Operations Research

Publication details, including instructions for authors and subscription information:
http://pubsonline.informs.org

High-Dimensional Learning Under Approximate Sparsity with Applications to Nonsmooth Estimation and Regularized Neural Networks

Hongcheng Liu, Yinyu Ye, Hung Yi Lee

To cite this article:
Hongcheng Liu, Yinyu Ye, Hung Yi Lee (2022) High-Dimensional Learning Under Approximate Sparsity with Applications to Nonsmooth Estimation and Regularized Neural Networks. Operations Research

Published online in Articles in Advance 23 Mar 2022

. https://doi.org/10.1287/opre.2021.2217

informs
http://pubsonline.informs.org/journal/opre

OPERATIONS RESEARCH
Articles in Advance, pp. 1–22
ISSN 0030-364X (print), ISSN 1526-5463 (online)

Methods

High-Dimensional Learning Under Approximate Sparsity with Applications to Nonsmooth Estimation and Regularized Neural Networks

Hongcheng Liu,[a],* Yinyu Ye,[b] Hung Yi Lee[a]

[a] Department of Industrial and Systems Engineering, University of Florida, Gainesville, Florida 32611; [b] Department of Management Science and Engineering, Stanford University, Stanford, California 94305
*Corresponding author
Contact: liu.h@ufl.edu, https://orcid.org/0000-0003-2451-5204 (HL); yyye@stanford.edu (YY); hungyilee@ufl.edu (HYL)

Received: March 1, 2019
Revised: September 22, 2020; May 30, 2021
Accepted: October 13, 2021
Published Online in Articles in Advance: March 23, 2022

Area of Review: Machine Learning and Data Science

https://doi.org/10.1287/opre.2021.2217

Abstract. High-dimensional statistical learning (HDSL) has wide applications in data analysis, operations research, and decision making. Despite the availability of multiple theoretical frameworks, most existing HDSL schemes stipulate the following two conditions: (a) the sparsity and (b) restricted strong convexity (RSC). This paper generalizes both conditions via the use of the folded concave penalty (FCP). More specifically, we consider an M-estimation problem where (i) (conventional) sparsity is relaxed into the approximate sparsity and (ii) RSC is completely absent. We show that the FCP-based regularization leads to poly-logarithmic sample complexity; the training data size is only required to be poly-logarithmic in the problem dimensionality. This finding can facilitate the analysis of two important classes of models that are currently less understood: high-dimensional nonsmooth learning and (deep) neural networks (NNs). For both problems, we show that poly-logarithmic sample complexity can be maintained. In particular, our results indicate that the generalizability of NNs under overparameterization can be theoretically ensured with the aid of regularization.

Funding: This work was supported by the National Science Foundation [Grant 2016571] and a University of Florida AI Catalyst Grant.
Supplemental Material: The e-companion is available at https://doi.org/10.1287/opre.2021.2217.

Keywords: neural network • folded concave penalty • high-dimensional learning • sparsity • support vector machine • nonsmooth learning • restricted strong convexity

1. Introduction

This paper is concerned with *high-dimensional statistical learning* (HDSL), which refers to the problems of estimating a large number of parameters with few training data. The HDSL problems are found in wide applications ranging from imaging and bioinformatics to deep learning. A standard setup of the HDSL is summarized here: We are given a sequence of n-many independent and identically distributed (i.i.d.) sample observations, denoted Z_i, $i=1,\ldots,n$. Those observations are copies of a random vector $\mathcal{Z}$, which has unknown support $\mathcal{W}\subseteq\Re^q$ (for some positive integer q) and an unknown probability distribution. In addition to the sample observations above, we are also given a function $L(\boldsymbol{\beta}, Z_i)$, where $L:\Re^p\times\mathcal{W}\to\Re$ measures the statistical loss with respect to the data point Z_i and the vector of fitting parameters $\boldsymbol{\beta}:=(\beta_j)\in\Re^p$. Here, the positive integer p is called the problem dimensionality (which is equal to the number of fitting parameters). Throughout this paper, we assume that L is measurable and deterministic, the expectation $\mathbb{E}[L(\boldsymbol{\beta},\mathcal{Z})]$ over $\mathcal{Z}$ is well defined for all $\boldsymbol{\beta}\in\Re^p$, and $\inf_{\boldsymbol{\beta}}\mathbb{E}[L(\boldsymbol{\beta},\mathcal{Z})]>-\infty$. Although no convexity assumption is imposed explicitly, many of our results are mainly useful when $L(\cdot, z)$ is convex. Given the previously stated setup, it is often essential to estimate the solution to the following *population-level problem* in many applications:

