geq t_{\min}$ and the residual of $\tilde{\pi}_S(y_k)$ satisfies (3.2), by Lemma 2.2 we have $e(\tilde{\pi}_S(y_k)) \leq \eta_k$. Therefore, the above inequalities imply

$$e(\bar{x}_k) \leq (1 - t_{\min})e(x_k) + \eta_k.$$

□

Let us denote by $\mathcal{D}_k^+$ the subset of all possible outcomes of $\mathcal{D}_k$ at iteration k for which the condition (3.7) is satisfied, i.e.,

$$\mathcal{D}_k^+ = \{\mathcal{D}_k \subset \mathcal{N} \mid f_{\mathcal{D}_k}(\bar{x}_k) \leq f_{\mathcal{D}_k}(x_k) - c\|s_k\|^2 + C\varepsilon_k\}. \tag{4.5}$$

We denote the complementary subset of outcomes at iteration k by

$$\mathcal{D}_k^- = \{\mathcal{D}_k \subset \mathcal{N} \mid f_{\mathcal{D}_k}(\bar{x}_k) > f_{\mathcal{D}_k}(x_k) - c\|s_k\|^2 + C\varepsilon_k\}. \tag{4.6}$$

Although the problem we consider is constrained and the algorithm is quite different from the one in [24, 26] with sampling that is not uniform, the following lemma, similar to the [26, Lemma 1] holds. Essentially, it says that either $N_k = N$ for k large enough or the condition (3.7) is satisfied infinitely many times. We state the proof for completeness.

Lemma 4.3. *Suppose that Assumption A1 holds. If $N_k < N$ for all $k \in \mathbb{N}$, then a.s. there exists $k_1 \in \mathbb{N}$ such that $\mathcal{D}_k^- = \emptyset$ for all $k \geq k_1$.*

Proof. Assume that $N_k < N$ for all $k \in \mathbb{N}$. Since the sample size sequence $\{N_k\}$ in IPAS Algorithm is non-decreasing there exists some $\overline{N} < N$ such that $N_k = \overline{N}$ for all k large enough. Now, let us assume that there is no $k_1 \in \mathbb{N}$ such that $\mathcal{D}_k^- = \emptyset$ for all $k \geq k_1$. Then there exists an infinite sub-sequence of iterations $K \subseteq \mathbb{N}$ such that $\mathcal{D}_k^- \neq \emptyset$ for all $k \in K$. Since $\mathcal{D}_k$ is chosen with finitely many possible outcomes with the same distribution for each k, there exists $q > 0$ such that $\mathbb{P}(\mathcal{D}_k \in \mathcal{D}_k^-) \geq q$ for all $k \in K$. In fact, given (2.13) and the fact that $D_k \leq N - 1$ for each k, we can conclude that $q = (\min_{s\in\{1,2,\ldots,N\}}\{w_s\})^{N-1}$. So, we have

$$\mathbb{P}(\mathcal{D}_k \in \mathcal{D}_k^+, k \in K) \leq \Pi_{k\in K}(1 - q) = 0.$$

Therefore we will almost surely encounter an iteration at which the sample size will be increased due to violation of the condition (3.7). This is a contradiction with the condition $N_k = \overline{N}$ for all k large enough and we conclude that the statement holds. □

As already stated, depending on the problem, IPAS algorithm yields two possibilities - either we generate an infinite sequence $\{x_k\}$ such that $N_k < N$ for all k, or we end up with $N_k = N$ for k large enough. So, the convergence analysis will cover these two cases separately. Let us first consider the mini-batch case, i.e. $N_k < N$ for all $k \in \mathbb{N}$.

The following lemma quantifies the progress in term of the (true) objective function and exact projections in the mini-batch case.

Lemma 4.4. *Suppose that Assumption A1 holds. If $N_k < N$ for all $k \in \mathbb{N}$, then a.s.*

$$f(x_{k+1}) \leq f(x_k) - \frac{c}{2}\|d(x_k)\|^2 + C_L \max\{\varepsilon_k, \eta_k^2\}$$

holds for all $k \geq k_1$ and some constant C_L, where k_1 is as in Lemma 4.3 and $d(x_k)$ is given by (2.7).

Proof. First, we prove that

$$f(\bar{x}_k) \leq f(x_k) - \frac{c}{2}\|d(x_k)\|^2 + C_L \max\{\varepsilon_k, \eta_k^2\}$$

holds a.s. for all $k \geq k_1$, where k_1 is as in Lemma 4.3 and some constant C_L.

Notice that Lemma 4.3 implies that a.s. (3.7) holds for all possible realizations of $\mathcal{D}_k$ and for all $k \geq k_1$. Thus, we conclude that a.s. for every $i = 1, 2, ..., N$ and every $k \geq k_1$ we have

$$f_i(\bar{x}_k) \leq f_i(x_k) - c\|z_k^i\|^2 + C\varepsilon_k, \tag{4.7}$$

where z_k^i denotes the direction obtained for $\mathcal{D}_k = \{i\}$ in Step 4 i.e.,

$$z_k^i = u_k^i - A^T\tilde{\lambda}_k(u_k^i) - x_k \tag{4.8}$$

where $u_k^i = x_k - \nabla f_i(x_k)$ and $\tilde{\lambda}_k(u_k^i)$ satisfies

$$\|AA^T\tilde{\lambda}_k(u_k^i) - Au_k^i + b\| \leq \eta_k. \tag{4.9}$$

Indeed, if there exists $i \in \mathcal{N}$ that violates (4.8), then there would exists at least one possible realization of $\mathcal{D}_k$ (namely, $\mathcal{D}_k = \{i, i, ..., i\}$) that violates (4.7) and thus we would have $\mathcal{D}_k^- \neq \emptyset$. Let us denote the residual by r_k^i, i.e., we have

$$r_k^i = AA^T\tilde{\lambda}_k(u_k^i) - Au_k^i + b, \quad \|r_k^i\| \leq \eta_k, \tag{4.10}$$

and

$$\tilde{\lambda}_k(u_k^i) = (AA^T)^{-1}r_k^i + (AA^T)^{-1}(Au_k^i - b). \tag{4.11}$$

Moreover,

$$z_k^i = u_k^i - A^T(AA^T)^{-1}r_k^i - A^T(AA^T)^{-1}(Au_k^i - b) - x_k. \tag{4.12}$$

Next, multiplying both sides of (4.7) with w_i satisfying (1.2) and summing over all $i \in \mathcal{N}$ we obtain that a.s. the following holds for all $k \geq k_1$

$$f(\bar{x}_k) \leq f(x_k) - c\sum_{i=1}^{N} w_i\|z_k^i\|^2 + C\varepsilon_k. \tag{4.13}$$

Further, writing $x_k = \sum_{i=1}^N w_i x_k$ and using (2.3) in Lemma 2.1 with $y^i = x_k - \nabla f_i(x_k)$ we obtain that the exact projection direction related to the

original objective function can be represented as follows

$$\begin{aligned} d(x_k) &= \pi_S(x_k - \nabla f(x_k)) - x_k \\ &= \pi_S(\sum_{i=1}^N w_i(x_k - \nabla f_i(x_k))) - \sum_{i=1}^N w_i x_k \\ &= \sum_{i=1}^N w_i(\pi_S(x_k - \nabla f_i(x_k)) - x_k) \\ &= \sum_{i=1}^N w_i(z_k^i + \pi_S(x_k - \nabla f_i(x_k)) - x_k - z_k^i) \\ &= \sum_{i=1}^N w_i(z_k^i + \pi_S(u_k^i) - x_k - z_k^i). \end{aligned} \tag{4.14}$$

