# Fast and Provable ADMM for Learning with Generative Priors

**Fabian Latorre, Armin Eftekhari and Volkan Cevher**
Laboratory for information and inference systems (LIONS)
EPFL, Lausanne, Switzerland
{firstname.lastname}@epfl.ch

## Abstract

In this work, we propose a (linearized) Alternating Direction Method-of-Multipliers (ADMM) algorithm for minimizing a convex function subject to a nonconvex constraint. We focus on the special case where such constraint arises from the specification that a variable should lie in the range of a neural network. This is motivated by recent successful applications of Generative Adversarial Networks (GANs) in tasks like compressive sensing, denoising and robustness against adversarial examples. The derived rates for our algorithm are characterized in terms of certain geometric properties of the generator network, which we show hold for feedforward architectures, under mild assumptions. Unlike gradient descent (GD), it can efficiently handle non-smooth objectives as well as exploit efficient partial minimization procedures, thus being faster in many practical scenarios.

## 1 Introduction

Generative Adversarial Networks (GANs) [Goodfellow et al., 2014] show great promise for faithfully modeling complex data distributions, such as natural images [Radford et al., 2015, Brock et al., 2019] or audio signals [Engel et al., 2019, Donahue et al., 2019]. Understanding and improving the theoretical and practical aspects of their training has thus attracted significant interest [Lucic et al., 2018, Mescheder et al., 2018, Daskalakis et al., 2018, Hsieh et al., 2018, Gidel et al., 2019].

Researchers have also begun to leverage the modeling power of GANs and other generative models like Variational Auto-encoders [Kingma and Welling, 2013] in applications ranging from compressive sensing [Bora et al., 2017], to image denoising [Lipton and Tripathi, 2017, Tripathi et al., 2018], to robustness against adversarial examples [Ilyas et al., 2017, Samangouei et al., 2018].

These and other [Dhar et al., 2018, Ulyanov et al., 2018] applications model high-dimensional data as the output of the generator network associated with a generative model, and often lead to a highly non-convex optimization problem of the form $\min_z f(G(z))$, where the the generator $G$ is nonlinear and $f$ is convex. We then find the optimal *latent vector* $z$, as illustrated in Section 5 with several examples.

This optimization problem involving a generative model poses various difficulties for existing first-order algorithms. Indeed, to our knowledge, the only existing provable algorithm for solving (1) relies on the existence of a projection oracle, and is limited to the special case of *compressive sensing* with a generative prior [Shah and Hegde, 2018, Hegde, 2018], see Section 4 for the details. The main computational bottleneck is of course the non-convex projection step, for which no convergence analysis in terms of the geometry of the underlying generator $G$ currently exists.

On the other hand, Gradient Descent (GD) and its adaptive variants [Kingma and Ba, 2014] cannot efficiently handle non-smooth objective functions, as they are entirely oblivious to the composite structure of the problem [Nesterov, 2013b]. A simple example is denoising with the $\ell_\infty$-norm,

33rd Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, Canada.

for which subgradient descent (as the standard non-smooth alternative to GD) fails in practice, as observed in Section 5.

With the explosion of generative models in popularity, there is consequently a pressing need for provable and flexible optimization algorithms to solve the resulting non-convex and (possibly) non-smooth problems. The present work addresses this need by focusing on the general optimization template

$$\begin{array}{ll} \underset{w,z}{\text{minimize}} & F(w,z) := L(w) + R(w) + H(z) \\ \text{subject to} & w = G(z), \end{array} \tag{1}$$

where $L : \mathbb{R}^d \to \mathbb{R}$ is convex and smooth, $R : \mathbb{R}^d \to \mathbb{R}$ and $H : \mathbb{R}^s \to \mathbb{R}$ are convex but not necessarily smooth, and $G : \mathbb{R}^s \to \mathbb{R}^d$ is differentiable but often non-linear, corresponding to the generator network associated with a generative model. Even though $R$ and $H$ might not be smooth, we assume throughout that their proximal mappings can be efficiently computed [Parikh et al., 2014].

For brevity, we refer to (1) as *optimization with a generative prior* whenever $G$ is given by the generator network associated with a generative model [Kingma and Welling, 2013, Goodfellow et al., 2014]. In this context, we make three key contributions, summarized below:

**1. Algorithm:** We propose an efficient and scalable (linearized) Alternating Direction Method-of-Multipliers (ADMM) framework to solve (1), see Algorithm 1. To our knowledge, this is the first non-convex and linearized ADMM algorithm for nonlinear constraints with provable fast rates to solve problem (1), see Section 4 for a detailed literature review.

We evaluate this algorithm numerically in the context of denoising with GANs in the presence of adversarial or stochastic noise, as well as compressive sensing [Bora et al., 2017]. In particular, Algorithm 1 allows for efficient denoising with the $\ell_\infty$- and $\ell_1$-norms, with applications in defenses against adversarial examples [Szegedy et al., 2013] and signal processing, respectively.

**2. Optimization guarantees:** We prove fast approximate convergence for Algorithm 1 under the assumptions of smoothness and near-isometry of $G$, as well as strong convexity of $L$. That is, we distill the key geometric attributes of the generative network $G$ responsible for the success of Algorithm 1. We then show how some common neural network architectures satisfy these geometric assumptions.

We also establish a close relation between a variant of Algorithm 1 and the gradient descent in [Bora et al., 2017] and, in this sense, provide the first rates for it, albeit in a limit case detailed in Section 3. Indeed, one key advantage of the primal-dual formulation studied in this paper is exactly this versatility, as well as the efficient handling of non-smooth objectives.

Lastly, we later relax the assumptions on $L$ to *restricted* strong convexity/smoothness, thus extending our results to the broader context of statistical learning with generative priors, which includes compressive sensing [Bora et al., 2017] as a special case.

**3. Statistical guarantees:** In the context of statistical learning with generative priors, where $L$ in (1) is replaced with an *empirical risk*, we provide the generalization error associated with Algorithm 1. That is, we use the standard notion of Rademacher complexity [Mohri et al., 2018] to quantify the number of training data points required for Algorithm 1 to learn the true underlying parameter $w^\natural$.

## 2 Algorithm

In this section, we adapt the powerful Alternating Descent Method of Multipliers (ADMM) [Glowinski and Marroco, 1975, Gabay and Mercier, 1976, Boyd et al., 2011] to solve the non-convex problem (1). We define the corresponding *augmented Lagrangian* with the dual variable $\lambda \in \mathbb{R}^p$ as

$$\mathcal{L}_\rho(w,z,\lambda) := L(w) + \langle w - G(z), \lambda \rangle + \frac{\rho}{2}\|w - G(z)\|_2^2, \tag{2}$$

for a penalty weight $\rho > 0$. By a standard duality argument, (1) is equivalent to

$$\min_{w,z} \max_{\lambda} \mathcal{L}_\rho(w,z,\lambda) + R(w) + H(z). \tag{3}$$

Applied to (3), every iteration of ADMM would minimize the augmented Lagrangian with respect to $z$, then with respect to $w$, and then update the dual variable $\lambda$. Note that $\mathcal{L}_\rho(w,z,\lambda)$ is often

non-convex with respect to $z$ due to the nonlinearity of the generator $G : \mathbb{R}^s \to \mathbb{R}^d$ and, consequently, the minimization step with respect to $z$ in ADMM is often intractable.

To overcome this limitation, we next *linearize* ADMM. In the following, we let $\mathrm{P}_R$ and $\mathrm{P}_H$ denote the *proximal maps* of $R$ and $H$, respectively [Parikh et al., 2014].

The equivalence of problems (1) and (3) motivates us to consider the following algorithm for the penalty weight $\rho > 0$, the primal step sizes $\alpha, \beta > 0$, and the positive dual step sizes $\{\sigma_t\}_{t\geq 0}$:

$$\begin{aligned} z_{t+1} &= \mathrm{P}_{\beta H}\left(z_t - \beta \nabla_z \mathcal{L}_\rho(w_t, z_t, \lambda_t)\right), \\ w_{t+1} &= \mathrm{P}_{\alpha R}\left(w_t - \alpha \nabla_w \mathcal{L}_\rho(w_t, z_{t+1}, \lambda_t)\right), \\ \lambda_{t+1} &= \lambda_t + \sigma_{t+1}(w_{t+1} - G(z_{t+1})). \end{aligned} \tag{4}$$

As opposed to ADMM, to solve (1), the linearized ADMM in (4) takes only one descent step in both $z$ and $w$, see Algorithm 1 for the summary. The particular choice of the dual step sizes $\{\sigma_t\}_t$ in Algorithm 1 ensures that the dual variables $\{\lambda_t\}_t$ remain bounded, see [Bertsekas, 1976] for a precedent in the convex literature.

**Algorithm 2.** Let us introduce an important variant of Algorithm 1. In our setting, $\mathcal{L}_\rho(w, z, \lambda)$ is in fact convex with respect to $w$ and therefore Algorithm 2 replaces the first step in (4) with exact minimization over $w$. This exact minimization step can be executed with an off-the-shelf convex solver, or might sometimes have a closed-form solution. Moreover, Algorithm 2 gradually increases the penalty weight to emulate a multi-scale structure. More specifically, for an integer $K$, consider the sequences of penalty weights and primal step sizes $\{\rho_k, \alpha_k, \beta_k\}_{k=1}^K$, specified as

$$\rho_k = 2^k \rho, \qquad \alpha_k = 2^{-k}\alpha, \qquad \beta_k = 2^{-k}\beta, \qquad k \leq K. \tag{5}$$

Consider also a sequence of integers $\{n_k\}_{k=1}^K$, where

$$n_k = 2^k n, \qquad k \leq K, \tag{6}$$

for an integer $n$. At (outer) iteration $k$, Algorithm 2 executes $n_k$ iterations of Algorithm 1 with exact minimization over $w$. Then it passes the current iterates of $w$, $z$, and dual step size to the next (outer) iteration. Loosely speaking, Algorithm 2 has a multi-scale structure, allowing it to take larger steps initially and then slowing down as it approaches the solution. As discussed in Section 3, the theoretical guarantees for Algorithm 1 also apply to Algorithm 2. The pseudocode for Algorithm 2 is given in Supplementary I.

As the closing remark, akin to the convex case [He et al., 2000, Xu et al., 2017], it is also possible to devise a variant of Algorithm 1 with adaptive primal step sizes, which we leave for a future work.

**Algorithm 1** Linearized ADMM for solving problem (1)

**Input:** Differentiable $L$, proximal-friendly convex regularizers $R$ and $H$, differentiable prior $G$, penalty weight $\rho > 0$, primal step sizes $\alpha, \beta > 0$, initial dual step size $\sigma_0 > 0$, primal initialization $w_0$ and $z_0$, dual initialization $\lambda_0$, stopping threshold $\tau_c > 0$.

**for** $t = 0, 1, \ldots, T-1$ **do**
  $z_{t+1} \leftarrow \mathrm{P}_{\beta H}\left(z_t - \beta \nabla_z \mathcal{L}_\rho(w_t, z_t, \lambda_t)\right)$ (primal updates)
  $w_{t+1} \leftarrow \mathrm{P}_{\alpha R}\left(w_t - \alpha \nabla_w \mathcal{L}_\rho(w_t, z_{t+1}, \lambda_t)\right)$
  $\sigma_{t+1} \leftarrow \min\left(\sigma_0, \dfrac{\sigma_0}{\|w_{t+1} - G(z_{t+1})\|_2 t \log^2(t+1)}\right)$ (dual step size)
  $\lambda_{t+1} \leftarrow \lambda_t + \sigma_{t+1}(w_{t+1} - G(z_{t+1}))$ (dual update)
  $s \leftarrow \dfrac{\|z_{t+1} - z_t\|_2^2}{\alpha} + \dfrac{\|w_{t+1} - w_t\|_2^2}{\beta} + \sigma_t \|w_t - G(z_t)\|_2^2 \leq \tau_c$ (stopping criterion)
  **if** $s \leq \tau_c$ **then**
    **return** $(w_{t+1}, z_{t+1})$
**return** $(w_T, z_T)$

## 3 Optimization Guarantees

Let us study the theoretical guarantees of Algorithm 1 for solving program (1), whose constraints are nonlinear and non-convex (since $G$ is specified by a neural network). The main contribution of this section is Theorem 1, which is inherently an optimization result stating that Algorithm 1 succeeds under certain assumptions on (1).

From an optimization perspective, to our knowledge, Theorem 1 is the first to provide (fast) rates for non-convex and linearized ADMM, see Section 4 for a detailed literature review. The assumptions imposed below on $L$ and the generator $G$ ensure the success of Algorithm 1 and are shortly justified for our setup, where $G$ is a generator network.

**Assumption 1. strong convexity / smoothness of $L$:** *We assume that $L$ in (1) is both strongly convex and smooth, namely, there exist $0 < \mu_L \leq \nu_L$ such that*

$$\frac{\mu_L}{2}\|w - w'\|^2 \leq L(w') - L(w) - \langle w' - w, \nabla L(w)\rangle \leq \frac{\nu_L}{2}\|w - w'\|^2, \quad \forall w, w' \in \mathbb{R}^d. \tag{7}$$

Assumption 1 is necessary to establish fast rates for Algorithm 1, and is readily met for $L(w) = \|w - \widehat{w}\|_2^2$ with $\mu_L = \nu_L = 1$, which renders Algorithm 1 applicable to $\ell_2$-denoising with generative prior in [Tripathi et al., 2018, Samangouei et al., 2018, Ilyas et al., 2017]. Here, $\widehat{w}$ is the noisy image.

In Supplementary A, we also relax the strong convexity/smoothness in Assumption 1 to *restricted* strong convexity/smoothness, which enables us to apply Theorem 1 in the context of statistical learning with a generative prior, for example in compressive sensing [Bora et al., 2017].

Under Assumption 1, even though $L$ and consequently the objective function of (1) are strongly convex, problem (1) might *not* have a unique solution, which is in stark contrast with convex optimization. Indeed, a simple example is minimizing $x^2 + y^2$ with the constraint $x^2 + y^2 = 1$. We next state our assumptions on the generator $G$.

**Assumption 2. Strong smoothness of $G$:** *Let $DG$ be the Jacobian of $G$. We assume that $G : \mathbb{R}^s \to \mathbb{R}^d$ is strongly smooth, namely, there exists $\nu_G \geq 0$ such that*

$$\|G(z') - G(z) - DG(z) \cdot (z' - z)\|_2 \leq \frac{\nu_G}{2}\|z' - z\|_2^2, \qquad \forall z, z' \in \mathbb{R}^s, \tag{8}$$

**Assumption 3. Near-isometry of $G$:** *We assume that the generative prior $G$ is a near-isometric map, namely, there exist $0 < \iota_G \leq \kappa_G$ such that*

$$\iota_G\|z' - z\|_2 \leq \|G(z') - G(z)\|_2 \leq \kappa_G\|z' - z\|_2, \qquad \forall z, z' \in \mathbb{R}^s. \tag{9}$$

The invertibility of certain network architectures have been established before in [Ma et al., 2018, Hand and Voroninski, 2017]. More concretely, Assumptions 2 and 3 hold for a broad class of generators, as summarized in Proposition 1 and proved in Supplementary B.

**Proposition 1.** *Let $G_\Xi : \mathcal{D} \subset \mathbb{R}^d \to \mathbb{R}^s$ be a feedforward neural network with weights $\Xi \in \mathbb{R}^h$, $k$ layers, non-decreasing layer sizes $s \leq s_1 \leq \ldots s_k \leq d$, with $\omega_i$ as activation function in the $i$-th layer, and compact domain $\mathcal{D}$. For every layer $i$, suppose that the activation $\omega_i : \mathbb{R} \to \mathbb{R}$ is of class $C^1$ (continuously-differentiable) and strictly increasing. Then, after an arbitrarily small perturbation to the weights $\Xi$, Assumptions 2 and 3 hold almost surely with respect to the Lebesgue measure.*

A few comments about the preceding result are in order.

**Choice of the activation function:** Strictly-increasing $C^1$ activation functions in Proposition 1, such as the Exponential Linear Unit (ELU) [Clevert et al., 2015] or softplus [Dugas et al., 2001], achieve similar or better performance compared to the commonly-used (but non-smooth) Rectified Linear Activation Unit (ReLU) [Xu et al., 2015, Clevert et al., 2015, Gulrajani et al., 2017, Kumar et al., 2017, Kim et al., 2018].

In our experiments in Section 5, we found that using ELU activations for the generator $G$ does not adversely affect the representation power of the trained generator. Lastly, the activation function for the final layer of the generator is typically chosen as the sigmoid or tanh [Radford et al., 2015], for which the conditions in Proposition 1 are also met.

**Compact domain:** The compactness requirement in Proposition 1 is mild. Indeed, even though the Gaussian distribution is the default choice as the input for the generator in GANs, training has also

been successful using compactly-supported distributions, such as the uniform distribution [Lipton and Tripathi, 2017].

Interestingly, even after training with Gaussian noise, limiting the resulting generator to a truncated Gaussian distribution can in fact boost the performance of GANs [Brock et al., 2019], as measured with common metrics like the Inception Score [Salimans et al., 2016] or Frechet Inception Distance [Heusel et al., 2017]. This evidence suggests that obtaining a good generator $G$ with compact domain is straightforward. In the experiments of Section 5, we use truncated Gaussian on an Euclidean ball centered at the origin.

