Probabilistic Iterative Hard Thresholding for Sparse Learning *

Matteo Bergamaschi[†], Andrea Cristofari[‡], Vyacheslav Kungurtsev[§], and Francesco Rinaldi[†]

Abstract. For statistical modeling wherein the data regime is unfavorable in terms of dimensionality relative to the sample size, finding hidden sparsity in the ground truth can be critical in formulating an accurate statistical model. The so-called "ℓ_0 norm", which counts the number of non-zero components in a vector, is a strong reliable mechanism of enforcing sparsity when incorporated into an optimization problem. However, in big data settings wherein noisy estimates of the gradient must be evaluated out of computational necessity, the literature is scant on methods that reliably converge. In this paper we present an approach towards solving expectation objective optimization problems with cardinality constraints. We prove convergence of the underlying stochastic process, and demonstrate the performance on two Machine Learning problems.

Key words. cardinality constraint, stochastic optimization

MSC codes. 68Q25, 68R10, 68U05

1 Introduction

In this paper we consider the cardinality constrained expectation objective problem,

$$\begin{array}{ll} \min\limits_{x\in\mathbb{R}^n} & f(x) := \mathbb{E}[F(x,\xi)] \\ \text{s.t.} & \|x\|_0 \le K, \end{array} \tag{1.1}$$

where $f(\cdot)$ is $L(f)$ continuously differentiable. We say that $x \in C_K$ if $\|x\|_0 \le K$ and thus a feasible x corresponds to $x \in C_K$.

This optimization problem is particularly important in applications of data science. In particular, the expectation objective serves to quantify the minimization of some empirical loss function that enforces the fit of a statistical model fit to empirical data. Cardinality constraints enforce sparsity in the model, enabling the discovery of the most salient features as far as prediction accuracy.

Cardinality constraints present a significant challenge to optimization solvers. The so-called (as it is not, formally) zero norm is a discontinuous function that results in a highly nonconvex and disconnected feasible set, as well as an unusual topology of stationary points and minimizers [21, 22]. Algorithmic development has been, as similar to many such problems, a parallel endeavor from the mathematical optimization and the machine learning communities. When dealing with a deterministic objective function, procedures attuned to the structure of the problem and seeking stationary points of various strength are presented, for instance, in [3]. Methods for deterministic optimization problems with sparse symmetric sets are proposed in, e.g., [4, 18], while methods for deterministic optimization problems with both cardinality and nonlinear constraints are described in, e.g., [8, 9, 10, 14, 24, 23, 25]. Simultaneously works appearing in machine learning conferences, e.g., [31, 30, 27, 19], exhibit weak theoretical convergence guarantees, but appear to scale more adequately as far as numerical experience. Thus, an algorithm that enjoys both reliable performance together with strong

*Submitted to the editors September 2, 2024.
Funding: This work was funded by the European Union's Horizon Europe research and innovation programme under grant agreement No. 101084642.
[†]Department of Mathematics "Tullio Levi-Civita", University of Padua (bergamas@math.unipd.it, rinaldi@math.unipd.it),
[‡]Department of Civil Engineering and Computer Science Engineering, University of Rome "Tor Vergata" (andrea.cristofari@uniroma2.it),
[§]Czech Technical University in Prague (kunguvya@fel.cvut.cz),

theoretical guarantees, as sought for the high dimensional high data volume model fitting problems in contemporary data science, is as of yet unavailable.

In this paper we attempt to reconcile these two and present an algorithm that is associated with reasonably strong theoretical convergence guarantees, while at the same time able to solve large scale problems of interest in statistics and machine learning. To this end we present a procedure under the framework of *Probabilistic Models*, which can be understood as a sequential linear Sample Average Approximation (SAA) scheme for solving problems with statistics in the objective function. First introduced in [2], then rediscovered with extensive analysis in [1, 15], this approach can exhibit asymptotic (and even worst case complexity) results to a local minimizer of the original problem, while still allowing the use of Newton-type second order iterations of subproblem solutions, and thus faster convergence as far as iteration count. The use of probabilistically accurate estimates within a certain bound in these methods permit a rather flexible approach to estimating the gradient, including techniques that introduce bias, while foregoing the necessity of a stepsize asymptotically diminishing to zero. However, asymptotic accurate convergence still requires increasing the batch size, so the tradeoffs in precision and certainty relative to computation become apparent, and adaptive for the user, in deciding at which point to stop the algorithm and return the current iterate as an estimate of the solution.

As contemporary Machine Learning applications, we shall consider Adversarial Attacks (see, e.g., [11, 16, 26] and references therein for further details), and Probabilistic Graphical Model training (see, e.g., [5, 28] and references therein for further details). In this paper we shall see how the use of a stochastic gradient and hard sparsity constraint can improve the performance and model quality in the considered problems.

The paper is organized as follows: In Section 2, we introduce some basic definitions and preliminary results related to optimality conditions of problem (1.1) that ease the theoretical analysis. We then describe the details of the proposed algorithmic scheme in Section 3. We then prove almost sure convergence to suitable stationary points in Section 4. Numerical results on some relevant Machine Learning applications are reported Section 5. Finally, we draw some conclusions and discuss some possible extensions in Section 6.

2 Background

Cardinality constrained optimization presents an extensive hierarchy of stationarity conditions, as due to the geometric complexity of the feasible set. This necessitates specialized notions of projection, and presents complications due to the projection operation's generic non-uniqueness.

Definitions and Preliminaries The active and inactive set of a vector $x \in \mathbb{R}^n$ are respectively denoted by

$$I_{\mathcal{A}}(x) := \{i \in \{1, ..., n\},\, x_i = 0\}, \quad I_{\mathcal{I}}(x) := \{i \in \{1, ..., n\},\, x_i \neq 0\}.$$

A set T is a *super-support* of $x \in C_K$ if $I_{\mathcal{A}}(x) \subseteq T$ and $|T| = s$. Let the permutation group of $\{1, ..., n\}$ be denoted as Σ_n and for a permutation $\sigma \in \Sigma_n$, we write $(x^\sigma)_i = x_{\sigma(i)}$. For a vector $x \in \mathbb{R}^n$ we denote with $M_i(x)$ the i-th largest absolute-value component of x, thus we have $M_1(x) \le M_2(x) \le \cdots \le M_n(x)$.

We finally define the orthogonal projection as

$$P_{C_K}(x) = \arg\min\{\|z - x\|^2, z \in C_K\},$$

that is, an n-length vector consisting of the s components of x with the largest absolute value. Such operator, as already highlighted in the previous section, is not single-valued due to the inherent non-convexity of the set C_K and plays a critical role in the development of algorithms for sparsity constrained optimization (see, e.g., [3, Section 2] for a discussion on this matter).

Optimality Conditions Now we define several optimality conditions for (1.1), borrowing heavily from [3]. Observe that a notable characteristic of cardinality constrained optimization is the presence of a hierarchy of optimality conditions, that is, a number of conditions that hold at optimal points that range across levels of restriction.

When restricted to a specific support, the "no descent directions" rule still provides a necessary optimality condition, which is referred to as basic feasibility. For a full support, this condition aligns with the standard stationarity condition, but only applies within the support set. If the support is not full, the stationarity condition must hold for any potential full support set that includes the given support, that is the gradient needs to be zero.

Definition 2.1. *$x^* \in C_K$ is* Basic Feasibile *(BF) for problem* (1.1) *when*
1. $\nabla f(x^*) = 0$, *if* $\|x^*\|_0 < K$,
2. $\nabla f_i(x^*) = 0$ *for all* $i \in I_{\mathcal{I}}(x^*)$, *if* $\|x^*\|_0 = K$.

We thus have that when a point $x^* \in C_K$ is optimal for problem (1.1), then x^* is a BF point (see Theorem 1 in [3]). The BF property is however a relatively weak necessary condition for optimality. Consequently, stronger necessary conditions are required to achieve higher quality solutions. This is why we use L-stationarity, an extension of the stationarity concept for convex constrained problems.

Definition 2.2. *$x^* \in C_K$ is L -* stationary *for problem* (1.1) *when*

$$x^* \in P_{C_K}\left(x^* - \frac{1}{L}\nabla f(x^*)\right). \tag{2.1}$$

An equivalent analytic property of L-stationarity is given by the following lemma.