Further, using (1.2) and the convexity of $\|\cdot\|^2$, we obtain

$$\begin{aligned} \|d(x_k)\|^2 &\le \sum_{i=1}^N w_i\|z_k^i + \pi_S(u_k^i) - x_k - z_k^i\|^2 \\ &\le 2\sum_{i=1}^N w_i\|z_k^i\|^2 + 2\sum_{i=1}^N w_i\|\pi_S(u_k^i) - x_k - z_k^i\|^2. \end{aligned} \tag{4.15}$$

Now, let us estimate $\|\pi_S(u_k^i) - x_k - z_k^i\|^2$. According to (2.1) and (4.12) we obtain

$$\begin{aligned} \pi_S(u_k^i) - x_k - z_k^i &= u_k^i - A^T(AA^T)^{-1}(Au_k^i - b) - x_k \\ &- (u_k^i - A^T(AA^T)^{-1}r_k^i - A^T(AA^T)^{-1}(Au_k^i - b) - x_k) \\ &= A^T(AA^T)^{-1}r_k^i. \end{aligned} \tag{4.16}$$

Thus, for $C_A = \|A^T(AA^T)^{-1}\|$ we get

$$\|\pi_S(u_k^i) - x_k - z_k^i\|^2 \le \|A^T(AA^T)^{-1}\|^2\|r_k^i\|^2 \le C_A^2\eta_k^2. \tag{4.17}$$

Therefore, from (4.15) we obtain

$$\begin{aligned} -\sum_{i=1}^N w_i\|z_k^i\|^2 &\le -\frac{1}{2}\|d(x_k)\|^2 + \sum_{i=1}^N w_i\|\pi_S(u_k^i) - x_k - z_k^i\|^2 \\ &\le -\frac{1}{2}\|d(x_k)\|^2 + C_A^2\eta_k^2. \end{aligned} \tag{4.18}$$

Now, (4.13) implies

$$f(\bar{x}_k) \le f(x_k) - \frac{c}{2}\|d(x_k)\|^2 + cC_A^2\eta_k^2 + C\varepsilon_k =: f(x_k) - \frac{c}{2}\|d(x_k)\|^2 + C_L\max\{\varepsilon_k, \eta_k^2\}. \tag{4.19}$$

Next, we show that in the mini-batch scenario we accept the trial point $\bar{x}_k$ for all $k \geq k_1$. Since we know that a.s. $\mathcal{D}_k^- = \emptyset$ for all $k \geq k_1$, i.e., (3.7) is satisfied, therefore, a.s., for all $k \geq k_1$ the trial point is accepted, i.e., $x_{k+1} = \bar{x}_k$, so the statement holds due to (4.19). □

Now, let us denote by Ω all possible sample paths of the IPAS algorithm. Further, let us denote by $\mathcal{A} \subseteq \Omega$ all possible sample paths that yield mini-batch scenario considered in the previous proposition and by $\bar{\mathcal{A}} \subseteq \Omega$ all possible sample paths that reach the full sample size eventually, i.e., the complement of $\mathcal{A}$. We use the following notation for the corresponding conditional expectations

$$\mathbb{E}_{\mathcal{A}}(\cdot) := \mathbb{E}(\cdot|\mathcal{A}), \quad \mathbb{E}_{\bar{\mathcal{A}}}(\cdot) := \mathbb{E}(\cdot|\bar{\mathcal{A}}).$$

In order to prove the convergence results, we impose the following assumption similar to one in [26].

Assumption A 2. *There exists a constant $C_{\mathcal{A}}$ such that $\mathbb{E}_{\mathcal{A}}(|f(x_{k_1})|) \leq C_{\mathcal{A}}$, where k_1 is as in Lemma 4.3.*

The following result shows that $d(x_k)$ defined in (2.7) converges to zero a.s. in the mini-batch scenario.

Corollary 4.5. *Suppose that the assumptions of Lemma 4.4 hold together with Assumption A2. Then*

$$P(\lim_{k\to\infty} d(x_k) = 0|\mathcal{A}) = 1$$

and every accumulation point of the sequence $\{x_k\}$ is a stationary point of problem (1.1) *a. s.*

Proof. Lemma 4.4 implies that a.s.

$$f(x_{k+1}) \leq f(x_k) - \frac{c}{2}\|d(x_k)\|^2 + C_L \max\{\varepsilon_k, \eta_k^2\}$$

holds for all $k \geq k_1$. Applying the conditional expectation $\mathbb{E}_{\mathcal{A}}$, by the induction argument we obtain that for each j there holds

$$\mathbb{E}_{\mathcal{A}}(f(x_{k_1+j})) \leq \mathbb{E}_{\mathcal{A}}(f(x_{k_1})) - \frac{c}{2}\sum_{i=0}^{j-1}\mathbb{E}_{\mathcal{A}}(\|d(x_{k_1+i})\|^2) + C_L\sum_{i=0}^{j-1}\max\{\varepsilon_{k_1+i}, \eta_{k_1+i}^2\}.$$

Now, by using Assumptions A1, A2, and (3.1), letting j tend to infinity we get

$$f_{low} \leq C_{\mathcal{A}} - \frac{c}{2}\sum_{i=0}^{\infty}\mathbb{E}_{\mathcal{A}}(\|d(x_{k_1+i})\|^2) + C_L \max\{\bar{\varepsilon}, \bar{\eta}\}$$

and we conclude that

$$\sum_{i=0}^{\infty} \mathbb{E}_{\mathcal{A}}(\|d(x_{k_1+i})\|^2) < \infty.$$

Now, by the extended version of Markov's inequality we have that for any $\epsilon > 0$

$$\mathbb{P}(\|d(x_{k_1+i})\| \geq \epsilon|\mathcal{A}) \leq \frac{\mathbb{E}_{\mathcal{A}}(\|d(x_{k_1+i})\|^2)}{\epsilon^2}$$

which implies

$$\sum_{i=0}^{\infty} \mathbb{P}(\|d(x_{k_1+i})\| \geq \epsilon|\mathcal{A}) < \infty$$

and Borel-Cantelli Lemma [20] implies that $P(\lim_{i\to\infty} \|d(x_{k_1+i})\| = 0|\mathcal{A}) = 1$, i.e.,

$$P(\lim_{i\to\infty} d(x_{k_1+i}) = 0|\mathcal{A}) = 1. \tag{4.20}$$

Now, recall that e_k is the measure of infeasibility of x_k defined in (2.6), i.e., $e_k := \|Ax_k - b\|$. Lemma 4.2 implies that

$$e_{k+1} \leq (1 - t_{\min})e_k + \eta_k$$

for all $k \geq k_1$. Therefore, due to Lemma 2.4 we obtain $\lim_{k\to\infty} e_k = 0$. Let us denote by $\tilde{x} = \lim_{k\in K} x_k$ an arbitrary accumulation point of IPAS algorithm in the mini-batch scenario. Then, we have that $\|A\tilde{x} - b\| = \lim_{k\in K} e_k = 0$ and we conclude that $\tilde{x}$ is feasible. Moreover, due to (4.20) and continuity of d, a.s.

$$d(\tilde{x}) = \lim_{k\in K} d(x_k) = 0$$

and we conclude that $\tilde{x}$ is a.s. a stationary point of (1.1) according to Theorem 2.3. □

□

Now, we analyze the case where the full sample is reached for some $k_2 \in \mathbb{N}_0$ and therefore $N_k = N$ for $k \geq k_2$. We will distinguish between two types of iterations for $k \geq k_2$, successful and unsuccessful, represented by sets of indices K_{su} and K_{un}. An iteration x_k, $k \geq k_2$ is unsuccessful if the condition (3.4) does not hold, i.e., if the direction p_k is not sufficiently decreasing and hence $x_{k+1} = \tilde{\pi}_S(x_k)$ and $r(x_k) \leq \eta_k$. Otherwise, x_k is successful and the new iteration is determined by the line search rule (3.5) and $x_{k+1} = \bar{x}_k$.