**Non-decreasing layer sizes:** This is a standard feature of popular generator architectures such as the DCGAN [Radford et al., 2015] or infoGAN [Chen et al., 2016]. This property is also exploited in the analysis of the optimization landscape of problem (1) by Hand and Voroninski [2017], Heckel et al. [2019] and for showing invertiblity of (de)convolutional generators [Ma et al., 2018].

**Necessity of assumptions on $G$:** Assumptions 2 and 3 on the generator $G$ are necessary for the provable success of Algorithm 1. Loosely speaking, Assumption 2 controls the curvature of the generative prior, without which the dual iterations can oscillate without improving the objective.

On the other hand, the lower bound in (9) means that the generative prior $G$ must be *stably* injective: Faraway latent parameters should be mapped to faraway outputs under $G$. As a pathological example, consider the parametrization of a circle as $\{(\sin z, \cos z) : z \in [0, 2\pi)\}$.

This stable injectivity property in (9) is necessary for the success of Algorithm 1 and is not an artifact of our proof techniques. Indeed, without this condition, the $z$ updates in Algorithm 1 might not reduce the feasibility gap $\|w - G(z)\|_2$. Geometric assumptions on nonlinear constraints have precedent in the optimization literature [Birgin et al., 2016, Flores-Bazán et al., 2012, Cartis et al., 2018] and to a lesser extent in the literature of neural networks too [Hand and Voroninski, 2017, Ma et al., 2018], which we further discuss in Section 4.

Having stated and justified our assumptions on $L$ and the generator $G$ in (1), we are now prepared to present the main technical result of this section. Theorem 1 states that Algorithm 1 converges linearly to a small neighborhood of a solution, see Supplementary C for the proof.

**Theorem 1. (guarantees for Algorithm 1)** *Suppose that Assumptions 1-3 hold. Let $(w^*, z^*)$ be a solution of program (1) and let $\lambda^*$ be a corresponding optimal dual variable. Let also $\{w_t, z_t, \lambda_t\}_{t\geq 0}$ denote the output sequence of Algorithm 1. Suppose that the primal step sizes $\alpha, \beta$ satisfy*

$$\alpha \leq \frac{1}{\nu_\rho}, \qquad \beta \leq \frac{1}{\xi_\rho + 2\alpha\tau_\rho^2}. \qquad \sigma_0 \leq \sigma_{0,\rho}. \tag{10}$$

*Then it holds that*

$$\frac{\|w_t - w^*\|_2^2}{\alpha} + \frac{\|z_t - z^*\|_2^2}{\beta} \leq 2(1-\eta_\rho)^t \Delta_0 + \frac{\overline{\eta}_\rho}{\rho}, \tag{11}$$

$$\|w_t - G(z_t)\|_2^2 \leq \frac{4(1-\eta_\rho)^t \Delta_0}{\rho} + \frac{\widetilde{\eta}_\rho}{\rho^2}, \tag{12}$$

*for every iteration $t$. Above, $\Delta_0 = \mathcal{L}_\rho(w_0, z_0, \lambda_0) - \mathcal{L}_\rho(w^*, z^*, \lambda^*)$ is the initialization error, see (2). The convergence rate $1 - \eta_\rho \in (0, 1)$ and the quantities $\nu_\rho, \xi_\rho, \tau_\rho, \sigma_{0,\rho}, \overline{\eta}_\rho, \widetilde{\eta}_\rho$ above depend on the parameters in Assumptions 1-3 and on $\lambda^*$, as specified in the proof. As an example, in the regime where $\mu_L \gg \rho$ and $\iota_G^2 \gg \nu_G$, we can take*

$$\alpha \approx \frac{1}{\nu_L}, \qquad \beta \approx \frac{1}{\rho\kappa_G^2}, \qquad \frac{\rho\nu_G}{\kappa_G^2} \lesssim \sigma_0 \lesssim \rho \min\left(\frac{\mu_L^2}{\nu_L^2}, \frac{\iota_G^4}{\kappa_G^4}\right),$$

$$\eta_\rho \approx \min\left(\frac{\mu_L}{\nu_L}, \frac{\iota_G^2}{\kappa_G^2}\right), \qquad \overline{\eta}_\rho \approx \widetilde{\eta}_\rho \approx \max\left(\frac{\nu_L}{\mu_L}, \frac{\kappa_G^2}{\iota_G^2}\right). \tag{13}$$

*Above, for the sake of clarity, $\approx$ and $\lesssim$ suppress the universal constants, dependence on the initial dual $\lambda_0$ and the corresponding step size $\sigma_0$.*

A few clarifying comments about Theorem 1 are in order.

**Error:** According to Theorem 1, if the primal and dual step sizes are sufficiently small and Assumptions 1-3 are met, Algorithm 1 converges linearly to a *neighborhood* of a solution $(w^*, z^*)$. The size of this neighborhood depends on the penalty weight $\rho$ in (2). For instance, in the example in Theorem 1, it is easy to verify that this neighborhood has a radius of $O(1/\rho)$, which can be made smaller by increasing $\rho$.

Theorem 1 is however silent about the behavior of Algorithm 1 within this neighborhood. This is to be expected. Indeed, even in the simpler convex case, where $G$ in program (1) would have been an affine map, provably no first-order algorithm could converge linearly to the solution [Ouyang and Xu, 2018, Agarwal et al., 2010].

Investigating the behavior of Algorithm 1 within this neighborhood, while interesting, arguably has little practical value. For example, in the convex case, ADMM would converge slowly (sublinearly) in this neighborhood, which does not appeal to the practitioners.

As another example, when Algorithm 1 is applied in the context of statistical learning, there is no benefit in solving (1) beyond the statistical accuracy of the problem at hand [Agarwal et al., 2010], see the discussion in Supplementary A.1. As such, we defer the study of the local behavior of Algorithm 1 to a future work.

**Feasibility gap:** Likewise, according to (24) in Theorem 1, the feasibility gap of Algorithm 1 rapidly reaches a plateau. In the example in Theorem 1, the feasibility gap rapidly reaches $O(1/\rho)$, where $\rho$ is the penalty weight in (2). As before, even in the convex case, no first-order algorithm could achieve exact feasibility at linear rate [Ouyang and Xu, 2018, Agarwal et al., 2010].

**Intution:** While the exact expressions for the quantities in Theorem 1 are given in Supplementary C, the example provided in Theorem 1 highlights the simple but instructive regime where $\mu_L \gg \rho$ and $\iota_G^2 \gg \nu_G$, see Assumptions 1-3. Intuitively, $\mu_L \gg \rho$ means that minimizing the objective of (1) is prioritized over reducing the feasibility gap, see (2). In addition, $\iota_G^2 \gg \nu_G$ suggests that the generative prior $G$ is very smooth.

In this regime, the primal step size $\alpha$ for $w$ updates is determined by how smooth $L$ is, and the primal step size $\beta$ in the latent variable $z$ is determined by how smooth $G$ is, see (13). Similar restrictions are standard in first-order algorithms to avoid oscillations [Nesterov, 2013a].

As discussed earlier, the algorithm rapidly reaches a neighborhood of size $O(1/\rho)$ of a solution and the feasibility gap plateaus at $O(1/\rho)$. Note the trade-off here for the choice of $\rho$: the larger the penalty weight $\rho$ is, the more accurate Algorithm 1 would be and yet increasing $\rho$ is restricted by the assumption $\rho \ll \mu_L$. Moreover, in this example, the rate $1 - \eta_\rho$ of Algorithm 1 depends only on the regularity of $L$ and $G$ in program (1), see (13). Indeed, the more well-conditioned $L$ is and the more near-isometric $G$ is, the larger $\eta_\rho$ and the faster the convergence would be.

Generally speaking, increasing the penalty weight $\rho$ reduces the bias of Algorithm 1 at the cost of a slower rate. Beyond our work, such dependence on the geometry of the constraints has precedent in the literature of optimization [Birgin et al., 2016, Flores-Bazán et al., 2012, Cartis et al., 2018] and manifold embedding theory [Eftekhari and Wakin, 2015, 2017].

**Relation to simple gradient descent:** Consider a variant of Algorithm 1 that replaces the linearized update for $w$ in (4) with exact minimization with respect to $w$, which can be achieved with an off-the-shelf convex solver or might have a closed-form solution in some cases. The exact minimization over $w$ and Lemma 7 together guarantee that Theorem 1 also applies to this variant of Algorithm 1.

Moreover, as a special case of (1) where $R \equiv 0$ and $H \equiv 0$, this variant is closely related to GD [Bora et al., 2017], presented there without any rates. In Appendix F, we establish that the updates of both algorithms match as the feasibility gap vanishes.

In this sense, Theorem 1 provides the first rates for GD, albeit in the limit case of vanishing feasibility gap. Indeed, one key advantage of the primal-dual formulation studied in this paper is exactly this versatility in providing a family of algorithms, such as Algorithms 1 and 2, that can be tuned for various scenarios and can also efficiently handle the non-smooth case where $R$ or $H$ are nonzero in (1).

## 4 Related Work

Bora et al. [2017] empirically tune gradient descent for compressive sensing with a generative prior

$$\min_z \|A \cdot G(z) - b\|_2^2, \tag{14}$$

which is a particular case of template (1) (without splitting). They also provide a statistical generalization error dependent on a certain *set restricted isometry property* on the matrix $A$. More generally, Theorem 4 in Supplementary A provides statistical guarantees for Algorithm 1 using the standard notion of empirical Rademacher complexity [Mohri et al., 2018].

Hand and Voroninski [2017] analyze the optimization landscape of (14) under the assumption that $G$ $(i)$ is composed of linear layers and ReLU activation functions, $(ii)$ is sufficiently expansive at each layer and $(iii)$ the network's weights have a Gaussian distribution or an equivalent deterministic *weight distribution condition*. Under such conditions, they show global existence of descent directions outside small neighborhoods around two points, but do not provide algorithmic convergence rates. Their analysis requires ReLU activation in all layers of the generator $G$, including the last one, which is often not met in practice.

On the other hand, our framework is not restricted to a particular network architecture and instead isolates the necessary assumptions on the network $G$ for the success of Algorithm 1. In doing so, we effectively decouple the learning task from the network structure $G$ and study them separately in Theorem 1 and Proposition 1, respectively. In particular, our theory in Section 3 (Supplementary A) applies broadly to any nonlinear map $G$ that meets Assumptions 1-3 (Assumptions 2-5), respectively.

In turn, Proposition 1 establishes that the standard feed forward network with common differentiable activation functions almost surely meets these assumptions. In this sense, let us also point to the work of Oymak et al. [2018], which is limited to linear regression with a nonlinear constraint, with its convex analogue studied in [Agarwal et al., 2010, Giryes et al., 2016].

Heckel et al. [2019] provides a convergence proof for a modified version of gradient descent, limited to (14) and without specifying a rate. We provide the convergence rate for a broad range of learning problems, and study the statistical generalization. Hand et al. [2018] studied the *phase retrieval* problem, with a non-convex objective function that is not directly covered by (1).

For the problem (14), Shah and Hegde [2018], Hegde [2018] proposed to use Projected Gradient Descent (PGD) after splitting in a manner similar to our template (1). If the projection (onto the range of the prior $G$) is successful, and under certain additional conditions, the authors establish linear convergence of PGD to a minimizer of (14). However, the projection onto the nonlinear range of $G$ is itself a difficult non-convex program without any theoretical guarantees. In contrast, we can solve the same problem without any projections while still providing a convergence rate.

From an optimization perspective, there are no fast rates for linearized ADMM with nonlinear constraints to our knowledge, but convergence to a first-order stationary point and special cases in a few different settings have been studied [Liu et al., 2017, Shen et al., 2016, Chen and Gu, 2014, Qiao et al., 2016]. Let us again emphasize that Assumptions 2 and 3 extract the key attributes of $G$ necessary for the success of Algorithm 1, which is therefore not limited to a generator network. It is also worth noting another line of work that applies tools from statistical physics to inference with deep neural networks, see [Manoel et al., 2017, Rezende et al., 2014] and the references therein.

## 5 Experiments

In this section we evaluate our algorithms for image recovery tasks with a generative prior. The datasets we consider are the CelebA dataset of face images [Liu et al., 2015] and the MNIST dataset of handwritten digits [LeCun and Cortes, 2010]. We train a generator $G$ with ELU activation functions Clevert et al. [2015], in order to satisfy Assumption 2. The generators are trained using the Wasserstein GAN framework [Arjovsky et al., 2017]. For the CelebA dataset we downsample the images to $64 \times 64$ pixels as in Gulrajani et al. [2017] and we use the same residual architecture [He et al., 2015] for the generator with four residual blocks followed by a convolutional layer. For MNIST, we use the same architecture as one in Gulrajani et al. [2017], which contains one fully connected layer followed by three deconvolutional layers.

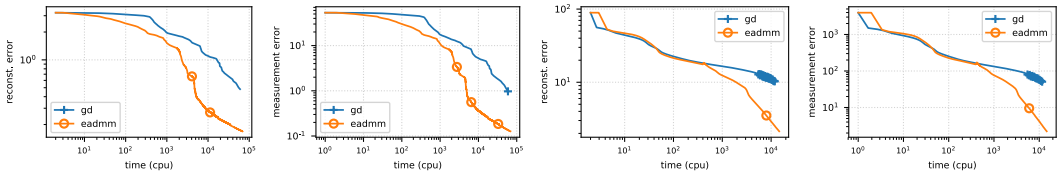


Figure 1: Reconstruction error and measurement error vs time (one tick equals the time of one GD iteration). MNIST (left) and CelebA (right).

We recover images on the range of the generator $G$, by choosing $z^\star \in \mathbb{R}^s$ and setting $w^\star := G(z^\star)$ as the true image to be recovered. This sets the global minimum of our objective functions at zero, and allows us to illustrate and compare the convergence rates of various algorithms.

Our Algorithm 1 mantains iterates $\{w_t, z_t\}_t$ where $w_t$ might not be feasible, namely, $w_t$ might not be in the range of $G$. As the goal in the following tasks is to recover an element in the range of $G$ (feasible points of (1)), we plot the objective value at the point $G(z_t)$.

**Baseline.** We compare to the most widely-used algorithm in the current literature, the gradient descent algorithm (GD) as used in [Bora et al., 2017], where a fixed number of iterations with constant step size are performed for the function $L(G(z))$. We tune its learning rate to be as large as possible without *overshooting*. (See Supplementary H for details on the hyperparameter tuning).

Our goal is to illustrate our theoretical results and highlight scenarios where Algorithm 1 can have better performance than GD in optimization problems with a generative prior. Hence, we do not compare with sparsity-prior based algorithms, such as LASSO [Tibshirani, 1996], or argue about GAN vs. sparsity priors as in Bora et al. [2017].

**Our algorithms.** We will use $(i)$ (linearized) ADMM (Algorithm 1), and $(ii)$ ADMM with exact minimization (Algorithm 2 a.k.a. EADMM), described in Section 2. For both ADMM and EADMM, we choose a starting iterate (random $z_0$ and $w_0 = G(z_0)$) and initial dual variable $\lambda_0 = 0$ (for GD we choose the same $z_0$ as initial iterate). We carefully track the objective function value vs. computation time for a fair comparison.

**Compressive sensing** The exact minimization step of EADMM involves the solution of a system of linear equations in each iteration. Performing Singular Value Decomposition (SVD) once on the measurement matrix $A$, and storing its components in memory, allows us to solve such linear systems with a very low per-iteration complexity (see Supplementary H.3). We plot the objective function value as well as the reconstruction error with $50\%$ relative measurements in Figure 1(average over 20 images (MNIST) and 10 images (CelebA)).

**Adversarial Denoising with $\ell_\infty$-norm** Projection onto the range of a deep-net prior has been considered by Samangouei et al. [2018], Ilyas et al. [2017] as a defense mechanism against adversarial examples [Szegedy et al., 2013]. In their settings, samples are denoised with a generative prior, before being fed to a classifier. Even though the adversarial noise introduced is typically bounded in $\ell_\infty$-norm, the projection is done in $\ell_2$-norm. Such projection corresponds to $F(w, z) = \|w - w^\natural\|^2$ in (1).

We instead propose to project using the $\ell_\infty$-norm that bounds the adversarial perturbation. To this end we let $F(w, z) = \gamma\|w - w^\natural\|_2^2 + \|w - w^\natural\|_\infty$ in the template (1), for some small value of $\gamma$. The proximal of the $\ell_\infty$ norm is efficiently computable [Duchi et al., 2008], allowing us to split $F(w, z)$ in its components $L(w) = \gamma\|w - w^\natural\|_2^2$ and $R(w) = \|w - w^\natural\|_\infty$ (Note that the small $\gamma$ ensures that Assumption 1 holds)

We compare the ADAM optimizer [Kingma and Ba, 2014], GD and ADMM (450 iterations and for GD and ADAM, and 300 iterations for EADMM). We use ADAM to solve the $\ell_2$ projection, while ADMM solves the $\ell_\infty$ projection. We evaluate on a test set of 2000 adversarial examples from the MNIST dataset, obtained with the Projected Gradient Algorithm of Madry et al. [2018] with 30 iterations, stepsize 0.01 and attack size 0.2. For the classifier, we use a standard convolutional network trained on clean MNIST samples. We also test ADAM, GD (3000 iterations) and EADMM (2000 iterations) on the $\ell_\infty$ denoising task.

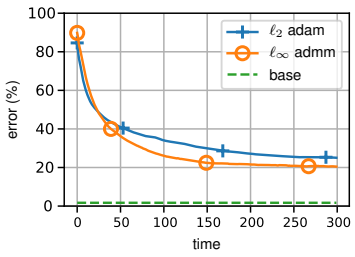


Figure 2: Test error on denoised adversarial examples vs computation time (average cpu time(s) over the sample).