Lemma 2.3. *[3, Lemma 2.2] L-stationarity at x^* is equivalent to $\|x^*\|_0 \leq K$ and*

$$|\nabla_i f(x^*)| \begin{cases} \leq LM_K(x^*) & i \in I_{\mathcal{A}}(x^*) \\ = 0 & i \in I_{\mathcal{I}}(x^*) \end{cases}.$$

The next result relates L-stationarity and Basic Feasibility:

Corollary 2.4. *[3, Corollary 2.1] Suppose that $x^* \in C_k$ is an L-stationary of problem* (1.1) *for some L. Then x^* is BF for problem* (1.1).

In addition, the likely intuition that the L-stationarity is related to the gradient Lispchitz constant is correct:

Theorem 2.5. *[3, Theorem 2.2] If x^* is an optimal solution for problem* (1.1) *then it is L-stationary for all $L > L(f)$.*

To see the distinction between BF and L-stationary, we can consider that if the Lipschitz constant of f is 1, then $x^* = (1, 0)$ with $\nabla f(x^*) = (-10, 1)$ satisfies BF but not L-stationarity. In particular it is clear from a linearization that $f((0, y)) < f(x^*)$ for y small.

In this sense L-stationarity is stronger than a linearized feasible direction stationarity measure, as constructed in [20]. This is because any feasible path for an active component, that is a direction from which a zero component becomes non-zero, would require a discrete jump from another component, that is the assignment of zero to a different component, in order to maintain the constraint. Thus there is no feasible linearized direction in which a zero component becomes non-negative on which to consider possible descent when the cardinality constraint is active. L-stationarity enables a relaxation of this by considering Lipschitz bounds on how much the function value can change along various directions depending on the gradient vector components.

Iterative Hard Thresholding. An important component of particularly machine learning literature procedures to solve (1.1) is the *Hard Thresholding Operator* (see, e.g., [3, 7] for further details). Consider the operator $\mathbf{HT}^x(v)$ applied to a vector v as one that projects v onto the sparsity constraint, i.e.,

$$\mathbf{HT}^x(v) \in \arg\min_{w} \{\|v-w\|,\ \|w\|_0 \le K\} := P_{C_K}(v)\,. \tag{2.2}$$

3 Algorithm

Rolling Projection Estimator Recall that, the sparse projection operation $P_{C_K}(v)$ for a vector v amounts to performing a sorting operation $\sigma \in \tilde{\Sigma}(v)$ on $\sigma(v)$, and then keeping the K largest magnitude components of v while setting the rest to zero.

Observe that an algorithmic iterative descent procedure would involve the negative of the gradient of f or an estimate thereof. Indeed, as the objective function is an expectation, we do not have access to the exact value of the $\nabla f(x)$ and hence the magnitude ranking of the its components. Thus, we must by necessity use noisy gradient estimates $\nabla F(x,\hat{\xi})$ to attempt to estimate the actual ranking of component magnitudes.

Asymptotically, we want to ensure that this sparse projector estimates the true ranking at any limit point. Given the natural source of asymptotically increasing sample sizes, this present a natural opportunity to use the Algorithm iterate sequence itself to perform this estimate, ultimately relying on consistency for statistical guarantees on accurate identification.

Let x_k correspond to the current iterate. Now we define our particular sequential estimate of the ranking of the magnitude of the vector components of the gradient of $f(x_k)$. Specifically, we are given a noisy evaluation $g_k \approx \nabla f(x_k)$, and an application

$$\sigma_k(g_k) \in \tilde{\Sigma}\left(x_k - \alpha \min\left\{1, \frac{\delta_k}{\alpha\,\|g_k\|}\right\} g_k\right). \tag{3.1}$$

At the same time, there exists a set of permutations $S_k = \{\sigma^{(j)}\}_{j\in[J]}$, $\sigma^{(j)} \in \Sigma_n$ with coefficient weights $\{\omega^{(j)}\}_{j\in[J]}$, $\omega \in \Delta_J$.

We now perform exponential smoothing (exponential moving average) on the estimate, with smoothing parameter α_s:

$$\begin{aligned} &\sigma_k = \sigma^{(j)} \in S_k, \Longrightarrow \begin{cases} \omega \leftarrow (1-\alpha_s)\omega, \\ \omega^{(j)} \leftarrow \omega^{(j)} + \alpha_s, \end{cases} \\ &\sigma_k \notin S_k \Longrightarrow \begin{cases} S_k \leftarrow S_k \cup \{\sigma_k\}, \\ \omega \leftarrow (1-\alpha_s)\omega, \\ \omega^{(|S_k|)} \leftarrow \alpha_s. \end{cases} \end{aligned} \tag{3.2}$$

This accomplishes the following: We maintain a set of possible permutations with associated mixture weights. With each new iteration, we sort the components of the noisy gradient estimate. If this sorting permutation has been found before, then we add to a weight corresponding to that permutation and lower the weights of others. Otherwise, i.e. this is a new permutation, we add it to the list of options.

Now, let $\hat{\sigma}^k$ be such that

$$\hat{\sigma}^k = \sigma^{(j)} \in S_k \text{ with } \omega^{(j)} = \arg\max_{l\in[S_k]} \omega^{(l)}. \tag{3.3}$$

Thus, rather than taking the maximal components based on the current sorting, we use the moving average historical estimate. Then, taking

$$I_k = \operatorname{supp}\left(\max_K \hat{\sigma}^k\right), \tag{3.4}$$

that is, the set of indices whose components are largest, we present the **Pseudo Hard Thresholding** operator corresponding to iteration k, defined as follows:

$$\mathbf{HT}^{x,\delta,k}(v) \in \arg\min_{w} \left\{ \|v-w\|,\ w_{[n]\setminus I_k} = 0,\ \|w-x\| \le \delta \right\}. \tag{3.5}$$

We take a *clipped* step, wherein we step in the negative direction of the scaled negative gradient $-\alpha \min\left\{1, \frac{\delta_k}{\alpha\|g_k\|}\right\} g_k$, with α being a positive constant. The Pseudo-Hard Thresholding algorithm can be computed in a straightforward closed form expression:

$$[\hat{x}_k]_i = \begin{cases} 0 & i \notin I_k \\ \left[x_k - \alpha \min\left\{1, \frac{\delta_k}{\alpha\|g_k\|}\right\} g_k\right]_i & i \in I_k. \end{cases} \tag{3.6}$$

From (3.6), observe that

$$\hat{x}_k = P_{I_k}\left(x_k - \alpha \min\left\{1, \frac{\delta_k}{\alpha\|g_k\|}\right\} g_k\right). \tag{3.7}$$

This presents an opportunity to get a sort of descent lemma in the context of cardinality constrained optimization problems. To this end, define

$$h_k(y) = f(x_k) + g_k^T(y - x_k), \tag{3.8}$$

so that

$$h_k(\hat{x}_k) - h_k(x_k) = g_k^T(\hat{x}_k - x_k) \le -\frac{1}{\alpha}\max\left\{1, \frac{\alpha\|g_k\|}{\delta_k}\right\}\|\hat{x}_k - x_k\|^2 \le -\frac{1}{\alpha}\|\hat{x}_k - x_k\|^2, \tag{3.9}$$

where the first inequality follows from known results on the projection operator [6].

Accuracy Estimates

Definition 3.1. *Define $s_k = \hat{x}_k - x_k$. The function estimates f_k^0 and f_k^s are ε_f-accurate estimates of $f(x_k)$ and $f(x_k + s_k)$, respectively, for a given δ_k if*

$$|f_k^0 - f(x_k)| \le \varepsilon_f \delta_k^2 \quad \text{and} \quad |f_k^s - f(x_k + s_k)| \le \varepsilon_f \delta_k^2. \tag{3.10}$$

Definition 3.2. *The model for generating the iterate is κ-δ_k, or (κ_f, κ_g)-δ_k accurate, when*

$$\|\nabla F(y) - g_k\| \le \kappa_g \delta_k \quad \text{and} \quad \left|f(y) - f(x_k) - g_k^T(y - x_k)\right| \le \kappa_f \|y - x_k\|\delta_k^2 \tag{3.11}$$

for all $y \in B(x_k, \delta_k)$.