The following Lemma states that the sequence $\{x_k\}_{k\geq k_2}$ converges to a feasible point and hence each accumulation point is feasible. We will rely on Lemmas 2.2 and 2.4 from Section 2.

Lemma 4.6. *Assume that Assumption A1 holds and let $\{x_k\}$ be an iterative sequence generated by IPAS algorithm such that $N_k = N$ for $k \geq k_2$. Then $\lim_{k\to\infty} e(x_k) = 0$ and each accumulation point is feasible.*

Proof. We will distinguish three different cases: 1) all iterations are successful for k large enough; 2) all iterations are unsuccessful for k large enough; and 3) we have an infinite sequence of successful and an infinite sequence of unsuccessful iterations.

Case 1. Assume, without loss of generality, that for all $k \geq k_2$ we have $k \in K_{su}$. In that case we have $x_{k+1} = \bar{x}_k$ and by Lemma 4.2

$$e(x_{k+1}) \leq (1 - t_{\min})e(x_k) + \eta_k,$$

so the statement holds by Lemma 2.4.

Case 2. Without loss of generality assume that $k \in K_{un}$ for all $k \geq k_2$. Then we have

$$x_{k+1} = \tilde{\pi}_S(x_k)$$

for all k and by Lemma 2.2

$$e(x_{k+1}) \leq \eta_k.$$

Given that $\lim_{k\to\infty} \eta_k = 0$ we obtain $\lim_{k\to\infty} e(x_k) = 0$.

Case 3. Assume now that both K_{su} and K_{un} are infinite. Let $k-1 \geq k_2$ and assume that $k-1 \in K_{un},\ k, \ldots, k+j-1 \in K_{su}$ and $k+j \in K_{un}$. Then we have

$$e(x_k) = e(\tilde{\pi}_S(x_{k-1})) \leq \eta_{k-1}$$

and for each $i = 1, ..., j$, by Lemma 4.2 there holds

$$e(x_{k+i}) \leq \theta e(x_{k+i-1}) + \eta_{k+i-1},$$

with $\theta := 1 - t_{min} \in (0, 1)$ and thus by the induction argument we get

$$e(x_{k+i}) \leq \theta^i \eta_{k-1} + ... + \theta\eta_{k+i-2} + \eta_{k+i-1}.$$

Since $\{\eta_k\}$ is nonincreasing, for each $i = 1, ..., j$ there holds

$$e(x_{k+i}) \leq \eta_{k-1} \sum_{t=0}^{i} \theta^t \leq \eta_{k-1} \sum_{t=0}^{\infty} \theta^t = \eta_{k-1} \frac{1}{1-\theta} = \eta_{k-1} \frac{1}{t_{min}}.$$

Thus, we can conclude that for each $k \geq k_2$ there holds

$$e(x_k) \leq \frac{1}{t_{min}} \eta_{k_{un}},$$

where k_{un} represents the index of last unsuccessful iteration before the iteration k. Letting $k \to \infty$ we conclude that $\lim_{k\to\infty} e(x_k) = 0$ in this case as well.

Therefore, in all cases we have $e(x_k) \to 0$. If $\tilde{x}$ is an arbitrary accumulation point of $\{x_k\}$ then clearly $e(\tilde{x}) = 0$ and hence $\tilde{x}$ is feasible. □

Lemma 4.7. *Assume that Assumption A1 holds and let $\{x_k\}$ be an iterative sequence generated by IPAS algorithm such that $N_k = N$ for $k \geq k_2$ and assume that $\{x_k\}_{k\geq k_2}$ is bounded. If there exists $k_3 \geq k_2$ such that $k \in K_{su}$ for all $k \geq k_3$ then each accumulation point of $\{x_k\}$ is a stationary point of (1.1).*

Proof. For each $k \geq k_3$ we have that x_{k+1} is successful and hence we have

$$\begin{aligned} f(x_{k+1}) &\leq f(x_k) + c_1 t_k (\nabla f(x_k))^T p_k + \varepsilon_k \\ &\leq f(x_k) - c_1 c t_k \|p_k\|^2 + \varepsilon_k \\ &\leq f(x_k) - c_1 c t_{\min} \|p_k\|^2 + \varepsilon_k. \end{aligned}$$

Given that f is continuous and $\{x_k\}_{k\geq k_2}$ is assumed to be bounded, there must exists a constant f_{up} such that $f(x_{k_3}) \leq f_{up}$. Moreover, $f(x_k) \geq f_{low}$ for each k and using the standard arguments we get $\sum_{k=k_3}^{\infty} \|p_k\|^2 < \infty$ and $\lim_{k\to\infty} \|p_k\| = 0$. Let us denote, as before, the exact projected gradient direction at x_k by $d(x_k)$. Then we have

$$\|d(x_k) - p_k\| \leq \|A^T(AA^T)^{-1}\| \|r(x_k)\| \leq C_A \eta_k$$

and

$$\lim_{k\to\infty} \|d(x_k)\| \leq \lim_{k\to\infty} \|p_k\| + \lim_{k\to\infty} \|d(x_k) - p_k\| \leq \lim_{k\to\infty} \|p_k\| + C_A \eta_k = 0.$$

So, for arbitrary accumulation point $\tilde{x} = \lim_{k\in K} x_k$ we have $\|d(\tilde{x})\| = \lim_{k\in K} \|d(x_k)\| = 0$. By Lemma 4.6 $\tilde{x}$ is feasible so the statement follows by Lemma 2.3. □

The following inequality will be used for further analysis.

Lemma 4.8. *For each $k \geq k_2$ there holds*

$$\begin{aligned} &\nabla^T f(x_k) p_k \leq -\|p_k\|^2 - p_k^T(\pi_S(y_k) - \tilde{\pi}_S(y_k)) \\ + \quad &(p_k + \nabla f(x_k) + \pi_S(y_k) - \tilde{\pi}_S(y_k))^T(\pi_S(x_k) - x_k + \tilde{\pi}_S(y_k) - \pi_S(y_k)). \end{aligned}$$

Proof. For any $y \in \mathbb{R}^n$ we have that $\pi_S(y)$ is a solution of convex problem

$$\min_{Az=b} \frac{1}{2}\|z - y\|^2.$$

Denoting $h(z) = \frac{1}{2}\|z - y\|^2$ we can state the KKT conditions as follows. A point z^* is a solution of $\min_{Az=b} h(z)$ iff $(\nabla h(z^*))^T(z - z^*) \geq 0$ for all z such that $Az = b$. The gradient of h is given by $\nabla h(z) = z - y$ and hence the optimality condition yields

$$(\pi_S(y) - y)^T(z - \pi_S(y)) \geq 0, \text{ for all feasible } z. \tag{4.21}$$