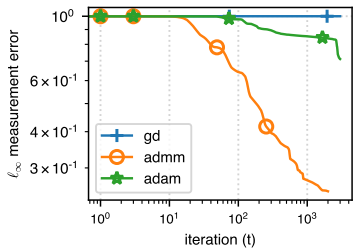


Figure 3: $\ell_\infty$ reconstruction error per iteration for ADAM, GD, and EADMM.

The test error as a function of computation time is in Figure 2. We observe that the $\ell_\infty$ denoising performs better when faced with $\ell_\infty$ bounded attacks, in the sense that it achieves a lower error with less computation time. In Figure 3, we plot the $\ell_\infty$ reconstruction error achieved by ADAM, GD and EADMM, averaged over 7 images. GD was unable to decrease the initial error, while ADAM takes a considerable number of iterations to do so. In contrast, our ADMM already achieves the final error of ADAM within its first 100 iterations.

## 6 Conclusions and Future Work

In this work, we have proposed a flexible linearized ADMM algorithm for the minimization of a convex function subject to a nonlinear constraint given by a neural network. Under mild assumptions we demonstrate a fast convergence rate to a neighborhood of a solution of its Lagrangian formulation (3) (Theorem 1). Empirical evaluation shows how it can handle non-smooth terms more efficiently when compared to gradient descent and its variants.

Some avenues of research are left open which could yield faster variants of our proposed approach. First, ADMM-type algorithms admit acceleration and restart schemes with faster convergence rates in the convex case [Goldstein et al., 2014] but their adaptation to the nonlinear constraint given by a neural network is non-trivial. Secondly, adaptivity in the choice of penalty parameter $\rho$ can potentially improve the performance of the method and reduce the need for tuning [He et al., 2000]. Finally, the denoising with $\ell_\infty$-norm shows promise as a defense against adversarial examples, and its performance on higher dimensional datasets is worth investigating.

## Acknowledgements

This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement 725594 - time-data), the Department of the Navy - Office of Naval Research (ONR) under a grant number N62909-17-1-2111, and from the Swiss National Science Foundation (SNSF) under grant number 200021_178865. FL is supported through a PhD fellowship of the Swiss Data Science Center, a joint venture between EPFL and ETH Zurich. VC acknowledges the 2019 Google Faculty Research Award.

## References

Alekh Agarwal, Sahand Negahban, and Martin J Wainwright. Fast global convergence rates of gradient methods for high-dimensional statistical recovery. In *Advances in Neural Information Processing Systems*, pages 37–45, 2010.

Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 214–223, International Convention Centre, Sydney, Australia, 06–11 Aug 2017. PMLR. URL `http://proceedings.mlr.press/v70/arjovsky17a.html`.

Dimitri P Bertsekas. On penalty and multiplier methods for constrained minimization. *SIAM Journal on Control and Optimization*, 14(2):216–235, 1976.

Ernesto G Birgin, JL Gardenghi, José Mario Martínez, SA Santos, and Ph L Toint. Evaluation complexity for nonlinear constrained optimization using unscaled kkt conditions and high-order models. *SIAM Journal on Optimization*, 26(2):951–967, 2016.

Ashish Bora, Ajil Jalal, Eric Price, and Alexandros G. Dimakis. Compressed Sensing using Generative Models. *arXiv:1703.03208 [cs, math, stat]*, March 2017. URL http://arxiv.org/abs/1703.03208. arXiv: 1703.03208.

Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, Jonathan Eckstein, et al. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine learning*, 3(1):1–122, 2011.

Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale GAN training for high fidelity natural image synthesis. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=B1xsqj09Fm.

Coralia Cartis, Nicholas IM Gould, and Ph L Toint. Optimality of orders one to three and beyond: characterization and evaluation complexity in constrained nonconvex optimization. *Journal of Complexity*, 2018.

Laming Chen and Yuantao Gu. The convergence guarantees of a non-convex approach for sparse recovery. *IEEE Transactions on Signal Processing*, 62(15):3754–3767, 2014.

Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems 29*, pages 2172–2180. Curran Associates, Inc., 2016.

Djork-Arné Clevert, Thomas Unterthiner, and Sepp Hochreiter. Fast and accurate deep network learning by exponential linear units (elus). *arXiv preprint arXiv:1511.07289*, 2015.

Constantinos Daskalakis, Andrew Ilyas, Vasilis Syrgkanis, and Haoyang Zeng. Training GANs with optimism. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=SJJySbbAZ.

Manik Dhar, Aditya Grover, and Stefano Ermon. Modeling sparse deviations for compressed sensing using generative models. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 1214–1223, Stockholmsmässan, Stockholm Sweden, 10–15 Jul 2018. PMLR. URL http://proceedings.mlr.press/v80/dhar18a.html.

Chris Donahue, Julian McAuley, and Miller Puckette. Adversarial audio synthesis. In *Proceedings of the 7th International Conference on Learning Representations*, 2019.

John Duchi, Shai Shalev-Shwartz, Yoram Singer, and Tushar Chandra. Efficient projections onto the l1-ball for learning in high dimensions. In *Proceedings of the 25th International Conference on Machine Learning*, ICML '08, pages 272–279, New York, NY, USA, 2008. ACM. ISBN 978-1-60558-205-4. doi: 10.1145/1390156.1390191. URL http://doi.acm.org/10.1145/1390156.1390191.

Charles Dugas, Yoshua Bengio, François Bélisle, Claude Nadeau, and René Garcia. Incorporating second-order functional knowledge for better option pricing. In T. K. Leen, T. G. Dietterich, and V. Tresp, editors, *Advances in Neural Information Processing Systems 13*, pages 472–478. MIT Press, 2001. URL http://papers.nips.cc/paper/1920-incorporating-second-order-functional-knowledge-for-better-option-pricing.pdf.

Armin Eftekhari and Michael B Wakin. New analysis of manifold embeddings and signal recovery from compressive measurements. *Applied and Computational Harmonic Analysis*, 39(1):67–109, 2015.

Armin Eftekhari and Michael B Wakin. What happens to a manifold under a bi-lipschitz map? *Discrete & Computational Geometry*, 57(3):641–673, 2017.

Jesse Engel, Kumar Krishna Agrawal, Shuo Chen, Ishaan Gulrajani, Chris Donahue, and Adam Roberts. GANSynth: Adversarial neural audio synthesis. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=H1xQVn09FX.

Fabián Flores-Bazán, Fernando Flores-Bazán, and Cristián Vera. A complete characterization of strong duality in nonconvex optimization with a single constraint. *Journal of Global Optimization*, 53(2):185–201, 2012.

Daniel Gabay and Bertrand Mercier. A dual algorithm for the solution of nonlinear variational problems via finite element approximation. *Computers & Mathematics with Applications*, 2:17–40, 12 1976. doi: 10.1016/0898-1221(76)90003-1.

D. J. H. Garling. *A Course in Mathematical Analysis*, volume 1. Cambridge University Press, 2014. doi: 10.1017/CBO9781139424516.

Gauthier Gidel, Reyhane Askari Hemmat, Mohammad Pezeshki, Rémi Le Priol, Gabriel Huang, Simon Lacoste-Julien, and Ioannis Mitliagkas. Negative momentum for improved game dynamics. In Kamalika Chaudhuri and Masashi Sugiyama, editors, *Proceedings of Machine Learning Research*, volume 89 of *Proceedings of Machine Learning Research*, pages 1802–1811. PMLR, 16–18 Apr 2019. URL http://proceedings.mlr.press/v89/gidel19a.html.

Raja Giryes, Yonina C Eldar, Alex M Bronstein, and Guillermo Sapiro. Tradeoffs between convergence speed and reconstruction accuracy in inverse problems. *arXiv preprint arXiv:1605.09232*, 2016.

R. Glowinski and A. Marroco. Sur l'approximation, par éléments finis d'ordre un, et la résolution, par pénalisation-dualité d'une classe de problèmes de dirichlet non linéaires. *ESAIM: Mathematical Modelling and Numerical Analysis - Modélisation Mathématique et Analyse Numérique*, 9(R2):41–76, 1975. URL http://www.numdam.org/item/M2AN_1975__9_2_41_0.

Tom Goldstein, Brendan O'Donoghue, Simon Setzer, and Richard Baraniuk. Fast alternating direction optimization methods. *SIAM Journal on Imaging Sciences*, 7(3):1588–1623, 2014.

I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative Adversarial Networks. *ArXiv e-prints*, June 2014. URL https://arxiv.org/abs/1406.2661.

Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. Improved training of wasserstein gans. In *Advances in Neural Information Processing Systems*, pages 5767–5777, 2017.

Paul Hand and Vladislav Voroninski. Global guarantees for enforcing deep generative priors by empirical risk. *arXiv preprint arXiv:1705.07576*, 2017.

Paul Hand, Oscar Leong, and Vlad Voroninski. Phase retrieval under a generative prior. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems 31*, pages 9154–9164. Curran Associates, Inc., 2018. URL http://papers.nips.cc/paper/8127-phase-retrieval-under-a-generative-prior.pdf.

BS He, Hai Yang, and SL Wang. Alternating direction method with self-adaptive penalty parameters for monotone variational inequalities. *Journal of Optimization Theory and applications*, 106(2):337–356, 2000.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. *arXiv e-prints*, art. arXiv:1512.03385, December 2015.

Reinhard Heckel, Wen Huang, Paul Hand, and Vladislav Voroninski. Deep denoising: Rate-optimal recovery of structured signals with a deep prior, 2019. URL https://openreview.net/forum?id=SklcFsAcKX.

C. Hegde. Algorithmic Aspects of Inverse Problems Using Generative Models. *ArXiv e-prints*, October 2018.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 6626–6637. Curran Associates, Inc., 2017.

Ya-Ping Hsieh, Chen Liu, and Volkan Cevher. Finding Mixed Nash Equilibria of Generative Adversarial Networks. *arXiv e-prints*, art. arXiv:1811.02002, Oct 2018.

Andrew Ilyas, Ajil Jalal, Eirini Asteri, Constantinos Daskalakis, and Alexandros G. Dimakis. The Robust Manifold Defense: Adversarial Training using Generative Models. *arXiv e-prints*, art. arXiv:1712.09196, December 2017.

Youngjin Kim, Minjung Kim, and Gunhee Kim. Memorization precedes generation: Learning unsupervised GANs with memory networks. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=rkO3uTkAZ.

Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. *arXiv e-prints*, art. arXiv:1412.6980, December 2014.

Diederik P Kingma and Max Welling. Auto-Encoding Variational Bayes. *arXiv e-prints*, art. arXiv:1312.6114, December 2013.

Abhishek Kumar, Prasanna Sattigeri, and Tom Fletcher. Semi-supervised learning with gans: Manifold invariance with improved inference. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 5534–5544. Curran Associates, Inc., 2017.

Yann LeCun and Corinna Cortes. MNIST handwritten digit database. 2010. URL http://yann.lecun.com/exdb/mnist/.

Zachary C. Lipton and Subarna Tripathi. Precise Recovery of Latent Vectors from Generative Adversarial Networks. *arXiv:1702.04782 [cs, stat]*, February 2017. URL http://arxiv.org/abs/1702.04782. arXiv: 1702.04782.

Qinghua Liu, Xinyue Shen, and Yuantao Gu. Linearized admm for non-convex non-smooth optimization with convergence analysis. *arXiv preprint arXiv:1705.02502*, 2017.

Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, 2015.

Mario Lucic, Karol Kurach, Marcin Michalski, Sylvain Gelly, and Olivier Bousquet. Are gans created equal? a large-scale study. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems 31*, pages 700–709. Curran Associates, Inc., 2018. URL http://papers.nips.cc/paper/7350-are-gans-created-equal-a-large-scale-study.pdf.

Fangchang Ma, Ulas Ayaz, and Sertac Karaman. Invertibility of convolutional generative networks from partial measurements. In *Advances in Neural Information Processing Systems*, pages 9651–9660, 2018.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=rJzIBfZAb.

Andre Manoel, Florent Krzakala, Marc Mézard, and Lenka Zdeborová. Multi-layer generalized linear estimation. In *2017 IEEE International Symposium on Information Theory (ISIT)*, pages 2098–2102. IEEE, 2017.

Lars Mescheder, Andreas Geiger, and Sebastian Nowozin. Which training methods for GANs do actually converge? In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 3481–3490, Stockholmsmässan, Stockholm Sweden, 10–15 Jul 2018. PMLR. URL http://proceedings.mlr.press/v80/mescheder18a.html.

M. Mohri, A. Rostamizadeh, A. Talwalkar, and F. Bach. *Foundations of Machine Learning*. MIT Press, 2018. ISBN 9780262039406. URL https://books.google.ch/books?id=V2B9DwAAQBAJ.

Sahand N Negahban, Pradeep Ravikumar, Martin J Wainwright, Bin Yu, et al. A unified framework for high-dimensional analysis of $m$-estimators with decomposable regularizers. *Statistical Science*, 27(4):538–557, 2012.

Y. Nesterov. *Introductory Lectures on Convex Optimization: A Basic Course*. Applied Optimization. Springer US, 2013a. ISBN 9781441988539. URL https://books.google.ch/books?id=2-ElBQAAQBAJ.

Yu Nesterov. Gradient methods for minimizing composite functions. *Mathematical Programming*, 140(1): 125–161, 2013b.

Yuyuan Ouyang and Yangyang Xu. Lower complexity bounds of first-order methods for convex-concave bilinear saddle-point problems. *arXiv preprint arXiv:1808.02901*, 2018.

Samet Oymak, Benjamin Recht, and Mahdi Soltanolkotabi. Sharp time–data tradeoffs for linear inverse problems. *IEEE Transactions on Information Theory*, 64(6):4129–4158, 2018.

Neal Parikh, Stephen Boyd, et al. Proximal algorithms. *Foundations and Trends® in Optimization*, 1(3): 127–239, 2014.

Linbo Qiao, Bofeng Zhang, Jinshu Su, and Xicheng Lu. Linearized alternating direction method of multipliers for constrained nonconvex regularized optimization. In *Asian Conference on Machine Learning*, pages 97–109, 2016.

A. Radford, L. Metz, and S. Chintala. Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks. *ArXiv e-prints*, November 2015.

Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. *arXiv preprint arXiv:1401.4082*, 2014.

Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, Xi Chen, and Xi Chen. Improved techniques for training gans. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems 29*, pages 2234–2242. Curran Associates, Inc., 2016. URL http://papers.nips.cc/paper/6125-improved-techniques-for-training-gans.pdf.

Pouya Samangouei, Maya Kabkab, and Rama Chellappa. Defense-GAN: Protecting classifiers against adversarial attacks using generative models. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=BkJ3ibb0-.

Viraj Shah and Chinmay Hegde. Solving Linear Inverse Problems Using GAN Priors: An Algorithm with Provable Guarantees. *arXiv:1802.08406 [cs, stat]*, February 2018. URL http://arxiv.org/abs/1802.08406. arXiv: 1802.08406.

Xinyue Shen, Laming Chen, Yuantao Gu, and Hing-Cheung So. Square-root lasso with nonconvex regularization: An admm approach. *IEEE Signal Processing Letters*, 23(7):934–938, 2016.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv e-prints*, art. arXiv:1312.6199, December 2013.

Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 58(1):267–288, 1996. ISSN 00359246. URL http://www.jstor.org/stable/2346178.

Subarna Tripathi, Zachary C. Lipton, and Truong Q. Nguyen. Correction by Projection: Denoising Images with Generative Adversarial Networks. *arXiv e-prints*, art. arXiv:1803.04477, March 2018.

Dmitry Ulyanov, Andrea Vedaldi, and Victor S. Lempitsky. Deep image prior. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9446–9454, 2018.

Bing Xu, Naiyan Wang, Tianqi Chen, and Mu Li. Empirical Evaluation of Rectified Activations in Convolutional Network. *arXiv e-prints*, art. arXiv:1505.00853, May 2015.

Yi Xu, Mingrui Liu, Qihang Lin, and Tianbao Yang. Admm without a fixed penalty parameter: Faster convergence with new adaptive penalization. In *Advances in Neural Information Processing Systems*, pages 1267–1277, 2017.

## A Statistical Learning with Generative Priors

So far, we have assumed $L$ to be strongly convex in (1), see Assumption 1 and Theorem 1. In this section, we relax this assumption on $L$ in the context of statistical learning with generative priors, thus extending Theorem 1 to applications such as compressive sensing. We also provide the corresponding generalization error in this section.

Here, we follow the standard setup in learning theory Mohri et al. [2018]. Consider the probability space $(\mathbb{X}, \chi)$, where $\mathbb{X} \subset \mathbb{R}^d$ is a compact set, equipped with the Borel sigma algebra, and $\chi$ is the corresponding probability measure. To learn an unknown parameter $w^\natural \in \mathbb{R}^d$, consider the optimization program

$$\min_{w \in \mathbb{R}^p} L(w), \qquad L(w) := \mathbb{E}_{x \sim \chi} l(w, x), \tag{15}$$

where $L : \mathbb{R}^p \to \mathbb{R}$ is the differentiable *population risk* and $l : \mathbb{R}^d \times \mathbb{R}^p \to \mathbb{R}$ is the corresponding *loss function*. We also assume that Program (15) has a unique solution $w^\natural \in \mathbb{R}^p$. The probability measure $\chi$ above is itself often unknown and we instead have access to $m$ samples drawn independently from $\chi$, namely, $\{x_i\}_{i=1}^m \sim \chi$. This allows us to form the *empirical loss*

$$L_m(w) := \frac{1}{m} \sum_{i=1}^m l(w, x_i). \tag{16}$$

Often, $m \ll p$ and to avoid an ill-posed problem, we must leverage any inherent structure in $w^\natural$. In this work, we consider a differentiable map $G : \mathbb{R}^s \to \mathbb{R}^d$ and we assume that $w^\natural \in G(\mathbb{R}^s)$. That is, there exists $z^\natural \in \mathbb{R}^s$ such that $w^\natural = G(z^\natural)$. While not necessary, we limit ourselves in this section to the important case where $G$ corresponds to a neural network, see Section 1.