Note that this implies:

$$\|[\nabla F(y) - g_k]_{I_k}\| \le \kappa_g \delta_k \quad \text{and} \quad \left|f(y) - f(x_k) - [g_k]_{I_k}^T[y - x_k]_{I_k}\right| \le \kappa_f \|y - x_k\|\delta_k^2 \tag{3.12}$$

for all $y \in B(x_k, \delta_k)$.

4 Convergence Theory

Now we develop our argument for justifying the long term convergence of the Algorithm based on classic arguments on probabilistic models given in [15](see also [1, 13]). To this end, we remark that the iterates, being dependent on random function and gradient estimates, define a stochastic process X_k. The Algorithm itself is a realization, thus denoting $x_k = X_k(\omega)$, $\delta_k = \Delta_k(\omega)$, etc. for ω the random element defining the realization. Similar as to the original, we can consider a filtration with the sigma algebra $\mathcal{F}_k$ defining the start of the iteration, and $\mathcal{F}_{k+\frac{1}{2}}$ defining the algebra after the minibatch has been sampled and g_k computed. This filtration will be implicit in the statements of the convergence results.

We begin with a standard assumption on a probability bound on the accuracy of the conditions given by Definition 3.1 and 3.2. To this end define θ, β to be the probability that a given sample of g_k.

Algorithm 3.1 Probabilistic Iterative Hard Thresholding

Initialization: $x_0 \in C_K$, $\delta_0 \in (0, \delta_{max}]$, Parameters $\delta_{max} > 0$, $\gamma \in (0, 1)$.
for $k = 0, 1, 2, \dots$ **do**
  Sample a minibatch $\xi_k \sim \Xi$ and compute $g_k = \nabla F(x_k, \xi_k)$
  Compute σ^k by (3.1) and update ω by (3.2)
  Compute $\hat{\sigma}^k$ from (3.3) and use it to define I_k by (3.4).
  Compute $\hat{x}_k$ from the Pseudo-Hard-Thresholding (3.6)
  Compute stochastic estimates $f_k^s \approx f(\hat{x}_k)$, $f_k^0 \approx f(x_k)$
  if $\dfrac{f_k^0 - f_k^s}{\|[g_k]_{I_k}\|\delta_k} \geq \eta_1$ and $\|[g_k]_{I_k}\| \geq \eta_2 \delta_k$ **then**
    Set $\delta_{k+1} = \min\{\gamma\delta_k, \delta_{max}\}$, let $x_{k+1} = \hat{x}_k$
  else
    Set $\delta_{k+1} = \gamma^{-1}\delta_k$, let $x_{k+1} = x_k$
  end if
end for

Assumption 4.1. *Given $\theta, \beta \in (0, 1)$ and ε_f, there exist κ_g, κ_f such that the sequence of $\{g_k\}$ is such that with probability θ, κ-δ_k-accuracy holds as per Definition 3.2, and with probability β, ε_f accuracy holds as by Definition 3.1.*

We can consider that [3, Lemma 3.1] provides for the enforcement of function decrease in the favorable probabilistic cases in the convergence theory. Indeed, one can derive the following lemma which also functionally corresponds to [15, Lemma 4.5].

Lemma 4.2. *If the model for generating the iterate k is κ-δ_k accurate according to Definition 3.2, with $\hat{x}_k$ and δ_k being such that*

$$\delta_k \leq \frac{1}{2\alpha\kappa_g\delta_{max}}\|x_k - \hat{x}_k\|, \tag{4.1}$$

then

$$f(x_k) - f(\hat{x}_k) \geq \frac{1}{2\alpha}\|\hat{x}_k - x_k\|^2. \tag{4.2}$$

Proof. Using the definition of h_k given in (3.8), we can write

$$\begin{aligned} f(\hat{x}_k) - f(x_k) &= f(\hat{x}_k) - h_k(\hat{x}_k) + h_k(\hat{x}_k) - h_k(x_k) + h_k(x_k) - f(x_k) \\ &= f(\hat{x}_k) - h_k(\hat{x}_k) + h_k(\hat{x}_k) - h_k(x_k) \\ &= f(\hat{x}_k) - f(x_k) - g_k^T(\hat{x}_k - x_k) + g_k^T(\hat{x}_k - x_k) \\ &\leq \kappa_g\|x_k - \hat{x}_k\|\delta_k^2 + g_k^T(\hat{x}_k - x_k), \end{aligned}$$

where the inequality follows from the second condition in (3.11). Using (3.9), we also have that

$$g_k^T(\hat{x}_k - x_k) \leq -\frac{1}{\alpha}\|\hat{x}_k - x_k\|^2.$$

Then, we obtain

$$f(\hat{x}_k) - f(x_k) \leq \kappa_g\|x_k - \hat{x}_k\|\delta_k^2 - \frac{1}{\alpha}\|\hat{x}_k - x_k\|^2 \leq \kappa_g\delta_{max}\|x_k - \hat{x}_k\|\delta_k - \frac{1}{\alpha}\|\hat{x}_k - x_k\|^2, \tag{4.3}$$

where the last inequality follows from the fact that $\delta_k \leq \delta_{max}$. Moreover, (4.1) implies that

$$\kappa_g\delta_{max}\|x_k - \hat{x}_k\|\delta_k \leq \frac{1}{2\alpha}\|\hat{x}_k - x_k\|^2.$$

Using this inequality in (4.3), the desired result follows. ∎

Now, taking inspiration from [15, Lemma 4.6], we can bound the decrease with respect to the projected real gradient.

Lemma 4.3. *If the model for generating the iterate k is κ-δ_k accurate according to Definition 3.2 and*

$$\delta_k \le a\left\| x_k - P_{I_k}\left(x_k - \alpha \min\left\{1, \frac{\delta_k}{\alpha\|\nabla f(x_k)\|}\right\}\nabla f(x_k)\right)\right\|, \tag{4.4}$$

where

$$a = \frac{1}{2\alpha\kappa_g\delta_{max} + 2\sqrt{K}} \tag{4.5}$$

and

$$\alpha > \frac{\sqrt{K}}{\kappa_g\delta_{max}}, \tag{4.6}$$

then

$$f(x_k) - f(\hat{x}_k) \ge c\left\| x_k - P_{I_k}\left(x_k - \alpha \min\left\{1, \frac{\delta_k}{\alpha\|\nabla f(x_k)\|}\right\}\nabla f(x_k)\right)\right\|^2, \tag{4.7}$$

with

$$c = \frac{1 - 4a\sqrt{K}}{2\alpha} > 0.$$

Proof. We can write

$$\begin{aligned}&\left\| x_k - P_{I_k}\left(x_k - \alpha \min\left\{1, \frac{\delta_k}{\alpha\|\nabla f(x_k)\|}\right\}\nabla f(x_k)\right)\right\| \le \\ &\|x_k - \hat{x}_k\| + \left\| \hat{x}_k - P_{I_k}\left(x_k - \alpha \min\left\{1, \frac{\delta_k}{\alpha\|\nabla f(x_k)\|}\right\}\nabla f(x_k)\right)\right\|.\end{aligned}$$

