Get rid of your constraints and reparametrize:
A study in NNLS and implicit bias

Hung-Hsu Chou[1]
Department of Mathematics
University of Pittsburgh
USA

Johannes Maly[1]
Department of Mathematics,
LMU Munich, and Munich
Center for Machine Learning (MCML),
Germany

Claudio Mayrink Verdun[1]
John A. Paulson School of
Engineering and Applied Sciences,
Harvard University
USA

Bernardo Freitas Paulo da Costa
Escola de Matemática Aplicada
Fundação Getulio Vargas
Brasil

Heudson Mirandola
Instituto de Matemática
Universidade Federal do Rio de Janeiro
Brasil

Abstract

Over the past years, there has been significant interest in understanding the implicit bias of gradient descent optimization and its connection to the generalization properties of overparametrized neural networks. Several works observed that when training linear diagonal networks on the square loss for regression tasks (which corresponds to overparametrized linear regression) gradient descent converges to special solutions, e.g., non-negative ones. We connect this observation to Riemannian optimization and view overparametrized GD with identical initialization as a Riemannian GD. We use this fact for solving non-negative least squares (NNLS), an important problem behind many techniques, e.g., non-negative matrix factorization. We show that gradient flow on the reparametrized objective converges globally to NNLS solutions, providing convergence rates also for its discretized counterpart. Unlike previous methods, we do not rely on the calculation of exponential maps or geodesics. We further show accelerated convergence using a second-order ODE, lending itself to accelerated descent methods. Finally, we establish the stability against negative per-

turbations and discuss generalization to other constrained optimization problems.

1 INTRODUCTION

In an attempt to understand the still mysterious generalization properties of unregularized gradient-based learning procedures for deep neural networks (Belkin et al., 2019; Zhang et al., 2021), a recent line of research has been focusing on examining the implicit bias of (stochastic) gradient descent (GD) on overparametrized models. The fundamental question is why the networks found by GD generalized so well despite being trained to a perfect interpolation regime. The mathematical analysis of this phenomenon was first considered in simplified problem instances like linear classification (Soudry et al., 2018; Ji and Telgarsky, 2019), matrix factorization (Gunasekar et al., 2017; Arora et al., 2019; Chou et al., 2024), matrix recovery (Geyer et al., 2020; Stöger and Soltanolkotabi, 2021), two-layer networks in the mean-field regime (Chizat and Bach, 2020) and linear regression (Vaskevicius et al., 2019; Zhao et al., 2019; Chou et al., 2023a). In such settings, GD and its underlying gradient flow provably converge to particular solutions, e.g., parsimoniously structured ones. Many of the theoretical contributions are restricted to deep linear networks (Gidel et al., 2019; Woodworth et al., 2020; Pesme et al., 2021) under the so-called *equal* or *balanced* initialization (Hardt and Ma, 2016; Arora et al., 2018; Azulay et al., 2021; Elkabetz and Cohen, 2021; Li et al., 2023). In this case, the overparametrization is equivalent to a reparametrization of the GD dynamics; cf. Remark 1.1 below. Moreover, the study

---

[1]Equal contribution.

of overparametrized models led to an investigation of constrained optimization problems. In the same way that the bias of GD has been shown to induce parsimonious solutions under certain assumptions, it is natural to ask if, for certain reparametrizations, the limit point found via GD could belong to a (convex) set with a certain geometry. As is folklore in the literature, the Riemannian interpretation of such reparametrization allows us to naturally express the geometric structure of the problem, in case it has constraints (Martínez-Rubio and Pokutta, 2023). However, to our knowledge, even in the overparametrized linear regression case, these contributions have only analyzed the continuous dynamics (Li et al., 2022) or second-order optimality conditions of such overparametrized objective functions (Ding and Wright, 2023).

In this paper, we take inspiration from the rich body of work on the implicit bias of GD to analyze the performance of reparametrized GD on constrained optimization problems and propose accelerated GD variants for such problems. We use the so-called *non-negative least-squares (NNLS)* problem as a popular representative. NNLS can be formulated in a rigorous way as follows: given a linear operator $\mathbf{A} \in \mathbb{R}^{M \times N}$ and data $\mathbf{b} \in \mathbb{R}^M$, one seeks for any solution

$$\mathbf{x}_+ \in S_+ := \arg\min_{\mathbf{z} \geq \mathbf{0}} \ell(\mathbf{z}), \qquad \text{(NNLS)}$$

where the least squares loss is defined as

$$\ell \colon \mathbb{R}^N \to [0, \infty), \qquad \ell(\mathbf{z}) := \frac{1}{2}\|\mathbf{A}\mathbf{z} - \mathbf{b}\|_2^2. \quad (1)$$

Since physical quantities of interest often are positive by nature, e.g., in deconvolution and demixing problems like source separation, it is common to search for a least-squared solution under additional non-negativity constraints by solving (NNLS). Large scale applications in which (NNLS) appears include NMR relaxometry (Stoch and Krzyżak, 2021), imaging deblurring (Benvenuto et al., 2009), biometric templates (Kho et al., 2019), transcriptomics (Kim et al., 2011), magnetic microscopy (Myre et al., 2019), sparse hyperspectral unmixing (Esser et al., 2013), and system identification (Chen et al., 2011). The formulation in (NNLS) is also closely related to non-negative matrix factorization (Chu et al., 2021) and supervised learning methods such as support vector machines (Vapnik, 1999). We refer the interested reader to Chen and Plemmons (2010) for a survey about the development of algorithms that enforce non-negativity. In comparison with standard least squares, the positivity constraint, however, imposes additional difficulties. By now, three main algorithmic approaches exist: (i) interior point methods, (ii) active set methods, and (iii) projected gradient methods. In (i), one uses that (NNLS) can be

recast as the quadratic problem

$$\arg\min_{\mathbf{x} \geq \mathbf{0}} \frac{1}{2}\langle \mathbf{x}, \mathbf{Q}\mathbf{x} \rangle + \langle \mathbf{c}, \mathbf{x} \rangle, \qquad (2)$$

where $\mathbf{Q} = \mathbf{A}^\top \mathbf{A}$ and $\mathbf{c} = -\mathbf{A}^\top \mathbf{b}$, and then be solved via interior point methods. For $M = N$, these are guaranteed to converge to an ε-solution in $\mathcal{O}(N^3 \ln \varepsilon^{-1})$ time (Bellavia et al., 2006). The (ii) active set methods (Lawson and Hanson, 1995) represent the most commonly used solution to (NNLS). They exploit the fact that $\mathbf{x}_+$ in (NNLS) can be found by solving an unconstrained problem with *inactive variables* that do not contribute to the constraints. Both (i) and (ii) require solving linear systems at each iteration, which limits their scalability. In contrast, (iii) projected gradient methods like projected gradient descent (PGD) only require matrix-vector multiplications, and the projection to the positive orthant can be trivially performed (e.g., Bierlaire et al., 1991, Lin, 2007, Kim et al., 2013). Nevertheless, the choice of step size is challenging for such methods. Even though PGD is guaranteed to converge with the stepsize given by the inverse of the Lipschitz constant, this implies very slow convergence for ill-conditioned problems. When using stepsize acceleration methods in such scenarios, e.g., the Barzilai-Borwein step-size, one, however, encounters cases in which PGD exhibits cyclic behavior and fails to converge for a choice that provably works for unconstrained gradient descent (Dai and Fletcher, 2005). For additional literature on implicit regularization of GD and NNLS we refer the reader to Appendix A.

Instead of solving a constrained problem, we approach NNLS by considering GD and its underlying flow on the *reparametrized least squares loss* given by

$$\mathcal{L} \colon \mathbb{R}^N \to [0, \infty), \qquad \mathcal{L}(\mathbf{x}) := \frac{1}{2L}\|\mathbf{A}\mathbf{x}^{\odot L} - \mathbf{b}\|_2^2. \quad (3)$$

where $\odot$ denotes the entry-wise (Hadamard) power; the additional $\frac{1}{L}$-factor is added for convenience and does not influence the set of minimizers. Therefore, we are going to analyze the flow

$$\mathbf{x}'(t) = -\nabla \mathcal{L}(\mathbf{x}(t)),$$

with $\mathbf{x}(0) = \mathbf{x}_0$ for a given $\mathbf{x}_0 \geq 0$, as well as its discrete counterpart, i.e., we will forget the non-negative constraints and analyze the behavior of the unconstrained non-convex flow. (In the following, we will often abbreviate $\tilde{\mathbf{x}} := \mathbf{x}^{\odot L}$ and use that $\mathcal{L}(\mathbf{x}) = \frac{1}{L}\ell(\tilde{\mathbf{x}})$.) Despite its simplicity, the loss in (3) — also called *Hadamard reparametrized least squares* — has been studied in various works (Arora et al., 2018, Gunasekar et al. 2018, Vaskevicius et al., 2019; Zhao et al., 2019, Chou et al., 2023a) because it is believed to be the first step in developing theoretical guarantees for models with deep structure.

Remark 1.1. *The reparametrized loss in* (3) *is related to the overparametrized loss* $\mathcal{L}_{over}\colon \mathbb{R}^N \times \cdots \times \mathbb{R}^N \to [0,\infty)$ *with*

$$\mathcal{L}_{over}\big(\mathbf{x}^{(1)},\ldots,\mathbf{x}^{(L)}\big) := \frac{1}{2} \left\| \mathbf{A}\big(\mathbf{x}^{(1)} \odot \cdots \odot \mathbf{x}^{(L)}\big) - \mathbf{b} \right\|_2^2, \tag{4}$$

which appeared before in the context of implicit ℓ_1-regularization of GD (Vaskevicius et al., 2019; Li et al., 2021; Chou et al., 2023a).[2] In these works, it was shown that if there exists a non-negative solution $\mathbf{x} \geq \mathbf{0}$ with $\mathbf{A}\mathbf{x} = \mathbf{b}$, and if all factors $\mathbf{x}^{(k)}$ are initialized with $\alpha\mathbf{1}$ at $t = 0$, for $\alpha > 0$ sufficiently small, and follow gradient flow with respect to $\mathcal{L}_{over}$, then their product $\mathbf{x}^{(1)}(t) \odot \cdots \odot \mathbf{x}^{(L)}(t)$ approximates an ℓ_1-minimizer among all positive solutions, for $t \to \infty$. A key observation by Vaskevicius et al. (2019); Li et al. (2021); Chou et al. (2023a) is that if all factors $\mathbf{x}^{(k)}$ are initialized by the same vector $\mathbf{x}_0$, then gradient flow on (4) will preserve this equality, and the trajectory of any $\mathbf{x}^{(k)}$ is equal to the trajectory of the gradient flow trajectory $\mathbf{x}$ on (3) (e.g., Chou et al., 2023a, Appendix A.1).

In light of Remark 1.1, we will restrict ourselves to reparametrized gradient flow/descent on (3). Any resulting statement automatically applies to gradient flow/descent on (4) under identical initialization of all factors $\mathbf{x}^{(k)}$. Let us highlight two points: first, whereas in the works of Vaskevicius et al. (2019); Li et al. (2021); Chou et al. (2023a) the positivity of the limit was a technical nuisance that needed to be circumvented by adapting (4), the present work makes use of it to restrict the flow to a constrained set and to solve (NNLS). Such a connection between overparameterization in ML and classical optimization problems opens up new avenues for algorithm design. Second, the fact that our results only apply to (4) with identical initialization, i.e., $\mathbf{x}^{(1)}(0) = \cdots = \mathbf{x}^{(L)}(0) = \mathbf{x}_0$ is not restrictive when solving (NNLS). In Appendix B, we argue that all solutions of (NNLS) are stationary points of (4) that can be described as the limit of gradient flow on (4) under suitably chosen *identical* initialization. We now present our main contributions, which demonstrate how constraints can be effectively traded for increased complexity in the objective function. This approach allows us to solve NNLS using unconstrained optimization techniques, leading to a scalable algorithm.

1.1 Contribution and Outline

In light of the connection between the implicit regularization of GD on (3) and finding non-negative solutions

in (NNLS), as discussed in Remark 1.1, we analyze in this work the continuous and the discrete dynamics of the reparameterized loss $\mathcal{L}$ from equation (3) and show that:

- The unconstrained gradient flow on (3) globally converges to solutions of (NNLS) at an $\mathcal{O}(1/t)$ decay rate on ℓ, cf. Theorem 2.1. Whereas a similar result can be obtained via Li et al. (2022), our presented proof does not directly rely on tools and concepts from Riemannian geometry, cf. Remark 2.3.

- An accelerated version of the gradient flow on the original coordinates leads to an $\mathcal{O}(1/t^2)$ decay rate on ℓ, cf. Theorem 2.4. On numerical examples, this scheme attains an $\mathcal{O}(1/k^3)$ decay. Surprisingly, accelerating the *reparameterized gradient descent* (that is, for $\mathcal{L}(\mathbf{x})$) yields even faster convergence, cf. Section 3.2.

- Gradient descent on (3) with decaying step-size schemes exhibits on ℓ a decay rate of $\mathcal{O}(1/k^\gamma)$, where $0 < \gamma < 1$ depends on the decay rate of the step-size, cf. Theorem 2.6. Such a result is stronger than the one given by mirror descent arguments on general convex loss function which, in turn, are similar to subgradient methods with decay rate $\mathcal{O}(1/\sqrt{k})$.[3] Our numerical simulations suggest that our bound matches the practical performance of GD, cf. Section 3.3.

A nice by-product of our approach is that, by choosing the initialization of gradient flow close to zero, one can add an additional ℓ_1-bias on top of non-negativity. This latter feature is inherited from previous works like Vaskevicius et al. (2019); Chou et al. (2023a) and is only of interest in the special case of applying NNLS to sparse recovery, cf. Section 2.4.

The main body of the paper is structured as follows: In Section 2, we present our theoretical contributions in detail. In Section 3, we evaluate our methods in numerical experiments. We conclude in Section 4 with a discussion of the results. All proofs and additional numerical experiments are provided in the appendix.

1.2 Notation

Before detailing our results, we need to introduce some additional notation. For any solution $\mathbf{x}_+ \in S_+$ as de-

---

[2] In fact, minimizing (4) corresponds to training an L-layer linear network with diagonal weight matrices that are fully characterized by their diagonals $\mathbf{x}^{(\ell)} \in \mathbb{R}^N$, cf. Nacson et al. (2022).

[3] Note that mirror descent with 1-strongly convex Legendre function provably reaches $\mathcal{O}(1/k)$ (Yao-Liang Yu 2017). However, this assumption is not fulfilled by the reparametrization in (3), which, to the best of our knowledge, is the only one considered in recent literature on Hadamard overparametrization.

fined in (NNLS), we define $\mathbf{b}_+ := \mathbf{A}\mathbf{x}_+$ which satisfies

$$\mathbf{b}_+ = \mathcal{P}_{C_+}\mathbf{b}, \text{ for } C_+ := \{\mathbf{A}\mathbf{z} : \mathbf{z} \in \mathbb{R}_{\geq 0}\} \subset \mathbb{R}^M. \quad (5)$$

Since C_+ is convex and closed, $\mathbf{b}_+$ is the unique Euclidean projection and thus independent of the choice of $\mathbf{x}_+$. Note that $\mathbb{R}^N_{\geq 0} = \{\mathbf{x} \in \mathbb{R}^N : \mathbf{x} \geq \mathbf{0}\}$ and $\mathbb{R}^N_{>0} = \{\mathbf{x} \in \mathbb{R}^N : \mathbf{x} > \mathbf{0}\}$ where, for convenience, we denote by $\mathbf{x} \geq \mathbf{y}$ the entry-wise bound $x_n \geq y_n$, for all n, and the all zero and all ones vector by $\mathbf{0}$ and $\mathbf{1}$ respectively. We denote ℓ_p-vector norms by $\|\cdot\|_p$ and the operator norm of matrices by $\|\cdot\|$. We use $\odot$ to denote the Hadamard product, i.e., the vectors $\mathbf{x} \odot \mathbf{y}$ and $\mathbf{x}^{\odot p}$ have entries $(\mathbf{x} \odot \mathbf{y})_n = x_n y_n$ and $(\mathbf{x}^{\odot p})_n = x_n^p$, respectively. Univariate functions like the logarithm are applied entry-wise to vectors, i.e., $\log(\mathbf{x}) \in \mathbb{R}^N$ with $\log(\mathbf{x})_n = \log(x_n)$. We denote the complement of a set I by I^c and its cardinality by $|I|$. The support of a vector $\mathbf{x} \in \mathbb{R}^N$ is the index set of its nonzero entries and denoted by $\text{supp}(\mathbf{x}) = \{j \in [N] : x_j \neq 0\}$. We call a vector s-sparse if $|\text{supp}(\mathbf{x})| \leq s$. We denote by $\mathbf{x}_I \in \mathbb{R}^N$ the projection of $\mathbf{x} \in \mathbb{R}^N$ onto the coordinates indexed by I. Furthermore, $\sigma_s(\mathbf{x})_{\ell_1}$ denotes the ℓ_1-error of the best s-term approximation of $\mathbf{x} \in \mathbb{R}^N$, i.e., $\sigma_s(\mathbf{x})_{\ell_1} = \inf\{\|\mathbf{x} - \mathbf{z}\|_1 : \mathbf{z} \in \mathbb{R}^N \text{ is } s\text{-sparse}\}$.

2 THEORETICAL RESULTS

The main theoretical results of this paper show (i) that reparametrized gradient flow on (3) converges to solutions of (NNLS) with decay rate $\mathcal{O}(1/t)$, see Section 2.1, (ii) that an accelerated version of the flow improves this rate to $\mathcal{O}(1/t^2)$, see Section 2.2, (iii) that GD on (3) with decaying stepsize reaches a decay rate of $\mathcal{O}(1/k^\gamma)$ where $0 < \gamma < 1$ depends on the decay rate of the step-size, see Section 2.3, and (iv) that our approach for solving NNLS comes with certain advantages compared to other NNLS methods when used for sparse recovery, see Section 2.4.[4]

2.1 Convergence rate of gradient flow

Our first contribution is to extend the argument of Chou et al. (2023a) and to show, for any $\mathbf{A}$, $\mathbf{b}$, and positive (identical) initialization $\mathbf{x}(0) = \mathbf{x}_0 > \mathbf{0}$, that the Hadamard power $\tilde{\mathbf{x}} = \mathbf{x}^{\odot L}$ of the gradient flow $\mathbf{x}$ on (3) converges to a solution $\mathbf{x}_+$ of (NNLS). *A crucial point here is that, in contrast to theoretical results by Chou et al. (2023a), the existence of a solution $\mathbf{A}\mathbf{x} = \mathbf{b}$ is not required anymore.* In addition, we characterize the decay rate of the loss as $\mathcal{O}(1/t)$.

---

[4]We emphasize at this point that, for a certain class of matrices $\mathbf{A}$, (NNLS) is a valid approach to identifying sparse solutions from $\mathbf{b}$, cf. our extended discussion in Appendix G.

Theorem 2.1. *Let $L \geq 2$, $\mathbf{A} \in \mathbb{R}^{M \times N}$ and $\mathbf{b} \in \mathbb{R}^M$. Let $\mathbf{x}_0 > \mathbf{0}$ be fixed and let $\mathbf{x}(t)$ follow the flow $\mathbf{x}'(t) = -\nabla\mathcal{L}(\mathbf{x}(t))$ with $\mathbf{x}(0) = \mathbf{x}_0$. Let S_+ be the set defined in (NNLS) and let $\tilde{\mathbf{x}} = \mathbf{x}^{\odot L}$.*

Then the limit $\tilde{\mathbf{x}}_\infty := \lim_{t\to\infty} \tilde{\mathbf{x}}(t)$ exists and lies in S_+. Also, there exists an absolute constant $C > 0$ that only depends on the choice of $\mathbf{A}$, $\mathbf{b}$, and $\mathbf{x}_0$ such that

$$\ell(\tilde{\mathbf{x}}(t)) - \ell(\mathbf{x}_+) \leq \frac{C}{t},$$

for any $t > 0$ and any $\mathbf{x}_+ \in S_+$.

Proof sketch. By using properties of orthogonal projections, see Lemma C.6, existence of $\tilde{\mathbf{x}}_\infty \in S_+$ can be obtained by slight modifications of the argument by Chou et al. (2023a). Interestingly, we can streamline the settings $L = 2$ and $L > 2$ by relating the potential of Chou et al. (2023a) to the *Tsallis q-logarithm* (Tsallis 1988). For deducing the convergence rate, we introduce a Lyapunov argument that has not appeared in this context before to the best of our knowledge. The detailed proof of Theorem 2.1 is given in Appendix C. □

Remark 2.2. *Note that non-negativity of the gradient flow trajectory follows from $\mathbf{x}_0 > \mathbf{0}$ and Picard-Lindelöf. Indeed, the Hadamard reparametrization induces $[\nabla\mathcal{L}(\mathbf{x})]_i = 0$ for $x_i = 0$, see (16). If $x(t_*)_i = 0$, for $t_* > 0$, this would give rise to an alternative trajectory with $(x_0)_i = 0$ contradicting the local uniqueness of the flow around t_*.*

2.2 Convergence rate of accelerated reparametrized flow

In order to accelerate the reparametrized flow $\mathbf{x}'(t) = -\nabla\mathcal{L}(\mathbf{x}(t))$ on (3), we note that by setting $q = 2 - \frac{2}{L} \in [1, 2)$ and recalling $\tilde{\mathbf{x}} = \mathbf{x}^{\odot L}$, the reparametrized flow induces a Riemannian metric $\langle \mathbf{u}, \mathbf{v}\rangle_{\tilde{\mathbf{x}}} = \langle \tilde{\mathbf{x}}^{\odot(-q)} \odot \mathbf{u}, \mathbf{v}\rangle$, such that the corresponding Riemannian gradient of ℓ is given by

$$\widetilde{\nabla}\ell(\tilde{\mathbf{x}}) := \tilde{\mathbf{x}}^{\odot q} \odot \nabla\ell(\tilde{\mathbf{x}}) \quad (6)$$

and we can replace the reparametrized flow on $\mathcal{L}$ by the Riemannian flow $\tilde{\mathbf{x}}'(t) = -L\widetilde{\nabla}\ell(\tilde{\mathbf{x}})$ on ℓ. A detailed derivation is provided in Appendix D.

Remark 2.3. *Interestingly, by using this Riemannian viewpoint, Theorem 2.1 can be derived from the results of Li et al. (2022). At the same time, the proof we presented in Section C does not directly rely on tools and concepts from Riemannian geometry but leads to the same result.*

Inspired by the Riemannian representation of the flow, we show significant improvement in the convergence rate of the flow by using a 2nd-order ODE for (3).

Theorem 2.4. *Let $L \geq 2$, $\mathbf{A} \in \mathbb{R}^{M \times N}$, $\mathbf{b} \in \mathbb{R}^M$, and $\mathbf{x}_0 > \mathbf{0}$ be fixed. Moreover, let $\boldsymbol{\xi}(t) \in \mathbb{R}^N$ be a vector such that $(\boldsymbol{\xi}(t), \mathbf{x}(t))$ satisfies the following dynamics*

$$\boldsymbol{\xi}'(t) = -\frac{t}{2}\boldsymbol{\xi}(t)^{\odot q} \odot \nabla\ell(\tilde{\mathbf{x}}(t)),$$
$$\tilde{\mathbf{x}}'(t) = \frac{2}{t}\big(\boldsymbol{\xi}(t) - \tilde{\mathbf{x}}(t)\big), \qquad \text{(ACC)}$$

with $\boldsymbol{\xi}(0) = \tilde{\mathbf{x}}(0) = \tilde{\mathbf{x}}_0 > \mathbf{0}$, where $q = 2 - \frac{2}{L} \in [1, 2)$ and ℓ has been defined in (1). Then, there exists an absolute constant $C > 0$ that only depends on q and the distance between $\tilde{\mathbf{x}}_0$ and $\mathbf{x}_+$ such that

$$\ell(\tilde{\mathbf{x}}(t)) - \ell(\mathbf{x}_+) \leq \frac{C}{t^2},$$

for any $t > 0$ and any $\mathbf{x}_+ \in S_+$ defined in (NNLS).