To learn $w^\natural$ with the generative prior $w^\natural = G(z^\natural)$, we propose to solve the program

$$\begin{array}{ll} \underset{w,z}{\text{minimize}} & L_m(w) + R(w) + H(z) \\ \text{subject to} & w = G(z), \end{array} \tag{17}$$

where $R : \mathbb{R}^p \to \mathbb{R}$ and $H : \mathbb{R}^s \to \mathbb{R}$ are convex but not necessarily smooth. Depending on the specific problem at hand, the *regularizers* $R$ and $H$ allow us to impose additional structure on $w$ and $z$, such as sparsity or set inclusion. Throughout, we again require that the proximal maps [Parikh et al., 2014] for $R$ and $H$ can be computed efficiently, as detailed in Section 2.

Let us now state our assumptions, some of which differ from Section 3.

**Assumption 4. Convexity / strong smoothness of loss:** *We assume that $l(\cdot, \cdot)$ is convex in both of its arguments. Moreover, we assume that $l(w, \cdot)$ is strongly smooth, namely, there exists $\sigma_l \geq 0$ such that for every $x, x' \in \mathbb{X}$*

$$D_l(x, x'; w) \leq \frac{\sigma_l}{2} \|x - x'\|_2^2, \tag{18}$$

*where $D_l$ stands for the* Bregman divergence *associated with $l(w, \cdot)$,*

$$D_l(x, x'; w) = l(w, x') - l(w, x) - \langle x' - x, \nabla_x l(w, x) \rangle.$$

**Assumption 5. Strong convexity / smoothness of the population risk:** *We assume that the population risk $L$ defined as*

$$L(w) := \mathbb{E}_{x \sim \chi} l(w, x), \tag{19}$$

*is both strongly convex and smooth, i.e., there exist $0 < \zeta_L \leq \sigma_L$ such that*

$$\frac{\zeta_L}{2} \|w - w'\|^2 \leq D_L(w, w') \leq \frac{\sigma_L}{2} \|w - w'\|^2,$$

$$D_L(w, w') = L(w') - L(w) - \langle w' - w, \nabla L(w) \rangle, \tag{20}$$

*for every $w, w' \in \mathbb{R}^d$. In the following we denote by $w^\natural$ the minimizer of (19). In view of our assumption, such minimizer is unique.*

Assumptions 4 and 5 are standard in statistical learning Mohri et al. [2018]. For example, in linear regression, we might take

$$l(w,x) = \frac{1}{2}|\langle w - w^\natural, x\rangle|^2,$$
$$L_m(w) = \frac{1}{2m}\sum_{i=1}^m |\langle w - w^\natural, x_i\rangle|^2,$$

for which both Assumptions 4 and 5 are met. Lastly, we require that the Assumptions 2 and 3 on $G$ hold in this section, see and Proposition 1 for when these assumptions hold for generative priors.

As a consequence of Assumption 4, we have that $L_m$ is convex. We additionally require $L_m$ to be strongly convex and smooth in the following restricted sense. Even though $L_m$ is random because of its dependence on the random training data $\{x_i\}_{i=1}^m$, we ensure later in this section that the next condition is indeed met with high probability when $m$ is large enough.

**Definition 1. Restricted strong convexity / smoothness of empirical loss:** *We say that $L_m$ is strongly convex and smooth on the set $W \subset \mathbb{R}^p$ if there exist $0 < \mu_L \le \nu_L$ and $\overline{\mu}_L, \overline{\nu}_L \ge 0$ such that*

$$D_{L_m}(w,w') \ge \frac{\mu_L}{2}\|w' - w\|_2^2 - \overline{\mu}_L,$$
$$D_{L_m}(w,w') \le \frac{\nu_L}{2}\|w' - w\|_2^2 + \overline{\nu}_L, \tag{21}$$

$$D_{L_m}(w,w') := L_m(w') - L_m(w) - \langle w' - w, \nabla L_m(w)\rangle,$$

*for every $w, w' \in W$.*

Under the above assumptions, a result similar to Theorem 1 holds, which we state without proof.

**Theorem 2. (guarantees for Algorithm 1)** *Suppose that Assumptions 2-5 hold. Let $(w^*, z^*)$ be a solution of program (1) and let $\lambda^*$ be a corresponding optimal dual variable. Let also $\{w_t, z_t, \lambda_t\}_{t\ge 0}$ denote the output sequence of Algorithm 1. Suppose that $L_m$ satisfies the restricted strong convexity and smoothness in Definition 1 for a set $W \subset \mathbb{R}^p$ that contains a solution $w^*$ of (1) and all the iterates $\{w_t\}_{t\ge 0}$ of Algorithm 1.[1] Suppose also that the primal step sizes $\alpha, \beta$ in Algorithm 1 satisfy*

$$\alpha \le \frac{1}{\nu_\rho}, \qquad \beta \le \frac{1}{\xi_\rho + 2\alpha\tau_\rho^2}. \qquad \sigma_0 \le \sigma_{0,\rho}, \tag{22}$$

*Then it holds that*

$$\frac{\|w_t - w^*\|_2^2}{\alpha} + \frac{\|z_t - z^*\|_2^2}{\beta} \le 2(1-\eta_\rho)^t\Delta_0 + \frac{\overline{\eta}_\rho}{\rho}, \tag{23}$$

$$\|w_t - G(z_t)\|_2^2 \le \frac{4(1-\eta_\rho)^t\Delta_0}{\rho} + \frac{\widetilde{\eta}_\rho}{\rho^2}, \tag{24}$$

*for every iteration $t$. Above, $\Delta_0 = \mathcal{L}_\rho(w_0, z_0, \lambda_0) - \mathcal{L}_\rho(w^*, z^*, \lambda^*)$ is the initialization error, see (2). The convergence rate $1 - \eta_\rho \in (0,1)$ and the quantities $\nu_\rho, \xi_\rho, \tau_\rho, \sigma_{0,\rho}, \overline{\eta}_\rho, \widetilde{\eta}_\rho$ above depend on the parameters in the Assumptions 2-5 and on $\lambda_0, \sigma_0$.*

The remarks after Theorem 1 apply here too.

### A.1 Generalization Error

Building upon the optimization guarantee in Theorem 4, our next result in this section is Theorem 4, which quantifies the convergence of the iterates $\{w_t\}_{t\ge 0}$ of Algorithm 1 to the true parameter $w^\natural$.

In other words, Theorem 4 below controls the generalization error of (1), namely, the error incurred by using the empirical risk $L_m$ in lieu of the population risk $L$. Indeed, Theorem 1 is silent about $\|w_t - w^\natural\|_2$. We address this shortcoming with the following result, proved in Section G of the supplementary material.

[1] If necessary, the inclusion $\{w_t\}_{t\ge 0} \subset W$ might be enforced by adding the indicator function of the convex hull of $W$ to $R$ in (1), similar to Agarwal et al. [2010].

**Lemma 3.** *Let $R = 1_W$ be the indicator function on $W \subset \mathbb{R}^p$ and set $H = 0$ in (1).*[2] *Suppose that $w^*$ belongs to the relative interior of $W$. Then it holds that*

$$\|w^\natural - w^*\|_2 \leq \frac{1}{\zeta_L} \max_{w \in W} \|\nabla L_m(w) - \nabla L(w)\|_2. \tag{25}$$

Before bounding the right-hand side of (25), we remark that it is possible to extend Lemma 3 to the case where the regularizer $R$ is a *decomposable* norm, along the lines of Negahban et al. [2012]. We will however not pursue this direction in the present work. Next note that (23) and Lemma 3 together imply that

$$\begin{aligned}
\frac{\|w_t - w^\natural\|_2^2}{\alpha^2} &\leq \left( \frac{\|w_t - w^*\|_2}{\alpha} + \frac{\|w^* - w^\natural\|_2}{\beta} \right)^2 \qquad \text{(triangle inequality)} \\
&\leq \frac{2\|w_t - w^*\|_2^2}{\alpha^2} + \frac{2\|w^* - w^\natural\|_2^2}{\beta^2} \quad ((a+b)^2 \leq 2a^2 + 2b^2) \\
&\leq 4(1 - \eta_\rho)^t \Delta_0 + \frac{2\overline{\eta}_\rho}{\rho} + \frac{2}{\zeta_L^2} \max_{w \in W} \|\nabla L_m(w) - \nabla L(w)\|_2^2.
\end{aligned} \tag{26}$$

According to Theorem 1, the right-hand side of (26) depends on $\mu_L, \overline{\mu}_L, \nu_L, \overline{\nu}_L$, which were introduced in Definition 1. Note that $\mu_L, \overline{\mu}_L, \nu_L, \overline{\nu}_L$ and the right-hand side of (25) are all random variables because they depend on $L_m$ and thus on the randomly drawn training data $\{x_i\}_{i=1}^m$. To address this issue, we apply a basic result in statistical learning theory as follows. For every $w \in \mathbb{R}^p$ and every pair $x, x' \in \mathbb{X}$, we use Assumption 4 to write that

$$\begin{aligned}
\|\nabla l(w, x) - \nabla l(w, x')\|_2 &\leq \sigma_l \|x - x'\|_2 \qquad \text{(see (18))} \\
&\leq \sigma_l \text{diam}(\mathbb{X}),
\end{aligned} \tag{27}$$

where $\text{diam}(\mathbb{X})$ denotes the diameter of the compact set $\mathbb{X}$. Note also that

$$\mathbb{E}_{\{x_i\}_i} \nabla L_m(w) = \nabla L(w), \qquad \forall w \in W, \tag{28}$$

where the expectation is over the training data $\{x_i\}_i$. Then, for $\varepsilon > 0$ and except with a probability of at most $e^{-\varepsilon}$, it holds that

$$\begin{aligned}
&\|\nabla L_m(w) - \nabla L(w)\|_2 \\
&\leq 2\mathcal{R}_W(x_1, \cdots, x_m) + 3\sigma_l \text{diam}(\mathbb{X}) \sqrt{\frac{\varepsilon + 2}{2m}} \\
&=: \Upsilon_{m,W}(\varepsilon),
\end{aligned} \tag{29}$$

for every $w \in W$ [Mohri et al., 2018]. Above,

$$\mathcal{R}_W(x_1, \cdots, x_m) = \mathbb{E}_E \left[ \max_{w \in W} \left\| \frac{1}{m} \sum_{i=1}^m e_i \nabla_w l(w, x_i) \right\|_2 \right], \tag{30}$$

is the *empirical Rademacher complexity* and $E = \{e_i\}_i$ is a Rademacher sequence, namely, a sequence of independent random variables taking $\pm 1$ with equal probabilities. We can now revisit (26) and write that

$$\|w_t - w^\natural\|_2^2 \leq 4\alpha^2 (1 - \eta_\rho)^t \Delta_0 + \frac{2\alpha^2 \overline{\eta}_\rho}{\rho} + \frac{2\alpha^2 \Upsilon_{m,W}^2(\varepsilon)}{\zeta_L^2}, \tag{31}$$

which holds except with a probability of at most $e^{-\varepsilon}$. In addition, for every $w, w' \in W$, we may write that

$$\begin{aligned}
&\|\nabla L_m(w) - \nabla L_m(w')\|_2 \\
&\leq \|\nabla L(w) - \nabla L(w')\|_2 + \|\nabla L_m(w) - \nabla L(w)\|_2 \\
&\qquad + \|\nabla L_m(w') - \nabla L(w')\|_2 \quad \text{(triangle inequality)} \\
&\leq \sigma_L \|w - w'\|_2 + 2\Upsilon_{m,W}(\varepsilon), \qquad \text{(see (20),(29))}
\end{aligned} \tag{32}$$

[2] To be complete, $1_W(w) = 0$ if $w \in W$ and $1_W(w) = \infty$ otherwise.

except with a probability of at most $e^{-\varepsilon}$. Likewise, for every $w, w' \in W$, we have that

$$\begin{aligned}
&\|\nabla L_m(w) - \nabla L_m(w')\|_2 \\
&\geq \|\nabla L_m(w) - \nabla L_m(w)\|_2 - \|\nabla L_m(w) - \nabla L(w)\|_2 \\
&\qquad - \|\nabla L_m(w') - \nabla L(w')\|_2 \qquad \text{(triangle inequality)} \\
&\geq \zeta_L \|w - w'\|_2 - 2\Upsilon_{m,W}(\varepsilon), \qquad \text{(see (20,29))}
\end{aligned} \tag{33}$$

except with a probability of at most $e^{-\varepsilon}$. Therefore, $L_m$ satisfies the restricted strong convexity and smoothness in Definition 1 with

$$\begin{aligned}
\mu_L = \sigma_L, \qquad \nu_L = \zeta_L, \\
\overline{\mu}_L = \overline{\zeta}_L = 2\Upsilon_{m,W}(\varepsilon).
\end{aligned} \tag{34}$$

Our findings in this section are summarized below.

**Theorem 4. (generalization error)** *Suppose that Assumptions 2-5 hold and recall that the training samples $\{x_i\}_{i=1}^m$ are drawn independently from the probability space $(\mathbb{X}, \chi)$ for a compact set $\mathbb{X} \subset \mathbb{R}^d$ with diameter $\operatorname{diam}(\mathbb{X})$.*

*For a set $W \subset \mathbb{R}^p$, let $R = 1_W$ be the indicator function on $W$, and set $H \equiv 0$ in (1). Suppose that solution $w^*$ of (1) belongs to the relative interior of $W$. For $\varepsilon > 0$, evaluate the quantities in Theorem 2 with*

$$\begin{aligned}
\mu_L = \sigma_L, \qquad \nu_L = \zeta_L, \\
\overline{\mu}_L = \overline{\zeta}_L = 4\mathcal{R}_W(x_1, \cdots, x_m) \\
+ 6\sigma_l \operatorname{diam}(\mathbb{X}) \sqrt{\frac{\varepsilon + 2}{2m}},
\end{aligned} \tag{35}$$

*where $\mathcal{R}_W(x_1, \cdots, x_m)$ in the empirical Rademacher complexity defined in (30). If the requirements on the step sizes in (22) hold, we then have that*

$$\begin{aligned}
\|w_t - w^\natural\|_2^2 \leq 4\alpha^2(1 - \eta_\rho)^t \Delta_0 + \frac{2\alpha^2 \overline{\eta}_\rho}{\rho} + \frac{8\alpha^2}{\zeta_L^2} \mathcal{R}_W(x_1, \cdots, x_m)^2 \\
+ \frac{18\alpha^2 \sigma_l^2 \operatorname{diam}(\mathbb{X})^2 (\varepsilon + 2)}{m},
\end{aligned} \tag{36}$$

*except with a probability of at most $e^{-\varepsilon}$.*

Most of the remarks about Theorem 1 also apply to Theorem 4 and we note that $\|w_t - w^\natural\|_2$ reduces by increasing the number of training samples $m$, before asymptotically reaching the generalization error

$$2\psi_\rho + \frac{8}{\zeta_L^2} \mathcal{R}_W(x_1, \cdots, x_m)^2. \tag{37}$$

Computing the Rademacher complexity above for specific choices of the network structure and loss is itself potentially a complicated task, which we will not pursue by the virtue of the generality of our results so far. The key technical challenge there is computing the corresponding *entropy integral*, which involves estimating the *covering numbers* of the set $W$ Mohri et al. [2018]. One last takeaway point from the statistical accuracy in (37) is the following. If

$$\overline{\eta}_\rho = O(\rho \cdot \mathcal{R}_W(x_1, \cdots, x_m)^2 / \zeta_L^2), \tag{38}$$

the asymptotic optimization error in Theorem 1 does not play an important role in determining the generalization error above. In words, if (38) holds, then Algorithm 1 converges to the ball of statistical accuracy around $w^\natural$. Here, $O$ stands for the standard Big-O notation.

## B Proof of Proposition 1

The feedforward network $G = G_\Xi : \mathbb{R}^s \to \mathbb{R}^d$ is a composition of linear maps and entry-wise applications of the activation functions, and hence is also of class $C^1$. Its Jacobian $DG : \mathbb{R}^s \to \mathbb{R}^{d \times s}$

is thus a continuous function and its restriction to the compact subset $\mathcal{D} \subseteq \mathbb{R}^s$ is Lipschitz-continuous. Therefore, there exists $\nu_G \geq 0$ such that

$$\|DG(z') - DG(z)\|_2 \leq \nu_G \|z' - z\|, \qquad \forall z, z' \in \mathcal{D}.$$

From standard arguments it then follows that Assumption 2 holds in the sense that

$$\begin{aligned}
\|G(z') - G(z) - DG(z)(z' - z)\|_2 &= \left\| \int_0^1 (DG(tz' + (1-t)z) - DG(z))(z' - z) dt \right\|_2 \\
&\leq \int_0^1 \|DG(tz' + (1-t)z) - DG(z)\|_2 \|z' - z\|_2 dt \\
&\leq \nu_G \int_0^1 t \|z' - z\|^2 dt = \frac{\nu_G}{2} \|z' - z\|_2^2,
\end{aligned}$$

for every $z, z' \in \mathbb{R}^s$.