Using (3.7), we get

$$\begin{aligned}&\left\| x_k - P_{I_k}\left(x_k - \alpha \min\left\{1, \frac{\delta_k}{\alpha\|\nabla f(x_k)\|}\right\}\nabla f(x_k)\right)\right\| = \\ &\|x_k - \hat{x}_k\| + \left\| \alpha \min\left\{1, \frac{\delta_k}{\alpha\|g_k\|}\right\}[g_k]_{I_k} - \alpha \min\left\{1, \frac{\delta_k}{\alpha\|\nabla f(x_k)\|}\right\}[\nabla f(x_k)]_{I_k}\right\| = \\ &\|x_k - \hat{x}_k\| + \delta_k\left\| \min\left\{\frac{\alpha\|g_k\|}{\delta_k}, 1\right\}\frac{[g_k]_{I_k}}{\|g_k\|} - \min\left\{\frac{\alpha\|\nabla f(x_k)\|}{\delta_k}, 1\right\}\frac{[\nabla f(x_k)]_{I_k}}{\|\nabla f(x_k)\|}\right\| \le \\ &\|x_k - \hat{x}_k\| + 2\sqrt{K}\delta_k,\end{aligned} \tag{4.8}$$

where the last inequality follows from the fact that $\|u - v\| \le \sqrt{K}\|u - v\|_\infty \le 2\sqrt{K}$ for all $u, v \in \mathbb{R}^K$ such that $\|u\| = \|v\| = 1$. From (4.4), the first term in (4.8) is greater of equal to δ_k/a, leading to

$$\frac{\delta_k}{a} \le \|x_k - \hat{x}_k\| + 2\sqrt{K}\delta_k.$$

Using the definition of a given in (4.5), it follows that (4.1) is satisfied and we can apply Lemma 4.2, obtaining

$$f(x_k) - f(\hat{x}_k) \geq \frac{1}{2\alpha}\|\hat{x}_k - x_k\|^2. \tag{4.9}$$

Finally, in order to lower bound the right-hand side term in the above inequality, using (4.8) we can write

$$\begin{aligned}
\|x_k - \hat{x}_k\|^2 \geq & \left(\left\|x_k - P_{I_k}\left(x_k - \alpha \min\left\{1, \frac{\delta_k}{\alpha\|\nabla f(x_k)\|}\right\}\nabla f(x_k)\right)\right\| - 2\sqrt{K}\delta_k\right)^2 \\
\geq & \left\|x_k - P_{I_k}\left(x_k - \alpha \min\left\{1, \frac{\delta_k}{\alpha\|\nabla f(x_k)\|}\right\}\nabla f(x_k)\right)\right\|^2 + \\
& - 4\sqrt{K}\delta_k\left\|x_k - P_{I_k}\left(x_k - \alpha \min\left\{1, \frac{\delta_k}{\alpha\|\nabla f(x_k)\|}\right\}\nabla f(x_k)\right)\right\| \\
\geq & \left(1 - 4a\sqrt{K}\right)\left\|x_k - P_{I_k}\left(x_k - \alpha \min\left\{1, \frac{\delta_k}{\alpha\|\nabla f(x_k)\|}\right\}\nabla f(x_k)\right)\right\|^2,
\end{aligned}$$

where the last inequality follows from (4.4). From (4.6), it also follows that $c > 0$, thus leading to the desired result. ∎

The next lemma states conditions on δ_k to guarantee that an iteration is successful, similarly as in [15, Lemma 4.7].

Lemma 4.4. *If, at iteration k, the estimates f_k^0, f_k^s are ε_f-accurate according to Definition 3.1 and the model is κ-δ_k accurate according to Definition 3.2, with*

$$\delta_k \leq \min\left\{\frac{1}{\eta_2}, \frac{1-\eta_1}{2\varepsilon_f + \kappa\delta_{max}}\right\}\|[g_k]_{I_k}\|,$$

then the step is accepted.

Proof. Define

$$\rho_k = \frac{f_k^0 - f_k^s}{\|[g_k]_{I_k}\|\delta_k}.$$

Using (3.10) and (3.11), we can write

$$\begin{aligned}
\rho_k &= \frac{f_k^0 - f(x_k)}{\|[g_k]_{I_k}\|\delta_k} + \frac{f(x_k) - f(\hat{x}_k)}{\|[g_k]_{I_k}\|\delta_k} + \frac{f(\hat{x}_k) - f_k^s}{\|[g_k]_{I_k}\|\delta_k} \\
&\leq \frac{2\varepsilon_f\delta_k}{\|[g_k]_{I_k}\|} + \frac{[g_k]_{I_k}^T[\hat{x}_k - x_k]_{I_k} + \kappa_g\|\hat{x}_k - x_k\|\delta_k^2}{\|[g_k]_{I_k}\|\delta_k} \\
&\leq \frac{2\varepsilon_f\delta_k}{\|[g_k]_{I_k}\|} + 1 + \frac{\kappa_g\delta_{max}\delta_k}{\|[g_k]_{I_k}\|},
\end{aligned}$$

where the last inequality follows from the fact that $\|\hat{x}_k - x_k\| \leq \delta_k$ and $\delta_k \leq \delta_{max}$. Then

$$|\rho_k - 1| \leq \frac{(2\varepsilon_f + \kappa_g\delta_{max})\delta_k}{\|[g_k]_{I_k}\|} \leq 1 - \eta_1,$$

where we have used the assumption on δ_k in the last inequality. Hence, $\rho_k \geq \eta_1$. Since we have also assumed that $\|[g_k]_{I_k}\| \geq \eta_2\delta_k$, from the instructions of the algorithm (see line 8 of Algorithm 3.1) it follows that the step is accepted. ∎

Lemma 4.5. *If the estimates f_k^0, f_k^s at iteration k are ε_f-accurate according to Definition 3.1 with $\epsilon_f < (\eta_1\eta_2)/2$ and the step is accepted, then*

$$f(x_{k+1}) - f(x_k) \leq -C\|\delta_k\|^2,$$

with $C = \eta_1\eta_2 - 2\epsilon_f > 0$.

Proof. Since the step is accepted, from the instructions of the algorithm (see line 8 of Algorithm 3.1) we can write

$$f_k^0 - f_k^s \geq \eta_1\|[g_k]_{I_k}\|\delta_k \geq \eta_1\eta_2\delta_k^2. \tag{4.10}$$

Moreover,

$$f(x_k + s_k) - f(x_k) = f(x_k + s_k) - f_k^s + f_k^s - f_k^0 + f_k^0 - f(x_k) \leq 2\epsilon_f\delta_k^2 - \eta_1\eta_2\delta_k^2,$$

where the inequality follows from (3.10) and (4.10). Then, using the definition of C given in the assertion, the desired result follows. ∎

Now we define the stochastic process

$$\Phi_k := \nu f(x_k) + (1-\nu)\delta_k^2. \tag{4.11}$$

The next Theorem is along the lines of Theorem 4.11 in [15]. The result requires a compactness assumption, which we present first.

Assumption 4.6. *Let $\mathcal{L}$ be the level set of the iterates generated by the algorithm, that is,*

$$\mathcal{L} = \{x : f(x) \leq f(x_k)\}, \forall\ x_k$$

noting that this depends on the stochastic realization of the iterates and gradient estimates. Assume that $\mathcal{L}$ is bounded below and that f is L-Lipschitz and its gradient is L-Lipschitz continuous on $\mathcal{L}$.

Theorem 4.7. *Let $\{x_k\}$ be the sequence of iterates generated by the Probabilistic Iterative Hard Thresholding Algorithm (Algorithm 3.1) under Assumption 4.1, and moreover assume that the function and iterates are such that Assumption 4.6 holds. Also assume that the step acceptance parameter η_2 satisfies*

$$\eta_2 \geq 3\kappa_f\alpha \tag{4.12}$$

and the function accuracy parameter ε_f satisfying,

$$\varepsilon_f \leq \min\{\kappa_f, \eta_1\eta_2\}. \tag{4.13}$$

Then it holds that the sequence of trust region radii $\{\delta_k\}$ satisfy the summability condition

$$\sum_{k=0}^{\infty}\delta_k^2 < \infty \tag{4.14}$$

almost surely.

Proof. We define the constants ζ together with ν appearing in (4.11) as satisfying,

$$\zeta \geq \max\left\{a^{-1}, \kappa_g + \max\left\{\eta_2, \frac{2\epsilon_f + \kappa_g\delta_{max}}{1-\eta_1}\right\}\right\}, \tag{4.15}$$

where we recall that

$$a = \frac{1}{2\alpha\kappa\delta_{max} + 2\sqrt{K}}$$

and

$$\frac{\nu}{1-\nu} > \max\left\{\frac{4\gamma^2}{\zeta c}, \frac{4\gamma^2}{\eta_1\eta_2}, \frac{\gamma^2}{\kappa_f}\right\}, \tag{4.16}$$

with c defined by Lemma 4.3.