$$\boldsymbol{\beta}^*\in\arg\inf_{\boldsymbol{\beta}\in\Re^p}\left\{\mathbb{L}(\boldsymbol{\beta}):=\mathbb{E}[L(\boldsymbol{\beta},\mathcal{Z})]\right\}. \tag{1}$$

Here, $\boldsymbol{\beta}^*$ is intuitively the vector of fitting parameters that yields the smallest population-level statistical loss (a.k.a., population risk). Therefore, $\boldsymbol{\beta}^*$ is considered the target of estimation and referred to as the vector of "true parameters." The HDSL problem of interest is then how to estimate (or approximate) $\boldsymbol{\beta}^*$, given the *a priori* knowledge of the samples $\mathbf{Z}_1^n:=(Z_1,Z_2,\ldots,Z_n)$ and the formulation of L, when $p\ge n$. We are especially interested in the more challenging case where the sample size n is much smaller than the dimensionality p (i.e., $p\gg n$). In measuring the approximation quality (a.k.a., recovery quality) of an estimator $\boldsymbol{\beta} \in \Re^p$, we consider a metric of generalization error calculated as $\mathbb{L}(\widehat{\boldsymbol{\beta}}) - \inf_{\boldsymbol{\beta}} \mathbb{L}(\boldsymbol{\beta})$. This metric is the same as the *excess risk*, which is discussed by Bartlett et al. (2006), Koltchinskii (2010), and Clémençon et al. (2008), among others, as an important, if not the primary, measure of generalization performance for their results.

Most traditional schemes are not applicable to the HDSL. For example, one popularly adopted scheme is to construct a surrogate for the population-level formulation in (1) through the sample average approximation (SAA):

$$\boldsymbol{\beta}^{SAA} \in \arg\inf_{\boldsymbol{\beta}} \left\{ \mathscr{L}_n(\boldsymbol{\beta}, \mathbf{Z}_1^n) := \frac{1}{n}\sum_{i=1}^{n} L(\boldsymbol{\beta}, Z_i) \right\}, \tag{2}$$

where the objective function $\mathscr{L}_n(\boldsymbol{\beta}, \mathbf{Z}_1^n)$ is often also called the *empirical risk function* in the context of statistical and machine learning. The SAA entails desirable computational and statistical properties (many of which are discussed by Shapiro et al. 2014 and references therein) but is not designed for handling high dimensionality. Indeed, the best known upper bound on the approximation error of the SAA solution is of the order $\mathscr{O}(\sqrt{p/n})$, where $\mathscr{O}(\cdot)$ hides some quantities independent of, or poly-logarithmic in, "$\cdot$". Consequently, the *estimator* of the true parameters generated by solving the SAA, as well as by most other traditional statistical learning approaches, may incur nontrivial errors when $p \gg n$.