In order to show that Assumption 3 (near-isometry) also holds, we will require the following simple fact:

**Lemma 5.** *Let $G : \mathcal{D} \subseteq \mathbb{R}^s \to \mathbb{R}^d$ have a left inverse $H : G(\mathcal{D}) \subseteq \mathbb{R}^d \to \mathbb{R}^s$ which is Lipschitz-continuous with constant $\iota_G > 0$. Then it holds that*

$$\frac{1}{\iota_G} \|z' - z\| \leq \|G(z') - G(z)\|, \qquad \forall z', z \in D.$$

*Proof.*

$$\|z' - z\| = \|H(G(z')) - H(G(z))\| \leq \iota_G \|G(z') - G(z)\|.$$

□

We now proceed to show that Assumption 3 holds. We suppose $G$ is of the form

$$G(z) = \omega_k W_k (\omega_{k-1} W_{k-1} \dots (\omega_1 W_1 z) \dots),$$

for weight matrices $\{W_k\}_k$. First note that, by the compactness of the domain of $G$, the values of the hidden layers are always contained in a product of compact intervals, and so we can replace $\omega_i$ by its restriction to such sets. Each $\omega_i$ is continuous, defined on a product of intervals, and is stricly increasing so that they have a continuous left inverse $\omega_i^{-1}$ [Garling, 2014, Proposition 6.4.5]. The assumption of non-decreasing layer sizes implies that the $W_i$ are tall matrices of dimensions $(m_i, n_i)$ with $m_i \geq n_i$, whose columns are almost surely linearly independent after an arbitrarily small perturbation. In such case they have a left matrix inverse $W_i^{-1}$, which as a bounded linear map, is continuous. It then follows that $G$ has a continuous left inverse of the form

$$G^{-1} = W_1^{-1} \circ \omega_1^{-1} \dots W_k^{-1} \circ \omega_k^{-1},$$

which is a continuous mapping and is defined on $G(\mathcal{D})$ which by continuity of $G$ is compact, hence $G^{-1}$ is Lipschitz-continuous. The result then follows by the Lipschitz continuity of the map $G$ (restricted to the compact domain $\mathcal{D}$) and Lemma 5.

## C Proof of Theorem 1

It is convenient throughout the supplementary material to use a slightly different notation for Lagrangian, compared to the body of the paper. To improve the readability of the proof, let us list here the assumptions on the empirical loss $L$ and prior $G$ that are used throughout this proof. For every iteration $t$, we assume that

$$\begin{aligned}
&L(w_t) - L(w^*) - \langle w_t - w^*, \nabla L(w^*) \rangle \\
&\geq \frac{\mu_L}{2} \|w_t - w^*\|_2^2, \qquad \text{(strong convexity of } L\text{)}
\end{aligned} \tag{39}$$

$$\begin{aligned}
&L(w_{t+1}) - L(w_t) - \langle w_{t+1} - w_t, \nabla L(w_t) \rangle \\
&\leq \frac{\nu_L}{2} \|w_{t+1} - w_t\|_2^2, \qquad \text{(strong smoothness of } L\text{)}
\end{aligned} \tag{40}$$

$$\begin{aligned}&\|G(z')-G(z)-DG(z)\cdot(z'-z)\|_2\\&\leq\frac{\nu_G}{2}\|z'-z\|_2^2,\qquad\text{(strong smoothness of } G)\end{aligned}\tag{41}$$

$$\iota_G\|z'-z\|_2\leq\|G(z')-G(z)\|_2\leq\kappa_G\|z'-z\|_2,\qquad\text{(near-isometry of } G)\tag{42}$$

$$\|DG(z)\cdot(z'-z)\|_2\leq\kappa_G\|z'-z\|_2,\qquad\text{(Lipschitz continuty of } G)\tag{43}$$

For the sake of brevity, let us set

$$v=(w,z)\in\mathbb{R}^{p+s},$$

$$\begin{aligned}\mathcal{L}_\rho(v,\lambda):=\mathcal{L}_\rho(w,z,\lambda):=L(w)+R(w)+H(z)+\langle w-G(z),\lambda\rangle\\+\frac{\rho}{2}\|w-G(z)\|_2^2,\qquad\text{(augmented Lagrangian)}\end{aligned}\tag{44}$$

$$\mathcal{L}'_\rho(v,\lambda):=\mathcal{L}'_\rho(w,z,\lambda)=L(w)+\langle w-G(z),\lambda\rangle+\frac{\rho}{2}\|w-G(z)\|_2^2,\tag{45}$$

$$A(v)=A(w,z):=w-G(z).\qquad\text{(feasibility gap)}\tag{46}$$

Let also $v^*=(w^*,z^*)$ be a solution of (1) and let $\lambda^*$ be a corresponding optimal dual variable. The first-order necessary optimality conditions for (1) are

$$\begin{cases}-\nabla_v\mathcal{L}'_\rho(v^*,\lambda^*)\in\partial R(w^*)\times\partial H(z^*),\\w^*=G(z^*),\end{cases}\tag{47}$$

where $\partial R(w^*)$ and $\partial H(z^*)$ are the subdifferentials of $R$ and $H$, respectively, at $w^*$ and $z^*$. Throughout the proof, we will also often use the notation

$$\Delta_t:=\mathcal{L}_\rho(v_t,\lambda_t)-\mathcal{L}_\rho(v^*,\lambda^*),\tag{48}$$

$$\Delta'_t:=\mathcal{L}'_\rho(v_t,\lambda_t)-\mathcal{L}'_\rho(v^*,\lambda^*),\tag{49}$$

$$\delta_t:=\|w_t-w^*\|_2,\qquad\delta'_t:=\|z_t-z^*\|_2,\tag{50}$$

$$A_t:=A(v_t)=w_t-G(z_t).\tag{51}$$

In particular, with this new notation, the dual update can be rewritten as

$$\lambda_{t+1}=\lambda_t+\sigma_{t+1}A_{t+1}.\qquad\text{(see Algorithm 1)}\tag{52}$$

First, in Appendix D, we control the smoothness of $\mathcal{L}'_\rho$ over the trajectory of the algorithm.

**Lemma 6.** *For every iteration $t$, it holds that*

$$\begin{aligned}&\mathcal{L}'_\rho(w_{t+1},z_{t+1},\lambda_t)-\mathcal{L}'_\rho(w_t,z_{t+1},\lambda_t)-\langle w_{t+1}-w_t,\nabla_w\mathcal{L}'_\rho(w_t,z_{t+1},\lambda_t)\rangle\\&\leq\frac{\nu_\rho}{2}\|w_{t+1}-w_t\|_2^2,\end{aligned}\tag{53}$$

$$\begin{aligned}&\mathcal{L}'_\rho(w_t,z_{t+1}\lambda_t)-\mathcal{L}'_\rho(w_t,z_t,\lambda_t)-\langle z_{t+1}-z_t,\nabla_z\mathcal{L}'_\rho(w_t,z_t,\lambda_t)\rangle\\&\leq\frac{\xi_\rho}{2}\|z_{t+1}-z_t\|_2^2,\end{aligned}\tag{54}$$

$$\|\nabla_w\mathcal{L}'_\rho(w_t,z_{t+1},\lambda_t)-\nabla_w\mathcal{L}'_\rho(w_t,z_t,\lambda_t)\|_2\leq\tau_\rho\|z_{t+1}-z_t\|_2^2,\tag{55}$$

*where*

$$\nu_\rho:=\nu_L+\rho.\tag{56}$$

$$\xi_\rho:=\nu_G(\lambda_{\max}+\rho\max_i\|A_i\|_2)+2\rho\kappa_G^2,\tag{57}$$

$$\tau_\rho:=\rho\kappa_G.\tag{58}$$

Second, in the following result we ensure that $\mathcal{L}_\rho$ and $\mathcal{L}'_\rho$ are sufficiently regular along the trajectory of our algorithm, see Appendix E for the proof.

**Lemma 7.** *For every iteration $t$, it holds that*

$$\Delta_t \geq \frac{\mu_\rho \delta_t^2}{2} + \frac{\mu'_\rho \delta_t'^2}{2} - \overline{\mu}_\rho, \tag{59}$$

$$\Delta'_t + \langle v^* - v_t, \nabla_v \mathcal{L}'_\rho(v_t)\rangle \leq \frac{\omega_\rho \delta_t^2}{2} + \frac{\omega'_\rho \delta_t'^2}{2}, \tag{60}$$

*where*

$$\mu_\rho := \mu_L - 2\rho, \qquad \mu'_\rho := \frac{\rho \iota_G^2}{2} - \nu_G \|\lambda^*\|_2, \tag{61}$$

$$\overline{\mu}_\rho := \frac{3}{\rho}\left(\lambda_{\max}^2 + \|\lambda^*\|_2^2\right), \tag{62}$$

$$\omega_\rho := 0, \qquad \omega'_\rho := \frac{\nu_G}{2}\left(\lambda_{\max} + \rho\right). \tag{63}$$

Having listed all the necessary technical lemmas above, we now proceed to prove Theorem 1. Using the smoothness of $\mathcal{L}'_\rho$, established in Lemma 6, we argue that

$$\begin{aligned}
&\mathcal{L}'_\rho(v_{t+1}, \lambda_{t+1}) \\
&= L(w_{t+1}) + \langle A_{t+1}, \lambda_{t+1}\rangle + \frac{\rho}{2}\|A_{t+1}\|_2^2 \qquad \text{(see (45))} \\
&= L(w_{t+1}) + \langle A_{t+1}, \lambda_t\rangle + \left(\frac{\rho}{2} + \sigma_{t+1}\right)\|A_{t+1}\|_2^2 \qquad \text{(see (52))} \\
&= \mathcal{L}'_\rho(w_{t+1}, z_{t+1}, \lambda_t) + \sigma_{t+1}\|A_{t+1}\|_2^2 \qquad \text{(see (44))} \\
&\leq \mathcal{L}'_\rho(w_t, z_{t+1}, \lambda_t) + \langle w_{t+1} - w_t, \nabla_w \mathcal{L}'_\rho(w_t, z_{t+1}, \lambda_t)\rangle + \frac{\nu_\rho}{2}\|w_{t+1} - w_t\|_2^2 \\
&\quad + \overline{\nu}_\rho + \sigma_{t+1}\|A_{t+1}\|_2^2 \qquad \text{(see (53))} \\
&\leq \mathcal{L}'_\rho(w_t, z_{t+1}, \lambda_t) + \langle w_{t+1} - w_t, \nabla_w \mathcal{L}'_\rho(w_t, z_{t+1}, \lambda_t)\rangle + \frac{1}{2\alpha}\|w_{t+1} - w_t\|_2^2 \\
&\quad + \overline{\nu}_\rho + \sigma_{t+1}\|A_{t+1}\|_2^2,
\end{aligned} \tag{64}$$

where the last line above holds if the step size $\alpha$ satisfies

$$\alpha \leq \frac{1}{\nu_\rho}. \tag{65}$$

According to Algorithm 1, we can equivalently write the $w$ updates as

$$w_{t+1} = \arg\min_w \left\langle w - w_t, \nabla_w \mathcal{L}'_\rho(w_t, z_{t+1}, \lambda_t)\right\rangle + \frac{1}{2\alpha}\|w - w_t\|_2^2 + R(w). \tag{66}$$

In particular, consider above the choice of $w = \theta w^* + (1-\theta)w_t$ for $\theta \in [0,1]$ to be set later. We can then bound the last line of (64) as

$$\begin{aligned}
&\mathcal{L}'_\rho(v_{t+1}, \lambda_{t+1}) + R(w_{t+1}) \\
&= \mathcal{L}'_\rho(w_t, z_{t+1}, \lambda_t) + \min_w \langle w - w_t, \nabla_w \mathcal{L}'_\rho(w_t, z_{t+1}, \lambda_t)\rangle \\
&\quad + \frac{1}{2\alpha}\|w - w_t\|_2^2 + R(w) + \sigma_{t+1}\|A_{t+1}\|_2^2 \qquad \text{(see (64, 66))} \\
&\leq \mathcal{L}'_\rho(w_t, z_{t+1}, \lambda_t) + \theta\langle w^* - w_t, \nabla_w \mathcal{L}'_\rho(w_t, z_{t+1}, \lambda_t)\rangle + \frac{\theta^2 \delta_t^2}{2\alpha} \\
&\quad + \theta R(w^*) + (1-\theta)R(w_t) + \sigma_{t+1}\|A_{t+1}\|_2^2 \qquad \text{(convexity of } R) \\
&= \mathcal{L}'_\rho(w_t, z_{t+1}, \lambda_t) + \theta\langle w^* - w_t, \nabla_w \mathcal{L}'_\rho(w_t, z_t, \lambda_t)\rangle + \frac{\theta^2 \delta_t^2}{2\alpha} \\
&\quad + \theta\langle w^* - w_t, \nabla_w \mathcal{L}'_\rho(w_t, z_{t+1}, \lambda_t) - \nabla_w \mathcal{L}'_\rho(w_t, z_t, \lambda_t)\rangle \\
&\quad + \theta R(w^*) + (1-\theta)R(w_t) + \sigma_{t+1}\|A_{t+1}\|_2^2.
\end{aligned} \tag{67}$$

The last inner product above can be controlled as

$$\begin{aligned}
&\theta\langle w^* - w_t, \nabla_w \mathcal{L}'_\rho(w_t, z_{t+1}, \lambda_t) - \nabla_w \mathcal{L}'_\rho(w_t, z_t, \lambda_t)\rangle \\
&\leq \frac{\theta^2 \delta_t^2}{2\alpha} + \frac{\alpha}{2}\|\nabla_w \mathcal{L}'_\rho(w_t, z_{t+1}, \lambda_t) - \nabla_w \mathcal{L}'_\rho(w_t, z_t, \lambda_t)\|_2^2 \qquad (2\langle a, b\rangle \leq \|a\|_2^2 + \|b\|_2^2 \text{ and (50)}) \\
&\leq \frac{\theta^2 \delta_t^2}{2\alpha} + \alpha\tau_\rho^2 \|z_{t+1} - z_t\|_2^2, \qquad \text{(see (55))}
\end{aligned} \tag{68}$$

which, after substituting in (67), yields that

$$\begin{aligned}
&\mathcal{L}'_\rho(v_{t+1}, \lambda_{t+1}) + R(w_{t+1}) \\
&\leq \mathcal{L}'_\rho(w_t, z_{t+1}, \lambda_t) + \theta\langle w^* - w_t, \nabla_w \mathcal{L}'_\rho(w_t, z_t, \lambda_t)\rangle + \frac{\theta^2 \delta_t^2}{\alpha} \\
&\quad + \alpha\tau_\rho^2 \|z_{t+1} - z_t\|_2^2 + \theta R(w^*) + (1-\theta) R(w_t) + \sigma_{t+1}\|A_{t+1}\|_2^2.
\end{aligned} \tag{69}$$

Regarding the right-hand side above, the smoothness of $\mathcal{L}'_\rho$ in Lemma 6 allows us to write that

$$\begin{aligned}
&\mathcal{L}'_\rho(w_t, z_{t+1}, \lambda_t) + \alpha\tau_\rho^2 \|z_{t+1} - z_t\|_2^2 \\
&\leq \mathcal{L}'_\rho(w_t, z_t, \lambda_t) + \langle z_{t+1} - z_t, \nabla_z \mathcal{L}'_\rho(w_t, z_t, \lambda_t)\rangle \\
&\quad + \left(\frac{\xi_\rho}{2} + \alpha\tau_\rho^2\right)\|z_{t+1} - z_t\|_2^2. \qquad \text{(see (54))}
\end{aligned} \tag{70}$$

If we assume that the primal step sizes $\alpha, \beta$ satisfy

$$\frac{\xi_\rho}{2} + \alpha\tau_\rho^2 \leq \frac{1}{2\beta}, \tag{71}$$

we can simplify (70) as

$$\begin{aligned}
&\mathcal{L}'_\rho(w_t, z_{t+1}, \lambda_t) + \alpha\tau_\rho^2 \|z_{t+1} - z_t\|_2^2 \\
&\leq \mathcal{L}'_\rho(w_t, z_t, \lambda_t) + \langle z_{t+1} - z_t, \nabla_z \mathcal{L}'_\rho(w_t, z_t, \lambda_t)\rangle + \frac{1}{2\beta}\|z_{t+1} - z_t\|_2^2. \qquad \text{(see (71))}
\end{aligned} \tag{72}$$