Proof sketch. The accelerated convergence rate is shown by a similar Lyapunov argument as in Theorem 2.1 using a higher-order Lyapunov function. The specific shape of (ACC) has been derived by reverse engineering the Lyapunov analysis of Theorem 2.1. The full derivation of the result is given in Appendix E. □

Remark 2.5. *Note that (ACC) differs from a naive application of Nesterov's scheme to (6) since $\nabla\ell$ is evaluated at $\tilde{\mathbf{x}}$ and not on $\boldsymbol{\xi}$.*

Our numerical simulations in Section 3.2 show promising results when discretizing (ACC), even exceeding the predicted error decay of Theorem 2.4.

2.3 Convergence rate of gradient descent

We now proceed with a convergence rate analysis of reparametrized gradient descent on (3), i.e., a discrete counterpart of Theorem 2.1. Since our proof technique relies on a discretization of the Riemannian flow introduced in (6), we directly work with GD defined on the products $\tilde{\mathbf{x}} = \mathbf{x}^{\odot L}$ and $\widetilde{\nabla}\ell$ defined in (6).

Theorem 2.6. *Let $\mathbf{x}_+ \in S_+$ where S_+ has been defined in (NNLS), and choose any $\tilde{\mathbf{x}}_0 > \mathbf{0}$ as initialization for the sequence $\tilde{\mathbf{x}}_k$ produced by*

$$\tilde{\mathbf{x}}_{k+1} = \tilde{\mathbf{x}}_k - \eta_k \widetilde{\nabla}\ell(\tilde{\mathbf{x}}_k) \qquad (7)$$

Fix $\delta > 0$, $q \in [1, 2)$, $\gamma \in (0, 1)$, and

$$0 < c \leq \min\left\{\frac{1}{\|\mathbf{A}^T\mathbf{b}\|_2^2(60\mathfrak{C})^q}, \frac{1}{\|\mathbf{A}^T\mathbf{A}\|_2(60\mathfrak{C})^{3q}}\right\}$$

such that $c \in (0, 1)$. The constant $\mathfrak{C}$ satisfies $1 \leq \mathfrak{C} \leq D_F(\mathbf{x}_+, \tilde{\mathbf{x}}_0) + c_{\max} + \|\mathbf{x}_+\|_\infty$ where D_F denotes the Bregman divergence with respect to the Tsallis q-logarithm, cf. Definition C.2 and (12), and $c_{\max}$ is defined as

$$c_{\max} := \begin{cases} \zeta(2\gamma) & \text{for } \gamma > \frac{1}{2} \\ 2\left(\log\left(\frac{2}{(2-q)\delta}\right) + (2-q)\delta\right) & \text{for } \gamma = \frac{1}{2} \\ \frac{1}{1-2\gamma}\left(\frac{2}{(2-q)\delta}\right)^{\frac{1}{\gamma}-2} + 2(2-q)\delta & \text{for } \gamma < \frac{1}{2} \end{cases},$$

where ζ denotes the Riemann zeta function. Then, if $\eta_k := \frac{c}{(k+2)^\gamma}$, we obtain the convergence rate

$$\ell(\tilde{\mathbf{x}}_k) - \ell(\mathbf{x}_+) \leq \frac{2(2-q)^{-1}c^{-1}}{k^{1-\gamma}}(D_F(\mathbf{x}_+, \tilde{\mathbf{x}}_0) + c_{\max})$$

for any $0 \leq k < K$ where

$$K := \min\{k \in \mathbb{N}: 2(\sqrt{\ell(\tilde{\mathbf{x}}_k)} - \sqrt{\ell(\mathbf{x}_+)}) < \sqrt{\delta}\}.$$

In addition, we have $\tilde{\mathbf{x}}_k \in \mathbb{R}_{>0}^N$ with $\|\tilde{\mathbf{x}}_k\|_2 \leq \mathfrak{C}$.

Proof sketch. To deduce the convergence rate, we carefully discretize the Lyapunov argument from Theorem 2.1 using Taylor's theorem. This results in an intricate inductive argument, which simultaneously controls three quantities: (i) the ℓ_2-norms of the iterates, (ii) the entry-wise signs of the iterates, and (iii) the decay in the Lyapunov function. The detailed proof of Theorem 2.6 is in Appendix F. □

Remark 2.7. *Note that for quickly decaying stepsizes, i.e., $\gamma > \frac{1}{2}$, we may set $\delta = 0$ in Theorem 2.6 which yields $K = \infty$ and thus convergence of the sequence of iterates to global optimality. For larger stepsizes, the result only controls the decay rate of the iterates until $\ell(\tilde{\mathbf{x}}_k) - \ell(\mathbf{x}_+) = \mathcal{O}(\sqrt{\delta})$, i.e., a neighborhood of S_+ is reached. Moreover, Figure 3 illustrates that the predicted rate of $\mathcal{O}(1/k^\gamma)$ also holds for constant stepsize, i.e., $\gamma = 0$. Our restriction to $\gamma \in (0, 1)$ and the neighborhood restrictions are probably artifacts of our proof technique.*

We leave it as an open question for future work to analyze a discrete analog of the accelerated flow in (ACC).

2.4 NNLS for sparse recovery

It is known in the field of sparse recovery that under certain assumptions on the measurement operator $\mathbf{A}$, solving (NNLS) promotes sparsity of its solution without any hyper-parameter tuning and comes with improved robustness. This observation dates back to the seminal works of Donoho and Tanner (2005, 2010). Therefore, it is expected that the same would happen for GD applied to the overparametrized function in (3). As a nice by-product of reusing existing theory in our proofs, we observe that if $\mathbf{x}_0$ is chosen sufficiently close to zero, the limit of gradient flow (approximately) minimizes the ℓ_1-norm among all possible solutions of

(NNLS), which has previously been observed by Chou et al. (2023a). Theorem C.1 in Appendix C provides the detailed relation between initialization scale and implicit ℓ_1-regularization. The precise statements on how to use this for sparse recovery are given in Appendix G. We will only treat the result informally here. According to Theorem G.4,

$$\|\tilde{\mathbf{x}}_\infty\|_1 - \min_{\mathbf{z} \in S_+} \|\mathbf{z}\|_1 \le \varepsilon,$$

where ε decreases as the initialization scale decreases. If we further assume that $\mathbf{A} \in \mathbb{R}^{N \times M}$ has the ℓ_2-robust null space property (Definition G.1) and ℓ_1-quotient property (Definition G.5), then we can stably recover sparse ground truths (Theorem G.6).

Theorem G.6 implies that our approach to solving (NNLS) is stable with respect to negative entries of the ground truth, i.e., the reconstruction error depends on the sparsity of the positive part $\mathbf{x}_+$ of $\mathbf{x}_*$ and the magnitude of the negative part $\mathbf{x}_-$ of $\mathbf{x}_*$. The experiments we perform in Section 3.5 suggest that the established solvers for (NNLS) are less stable under such perturbations.

We finally mention that if one is using NNLS for sparse recovery, then an alternative approach that leverages Hadamard overparametrization is the one of Poon and Peyré (2023) in which the authors smoothen an ℓ_1-regularized objective by Hadamard overparametrization combined with explicit ℓ_2-regularization. Due to the explicit use of regularization terms, however, this is different from the implicit bias setting analyzed in this work and clearly designed for sparse recovery.

3 NUMERICAL EXPERIMENTS

Let us finally turn to a numerical evaluation of our theoretical insights.[5] We compare the following six methods for solving NNLS here:

- **GD-nL:** Vanilla gradient descent applied to $\mathcal{L}$ in (3) with n layers, for $n \in \mathbb{N}$. This is the discretized version of the gradient flow we considered in Section 2. As initialization we use $\alpha\mathbf{1}$, for $\alpha > 0$.[6]

- **SGD-nL:** Stochastic gradient descent applied to $\mathcal{L}$ in (3) with n layers, for $n \in \mathbb{N}$. A probabilistic variant of **GD-nL**. In the experiments we use $M/10$ as batch-size for **SGD-nL** and initialize with $\alpha\mathbf{1}$, for $\alpha > 0$.

- **CVX-NNLS:** Solving the quadratic formulation of (NNLS) described in (2) via ADMM. We use the Python-embedded modeling language CVX,[7] which, in turn, uses the solver OSQP[8] for NNLS.

- **LH-NNLS:** The standard Python NNLS-solver *scipy.optimize.nnls*, which is an active set method and is based on the original Lawson-Hanson method (Lawson and Hanson, 1995). It is not scalable to high dimensions since it requires solving linear systems in each iteration. (An accelerated version of **LH-NNLS** is provided by Bro and De Jong (1997). Since both methods produce the same outcome in our experiments, we only provide the results for **LH-NNLS**.)

- **TNT-NN:** An alternative but more recent active set method that heuristically works well and dramatically improves over **LH-NNLS** in performance (Myre et al., 2017). We used the recent Python implementation available at https://github.com/gdcs92/pytntnn.

- **PGD**: Projected gradient descent for NNLS as described by Polyak (2015).

We begin by examining the convergence rates of **GD-nL** and **SGD-nL** for various choices of step size, including constant step-size and Nesterov acceleration. We also compare it to the Barzilai-Borwein (BB) stepsize rule due to its popularity and its numerical efficiency. Next, we analyze gradient acceleration corresponding to the differential equation (ACC), with $L = 2$, and illustrate its performance in comparison to the discretized accelerated gradient method for the dynamics in the original variable $\mathbf{x}'(t) = -\nabla\mathcal{L}(\mathbf{x})$. We proceed to the analysis of stepsize decay rates, as required by Theorem 2.6. We then investigate the role of initialization and the number of layers on the implicit sparsity retrieval behavior of **GD-nL**. Finally, we compare the stability of the proposed overparametrized methods against those aforementioned established methods when recovering perturbed signals that are not strictly non-negative. Additional large-scale experiments are provided in Appendix I.

3.1 Different stepsizes

In the first experiment, we compare the convergence rates of **GD-nL** and **SGD-nL** for various choices of step size. Apart from using a constant step-size $\eta = 10^{-2}$, we also consider Nesterov acceleration (Nesterov, 1983) and the BB stepsize rules (Fletcher, 2005; Raydan, 1997), and start all methods from $\mathbf{x} = \alpha \cdot \mathbf{1}$,

---

[5] Our code is available at https://github.com/claudioverdun/NNLS_AISTATS_2025.

[6] Since all factors $\mathbf{x}^{(k)}$ stay identical over time, cf. Remark 1.1, one only needs to store $\mathbf{x}^{(1)}$ and there is no memory overhead of **GD-nL** in comparison to GD on (NNLS).

[7] https://www.cvxpy.org/

[8] https://osqp.org/docs/solver/index.html

with $\alpha = 10^{-2}$. Figure 1 shows the decay in training error, i.e., objective value over time in two different settings: the dense case, i.e., we have a squared system with $M = N = 50$ and a dense ground-truth, and the sparse case, i.e., we have an underdetermined system with $M = 30$, $N = 50$, and a 3-sparse ground-truth. In both settings, $\mathbf{A}$ has Gaussian entries. As Figures 1a and 1b show, advanced choices of step size notably improve the convergence rate of the gradient methods. Moreover, **GD-nL** seems to profit more from the acceleration than **SGD-nL**.
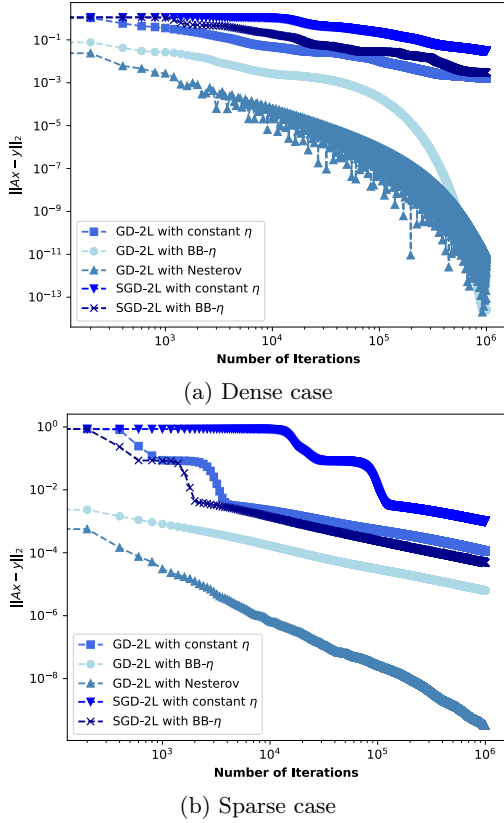

(a) Dense case

(b) Sparse case

Figure 1: Convergence rate for various choices of step-size, see Section 3.1.

3.2 Acceleration

In our next experiment, we analyze gradient acceleration corresponding to the differential equation (ACC), with $L = 2$, and a 20×50 linear system recovering a 3-sparse ground truth. We first solve this equation with an adaptive Runge-Kutta method (Dormand and Prince, 1980), to evaluate its convergence rate, which turn out to be faster than predicted, following an $\mathcal{O}(t^{-3})$ regime, as observed in Figure 2a. The ODE was started at $\tilde{\mathbf{x}}_0 = 10^{-3} \cdot \mathbf{1}$.

Then, we compare two accelerated versions of gradi-

ent descent, the first coming from the Riemannian interpretation of equation (ACC), in the variables $\tilde{\mathbf{x}}$; and a second from Nesterov's scheme directly in the variables $\mathbf{x}$. We start both methods from the same point as the ODE, and use $\eta = 0.1$; detailed step rules are provided in Appendix H. In Figure 2b, we observe that the accelerated Riemannian method reproduces the same convergence rate as the ODE. Moreover, and remarkably, Nesterov's accelerated gradient descent in the coordinates $\mathbf{x}$ simultaneously features faster convergence and smaller oscillations compared to both the ODE solution and the first accelerated scheme.
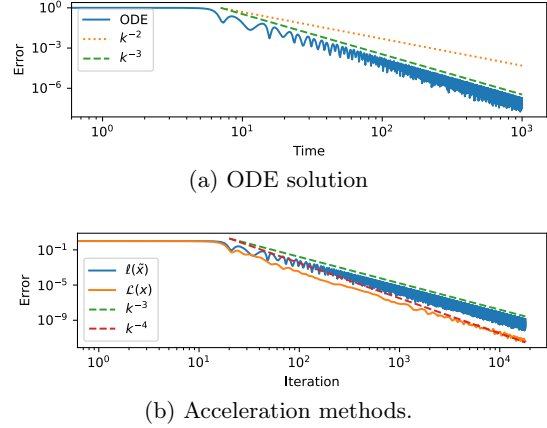

(a) ODE solution

(b) Acceleration methods.

Figure 2: Accelerated gradient methods: (a) Numerical solution of ODE (ACC) and (b) behavior of discretized accelerated gradient.

3.3 Stepsize decay

Now, we illustrate the choice of decay rates $\eta = Ck^{-\gamma}$ from Theorem 2.6 for **GD-nL**. We set $\alpha = 0.1$ and $\eta = 0.1$ for these experiments. The dashed lines in Figure 3 correspond to predicted decay rates $\mathcal{O}(k^{1-\gamma})$, which are in excellent agreement in our experiment, where $M = 20$, $N = 50$ and $x \in \mathbb{R}^N$ is a 3-sparse vector. It is remarkable that the error decreases in almost discrete steps, which are "just-in-time" to keep the global decay rates and that the effect is stronger for larger L. This step-wise decay is related to saddle-to-saddle dynamics as observed by Berthier (2023); Pesme and Flammarion (2023); Chou et al. (2024).

3.4 Initialization and Number of Layers

In the fourth experiment, we validate the ℓ_1-norm regularization that, according to Theorem G.4, can be induced by using a small initialization for **GD-nL**. For $M = 10$ and $N = 50$, we draw a random Gaussian matrix $\mathbf{A} \in \mathbb{R}^{M \times N}$, create a 3-sparse ground-truth $\mathbf{x} \in \mathbb{R}^N$, and set $\mathbf{b} = \mathbf{A}\mathbf{x}$. Figure 4a depicts
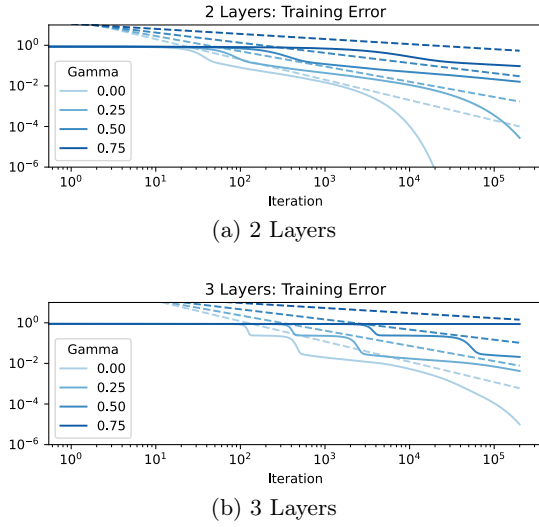
(a) 2 Layers

(b) 3 Layers

Figure 3: Impact of decaying stepsize rate γ, for (a) $L = 2$ and (b) $L = 3$.

(a) Initialization.

(b) Number of layers.

Figure 4: Influence of initialization and number of layers on **GD-nL**, cf. Section 3.4.

the ℓ_1-norm of the limits of **GD-2L** and **GD-3L**, for $10^{-3} \leq \alpha \leq 10^{-1}$ and constant step-size $\eta = 10^{-1}$. As a benchmark, the ℓ_1 minimizer among all feasible solutions is computed via basis pursuit (**L1-MIN**). Figure 4a shows that **GD-2L** and **GD-3L** converge to ℓ_1-norm minimizers if α is sufficiently small. As predicted by Theorem G.4, the requirements on α to allow such regularization are milder for the 3-layer case **GD-3L**. Let us finally mention that neither **LH-NNLS** nor **CVX-NNLS** reach ℓ_1-minimality; the former comes close, but the latter arrives at a solution with ℓ_1-norm of 3.514. Seemingly, the matrix **A**, although sufficiently well-behaved for sparse recovery in general, does not guarantee the uniqueness of the NNLS solution here, cf. Section G.

In a further experiment, we take a closer look at the reconstruction behavior of **GD-nL**. In particular, we compare how different entries of the ground truth are approximated over time for $n = 2$ and $n = 3$ layers. For initialization magnitude and step-size we choose $\alpha = 10^{-2}$ and $\eta = 10^{-2}$. We set $M = 30$ and $N = 50$, draw a random Gaussian matrix $\mathbf{A} \in \mathbb{R}^{M \times N}$, create a 3-sparse non-negative ground-truth $\mathbf{x} \in \mathbb{R}^N$, and set $\mathbf{b} = \mathbf{A}\mathbf{x}$. Note that **A** satisfies the assumptions of Theorem G.6. Figure 4b depicts the entry-wise error between the three non-zero ground-truth entries and the corresponding entries of the iterates of **GD-2L** and **GD-3L**. As already observed in previous related works, we see that a deeper factorization leads to sharper error transitions that occur later and that more dominant entries are recovered faster than the rest. Interestingly, some kind of overshooting occurs in the dominant entries: dashed curves, for the second-highest coordinate,
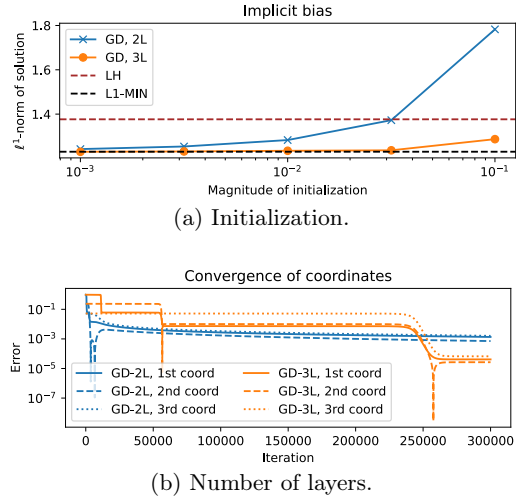
suggest that **GD-nL** does not monotonically decrease the error in all components but rather concentrates heavily on the leading component(s) in the beginning and only starts distributing the error over time. In this way, **GD-nL** could be interpreted as a self-correcting greedy method.

3.5 Stability with Negative Entries

In our final experiment, we compare the stability of the proposed methods when recovering perturbed signals that are not strictly non-negative anymore. We consider two different set-ups. We use images from the MNIST data set and CIFAR10. In Appendix I.2 we analyze the perturbed behavior with generic random signals. In both scenarios we set $\alpha = 10^{-2}$ and $\eta = 10^{-2}$ for **GD-nL** and **SGD-nL**. It is noteworthy that in all tested instances **GD-nL** and **SGD-nL** outperform the established methods in reconstruction quality, cf. Figure 5. Whereas **HL-NNLS** and **TNT-NN** are designed to retrieve only non-negative signals, the gradient-based methods can stably deal with vectors that have small negative components by not using explicit constraints.

MNIST signals. Again $\mathbf{A} \in \mathbb{R}^{M \times N}$ is a random Gaussian matrix, where now $M = 300$ and $N = 28^2 = 784$. We take $\mathbf{x}_+$ to be the original MNIST image (number three) and define a corrupted signal $\mathbf{x}$ including negative (mean zero) Gaussian noise $\mathbf{x}_-$. For $q \in [0, 1]$, we scale $\mathbf{x}_+, \mathbf{x}_-$ such that

$$\|\mathbf{x}_+\|_2^2 = 1 - q \quad \text{and} \quad \|\mathbf{x}_-\|_2^2 = q. \tag{8}$$

The perturbed signal is then given by $\mathbf{x} = \mathbf{x}_+ - \mathbf{x}_-$, and q regulates the negative corruption. We regard
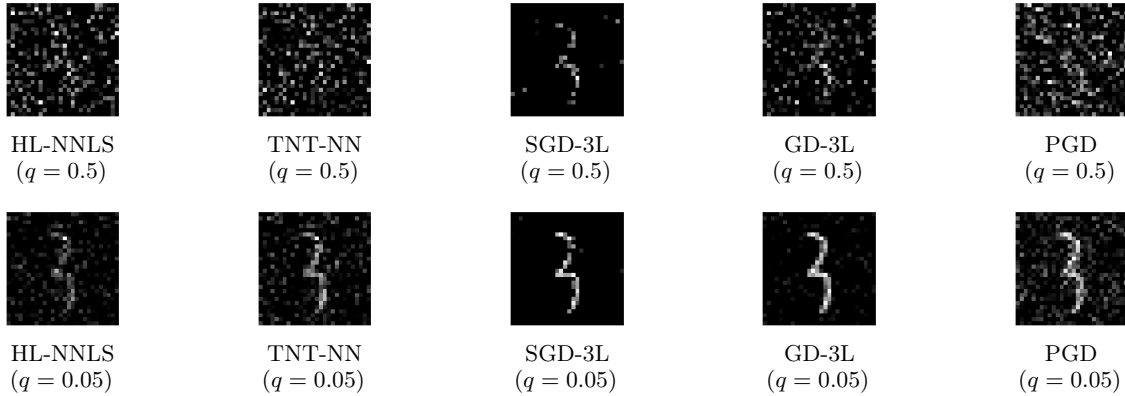
HL-NNLS ($q = 0.5$)	TNT-NN ($q = 0.5$)	SGD-3L ($q = 0.5$)	GD-3L ($q = 0.5$)	PGD ($q = 0.5$)
HL-NNLS ($q = 0.05$)	TNT-NN ($q = 0.05$)	SGD-3L ($q = 0.05$)	GD-3L ($q = 0.05$)	PGD ($q = 0.05$)

Figure 5: Illustration of the MNIST reconstruction, see Section 3.5.

the copy of $\mathbf{x}_+$ scaled to ℓ_2-norm $(1 - q)$ as ground truth and, by abuse of notation, also refer to it as $\mathbf{x}_+$. The corresponding measurements are given as $\mathbf{b} = \mathbf{A}\mathbf{x} = \mathbf{A}(\mathbf{x}_+ - \mathbf{x}_-)$.

These figures clearly show that the gradient descent-based methods outperform the established NNLS solvers. Only for small negative noise levels and MNIST data **SGD-3L** yields worse results than **HL-NNLS** and **TNT-NN**. As can be seen from Figure 5, this worse error is mainly caused by incorrect values on the support of $\mathbf{x}_+$. Visually, even for small q, the MNIST reconstruction of **SGD-3L** is far better than the one of **HL-NNLS** and **TNT-NN**. The experiment also reveals two interesting points. First, whereas it has numerically been observed by Pesme et al. (2021) that, compared to gradient descent, stochastic gradient descent reduces the generalization (resp. approximation) error if measured in ℓ_2-norm, we observe this (in the case of NNLS) only for large values of q, i.e., strong negative perturbations. For small values of q, Figure 1 rather suggests that **GD-3L** outperforms **SGD-3L**. Second, for all values of q, the solutions computed by **SGD-3L** are visually closer to the ground truth than the ones computed by **GD-3L**, cf. Figure 5. This suggests that, even in the simple context of sparse recovery, (i) the ℓ_2-norm might not be the appropriate measure for the generalization error and (ii) the stochasticity in **SGD-3L** apparently improves the generalization quality. Formalizing and proving this observation is an appealing topic for future research. We also show that, despite fast, **PGD** does not induce an implicit bias as prominent as **GD-3L** or **SGD-3L**.