To address high dimensionality, several statistical schemes are available. (See Bühlmann and van de Geer 2011 and Fan et al. 2014 for excellent reviews.) Among them, this paper follows and generalizes one of the most successful HDSL techniques introduced by Fan and Li (2001) and Zhang (2010) as in the following formulation:

$$\inf_{\boldsymbol{\beta} \in \Re^p} \left\{ \mathscr{L}_{n,\lambda}(\boldsymbol{\beta}, \mathbf{Z}_1^n) := \mathscr{L}_n(\boldsymbol{\beta}, \mathbf{Z}_1^n) + \sum_{j=1}^{p} P_\lambda(|\beta_j|) \right\}, \tag{3}$$

where $P_\lambda : \Re_+ \to \Re_+$ is a term of sparsity-inducing regularization in the form of a *folded concave penalty* (FCP). One mainstream special case of the existing FCPs, called the *minimax concave penalty* (MCP) (Zhang 2010), is of particular consideration. The MCP is formulated as

$$P_\lambda(\theta) = \int_0^\theta \frac{[a\lambda - t]_+}{a} dt, \qquad \theta \geq 0, \tag{4}$$

with $[\cdot]_+ := \max\{0, \cdot\}$ and tuning parameters $a, \lambda > 0$. (Hereafter, we use the term FCP to refer to the MCP exclusively.) Equation (3) is nonconvex, to which the local and/or global solutions have been shown to entail desirable statistical properties (Zhang and Zhang 2012; Wang et al. 2013, 2014; Loh and Wainwright 2015; Loh 2017). To understand the roles of the tuning parameters a and λ to the FCP, we may observe that its first derivative, $P'_\lambda(\theta)$, is a nonincreasing function with $P'_\lambda(0) = \lambda$ and $P'_\lambda(\theta) = 0$ for all $\theta \geq a\lambda$. This means that λ determines how intense the penalty is to induce a fitting parameter that is almost zero to be exactly zero. The effect of this penalty becomes smaller as the magnitude of the corresponding fitting parameter increases. Once the absolute value of that parameter is beyond the threshold $a\lambda$, the penalty becomes a constant and thus (locally) ineffective. Furthermore, we also observe that $P''_\lambda(\theta) = -\frac{1}{a}$ for all $\theta \in (0, a\lambda)$ and $P''_\lambda(\theta) = 0$ for all $\theta > a\lambda$. Therefore, a determines the curvature of the FCP near the origin.

Alternative sparsity-inducing penalties, such as the smoothly clipped absolute deviation (SCAD) introduced by Fan and Li (2001), the least absolute shrinkage and selection operator (Lasso) proposed by Tibshirani (2011), and the bridge penalty (a.k.a., the $\ell_{\mathbf{q}}$ penalty with $0 < \mathbf{q} < 1$) as discussed by Frank and Friedman (1993), have all been shown to be very effective in HDSL by many results from Fan and Li (2001), Bickel et al. (2009), Fan and Lv (2011), Fan et al. (2014), Loh and Wainwright (2015), Raskutti et al. (2011), Negahban et al. (2012), Wang et al. (2013, 2014), Zhang and Zhang (2012), Zou (2006), Zou and Li (2008), Liu et al. (2017, 2019), and Loh (2017), to name only a few. Many of those results provide *oracle inequalities*, which "relates the performance of a real estimator with that of an ideal estimator" (Candes 2006, p. 278). Ndiaye et al. (2017), Ghaoui et al. (2010), Fan and Li (2001), Chen et al. (2010), and Liu et al. (2017) have presented thresholding rules and bounds on the number of nonzero dimensions for a high-dimensional linear regression problem with different penalty functions.

Despite the availability of several analytical frameworks for HDSL in the current literature, most existing HDSL theories require the following two assumptions, which are sometimes overly critical, to guarantee any generalization performance.