We observe that on successful, or accepted, iterations,

$$\Phi_{k+1} - \Phi_k \leq \nu(f(x_{k+1}) - f(x_k)) + (1-\nu)(\gamma^2 - 1)\delta_k^2 \tag{4.17}$$

and on unsuccessful iterations,

$$\Phi_{k+1} - \Phi_k \leq (1-\nu)\left(\frac{1}{\gamma^2} - 1\right)\delta_k^2 < 0. \tag{4.18}$$

Let us define the event sequence I_k as the satisfaction of model accuracy according to Definition 3.2:

$$\|\nabla F(y) - g_k\| \leq \kappa\delta_k, \quad \text{and} \quad |f(y) - f(x_k) - g_k^T(y - x_k)| \leq \kappa\|y - x_k\|\delta_k^2 \quad \forall y \in B(x_k, \delta_k).$$

And J_k is defined as the satisfaction of function evaluation accuracy according to Definition 3.1:

$$|f_k^0 - f(x_k)| \leq \varepsilon_f\delta_k^2, \quad \text{and} \quad |f_k^s - f(x_k + s_k)| \leq \varepsilon_f\delta_k^2.$$

Now we break down the different cases of an approximate stationarity condition denoted as:

$$\|(\nabla f(x_k))_{I_k}\| \leq \epsilon,$$

Case 1 $\|(\nabla f(x_k))_{I_k}\| \geq \zeta\delta_k$

We examine the following subcases based on different events:

(a) $I_k \cap J_k$: The model g_k satisfies the κ-δ_k accuracy condition as well as having ε_f accurate function evaluations. Applying (4.15),

$$\|(\nabla f(x_k))_{I_k}\| \geq \delta_k/a.$$

Rearranging, we obtain

$$\delta_k \leq a\|(\nabla f(x_k))_{I_k}\| \leq \frac{a\max\{\delta_k, \alpha\|(\nabla f(x_k))_{I_k}\|\}}{\alpha}$$

Notice that this implies (4.4), that is,

$$\delta_k \leq a\left\|x_k - P_{I_k}\left(x_k - \alpha\min\left\{1, \frac{\delta_k}{\alpha\|\nabla f(x_k)\|}\right\}\nabla f(x_k)\right)\right\|,$$

and so we can apply Lemma 4.3 to conclude that

$$f(x_k) - f(\hat{x}_k) \geq \frac{1}{2\alpha}\|\hat{x}_k - x_k\|^2.$$

Moreover, due to model accuracy it holds that

$$\|g_k\| \geq \|\nabla f(x_k)\| - \kappa_g\delta_k \geq (\zeta - \kappa_g)\delta_k \geq \min\left\{\frac{1}{\eta_2}, \frac{1-\eta_1}{2\varepsilon_f + \kappa\delta_{max}}\right\}\delta_k.$$

As such, we can apply Lemma 4.5 to conclude that the step is accepted and Lemma 4.3 to conclude that the stochastic process proceeds as

(4.19)
$$\begin{aligned}\Phi_{k+1}-\Phi_k &\le -\nu c\delta_k \left\| x_k - P_{I_k}\left(x_k - \alpha\min\left\{1, \tfrac{\delta_k}{\alpha\|\nabla f(x_k)\|}\right\}\nabla f(x_k)\right)\right\| + (1-\nu)(\gamma^2-1)\delta_k^2 \\ &\le \left[-\nu c\zeta + (1-\nu)(\gamma^2-1)\right]\delta_k^2 < 0\end{aligned}$$

where the second inequality uses the case assumption.

(b) $I_k \cap J_k^c$: The function values f_k^0, f_k^s do not satisfy the ε_f-accuracy condition, while model accuracy still holds. In this case the same argument as part a holds, with the caveat that erronous function estimates could lead to a step rejection. In that case, the change in the stochastic process is bounded by (4.18), that is,

$$\Phi_{k+1}-\Phi_k = (1-\nu)\left(\frac{1}{\gamma^2}-1\right)\delta_k^2 < 0.$$

(c) $I_k^c \cap J_k$: If the step is unsuccessful then again we can apply (4.18). Otherwise, with accurate function estimates, we know from Lemma 4.5 together with (4.13) that in this case

$$\Phi_{k+1}-\Phi_k \le \left[-\nu\eta_1\eta_2 + (1-\nu)(\gamma^2-1)\right]\delta_k^2,$$

which is still bounded by (4.18) on account of (4.16).

(d) $I_k^c \cap J_k^c$ In this case, standard Lipschitz arguments give the following bound on the increase in the value of Φ:

$$\Phi_{k+1}-\Phi_k \le \nu C_L \|\left(\nabla f(x_k)\right)_{I_k}\|\delta_k + (1-\nu)(\gamma^2-1)\delta_k^2,\; C_L := \left(1+\frac{3L}{2\zeta}\right).$$

We can finally combine these results to obtain, using the definitions of the probabilities θ and β,

$$\begin{aligned}\mathbb{E}\left[\Phi_{k+1}-\Phi_k|\mathcal{F}_k\right] \le &\theta\beta[-\nu c\|\|(\nabla f(X_k))_{I_k}\|]\Delta_k + (1-\nu)(\gamma^2-1)\Delta_k] \\ &+ [\theta(1-\beta)+(1-\theta)\beta](1-\nu)\left(\frac{1}{\gamma^2}-1\right)\Delta_k^2 \\ &+ (1-\theta)(1-\beta)\left[C_L\|\left(\nabla f(X_k)\right)_{I_k}\|\delta_k + (1-\nu)(\gamma^2-1)\Delta_k^2\right].\end{aligned}$$

We can observe that we can proceed along the same lines as the proof of Case 1 in [15, Theorem 4.11] to conclude that with θ,β chosen to satisfy

$$\frac{(\theta\beta-1/2)}{(1-\theta)(1-\beta)} \ge \frac{C_L}{c}, \tag{4.20}$$

we can apply (4.16) to obtain that both

$$\mathbb{E}\left[\Phi_{k+1}-\Phi_k|\mathcal{F}_k, \{\|\left(\nabla f(x_k)\right)_{I_k}\| \ge \zeta\Delta_k\}\right] \le -\frac{1}{4}c\nu\|\nabla f(X_k)\|\Delta_k \tag{4.21}$$

and

$$\mathbb{E}\left[\Phi_{k+1}-\Phi_k|\mathcal{F}_k, \{\|\left(\nabla f(x_k)\right)_{I_k}\| \ge \zeta\Delta_k\}\right] \le -\frac{1}{2}(1-\nu)(\gamma^2-1)\Delta_k^2. \tag{4.22}$$

Case 2: $\|(\nabla f(x_k))_{I_k}\| < \zeta\delta_k$

If $\|g_k\| < \eta\delta_k$ then (4.18) holds. Now assume that $\|g_k\| \geq \eta_2\delta_k$. We again examine the following subcases based on different events:

(a) $I_k \cap J_k$: The model g_k satisfies the κ-δ_k accuracy condition as well as having ε_f accurate function evaluations. In this case, since it cannot be ensured that the step is accepted, we can apply the argument of Case 1c to conclude that again (4.18) holds.

(b) $I_k \cap J_k^c$: The function values f_k^0, f_k^s do not satisfy the ε_f-accuracy condition, while model accuracy still holds. An unsucessful iteration yields (4.18) a successful iteration satisfies

$$f(x_k)-f(x_{k+1}) = f(x_k)-h_k(x_k)+h_k(x_k)-h_k(\hat{x}_k)+h_k(\hat{x}_k)-f(\hat{x}_k) \leq (\eta_2/\alpha-2\kappa_f)\delta_k^2 \geq \kappa_f\delta_k^2$$

with (4.12) responsible for the last inequality. Finally (4.16) implies (4.18) holds again.

(c) $I_k^c \cap J_k$: It is the same as Case 1c .

(d) $I_k^c \cap J_k^c$: It is the same as Case 1d.