CIFAR10 signals. We conduct similar experiments on perturbed CIFAR10 images in gray-scale, as shown in Figure 11. Those images are in general less sparse than MNIST images, which affects the performance of each algorithm. The most significant difference between reconstructions from MNIST and from CIFAR10 is the outcome of SGD. For CIFAR10, GD yields the best

reconstruction while SGD induces a too strong sparsity regularization and yields the worst reconstruction.

4 CONCLUSION

In this paper, we have shown that vanilla gradient flow/descent on the least squares objective can trade constraints for the complexity of the objective function. We use that to scalably solve NNLS. Our approach comes with strong theoretical guarantees, including global convergence to NNLS solutions at an $\mathcal{O}(1/t)$ rate for gradient flow and an $\mathcal{O}(1/k)$ rate for gradient descent with decaying step sizes. Moreover, we prove that an accelerated gradient flow achieves an improved $\mathcal{O}(1/t^2)$ rate and we numerically show that this scheme attains an $\mathcal{O}(1/k^3)$ decay. These results provide a new perspective on solving NNLS by leveraging the implicit bias of gradient descent, avoiding limitations such as the need for careful step size tuning and the lack of theoretical guarantees that affect some existing solvers. Our analysis also reveals intriguing connections between the overparametrization literature and classical optimization problems. We believe the techniques developed here, trading geometry by reparametrization, could potentially be extended to other constrained optimization problems, opening up an exciting new paradigm for optimization algorithm design.

There remain many promising directions for future work, including extending our discrete analysis to the accelerated case and exploring applications to large-scale problems in areas such as non-negative matrix factorization. More broadly, we see great potential in exploiting the implicit bias of gradient descent, a phenomenon that has primarily been studied for explaining neural network training, to design new optimization methods for classical numerical problems.

References

Mikhail Belkin, Daniel Hsu, Siyuan Ma, and Soumik Mandal. Reconciling modern machine-learning practice and the classical bias–variance trade-off. *Proceedings of the National Academy of Sciences*, 116 (32):15849–15854, 2019.

Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning (still) requires rethinking generalization. *Communications of the ACM*, 64(3):107–115, 2021.

Daniel Soudry, Elad Hoffer, Mor Shpigel Nacson, Suriya Gunasekar, and Nathan Srebro. The implicit bias of gradient descent on separable data. *The Journal of Machine Learning Research*, 19(1):2822–2878, 2018.

Ziwei Ji and Matus Telgarsky. The implicit bias of gradient descent on nonseparable data. In *Conference on learning theory*, pages 1772–1798. PMLR, 2019.

Suriya Gunasekar, Blake E Woodworth, Srinadh Bhojanapalli, Behnam Neyshabur, and Nati Srebro. Implicit regularization in matrix factorization. In *Advances in Neural Information Processing Systems*, pages 6151–6159, 2017.

Sanjeev Arora, Nadav Cohen, Wei Hu, and Yuping Luo. Implicit regularization in deep matrix factorization. In *Advances in Neural Information Processing Systems*, pages 7413–7424, 2019.

Hung-Hsu Chou, Carsten Gieshoff, Johannes Maly, and Holger Rauhut. Gradient descent for deep matrix factorization: Dynamics and implicit bias towards low rank. *Applied and Computational Harmonic Analysis*, 68:101595, 2024.

Kelly Geyer, Anastasios Kyrillidis, and Amir Kalev. Low-rank regularization and solution uniqueness in over-parameterized matrix sensing. In *Proceedings of the 23rd International Conference on Artificial Intelligence and Statistics*, pages 930–940, 2020.

Dominik Stöger and Mahdi Soltanolkotabi. Small random initialization is akin to spectral learning: Optimization and generalization guarantees for over-parameterized low-rank matrix reconstruction. *Advances in Neural Information Processing Systems*, 34, 2021.

Lenaic Chizat and Francis Bach. Implicit bias of gradient descent for wide two-layer neural networks trained with the logistic loss. In *Conference on learning theory*, pages 1305–1338. PMLR, 2020.

Tomas Vaskevicius, Varun Kanade, and Patrick Rebeschini. Implicit regularization for optimal sparse recovery. In *Advances in Neural Information Processing Systems*, pages 2972–2983, 2019.

Peng Zhao, Yun Yang, and Qiao-Chu He. Implicit Regularization via Hadamard Product Over-Parametrization in High-Dimensional Linear Regression. *arXiv preprint: 1903.09367*, 2019.

Hung-Hsu Chou, Johannes Maly, and Holger Rauhut. More is less: inducing sparsity via overparameterization. *Information and Inference: A Journal of the IMA*, 12(3):1437–1460, 2023a.

Gauthier Gidel, Francis Bach, and Simon Lacoste-Julien. Implicit regularization of discrete gradient dynamics in linear neural networks. In *Advances in Neural Information Processing Systems*, pages 3202–3211, 2019.

Blake Woodworth, Suriya Gunasekar, Jason D. Lee, Edward Moroshko, Pedro Savarese, Itay Golan, Daniel Soudry, and Nathan Srebro. Kernel and Rich Regimes in Overparametrized Models. In *Proceedings of Thirty Third Conference on Learning Theory*, pages 3635–3673, 2020.

Scott Pesme, Loucas Pillaud-Vivien, and Nicolas Flammarion. Implicit Bias of SGD for Diagonal Linear Networks: a Provable Benefit of Stochasticity. *Advances in Neural Information Processing Systems*, 34, 2021.

Moritz Hardt and Tengyu Ma. Identity matters in deep learning. In *International Conference on Learning Representations*, 2016.

Sanjeev Arora, Nadav Cohen, and Elad Hazan. On the optimization of deep networks: Implicit acceleration by overparameterization. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018*, pages 244–253, 2018.

Shahar Azulay, Edward Moroshko, Mor Shpigel Nacson, Blake E Woodworth, Nathan Srebro, Amir Globerson, and Daniel Soudry. On the implicit bias of initialization shape: Beyond infinitesimal mirror descent. In *International Conference on Machine Learning*, pages 468–477. PMLR, 2021.

Omer Elkabetz and Nadav Cohen. Continuous vs. discrete optimization of deep neural networks. *Advances in Neural Information Processing Systems*, 34:4947–4960, 2021.

Qiuwei Li, Daniel McKenzie, and Wotao Yin. From the simplex to the sphere: faster constrained optimization using the hadamard parametrization. *Information and Inference: A Journal of the IMA*, 12 (3):1898–1937, 2023.

David Martínez-Rubio and Sebastian Pokutta. Accelerated riemannian optimization: Handling constraints with a prox to bound geometric penalties. In *The Thirty Sixth Annual Conference on Learning Theory*, pages 359–393. PMLR, 2023.

Zhiyuan Li, Tianhao Wang, Jason D Lee, and Sanjeev Arora. Implicit bias of gradient descent on

reparametrized models: On equivalence to mirror descent. *Advances in Neural Information Processing Systems*, 35:34626–34640, 2022.

Lijun Ding and Stephen J Wright. On squared-variable formulations. *arXiv preprint arXiv:2310.01784*, 2023.

Grzegorz Stoch and Artur T Krzyżak. Enhanced Resolution Analysis for Water Molecules in MCM-41 and SBA-15 in Low-Field T2 Relaxometric Spectra. *Molecules*, 26(8):2133, 2021.

Federico Benvenuto, Riccardo Zanella, Luca Zanni, and Mario Bertero. Nonnegative least-squares image deblurring: improved gradient projection approaches. *Inverse Problems*, 26(2):025004, 2009.

Jun Beom Kho, Jaihie Kim, Ig-Jae Kim, and Andrew BJ Teoh. Cancelable fingerprint template design with randomized non-negative least squares. *Pattern Recognition*, 91:245–260, 2019.

Hyunsoo Kim, Yingtao Bi, Sharmistha Pal, Ravi Gupta, and Ramana V Davuluri. IsoformEx: isoform level gene expression estimation using weighted nonnegative least squares from mRNA-Seq data. *BMC bioinformatics*, 12(1):1–9, 2011.

Joseph M Myre, Ioan Lascu, Eduardo A Lima, Joshua M Feinberg, Martin O Saar, and Benjamin P Weiss. Using TNT-NN to unlock the fast full spatial inversion of large magnetic microscopy data sets. *Earth, Planets and Space*, 71(1):1–26, 2019.

Ernie Esser, Yifei Lou, and Jack Xin. A method for finding structured sparse solutions to nonnegative least squares problems with applications. *SIAM Journal on Imaging Sciences*, 6(4):2010–2046, 2013.

Jie Chen, Cédric Richard, José Carlos M Bermudez, and Paul Honeine. Nonnegative least-mean-square algorithm. *IEEE Transactions on Signal Processing*, 59(11):5225–5235, 2011.

Delin Chu, Weya Shi, Srinivas Eswar, and Haesun Park. An alternating rank-k nonnegative least squares framework (ARkNLS) for nonnegative matrix factorization. *SIAM Journal on Matrix Analysis and Applications*, 42(4):1451–1479, 2021.

Vladimir Vapnik. *The nature of statistical learning theory*. Springer science & business media, 1999.

Donghui Chen and Robert J Plemmons. Nonnegativity constraints in numerical analysis. In *The birth of numerical analysis*, pages 109–139. World Scientific, 2010.

Stefania Bellavia, Maria Macconi, and Benedetta Morini. An interior point Newton-like method for non-negative least-squares problems with degenerate solution. *Numerical Linear Algebra with Applications*, 13(10):825–846, 2006.

Charles L Lawson and Richard J Hanson. *Solving least squares problems.* SIAM, 1995.

Michel Bierlaire, Ph L Toint, and Daniel Tuyttens. On iterative algorithms for linear least squares problems with bound constraints. *Linear Algebra and its Applications*, 143:111–143, 1991.

Chih-Jen Lin. Projected gradient methods for nonnegative matrix factorization. *Neural computation*, 19 (10):2756–2779, 2007.

Dongmin Kim, Suvrit Sra, and Inderjit S Dhillon. A non-monotonic method for large-scale non-negative least squares. *Optimization Methods and Software*, 28(5):1012–1039, 2013.

Yu-Hong Dai and Roger Fletcher. Projected Barzilai-Borwein methods for large-scale box-constrained quadratic programming. *Numerische Mathematik*, 100(1):21–47, 2005.

Suriya Gunasekar, Jason D Lee, Daniel Soudry, and Nati Srebro. Implicit bias of gradient descent on linear convolutional networks. In *Advances in Neural Information Processing Systems*, pages 9461–9471, 2018.

Jiangyuan Li, Thanh Nguyen, Chinmay Hegde, and Ka Wai Wong. Implicit sparse regularization: The impact of depth and early stopping. In *Advances in Neural Information Processing Systems*, 2021.

Mor Shpigel Nacson, Kavya Ravichandran, Nathan Srebro, and Daniel Soudry. Implicit bias of the step size in linear diagonal neural networks. In *International Conference on Machine Learning*, pages 16270–16295. PMLR, 2022.

Yao-Liang Yu. Co673/cs794 - optimization for data science lecture notes, 2017. Accessed May 1, 2024. https://cs.uwaterloo.ca/~y328yu/mycourses/794/794-note-md.pdf.

Constantino Tsallis. Possible generalization of boltzmann-gibbs statistics. *Journal of statistical physics*, 52:479–487, 1988.

David L Donoho and Jared Tanner. Sparse nonnegative solution of underdetermined linear equations by linear programming. *Proceedings of the National Academy of Sciences*, 102(27):9446–9451, 2005.

David L Donoho and Jared Tanner. Counting the faces of randomly-projected hypercubes and orthants, with applications. *Discrete & Computational Geometry*, 43(3):522–541, 2010.

Clarice Poon and Gabriel Peyré. Smooth overparameterized solvers for non-smooth structured optimization. *Mathematical programming*, 201(1):897–952, 2023.

Rasmus Bro and Sijmen De Jong. A fast non-negativity-constrained least squares algorithm. *Journal of*

Chemometrics: A Journal of the Chemometrics Society, 11(5):393–401, 1997.

Joe M Myre, Erich Frahm, David J Lilja, and Martin O Saar. TNT-NN: a fast active set method for solving large non-negative least squares problems. *Procedia Computer Science*, 108:755–764, 2017.

Roman A Polyak. Projected gradient method for non-negative least square. *Contemp Math*, 636:167–179, 2015.

Yurii E Nesterov. A method for solving the convex programming problem with convergence rate $o(1/k^2)$. In *Dokl. akad. nauk SSSR*, volume 269, pages 543–547, 1983.

Roger Fletcher. On the Barzilai-Borwein method. In *Optimization and control with applications*, pages 235–256. Springer, 2005.

Marcos Raydan. The Barzilai and Borwein gradient method for the large scale unconstrained minimization problem. *SIAM Journal on Optimization*, 7(1): 26–33, 1997.

J.R. Dormand and P.J. Prince. A family of embedded runge-kutta formulae. *Journal of Computational and Applied Mathematics*, 6(1):19–26, 1980. ISSN 0377-0427. doi: https://doi.org/10.1016/0771-050X(80) 90013-3.

Raphaël Berthier. Incremental learning in diagonal linear networks. *Journal of Machine Learning Research*, 24(171):1–26, 2023.

Scott Pesme and Nicolas Flammarion. Saddle-to-saddle dynamics in diagonal linear networks. *Advances in Neural Information Processing Systems*, 36:7475–7505, 2023.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014.

Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning requires rethinking generalization. In *International Conference on Learning Representations*, 2017.

Behnam Neyshabur, Ryota Tomioka, Ruslan Salakhutdinov, and Nathan Srebro. Geometry of optimization and implicit regularization in deep learning. *arXiv preprint arXiv:1705.03071*, 2017.

Behnam Neyshabur, Ryota Tomioka, and Nathan Srebro. In search of the real inductive bias: On the role of implicit regularization in deep learning. In *International Conference on Learning Representations*, 2015.

Sepp Hochreiter and Jürgen Schmidhuber. Flat minima. *Neural computation*, 9(1):1–42, 1997.

Daniel Gissin, Shai Shalev-Shwartz, and Amit Daniely. The implicit bias of depth: How incremental learning drives generalization. *International Conference on Learning Representations (ICLR).*, 2020.

Fan Wu and Patrick Rebeschini. Implicit Regularization in Matrix Sensing via Mirror Descent. *Advances in Neural Information Processing Systems*, 34, 2021.

Yuqing Wang, Minshuo Chen, Tuo Zhao, and Molei Tao. Large Learning Rate Tames Homogeneity: Convergence and Balancing Effect. In *International Conference on Learning Representations*, 2022.

Sheng Liu, Zhihui Zhu, Qing Qu, and Chong You. Robust training under label noise by over-parameterization. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato, editors, *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 14153–14172. PMLR, 17–23 Jul 2022.

Chong You, Zhihui Zhu, Qing Qu, and Yi Ma. Robust recovery via implicit bias of discrepant learning rates for double over-parameterization. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 17733–17744, 2020.

Hung-Hsu Chou, Johannes Maly, and Dominik Stöger. How to induce regularization in generalized linear models: A guide to reparametrizing gradient flow. *arXiv preprint arXiv:2308.04921*, 2023b.

Walid Krichene, Alexandre Bayen, and Peter L Bartlett. Accelerated mirror descent in continuous and discrete time. *Advances in neural information processing systems*, 28, 2015.

Suriya Gunasekar, Blake Woodworth, and Nathan Srebro. Mirrorless mirror descent: A natural derivation of mirror descent. In *International Conference on Artificial Intelligence and Statistics*, pages 2305–2313. PMLR, 2021.

Sanjeev Arora, Zhiyuan Li, and Abhishek Panigrahi. Understanding gradient descent on the edge of stability in deep learning. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvári, Gang Niu, and Sivan Sabato, editors, *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, volume 162 of *Proceedings of Machine Learning Research*, pages 948–1024. PMLR, 2022.

Jingfeng Wu, Vladimir Braverman, and Jason D. Lee. Implicit bias of gradient descent for logistic regression at the edge of stability. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023.

Yuqing Wang, Zhenghao Xu, Tuo Zhao, and Molei Tao. Good regularity creates large learning rate implicit biases: edge of stability, balancing, and catapult. *arXiv preprint arXiv:2310.17087*, 2023.

Arthur Jacot, Franck Gabriel, and Clément Hongler. Neural tangent kernel: Convergence and generalization in neural networks. *Advances in Neural Information Processing Systems*, 31, 2018.

Emmanuel J. Candès and Terence Tao. Near-Optimal Signal Recovery From Random Projections: Universal Encoding Strategies? *IEEE Transactions on Information Theory*, 52(12):5406–5425, 2006. ISSN 0018-9448.

David L Donoho. Compressed sensing. *IEEE Transactions on Information Theory*, 52(4):1289–1306, 2006.

Simon Foucart and Holger Rauhut. *A Mathematical Introduction to Compressive Sensing*. Birkhäuser, New York, NY, 2013.

Aaron Berk, Yaniv Plan, and Ozgur Yilmaz. On the best choice of LASSO program given data parameters. *IEEE Transactions on Information Theory*, 2021.

Josef Stoer. On the numerical solution of constrained least-squares problems. *SIAM journal on Numerical Analysis*, 8(2):382–411, 1971.

Gene H. Golub and Michael A. Saunders. Linear least squares and quadratic programming. In Jean Abadie and Philip Wolfe, editors, *Integer and Nonlinear Programming*, pages 229–256. North Holland Pub. Co., Amsterdam, 1970.

Philip E Gill and Walter Murray. A numerically stable form of the simplex algorithm. *Linear algebra and its applications*, 7(2):99–138, 1973.

Mark H Van Benthem and Michael R Keenan. Fast algorithm for the solution of large-scale non-negativity-constrained least squares problems. *Journal of Chemometrics: A Journal of the Chemometrics Society*, 18(10):441–450, 2004.

Yuancheng Luo and Ramani Duraiswami. Efficient parallel nonnegative least squares on multicore architectures. *SIAM Journal on Scientific Computing*, 33(5):2848–2863, 2011.

Monica Dessole, Marco Dell'Orto, and Fabio Marcuzzi. The lawson-hanson algorithm with deviation maximization: Finite convergence and sparse recovery.

Numerical Linear Algebra with Applications, page e2490, 2021.

Ake Björck. *Numerical Methods for Least Squares Problems*. Society for Industrial and Applied Mathematics, 1996.

L.M. Bregman. The relaxation method of finding the common point of convex sets and its application to the solution of problems in convex programming. *USSR Computational Mathematics and Mathematical Physics*, 7(3):200–217, 1967. ISSN 0041-5553.

Sébastien Bubeck et al. Convex optimization: Algorithms and complexity. *Foundations and Trends® in Machine Learning*, 8(3-4):231–357, 2015.

Nicolas Boumal. *An introduction to optimization on smooth manifolds*. Cambridge University Press, 2023.

David L Donoho, Iain M Johnstone, Jeffrey C Hoch, and Alan S Stern. Maximum entropy and the nearly black object. *Journal of the Royal Statistical Society: Series B (Methodological)*, 54(1):41–67, 1992.

Alfred M Bruckstein, Michael Elad, and Michael Zibulevsky. On the uniqueness of nonnegative sparse solutions to underdetermined systems of equations. *IEEE Transactions on Information Theory*, 54(11):4813–4820, 2008.

Meng Wang, Weiyu Xu, and Ao Tang. A unique "nonnegative" solution to an underdetermined system: From vectors to matrices. *IEEE Transactions on Signal Processing*, 59(3):1007–1016, 2010.

Richard Kueng and Peter Jung. Robust nonnegative sparse recovery and the nullspace property of 0/1 measurements. *IEEE Transactions on Information Theory*, 64(2):689–703, 2017.

Yonatan Shadmi, Peter Jung, and Giuseppe Caire. Sparse non-negative recovery from biased subgaussian measurements using NNLS. *arXiv preprint arXiv:1901.05727*, 2019.

Martin Slawski and Matthias Hein. Sparse recovery by thresholded non-negative least squares. *Advances in Neural Information Processing Systems*, 24, 2011.

Martin Slawski and Matthias Hein. Non-negative least squares for high-dimensional linear models: Consistency and sparse recovery without regularization. *Electronic Journal of Statistics*, 7:3004–3056, 2013.

Nicolai Meinshausen. Sign-constrained least squares estimation for high-dimensional regression. *Electronic Journal of Statistics*, 7:1607–1631, 2013.

Holger Rauhut and Rachel Ward. Interpolation via weighted ℓ_1-minimization. *Applied and Computational Harmonic Analysis*, 40(2):321–351, 2016.

Ji Peng, Jerrad Hampton, and Alireza Doostan. A weighted ℓ_1-minimization approach for sparse poly-

nomial chaos expansions. *Journal of Computational Physics*, 267:92–111, 2014.

Hedy Attouch, Jérôme Bolte, and Benar Fux Svaiter. Convergence of descent methods for semi-algebraic and tame problems: proximal algorithms, forward–backward splitting, and regularized gauss–seidel methods. *Mathematical Programming*, 137(1):91–129, 2013.

Jonathan Barzilai and Jonathan M Borwein. Two-point step size gradient methods. *IMA journal of numerical analysis*, 8(1):141–148, 1988.

Vincent Roulet and Alexandre d'Aspremont. Sharpness, restart and acceleration. *Advances in Neural Information Processing Systems*, 30, 2017.

Supplementary Materials to the paper "Get rid of your constraints and reparametrize: A study in NNLS and implicit bias"

This supplementary material provides additional details, proofs, and experiments to support the main paper. The contents are organized as follows:

In Appendix A, we provide an extensive review of related work, covering both the literature on overparameterization and implicit regularization as well as previous approaches to solving non-negative least squares problems. Appendix B offers a detailed proof of Remark 1.1, establishing the connection between our approach and overparameterized models. Appendix C contains the complete proof for Theorem 2.1 which is central to our convergence analysis. In Appendix D, we derive the Riemannian metric and Riemannian gradient associated with our reparameterized flow. Appendix E presents the proof of Theorem 2.4, demonstrating the accelerated convergence rate for our proposed continuous dynamics. Appendix F provides a comprehensive proof of Theorem 2.6, establishing convergence rates for discrete gradient descent. In Appendix G, we explore the application of our method to sparse recovery problems, including the proof of Theorem G.6 on stability as well as the proof of Remark G.7 discussing weighted ℓ_1-norm regularization. Finally, Appendices H and I present implementation details of the accelerated methods and additional numerical experiments, including large-scale problems and comparisons with projected gradient descent, to further validate our theoretical results and demonstrate the practical efficacy of our approach.

Contents

A   RELATED WORK — EXTENDED

The results we presented in Section 2 unite two very different and (up to this point) independent lines of mathematical research: the decade-old question of how to solve NNLS in an efficient way and the rather recent question of what kind of implicit bias gradient descent exhibits.

A.1   Related work — Overparameterization and implicit regularization

The recent success of overparametrization in machine learning models (Goodfellow et al., 2014) has raised the question of how a plain algorithm like (stochastic) gradient descent can succeed in solving highly non-convex optimization problems like the training of deep neural networks and, in particular, find "good" solutions, i.e., parameter configurations for which the network generalizes well to unseen data. The objective functions in such tasks typically have infinitely many global minimizers (usually there are infinitely many networks fitting the training samples exactly in the overparameterized regime (Zhang et al., 2017)), which means that the choice of algorithm strongly influences which minimum is picked. This intrinsic tendency of an optimization method towards minimizers of a certain shape has been dubbed "implicit bias" and, in the case of gradient descent, has led to an active line of research in the past few years.