A. *The satisfaction of the (conventional) sparsity condition, written as $\|\boldsymbol{\beta}^*\|_0 \ll p$, where $\|\cdot\|_0$ denotes the number of nonzero entries of a vector.*

B. *The satisfaction of regularity conditions on the eigenvalues of the Hessian matrix of $L(\cdot, \mathcal{Z})$ in the form of the restricted strong convexity (RSC) (Negahban et al. 2012), the restricted isotropic property (RIP) (Candes and Tao 2007), or the restricted eigenvalue (RE) condition (Bickel et al. 2009).*

The sparsity assumption essentially means that few dimensions "matter" despite that the total number of dimensions is very high. Meanwhile, the RSC, RIP, and RE can all be interpretable as the stipulation that

$\mathscr{L}(\cdot,\mathbf{Z}_1^n)$ is *strongly convex* everywhere in some subset of $\Re^p$. The RSC is implied by the RE and RIP for some choices of parameters (van de Geer et al. 2009, Negahban et al. 2012). Except for some special cases of the generalized linear models (Bickel et al. 2009), when both Assumptions A and B mentioned previously are violated, little is known about the performance of (3) or that of most other HDSL schemes in terms of their generalization performance in general. Negahban et al. (2012) has considered HDSL under weak sparsity, but the RSC is still assumed for establishing the generalization error bounds.

In contrast to the literature, this paper is concerned with the effectiveness of (3) in addressing the HDSL problems when the RSC is completely absent, and the traditional sparsity is relaxed into the approximate sparsity (A-sparsity) in the following.

Assumption 1. *The A-sparsity holds; that is,* $\mathbb{L}(\boldsymbol{\beta}^*_{\varepsilon_A})-\inf_{\boldsymbol{\beta}}\mathbb{L}(\boldsymbol{\beta})\le\varepsilon_A$ *and* $s:=\|\boldsymbol{\beta}^*_{\varepsilon_A}\|_0\ll p$ *for some* $\varepsilon_A\ge 0$, $\boldsymbol{\beta}^*_{\varepsilon_A}:\|\boldsymbol{\beta}^*_{\varepsilon_A}\|_\infty\le R$, *and* $R\ge 1$.

Intuitively, Assumption 1 means that, although $\boldsymbol{\beta}^*$ can be dense, replacing most of the nonzero entries of $\boldsymbol{\beta}^*$ by zero does not cause the population risk to increase too much. It is evident that, if $\varepsilon_A=0$, Assumption 1 is reduced to the (traditional) sparsity.

In certain applications of HDSL (e.g., the deep neural networks to be discussed subsequently), it is more convenient to consider a (slight) generalization to Assumption 1 in the following.

Assumption 2. *There exists* $L_g^*:L_g^*\le\inf_{\boldsymbol{\beta}}\mathbb{L}(\boldsymbol{\beta})$ *such that* $\mathbb{L}(\boldsymbol{\beta}^*_{\varepsilon_A})-L_g^*\le\varepsilon_A$ *and* $s:=\|\boldsymbol{\beta}^*_{\varepsilon_A}\|_0\ll p$ *for some* $\varepsilon_A\ge 0$, $\boldsymbol{\beta}^*_{\varepsilon_A}:\|\boldsymbol{\beta}^*_{\varepsilon_A}\|_\infty\le R$, *and* $R\ge 1$.

Apparently, Assumption 2 is more general than Assumption 1, and the two are equivalent when $L_g^*=\inf_{\boldsymbol{\beta}}\mathbb{L}(\boldsymbol{\beta})$. Hereafter, both Assumptions 1 and 2 are referred to as A-sparsity when there is no ambiguity. Without loss of generality, we let $s>1$ throughout this paper.

The assumption of $\|\boldsymbol{\beta}^*_{\varepsilon_A}\|_\infty\le R$ is noncritical. It is comparable to, if not less restrictive than, some common assumptions in the literature. For example, in addressing HDSL under (the conventional) sparsity, Loh (2017) and Loh and Wainwright (2015) both assume the estimator and the vector of true parameters to be contained within a convex and bounded set of $\{\boldsymbol{\beta}:|\boldsymbol{\beta}|\le R_{\ell_1}\}$ for some $R_{\ell_1}>0$. Verifiably, under their assumptions, $\|\boldsymbol{\beta}^*_{\varepsilon_A}\|_\infty\le R$ holds with some $R\le R_{\ell_1}$. Furthermore, we later show that our generalization error bounds depend only logarithmically on R. Thus, it is flexible to pick the value of R in practice; we only need to have a coarse estimation of an upper bound on $\|\boldsymbol{\beta}^*_{\varepsilon_A}\|_\infty$. Even if R overestimates $\|\boldsymbol{\beta}^*_{\varepsilon_A}\|_\infty$ too much, the performance of the proposed scheme would probably not be impacted significantly.