Now, with θ, β chosen such that

$$(1-\theta)(1-\beta) \leq \frac{\gamma^2-1}{\gamma^4-1+2\gamma^2 C_L \zeta \frac{\nu}{1-\nu}}, \tag{4.23}$$

we follow similar arguments to obtain

$$\mathbb{E}\left[\Phi_{k+1}-\Phi_k|\mathcal{F}_k, \{\|\left(\nabla f(x_k)\right)_{I_k}\| < \zeta\Delta_k\}\right] \leq -\frac{1}{2}(1-\nu)\left(1-\frac{1}{\gamma^2}\right)\Delta_k^2. \tag{4.24}$$

Finally, combining the two cases yields that

$$\mathbb{E}\left[\Phi_{k+1}-\Phi_k|\mathcal{F}_k\right] \leq -\sigma\Delta_k^2$$

with $\sigma > 0$, and the theorem has been proven. ∎

We may proceed now to the main and final result. The rest of the original convergence argument can be applied directly to $\|(\nabla f(x_k))_{I_k}\|$. However, recall that this is not the object that is of primary interest. We are indeed interested in proving that the proposed algorithm gives us a point satisfying some suitable optimality condition with high probability.

Theorem 4.8. *Almost surely,*

$$\lim_{k\to\infty} \|(\nabla f(x_k))_{I_k}\| = 0. \tag{4.25}$$

Moreover, for θ sufficiently large, if it holds that, almost surely, for any limit point x^* of a realization of iterates $\{x_k\}$ satisfying

$$|\nabla f(x^*)|_{\sigma(K)} \geq |\nabla f(x^*)|_{\sigma(K+1)} + \chi, \text{ with } \chi > 0, \tag{4.26}$$

it holds that, for some S, for all $k \geq S$,

$$I_k = I_{\mathcal{I}}(x^*) = I_{\mathcal{I}}\left(x^* - \frac{1}{L}\nabla f(x^*)\right) \tag{4.27}$$

and x^* satisfies L-stationarity. Moreover at least one such limit point exists.

Proof. The first part of the statement follows directly from the identical arguments in [15, Theorem 16, Lemma 17, Theorem 18].

For the second statement: first observe that $\Delta_k \to 0$ almost surely and thus $\|X_{k+1}-X_k\|$ almost surely, and so on a set of dense probability, $\{X_k\}$ is a Cauchy sequence. As such, for any realization there exists a limit point x^* satisfying $x_k \to x^*$. Now fix the realization for the remainder of the proof.

We compare the ranking of the gradient components, that is $\sigma(\{|g_i|\})$, $\sigma \in \bar{\Sigma}(\{|[g_k]_i|\})$ to $\sigma(\{|(\nabla f(x^*))_i|\})$. To begin with we see that for the subsequence $\mathcal{S}_g$ wherein the model is $\kappa - \delta$ accurate we have that $k \in \mathcal{S}_g$ iterations satisfy

$$([g_k]_i - [\nabla f(x_k)]_i) + ([\nabla f(x_k)]_i - [\nabla f(x^*)]_i) \to 0$$

where the first summand goes to zero from $\delta_k \to 0$ and the second from the continuity of ∇f and the convergence of $x_k \to x^*$. Thus for sufficiently large $\bar{S}$, for $k \geq \bar{S}$ and $k \in \mathcal{S}_g$, it holds that

$$[|g_k|]_i > |\nabla f(x^*)|_{\sigma(K+1)} + \chi/2$$

for $i \in I_{\mathcal{I}}(x^*)$, and

$$[|g_k|]_i < |\nabla f(x^*)|_{\sigma(K)} + \chi/2$$

for $i \in I_{\mathcal{A}}(x^*)$. Thus, with probability θ, $\sigma_k \in \bar{\Sigma}_k$ satisfies that $\sigma_k[1:K] = I_{\mathcal{I}}(x^*)$.

When θ is sufficiently large, it holds that for $k \geq \bar{S}$ sufficiently large, by smoothing properties [17], $\hat{\sigma}^k$ satisfies $\{\hat{\sigma}^k_{(1)}, \cdots, \hat{\sigma}^k_{(K)}) = I_{\mathcal{I}}(x^*)$.

This together with Lemma 2.3 proves the statement (4.27). ∎

The restriction on θ is just that $\theta > \frac{1}{2}$ if all the components are separated, i.e.,

$$[|\nabla f(x^*)|]_{\sigma(1)} > [|\nabla f(x^*)|]_{\sigma(2)} > [|\nabla f(x^*)|]_{\sigma(3)} > \cdots > [|\nabla f(x^*)|]_{\sigma(n)}$$

A larger θ would be necessary otherwise, in case ties prevent a unique $\hat{\sigma}^k$.

5 Numerical Results

In this section, we present two machine learning applications of the algorithm 3.1: adversarial attacks on neural networks and the reconstruction of sparse Gaussian graphical models. The implementation was carried out using the Python programming language, using the `NumPy`, `Keras`, `Tensorflow`, `scikit-learn`, and `Pandas` libraries. The hyperparameters were selected as follows: $\eta_1 = 10^{-4}$, $\eta_2 = 10^{-4}$, $\delta_0 = 1$, $\delta_{\max} = 10$, and $\gamma = 2$. All the experiments were conducted on a machine equipped with an 11th Gen Intel(R) Core(TM) i7-1165G7 CPU @ 2.80GHz (1.69 GHz). The code is available at https://github.com/Berga53/Probabilistic_iterative_hard_thresholding.

Both applications involve high-dimensional data, making the use of the Pseudo Hard Thresholding operator, as defined in 3, computationally expensive. For practical implementation, we instead utilize the classic Hard Thresholding operator [3]. However, tests on smaller instances have shown that the two operators perform similarly when a suitable value of α_s is chosen.

5.1 Adversarial Attacks on Neural Networks

Adversarial attacks are techniques used to craft imperceptible perturbations that, when added to regular data inputs, induce misclassifications in neural network models. These perturbations are typically designed to evade human detection while successfully fooling the model's classification process. One of the most powerful type of adversarial attack is the Carlini and Wagner [12], characterized by the following formulation:

$$\begin{aligned} &\min_{\delta} D(x, x+\delta) + c \cdot f(x+\delta) \\ &\text{such that } x+\delta \in [0,1]^n \end{aligned} \tag{5.1}$$

with δ being the perturbation, D being usually the ℓ_2 or ℓ_0 distance, and

$$f(x) = \left(\max_{i \neq t}(F(x)_i) - F(x)_t\right)^+.$$

Using our algorithm, we can incorporate the ℓ_0 penalty directly in the constraint, so our final formulation of the problem is

$$\begin{aligned} &\min_{\|\delta\|_0 \le K} \|\delta\|_2 + c \cdot f(x+\delta) \\ &\text{such that } x+\delta \in [0,1]^n \end{aligned} \tag{5.2}$$

In practice, this allows us to decide how many pixels to perturb during the attack. While usual attacks are trained against selected samples of the dataset, in this paper, we will demonstrate a universal adversarial attack: the attack is performed against the entirety of the dataset, producing only one global perturbation. We will show that, in both targeted and untargeted attacks, we can significantly lower a model's accuracy using very few pixels. We tested the attack on the MNIST dataset, which consists of 60,000 images of handwritten digits (0-9) that are 28 $\times$ 28 pixels in size. We performed both targeted and untargeted attacks. In the targeted attack, we aimed to misclassify the images into a specific class, while in the untargeted attack, we simply aimed to cause any misclassification. However, the untargeted attack is generally a bit weaker in the context of the Carlini and Wagner Attack. We will show that, in both targeted and untargeted attacks, we can significantly lower a model's accuracy using very few pixels. We gradually increase the sparsity constraint and observe that this gradually increases the errors made by the model. In particular, in Figure 1, we can see both the accuracy decreasing and the number of samples predicted as the attack target increasing, indicating that the attack is performed as desired.