From Algorithm 1, recall the equivalent expression of the $z$ updates as

$$z_{t+1} = \arg\min_z \langle z - z_t, \nabla_z \mathcal{L}'_\rho(w_t, z_t, \lambda_t)\rangle + \frac{1}{2\beta}\|z - z_t\|_2^2 + H(z), \tag{73}$$

and consider the choice of $z = \theta z^* + (1-\theta) z_t$ above, with $\theta \in [0, 1]$ to be set later. Combining (72, 73) leads us to

$$\begin{aligned}
&\mathcal{L}'_\rho(w_t, z_{t+1}, \lambda_t) + \alpha\tau_\rho^2 \|z_{t+1} - z_t\|_2^2 + H(z_{t+1}) \\
&= \mathcal{L}'_\rho(w_t, z_t, \lambda_t) + \min_z \langle z - z_t, \nabla_z \mathcal{L}'_\rho(w_t, z_t, \lambda_t)\rangle + \frac{1}{2\beta}\|z - z_t\|_2^2 + H(z) \qquad \text{(see (72, 73))} \\
&\leq \mathcal{L}'_\rho(w_t, z_t, \lambda_t) + \theta\langle z^* - z_t, \nabla_z \mathcal{L}'_\rho(w_t, z_t, \lambda_t)\rangle + \frac{\theta^2 \delta_t'^2}{2\beta} + H(\theta z^* + (1-\theta) z_t) \\
&\leq \mathcal{L}'_\rho(w_t, z_t, \lambda_t) + \theta\langle z^* - z_t, \nabla_z \mathcal{L}'_\rho(w_t, z_t, \lambda_t)\rangle + \frac{\theta^2 \delta_t'^2}{2\beta} \\
&\quad + \theta H(z^*) + (1-\theta) H(z_t). \qquad \text{(convexity of } H\text{)}
\end{aligned} \tag{74}$$

By combining (69,74), we reach

$$
\begin{aligned}
&\mathcal{L}_\rho(v_{t+1},\lambda_{t+1})\\
&=\mathcal{L}'_\rho(v_{t+1},\lambda_{t+1})+R(w_{t+1})+H(z_{t+1}) \qquad \text{(see (44,45))}\\
&\le \mathcal{L}'_\rho(w_t,z_{t+1},\lambda_t)+\theta\langle w^*-w_t,\nabla_w\mathcal{L}'_\rho(w_t,z_t,\lambda_t)\rangle+\frac{\theta^2\delta_t^2}{\alpha}+\alpha\tau_\rho^2\|z_{t+1}-z_t\|_2^2\\
&\quad+\theta R(w^*)+(1-\theta)R(w_t)+H(z_{t+1})+\sigma_{t+1}\|A_{t+1}\|_2^2 \qquad \text{(see (69))}\\
&\le \mathcal{L}'_\rho(v_t,\lambda_t)+\theta\langle v^*-v_t,\nabla_z\mathcal{L}'_\rho(v_t,\lambda_t)\rangle+\frac{\theta^2\delta_t^2}{\alpha}+\frac{\theta^2\delta_t'^2}{2\beta}\\
&\quad+\theta R(z^*)+(1-\theta)R(z_t)+\theta H(z^*)+(1-\theta)H(z_t)\\
&\quad+\sigma_{t+1}\|A_{t+1}\|_2^2 \qquad \text{(see (74))}\\
&=\mathcal{L}_\rho(v_t,\lambda_t)+\theta\langle v^*-v_t,\nabla_z\mathcal{L}'_\rho(v_t,\lambda_t)\rangle+\frac{\theta^2\delta_t^2}{\alpha}+\frac{\theta^2\delta_t'^2}{2\beta}\\
&\quad+\theta(R(z^*)+H(z^*)-R(z_t)-H(z_t))+\sigma_{t+1}\|A_{t+1}\|_2^2 \qquad \text{(see (44,45))}\\
&\le \mathcal{L}_\rho(v_t,\lambda_t)+\theta\left(\frac{\omega_\rho\delta_t^2}{2}+\frac{\omega'_\rho\delta_t'^2}{2}-\Delta'_t\right)+\frac{\theta^2\delta_t^2}{\alpha}+\frac{\theta^2\delta_t'^2}{2\beta}\\
&\quad+\theta(R(z^*)+H(z^*)-R(z_t)-H(z_t))+\sigma_{t+1}\|A_{t+1}\|_2^2 \qquad \text{(see (60))}\\
&=\mathcal{L}_\rho(v_t,\lambda_t)+\theta\left(\frac{\omega_\rho\delta_t^2}{2}+\frac{\omega'_\rho\delta_t'^2}{2}-\Delta_t\right)+\frac{\theta^2\delta_t^2}{\alpha}+\frac{\theta^2\delta_t'^2}{2\beta}\\
&\quad+\sigma_{t+1}\|A_{t+1}\|_2^2 \qquad \text{(see (44,45))}
\end{aligned}
\tag{75}
$$

After recalling (48) and by subtracting $\mathcal{L}_\rho(v^*,\lambda^*)$ from both sides, (75) immediately implies that

$$
\begin{aligned}
\Delta_{t+1}\le\Delta_t+\frac{\omega_\rho\delta_t^2}{2}+\frac{\omega'_\rho\delta_t'^2}{2}+\theta\left(\overline{\omega}_\rho-\Delta_t\right)+\frac{\theta^2\delta_t^2}{\alpha}+\frac{\theta^2\delta_t'^2}{2\beta}\\
+\sigma_{t+1}\|A_{t+1}\|_2^2, \quad \text{(see (48,75))}
\end{aligned}
\tag{76}
$$

where we also used the assumption that $\theta\le 1$ above. To remove the feasibility gap $\|A_{t+1}\|_2$ from the right-hand side above, we write that

$$
\begin{aligned}
\|A_{t+1}\|_2&=\|w_{t+1}-G(z_{t+1})\|_2 \qquad \text{(see (51))}\\
&=\|w_{t+1}-w^*-(G(z_{t+1})-G(z^*))\|_2 \qquad ((w^*,z^*)\text{ is a solution of (1))}\\
&\le\|w_{t+1}-w^*\|_2+\|G(z_{t+1})-G(z^*)\|_2 \qquad \text{(triangle inequality)}\\
&\le\|w_{t+1}-w^*\|_2+\kappa_G\|z_{t+1}-z^*\|_2 \qquad \text{(see (42))}\\
&=\delta_{t+1}+\kappa_G\delta'_{t+1}, \qquad \text{(see (50))}
\end{aligned}
\tag{77}
$$

which, after substituting in (76), yields that

$$
\begin{aligned}
\Delta_{t+1}&\le\Delta_t+\frac{\omega_\rho\delta_t^2}{2}+\frac{\omega'_\rho\delta_t'^2}{2}+\theta\left(\overline{\omega}_\rho-\Delta_t\right)+\frac{\theta^2\delta_t^2}{\alpha}+\frac{\theta^2\delta_t'^2}{2\beta}+2\sigma_{t+1}\delta_{t+1}^2+2\sigma_{t+1}\kappa_G^2\delta_{t+1}'^2\\
&\qquad\text{(see (77) and }(a+b)^2\le 2a^2+2b^2)\\
&\le\Delta_t+\frac{\omega_\rho\delta_t^2}{2}+\frac{\omega'_\rho\delta_t'^2}{2}+\theta\left(\overline{\omega}_\rho-\Delta_t\right)+\frac{\theta^2\delta_t^2}{\alpha}+\frac{\theta^2\delta_t'^2}{2\beta}+2\sigma_0\delta_{t+1}^2+2\sigma_0\kappa_G^2\delta_{t+1}'^2.\\
&\qquad.\qquad(\sigma_{t+1}\le\sigma_0\text{ in Algorithm 1})
\end{aligned}
\tag{78}
$$

For every iteration $t$, suppose that

$$
\frac{\delta_t^2}{\alpha}+\frac{\delta_t'^2}{\beta}\ge\overline{\eta}_\rho\ge\frac{\overline{\mu}_\rho}{\min\left(\frac{\alpha\mu_\rho}{4},\frac{\beta\mu'_\rho}{2}\right)-\sqrt{\max\left(\frac{\alpha}{2}(\omega_\rho+4\sigma_0),\beta(\omega'_\rho+4\sigma_0\kappa_G^2)\right)}},
\tag{79}
$$

for $\overline{\eta}_\rho$ to be set later. Consequently, it holds that

$$\begin{aligned}
\frac{\Delta_t}{\frac{2\delta_t^2}{\alpha}+\frac{\delta_t'^2}{\beta}} &\geq \frac{\frac{\mu_\rho \delta_t^2}{2}+\frac{\mu_\rho' \delta_t'^2}{2}-\overline{\mu}_\rho}{\frac{2\delta_t^2}{\alpha}+\frac{\delta_t'^2}{\beta}} \qquad \text{(see (59))} \\
&\geq \min\left(\frac{\alpha\mu_\rho}{4},\frac{\beta\mu_\rho'}{2}\right)-\frac{\overline{\mu}_\rho}{\frac{2\delta_t^2}{\alpha}+\frac{\delta'^2}{\beta}} \\
&\geq \min\left(\frac{\alpha\mu_\rho}{4},\frac{\beta\mu_\rho'}{2}\right)-\frac{\overline{\mu}_\rho}{\overline{\eta}_\rho} \qquad \text{(see (79))} \\
&\geq \sqrt{\max\left(\frac{\alpha}{2}\left(\omega_\rho+4\sigma_0\right),\beta(\omega_\rho'+4\sigma_0\kappa_G^2)\right)}. \qquad \text{(see (79))} \qquad (80)
\end{aligned}$$

We now set

$$\widehat{\theta}_t := \min\left(\sqrt{\frac{\Delta_t^2}{\left(\frac{2\delta_t^2}{\alpha}+\frac{\delta_t'^2}{\beta}\right)^2}-\max\left(\frac{\alpha}{2}\left(\omega_\rho+4\sigma_0\right),\beta\left(\omega_\rho'+4\sigma_0\kappa_G^2\right)\right)},1\right), \qquad (81)$$

which is well-defined, as verified in (80). From (80,81), it also immediately follows that

$$\widehat{\theta}_t \in [0,1], \qquad \forall t, \qquad (82)$$

$$\Delta_t \geq 0, \qquad \forall t, \qquad (83)$$

which we will use later on in the proof. Consider first the case where $\widehat{\theta}_t < 1$. To study the choice of $\theta = \widehat{\theta}_t$ in (76), we will need the bound

$$\begin{aligned}
&-\widehat{\theta}_t\Delta_t+\widehat{\theta}_t^2\left(\frac{\delta_t^2}{\alpha}+\frac{\delta_t'^2}{2\beta}\right) \\
&=-\sqrt{\frac{\Delta_t^4}{\left(\frac{2\delta_t^2}{\alpha}+\frac{\delta_t'^2}{\beta}\right)^2}-\Delta_t^2\max\left(\frac{\alpha}{2}\left(\omega_\rho+4\sigma_0\right),\beta\left(\omega_\rho'+4\sigma_0\kappa_G^2\right)\right)} \\
&\quad+\frac{\Delta_t^2}{\frac{4\delta_t^2}{\alpha}+\frac{2\delta_t'^2}{\beta}}-\max\left(\frac{\alpha}{2}\left(\omega_\rho+4\sigma_0\right),\beta\left(\omega_\rho'+4\sigma_0\kappa_G^2\right)\right)\left(\frac{\delta_t^2}{\alpha}+\frac{\delta_t'^2}{2\beta}\right) \qquad \text{(see (83))} \\
&\leq -\frac{\Delta_t^2}{\frac{4\delta_t^2}{\alpha}+\frac{2\delta'^2}{\beta}}+\Delta_t\sqrt{\max\left(\frac{\alpha}{2}\left(\omega_\rho+4\sigma_0\right),\beta\left(\omega_\rho'+4\sigma_0\kappa_G^2\right)\right)} \\
&\quad-\max\left(\frac{\alpha}{2}\left(\omega_\rho+4\sigma_0\right),\beta\left(\omega_\rho'+4\sigma_0\kappa_G^2\right)\right)\left(\frac{\delta_t^2}{\alpha}+\frac{\delta_t'^2}{2\beta}\right), \qquad (84)
\end{aligned}$$

where the inequality above uses $\sqrt{a-b} \geq \sqrt{a}-\sqrt{b}$. Substituting (84) back into (78), we reach

$$\begin{aligned}
\Delta_{t+1} &\leq \Delta_t-\frac{\Delta_t^2}{\frac{4\delta_t^2}{\alpha}+\frac{2\delta'^2}{\beta}}+\Delta_t\sqrt{\max\left(\frac{\alpha}{2}\left(\omega_\rho+4\sigma_0\right),\beta\left(\omega_\rho'+4\sigma_0\kappa_G^2\right)\right)} \qquad \text{(see (78,84))} \\
&\leq \Delta_t-\left(\min\left(\frac{\alpha\mu_\rho}{4},\frac{\beta\mu_\rho'}{2}\right)-\frac{\overline{\mu}_\rho}{\overline{\eta}_\rho}\right)\frac{\Delta_t}{2} \\
&\quad+\Delta_t\sqrt{\max\left(\frac{\alpha}{2}\left(\omega_\rho+4\sigma_0\right),\beta\left(\omega_\rho'+4\sigma_0\kappa_G^2\right)\right)} \qquad \text{(see third line of (80) and (83))} \\
&\leq \left(1-\min\left(\frac{\alpha\mu_\rho}{8},\frac{\beta\mu_\rho'}{4}\right)+\frac{\overline{\mu}_\rho}{2\overline{\eta}_\rho}+\sqrt{\max\left(\frac{\alpha}{2}\left(\omega_\rho+4\sigma_0\right),\beta\left(\omega_\rho'+4\sigma_0\kappa_G^2\right)\right)}\right)\Delta_t \\
&=: \eta_{\rho,1}\Delta_t, \qquad \text{if } \Delta_t < \frac{\delta_t^2}{\alpha}+\frac{\delta_t'^2}{\beta}. \qquad (85)
\end{aligned}$$

Next consider the case where $\widehat{\theta}_t = 1$. With the choice of $\theta = \widehat{\theta}_t = 1$ in (78), we find that

$$\begin{aligned}
\Delta_{t+1} &\leq \left(\frac{\omega_\rho}{2} + \frac{1}{\alpha} + \rho\right)\delta_t^2 + \left(\frac{\omega'_\rho}{2} + \frac{1}{2\beta} + \rho\kappa_G^2\right)\delta_t'^2 \qquad \text{(see (78))} \\
&\leq \frac{1}{2}\left(1 + \max\left(\frac{\alpha}{2}(\omega_\rho + 4\sigma_0), \beta(\omega'_\rho + 4\sigma_0\kappa_G^2)\right)\right) \cdot \left(\frac{2\delta_t^2}{\alpha} + \frac{\delta_t'^2}{\beta}\right) \\
&\leq \frac{1}{2}\sqrt{1 + \max\left(\frac{\alpha}{2}(\omega_\rho + 4\sigma_0), \beta(\omega'_\rho + 4\sigma_0\kappa_G^2)\right)}\Delta_t \qquad \text{(see (81))} \\
&=: \eta_{\rho,2}\Delta_t, \qquad \text{if } \Delta_t \geq \frac{\delta_t^2}{\alpha} + \frac{\delta_t'^2}{\beta}.
\end{aligned} \tag{86}$$

To simplify the above expressions, let us assume that

$$\sqrt{\max\left(\frac{\alpha}{2}(\omega_\rho + 4\sigma_0), \beta(\omega'_\rho + 4\sigma_0\kappa_G^2)\right)} \leq \min\left(\frac{\alpha\mu_\rho}{16}, \frac{\beta\mu'_\rho}{8}\right) \leq \frac{1}{2}, \tag{87}$$

from which it follows that

$$\begin{aligned}
\max(\eta_{\rho,1}, \eta_{\rho,2}) &\leq 1 - \min\left(\frac{\alpha\mu_\rho}{16}, \frac{\beta\mu'_\rho}{8}\right) + \frac{\overline{\mu}_\rho}{2\overline{\eta}_\rho} \\
&\leq 1 - \min\left(\frac{\alpha\mu_\rho}{32}, \frac{\beta\mu'_\rho}{16}\right) \\
&=: 1 - \eta_\rho \in [0, 1),
\end{aligned} \tag{88}$$

where the second line above holds if

$$\overline{\eta}_\rho \geq \frac{\overline{\mu}_\rho}{\min\left(\frac{\alpha\mu_\rho}{16}, \frac{\beta\mu'_\rho}{8}\right)}. \tag{89}$$

Then, by unfolding (85)(86), we reach

$$\Delta_t \leq (1 - \eta_\rho)^t \Delta_0. \tag{90}$$

Moreover, by combining (59)(90), we can bound the error, namely,

$$\begin{aligned}
\frac{\delta_t^2}{\alpha} + \frac{\delta_t'^2}{\beta} &\leq \max(\alpha\mu_\rho, \beta\mu'_\rho)\left(\mu_\rho\delta_t^2 + \mu'_\rho\delta_t'^2\right) \\
&\leq \mu_\rho\delta_t^2 + \mu'_\rho\delta_t'^2 \qquad \text{(see (65)(71), Lemmas 6 and 7)} \\
&\leq 2(\Delta_t + \overline{\mu}_\rho) \qquad \text{(see (59))} \\
&\leq 2(1 - \eta_\rho)^t\Delta_0 + \frac{2\overline{\mu}_\rho}{\eta_\rho} \\
&\leq 2(1 - \eta_\rho)^t\Delta_0 + \frac{2\overline{\mu}_\rho}{\min\left(\frac{\alpha\mu_\rho}{16}, \frac{\beta\mu'_\rho}{8}\right)} \qquad \text{(see (88))} \\
&=: 2(1 - \eta_\rho)^t\Delta_0 + \frac{\overline{\eta}_\rho}{\rho}. \qquad \text{(this choice of } \overline{\eta}_\rho \text{ satisfies (79)(89)).}
\end{aligned} \tag{91}$$