Based on numerical simulations, several works (Neyshabur et al., 2017, 2015; Zhang et al., 2017, 2021) systematically studied this implicit bias of gradient descent in the training of deep neural networks. Whereas it is not even clear by now how to measure the implicit bias — one could quantify it, e.g., in low complexity (Neyshabur et al., 2015) or in high generalizability (Hochreiter and Schmidhuber, 1997) —, the empirical studies observed that the factorized structure of such networks is crucial for successful training. However, due to the complexity of the model class, there is still no corresponding thorough theoretical analysis/understanding available. To close the gap between theory and practice, simplified "training" models have been proposed and analyzed in a great number

of works: vector factorization for data with (Vaskevicius et al., 2019; Zhao et al., 2019) or without (Arora et al., 2018; Gunasekar et al., 2018; Chou et al., 2023a) restricted isometry property, and matrix factorization with deep structure (Arora et al., 2019; Chou et al., 2024; Gunasekar et al., 2017; Neyshabur et al., 2017, 2015) or shallow structure (Geyer et al., 2020; Gidel et al., 2019; Gissin et al., 2020; Stöger and Soltanolkotabi, 2021), kernel-like models (Soudry et al., 2018; Woodworth et al., 2020), mirror descent (Wu and Rebeschini, 2021), together with techniques such as balancing (Wang et al., 2022), early stopping (Li et al., 2021), and analysis regarding to the robustness of those models (Liu et al., 2022; You et al., 2020).

Essentially, all of these works agree in the point that, if initialized close to zero and applied to a plain least-squares formulation of factorized shape, cf. $\mathcal{L}_{\text{over}}$ in (4), vanilla gradient descent/flow exhibits an implicit bias towards global minimizers that are sparse (vector case) resp. of low-rank (matrix case). This is remarkable since it shows that in overparametrized regimes, gradient descent has an implicit tendency toward "simple" solutions. In particular, it has been observed that such an implicit bias of gradient descent can be explained by three complementary perspectives: implicit reparametrization of the underlying gradient flow (Neyshabur et al., 2017; Li et al., 2022; Chou et al., 2023b), mirror flow (Krichene et al., 2015; Gunasekar et al., 2021; Li et al., 2022), and Riemannian gradient flow (Li et al., 2022). One can roughly distinguish three phases on the way from the first partial results that aimed to understand the surprising properties of GD in the overparametrized setting to a general research program focused on the theory of implicit bias: (i.) In the beginning, researchers characterized the limit (Gunasekar et al., 2017; Arora et al., 2019) assuming convergence; in many instances, they needed additional assumptions such as infinitesimal initialization. Such a characterization can also be analyzed from a mirror flow perspective (Krichene et al., 2015; Gunasekar et al., 2021; Li et al., 2022). (ii.) Once the characterization has been established, many works started to derive sufficient conditions for convergence (Gidel et al., 2019; Chou et al., 2023a; Li et al., 2022) and extended the results to a range of (small) initialization as opposed to infinitesimal initialization. (iii.) Most of the above-mentioned works only analyze gradient flow since it is in general difficult to establish convergence rate results for GD in the overparametrized regime due to non-convexity in the loss function. In principle, recent work on reparametrized gradient flow (Chou et al., 2023b) suggests linear convergence of gradient methods on overparametrized linear regression. Due to a strong dependence of the therein appearing constants on the conditioning of the optimization problem, however, the obtained linear rates do not explain the convergence rates observed in practice. Another direction is to take step sizes as large as possible to enter the edge of stability regime (Arora et al., 2022; Wu et al., 2023; Wang et al., 2023) for which hardly any theory exists so far. In the infinite width regime, such results can even be connected to general neural networks via the neural tangent kernel (Jacot et al., 2018). However, fully understanding the implicit bias of GD in the training of finite-width networks remains a challenge.

Whereas the obtained insights on vector and matrix factorization might not yet resolve the mystery of deep learning, they provide valuable tools for more classical problems. One such example is sparse recovery, which lies at the interface of high-dimensional statistics and signal processing. Over the past two decades, several methods have been developed to recover intrinsically low-dimensional data, such as sparse vectors or low-rank matrices. The underlying theory, which became known under the name compressive sensing (Candès and Tao, 2006; Donoho, 2006; Foucart and Rauhut, 2013), establishes conditions under which such data can be uniquely and efficiently reconstructed. Nonetheless, under noise corruption, most existing methods require tuning hyper-parameters with respect to the (in principle unknown) noise level. It is still an intense topic of research to understand such tuning procedures (Berk et al., 2021). As already mentioned by Vaskevicius et al. (2019); Chou et al. (2023a), the implicit sparsity regularization creates a bridge between the recent studies on gradient descent and compressive sensing. In fact, the overparametrized gradient descent provides a tuning-free alternative to established algorithms like LASSO and Basis Pursuit. In a similar manner, our present contribution stems from the insight that in most of the above papers (Arora et al., 2018, 2019; Chou et al., 2024, 2023a), the signs of components do not change over time when gradient flow is applied. Instead of viewing this feature as an obstacle (e.g., Chou et al., 2023a, Section 2), we use it to naturally link the implicit bias of gradient descent/flow to another ubiquitous problem of numerical mathematics, namely (NNLS).

A.2 Related work — NNLS

The first algorithm proposed to solve (NNLS) appeared in 1974 in the book of Lawson and Hanson (1995), Chapter 23, where its finite convergence was proved, and a Fortran routine was presented.[9] Like the previous papers (Stoer, 1971; Golub and Saunders, 1970; Gill and Murray, 1973) it builds upon the solution of linear systems. Similar to the simplex method, the algorithm is an active-set algorithm that iteratively sets parts of the variables to zero in an attempt to identify the active constraints and solves the unconstrained least squares sub-problem for this active set of constraints. It is still, arguably, the most famous method for solving (NNLS), and several improvements have been proposed in a series of follow-up papers (Bro and De Jong, 1997), (Van Benthem and Keenan, 2004), (Myre et al., 2017), (Luo and Duraiswami, 2011), (Dessole et al., 2021). Its caveat, however, is that it depends on the normal equations, which makes it infeasible for ill-conditioned or large-scale problems. Moreover, up to this point, there exist no better theoretical guarantees for the algorithm and its modifications than convergence in finitely many steps (Lawson and Hanson, 1995, Chapter 23), (Dessole et al., 2021, Theorem 3).

Another line of research has been developing projected gradient methods for solving (NNLS), which come with linear convergence guarantees (Kim et al., 2013), (Polyak, 2015) (Lin, 2007). In contrast to active set and interior point methods, projected gradient methods do not require solving a linear system of equations at each step and thus scale better in high-dimensional problems. However, the success of these methods heavily depends on a good choice of step size and, as we already highlighted in Section 1, the projection step may render established acceleration methods for vanilla gradient descent useless.

B PROOF OF REMARK 1.1

Since the problem in (NNLS) is convex, the Karush-Kuhn-Tucker (KKT) conditions are necessary and sufficient for optimality (e.g., Björck, 1996).

Theorem B.1. *(Karush-Kuhn-Tucker conditions for NNLS) A point $\mathbf{x}_+ \in \mathbb{R}^N$ is a solution of problem (NNLS) if and only if there exists $\boldsymbol{\mu}^* \in \mathbb{R}^n$ such that*

$$\boldsymbol{\mu}^* = \mathbf{A}^T(\mathbf{A}\mathbf{x}_+ - \mathbf{b}), \quad \boldsymbol{\mu}^* \odot \mathbf{x}_+ = \mathbf{0}, \quad \boldsymbol{\mu}^*, \mathbf{x}_+ \geq \mathbf{0}.$$

From these conditions, one can observe that all solutions of (NNLS) can be represented by stationary points of the functional

$$\mathcal{L}_{\text{over}}\big(\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(L)}\big) := \frac{1}{2} \left\| \mathbf{A}\big(\mathbf{x}^{(1)} \odot \cdots \odot \mathbf{x}^{(L)}\big) - \mathbf{b} \right\|_2^2$$

in (4), i.e., points that satisfy, for all $\ell \in [L]$, the equation

$$\nabla \mathcal{L}_{\mathbf{x}^{(\ell)}}\big(\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(L)}\big) = \left[\mathbf{A}^{\mathrm{T}}\Big(\mathbf{A}\big(\mathbf{x}^{(1)} \odot \cdots \odot \mathbf{x}^{(L)}\big) - \mathbf{b}\Big) \right] \odot \left(\bigodot_{k \neq \ell} \mathbf{x}^{(k)} \right) = \mathbf{0}. \tag{9}$$

Indeed, for any given optimal point $\mathbf{x}_+$ of the (NNLS) problem, it is straightforward to check that the conditions in Theorem B.1 imply that $\mathbf{x}^{(1)} = \cdots = \mathbf{x}^{(L)} = \mathbf{x}_+^{\odot \frac{1}{L}}$ is a stationary point of (4) with $\mathbf{x}^{(1)} \odot \cdots \odot \mathbf{x}^{(L)} = \mathbf{x}_+$. The same argument holds for stationary points of the reduced functional (3). In particular, this implies that any solution of (NNLS) can be described as the limit of gradient flow on (4) under suitably chosen *identical* initialization.

C PROOFS OF THEOREMS 2.1 AND G.4

Let us for convenience recall the statements of Theorems 2.1 and G.4 before starting the proof. For simplicity, we merge both results into a single statement.

Theorem C.1. *Let $L \geq 2$, $\mathbf{A} \in \mathbb{R}^{M \times N}$ and $\mathbf{b} \in \mathbb{R}^M$. Let $\mathbf{x}_0 > \mathbf{0}$ be fixed and let $\mathbf{x}(t)$ follow the flow $\mathbf{x}'(t) = -\nabla \mathcal{L}(\mathbf{x}(t))$ with $\mathbf{x}(0) = \mathbf{x}_0$. Let S_+ be the set defined in (NNLS) and set $\tilde{\mathbf{x}} = \mathbf{x}^{\odot L}$.*

---

[9]This algorithm is the standard one implemented in many languages: *optimize.nnls* in the SciPy package, *nnls* in R, *lsqnonneg* in MATLAB and *nnls.jl* in Julia.

Then the limit $\tilde{\mathbf{x}}_\infty := \lim_{t\to\infty} \tilde{\mathbf{x}}(t)$ exists and lies in S_+. Also, there exists an absolute constant $C > 0$ that only depends on the choice of $\mathbf{A}$, $\mathbf{b}$, and $\mathbf{x}_0$ such that

$$\ell(\tilde{\mathbf{x}}(t)) - \ell(\mathbf{x}_+) \leq \frac{C}{t},$$

for any $t > 0$. Let $\varepsilon > 0$ and assume in addition that $\mathbf{x}_0 = \alpha\mathbf{1}$. If

$$\alpha \leq h(Q_+, \varepsilon) := \begin{cases} \exp\left(-\frac{1}{2} - \frac{Q_+^2 + Ne^{-1}}{2\varepsilon}\right) & \text{if } L = 2 \\ \left(\frac{2\varepsilon}{L(Q_+ + N + \varepsilon)}\right)^{\frac{1}{L-2}} & \text{if } L > 2 \end{cases}, \tag{10}$$

where $Q_+ = \min_{\mathbf{z} \in S_+} \|\mathbf{z}\|_1$, then the ℓ_1-norm of $\tilde{\mathbf{x}}_\infty$ satisfies

$$\|\tilde{\mathbf{x}}_\infty\|_1 - \min_{\mathbf{z}\in S_+} \|\mathbf{z}\|_1 \leq \varepsilon.$$

The proof consists of two major steps: First, we prove that the flow on (3) converges and characterize its limits as the minimizer of a specific optimization problem. Second, we show that if $\mathbf{x}_0 = \alpha\mathbf{1}$, the limit approximately minimizes the ℓ_1-norm among all possible solutions of (NNLS). Whereas the second step, including the non-asymptotic dependence between α and ε, is taken unchanged from Chou et al. (2023a), the first step, which is the backbone of the argument, requires a different reasoning due to the (possible) non-existence of solutions $\mathbf{A}\mathbf{z} = \mathbf{b}$. We will rely on the concept of Bregman divergence.

Definition C.2 (Bregman Divergence). *Let $F : \Omega \to \mathbb{R}$ be a continuously-differentiable, strictly convex function defined on a closed convex set Ω. The Bregman divergence associated with F for points $p, q \in \Omega$ is defined as*

$$D_F(p, q) = F(p) - F(q) - \langle \nabla F(q), p - q \rangle. \tag{11}$$

Lemma C.3 ((Bregman, 1967)). *The Bregman divergence D_F is non-negative and strictly convex in p.*

Theorem C.5 below, which shows convergence of $\mathbf{x}^{\odot L}$ and characterizes the limit $\tilde{\mathbf{x}}_\infty$, resembles Theorem 5 by Chou et al. (2023a). Note, however, that the definition of S_+ is different and that Theorem C.5 does not require the existence of a solution of $\mathbf{A}\mathbf{x} = \mathbf{b}$. Moreover, we consider a function $F : \mathbb{R}_{\geq 0}^N \to \mathbb{R}$ that differs slightly from its counterpart in Eq. (2.7) of Chou et al. (2023a) and is based on an important concept that appears in thermodynamics, namely, the so-called *Tsallis q-logarithm*, introduced in a seminal paper by Tsallis (1988). It is given by

$$F(\mathbf{x}) = \langle \mathbf{x}, \ln_q(\mathbf{x}) \rangle, \tag{12}$$

where $q = 2 - \frac{2}{L} \in [1, 2)$ and the q-logarithm is defined as

$$\ln_q(u) = \begin{cases} \frac{x^{1-q}-1}{1-q}, & \text{if } q \neq 1; \\ \ln(x), & \text{if } q = 1, \end{cases}$$

where we set $0\log(0) = \lim_{z\to 0} z \log(z) = 0$. The q-logarithm $\ln_q(u)$ has the following basic properties:

(i) $\ln_q(1/u) = -u^{q-1}\ln_q(u)$;

(ii) $\ln_q(uv) = \ln_q(u) + \ln_q(v) + (1-q)\ln_q(u)\ln_q(v)$

(iii) $\ln_q(u/v) = v^{q-1}(\ln_q(u) - \ln_q(v))$

Furthermore, its derivative is given by $\frac{d}{du}(\ln_q(u)) = u^{-q}$. Using these properties, it follows that the gradient of F is

$$\nabla F(\mathbf{x}) = \ln_q(\mathbf{x}) + \mathbf{x}^{\odot 1-q} \tag{13}$$

and that the Hessian of F is

$$\nabla^2 F(\mathbf{x}) = (2 - q)\text{diag}(\mathbf{x}^{\odot -q}). \tag{14}$$

In particular, the function F is convex, since $q = 2 - 2/L < 2$ and, therefore, $\nabla^2 F(\mathbf{x}) \succcurlyeq 0$. Thus, the Bregman divergence D_F is well-defined. Note that F can be written as

$$F(\mathbf{x}) = \langle \mathbf{x}, \ln_q(\mathbf{x}) \rangle = \begin{cases} \frac{1}{2} \langle \mathbf{x} \log \mathbf{x}, \mathbf{1} \rangle & \text{if } L = 2 \\ \frac{L}{2-L} \langle \mathbf{x}^{\odot \frac{2}{L}} - \mathbf{x}, \mathbf{1} \rangle & \text{if } L > 2, \end{cases} \tag{15}$$

which is worthwhile to mention since it differs from F in Eq. (2.7) of Chou et al. (2023a) only by affine shifts, i.e., both variants of F induce the same Bregman divergence D_F and the following observation can directly be transferred.

Lemma C.4 (Lemma 7 of Chou et al. (2023a))**.** *Let F be the function defined in (15), $\mathbf{z} \geq \mathbf{0}$ be fixed, and $\mathbf{x}(t) \colon \mathbb{R}_{\geq 0} \to \mathbb{R}^N_{\geq 0}$ be a continuous function with $\mathbf{x}(0) > \mathbf{0}$ and $\|\mathbf{x}(t)\|_2 \to \infty$. Then $D_F(\mathbf{z}, \mathbf{x}(t)) \to \infty$.*

We are now ready to state Theorem C.5.

Theorem C.5. *Let $\tilde{\mathbf{x}}(t) = \mathbf{x}(t)^{\odot L}$ and*

$$\mathbf{x}'(t) = -\nabla \mathcal{L}(\mathbf{x}(t)) = - \left[\mathbf{A}^T (\mathbf{A} \mathbf{x}^{\odot L}(t) - \mathbf{b}) \right] \odot \mathbf{x}^{\odot L-1}(t) \tag{16}$$

with $\mathbf{x}(0) \geq 0$. Then $\tilde{\mathbf{x}}_\infty := \lim_{t \to \infty} \tilde{\mathbf{x}}(t)$ exists and

$$\tilde{\mathbf{x}}_\infty \in \underset{\mathbf{z} \in S_+}{\arg\min} \, D_F(\mathbf{z}, \tilde{\mathbf{x}}(0)), \tag{17}$$

where S_+ is defined in (NNLS). Moreover, there exists an absolute constant $C > 0$ that only depends on the choice of $\mathbf{A}$, $\mathbf{b}$, and $\mathbf{x}_0$ such that, for all $t > 0$,

$$\|\mathbf{A} \tilde{\mathbf{x}}(t) - \mathbf{b}\|_2^2 - \|\mathbf{A} \mathbf{x}_+ - \mathbf{b}\|_2^2 \leq \frac{C}{t}.$$

In the proof of Theorem C.5, we require the following geometric observation.

Lemma C.6. *Let $\mathcal{C} \subset \mathbb{R}^n$ be a convex and closed set. Let $\mathbf{w} \in \mathcal{C}$, $\mathbf{b} \in \mathbb{R}^n$ with $\mathbf{b} \neq \mathbf{w}$, and let $\mathcal{P}_\mathcal{C} \mathbf{b}$ denote the (unique) Euclidean projection of $\mathbf{b}$ onto $\mathcal{C}$. Then*

$$\langle \mathbf{w} - \mathbf{b}, \mathbf{w} - \mathcal{P}_\mathcal{C} \mathbf{b} \rangle \geq \frac{1}{2} \|\mathbf{w} - \mathcal{P}_\mathcal{C} \mathbf{b}\|_2^2. \tag{18}$$

Furthermore, it holds that

$$\|\mathbf{w} - \mathcal{P}_\mathcal{C} \mathbf{b}\|_2^2 \leq \|\mathbf{w} - \mathbf{b}\|_2^2 - \|\mathbf{b} - \mathcal{P}_\mathcal{C} \mathbf{b}\|_2^2.$$

Proof. By definition of the projection, we have that

$$\|\mathbf{w} - \mathbf{b}\|_2^2 \geq \|\mathcal{P}_\mathcal{C} \mathbf{b} - \mathbf{b}\|_2^2 = \| -(\mathbf{w} - \mathcal{P}_\mathcal{C} \mathbf{b}) + \mathbf{w} - \mathbf{b}\|_2^2$$
$$= \|\mathbf{w} - \mathcal{P}_\mathcal{C} \mathbf{b}\|_2^2 - 2\langle \mathbf{w} - \mathcal{P}_\mathcal{C} \mathbf{b}, \mathbf{w} - \mathbf{b} \rangle + \|\mathbf{w} - \mathbf{b}\|_2^2.$$

Rearranging the terms yields

$$2\langle \mathbf{w} - \mathbf{b}, \mathbf{w} - \mathcal{P}_\mathcal{C} \mathbf{b} \rangle \geq \|\mathbf{w} - \mathcal{P}_\mathcal{C} \mathbf{b}\|_2^2$$

and thus the first claim. Now note that $\langle \mathbf{w} - \mathcal{P}_\mathcal{C} \mathbf{b}, \mathbf{b} - \mathcal{P}_\mathcal{C} \mathbf{b} \rangle \leq 0$ by $\mathcal{P}_\mathcal{C}$ being a convex projection (e.g., Bubeck et al., 2015, Lemma 3.1). This yields

$$\|\mathbf{w} - \mathbf{b}\|_2^2 = \|\mathbf{w} - \mathcal{P}_\mathcal{C} \mathbf{b} - (\mathbf{b} - \mathcal{P}_\mathcal{C} \mathbf{b})\|_2^2$$
$$= \|\mathbf{w} - \mathcal{P}_\mathcal{C} \mathbf{b}\|_2^2 - 2\langle \mathbf{w} - \mathcal{P}_\mathcal{C} \mathbf{b}, \mathbf{b} - \mathcal{P}_\mathcal{C} \mathbf{b} \rangle + \|\mathbf{b} - \mathcal{P}_\mathcal{C} \mathbf{b}\|_2^2$$
$$\geq \|\mathbf{w} - \mathcal{P}_\mathcal{C} \mathbf{b}\|_2^2 + \|\mathbf{b} - \mathcal{P}_\mathcal{C} \mathbf{b}\|_2^2$$

and thus the second claim. $\square$

With these tools at hand, we can finally provide a proof for Theorem C.5.

Proof of Theorem C.5. Let us begin with a brief outline of the proof. We will first compute the time derivative of $D_F(\mathbf{x}_+, \tilde{\mathbf{x}}(t))$, for $\mathbf{x}_+ \in S_+$ where S_+ was defined in (NNLS). These first steps appeared before in the proof of Lemma 8 by Chou et al. (2023a) and are only detailed for the reader's convenience. With the help of Lemma C.6 we can then show that $\mathbf{A}\tilde{\mathbf{x}}(t) \to \mathbf{b}_+$ where $\mathbf{b}_+$ has been defined in (5). This part of the proof requires more effort than in existing works (e.g., Chou et al., 2023a) due to the possible non-existence of pre-images for $\mathbf{b}$ under $\mathbf{A}$. We wrap up (i.e., deduce boundedness and the convergence of $\|\tilde{\mathbf{x}}(t)\|_2$ via Lemma C.4, and characterize the limit $\tilde{\mathbf{x}}_\infty$) by following the steps of Chou et al. (2023a). Keep in mind that the definition of the set S_+ is different from the one used by Chou et al. (2023a).

We start with computing $\partial_t D_F(\mathbf{z}, \tilde{\mathbf{x}}(t))$. For any $\mathbf{z} \in \mathbb{R}^N$, we have

$$
\begin{aligned}
\partial_t D_F(\mathbf{z}, \tilde{\mathbf{x}}(t)) &= \partial_t \left[F(\mathbf{z}) - F(\tilde{\mathbf{x}}(t)) - \langle \nabla F(\tilde{\mathbf{x}}(t)), \mathbf{z} - \tilde{\mathbf{x}}(t) \rangle \right] \\
&= 0 - \langle \nabla F(\tilde{\mathbf{x}}(t)), \tilde{\mathbf{x}}'(t) \rangle - \langle \partial_t \nabla F(\tilde{\mathbf{x}}(t)), \mathbf{z} - \tilde{\mathbf{x}}(t) \rangle + \langle \nabla F(\tilde{\mathbf{x}}(t)), \tilde{\mathbf{x}}'(t) \rangle \\
&= -\langle \partial_t \nabla F(\tilde{\mathbf{x}}(t)), \mathbf{z} - \tilde{\mathbf{x}}(t) \rangle.
\end{aligned}
$$

Using (14), we calculate that

$$
\begin{aligned}
\partial_t \nabla F(\tilde{\mathbf{x}}(t)) &= \nabla^2 F(\tilde{\mathbf{x}}(t)) \cdot \tilde{\mathbf{x}}'(t) = \frac{2}{L} \tilde{\mathbf{x}}(t)^{\odot(-2+\frac{2}{L})} \odot \tilde{\mathbf{x}}'(t) = \frac{2}{L} \mathbf{x}(t)^{\odot(-2L+2)} \odot \left(L\mathbf{x}(t)^{\odot(L-1)} \odot \mathbf{x}'(t) \right) \\
&= 2\mathbf{x}(t)^{\odot(-2L+2)} \odot \mathbf{x}(t)^{\odot(L-1)} \odot \left(-\left[\mathbf{A}^{\mathrm{T}}(\mathbf{A}\mathbf{x}^{\odot L}(t) - \mathbf{b}) \right] \odot \mathbf{x}^{\odot L-1}(t) \right) \\
&= -2 \left[\mathbf{A}^{\mathrm{T}}(\mathbf{A}\tilde{\mathbf{x}}(t) - \mathbf{b}) \right]
\end{aligned}
$$

By definition, it is clear that $S_+ \neq \emptyset$. Let $\mathbf{x}_+$ be any element of S_+. Since $C_+ = \{\mathbf{A}\mathbf{z} : \mathbf{z} \in \mathbb{R}_{\geq 0}\} \subset \mathbb{R}^M$ is a closed convex set and $\mathbf{b}_+ = \mathbb{P}_{C_+}\mathbf{b}$, we can set $\mathcal{C} = C_+$, $\mathbf{w} = \mathbf{A}\tilde{\mathbf{x}}(t) \in C_+$, and $\mathbf{b} = \mathbf{b}$ in Lemma C.6 and obtain that

$$
\begin{aligned}
\partial_t D_F(\mathbf{x}_+, \tilde{\mathbf{x}}(t)) &= -2\langle \mathbf{A}\tilde{\mathbf{x}}(t) - \mathbf{b}, \mathbf{A}\tilde{\mathbf{x}}(t) - \mathbf{A}\mathbf{x}_+ \rangle \\
&\leq -\|\mathbf{A}\tilde{\mathbf{x}}(t) - \mathbf{b}_+\|_2^2.
\end{aligned}
\tag{19}
$$

Note that $D_F(\mathbf{x}_+, \tilde{\mathbf{x}}(t))$ must converge for $t \to \infty$ as it is non-negative and non-increasing. In particular, $0 \leq D_F(\mathbf{x}_+, \tilde{\mathbf{x}}(t)) \leq D_F(\mathbf{x}_+, \tilde{\mathbf{x}}(0))$ for all $t \geq 0$.