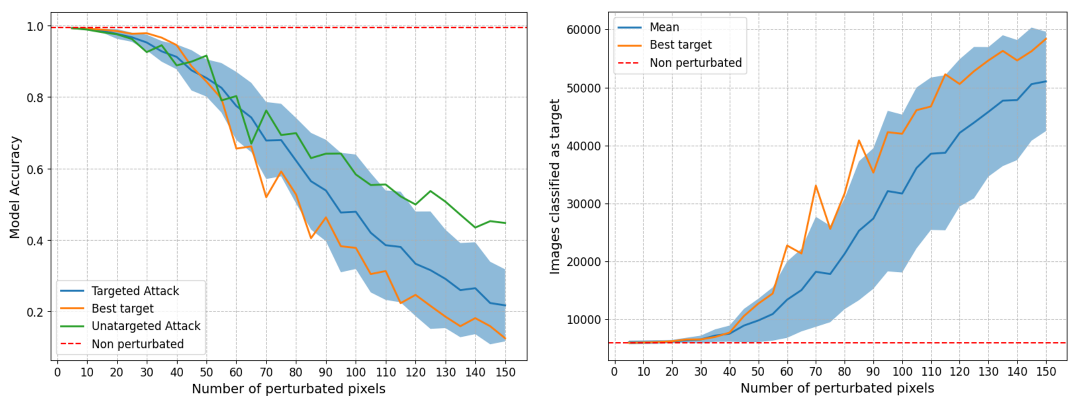

Figure 1. *Effect of increasing the sparsity constraint on accuracy and targeted attack predictions.*

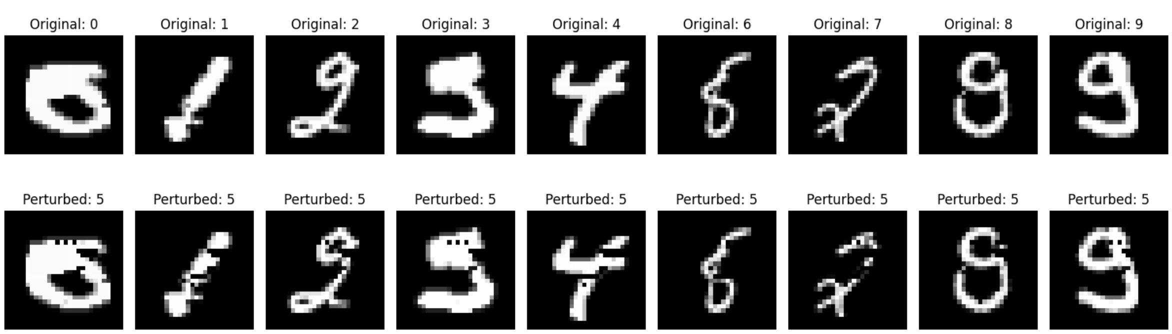

Figure 2. *Example of perturbed images with* $\|\delta\|_0 = 25$ *and target 5*

5.2 Sparse Gaussian Graphical Models Probabilistic Graphical Models are a popular tool in machine learning to model the relationships between random variables. The Gaussian Graphical Model is an undirected graph with each edge corresponding to a Gaussian conditional probability of one variable at the end of the edge to another. By learning the adjacency matrix together with the model weights, we can infer the proximal physical, and possibly causal, relationships between quantities.

This is of special importance in high dimensional settings (see, e.g., [29]). Whereas in many contemporary "big data" approaches the sample size is many orders of magnitudes larger than the dimensionality of feature space, there are a number of settings wherein obtaining data samples is costly, and such a regime cannot be expected to hold. Indeed this is often the case in medical applications, wherein recruiting volunteers for a clinical trial, or even obtaining health records, presents formidable costs to significant scaling in sample size. On the other hand, the precision of instrumentation has led to detailed "omics" data, yielding a very high dimensional feature space. One associated observation is that in the underdetermined case, when the dimensionality of the features exceeds the number of samples, some of the guarantees associated with the ℓ_1 proxy for sparsity are no longer applicable, bringing greater practical salience to having a reliable algorithm enforcing sparsity explicitly.

The recent work [5] presented an integer programming formulation for training sparse Gaussian graphical models. Prior to redefining the sparsity regularization using binary variables, their ℓ_0 optimization problem is given as

$$\min_{\Theta \in \mathbb{S}^p} F_0(\Theta) := \sum_{i=1}^{p} \left(-\log(\theta_{ii}) + \frac{1}{\theta_{ii}} \|\tilde{X}\theta_i\|^2 \right) + \lambda_0 \|\Theta\|_0 + \lambda_2 \|\Theta\|_2^2 \tag{5.3}$$

with $\Theta \in \mathbb{S}^p$ being the weights associated with the graph and $\tilde{X} = \frac{1}{\sqrt{n}} X$ the scaled feature matrix, with $X \in \mathbb{R}^{p \times n}$ consisting of p measures and n samples. Functionally, Θ_{ij} defines an edge between node i and j in the graph, with a nonzero indicating the presence of an active edge, which corresponds to a direct link in the perspective of DAG structure of the group. The value associated with the edge corresponds to the weight defining the strength of the interaction between the features i and j. We seek to regularize cardinality for the sake of encouraging parsimonious models, as well as minimizing the total norm of the weights for general regularization.

Due to the structure of our algorithm, we can modify the formulation of the problem by incorporating the $\ell 0$ constraint. The final formulation of the problem is then expressed as follows:

$$\min_{\Theta \in \mathbb{S}^p, \|\Theta\|_0 \le K} F_0(\Theta) := \sum_{i=1}^{p} \left(-\log(\theta_{ii}) + \frac{1}{\theta_{ii}} \|\tilde{X}\theta_i\|^2 \right) + \lambda_2 \|\Theta\|_2^2 \tag{5.4}$$

We also observed that the ℓ_0 constraint in our formulation is very strong. In practical applications, we eliminate λ_2 penalty term, as the ℓ_0 constraint was the dominant factor in the model.

We applied the model to the GDS2910 dataset from the Gene Expression Omnibus (GEO). This dataset consists of gene expression profiles, which naturally yield a high-dimensional feature space, with 1900 features and 191 samples. Given this feature-to-sample ratio, we can assume some level of sparsity in the final adjacency matrix. Since there is no ground truth for the underlying structure, our goal is to investigate how changing the ℓ_0 constraint affects the results of our method, while also gathering information on the true sparsity nature of the data. We performed the test by gradually increasing K, the ℓ_0 constraint, from 5000 to

15000. This range was previously determined to be optimal based on preliminary tests. Note that the adjacency matrix we are searching for is of size 1900×1900, resulting in a total of $3,610,000$ entries. To ensure the robustness of the results, for each value of K, we performed ten runs starting from different randomly chosen feasible points, and the algorithm was given a total of 1000 iteration for every run. We also decided to set the λ_2 parameter to zero, as we observed that the strong ℓ_0 constraint was dominant over the ℓ_2 penalty.

We also divided the dataset into training and validation sets to determine whether the reconstructed matrix is a result of overfitting. In Figure 3, we show the effect of varying K, which represents the number of nonzero entries that the matrix is allowed to have. The figure on the left, which shows the average objective value found over the ten runs, demonstrates that increasing K eventually stops being beneficial to the model's performance. Additionally, we observe that the number of mean accepted iterations also stops increasing, indicating that the model cannot extract more information from the data. This suggests that the true sparsity of the data can be estimated by identifying the point at which further increasing K no longer improves the model's results. In Figure 4, we present an example from our tests where the objective function decreases over the successful iterations.