It remains to bound the feasibility gap $\|A_t\|_2$, see (51). Instead of (77), we consider the following alternative approach to bound $\|A_t\|_2$. Using definition of $\Delta_t$ in (48), we write that

$$\begin{aligned}
\Delta_t &= \mathcal{L}_\rho(v_t, \lambda_t) - \mathcal{L}_\rho(v^*, \lambda^*) \qquad \text{(see (48))} \\
&= \mathcal{L}_\rho(v_t, \lambda_t) - \mathcal{L}_\rho(v_t, \lambda^*) + \mathcal{L}_\rho(v_t, \lambda^*) - \mathcal{L}_\rho(v^*, \lambda^*) \\
&= \langle A_t, \lambda_t - \lambda^*\rangle + \mathcal{L}(v_t, \lambda^*) - \mathcal{L}(v^*, \lambda^*) + \frac{\rho}{2}\|A_t\|_2^2,
\end{aligned} \tag{92}$$

where

$$\mathcal{L}(v, \lambda) = \mathcal{L}(w, z, \lambda) := L(w) + R(w) + H(z) + \langle w - G(z), \lambda\rangle. \tag{93}$$

It is not difficult to verify that $\mathcal{L}(v^*, \lambda^*) = \mathcal{L}_\rho(v^*, \lambda^*)$ is the optimal value of problem (1) and that $\mathcal{L}(v_t, \lambda^*) \geq \mathcal{L}(v^*, \lambda^*)$, from which it follows that

$$\begin{aligned}
\Delta_t &\geq \langle A_t, \lambda_t - \lambda^* \rangle + \frac{\rho}{2}\|A_t\|_2^2 \qquad \text{(see (92))} \\
&\geq -\frac{\rho}{4}\|A_t\|_2^2 - \frac{1}{\rho}\|\lambda_t - \lambda^*\|_2^2 + \frac{\rho}{2}\|A_t\|_2^2 \qquad \text{(Holder's inequality and } 2ab \leq a^2 + b^2) \\
&\geq -\frac{2}{\rho}\|\lambda_t\|_2^2 - \frac{2}{\rho}\|\lambda^*\|_2^2 + \frac{\rho}{4}\|A_t\|_2^2 \qquad ((a+b)^2 \leq 2a^2 + 2b^2) \\
&\geq -\frac{2\lambda_{\max}^2}{\rho} - \frac{2\|\lambda^*\|_2^2}{\rho} + \frac{\rho}{4}\|A_t\|_2^2, \qquad \text{(see (100))}
\end{aligned} \tag{94}$$

which, in turn, implies that

$$\begin{aligned}
\|A_t\|_2^2 &\leq \frac{4}{\rho}\left(\Delta_t + \frac{2\lambda_{\max}^2}{\rho} + \frac{2\|\lambda^*\|_2^2}{\rho}\right) \qquad \text{(see (94))} \\
&\leq \frac{4}{\rho}\left((1-\eta_\rho)^t\Delta_0 + \frac{2(\overline{\eta}_{\rho,1} + \overline{\eta}_{\rho,2})}{\eta_\rho} + \frac{2\lambda_{\max}^2}{\rho} + \frac{2\|\lambda^*\|_2^2}{\rho}\right) \qquad \text{(see (90))} \\
&\leq \frac{4}{\rho}\left((1-\eta_\rho)^t\Delta_0 + \frac{\overline{\eta}_\rho + 2\lambda_{\max}^2 + 2\|\lambda^*\|_2^2}{\rho}\right) \qquad \text{(see (91))} \\
&=: \frac{4(1-\eta_\rho)^t\Delta_0}{\rho} + \frac{\widetilde{\eta}_\rho}{\rho^2}.
\end{aligned} \tag{95}$$

This completes the proof of Theorem 1.

Let us also inspect the special case where $\mu_L \gg \rho \gtrsim 1$ and $\iota_G^2 \gg \nu_G$, where $\approx$ and $\gtrsim$ suppress any universal constants and dependence on the dual optimal variable $\lambda^*$, for the sake of simplicity. From Lemmas 6 and 7, it is easy to verify that

$$\nu_\rho \approx \nu_L, \qquad \xi_\rho \approx \rho\kappa_G^2, \qquad \tau_\rho = \rho\kappa_G,$$

$$\mu_\rho \approx \mu_L, \qquad \mu_\rho' \approx \rho\iota_G^2, \qquad \overline{\mu}_\rho \approx \rho^{-1}, \qquad \omega_\rho' \approx \rho\nu_G. \tag{96}$$

We can then take

$$\begin{aligned}
\alpha &\approx \frac{1}{\nu_L}, \qquad \text{(see (65))} \\
\beta &\approx \frac{1}{\xi_\rho} \approx \frac{1}{\rho\kappa_G^2}, \qquad \text{(see (71))} \\
\eta_\rho &\approx \min\left(\frac{\mu_L}{\nu_L}, \frac{\iota_G^2}{\kappa_G^2}\right), \qquad \text{(see (88))} \\
\overline{\eta}_\rho &\approx \frac{\rho\overline{\mu}_\rho}{\min\left(\alpha\mu_\rho, \beta\mu_\rho'\right)} \approx \max\left(\frac{\nu_L}{\mu_L}, \frac{\kappa_G^2}{\iota_G^2}\right), \qquad \text{(see (91))} \\
\widetilde{\eta}_\rho &\approx \overline{\eta}_\rho \approx \max\left(\frac{\nu_L}{\mu_L}, \frac{\kappa_G^2}{\iota_G^2}\right). \qquad \text{(see (95))}
\end{aligned} \tag{97}$$

Lastly, for (87) to hold, it suffices that

$$\sigma_0 \lesssim \rho\min\left(\frac{\mu_L^2}{\nu_L^2}, \frac{\iota_G^4}{\kappa_G^4}\right) =: \sigma_{0,\rho}. \tag{98}$$

## D Proof of Lemma 6

To prove (53), we write that

$$\begin{aligned}
&\mathcal{L}'_\rho(w_{t+1}, z_{t+1}, \lambda_t) - \mathcal{L}'_\rho(w_t, z_{t+1}, \lambda_t) - \langle w_{t+1} - w_t, \nabla_w \mathcal{L}'_\rho(w_t, z_{t+1}, \lambda_t)\rangle \\
&= L(w_{t+1}) - L(w_t) - \langle w_{t+1} - w_t, \nabla_w L(w_t)\rangle \\
&\quad + \frac{\rho}{2}\|w_{t+1} - G(z_{t+1})\|_2^2 - \frac{\rho}{2}\|w_t - G(z_{t+1})\|_2^2 - 2\rho\langle w_{t+1} - w_t, w_t - G(z_{t+1})\rangle \qquad \text{(see (45))} \\
&\le \frac{\nu_L}{2}\|w_{t+1} - w_t\|_2^2 + \overline{\nu}_L + \frac{\rho}{2}\|w_{t+1} - w_t\|_2^2 \qquad \text{(see (40))} \\
&=: \frac{\nu_\rho}{2}\|w_{t+1} - w_t\|_2^2 + \overline{\nu}_\rho. \qquad (99)
\end{aligned}$$

To prove (54), let us first control the dual sequence $\{\lambda_t\}_t$ by writing that

$$\begin{aligned}
\|\lambda_t\|_2 &= \|\lambda_0 + \sum_{i=1}^t \sigma_i A_i\|_2 \qquad \text{(see (52))} \\
&\le \|\lambda_0\|_2 + \sum_{i=1}^t \sigma_i \|A_i\|_2 \qquad \text{(triangle inequality)} \\
&\le \|\lambda_0\|_2 + \sum_{t'=1}^t \frac{\sigma_0}{i\log^2(i+1)} \\
&\le \|\lambda_0\|_2 + c\sigma_0 \\
&=: \lambda_{\max}, \qquad (100)
\end{aligned}$$

where

$$c \ge \sum_{t=1}^\infty \frac{1}{t\log^2(t+1)}. \qquad (101)$$

We now write that

$$\begin{aligned}
&\mathcal{L}'_\rho(w_t, z_{t+1}, \lambda_t) - \mathcal{L}'_\rho(w_t, z_t, \lambda_t) - \langle z_{t+1} - z_t, \nabla_z \mathcal{L}'_\rho(w_t, z_t, \lambda_t)\rangle \\
&= -\langle G(z_{t+1}) - G(z_t) - DG(z_t)(z_{t+1} - z_t), \lambda_t\rangle \\
&\quad + \frac{\rho}{2}\|w_t - G(z_{t+1})\|_2^2 - \frac{\rho}{2}\|w_t - G(z_t)\|_2^2 \\
&\quad + \rho\langle DG(z_t)(z_{t+1} - z_t), w_t - G(z_t)\rangle. \qquad \text{(see (45))} \qquad (102)
\end{aligned}$$

To bound the first inner product on the right-hand side above, we write that

$$\begin{aligned}
&\langle G(z_{t+1}) - G(z_t) - DG(z_t)(z_{t+1} - z_t), \lambda_t\rangle \\
&\le \|G(z_{t+1}) - G(z_t) - DG(z_t)(z_{t+1} - z_t)\|_2 \cdot \|\lambda_t\|_2 \qquad \text{(Cauchy-Shwartz's inequality)} \\
&\le \frac{\nu_G \lambda_{\max}}{2}\|z_{t+1} - z_t\|_2^2 \qquad \text{(see (41), (100))} \qquad (103)
\end{aligned}$$

The remaining component on the right-hand side of (102) can be bounded as

$$\begin{aligned}
&\|w_t - G(z_{t+1})\|_2^2 - \|w_t - G(z_t)\|_2^2 + 2\langle DG(z_t)(z_{t+1} - z_t), w_t - G(z_t)\rangle \\
&= \|w_t - G(z_{t+1})\|_2^2 - \|w_t - G(z_t)\|_2^2 + 2\langle G(z_{t+1}) - G(z_t), w_t - G(z_t)\rangle \\
&\quad - 2\langle G(z_{t+1}) - G(z_t) - DG(z_t)(z_{t+1} - z_t), w_t - G(z_t)\rangle \\
&= \|G(z_{t+1}) - G(z_t)\|_2^2 \\
&\quad + 2\langle G(z_{t+1}) - G(z_t) - DG(z_t)(z_{t+1} - z_t), w_t - G(z_t)\rangle \\
&\le \|G(z_{t+1}) - G(z_t)\|_2^2 \\
&\quad + 2\|G(z_{t+1}) - G(z_t) - DG(z_t)(z_{t+1} - z_t)\|_2 \cdot \|w_t - G(z_t)\|_2 \qquad \text{(Cauchy-Shwartz's inequality)} \\
&\le \kappa_G^2\|z_{t+1} - z_t\|_2^2 + \nu_G\|z_{t+1} - z_t\|_2^2\|w_t - G(z_t)\|_2 \qquad \text{(see (41), (42))} \\
&\le \kappa_G^2\|z_{t+1} - z_t\|_2^2 + \nu_G\|z_{t+1} - z_t\|_2^2 \max_i \|A_i\|_2. \qquad \text{(see (51))} \qquad (104)
\end{aligned}$$

Substituting the bounds in (103,104) back into (102), we find that

$$\begin{aligned}
&\mathcal{L}'_\rho(w_t, z_{t+1}, \lambda_t) - \mathcal{L}'_\rho(w_t, z_t, \lambda_t) - \langle z_{t+1} - z_t, \nabla_z \mathcal{L}'_\rho(w_t, z_t, \lambda_t)\\
&\leq \frac{1}{2}\left(\nu_G(\lambda_{\max} + \rho \max_i \|A_i\|_2) + \rho\kappa_G^2\right)\|z_{t+1} - z_t\|_2^2\\
&=: \frac{\xi_\rho}{2}\|z_{t+1} - z_t\|_2^2 + \overline{\xi}_\rho, \qquad (105)
\end{aligned}$$

which proves (54). To prove (55), we write that

$$\begin{aligned}
&\|\nabla_w \mathcal{L}'_\rho(w_t, z_{t+1}, \lambda_t) - \nabla_w \mathcal{L}'_\rho(w_t, z_t, \lambda_t)\|_2\\
&= \rho\|G(z_{t+1}) - G(z_t)\|_2 \qquad \text{(see (45))}\\
&\leq \rho\kappa_G\|z_{t+1} - z_t\|_2 \qquad \text{(see (42))}\\
&=: \tau_\rho\|z_{t+1} - z_t\|_2 + \overline{\tau}_\rho. \qquad (106)
\end{aligned}$$

This completes the proof of Lemma 6.

## E Proof of Lemma 7

For future reference, we record that

$$\begin{aligned}
&\langle v_t - v^*, \nabla_v \mathcal{L}'_\rho(v^*)\rangle\\
&= \langle w_t - w^*, \nabla_w \mathcal{L}'_\rho(v^*)\rangle + \langle z_t - z^*, \nabla_z \mathcal{L}'_\rho(v^*)\rangle \qquad (v = (w, z))\\
&= \langle w_t - w^*, \nabla L(w^*) + \lambda^* + \rho(w^* - G(z^*)\rangle - \langle DG(z^*)(z_t - z^*), \lambda^* + \rho(w^* - G(z^*))\rangle \qquad \text{(see (45))}\\
&= \langle w_t - w^*, \nabla L(w^*) + \lambda^*\rangle - \langle DG(z^*)(z_t - z^*), \lambda^*\rangle, \qquad \text{(see (47))} \qquad (107)
\end{aligned}$$

where the last line above uses the feasibility of $v^*$ in (1). To prove (59), we use the definition of $\mathcal{L}_\rho$ in (44) to write that

$$\begin{aligned}
&\mathcal{L}_\rho(v_t, \lambda_t) - \mathcal{L}_\rho(v^*, \lambda^*)\\
&= \mathcal{L}'_\rho(v_t, \lambda_t) - \mathcal{L}'_\rho(v^*, \lambda^*) + R(w_t) - R(w^*) + L(z_t) - L(z^*) \qquad \text{(see (44,45))}\\
&\geq \mathcal{L}'_\rho(v_t, \lambda_t) - \mathcal{L}'_\rho(v^*, \lambda^*) - \langle v_t - v^*, \nabla_v \mathcal{L}'_\rho(v^*, \lambda^*)\rangle \qquad \text{(see (47))}\\
&= L(w_t) - L(w^*) - \langle w_t - w^*, \nabla L(u^*)\rangle\\
&\qquad + \langle A_t, \lambda_t\rangle - \langle w_t - w^* - DG(z^*)(z_t - z^*), \lambda^*\rangle + \frac{\rho}{2}\|A_t\|_2^2 \qquad \text{(see (107))}\\
&\geq \frac{\mu_L\delta_t^2}{2} + \langle A_t, \lambda_t - \lambda^*\rangle + \frac{\rho}{2}\|A_t\|_2^2\\
&\qquad + \langle G(z_t) - G(z^*) - DG(z^*)(z_t - z_k^*), \lambda^*\rangle \qquad \text{(see (39,50))}\\
&\geq \frac{\mu_L\delta_t^2}{2} + \langle A_t, \lambda_t - \lambda^*\rangle + \frac{\rho}{2}\|A_t\|_2^2 - \frac{\nu_G\delta_t'^2}{2}\|\lambda^*\|_2. \qquad \text{(see (41,50))} \qquad (108)
\end{aligned}$$

To control the terms involving $A_t$ in the last line above, we write that

$$\begin{aligned}
&\langle A_t, \lambda_t - \lambda^*\rangle + \frac{\rho}{2}\|A_t\|_2^2\\
&= \frac{\rho}{2}\left\|A_t - \frac{\lambda_t - \lambda^*}{\rho}\right\|_2^2 - \frac{\|\lambda_t - \lambda^*\|_2^2}{2\rho}\\
&= \frac{\rho}{2}\left\|w_t - w^* - (G(z_t) - G(z^*)) - \frac{\lambda_t - \lambda^*}{\rho}\right\|_2^2 - \frac{\|\lambda_t - \lambda^*\|_2^2}{2\rho} \qquad \text{(see (47,51))}\\
&\geq \frac{\rho}{4}\|G(z_t) - G(z^*)\|_2^2 - \rho\delta_t^2 - \frac{3\|\lambda_t - \lambda^*\|_2^2}{2\rho} \qquad \left(\|a - b - c\|_2^2 \geq \frac{\|a\|_2^2}{2} - 2\|b\|_2^2 - 2\|c\|_2^2\right)\\
&\geq \frac{\rho\iota_G^2\delta_t'^2}{4} - \rho\delta_t^2 - \frac{3\|\lambda_t - \lambda^*\|_2^2}{2\rho} \qquad \text{(see (50,42))}\\
&\geq \frac{\rho\iota_G^2\delta_t'^2}{4} - \rho\delta_t^2 - \frac{3}{\rho}(\lambda_{\max}^2 + \|\lambda^*\|_2^2), \qquad ((a+b)^2 \leq 2a^2 + 2b^2 \text{ and (100)}) \qquad (109)
\end{aligned}$$

which, after substituting in (108), yields that

$$\begin{aligned}
&\mathcal{L}_\rho(v_t,\lambda_t)-\mathcal{L}_\rho(v^*,\lambda^*)\\
&\geq \frac{\mu_L-2\rho}{2}\delta_t^2+\frac{1}{2}\left(\frac{\rho\iota_G^2}{2}-\nu_G\|\lambda^*\|_2\right)\delta_t'^2-\frac{3}{\rho}\left(\lambda_{\max}^2+\|\lambda^*\|_2^2\right)\\
&\geq \frac{\mu_\rho\delta_t^2}{2}+\frac{\mu_\rho'\delta_t'^2}{2}-\overline{\mu}_\rho,
\end{aligned}\tag{110}$$

where

$$\mu_\rho:=\mu_L-2\rho,\qquad \mu_\rho':=\frac{\rho\iota_G^2}{2}-\nu_G\|\lambda^*\|_2,\tag{111}$$

$$\overline{\mu}_\rho:=\frac{3}{\rho}\left(\lambda_{\max}^2+\|\lambda^*\|_2^2\right).\tag{112}$$