Let us now consider $k \geq k_2$ and take $y = y_k = x_k - \nabla f(x_k)$ and $z = \pi_S(x_k)$ in (4.21). We get

$$(\pi_S(y_k) - y_k \pm \tilde{\pi}_S(y_k))^T(\pi_S(x_k) \pm x_k - \pi_S(y_k) \pm \tilde{\pi}_S(y_k)) \geq 0.$$

Recall that $p_k = \tilde{\pi}_S(y_k) - x_k$. The previous inequality actually states

$$(p_k + \nabla f(x_k) + \pi_S(y_k) - \tilde{\pi}_S(y_k))^T(-p_k + \tilde{\pi}_S(y_k) - \pi_S(y_k) + \pi_S(x_k) - x_k) \geq 0$$

and the statement follows. □

Lemma 4.9. *Assume that Assumption A1 holds and let $\{x_k\}$ be an iterative sequence generated by IPAS algorithm such that $N_k = N$ for $k \geq k_2$ and assume that $\{x_k\}_{k \geq k_2}$ is bounded. If the sequence of unsuccessful iterations $\{x_k\}_{k \in K_{un}}$ is infinite then there exists an accumulation point $\tilde{x}$ of $\{x_k\}$ such that $\tilde{x}$ is stationary point for the problem* (1.1).

Proof. First of all let us notice that boundedness of $\{x_k\}$ together with A1 implies that $\{p_k\}$ is bounded as well. Namely, for $y_k = x_k - \nabla f(x_k)$ we have

$$\begin{aligned} p_k &= \tilde{\pi}_S(y_k) - x_k = y_k - A^T\tilde{\lambda}(y_k) - x_k \\ &= -\nabla f(x_k) - A^T((AA^T)^{-1}(Ay_k - b) + (AA^T)^{-1}r(y_k)). \end{aligned}$$

Now, for $C_A = \|A^T(AA^T)^{-1}\|$, having $\|x_k\| \leq C_x, \|\nabla f(x_k)\| \leq C_g$, and therefore $\|y_k\| \leq \|x_k\| + \|\nabla f(x_k)\| \leq C_x + C_g$, with $\|r(y_k)\| \leq \eta_k$ by conditions of IPAS algorithm we get

$$\|p_k\| \leq C_g + C_A(\|A\|(C_x + C_g) + \|b\| + \eta_k) := C_p. \tag{4.22}$$

Let us assume that there exists $\tilde{\varepsilon} > 0$ such that $\|p_k\| \geq \tilde{\varepsilon} > 0$ for all $k \geq k_2$. We will consider unsuccessful iterations, i.e. $k \geq k_2, k \in K_{un}$. For these iterations we have $(\nabla f(x_k))^T d_k > -c\|p_k\|^2$. Lemma 4.8 implies

$$\begin{aligned} 0 \;<\; & -(1-c)\|p_k\|^2 - p_k^T(\pi_S(y_k) - \tilde{\pi}_S(y_k)) \\ +\; & (p_k + \nabla f(x_k) + \pi_S(y_k) - \tilde{\pi}_S(y_k))^T(\tilde{\pi}_S(y_k) - \pi_S(y_k) + \pi_S(x_k) - x_k) \\ \leq\; & -(1-c)\varepsilon^2 + \|p_k\|\|\pi_S(y_k) - \tilde{\pi}_S(y_k)\| \\ +\; & (\|p_k\| + \|\nabla f(x_k)\| + \|\pi_S(y_k) - \tilde{\pi}_S(y_k))\|) \\ & \cdot(\|\tilde{\pi}_S(y_k) - \pi_S(y_k)\| + \|\pi_S(x_k) - x_k\|). \end{aligned} \tag{4.23}$$

As

$$\pi_S(y_k) - \tilde{\pi}_S(y_k) = A^T(AA^T)^{-1}r(y_k),$$

we have

$$\|\tilde{\pi}_S(y_k) - \pi_S(y_k)\| \leq C_A\eta_k. \tag{4.24}$$

Furthermore,

$$\|\pi_S(x_k) - x_k\| = \|x_k - A^T\lambda(x_k) - x_k\| = \|A^T(AA^T)^{-1}(Ax_k - b)\| \leq C_A e(x_k), \tag{4.25}$$

and $e(x_k) \to 0$ by Lemma 4.6. Putting together (4.22), (4.23)-(4.25) we get

$$0 < -(1-c)\tilde{\varepsilon}^2 + C_p C_A \eta_k + (C_p + C_g + C_A\eta_k)(C_A\eta_k + C_A e(x_k)).$$

Taking the limit for $k \in K_{un}, k \to \infty$ in the above inequality we get

$$0 \leq -(1-c)\tilde{\varepsilon}^2$$

which can not be true. Thus there is no $\tilde{\varepsilon} > 0$ such that $\|p_k\| \geq \tilde{\varepsilon} > 0$ for all $k \geq k_2$. Therefore there exists an infinite $K \subset \mathbb{N}$ such that $\lim_{k\in K} p_k = 0$ and therefore $\lim_{k\in K} d(x_k) = 0$, as in the proof of Lemma 4.7. As $\{x_k\}_{k\geq k_2}$ is bounded then there exists $\tilde{K} \subset K$ such that $\lim_{k\in\tilde{K}} x_k = \tilde{x}$ and thus $d(\tilde{x}) = 0$. By Lemma 4.6 $\tilde{x} \in S$ and hence the statement follows by Lemma 2.3.

Theorem 4.10. *Let Assumption A1 holds and assume that $\{x_k\}$ generated by IPAS algorithm is bounded. Then a.s. there exists an accumulation point of $\{x_k\}$ which is a stationary point of* (1.1).

Proof. If $N_k < N$ for all k we have that the sequence $\{x_k\}$ is bounded so Assumption A2 holds and there exists at least one accumulation point of $\{x_k\}$. That point is stationary by Corollary 4.5. In the case of $N_k = N$ for k large enough the statement follows by Lemma 4.7 and Lemma 4.9. □

5 Numerical results

In this section we demonstrate the efficiency of the proposed algorithm through a series of ML experiments. We apply IPAS algorithm to solve equality constrained logistic regression problems. To evaluate IPAS, we compare its performance with two other notable methods - Stochastic Sequential Quadratic Programming method (Stochastic SQP) [4], and ASPEN [23]. Both of these algorithms are designed to deal with equality constrained optimization problems, making them suitable benchmarks for comparison. Additionally, in order to provide better insights into the proposed method's behavior, we compare four different versions of IPAS obtained by specifying some of the key parameters, namely η_k and ε_k, and the sample size update.

The considered logistic regression problems are given by

$$\min_{Ax=b} f(x) := \frac{1}{N}\sum_{i=1}^{N} \log(1 + e^{(-y_i(x^T z_i))}), \tag{5.1}$$

where N is the number of samples, $z_i \in \mathbb{R}^n$ are sample attributes, $y_i \in \{-1, 1\}$ are respective labels, and $A \in \mathbb{R}^{m\times n}$, $b \in \mathbb{R}^m$ define the feasible set.

The equality constraints are simulated to provide a full ranked matrix A with $m = 2n/3$ rows. The algorithm has been implemented and evaluated on 8 different datasets from LIBSVM repository [30]. Each set has been divided into training and test set, in the 4:1 ratio, and the details can be seen in the following table.