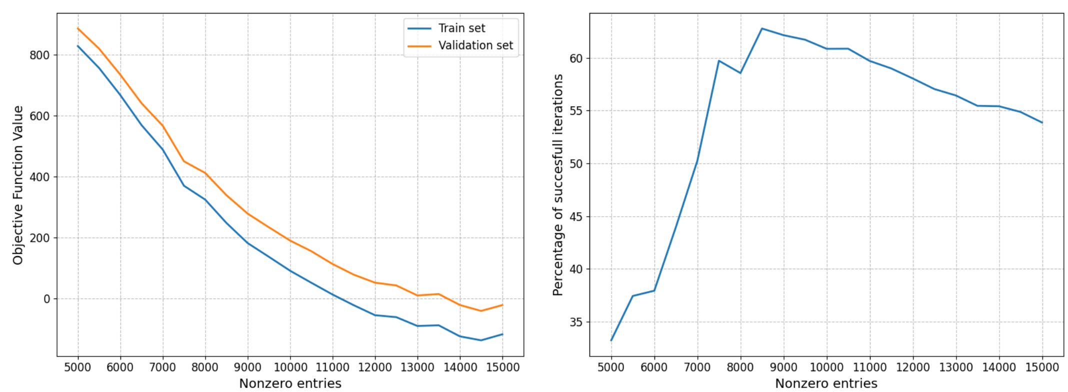

Figure 3. *Effect of increasing the sparsity constraint K.*

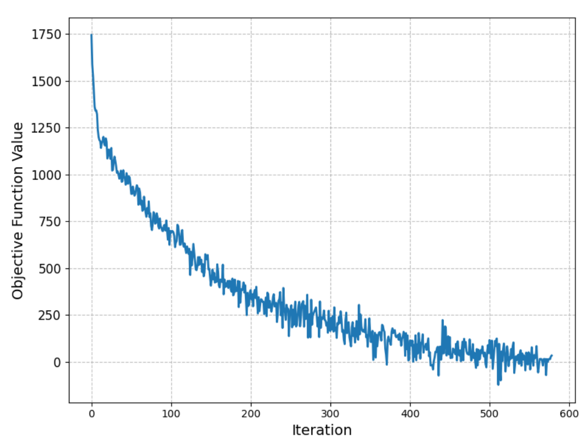

Figure 4. *Objective function over the iterations.*

6 Conclusions

In this paper, we addressed the stochastic cardinality-constrained optimization problem, providing a well defined algorithm, convergence theory and illustrative experiments. Many contemporary machine learning applications involve scenarios where sparsity is crucial for high-dimensional model fitting. We proposed an iterative hard-thresholding like algorithm based on probabilistic models that nicely balances computational efficiency and solution precision by allowing flexible gradient estimates while incorporating hard sparsity constraints.

We analyzed the theoretical properties of the method and proved almost sure convergence to L-stationary points under mild assumptions. This extends previous work in the optimization literature on finding solutions with strong stationarity guarantees together with machine learning articles that perform iterative hard thresholding with stochastic gradients to achieve a novel balance between ease of a fast implementation and formal guarantees of performance. The numerical experiments confirmed the practical effectiveness of our method, showcasing its potential in machine learning tasks such as adversarial attacks and probabilistic graphical model training. By enforcing explicit cardinality constraints, our approach was able to produce models with enhanced sparsity and interpretability in the end.

Future work may involve extending the algorithm to accommodate additional nonlinear constraints, exploring techniques to further improve scalability and performance, as well as testing the algorithm on some other relevant Machine Learning applications, like, e.g., sparse Dynamic Bayesian Network training.

REFERENCES

[1] A. S. Bandeira, K. Scheinberg, and L. N. Vicente, *Convergence of trust-region methods based on probabilistic models*, SIAM Journal on Optimization, 24 (2014), pp. 1238–1264.

[2] F. Bastin, C. Cirillo, and P. L. Toint, *An adaptive monte carlo algorithm for computing mixed logit estimators*, Computational Management Science, 3 (2006), pp. 55–79.

[3] A. Beck and Y. C. Eldar, *Sparsity constrained nonlinear optimization: Optimality conditions and algorithms*, SIAM Journal on Optimization, 23 (2013), pp. 1480–1509.

[4] A. Beck and N. Hallak, *On the minimization over sparse symmetric sets: projections, optimality conditions, and algorithms*, Mathematics of Operations Research, 41 (2016), pp. 196–223.

[5] K. Behdin, W. Chen, and R. Mazumder, *Sparse gaussian graphical models with discrete optimization: Computational and statistical perspectives*, arXiv preprint arXiv:2307.09366, (2023).

[6] D. P. Bertsekas, *Nonlinear programming*, Athena Scientific, Belmont, MA, 1999.

[7] T. Blumensath and M. E. Davies, *Iterative thresholding for sparse approximations*, Journal of Fourier analysis and Applications, 14 (2008), pp. 629–654.

[8] M. Branda, M. Bucher, M. Červinka, and A. Schwartz, *Convergence of a scholtes-type regularization method for cardinality-constrained optimization problems with an application in sparse robust portfolio optimization*, Computational Optimization and Applications, 70 (2018), pp. 503–530.

[9] M. Bucher and A. Schwartz, *Second-order optimality conditions and improved convergence results for regularization methods for cardinality-constrained optimization problems*, Journal of Optimization Theory and Applications, 178 (2018), pp. 383–410.

[10] O. Burdakov, C. Kanzow, and A. Schwartz, *Mathematical programs with cardinality constraints: Reformulation by complementarity-type conditions and a regularization method*, SIAM Journal on Optimization, 26 (2016), pp. 397–425.

[11] N. Carlini and D. Wagner, *Towards evaluating the robustness of neural networks*, in 2017 ieee symposium on security and privacy (sp), Ieee, 2017, pp. 39–57.

[12] N. Carlini and D. A. Wagner, *Towards evaluating the robustness of neural networks*, CoRR, abs/1608.04644 (2016), http://arxiv.org/abs/1608.04644, https://arxiv.org/abs/1608.04644.

[13] C. Cartis and K. Scheinberg, *Global convergence rate analysis of unconstrained optimization methods based on probabilistic models*, Mathematical Programming, 169 (2018), pp. 337–375.

[14] M. Červinka, C. Kanzow, and A. Schwartz, *Constraint qualifications and optimality conditions for optimization problems with cardinality constraints*, Mathematical Programming, 160 (2016), pp. 353–377.

[15] R. Chen, M. Menickelly, and K. Scheinberg, *Stochastic optimization using a trust-region method and random models*, Mathematical Programming, 169 (2018), pp. 447–487.

[16] F. Croce and M. Hein, *Sparse and imperceivable adversarial attacks*, in Proceedings of the IEEE/CVF international conference on computer vision, 2019, pp. 4724–4732.

[17] E. S. Gardner Jr, *Exponential smoothing: The state of the art*, Journal of forecasting, 4 (1985), pp. 1–28.

[18] N. Hallak, *A path-based approach to constrained sparse optimization*, SIAM Journal on Optimization, 34 (2024), pp. 790–816.

[19] P. Jain, A. Tewari, and P. Kar, *On iterative hard thresholding methods for high-dimensional m-estimation*, Advances in neural information processing systems, 27 (2014).

[20] C. Kanzow, A. B. Raharja, and A. Schwartz, *Sequential optimality conditions for cardinality-constrained optimization problems with applications*, Computational Optimization and Applications, 80 (2021), pp. 185–211.

[21] S. Lämmel and V. Shikhman, *On nondegenerate m-stationary points for sparsity constrained nonlinear optimization*, Journal of Global Optimization, 82 (2022), pp. 219–242.

[22] S. Lämmel and V. Shikhman, *Critical point theory for sparse recovery*, Optimization, 72 (2023), pp. 521–549.

[23] M. Lapucci, T. Levato, F. Rinaldi, and M. Sciandrone, *A unifying framework for sparsity-constrained optimization*, Journal of Optimization Theory and Applications, 199 (2023), pp. 663–692.

[24] M. Lapucci, T. Levato, and M. Sciandrone, *Convergent inexact penalty decomposition methods for cardinality-constrained problems*, Journal of Optimization Theory and Applications, 188 (2021), pp. 473–496.

[25] Z. Lu and Y. Zhang, *Sparse approximation via penalty decomposition methods*, SIAM Journal on Optimization, 23 (2013), pp. 2448–2478.

[26] A. Modas, S.-M. Moosavi-Dezfooli, and P. Frossard, *Sparsefool: a few pixels make a big difference*, in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2019, pp. 9087–9096.

[27] T. Murata and T. Suzuki, *Sample efficient stochastic gradient iterative hard thresholding method for stochastic sparse linear regression with limited attribute observation*, Advances in Neural Information Processing Systems, 31 (2018).

[28] M. M. Negri, F. Arend Torres, and V. Roth, *Conditional matrix flows for gaussian graphical models*, Advances in Neural Information Processing Systems, 36 (2023), pp. 25095–25111.

[29] M. J. Wainwright, *High-dimensional statistics: A non-asymptotic viewpoint*, vol. 48, Cambridge university press, 2019.

[30] B. Zhou, F. Chen, and Y. Ying, *Stochastic iterative hard thresholding for graph-structured sparsity optimization*, in International Conference on Machine Learning, PMLR, 2019, pp. 7563–7573.

[31] P. Zhou, X. Yuan, and J. Feng, *Efficient stochastic gradient hard thresholding*, Advances in Neural Information Processing Systems, 31 (2018).