We now show that $\mathbf{A}\tilde{\mathbf{x}}(t) \to \mathbf{b}_+$ with the given rate. To this end, we define the Lyapunov function

$$
\mathcal{E}(t) = \frac{t}{2}(\|\mathbf{A}\tilde{\mathbf{x}}(t) - \mathbf{b}\|_2^2 - \|\mathbf{b}_+ - \mathbf{b}\|_2^2) + \frac{1}{2}D_F(\mathbf{x}_+, \tilde{\mathbf{x}}(t)).
$$

The derivative $\mathcal{E}'(t)$ is given by

$$
\begin{aligned}
\mathcal{E}'(t) &= \frac{1}{2}\left(\|\mathbf{A}\tilde{\mathbf{x}}(t) - \mathbf{b}\|_2^2 - \|\mathbf{b} - \mathbf{b}_+\|_2^2 \right) + \frac{t}{2}\partial_t\|\mathbf{A}\tilde{\mathbf{x}}(t) - \mathbf{b}\|_2^2 + \frac{1}{2}\partial_t D_F(\mathbf{x}_+, \tilde{\mathbf{x}}(t)) \\
&= \ell(\tilde{\mathbf{x}}(t)) - \ell(\mathbf{x}_+) + Lt\langle \nabla\mathcal{L}(\mathbf{x}(t)), \mathbf{x}'(t) \rangle - \langle \mathbf{A}^T(\mathbf{A}\tilde{\mathbf{x}}(t) - \mathbf{b}), \mathbf{x}_+ - \tilde{\mathbf{x}}(t) \rangle \\
&= \ell(\tilde{\mathbf{x}}(t)) - \ell(\mathbf{x}_+) - Lt\|\nabla\mathcal{L}(\mathbf{x}(t))\|_2^2 + \langle \nabla\ell(\tilde{\mathbf{x}}(t)), \mathbf{x}_+ - \tilde{\mathbf{x}}(t) \rangle \\
&= -D_\ell(\mathbf{x}_+, \tilde{\mathbf{x}}(t)) - Lt\|\nabla\mathcal{L}(\mathbf{x}(t))\|_2^2 \leq 0.
\end{aligned}
$$

Hence, the function $\mathcal{E}(t)$ is non-increasing, which yields

$$
t\left(\|\mathbf{A}\tilde{\mathbf{x}}(t) - \mathbf{b}\|_2^2 - \|\mathbf{b} - \mathbf{b}_+\|_2^2 \right) \leq 2\mathcal{E}(t) \leq 2\mathcal{E}(0) = D_F(\mathbf{x}_+, \tilde{\mathbf{x}}(0)).
$$

This implies that $\|\mathbf{A}\tilde{\mathbf{x}}(t) - \mathbf{b}\|_2^2$ converges to $\|\mathbf{b} - \mathbf{b}_+\|_2^2$ with order $\mathcal{O}(1/t)$. In particular, we conclude that $\lim_{t\to\infty} \mathbf{A}\tilde{\mathbf{x}}(t) = \mathbf{b}_+$ since $\mathbf{b}_+$ is the unique projection of $\mathbf{b}$ onto C_+ as defined in (5).

Using Lemma C.4, boundedness of $\|\tilde{\mathbf{x}}(t)\|_2$ follows as in the work of Chou et al. (2023a). Just fix any $\mathbf{x}_+ \in S_+$ and notice that $D_F(\mathbf{x}_+, \tilde{\mathbf{x}}(t)) \to \infty$ if $\|\tilde{\mathbf{x}}(t)\|_2 \to \infty$ which contradicts the fact that $0 \leq D_F(\mathbf{x}_+, \tilde{\mathbf{x}}(t)) \leq D_F(\mathbf{x}_+, \tilde{\mathbf{x}}(0))$. Let us denote by B a sufficiently large compact ball around the origin such that $B \cap S_+ \neq \emptyset$ and $\tilde{\mathbf{x}}(t) \in B$, for all $t \geq 0$.

Now assume that there exists no $\mathbf{x}_+ \in S_+ \cap B$ such that $\lim_{t\to\infty} D_F(\mathbf{x}_+, \tilde{\mathbf{x}}(t)) = 0$, i.e., by compactness of $S_+ \cap B$ there exists $\varepsilon > 0$ such that $\lim_{t\to\infty} D_F(\mathbf{x}_+, \tilde{\mathbf{x}}(t)) > \varepsilon$, for all $\mathbf{x}_+ \in S_+ \cap B$. By strict convexity of $D_F(\cdot, \tilde{\mathbf{x}}(t))$, this implies that $\tilde{\mathbf{x}}$ is bounded away from the set S_+ on B and $\|\mathbf{A}\tilde{\mathbf{x}}(t) - \mathbf{b}_+\|_2$ cannot converge to zero contradicting the just obtained convergence $\lim_{t\to\infty} \mathbf{A}\tilde{\mathbf{x}}(t) = \mathbf{b}_+$. Hence, there exists $\mathbf{x}_+ \in S_+$ with $\lim_{t\to\infty} D_F(\mathbf{x}_+, \tilde{\mathbf{x}}(t)) = 0$.

For any such $\mathbf{x}_+$, let us assume that $\tilde{\mathbf{x}}(t) \nrightarrow \mathbf{x}_+$. Then there exists $\varepsilon > 0$ and a sequence of times $t_0, t_1, \ldots$ with $\|\mathbf{x}_+ - \tilde{\mathbf{x}}(t_k)\|_2 \geq \varepsilon$ and $\lim_{k\to\infty} D_F(\mathbf{x}_+, \tilde{\mathbf{x}}(t_k)) = 0$. Since $\tilde{\mathbf{x}}(t_k)$ is a bounded sequence, a (not relabeled) subsequence converges to some $\bar{\mathbf{x}}$ with $\|\mathbf{x}_+ - \bar{\mathbf{x}}\|_2 \geq \varepsilon$ and $D_F(\mathbf{x}_+, \bar{\mathbf{x}}) = 0$. Since $D_F(\bar{\mathbf{x}}, \bar{\mathbf{x}}) = 0$ and D_F is non-negative, this is a contradiction to the strict convexity of $D_F(\cdot, \bar{\mathbf{x}})$. Hence, $\tilde{\mathbf{x}}_\infty = \lim_{t\to\infty} \tilde{\mathbf{x}}(t) \in S_+$ exists and is the unique solution satisfying $\lim_{t\to\infty} D_F(\tilde{\mathbf{x}}_\infty, \tilde{\mathbf{x}}(t)) = 0$.

Because $\partial_t D_F(\mathbf{x}_+, \tilde{\mathbf{x}}(t))$ is identical for all $\mathbf{x}_+ \in S_+$ (the second line of (19) does not depend on the choice of $\mathbf{x}_+$), the difference

$$\Delta_{\mathbf{x}_+} = D_F(\mathbf{x}_+, \tilde{\mathbf{x}}(0)) - D_F(\mathbf{x}_+, \tilde{\mathbf{x}}_\infty) \tag{20}$$

is also identical for all $\mathbf{x}_+ \in S_+$. By non-negativity of D_F,

$$D_F(\mathbf{x}_+, \tilde{\mathbf{x}}(0)) \geq \Delta_{\mathbf{x}_+} = \Delta_{\tilde{\mathbf{x}}_\infty} = D_F(\tilde{\mathbf{x}}_\infty, \tilde{\mathbf{x}}(0)). \tag{21}$$

Thus

$$\tilde{\mathbf{x}}_\infty \in \arg\min_{\mathbf{z}\in S_+} D_F(\mathbf{z}, \tilde{\mathbf{x}}(0)). \tag{22}$$

This complets the proof. □

Having Theorem C.5 at hand, it is straight-forward to derive Theorem C.1.

Proof of Theorem C.1. By Theorem C.5, the limit $\mathbf{x}_\infty := \lim_{t\to\infty} \mathbf{x}(t)$ exists and $\tilde{\mathbf{x}}_\infty = \mathbf{x}_\infty^{\odot L}$ lies in S_+. The quantitative bound in (10) can be deduced from (17) by following the proof of Theorem 4 by Chou et al. (2023a) since D_F based on F in (15) and D_F of Chou et al. (2023a) are the same, cf. our discussion below (15). □

D   RIEMANNIAN METRIC AND RIEMANNIAN GRADIENT

The reparametrized flow we consider in this work induces a Riemannian metric on the manifold $\mathbb{R}_{>0}$ and our method can be regarded as a Riemannian gradient approach. From this point of view, the underlying metric distorts the space and ensures that positive solutions are always obtained. Indeed, by definition of the flow in Theorem 2.1 we have that

$$\mathbf{x}'(t) = -\mathbf{x}^{\odot L-1} \odot [\mathbf{A}^\top (\mathbf{A}\mathbf{x}^{\odot L} - \mathbf{b})].$$

From this we can deduce the dynamics governing $\tilde{\mathbf{x}} = \mathbf{x}^{\odot L}$ via

$$\begin{aligned}
\tilde{\mathbf{x}}'(t) &= L\mathbf{x}^{\odot(L-1)} \odot \mathbf{x}'(t) = -L\mathbf{x}^{\odot(L-1)} \odot [\mathbf{x}^{\odot(L-1)} \odot [\mathbf{A}^\top (\mathbf{A}\mathbf{x}^{\odot L} - \mathbf{b})]] \\
&= -L\mathbf{x}^{\odot(2L-2)} \odot [\mathbf{A}^\top (\mathbf{A}\mathbf{x}^{\odot L} - \mathbf{b})] = -L\tilde{\mathbf{x}}^{\odot(2-\frac{2}{L})} \odot [\mathbf{A}^\top (\mathbf{A}\tilde{\mathbf{x}} - \mathbf{b})] \\
&= -L\tilde{\mathbf{x}}^{\odot(2-\frac{2}{L})} \odot \nabla \ell(\tilde{\mathbf{x}}) = -L\tilde{\mathbf{x}}(t)^{\odot q} \odot \nabla \ell(\tilde{\mathbf{x}}),
\end{aligned}$$

where we define $q = 2 - \frac{2}{L} \in [1, 2)$. The matrix

$$G(\tilde{\mathbf{x}}) = \mathrm{diag}(\tilde{\mathbf{x}})^{\frac{2}{L}-2} = \mathrm{diag}(\tilde{\mathbf{x}})^{-q},$$

defines the Riemannian metric $\langle \mathbf{u}, \mathbf{v} \rangle_{\tilde{\mathbf{x}}} = \langle \tilde{\mathbf{x}}^{\odot(-q)} \odot \mathbf{u}, \mathbf{v} \rangle$ on $\mathbb{R}_{>0}$ such that the dynamics of $\tilde{\mathbf{x}}'(t)$ can be recast as

$$\tilde{\mathbf{x}}'(t) = -LG(\tilde{\mathbf{x}})^{-1}\nabla \ell(\tilde{\mathbf{x}}) =: -L\widetilde{\nabla}\ell(\tilde{\mathbf{x}}), \tag{23}$$

where $\widetilde{\nabla}$ denotes the Riemannian gradient. For further details on Riemannian metrics and gradients, we refer the reader to Chapter 3 in the book of Boumal (2023).

E   PROOF OF THEOREM 2.4

We now analyze the dynamics given by

$$\boldsymbol{\xi}'(t) = -\frac{t}{2}\boldsymbol{\xi}(t)^{\odot q} \odot \nabla \ell(\tilde{\mathbf{x}}(t)),$$

$$\tilde{\mathbf{x}}'(t) = \frac{2}{t}(\boldsymbol{\xi}(t) - \tilde{\mathbf{x}}(t)),$$

$$\boldsymbol{\xi}(0) = \tilde{\mathbf{x}}(0) = \tilde{\mathbf{x}}_0 > \mathbf{0}.$$

Remark E.1. *Let us highlight to observations:*

(i) We observe that the sign of $\boldsymbol{\xi}(t)$ does not change. In fact, if $\xi(t_0)_i = 0$, for some $t_0 \geq 0$ and $i \in [N]$, then we have $\xi(t_0)_i = \xi'(t_0)_i = 0$. By the Picard-Lindelöf theorem, we know that $\xi(t)_i = 0$, for all $t \geq 0$, which contradicts $\xi(0)_i > 0$. Thus, the ODE admits a unique solution $\boldsymbol{\xi}(t) > \mathbf{0}$, for all $t \geq 0$.

(ii) The equivalent second order EDO is given by

$$\tilde{\mathbf{x}}''(t) + \frac{3}{t}\tilde{\mathbf{x}}'(t) = -\left(\tilde{\mathbf{x}}(t) + \frac{t}{2}\tilde{\mathbf{x}}'(t)\right)^{\odot q} \odot \nabla \ell(\tilde{\mathbf{x}}(t)).$$

Proof of Theorem 2.4. Consider $\mathbf{x}_+ \in S_+$. We then have that $\ell(\mathbf{x}_+) = \min_{\mathbf{z} \in \mathbb{R}_{\geq 0}^N} \ell(\mathbf{z}) = \frac{1}{2}\|\mathbf{b}_+ - \mathbf{b}\|_2^2$ where $\mathbf{b}_+$ has been defined in (5). Now, consider the Lyapunov functional given by

$$\mathcal{E}(t) = t^2(\ell(\tilde{\mathbf{x}}(t)) - \ell(\mathbf{x}_+)) + \frac{4}{2-q}D_F(\mathbf{x}_+, \boldsymbol{\xi}(t))$$

where the function $F(\mathbf{x}) = \langle \mathbf{x}, \ln_q(\mathbf{x}) \rangle$ has been defined in (15) and the Bregman divergence D_F has been defined in Definition C.2. We start by proving that $\mathcal{E}(t)$ decreases in t, i.e., $\mathcal{E}'(t) \leq 0$. First we will calculate the derivative $\frac{d}{dt}D_F(\mathbf{x}_+, \boldsymbol{\xi}(t))$. Remember from (13) and (14) that the gradient of F is given by $\nabla F(\boldsymbol{\xi}) = \ln_q(\boldsymbol{\xi}) + \boldsymbol{\xi}^{\odot 1-q}$ and the Hessian is described by $\nabla^2 F(\boldsymbol{\xi}) = (2-q)\text{diag}(\boldsymbol{\xi}^{\odot -q})$. Using this together with $\boldsymbol{\xi}'(t) = -\frac{t}{2}\boldsymbol{\xi}(t)^{\odot q} \odot \nabla \ell(\tilde{\mathbf{x}})$, we have that

$$\partial_t \nabla F(\boldsymbol{\xi}(t)) = \nabla^2 F(\boldsymbol{\xi}(t))\boldsymbol{\xi}'(t) = (2-q)\text{diag}(\boldsymbol{\xi}(t)^{\odot -q})\boldsymbol{\xi}'(t)$$
$$= -\frac{t}{2}(2-q)\nabla \ell(\tilde{\mathbf{x}}(t)),$$

which further yields that

$$\frac{d}{dt}D_F(\mathbf{x}_+, \boldsymbol{\xi}(t)) = \frac{d}{dt}\left(F(\mathbf{x}_+) - F(\boldsymbol{\xi}(t)) - \langle \nabla F(\boldsymbol{\xi}(t)), \mathbf{x}_+ - \boldsymbol{\xi}(t) \rangle\right)$$
$$= -\langle \nabla F(\boldsymbol{\xi}(t)), \boldsymbol{\xi}'(t) \rangle - \langle \nabla F(\boldsymbol{\xi}(t)), -\boldsymbol{\xi}'(t) \rangle - \langle \partial_t \nabla F(\boldsymbol{\xi}(t)), \mathbf{x}_+ - \boldsymbol{\xi}(t) \rangle$$
$$= -\langle \partial_t \nabla F(\boldsymbol{\xi}(t)), \mathbf{x}_+ - \boldsymbol{\xi}(t) \rangle$$
$$= \frac{t}{2}(2-q)\langle \nabla \ell(\tilde{\mathbf{x}}(t)), \mathbf{x}_+ - \boldsymbol{\xi}(t) \rangle. \tag{24}$$

Thus,

$$\mathcal{E}'(t) = 2t(\ell(\tilde{\mathbf{x}}(t)) - \ell(\mathbf{x}_+)) + t^2\langle \nabla \ell(\tilde{\mathbf{x}}), \tilde{\mathbf{x}}'(t) \rangle + \frac{4}{2-q}\frac{d}{dt}D_F(\mathbf{x}_+, \boldsymbol{\xi}(t)).$$
$$= 2t(\ell(\tilde{\mathbf{x}}(t)) - \ell(\mathbf{x}_+)) + t^2\langle \nabla \ell(\tilde{\mathbf{x}}), \tilde{\mathbf{x}}'(t) \rangle + 2t\langle \nabla \ell(\tilde{\mathbf{x}}(t)), \mathbf{x}_+ - \boldsymbol{\xi}(t) \rangle$$
$$= 2t(\ell(\tilde{\mathbf{x}}(t)) - \ell(\mathbf{x}_+)) + t^2\langle \nabla \ell(\tilde{\mathbf{x}}), \tilde{\mathbf{x}}'(t) \rangle + 2t\langle \nabla \ell(\tilde{\mathbf{x}}(t)), \mathbf{x}_+ - \tilde{\mathbf{x}}(t) - \frac{t}{2}\tilde{\mathbf{x}}'(t) \rangle$$
$$= 2t(\ell(\tilde{\mathbf{x}}(t)) - \ell(\mathbf{x}_+)) + 2t\langle \nabla \ell(\tilde{\mathbf{x}}(t)), \mathbf{x}_+ - \tilde{\mathbf{x}}(t) \rangle$$
$$= -2tD_\ell(\mathbf{x}_+, \tilde{\mathbf{x}}(t)) \leq 0.$$

We can finally establish the accelerated convergence rate for the dynamics given by (ACC). Indeed, since $\mathcal{E}(t)$ is decreasing, we conclude that

$$t^2(\ell(\tilde{\mathbf{x}}(t)) - \ell(\mathbf{x}_+)) \leq \mathcal{E}(t) \leq \mathcal{E}(0) = \frac{4}{2-q}D_F(\mathbf{x}_+, \tilde{\mathbf{x}}_0)$$

which implies that

$$0 \leq \ell(\tilde{\mathbf{x}}(t)) - \ell(\mathbf{x}_+) \leq \frac{4D_F(\mathbf{x}_+, \tilde{\mathbf{x}}_0)}{(2-q)t^2}.$$

$\square$

F   DISCRETE ANALYSIS: PROOF OF THEOREM 2.6

Let us now establish that gradient descent on (3) converges to the solution of NNLS by proving Theorem 2.6. Let us abbreviate $M = \|\mathbf{A}^T\mathbf{A}\|$, let us fix $\mathbf{x}_0 > \mathbf{0}$, and consider GD on the product $\tilde{\mathbf{x}} = \mathbf{x}^{\odot L}$ defined via

$$\tilde{\mathbf{x}}_{k+1} = \tilde{\mathbf{x}}_k - \eta_k \tilde{\mathbf{x}}_k^{\odot q} \odot \mathbf{A}^T(\mathbf{A}\tilde{\mathbf{x}}_k - \mathbf{b}), \tag{25}$$

with $k \geq 0$, where $\eta_k > 0$ is a decreasing sequence. We start by providing conditions under which $\ell(\tilde{\mathbf{x}}_k)$ forms a decreasing sequence. Recall that

$$\widetilde{\nabla}\ell(\tilde{\mathbf{x}}) = \tilde{\mathbf{x}}^{\odot q} \odot \nabla\ell(\tilde{\mathbf{x}}) = \tilde{\mathbf{x}}^{\odot q} \odot \mathbf{A}^T(\mathbf{A}\tilde{\mathbf{x}} - \mathbf{b})$$

is the Riemannian gradient vector, with respect to Riemannian metric as described in Section D. Thus, the reparametrized GD in (25) reduces to the Riemannian gradient descent algorithm

$$\tilde{\mathbf{x}}_{k+1} = \tilde{\mathbf{x}}_k - \eta_k \widetilde{\nabla}\ell(\tilde{\mathbf{x}}_k) \tag{26}$$

without retraction step.

Remark F.1. *It is noteworthy that in (26) we work with a GD algorithm defined on the product representation* $\tilde{\mathbf{x}}$. *Wouldn't it be more natural to analyze GD on* $\mathbf{x}$ *instead? Fixing* $\mathbf{x}_0 > \mathbf{0}$ *this would mean to define*

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \eta_k \nabla\mathcal{L}(\mathbf{x}_k),$$

where $\eta_k > 0$ *is a decreasing sequence. In contrast to the continuous case, estimating a convergence order for this algorithm is difficult, given the non-convex nature of* $\mathcal{L}(\mathbf{x})$. *Moreover, the Hessian* $\nabla^2\mathcal{L}(\mathbf{x})$ *does not have a uniform bound given its nonlinear dependency on* $\mathbf{x}$. *Taking the L-th Hadamard power on both sides leads to*

$$\tilde{\mathbf{x}}_{k+1} = (\mathbf{x}_k - \eta_k\nabla\mathcal{L}(\mathbf{x}_k))^{\odot L}$$

which can be linearized by a first-order Taylor approximation of $f(\mathbf{h}) = (\mathbf{x}_k + \mathbf{h})^{\odot L}$ *at* $\mathbf{h} = \mathbf{0}$ *to yield*

$$\begin{aligned}
\tilde{\mathbf{x}}_{k+1} &\approx \tilde{\mathbf{x}}_k - L \cdot \mathbf{x}_k^{\odot(L-1)} \cdot \eta_k\nabla\mathcal{L}(\mathbf{x}_k) \\
&= \tilde{\mathbf{x}}_k - L \cdot \mathbf{x}_k^{\odot(L-1)} \cdot \eta_k \cdot \frac{1}{L}\mathbf{x}_k^{\odot(L-1)} \cdot \nabla\ell(\tilde{\mathbf{x}}_k) \\
&= \tilde{\mathbf{x}}_k - \eta_k\tilde{\mathbf{x}}_k^q \cdot \nabla\ell(\tilde{\mathbf{x}}_k) \\
&= \tilde{\mathbf{x}}_k - \eta_k\widetilde{\nabla}\ell(\tilde{\mathbf{x}}_k).
\end{aligned}$$

Our method in (26) can thus be interpreted as a linearization of the naive GD routine for $\mathcal{L}$. *Importantly, both algorithms follow the same continuous flow* $\mathbf{x}'(t) = -\nabla\mathcal{L}(\mathbf{x}(t))$.

Lemma F.2. *Consider GD as defined in (26). Let* $C \geq 0$, *let* $\eta_{k-1} \leq \varepsilon := \min\{\frac{1}{2\|\mathbf{A}^T\mathbf{b}\|_2 C^{q-1}}, \frac{1}{2MC^q}\}$, *and let the* $(k-1)$-*th iterate* $\tilde{\mathbf{x}}_{k-1} \in \mathbb{R}_{>0}^N$ *satisfy* $\|\tilde{\mathbf{x}}_{k-1}\|_2 \leq C$. *Then,* $\tilde{\mathbf{x}}_k \in \mathbb{R}_{>0}^N$ *and* $\ell(\tilde{\mathbf{x}}_k) \leq \ell(\tilde{\mathbf{x}}_{k-1})$.