	N_{train}	N_{test}	n
MUSHROOM	6499	1625	112
A9A	19507	4877	123
W7A	19754	4938	300
MNIST	9419	2355	780
EPSILON	16000	4000	2000
CIFAR	8000	2000	3072
SVHN	19557	4889	3072
GISETTE	4800	1200	5000

Table 1: Classification dataset details

Performance analysis is conducted by showing the algorithms' optimality gap against computational cost measure on training set (Figures 1-8), and the accuracy measure tracked over both training and test set (Figure 9). More precisely, the optimality measure is given by $\|x_k - x^*\|_2$, where x^* is obtained by using Matlab *fmincon* function with optimality criterion 10^{-4}. The computational cost is modeled by number of scalar products needed to reach iteration k and it is greatly influenced by the sample size behavior. When calculating the number of scalar products, we also take into account the cost of performing inexact projections. We use the Conjugate Gradient method to solve the linear systems inexactly at each iteration, therefore, every time we use the solver, we account for $(m+4) \cdot iter$ number of scalar products, where $iter$ is the number of iterations Conjugate Gradient method made to achieve the tolerance η_k. It is important to emphasize that we are not relying on the "optimal" implementation, but rather we are counting the computational cost that is theoretically derived from the algorithm. This is also the case for ASPEN and STO-SQP implementation, thus a fair comparison of all the considered methods is provided.

The initial point x_0 was fixed for all the tested methods and generated by means of standard Gaussian distribution. For all tested variants of IPAS method we use additional sample size equal to 1, i.e., $|\mathcal{D}_k| = 1$, and the initial subsampling size $N_0 = \lceil 0.01N \rceil$. Furthermore, we set $c_1 = 10^{-4}, \beta = 0.8, c = 10^{-4}$, and $C = 1$. For the descent conditions (3.5) and (3.7), we set the relaxation parameter $\varepsilon_k = (\frac{1}{k^{0.51}})^2$ which satisfies theoretical recommendations. STO-SQP and ASPEN are implemented with the configurations recommended in [4], [23] respectively.

Basic IPAS algorithm uses $\eta_k = \frac{1}{k^{0.51}}$ to control inexact projections,

which yields relatively slow decay of infeasibility. When invoked, the increase of the sample size is done by $N_{k+1} = N_k + 1$. In order to demonstrate the efficacy of inexact projections, we compare IPAS to its version named EXACT which makes uniformly accurate projections with tolerance $\eta_k = 10^{-6}$, while all the other parameters are the same as in basic IPAS. On the other hand, we also test IPAS-R - a version of IPAS which allows relatively big infeasibility by setting $\eta_k = 10000/k^{0.51}$, while keeps the same configuration of the remaining parameters as IPAS. We also examine the influence of the sample size scheduling by testing two more versions of IPAS - IPAS-M and IPAS-H - which update the subsampling size by $N_{k+1} = 1.01N_k$ and $N_{k+1} = 1.1N_k$, respectively. All the other parameters are set as for basic IPAS, including η_k. The following table summarizes the choices of the relevant parameters for tested IPAS versions.

	η_k	ε_k	N_{k+1}
IPAS	$1/k^{0.51}$	$(1/k^{0.51})^2$	$N_k + 1$
IPAS-R	$10000/k^{0.51}$	$(1/k^{0.51})^2$	$N_k + 1$
EXACT	10^{-6}	$(1/k^{0.51})^2$	$N_k + 1$
IPAS-M	$1/k^{0.51}$	$(1/k^{0.51})^2$	$1.01N_k$
IPAS-H	$1/k^{0.51}$	$(1/k^{0.51})^2$	$1.1N_k$

Table 2: IPAS versions' parameters

Figures 1-8 show the behavior of all the consider algorithms applied on problem (5.1) with datasets from Table 1. The graphs on the left show the optimality gap with respect to computational costs (scalar products), on the training set. The graphs in the middle show the portion of the full training sample size used in each iteration, while the graphs on the right track infeasibility measure $e(x_k) = \|Ax_k - b\|_2$ in terms of computational cost. Notice that none of the IPAS configurations reaches full sample size for the duration of the simulation.

In Figures 1, 4, 5 and 8 it can be seen that IPAS-R is closest to the optimal point for the given budget. In Figures 2, 3, 6 and 7 IPAS-R showed worse results, implying that the over-relaxation was not a good choice in these examples. This is most likely due to heterogeneity of the datasets. Nonetheless, basic IPAS provided consistently good results overall. It outperforms ASPEN and STO-SQP in almost all the considered examples. ASPEN is farthest from optimality in most cases, which might be the consequence of high cost of evaluating the penalization parameter for every function evaluation and heterogeneity. This behavior is expected given that ASPEN is constructed for more general problems with nonlinear equality constraints, while IPAS focuses on the linear case. STO-SQP is more competitive and mostly performs better then ASPEN, but it also shows instabilities, especially for EPSILON dataset. Moreover, notice that IPAS also performs bet-

ter then EXACT in most of the cases, which implies that inexact projections influenced the reduction of the overall computational costs.

Considering the sample size behavior, IPAS-H provides the fastest increase as expected, and it is followed by IPAS-M. It is notable that IPAS-H reaches the full sample size in a substantial number of cases, while all the other methods stay well below except for SVHN dataset where IPAS-M also attains the full sample within the given budget. However, this aggressive sample size increase seems to be beneficial in some cases such as for dataset A9A in Figure 2, most probably because of heterogeneous data. Although basic IPAS method showed to be robust and provides a safe choice in general for all the considered problems, it is evident that an the sample size update heavily influences the performance of the algorithm and is is problem dependent. This opens a new research direction for future work.

Finally, Figure 9 shows the training and test accuracies for all the considered datasets. It can be seen that IPAS-R reaches high accuracies, while other IPAS versions also demonstrate large increases in accuracies for the least amount of computational cost for most of the datasets. However, the constraints of the considered problems are simulated and reaching the feasibility may deteriorate the accuracy overall.

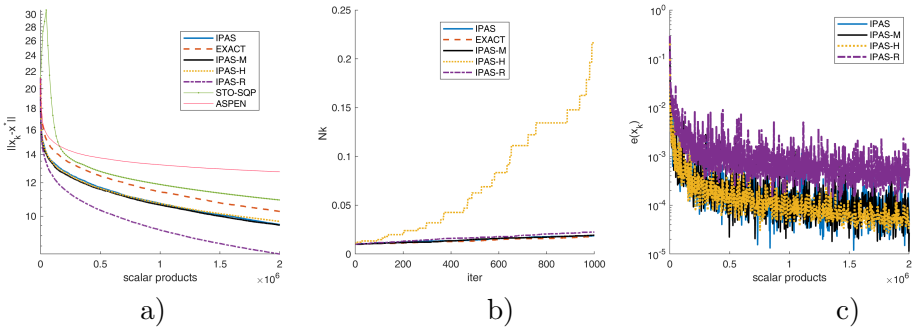


Figure 1: MUSHROOM dataset, $N = 6499, n = 112$. Comparison of algorithms in Table 2 with ASPEN and STO-SQP on the training set: a) optimality gap against computational costs; b) sample size behavior in terms of iterations; c) infeasibility measure against computational costs.