This proves (59). To prove (60), we use the definition of $\mathcal{L}_\rho'$ in (45) to write that

$$\begin{aligned}
&\mathcal{L}_\rho'(v^*,\lambda^*)-\mathcal{L}_\rho'(v_t,\lambda_t)-\langle v^*-v_t,\nabla_v\mathcal{L}_\rho'(v_t,\lambda_t)\rangle\\
&=L(w^*)-L(w_t)-\langle w^*-w_t,\nabla L(w_t)\rangle\\
&\quad-\langle A_t+DA(v_t)(v^*-v_t),\lambda_t\rangle\\
&\quad-\frac{\rho}{2}\langle A_t+2DA(v_t)(v^*-v_t),A_t\rangle,\qquad\text{(see (45))}
\end{aligned}\tag{113}$$

where

$$DA(v)=\begin{bmatrix} I_d & -DG(z)\end{bmatrix},\tag{114}$$

is the Jacobian of the map $A$. The second inner product on the right-hand side of (113) can be bounded as

$$\begin{aligned}
&-\langle A_t+DA(v_t)(v^*-v_t),\lambda_t\rangle\\
&=-\langle w_t-G(z_t)+(w^*-w_t)-DG(z_t)(z^*-z_t),\lambda_t\rangle\qquad\text{(see (51,114))}\\
&=-\langle G(z^*)-G(z_t)-DG(z_t)(z^*-z_t),\lambda_t\rangle\qquad(w^*=G(z^*))\\
&\geq-\frac{\nu_G\delta_t'^2}{2}\|\lambda_t\|_2\qquad\text{(see (41,50))}\\
&\geq-\frac{\nu_G\delta_t'^2}{2}\lambda_{\max}.\qquad\text{(see (100))}
\end{aligned}\tag{115}$$

To control the last inner product on the right-hand side of (113), we write that

$$\begin{aligned}
&-\frac{\rho}{2}\langle A_t+2DA(v_t)(v^*-v_t),A_t\rangle\\
&=\frac{\rho}{2}\|A_t\|_2^2-\rho\langle A_t+DA(v_t)(v^*-v_t),A_t\rangle\\
&\geq-\rho\|A_t+DA(v_t)(v^*-v_t)\|_2\|A_t\|_2\qquad\text{(Holder's inequality)}\\
&=-\rho\|(w^*-G(z^*))-(w_t-G(z_t))-(w^*-w_t)+DG(z_t)(z^*-z_t)\|_2\qquad\text{(see (51,114) and } w^*=G(z^*))\\
&=-\rho\|G(z^*)-G(z_t)-DG(z_t)(z^*-z_t)\|_2\\
&\geq-\frac{\rho\nu_G}{2}\|z^*-z_t\|_2^2\qquad\text{(see (41))}\\
&=-\frac{\rho\nu_G\delta_t'^2}{2}.\qquad\text{(see (50))}
\end{aligned}\tag{116}$$

By substituting the bounds in (115,116) back into (113) and also using the convexity of $L$, we reach

$$\begin{aligned}
&\mathcal{L}_\rho'(v^*,\lambda^*)-\mathcal{L}_\rho'(v_t,\lambda_t)-\langle v^*-v_t,\nabla_v\mathcal{L}_\rho'(v_t,\lambda_t)\rangle\\
&\geq-\frac{\nu_G}{2}\left(\lambda_{\max}+\rho\right)\delta_t'^2.
\end{aligned}\tag{117}$$

This proves (60), thus completing the proof of Lemma 7.

## F Relation with Gradient Descent

Throughout this section, we set $R \equiv 0$ and $H \equiv 0$ in problem (1) and consider the updates in Algorithm 2, namely,

$$\begin{aligned} z_{t+1} &= z_t - \beta \boldsymbol{\nabla}_z \mathcal{L}_\rho(w_t, z_t, \lambda_t), \\ w_{t+1} &\in \operatorname*{argmin}_w \mathcal{L}_\rho(w, z_{t+1}, \lambda_t), \\ \lambda_{t+1} &= \lambda_t + \sigma_{t+1}(w_{t+1} - G(z_{t+1})). \end{aligned} \tag{118}$$

From (2), recall that $\mathcal{L}_\rho(w, z, \lambda)$ is convex in $w$ and the second step in (118) is therefore often easy to implement with any over-the-shelf standard convex solver. Recalling (2), note also that the optimality condition for $w_{t+1}$ in (118) is

$$w_{t+1} - G(z_t) = -\frac{1}{\rho}(\nabla L_m(w_{t+1}) + \lambda_t). \tag{119}$$

Using (2) again, we also write that

$$\begin{aligned} &\nabla_z \mathcal{L}_\rho(w_{t+1}, z_t, \lambda_t) \\ &= -DG(z_t)^\top(\lambda_t + \rho(w_{t+1} - G(z_t)) \\ &= -DG(z_t)^\top(\lambda_t - \lambda_{t-1} - \nabla L_m(w_t)) \\ &= -DG(z_t)^\top(\sigma_t(w_t - G(z_t)) - \nabla L(w_t)), \end{aligned} \tag{120}$$

where the last two lines above follow from (119,118), respectively. Substituting back into the $z$ update in (118), we reach

$$z_{t+1} = z_t + \beta\sigma_t DG(z_t)^\top(w_t - G(z_t)) - \beta\nabla L(w_t) \qquad \text{(see (118,120))}, \tag{121}$$

from which it follows that

$$\begin{aligned} &\|z_{t+1} - (z_t - \beta\nabla L(G(z_t)))\|_2 \\ &\leq \beta\sigma_t \|DG(z_t)^\top(w_t - G(z_t))\|_2 + \beta\|\nabla L(w_t) - \nabla L(G(z_t))\|_2 \qquad \text{(see (121))} \\ &\leq \beta\left(\sigma_t\kappa_G + \nu_L\right)\|w_t - G(z_t)\|_2. \qquad \text{(see Assumptions 1 and 3)} \end{aligned} \tag{122}$$

That is, as the feasibility gap vanishes in (24) in Theorem 1, the updates of Algorithm 2 match those of GD.

## G Proof of Lemma 3

Recall that $R = 1_W$ and $H \equiv 0$ for this proof. Using the optimality of $w^* \in \operatorname{relint}(W)$ in (17), we can write that

$$\begin{aligned} \|\nabla L(w^*)\|_2 &\leq \|\nabla L_m(w^*)\|_2 + \|\nabla L_m(w^*) - \nabla L(w^*)\|_2 \qquad \text{(triangle inequality)} \\ &= \|\nabla L_m(w^*) - \nabla L(w^*)\|_2 \qquad (\nabla L_m(w^*) = 0) \\ &\leq \max_{w \in W} \|\nabla L_m(w) - \nabla L(w)\|_2. \end{aligned} \tag{123}$$

On the other hand, using the strong convexity of $L$ in (20), we can write that

$$\begin{aligned} \|w^\natural - w^*\|_2 &\leq \frac{1}{\zeta_L}\|\nabla L(w^\natural) - \nabla L(w^*)\|_2 \qquad \text{(see (20))} \\ &= \frac{1}{\zeta_L}\|\nabla L(w^*)\| \qquad (\nabla L(w^\natural) = 0) \\ &\leq \frac{1}{\zeta_L}\max_{w \in W}\|\nabla L_m(w) - \nabla L(w)\|_2, \qquad \text{(see (123))} \end{aligned} \tag{124}$$

which completes the proof of Lemma 3.

# H Experimental Setup Details

## H.1 Per-Iteration Computational Complexity

The gradient of the function

$$h(z) = \frac{1}{2}\|AG(z) - b\|_2^2 \tag{125}$$

follows the formula

$$\nabla h(z) = \nabla G(z) A^\top (AG(z) - b) \tag{126}$$

which involves one forward pass through the network $G$, in order to compute $G(z)$, as well as one backward pass to compute $\nabla G(z)$, and finally matrix-vector products to compute the final result.

On the other hand our ADMM first computes the iterate $z_{t+1}$ with gradient descent on the augmented lagrangian (2) as

$$z_{t+1} = z_t - \beta \nabla_z \mathcal{L}_\rho(w_t, z_t, \lambda_t) = -\nabla G(z_t)\lambda_t^\top - \rho \nabla G(z_t)(w_t - G(z_t))^\top \tag{127}$$

which involves one forward and one backward pass on the network $G$, as well as matrix-vector products. Then we perform the exact minimization procedure on the $w$ variable, which requires recomputing $G(z)$ on the new iterate $z_{t+1}$, involving one forward pass through the network, as well as the matrix-vector operations as described before. Recomputing the quantity $w_{t+1} - G(z_{t+1})$ is immediate upon which the dual stepsize $\sigma_{t+1}$ can be computed at negligible cost. Finally the dual variable update reads as

$$\lambda_{t+1} = \lambda_t + \sigma(w_{t+1} - G(z_{t+1})) \tag{128}$$

which involves only scalar products and vector additions of values already computed. All in all each GD iteration involves one forward and one backward pass, while ADMM computes two forward and one backward pass. Both algorithms require a few additional matrix-vector operations of similar complexity. For networks with multiple large layers, as usually encountered in practice, the complexity per iteration can then be estimated as the number of forward and backward passes, which are of similar complexity.

## H.2 Parameter Tuning

We run a grid search for the gradient descent (GD) algorithm In order to do so we fix a number of iterations and compare the average objective function over a batch of 100 random images and choose the best performing parameters. We repeat the tuning in all possible escenarios in the experiments. The results figures 4 - 5 (GD, Compressive sensing setup).

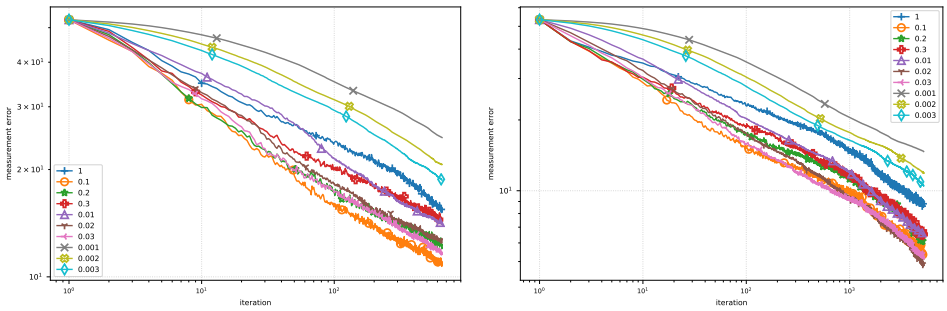


Figure 4: Performance of GD on the compressive sensing task for different step sizes. MNIST dataset. 156 (top) and 313 (bottom) linear measurements.

## H.3 Fast Exact Augmented Lagrangian Minimization with Respect to Primal Variable $w$

In the compressive sensing setup, the augmented lagrangian takes the form

$$\mathcal{L}_\rho(w, z, \lambda) := \frac{1}{2}\|Aw - b\|_2^2 + \langle \lambda, w - G(z)\rangle + \frac{\rho}{2}\|w - G(z)\|_2^2 \tag{129}$$

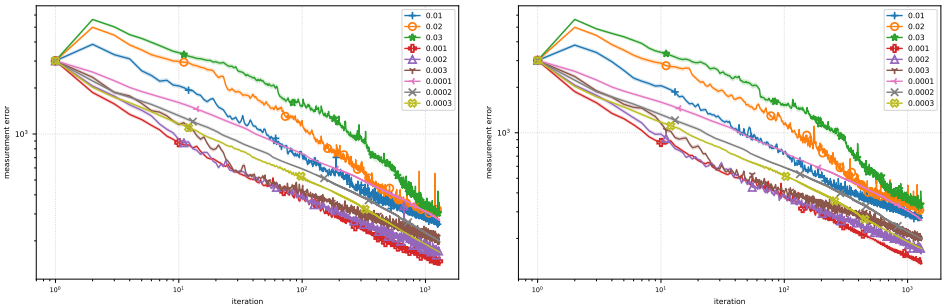


Figure 5: Performance of GD on the compressive sensing task for different step sizes. CelebA dataset. 2457 (top) and 4915 (bottom) linear measurements.

with respect to $w$, this is a strongly convex function which admits a unique minimizer given by the first order optimality condition

$$\nabla_w \mathcal{L}_\rho(w, z, \lambda) = A^\top(Aw - b) + \lambda + \rho(w - G(z)) = 0 \tag{130}$$

with solution

$$w^* = (A^\top A + \rho I)^{-1}(-\lambda + G(z) + A^\top b) \tag{131}$$

Given the SVD of $A = USV^\top$ we have $A^\top A = VDV^\top$, where $D$ corresponds to the diagonal matrix with the eigenvalues of $A^T A$. We then have that $A^\top A + \rho I = V(D + \rho I)V^\top$ so that

$$w^* = V(D + \rho I)V^\top(-\lambda + G(z) + A^\top b) \tag{132}$$

which involves only a fixed number of matrix-vector products per-iteration.

### H.4 Per-Iteration Computational Complexity

The gradient of the function

$$h(z) = \frac{1}{2}\|AG(z) - b\|_2^2 \tag{133}$$

follows the formula

$$\nabla h(z) = \nabla G(z) A^\top(AG(z) - b) \tag{134}$$

which involves one forward pass through the network $G$, in order to compute $G(z)$, as well as one backward pass to compute $\nabla G(z)$, and finally matrix-vector products to compute the final result.

On the other hand our ADMM first computes the iterate $z_{t+1}$ with gradient descent on the augmented lagrangian (129)

$$z_{t+1} = z_t - \beta \nabla_z \mathcal{L}_\rho(w_t, z_t, \lambda_t) = -\nabla G(z_t)\lambda_t^\top - \rho \nabla G(z_t)(w_t - G(z_t))^\top \tag{135}$$

which involves one forward and one backward pass on the network $G$, as well as matrix-vector products. Then we perform the exact minimization procedure on the $w$ variable, as described in H.3, which requires recomputing $G(z)$ on the new iterate $z_{t+1}$, involving one forward pass through the network, as well as the matrix-vector operations as described before. Recomputing the quantity $w_{t+1} - G(z_{t+1})$ is immediate upon which the dual stepsize $\sigma_{t+1}$ can be computed at negligible cost. Finally the dual variable update reads as

$$\lambda_{t+1} = \lambda_t + \sigma(w_{t+1} - G(z_{t+1})) \tag{136}$$

which involves only scalar products and vector additions of values already computed. All in all each GD iteration involves one forward and one backward pass, while ADMM computes two forward and one backward pass. Both algorithms require a few additional matrix-vector operations of similar complexity. For networks with multiple large layers, as usually encountered in practice, the complexity per iteration can then be estimated as the number of forward and backward passes, which are of similar complexity.

# I Pseudocode for Algorithm 2

**Algorithm 2** Multi-scale Linearized ADMM

**Input:** Differentiable $L$, proximal-friendly convex regularizers $R$ and $H$, differentiable prior $G$, penalty weight $\rho > 0$, primal step sizes $\alpha, \beta > 0$, initial dual step size $\sigma_0 > 0$, primal initialization $w_0$ and $z_0$, dual initialization $\lambda_0$, stopping threshold $\tau_c > 0$, iterations parameter $n$.

$z_{0,0} \leftarrow z_0$, $w_{0,0} \leftarrow w_0$

**for** k=0,..., K **do**

  $\rho_k \leftarrow \rho 2^k$, $\alpha_k \leftarrow \alpha 2^{-k}$, $\beta_k \leftarrow \beta 2^{-k}$

  $z_0 \leftarrow z_{0,k}$, $w_0 \leftarrow w_{0,k}$

  **for** $t = 0, 1, \ldots, 2^k n$ **do**

    $z_{t+1} \leftarrow \mathrm{P}_{\beta_k H}\left(z_t - \beta_k \nabla_z \mathcal{L}_{\rho_k}(w_t, z_t, \lambda_t)\right)$ (primal updates)

    $w_{t+1} \leftarrow \mathrm{P}_{\alpha_k R}\left(w_t - \alpha_k \nabla_w \mathcal{L}_{\rho}(w_t, z_{t+1}, \lambda_t)\right)$

    $\sigma_{t+1} \leftarrow \min\left(\sigma_0, \dfrac{\sigma_0}{\|w_{t+1} - G(z_{t+1})\|_2 t \log^2(t+1)}\right)$ (dual step size)

    $\lambda_{t+1} \leftarrow \lambda_t + \sigma_{t+1}(w_{t+1} - G(z_{t+1}))$ (dual update)

    $s \leftarrow \dfrac{\|z_{t+1} - z_t\|_2^2}{\alpha_k} + \dfrac{\|w_{t+1} - w_t\|_2^2}{\beta_k} + \sigma_t \|w_t - G(z_t)\|_2^2 \leq \tau_c$ (stopping criterion)

    **if** $s \leq \tau_c$ **then return** $(w_{t+1}, z_{t+1})$

    $(w_{0,k+1}, z_{0,k+1}) \leftarrow (w_{t+1}, z_{t+1})$

**return** $(w_{0,K+1}, z_{0,K+1})$