Proof. To see the first claim, note that

$$\tilde{\mathbf{x}}_k = \tilde{\mathbf{x}}_{k-1} - \eta_{k-1}\tilde{\mathbf{x}}_{k-1}^{\odot q} \odot \mathbf{A}^T(\mathbf{A}\tilde{\mathbf{x}}_{k-1} - \mathbf{b})$$
$$= \tilde{\mathbf{x}}_{k-1} \odot \left(1 - \eta_{k-1}\tilde{\mathbf{x}}_{k-1}^{\odot q-1} \odot \mathbf{A}^T(\mathbf{A}\tilde{\mathbf{x}}_{k-1} - \mathbf{b})\right)$$
$$> 0$$

by the choice of ε and the assumption that $\|\tilde{\mathbf{x}}_{k-1}\|_2 \le C$.

Let us now turn to the second claim. Since the Hessian matrix of ℓ satisfies $\nabla^2\ell(\mathbf{x}) = \mathbf{A}^T\mathbf{A}$, Taylor's formula yields

$$\ell(\mathbf{x} + \mathbf{d}) \le \ell(\mathbf{x}) + \langle\nabla\ell(\mathbf{x}), \mathbf{d}\rangle + \frac{M\|\mathbf{d}\|^2}{2},$$

for all $\mathbf{x}, \mathbf{d} \in \mathbb{R}^N$. Applying this to $\mathbf{x} = \tilde{\mathbf{x}}_{k-1}$ and $\mathbf{d} = -\eta_{k-1}\widetilde{\nabla}\ell(\tilde{\mathbf{x}}_{k-1}) = -\eta_{k-1}\tilde{\mathbf{x}}_{k-1}^{\odot q} \odot \nabla\ell(\tilde{\mathbf{x}}_{k-1})$, we obtain

$$\ell(\tilde{\mathbf{x}}_k) \le \ell(\tilde{\mathbf{x}}_{k-1}) - \eta_{k-1}\langle\nabla\ell(\tilde{\mathbf{x}}_{k-1}), \widetilde{\nabla}\ell(\tilde{\mathbf{x}}_{k-1})\rangle + \frac{M\eta_{k-1}^2}{2}\|\widetilde{\nabla}\ell(\tilde{\mathbf{x}}_{k-1})\|^2$$
$$= \ell(\tilde{\mathbf{x}}_{k-1}) - \eta_{k-1}\left\langle\widetilde{\nabla}\ell(\tilde{\mathbf{x}}_{k-1}), \left(1 - \frac{M\eta_{k-1}}{2}\tilde{\mathbf{x}}_{k-1}^{\odot q}\right) \odot \nabla\ell(\tilde{\mathbf{x}}_{k-1})\right\rangle$$
$$= \ell(\tilde{\mathbf{x}}_{k-1}) - \eta_{k-1}\left\langle\nabla\ell(\tilde{\mathbf{x}}_{k-1}), \tilde{\mathbf{x}}_{k-1}^{\odot q} \odot \left(1 - \frac{M\eta_{k-1}}{2}\tilde{\mathbf{x}}_{k-1}^{\odot q}\right) \odot \nabla\ell(\tilde{\mathbf{x}}_{k-1})\right\rangle.$$

By assumption we have that $\tilde{\mathbf{x}}_{k-1}^{\odot q} > 0$ and $(1 - \frac{M}{2}\eta_{k-1}\tilde{\mathbf{x}}_{k-1}^{\odot q}) \ge 0$ which implies that $\ell(\tilde{\mathbf{x}}_k) \le \ell(\tilde{\mathbf{x}}_{k-1})$ and thus the claim. $\qquad\square$

Second, we show that the *Bregman divergence balls* derived from the Tsallis q-logarithm F defined in (15) are bounded.

Lemma F.3. *Let $F(\mathbf{z}) = \langle\mathbf{z}, \ln_q(\mathbf{z})\rangle$ as in (15), for $q \in [1, 2)$. Then, for any $C \ge 0$ and $\mathbf{x} \in \mathbb{R}_{\ge 0}^N$, the set*

$$B_{D_F, C, \mathbf{x}} := \{\mathbf{z} \in \mathbb{R}_{\ge 0}^N : D_F(\mathbf{x}, \mathbf{z}) \le C\}$$

is bounded. In particular, $B_{D_F, C, \mathbf{x}} \subset RB_\infty$ where B_∞ is the ℓ_∞ unit ball and

$$R = \begin{cases} 2\max(\|\mathbf{x}\|_\infty, C + \ln(2)\|\mathbf{x}\|_\infty) & \text{if } q = 1 \\ \left(C + \frac{1}{q-1}\|\mathbf{x}\|_\infty^{2-q}\right)^{\frac{1}{2-q}} & \text{if } q \in (1, 2). \end{cases} \tag{27}$$

Proof. To show the claim, we argue that unbounded sequences are not contained in $B_{D_F, C, \mathbf{x}}$. First note that

$$D_F(\mathbf{x}, \mathbf{z}) = \sum_{i=1}^N D_f(x_i, z_i),$$

where $f(z) = z\ln_q(z)$ is a univariate marginal of F and $D_f(x, z) \ge 0$ by convexity of f.

Let now $\mathbf{z}_k \subset \mathbb{R}_{\ge 0}^N$ be a sequence with $\|\mathbf{z}_k\|_\infty \to \infty$. Then, there exists $i \in [N]$ and a sub-sequence k_j such that $\lim_{j\to\infty}(z_{k_j})_i = \infty$. Hence, we obtain that

$$D_F(\mathbf{x}, \mathbf{z}_k) \ge D_f(x_i, (z_k)_i)$$
$$= x_i\ln_q(x_i) - (z_k)_i\ln_q((z_k)_i) - (\ln_q((z_k)_i) + (z_k)_i^{1-q})(x_i - (z_k)_i)$$
$$= x_i\ln_q(x_i) - x_i(\ln_q((z_k)_i) + (z_k)_i^{1-q}) + (z_k)_i^{2-q}$$
$$= x_i\ln_q(x_i) + (z_k)_i^{2-q}\left(1 - \frac{x_i}{(z_k)_i} - \frac{x_i\ln_q((z_k)_i)}{(z_k)_i^{2-q}}\right)$$
$$\overset{k\to\infty}{\to} \infty$$

by assumption on q. We now derive the non-asymptotic bound. Since $D_F(\mathbf{x}, \mathbf{z}) = \sum_{i=1}^{N} D_f(x_i, z_i)$ and $D_f \geq 0$, the i-th entry z_i (and with it $\|\mathbf{z}\|_\infty$) can be chosen largest if $z_j = x_j$, for all $j \neq i$. Thus, we simply need to control

$$\max_{z \geq 0} z \quad \text{subject to} \quad D_f(x, z) \leq C.$$

To get the bound, we separate the case $q = 1$ and $q \neq 1$. When $q = 1$, we have

$$D_f(x, z) = x \ln(x) - x + z - \ln(z)x.$$

Taking into consideration the constraint, we obtain

$$z - \ln(z)x \leq C + x - x \ln(x). \tag{28}$$

Let $f(z) := z - \ln(z)x$. Note that f is convex with minimum at $z = x$. Furthermore, $f'(z) \geq \frac{1}{2}$ for all $z \geq 2x$. Therefore

$$f(z) \geq \begin{cases} f(x) & \text{if } z < 2x \\ f(2x) + \frac{1}{2}(z - 2x) & \text{if } z \geq 2x \end{cases}$$

$$\geq \begin{cases} x - \ln(x)x & \text{if } z < 2x \\ x - \ln(2x)x + \frac{1}{2}z & \text{if } z \geq 2x. \end{cases}$$

Since $z, x \geq 0$, we have that if $z \geq 2x$ then (28) implies

$$x - \ln(2x)x + \frac{1}{2}z \leq C + x - x \ln(x),$$

or equivalently, $z \leq 2(C + \ln(2)x)$. Applying a case distinction between $z \geq 2x$ and $z < 2x$, we obtain that

$$\{\mathbf{z} \in \mathbb{R}_{\geq 0}^N : D_F(\mathbf{x}, \mathbf{z}) \leq C\} \subset \{\mathbf{z} \in \mathbb{R}_{\geq 0}^N : z_i \leq 2 \max(x_i, C + \ln(2)x_i)\} \subset RB_\infty.$$

When $q \neq 1$, we have

$$D_f(x, z) = x \ln_q(x) + \frac{x}{1-q} - \frac{2-q}{1-q} x z^{1-q} + z^{2-q}$$

$$= \frac{x^{2-q}}{1-q} - \frac{2-q}{1-q} x z^{1-q} + z^{2-q}.$$

Taking into consideration the constraint $C \geq D_f(x, z)$, we obtain

$$z^{2-q} + \frac{2-q}{q-1} x z^{1-q} \leq C + \frac{x^{2-q}}{q-1}.$$

Since $q \in (1, 2)$, we know that $\frac{2-q}{q-1} x \geq 0$. Hence we have

$$z^{1-q} \leq \frac{(q-1)C + x^{2-q}}{(2-q)x}$$

$$z^{2-q} \leq C + \frac{x^{2-q}}{q-1}.$$

and consequently

$$\{\mathbf{z} \in \mathbb{R}_{\geq 0}^N : D_F(\mathbf{x}, \mathbf{z}) \leq C\} \subset \left\{\mathbf{z} \in \mathbb{R}_{\geq 0}^N : \left(\frac{(2-q)x}{(q-1)C + x^{2-q}}\right)^{\frac{1}{q-1}} \leq z_i \leq \left(C + \frac{x^{2-q}}{q-1}\right)^{\frac{1}{2-q}}\right\} \subset RB_\infty.$$

This completes the proof. $\qquad\square$

We have now all tools at hand to prove Theorem 2.6. Since the proof strategy is a discrete analog of the proof of Theorem 2.4, it is convenient to read the latter before turning to the proof below.

Proof of Theorem 2.6. For $\mathbf{x}_+ \in S_+$, consider the energy function

$$\mathcal{E}_k = \eta_k k(\ell(\tilde{\mathbf{x}}_k) - \ell(\mathbf{x}_+)) + \frac{1}{2-q} D_F(\mathbf{x}_+, \tilde{\mathbf{x}}_k)$$

with η_k as defined in the proposition statement and let $\mathfrak{C} \geq 1$ be sufficiently large such that

$$\{\mathbf{z} \in \mathbb{R}_{\geq 0}^N \colon D_F(\mathbf{x}_+, \mathbf{z}) \leq D_F(\mathbf{x}_+, \tilde{\mathbf{x}}_0) + c_{\max}\} \cup \{\mathbf{x}_+\} \subset \mathfrak{C}B_2, \qquad (29)$$

where B_2 is the ℓ_2 unit ball. By Lemma F.3 such a $\mathfrak{C} \leq D_F(\mathbf{x}_+, \tilde{\mathbf{x}}_0) + c_{\max} + \|\mathbf{x}_+\|_\infty$ always exists and only depends on $c_{\max}$ and the distance between $\mathbf{x}_+$, $\tilde{\mathbf{x}}_0$, and the origin. Clearly, $\tilde{\mathbf{x}}_0 \in \mathfrak{C}B_2 \cap \mathbb{R}_{>0}^N$. Recall that $\eta_k = \frac{c}{(k+2)^\gamma}$, for $c > 0$ having been defined in the theorem statement and depending on $\mathfrak{C}$.

We first prove by induction that $\tilde{\mathbf{x}}_k \in \mathfrak{C}B_2 \cap \mathbb{R}_{>0}^N$ and $\mathcal{E}_k \leq \mathcal{E}_{k-1} + c\left(\frac{(2-q)^{-1}}{(k+1)^{2\gamma}} - \frac{\delta}{2(k+1)^\gamma}\right)$, for any $0 \leq k \leq K$, where we set $\mathcal{E}_{-1} = \infty$. For $k = 0$, the claim is trivially fulfilled.

Assume now the claim holds for $(k-1) \in [0, K)$. Then $\tilde{\mathbf{x}}_{k-1} \in \mathbb{R}_{>0}^N$ and $\|\tilde{\mathbf{x}}_{k-1}\|_2 \leq \mathfrak{C}$, which gives for any $i \in [N]$ that

$$\begin{aligned}
(\tilde{x}_k)_i &= (\tilde{x}_{k-1})_i - [\eta_{k-1}\widetilde{\nabla}\ell(\tilde{\mathbf{x}}_{k-1})]_i \\
&= (\tilde{x}_{k-1})_i \cdot \left(1 - \eta_{k-1}(\tilde{x}_{k-1})_i^{q-1}[\mathbf{A}^T(\mathbf{A}\tilde{\mathbf{x}}_{k-1} - \mathbf{b})]_i\right) \\
&\geq \frac{1}{2}(\tilde{x}_{k-1})_i,
\end{aligned} \qquad (30)$$

where we used that, by definition of c in η_{k-1},

$$\eta_{k-1}(\tilde{x}_{k-1})_i^{q-1}[\mathbf{A}^T(\mathbf{A}\tilde{\mathbf{x}}_{k-1} - \mathbf{b})]_i \leq \eta_{k-1}(\mathfrak{C}^q M + \mathfrak{C}^{q-1}\|\mathbf{A}^T\mathbf{b}\|_2) \leq \frac{1}{2}.$$

Consequently, $\tilde{\mathbf{x}}_k \in \mathbb{R}_{>0}^N$ as well. Let us now observe that, by (13) and (14), gradient and Hessian of $\tilde{\mathbf{x}} \mapsto D_F(\mathbf{x}_+, \cdot)|_{\tilde{\mathbf{x}}}$ are given by

$$\begin{aligned}
\nabla D_F(\mathbf{x}_+, \cdot)|_{\tilde{\mathbf{x}}} &= \nabla_{\tilde{\mathbf{x}}}\left(F(\mathbf{x}_+) - F(\tilde{\mathbf{x}}) - \langle\nabla F(\tilde{\mathbf{x}}), \mathbf{x}_+ - \tilde{\mathbf{x}}\rangle\right) \\
&= -\nabla F(\tilde{\mathbf{x}}) - \nabla^2 F(\tilde{\mathbf{x}})(\mathbf{x}_+ - \tilde{\mathbf{x}}) + \nabla F(\tilde{\mathbf{x}}) \\
&= -\nabla^2 F(\tilde{\mathbf{x}})(\mathbf{x}_+ - \tilde{\mathbf{x}}) = -(2-q) \cdot \mathrm{diag}(\tilde{\mathbf{x}}^{\odot -q})(\mathbf{x}_+ - \tilde{\mathbf{x}}) \\
&= -(2-q) \cdot \tilde{\mathbf{x}}^{\odot -q} \odot (\mathbf{x}_+ - \tilde{\mathbf{x}})
\end{aligned}$$

and

$$\nabla^2 D_F(\mathbf{x}_+, \cdot)|_{\tilde{\mathbf{x}}} = -(2-q) \cdot \mathrm{diag}\left((q-1) \cdot \tilde{\mathbf{x}}^{\odot -q} - q \cdot \mathbf{x}_+ \odot \tilde{\mathbf{x}}^{\odot -(q+1)}\right).$$

We thus can approximate $D_F(\mathbf{x}_+, \tilde{\mathbf{x}}_k)$ by the second-order Taylor approximation of $\tilde{\mathbf{x}} \mapsto D_F(\mathbf{x}_+, \cdot)|_{\tilde{\mathbf{x}}}$ around $\tilde{\mathbf{x}}_{k-1}$

$$\begin{aligned}
&D_F(\mathbf{x}_+, \tilde{\mathbf{x}}_k) - D_F(\mathbf{x}_+, \tilde{\mathbf{x}}_{k-1}) \\
&\leq -\eta_{k-1}\langle\nabla D_F(\mathbf{x}_+, \cdot)|_{\tilde{\mathbf{x}}_{k-1}}, \widetilde{\nabla}\ell(\tilde{\mathbf{x}}_{k-1})\rangle + \sup_{\tilde{\mathbf{x}} \in [\tilde{\mathbf{x}}_{k-1}, \tilde{\mathbf{x}}_k]} \eta_{k-1}^2\left|\widetilde{\nabla}\ell(\tilde{\mathbf{x}}_{k-1})^T \cdot \nabla^2 D_F(\mathbf{x}_+, \cdot)|_{\tilde{\mathbf{x}}} \cdot \widetilde{\nabla}\ell(\tilde{\mathbf{x}}_{k-1})\right| \\
&\leq (2-q)\eta_{k-1}\langle\mathbf{x}_+ - \tilde{\mathbf{x}}_{k-1}, \nabla\ell(\tilde{\mathbf{x}}_{k-1})\rangle + \frac{c}{(k+1)^{2\gamma}},
\end{aligned} \qquad (31)$$

where $[\tilde{\mathbf{x}}_{k-1}, \tilde{\mathbf{x}}_k]$ denotes the connecting line between $\tilde{\mathbf{x}}_{k-1}$ and $\tilde{\mathbf{x}}_k$, and we used the following two observations: first,

$$\begin{aligned}
\langle\nabla D_F(\mathbf{x}_+, \cdot)|_{\tilde{\mathbf{x}}_{k-1}}, \widetilde{\nabla}\ell(\tilde{\mathbf{x}}_{k-1})\rangle &= -(2-q)\langle\tilde{\mathbf{x}}_{k-1}^{\odot -q} \odot (\mathbf{x}_+ - \tilde{\mathbf{x}}_{k-1}), \tilde{\mathbf{x}}_{k-1}^{\odot q}\nabla\ell(\tilde{\mathbf{x}}_{k-1})\rangle \\
&= -(2-q)\langle\mathbf{x}_+ - \tilde{\mathbf{x}}_{k-1}, \nabla\ell(\tilde{\mathbf{x}}_{k-1})\rangle;
\end{aligned} \qquad (32)$$

and second,

$$\sup_{\tilde{\mathbf{x}} \in [\tilde{\mathbf{x}}_{k-1}, \tilde{\mathbf{x}}_k]} \eta_{k-1}^2 \left| \widetilde{\nabla} \ell(\tilde{\mathbf{x}}_{k-1})^T \cdot \nabla^2 D_F(\mathbf{x}_+, \cdot)|_{\tilde{\mathbf{x}}} \cdot \widetilde{\nabla} \ell(\tilde{\mathbf{x}}_{k-1}) \right|$$

$$\leq \sup_{\tilde{\mathbf{x}} \in [\tilde{\mathbf{x}}_{k-1}, \tilde{\mathbf{x}}_k]} \eta_{k-1}^2 \sum_{i=1}^N \left| [\nabla^2 D_F(\mathbf{x}_+, \cdot)|_{\tilde{\mathbf{x}}}]_{i,i} \right| \cdot [\widetilde{\nabla} \ell(\tilde{\mathbf{x}}_{k-1})]_i^2$$

$$\leq \sup_{\tilde{\mathbf{x}} \in [\tilde{\mathbf{x}}_{k-1}, \tilde{\mathbf{x}}_k]} \eta_{k-1}^2 \sum_{i=1}^N (2-q) \cdot \left(|(q-1) \cdot \tilde{x}_i^{-q}| + |q \cdot (x_+)_i \cdot \tilde{x}_i^{-(q+1)}| \right) \cdot (\tilde{x}_{k-1})_i^{2q} \cdot [\mathbf{A}^T(\mathbf{A}\tilde{\mathbf{x}}_{k-1} - \mathbf{b})]_i^2$$

$$\leq 30 \cdot \mathfrak{C}^q \cdot \eta_{k-1}^2 \cdot \|\mathbf{A}^T(\mathbf{A}\tilde{\mathbf{x}}_{k-1} - \mathbf{b})\|_2^2 \leq \eta_{k-1}^2 \cdot 60 \cdot \mathfrak{C}^q \cdot (\mathfrak{C}^{2q} M^2 + \|\mathbf{A}^T\mathbf{b}\|_2^2)$$

$$\leq \frac{c}{(k+1)^{2\gamma}},$$

where we used in the third from last step that $\tilde{x}_i \geq \frac{1}{2}(\tilde{x}_{k-1})_i$ for any $\tilde{\mathbf{x}} \in [\tilde{\mathbf{x}}_{k-1}, \tilde{\mathbf{x}}_k]$ by a similar calculation as in (30), that $(2-q)(q-1)$ and $(2-q)q$ are both upper bounded by $\frac{9}{4}$, and that $\|\tilde{\mathbf{x}}_{k-1}\|_2, \|\mathbf{x}_+\|_\infty \leq \mathfrak{C}$ by our induction hypothesis and (29). The ultimate inequality then just follows from recalling the definition of c in η_{k-1}. Since by $\eta_k \leq \eta_{k-1}$ and Lemma F.2

$$\mathcal{E}_k - \mathcal{E}_{k-1} = \eta_k k(\ell(\tilde{\mathbf{x}}_k) - \ell(\mathbf{x}_+)) - \eta_{k-1}(k-1)(\ell(\tilde{\mathbf{x}}_{k-1}) - \ell(\mathbf{x}_+)) + \frac{1}{2-q}\Big(D_F(\mathbf{x}_+, \tilde{\mathbf{x}}_k) - D_F(\mathbf{x}_+, \tilde{\mathbf{x}}_{k-1})\Big)$$

$$\leq \eta_k(\ell(\tilde{\mathbf{x}}_k) - \ell(\mathbf{x}_+)) + \eta_{k-1}(k-1)(\ell(\tilde{\mathbf{x}}_k) - \ell(\tilde{\mathbf{x}}_{k-1})) + \frac{1}{2-q}\Big(D_F(\mathbf{x}_+, \tilde{\mathbf{x}}_k) - D_F(\mathbf{x}_+, \tilde{\mathbf{x}}_{k-1})\Big)$$

$$\leq \eta_k(\ell(\tilde{\mathbf{x}}_k) - \ell(\mathbf{x}_+)) + \frac{1}{2-q}\Big(D_F(\mathbf{x}_+, \tilde{\mathbf{x}}_k) - D_F(\mathbf{x}_+, \tilde{\mathbf{x}}_{k-1})\Big)$$

$$\leq \eta_{k-1}(\ell(\tilde{\mathbf{x}}_{k-1}) - \ell(\mathbf{x}_+)) + \frac{1}{2-q}\Big(D_F(\mathbf{x}_+, \tilde{\mathbf{x}}_k) - D_F(\mathbf{x}_+, \tilde{\mathbf{x}}_{k-1})\Big),$$

we obtain from (31) that

$$\mathcal{E}_k \leq \mathcal{E}_{k-1} + \eta_{k-1}\big(\ell(\tilde{\mathbf{x}}_{k-1}) - \ell(\mathbf{x}_+)\big) + \eta_{k-1}\langle \mathbf{x}_+ - \tilde{\mathbf{x}}_{k-1}, \nabla \ell(\tilde{\mathbf{x}}_{k-1})\rangle + \frac{c(2-q)^{-1}}{(k+1)^{2\gamma}}$$

$$= \mathcal{E}_{k-1} - \eta_{k-1} D_\ell(\mathbf{x}_+, \tilde{\mathbf{x}}_{k-1}) + \frac{c(2-q)^{-1}}{(k+1)^{2\gamma}}$$

$$= \mathcal{E}_{k-1} - \frac{\eta_{k-1}}{2}\|\mathbf{A}(\mathbf{x}_+ - \tilde{\mathbf{x}}_{k-1})\|_2^2 + \frac{c(2-q)^{-1}}{(k+1)^{2\gamma}}$$

$$\leq \mathcal{E}_{k-1} + c\left(\frac{(2-q)^{-1}}{(k+1)^{2\gamma}} - \frac{\delta}{2(k+1)^{\gamma}}\right),$$

where we calculated the explicit form of D_ℓ in the penultimate step and used that by assumption

$$\|\mathbf{A}(\mathbf{x}_+ - \tilde{\mathbf{x}}_{k-1})\|_2^2 \geq \Big(\|\mathbf{A}\tilde{\mathbf{x}}_{k-1} - \mathbf{b}\|_2 - \|\mathbf{A}\mathbf{x}_+ - \mathbf{b}\|_2\Big)^2 \geq \delta,$$

for $k-1 < K$. To conclude the induction step, we need to argue that $\|\tilde{\mathbf{x}}_k\|_2 \leq \mathfrak{C}$. To this end, note that

$$D_F(\mathbf{x}_+, \tilde{\mathbf{x}}_k) \leq (2-q)\mathcal{E}_k \leq (2-q)\mathcal{E}_0 + c\sum_{j=1}^k \left(\frac{1}{(j+1)^{2\gamma}} - \frac{(2-q)\delta}{2(j+1)^{\gamma}}\right)$$

$$\leq D_F(\mathbf{x}_+, \tilde{\mathbf{x}}_0) + c_{\max},$$

(33)

where we used Lemma F.4 and $c < 1$ in the last step. Clearly, this implies $\|\tilde{\mathbf{x}}_k\|_2 \leq \mathfrak{C}$ by (29).