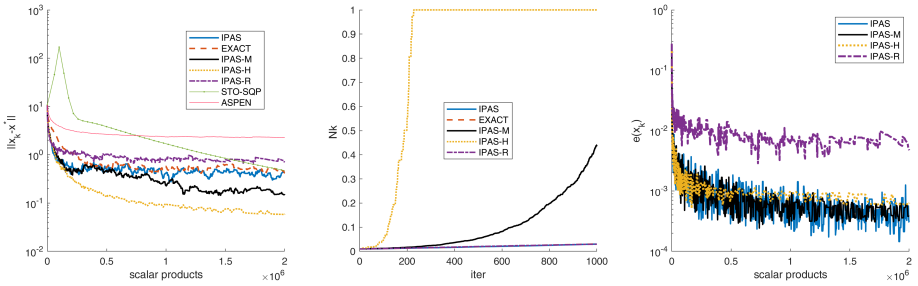


Figure 2: A9A dataset, $N = 19507, n = 123$. Comparison of algorithms in Table 2 with ASPEN and STO-SQP on the training set: a) optimality gap against computational costs; b) sample size behavior in terms of iterations; c) infeasibility measure against computational costs.

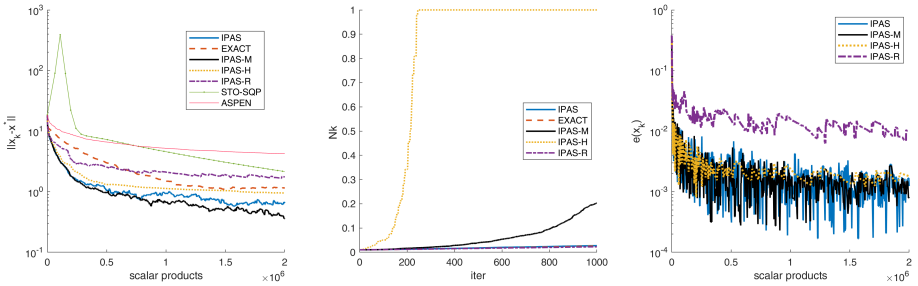


Figure 3: W7A dataset, $N = 19754, n = 300$. Comparison of algorithms in Table 2 with ASPEN and STO-SQP on the training set: a) optimality gap against computational costs; b) sample size behavior in terms of iterations; c) infeasibility measure against computational costs.

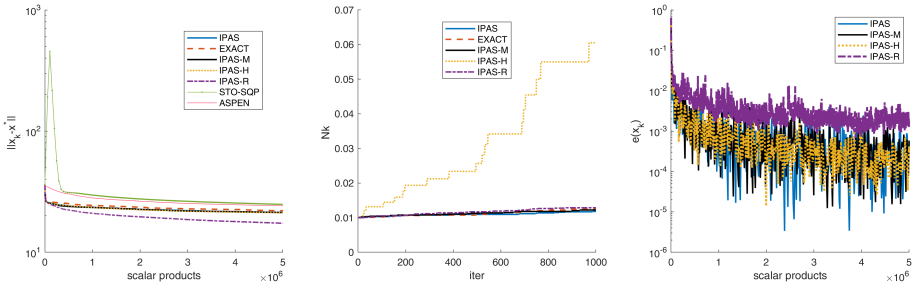


Figure 4: MNIST dataset, $N = 9419, n = 780$. Comparison of algorithms in Table 2 with ASPEN and STO-SQP on the training set: a) optimality gap against computational costs; b) sample size behavior in terms of iterations; c) infeasibility measure against computational costs.

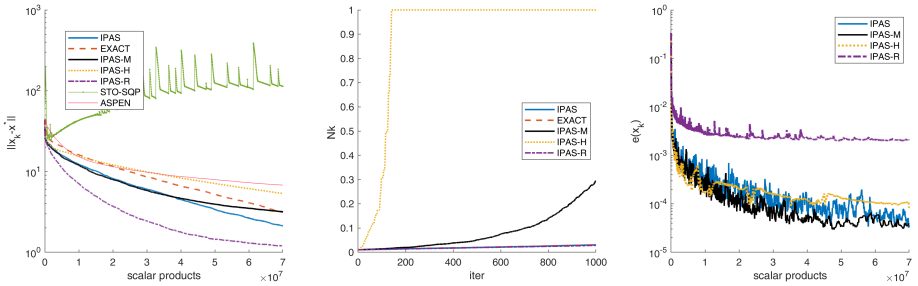


Figure 5: EPSILON dataset, $N = 16000, n = 2000$. Comparison of algorithms in Table 2 with ASPEN and STO-SQP on the training set: a) optimality gap against computational costs; b) sample size behavior in terms of iterations; c) infeasibility measure against computational costs.

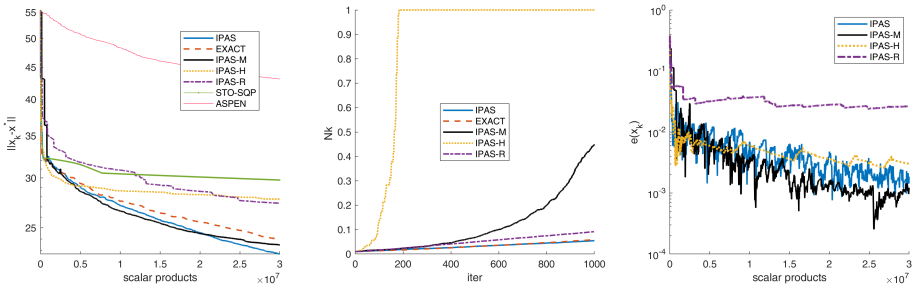


Figure 6: CIFAR dataset, $N = 8000, n = 3072$. Comparison of algorithms in Table 2 with ASPEN and STO-SQP on the training set: a) optimality gap against computational costs; b) sample size behavior in terms of iterations; c) infeasibility measure against computational costs.

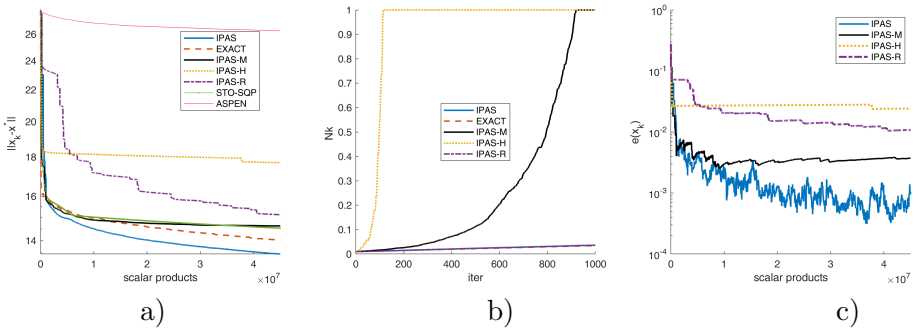
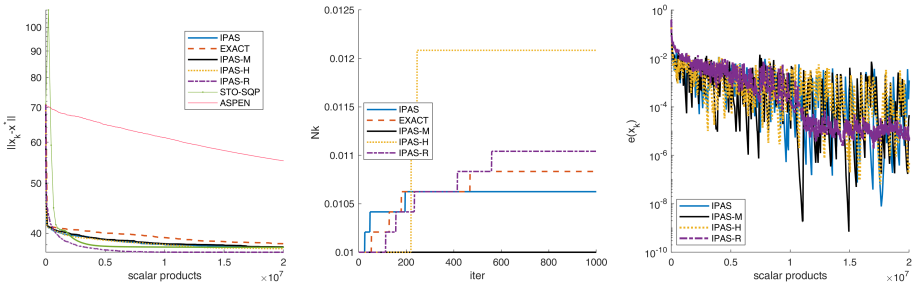


Figure 7: SVHN dataset, $N = 19557, n = 3072$. Comparison of algorithms in Table 2 with ASPEN and STO-SQP on the training set: a) optimality gap against computational costs; b) sample size behavior in terms of iterations; c) infeasibility measure against computational costs.



a) b) c)

Figure 8: GISETTE dataset, $N = 4800, n = 5000$. Comparison of algorithms in Table 2 with ASPEN and STO-SQP on the training set: a) optimality gap against computational costs; b) sample size behavior in terms of iterations; c) infeasibility measure against computational costs.

6 Conclusions

The proposed algorithm represents a novel approach for solving weighted sum problems with possibly large number of linear equality constraints. It adapts additional sampling approach originally constructed for finite sum unconstrained problems to more general problems of the form (1.1). Moreover, inexact projections are allowed, but controlled by a predefined sequence of parameters. Allowing inexact projections shows to be very important in terms of computational costs, especially when the number of constraints is large. The almost sure convergence of the proposed method is proved under a set of standard assumptions for the stochastic framework, without the convexity assumption. Preliminary numerical results on a number of machine learning problems with real-world data and simulated constraints show that IPAS is competitive with the relevant benchmark methods in this field. Possible future work may include fine-tuning for some special subclasses of the considered problems and finding an optimal sample size increase strategy within the proposed framework.

7 Funding

This research was supported by the Science Fund of the Republic of Serbia, GRANT No 7359, Project title - LASCADO.

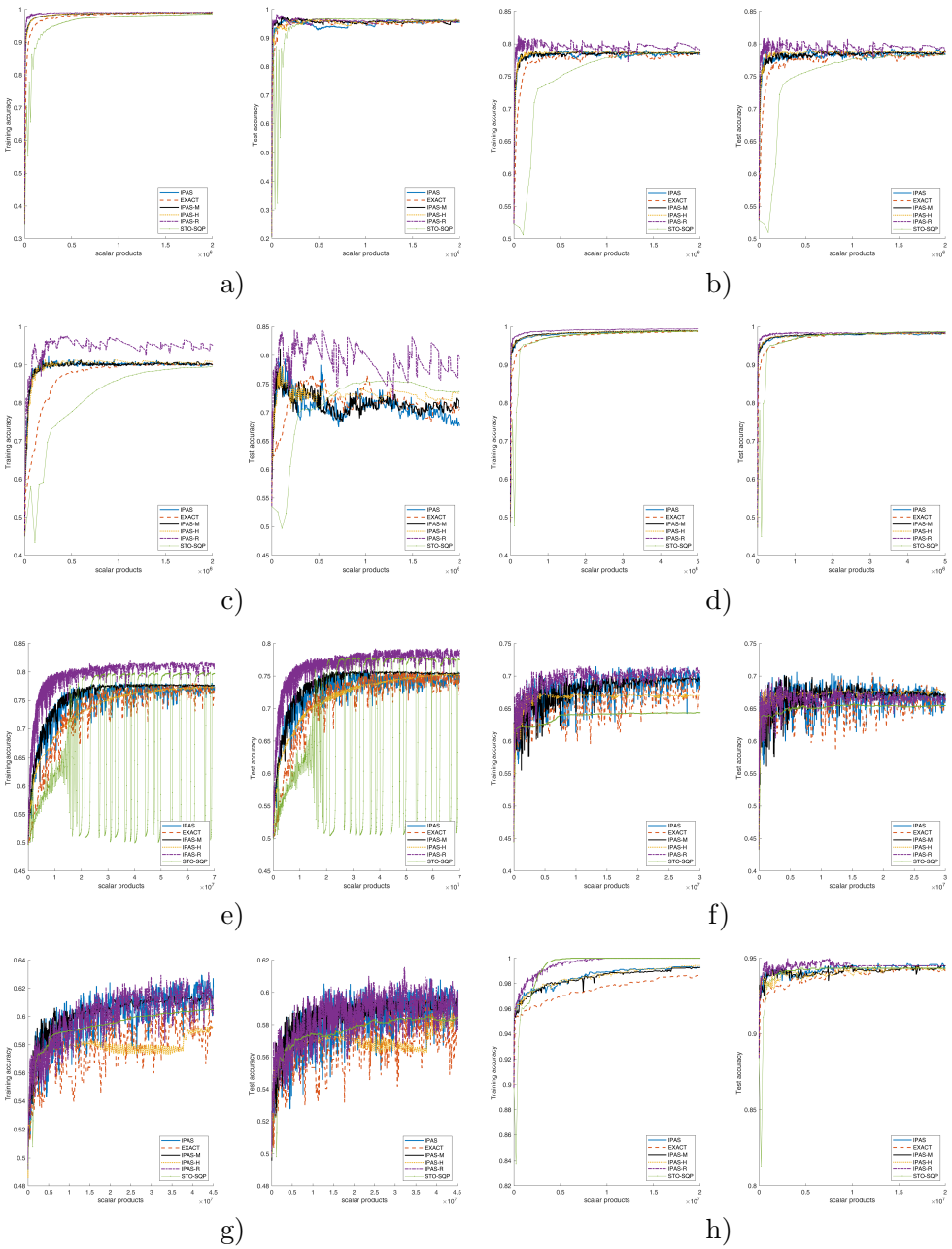


Figure 9: Model evaluation comparison for different datasets. Training error on the left, and test error on the right. Datasets: a) MUSHROOM ; b) A9A ; c) W7A ; d) MNIST ; e) EPSILON ; f) CIFAR ; g) SVHN ; h) GISETTE

References

[1] S. Bellavia, T. Bianconcini, N. Krejić, B. Morini, Subsampled first-order optimization methods with applications in imaging, *Handbook of Mathematical Models and Algorithms in Computer Vision and Imaging, (2021).*

[2] S. Bellavia, N. Krejić, N. Krklec Jerinkić, M. Raydan, SLiSeS: Subsampled Line Search Spectral Gradient Method for Finite Sums, *Optimization Methods and Software, (2024), pp. 1–26, https://doi.org/10.1080/10556788.2024.2426620*

[3] S. Bellavia, N. Krejić, B. Morini, S. Rebegoldi, A stochastic first-order trust-region method with inexact restoration for finite-sum minimization, *Computational Optimization and Applications 84(1), pp. 53–84 (2023), https://doi.org/ 10.1007/s10589-022-00430-7*

[4] A.S. Berahas, F. E. Curtis, D. Robinson, B. Zhou, Sequential Quadratic Optimization for Nonlinear Equality Constrained Stochastic Optimization, *SIAM Journal on Optimization 31 (2021), pp. 1352-1379*

[5] A.S. Berahas, M. Xie, B. Zhou, A sequential quadratic programming method with high probability complexity bounds for nonlinear equality constrained stochastic optimization, *arXiv preprint arXiv:2301.00477 (2024), https://doi.org/10.48550/arXiv.2301.00477*

[6] E.G. Birgin, J.M. Martínez, M. Raydan, Nonmonotone Spectral Projected Gradients on Convex Sets, *SIAM Journal on Optimization 10 (2000) pp. 1196-1211, https://doi.org/10.1137/S1052623497330963.*

[7] J. Blanchet, C. Cartis, M. Menickelly, K. Scheinberg, Convergence rate analysis of a stochastic trust-region method via supermartingales, *INFORMS journal on optimization 1(2) (2019), pp. 92–119, https://doi.org/10.1287/ijoo.2019.0016*

[8] R. Bollapragada, J. Nocedal, D. Mudigere, H.J. Shi, P.T.P. Tang, A progressive batching L-BFGS method for machine learning, *In: International Conference on Machine Learning (2018), pp. 620–629.*