Using once more the recursive bound of $\mathcal{E}_k$ appearing in (33), we estimate for any $k \leq K$ that

$$\eta_k k(\ell(\tilde{\mathbf{x}}_k) - \ell(\mathbf{x}_+)) \leq \mathcal{E}_k \leq \frac{1}{2-q}(D_F(\mathbf{x}_+, \tilde{\mathbf{x}}_0) + c_{\max}).$$

Reorganizing the estimate and using that $(k+2)^\gamma / k \leq 2k^{1-\gamma}$, for $k \geq 2$, yields the claim. $\qquad \square$

To conclude the proof of Theorem 2.6, we finally provide the estimate of the sum in (33).

Lemma F.4. *Let $\gamma \in (0,1)$, $q \in [1,2)$, and $\delta > 0$. Then,*

$$\sum_{j=1}^{k} \left(\frac{1}{(j+1)^{2\gamma}} - \frac{(2-q)\delta}{2(j+1)^{\gamma}} \right) \leq \begin{cases} \zeta(2\gamma) & \text{for } \gamma > \frac{1}{2} \\ 2\left(\log\left(\frac{2}{(2-q)\delta}\right) + (2-q)\gamma \right) & \text{for } \gamma = \frac{1}{2} \\ \frac{1}{1-2\gamma}\left(\frac{2}{(2-q)\delta}\right)^{\frac{1}{\gamma}-2} + 2(2-q)\delta & \text{for } \gamma < \frac{1}{2} \end{cases}, \tag{34}$$

where ζ denotes the Riemann zeta function.

Proof. For $\gamma > \frac{1}{2}$, we have that

$$\sum_{j=1}^{k} \left(\frac{1}{(j+1)^{2\gamma}} - \frac{(2-q)\delta}{2(j+1)^{\gamma}} \right) \leq \sum_{j=1}^{k} \frac{1}{(j+1)^{2\gamma}} \leq \zeta(2\gamma).$$

For $\gamma \leq \frac{1}{2}$, observe that

$$\sum_{j=1}^{k} \frac{1}{(j+1)^{2\gamma}} = \sum_{j=2}^{k+1} \int_{j-1}^{j} \frac{1}{j^{2\gamma}} \, d\tau \leq \sum_{j=2}^{k+1} \int_{j-1}^{j} \frac{1}{\tau^{2\gamma}} \, d\tau = \int_{1}^{k+1} \frac{1}{\tau^{2\gamma}} \, d\tau \leq h(k+1),$$

where $h(k+1) = \log(k+1)$, for $\gamma = \frac{1}{2}$, and $h(k+1) = (1-2\gamma)^{-1}(k+1)^{1-2\gamma}$ else. Similarly,

$$\sum_{j=1}^{k} \frac{1}{(j+1)^{\gamma}} = \sum_{j=2}^{k+1} \int_{j}^{j+1} \frac{1}{j^{\gamma}} \, d\tau \geq \int_{2}^{k+2} \frac{1}{\tau^{\gamma}} \, d\tau = (1-\gamma)^{-1}\left((k+2)^{1-\gamma} - 2^{1-\gamma}\right).$$

Thus, the sum in (34) can be bounded by the maximal value of

$$f\colon [0,\infty) \to \mathbb{R}, \qquad f(t) := h(t+1) - \frac{(2-q)\delta}{2}(1-\gamma)^{-1}\left((t+2)^{1-\gamma} - 2^{1-\gamma}\right). \tag{35}$$

Since

$$f(t) \leq g(t) := h(t+2) - \frac{(2-q)\delta}{2}(1-\gamma)^{-1}\left((t+2)^{1-\gamma} - 2^{1-\gamma}\right),$$

one has that $\max_{t\geq 0} f(t) \leq \max_{t\geq 0} g(t)$, which can be determined by basic calculus to be

$$\begin{aligned}
\max_{t\geq 0} g(t) &= \begin{cases} 2\log\left(\frac{2}{(2-q)\delta}\right) - 2 + \sqrt{2}(2-q)\delta & \text{for } \gamma = \frac{1}{2} \\ \frac{1}{1-2\gamma}\left(\frac{2}{(2-q)\delta}\right)^{\frac{1}{\gamma}-2} - \frac{(2-q)\delta}{2(1-\gamma)}\left(\frac{2}{(2-q)\delta}\right)^{\frac{1}{\gamma}-1} + \frac{2^{1-\gamma}(2-q)\delta}{2(1-\gamma)} & \text{for } \gamma < \frac{1}{2} \end{cases} \\
&\leq \begin{cases} 2\left(\log\left(\frac{2}{(2-q)\delta}\right) + (2-q)\delta \right) & \text{for } \gamma = \frac{1}{2} \\ \frac{1}{1-2\gamma}\left(\frac{2}{(2-q)\delta}\right)^{\frac{1}{\gamma}-2} + 2(2-q)\delta & \text{for } \gamma < \frac{1}{2} \end{cases}
\end{aligned}$$

This completes the proof. $\qquad\square$

G   APPLICATION TO SPARSE RECOVERY

Apart from applications that seek non-negative physical quantities, see Section 1, NNLS is also an attractive method for compressive sensing if the sparse ground truth is non-negative. Indeed, under certain assumptions on the measurement operator $\mathbf{A}$, solving (NNLS) promotes the sparsity of its solution for free (no hyper-parameter tuning) and comes with improved robustness. This observation dates back to the seminal works of Donoho and Tanner (2005, 2010), which connect the topic to the theory of convex polytopes. (Interestingly, even before the modern theory of sparse recovery emerged in the form of compressive sensing, Donoho et al. (1992) exploited non-negativity to recover sparse objects.) Consequent works studied the uniqueness of positive solutions of

underdetermined systems (Bruckstein et al., 2008) and the generalization to low-rank solutions on the cone of positive-definite matrices (Wang et al., 2010). Other works established conditions under which NNLS would succeed to retrieve a sparse vector even in a noisy setting (Kueng and Jung, 2017; Shadmi et al., 2019; Slawski and Hein, 2011, 2013; Meinshausen, 2013). Two key concepts for such results are the *null space property* (Foucart and Rauhut, 2013), and the $\mathcal{M}_+$ *criterion* (Bruckstein et al., 2008).

Definition G.1 ((Foucart and Rauhut, 2013, Definition 4.17)). *A matrix* $\mathbf{A} \in \mathbb{R}^{M \times N}$ *is said to satisfy the robust null space property (NSP) of order* $s \in [N]$ *with constants* $0 < \rho < 1$ *and* $\tau > 0$ *if for any set* $S \subset [N]$ *of cardinality* $|S| \leq s$, *it holds that* $\|\mathbf{v}_S\|_1 \leq \rho \|\mathbf{v}_{S^c}\|_1 + \tau \|\mathbf{A}\mathbf{v}\|$, *for all* $\mathbf{v} \in \mathbb{C}^N$.

Definition G.2 ((Bruckstein et al., 2008)). *Let* $\mathbf{A} \in \mathbb{R}^{M \times N}$. *We say* $\mathbf{A}$ *obeys the* $\mathcal{M}_+$ *criterion with vector* $\mathbf{u}_{\mathcal{M}_+}$ *if there exists* $\mathbf{u}_{\mathcal{M}_+} \in \mathbb{R}^M$ *such that* $\mathbf{A}^\top \mathbf{u}_{\mathcal{M}_+} > 0$, *i.e., if* $\mathbf{A}$ *admits a strictly-positive linear combination of its rows.*

Note that the $\mathcal{M}_+$ criterion is a necessary condition for an underdetermined ($M < N$) system to admit a unique non-negative solution (Wang et al., 2010, Theorem 5). Examples of matrices $\mathbf{A} \in \mathbb{R}^{M \times N}$ satisfying the $\mathcal{M}_+$ criterion are (i) matrices with i.i.d. Bernoulli entries, (ii) matrices the columns of which can be written as independent 1-subgaussian random vectors (Shadmi et al., 2019), (iii) matrices the columns of which form an outwardly k-neighborly polytope (Donoho and Tanner, 2005), and (iv) adjacency matrices of bipartite expander graphs (Wang et al., 2010). A recent robustness result relying on null space property and $\mathcal{M}_+$ criterion is the following.

Theorem G.3 ((Kueng and Jung, 2017, Theorem 4)). *Suppose that* $\mathbf{A} \in \mathbb{R}^{M \times N}$ *obeys the NSP of order* $s \leq N$ *with constants* $0 < \rho < 1$ *and* $\tau > 0$ *and the* $\mathcal{M}_+$ *criterion with the vector* $\mathbf{u}_{\mathcal{M}_+}$. *Then* $\mathbf{A}$ *allows stable reconstruction of any non-negative s-sparse vector* $\mathbf{x}_*$ *from* $\mathbf{b} = \mathbf{A}\mathbf{x}_* + \boldsymbol{\varepsilon}$ *via* (NNLS). *In particular, for any* $1 \leq p \leq q$, *the unique solution* $\mathbf{x}_+$ *of* (NNLS) *is guaranteed to obey*

$$\|\mathbf{x}_+ - \mathbf{x}_*\|_p \leq \frac{C}{s^{1-1/p}} \sigma_s(\mathbf{x})_1 + \frac{D}{s^{1/q-1/p}} (\|\mathbf{u}_{\mathcal{M}_+}\|_2 + \tau) \|\boldsymbol{\varepsilon}\|_2, \tag{36}$$

where C and D only depend on ρ, the condition number of the diagonal matrix $\text{diag}(\mathbf{A}^\top \mathbf{u}_{\mathcal{M}_+})$.

Theorem G.3, which was later generalized to arbitrary ℓ_p-quasinorms (Shadmi et al., 2019, Theorem 2), shows that if $\mathbf{A}$ behaves sufficiently well, the solution of (NNLS) stably reconstructs any non-negative s-sparse vector from noisy measurements at least as well as conventional programs for sparse recovery. In particular, neither sparsity regularization nor parameter tuning is required. The program only relies on the geometry imposed by its constraints. The work of Shadmi et al. (2019) even showed that (NNLS) outperforms Basis Pursuit Denoising in retrieving a sparse solution from noisy measurements. However, let us mention that measurement operators $\mathbf{A}$ appearing in applications normally do not satisfy the assumptions of Theorem G.3. In such scenarios, an additional sparsity regularization is still needed when working with (NNLS).

In our proposed approach, we get sparse regularization as a by-product of reusing existing theory in our proofs. Indeed, we finally observe that if $\mathbf{x}_0$ is chosen sufficiently close to zero, the limit of the reparametrized gradient flow is approximately minimizing a (weighted) ℓ_1-norm among all possible solutions of (NNLS). The following theorem formalizes these claims. Let us emphasize that a small initialization is only required in Theorem G.4 to obtain additional ℓ_1-regularization. General instances of NNLS can be solved by our method with arbitrary positive initialization. Further note that in order to guarantee a similar additional regularization in the case of general measurements $\mathbf{A}$, the established techniques for solving (NNLS) — grouped as (i)-(iii) in the discussion in Appendix A — would require notable adjustments both in methodology and in theory.

Theorem G.4. *Let* $\varepsilon > 0$. *In addition to the assumptions of Theorem 2.1 assume that* $\mathbf{x}_0 = \alpha\mathbf{1}$. *If*

$$\alpha \leq h(Q_+, \varepsilon) \tag{37}$$

$$:= \begin{cases} \min\left(e^{-\frac{1}{2}}, \exp\left(\frac{1}{2} - \frac{Q_+^2 + Ne^{-1}}{2\varepsilon}\right)\right) & \text{if } L = 2 \\ \left(\frac{2\varepsilon}{L(Q_+ + N + \varepsilon)}\right)^{\frac{1}{L-2}} & \text{if } L > 2 \end{cases},$$

where $Q_+ = \min_{\mathbf{z} \in S_+} \|\mathbf{z}\|_1$, *then the* ℓ_1-norm *of* $\tilde{\mathbf{x}}_\infty$ *satisfies*

$$\|\tilde{\mathbf{x}}_\infty\|_1 - \min_{\mathbf{z} \in S_+} \|\mathbf{z}\|_1 \leq \varepsilon.$$

We have proved Theorem G.4 together with Theorem 2.1 in Section C above. The additional ℓ_1-regularization that is described in Theorem G.4, for small $\alpha > 0$, allows finding NNLS-solutions that are of lower complexity since there is a strong connection between small ℓ_1-norm and effective sparsity. In particular, this allows stable reconstruction of (almost) non-negative ground truths if $\mathbf{A}$ is well-behaved, e.g., if $\mathbf{A}$ satisfies standard assumptions for sparse recovery like suitable robust null space and quotient properties (Foucart and Rauhut, 2013).

Definition G.5 ((Foucart and Rauhut, 2013, Definition 11.11)). *A measurement matrix $\mathbf{A} \in \mathbb{R}^{M \times N}$ is said to possess the ℓ_1-quotient property with constant d relative to the ℓ_2-norm if, for all $\mathbf{b} \in \mathbb{R}^M$, there exists $\mathbf{u} \in \mathbb{R}^N$ with $\mathbf{Au} = \mathbf{b}$ and $\|\mathbf{u}\|_1 \leq d\sqrt{s_*}\|\mathbf{b}\|_2$, where $s_* = M/\ln(eN/M)$.*

Note that many types of matrices satisfy the robust NSP of order s and the ℓ_1-quotient property, e.g., Gaussian random matrices, randomly subsampled Fourier-matrices, and randomly subsampled circulant matrices. For instance, a properly scaled matrix with i.i.d. Gaussian entries satisfies both properties with high probability if $M \geq Cs\log(eN/s)$ and $M \leq N/2$, where the constant $C > 0$ only depends on the NSP parameters ρ and τ (Foucart and Rauhut, 2013). Combining Theorem G.4 and Theorem 3 of Chou et al. (2023a), the following stable recovery result can be derived.

Theorem G.6 (Stability). *Let $\mathbf{A} \in \mathbb{R}^{N \times M}$ be a matrix satisfying the ℓ_2-robust null space property with constants $0 \leq \rho < 1$ and $\tau > 0$ of order $s := cM/\log(eN/M)$ and the ℓ_1-quotient property with respect to the ℓ_2-norm with constant $d > 0$.*

For $\mathbf{x}_* \in \mathbb{R}^N$ and $\mathbf{b} = \mathbf{Ax}_*$, recall $\mathbf{b}_+$ and C_+ from (5). Decompose $\mathbf{x}_*$ into

$$\mathbf{x}_* = \mathbf{x}_+ - \mathbf{x}_- \tag{38}$$

where $\mathbf{Ax}_+ = \mathbf{b}_+$. For $\varepsilon > 0$ assume that $\alpha > 0$ satisfies

$$\alpha \leq h(\|\mathbf{x}_+\|_1, \varepsilon)$$

for h defined in (37). Then the limit $\tilde{\mathbf{x}}_\infty$ defined in Theorems 2.1 and G.4 yields reconstruction error

$$\|\tilde{\mathbf{x}}_\infty - \mathbf{x}_*\|_2 \leq \frac{C}{\sqrt{s}}(2\sigma_s(\mathbf{x}_+)_{\ell_1} + \varepsilon) + \|\mathbf{x}_-\|_2. \tag{39}$$

The constants $C, C' > 0$ only depend on ρ, τ, c, d.

As can be seen from Theorem G.6, our approach to solving (NNLS) is stable with respect to negative entries of the ground truth, i.e., the reconstruction error depends on the sparsity of the positive part $\mathbf{x}_+$ of $\mathbf{x}_*$ and the magnitude of the negative part $\mathbf{x}_-$ of $\mathbf{x}_*$. The experiments we perform in Section 3.5 suggest that the established solvers for (NNLS) are less stable under such perturbations.

Remark G.7. *The very restrictive form of $\mathbf{x}_0 = \alpha\mathbf{1}$ in Theorem G.4 is only required to get an implicit ℓ_1-bias, which is a classical regularizer for sparse recovery (Foucart and Rauhut, 2013). By changing the initialization, one can also achieve other biases like weighted ℓ_1-norms. Below we show that, for $L = 2$ and any $\mathbf{w} \in (0,1]^N$ with $\|\mathbf{w}\|_\infty = 1$, the initialization vector $\mathbf{x}_0$ defined as*

$$\mathbf{x}_0 = e^{-\frac{1}{2}(\mathbf{1}+\theta\mathbf{w})}$$

will yield an approximate $\ell_{\mathbf{w},1}$-bias in the limit of gradient flow, where $\|\mathbf{z}\|_{\mathbf{w},1} = \|\mathbf{z} \odot \mathbf{w}\|_1$ and $\theta > 0$ has to be chosen sufficiently large depending on the aimed for accuracy. Weighted ℓ_1-norms have been used in various applications, e.g., polynomial interpolation or sparse polynomial chaos approximation (Rauhut and Ward, 2016), (Peng et al., 2014).

The statement in Remark G.7 can be deduced from Theorem G.8 below by noticing that, for $L = 2$, the initialization vector $\mathbf{x}_0$ defined as

$$(x_0)_i = e^{-\frac{1}{2}(1+\theta w_i)}$$

satisfies $\mathbf{w} = \frac{(-1-\log(\tilde{\mathbf{x}}(0)))}{\max_{n \in [N]}(-1-\log((\tilde{x}(0))_n))}$ (recall that $\tilde{\mathbf{x}}(0) = \mathbf{x}_0^{\odot 2}$ in this case). The scaling $\theta > 0$, which has no effect on $\mathbf{w}$, only has to be chosen sufficiently large to guarantee $\mathbf{x}_0 \leq h(Q_+, \varepsilon)$.

Theorem G.8. *Let $\varepsilon > 0$ and $L = 2$. Under the assumptions of Theorem G.4 with $\mathbf{x}_0 \leq \alpha \leq h(Q_+, \varepsilon)$, we get that*

$$\|\tilde{\mathbf{x}}_\infty\|_{\mathbf{w},1} - \min_{\mathbf{z} \in S_+} \|\mathbf{z}\|_{\mathbf{w},1} \leq \varepsilon,$$

where $\|\mathbf{z}\|_{\mathbf{w},1} = \|\mathbf{z} \odot \mathbf{w}\|_1$ denotes the weighted ℓ_1-norm for

$$\mathbf{w} = \delta(-\mathbf{1} - \log(\tilde{\mathbf{x}}_0))$$

and

$$\delta = \frac{1}{\max_{n \in [N]}(-1 - \log((\tilde{x}_0)_n))}.$$

Proof. For $L = 2$, we obtain from (17) that

$$\tilde{\mathbf{x}}_\infty \in \arg\min_{\mathbf{z} \in S_+} D_F(\mathbf{z}, \tilde{\mathbf{x}}(0)) = \langle \mathbf{z}, \log(\mathbf{z}) - \mathbf{1} - \log(\tilde{\mathbf{x}}_0) \rangle,$$

where we used the Equation below (2.10) of Chou et al. (2023a) together with the fact that D_F based on F in (15) and D_F of Chou et al. (2023a) are the same, cf. our discussion below (15). Hence

$$\langle \tilde{\mathbf{x}}_\infty, \log(\tilde{\mathbf{x}}_\infty) - \mathbf{1} - \log(\tilde{\mathbf{x}}_0) \rangle \leq \langle \mathbf{z}, \log(\mathbf{z}) - \mathbf{1} - \log(\tilde{\mathbf{x}}_0) \rangle,$$

which may be re-stated as

$$\langle \tilde{\mathbf{x}}_\infty - \mathbf{z}, -\mathbf{1} - \log(\tilde{\mathbf{x}}_0) \rangle \leq \langle \mathbf{z}, \log(\mathbf{z}) \rangle - \langle \tilde{\mathbf{x}}_\infty, \log(\tilde{\mathbf{x}}_\infty) \rangle.$$

Let $\alpha \in (0, e^{-\frac{1}{2}})$ and assume that $\alpha \geq \mathbf{x}_0 > 0$ (so that $\log(\tilde{\mathbf{x}}_0) < -1$). Note that $1 \geq \mathbf{w} \geq 0$ by assumption. Denote

$$\|\mathbf{z}\|_{\mathbf{w},1} = \|\mathbf{z} \odot \mathbf{w}\|_1$$

to be the weighted ℓ_1-norm. By non-negativity of $\mathbf{w}, \tilde{\mathbf{x}}_\infty, \mathbf{z}$, we get that

$$\|\tilde{\mathbf{x}}_\infty\|_{\mathbf{w},1} - \|\mathbf{z}\|_{\mathbf{w},1} \leq \delta(\langle \mathbf{z}, \log(\mathbf{z}) \rangle - \langle \tilde{\mathbf{x}}_\infty, \log(\tilde{\mathbf{x}}_\infty) \rangle).$$

Since $\xi^2 \geq \xi \log(\xi) \geq -e^{-1}$ for $\xi \geq 0$,

$$\|\tilde{\mathbf{x}}_\infty\|_{\mathbf{w},1} - \|\mathbf{z}\|_{\mathbf{w},1} \leq \delta(\|\mathbf{z}\|_2^2 + N e^{-1}) \leq \delta(\|\mathbf{z}\|_1^2 + N e^{-1}) \leq \varepsilon$$

because $\alpha \leq h(Q_+, \varepsilon)$ and $\delta \leq -\frac{1}{1 + 2\log(\alpha)}$. Take the minimum overall $\mathbf{z} \in S_+$, and we get our conclusion. $\qquad\square$

H   DETAILS FOR ACCELERATED GRADIENT METHODS

To transform (ACC) in an optimization method, we perform a change of variables. In the case where $L = 2$, $q = 1$ and the new variable is $\boldsymbol{\zeta} = \log(\boldsymbol{\xi})$, so that the $\boldsymbol{\xi}$ equation becomes equivalent to $\boldsymbol{\zeta}' = -\frac{t}{2}\nabla\ell(\tilde{\mathbf{x}}(t))$. Using an implicit step for $\tilde{\mathbf{x}}$ and an explicit step for $\boldsymbol{\zeta}$ yields the following update scheme:

$$\tilde{\mathbf{x}}_{k+1} = \frac{\tilde{\mathbf{x}}_k + 2/(k+2)e^{\boldsymbol{\zeta}_k}}{1 + 2/(k+2)},$$

$$\boldsymbol{\zeta}_{k+1} = \boldsymbol{\zeta}_k - \eta\frac{k+2}{2}\nabla\ell(\tilde{\mathbf{x}}). \tag{40}$$

Nesterov's acceleration for the dynamics $\mathbf{x}'(t) = -\nabla\mathcal{L}(\mathbf{x})$ is an usual one:

$$\mathbf{y}_{k+1} = \mathbf{x}_k - \eta\nabla\mathcal{L}(\mathbf{x}_k),$$

$$\mathbf{x}_{k+1} = \mathbf{y}_{k+1} + \frac{k}{k+3}(\mathbf{y}_{k+1} - \mathbf{y}_k). \tag{41}$$

I   ADDITIONAL NUMERICAL EXPERIMENTS

In this section, we provide additional empirical evidence for our theoretical claims. In particular, (i.) we show, for Gaussian signals, the stability of our approach with respect to negative entries, as similarly done in Section 3.5, (ii.) we illustrate the performance of our method in an over-determined large-scale NNLS problem, and (iii.) we compare our method to projected gradient descent (**PGD**) (Polyak, 2015) in terms of the number of iterations, convergence rate, and running time since it is known that **PGD** converges linearly to the global minimizer (Attouch et al., 2013; Polyak, 2015) and it is a numerically efficient method due to the fast calculation of the projection step.

I.1   Experimental setup

The results of this paper were obtained on a desktop computer running Python 3.9 on an Intel Core i5-7500 CPU with 8GB of RAM.

I.2   NNLS Stability with Negative Entries

Here, we continue the comparison of Section 3.5 and analyze the reconstruction error of several NNLS algorithms with respect to the magnitude of the negative component.