[9] J.J. Brust, R.F. Marcia, C.G. Petra, M.A. Saunders, Large-scale Optimization with Linear Equality Constraints using Reduced Compact Representation, *arXiv:2101.11048 (2021), https://doi.org/10.48550/arXiv.2101.11048*

[10] R. Chen, M. Menickelly, K. Scheinberg, Stochastic optimization using a trust region method and random models, *Mathematical Programming 169(2) (2018), pp. 447–487, https://doi.org/10.1007/s10107-017-1141-8*

[11] F.E. Curtis, D.P. Robinson, B. Zhou, A stochastic inexact sequential quadratic optimization algorithm for nonlinear equality-constrained optimization, *INFORMS Journal on Optimization 6 (3-4)(2024)*

[12] D. Di Serafino, N. Krejić, N. Krklec Jerinkić, M. Viola, LSOS: Line-search Second-Order Stochastic optimization methods for nonconvex finite sums, *Mathematics of Computation 2023.*

[13] D.Z. Du, F. Wu, X.S. Zhang, On Rosen's gradient projection methods. *Ann Oper Res 24, 9–28 (1990). https://doi.org/10.1007/BF02216813*

[14] Y.Fang, S. Na, M. Mahoney,M. Kolar, Fully Stochastic Trust-Region Sequential Quadratic Programming for Equality-Constrained Optimization Problems, *SIAM Journal on Optimization 34 (2024), pp. 2007 - 2037*

[15] C. Gambella, B. Ghaddar, J. Naoum-Sawaya, Optimization problems for machine learning: A survey, *European Journal of Operational Research 290 (2021), pp. 807–828.*

[16] S. Gratton, Ph.L.Toint, S2MPJ and CUTEst optimization problems for Matlab, Python and Julia *arXiv:2407.07812*

[17] L. Grippo, F. Lampariello, S. Lucidi, A nononotone line search technique for Newton's method, *SIAM Journal on Numerical Analysis 23(4) (1986), pp. 707-716, https://doi.org/10.1137/0723046.*

[18] T. Hastie, R. Tibshirani, J. Friedman, Elements of Statistical Learning, *Springer, 2009.*

[19] A. N. Iusem, A. Jofré, R. I. Oliveira, P. Thompson, Variance-based extragradient methods with line search for stochastic variational inequalities, *SIAM Journal on Optimization 29(1) (2019), pp. 175–206, https://doi.org/10.1137/17M1144799.*

[20] A. Klenke, Probability Theory: A Comprehensive Course, third edition, 2020 *Springer Nature link, eBook ISBN 978-3-030-56402-5, https://link.springer.com/book/10.1007/978-3-030-56402-5*

[21] N. Krejić, N. Krklec Jerinkić, Nonmonotone line search methods with variable sample size, *Numer. Algorithms 68(4) (2015), pp. 711-739, https://doi.org/10.1007/s11075-014-9869-1.*

[22] N. Krejić, N. Krklec Jerinkić, Spectral projected gradient method for stochastic optimization, *Journal of Global Optimization 73 (2018), pp. 59–81, https://doi.org/10.1007/s10898-018-0682-6.*

[23] N. Krejić, N. Krklec Jerinkić, T. Ostojić, N. Vučićević, ASPEN: An Additional Sampling Penalty Method for Finite-Sum Optimization Problems with Nonlinear Equality Constraints, *arXiv:2508.02299 (2025), https://doi.org/10.48550/arXiv.2508.02299*

[24] N. Krejić, N. Krklec Jerinkić, Martinez, A., Yousefi, M., A non-monotone trust-region method with noisy oracles and additional sampling, *Computational Optimization and Applications (2024) 89, pp. 247–278, https://doi.org/10.1007/s10589-024-00580-w*

[25] N. Krejić, Z. Lužanin, Z. Ovcin, I. Stojkovska, Descent direction method with line search for unconstrained optimization in noisy environment, *Optimization Methods and Software 30(6) (2015), pp. 1164-1184, https://doi.org/10.1080/10556788.2015.1025403.*

[26] N. Krklec Jerinkić, V. Ruggiero, I. Trombini, Spectral Stochastic Gradient Method with Additional Sampling for Finite and Infinite Sums, *Computational Optimization and Applications, (2025), https://doi.org/10.1007/s10589-025-00664-1*

[27] G. Lan, T. Li, Y. Xu, Projected gradient methods for nonconvex and stochastic optimization: new complexities and auto-conditioned stepsizes, *arXiv:2412.14291 (2024), https://doi.org/10.48550/arXiv.2412.14291*

[28] D.H. Li, M. Fukushima, A derivative-free line search and global convergence of Broyden-like method for nonlinear equations, *Opt. Methods Software 13 (2000), pp. 181-201, DOI:10.1080/10556780008805782.*

[29] M. Lichman, UCI machine learning repository (2013), *https://archive.ics. uci.edu/ml/index.php*

[30] LIBSVM Data: Classification (Binary Class), *https://www.csie.ntu.edu.tw/∼cjlin/libsvmtools/datasets/binary.html*

[31] Q. Lin, R. Ma, T. Yang, *Proceedings of the 35th International Conference on Machine Learning 80 (2018), PMLR, pp. 3112-3121.*

[32] J. Linderoth, A. Shapiro, S. Wright, The empirical behavior of sampling methods for stochastic programming. *Ann Oper Res 142, 215–241 (2006). https://doi.org/10.1007/s10479-006-6169-8*

[33] M. Loreto, A. Crema, Convergence analysis for the modified spectral projected subgradient method,*Optimization Letters 9(5) (2015), pp. 915-929, https://doi.org/10.1007/s11590-014-0792-0.*

[34] G. Negiar, G. Dresdner, A. Tsai, L. El Ghaoui, F. Locatello, R. Freund, F. Pedregosa, *Proceedings of the 37th International Conference on Machine Learning 119 (2020), PMLR, pp. 7253-7262.*

[35] F. Oztoprak, R. Byrd, J. Nocedal, Constrained optimization in the presence of noise, *SIAM Journal on Optimization 33(3) (2023), pp. 2118–2136*

[36] H. Robbins, S. Monro, A stochastic approximation method, *The annals of mathematical statistics, pp. 400-407,(1951).*

[37] J. Scott, M. Tůma, Solving large linear least squares problems with linear equality constraints, *Bit Numer Math 62 (2022), pp. 1765–1787*

[38] S. Sun, J. Nocedal, A Trust-Region Algorithm for Noisy Equality Constrained Optimization, *arXiv:2411.02665 (2024), https://doi.org/10.48550/arXiv.2411.02665*

[39] S. Sundhar Ram, A. Nedić, V.V. Veeravalli, Distributed Stochastic Subgradient Projection Algorithms for Convex Optimization, *J. Optim. Theory Appl. (2010) 147: 516–545*

[40] Q. Wang, C. Piermarini, Y. Zhu, F.E. Curtis, Projected Stochastic Momentum Methods for Nonlinear Equality-Constrained Optimization for Machine Learning, *arXiv:2601.11795 (2026), https://doi.org/10.48550/arXiv.2601.11795*

[41] Y. Zeng, J. Bai, S. Wanga, Z. Wanga, X. Shena, A hybrid stochastic alternating direction method of multipliers for nonconvex and nonsmooth composite optimization, *European Journal of Operational Research 329 (2026), pp. 63–78.*