Gaussian signals   Let $\mathbf{A} \in \mathbb{R}^{M \times N}$ be a random Gaussian matrix, where $M = 30$ and $N = 50$. We pick a 3-sparse non-negative vector $\mathbf{x}_+ \in \mathbb{R}^N_{\geq 0}$ at random. We, furthermore, define a noise vector $\mathbf{x}_- \in \mathbb{R}^N_{\geq 0}$ that is 0 on the support of $\mathbf{x}_+$ and has positive Gaussian entries everywhere else. For $q \in [0, 1]$, we scale $\mathbf{x}_+, \mathbf{x}_-$ such that

$$\|\mathbf{x}_+\|_2^2 = 1 - q \quad \text{and} \quad \|\mathbf{x}_-\|_2^2 = q. \tag{42}$$

The perturbed signal is then given by $\mathbf{x} = \mathbf{x}_+ - \mathbf{x}_-$, and q regulates the negative corruption. We regard the copy of $\mathbf{x}_+$ scaled to ℓ_2-norm $(1 - q)$ as ground truth and, by abuse of notation, also refer to it as $\mathbf{x}_+$. The corresponding measurements are given as $\mathbf{b} = \mathbf{A}\mathbf{x} = \mathbf{A}(\mathbf{x}_+ - \mathbf{x}_-)$.

Figures 6a, 6b and 6c compare the reconstruction error $\|\hat{\mathbf{x}} - \mathbf{x}_+\|_2$ of **HL-NNLS**, **TNT-NN**, **GD-3L**, and **SGD-3L** over various choices of q. The results support our claim that, when applying NNLS to sparse recovery, our approach comes with improved stability against negative perturbations of the ground-truth.
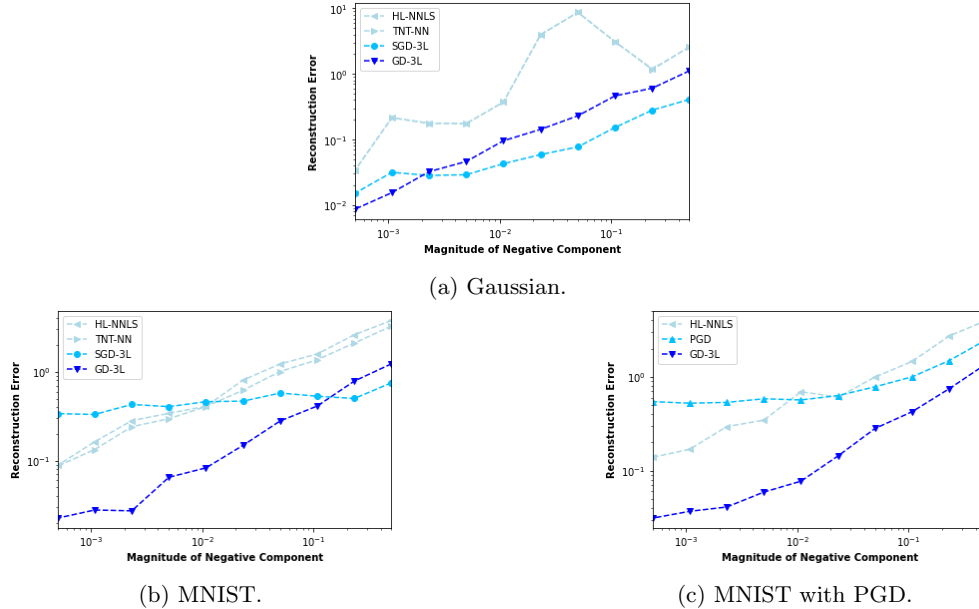

(a) Gaussian.

(b) MNIST.

(c) MNIST with PGD.

Figure 6: Comparison of stability with respect to negative entries, see Section I.2.

I.3   Comparison of GD variants for solving NNLS

In this section, we present experiments comparing various gradient descent (GD) variants for solving non-negative least squares (NNLS) problems. We examine the performance of these methods on both dense and sparse problems, focusing on convergence behavior and support identification. We consider two scenarios:

1. Dense case: A 512×512 linear system with a dense vector with sparsity equal to 512.

2. Sparse case: A 512×1024 linear system with an 8-sparse vector $\mathbf{x} \in \mathbb{R}^{512}$.

For both cases, we use a vanilla step size of 3.16×10^{-1}. We compare the following methods: Vanilla **GD-2L** with constant stepsize, ACC (our accelerated method based on Theorem 2.4), Nesterov's accelerated gradient descent, Restarted Nesterov's method, **GD-2L** with the Barzilai-Borwein step size Barzilai-Borwein (BB) stepsize

$$\eta = \frac{\|\mathbf{x}_t - \mathbf{x}_{t-1}\|_2^2}{\|\mathbf{A}(\mathbf{x}_t - \mathbf{x}_{t-1})\|_2^2},$$

(e.g., Barzilai and Borwein, 1988), and Projected Gradient Descent (**PGD**), which proceeds as follows.

PGD: Initialize with $\mathbf{x}_0 \geq 0$, e.g., $\mathbf{x}_0 = 0$. And for $k = 0, 1, 2, \ldots$ until convergence:

(i) Compute the gradient: $\nabla f(\mathbf{x}_k) = \mathbf{A}^T(\mathbf{A}\mathbf{x}_k - \mathbf{b})$.

(ii) Choose step size.

(iii) Update: $\mathbf{x}_{k+1} = P_+[\mathbf{x}_k - \alpha_k \nabla f(\mathbf{x}_k)]$, where P_+ is the projection onto the non-negative orthant $P_+[\mathbf{x}]_i = \max(0, \mathbf{x}_i)$.

We also examine the effect of different initialization scales, ranging from $\alpha = 10^{-9}$ to 10^{-1}.
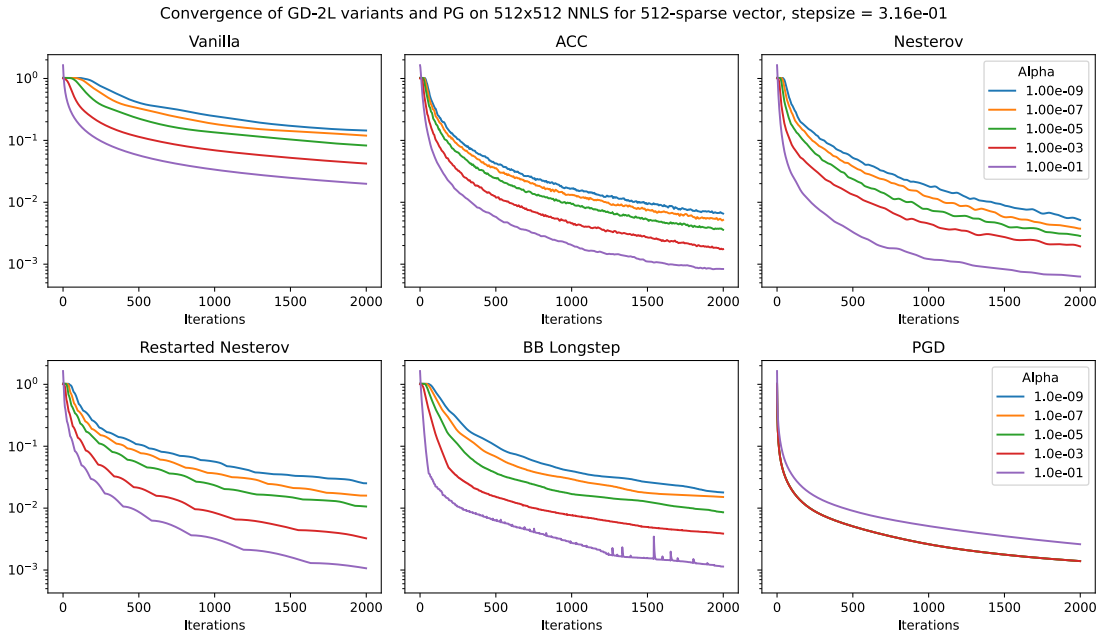

Figure 7: Error $\|\mathbf{A}\tilde{\mathbf{x}} - \mathbf{b}\|_2$ for GD-2L variants and **PGD** in the dense case.

Figure 7 shows the convergence behavior of various **GD-2L** variants and **PGD** for the dense case. The y-axis represents the error $\|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2$, and the x-axis shows the number of iterations. We see that all methods decreased the error below 10^{-1} provided a sufficient number of iterations. The ACC and Nesterov variants demonstrate improved convergence rates compared to vanilla **GD-2L**, while the BB Longstep method shows competitive

performance, especially in later iterations. Notably, the initialization scale (α) has a significant impact on convergence speed, with larger α generally leading to faster convergence.

Figure 8 presents two aspects of the sparse case: convergence behavior, cf. Subfigure 8 (a) and support identification, cf. Subfigure 8(b). We observe that the overall trends are similar to the dense case but with faster convergence due to the problem's sparsity. **PGD** again shows rapid initial progress, while the accelerated methods (ACC, Nesterov, Nesterov with a restarting strategy that improves its convergence, e.g., (Roulet and d'Aspremont, 2017)) demonstrate improved convergence rates over vanilla **GD-2L**. Unlike the dense case, smaller initialization scales generally lead to faster convergence, especially for the accelerated methods.

Figure 8 illustrates how well each method identifies the true support of the sparse solution over iterations. We find that despite being a fast method, **PGD** struggles to achieve accurate support identification. In contrast, all variants of the overparametrized GD perform well in this task.

The experiments demonstrate that the variants of overparametrized **GD-2L**, particularly the accelerated methods (ACC, Nesterov, and Restarted Nesterov), offer significant advantages in solving NNLS problems. These methods consistently show improved convergence rates compared to vanilla **GD-2L**, especially in the sparse case. Moreover, they excel in support identification, a crucial aspect in many applications requiring sparse solutions.

The overparametrized GD variants demonstrate remarkable flexibility across different problem settings. They perform well in both dense and sparse scenarios, adapting effectively to various initialization scales. This adaptability, combined with their strong performance in support identification, makes them particularly suitable for a wide range of NNLS applications.

The BB Longstep variant also demonstrates competitive performance, especially in later iterations, offering another valuable option within the overparametrized GD framework. Furthermore, the ACC method, based on our theoretical results in Theorem 2.4, shows promising performance, validating the practical relevance of our theoretical contributions.

I.4   Experiments on large linear systems

In this experiment, cf. Figure 9, we illustrate the performance of our method for two large-scale NNLS problems. The first, in Figure 9a, treats an overdetermined setting, where the matrix $\mathbf{A} \in \mathbb{R}^{2048 \times 1024}$ is standard Gaussian and, for a Gaussian random vector $\mathbf{x} \in \mathbb{R}^{1024}$, $\mathbf{b}$ is created as a perturbed version of $\mathbf{Ax}$ such that $\mathbf{b}$ is not in the range of $\mathbf{A}$. We take $\mathbf{x}_0 = \mathbf{1}$ as a generic initialization. Figure 9a shows the error $\|\mathbf{Ax}^{\odot L} - \mathbf{b}\|_2$ over the iterations of gradient descent. As predicted by Theorem 2.1, our method converges to a solution that solves the NNLS problem (compare the error to the benchmark given by the Lawson-Hanson algorithm).

A second experiment deals with a subsampled FFT, using $8\,810$ random rows of a $100\,000$-dimensional FFT to build the matrix $\mathbf{A}$. A ground-truth $\mathbf{x}^*$ with 50 positive entries is chosen, and again we set $\mathbf{b} = \mathbf{Ax}^*$. We use two layers, set a constant stepsize of 0.2, and vary the initialization $\mathbf{x}_0 = \alpha \mathbf{1}$ for α from 10^{-6} to 10^{-2}. As shown in Figure 9b, we obtain two different error regimes: a smoothly decaying error for the relatively "large" initialization 10^{-2} and a slower but sharper decay that essentially stagnates after a given point, for the remaining ones. Notice that the stagnation is smaller as the initialization gets closer to zero, as expected by our theory, and it corresponds to better support identification, while the initialization at $\alpha = 10^{-2}$ leads to a solution that is significantly different from the initial sparse ground truth.

Our third experiment analyses the behavior of several algorithms for NNLS over a sequence of square problems with dimensions 500, 1000, 2000 and 4000. In addition to our reparameterized GD algorithms, indicated as **GD** (for 2-layer gradient descent) and its accelerated versions **GD-BB** (with Barzilai-Borwein large-step), **GD-Nesterov** (accelerating GD, equation (41)) and **MD-Nesterov** (as in equation (40), the discrete version of (ACC)), we include Lawson and Hanson's algorithm **LH**, projected gradient descent **PG** and accelerated projected gradient descent **APG**. For each matrix size, we selected 10 Gaussian matrices and a target vector with positive entries, and stored running time and final error, $\|Ax - b\|_2$ at $10\,000$ iterations.

In Figure 10a, we notice that the Lawson and Hanson algorithm does not scale as well as the others, essentially due to its need to solve larger and larger linear systems, whereas the others only rely on matrix-vector multiplication. Besides that, we emphasize once more that this algorithm, to our best knowledge, does not have asymptotic convergence rates, but it is only known that it stops after a finite number of iterations.

In Figure 10b, we observe that the error that each method obtains after $10\,000$ iterations is relatively stable as the matrix size increases. Lawson and Hanson's algorithm reaches an almost exact solution and is therefore not displayed to avoid distorting the axis. Among the visible methods, **APG** achieves the lowest error (around 10^{-5}), showing a significant advantage over other approaches. Our accelerated methods (GD-BB, GD-Nesterov, and MD-Nesterov) form a middle cluster with errors between $\mathrm{O}(10^{-4})$ and $\mathrm{O}(10^{-3})$, demonstrating the effectiveness of acceleration techniques in our reparameterized framework. As expected from our previous experiments, vanilla gradient descent converges more slowly, reaching only $\mathrm{O}(10^{-2})$ error within the same number of iterations.

While projected gradient methods like **PG** and **APG** achieve smaller regression errors for the same number of iterations, our experiments demonstrate that they are less effective for sparse recovery tasks. Our reparameterized GD methods excel in identifying the correct support of sparse solutions and show better stability when dealing with negative perturbations in the ground truth signal. Moreover, Figure Figure 10a confirms that our methods scale efficiently with problem size, making them particularly suitable for large-scale applications where both computational efficiency and support identification are important considerations.

I.5 Visual experiments: using NNLS for retrieving images

In Figure 11 we report the result of the CIFAR10 experiment in Section 3.5. These visualizations demonstrate how different NNLS methods perform when reconstructing grayscale CIFAR10 images corrupted with negative components. The images clearly illustrate the varying abilities of these algorithms to recover visual information under challenging conditions. Unlike the MNIST experiments, we observe that for CIFAR10 images (which typically have less distinct sparse structures), **GD-3L** provides the most accurate reconstructions while **SGD-3L** tends to induce excessive sparsity, particularly at higher corruption levels ($q = 0.5$).
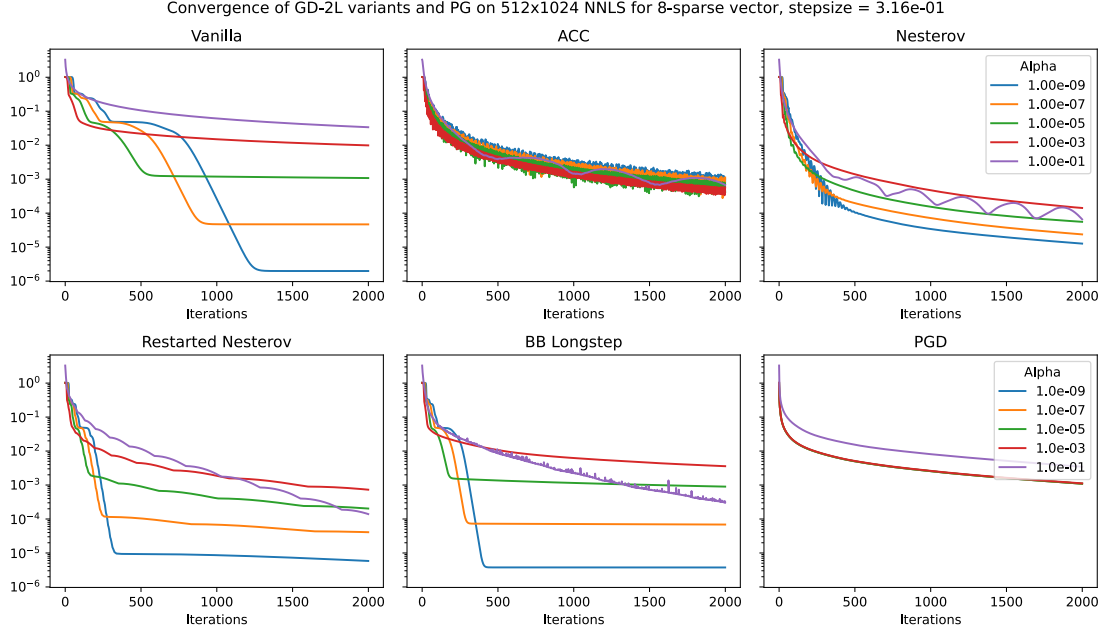
(a) Error $\|\mathbf{A}\tilde{\mathbf{x}} - \mathbf{b}\|_2$ along iterations.
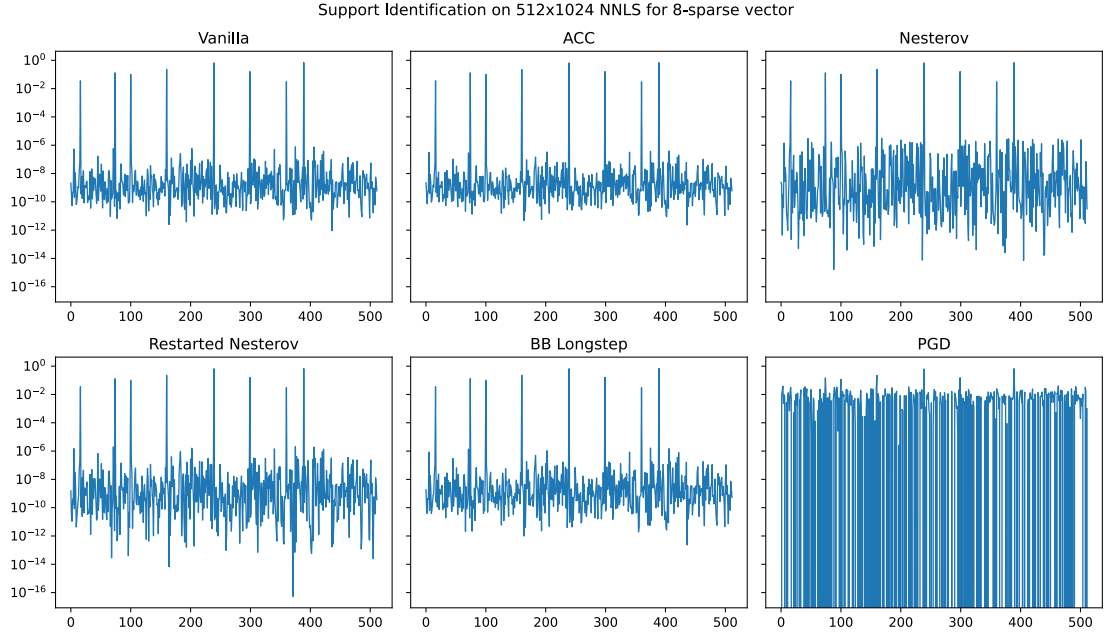
(b) Support identification

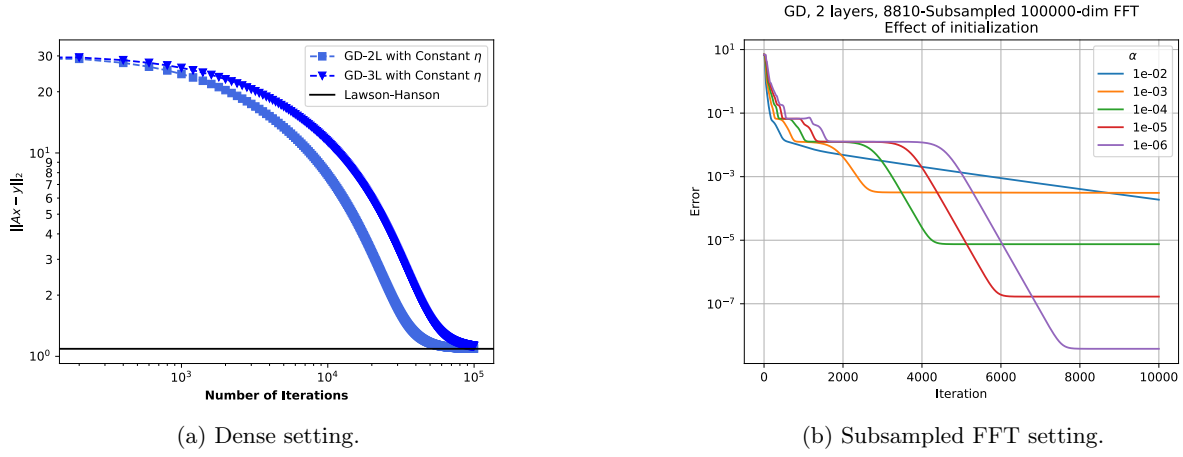Figure 8: Comparison of **GD-2L** variants and **PGD** in the sparse case.

(a) Dense setting.

(b) Subsampled FFT setting.

Figure 9: Large-scale NNLS problems, see Section I.4.

(a) Time to perform 10k iterations

(b) Error at 10k iterations
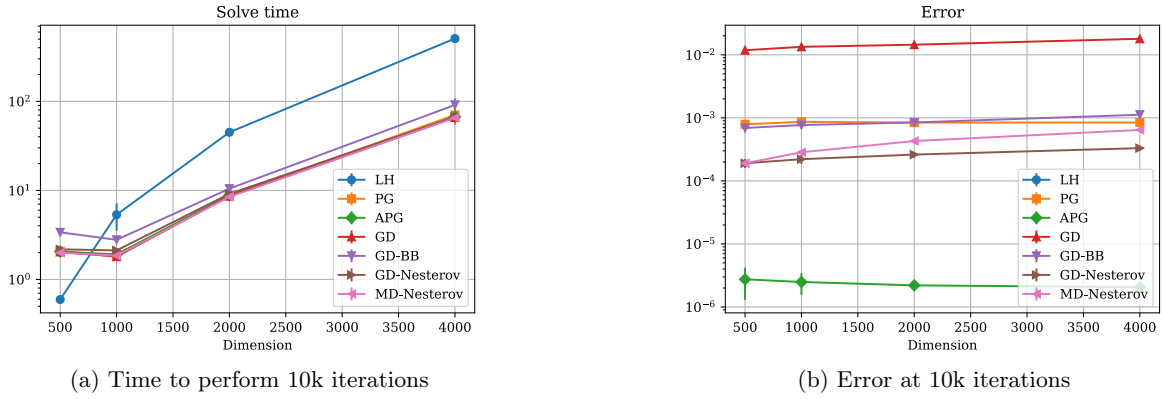
Figure 10: Timing and error behavior across matrix sizes for square NNLS problems, see Section I.4.

HL-NNLS
$(q = 0.5)$

TNT-NN
$(q = 0.5)$

SGD-3L
$(q = 0.5)$

GD-3L
$(q = 0.5)$

PGD
$(q = 0.5)$

HL-NNLS
$(q = 0.05)$

TNT-NN
$(q = 0.05)$
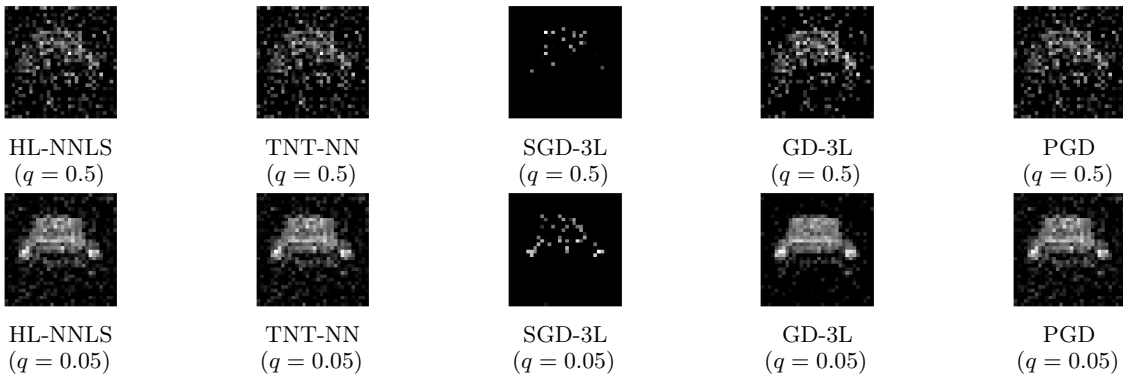
SGD-3L
$(q = 0.05)$

GD-3L
$(q = 0.05)$

PGD
$(q = 0.05)$

Figure 11: Illustration of the CIFAR10 reconstruction, see Section 3.5