We believe that the flexibility of A-sparsity and the relaxation of the RSC can allow the HDSL theories to cover a more comprehensive class of applications. Indeed, as we are to articulate later, our results on HDSL under A-sparsity can facilitate the comprehension of two important classes of problems whose theoretical underpinnings are currently lacking from the literature: (i) a high-dimensional nonsmooth learning problem (nonsmooth HDSL), that is, an HDSL problem with a nonsmooth empirical risk function, and (ii) a (deep and over parameterized) neural network (NN) model.

More general forms of sparsity, such as the weak sparsity assumption (Negahban et al. 2012), have been discussed previously. However, the only existing discussions on simultaneously relaxing both the sparsity and the RSC assumptions are from Liu et al. (2019) to our knowledge. Their results imply that the excess risk of an estimator $\widehat{\boldsymbol{\beta}}\in\Re^p$ generated as a certain stationary point to Formulation (3) can be bounded by

$$\mathscr{O}\left(\frac{\sqrt{\ln p}}{n^{1/4}}\cdot(1+\sqrt{\varepsilon_A})+\varepsilon_A\right).$$

This bound is reduced to

$$\mathscr{O}\left(\frac{\sqrt{\ln p}}{n^{1/4}}\right)$$

when $\varepsilon_A=0$. In contrast, our findings in the current paper can strengthen the previous results. More specifically, we relax the subgaussian assumption stipulated by Liu et al. (2019) and impose the weaker, subexponential, condition instead. In addition, the assumption of twice-differentiability made by Liu et al. (2019) is also weakened. In the more general settings, we further show that sharper error bounds can be achieved at a stationary point that (a) satisfies a set of significant subspace second-order necessary conditions (S^{3}ONC) to be formalized subsequently and (b) has an objective function value no worse than that of the solution to the Lasso problem, formulated as follows:

$$\min_{\boldsymbol{\beta}\in\Re^p}\left\{\mathscr{L}_n(\boldsymbol{\beta},\mathbf{Z}_1^n)+\sum_{j=1}^p\lambda\cdot|\beta_j|\right\}.\tag{5}$$

We discuss some S^{3}ONC-guaranteeing algorithms to meet the first requirement soon afterward. To meet the second requirement, we may always initialize the S^{3}ONC-guaranteeing algorithm with a solution to (5), which is often polynomial-time solvable if $\mathscr{L}_n(\cdot,\mathbf{Z}_1^n)$ is convex.

Our new bounds on those S^{3}ONC solutions are summarized here. First, in the case where $\varepsilon_A=0$, we can bound the excess risk by

$$\mathscr{O}\left(\frac{\ln p}{n^{2/3}}+\frac{\sqrt{\ln p}}{n^{1/3}}\right),$$

which is better than the aforementioned result by Liu et al. (2019) in terms of the dependance on n.

Second, when ε_A is nonzero, the excess risk is then bounded by

$$\mathscr{O}\left(\frac{\ln p}{n^{2/3}}+\frac{\sqrt{\ln p}}{n^{1/3}}+\sqrt{\frac{\varepsilon_A}{n^{1/3}}+\varepsilon_A}\right). \tag{6}$$

Third, if we further relax the previous requirement and consider an arbitrary S^{3}ONC solution, then the excess risk becomes

$$\mathscr{O}\left(\frac{\ln p}{n^{2/3}}+\sqrt{\frac{\ln p}{n}}+\frac{1}{n^{1/3}}+\sqrt{\frac{\Gamma+\varepsilon_A}{n^{1/3}}+\Gamma+\varepsilon_A}\right), \tag{7}$$

where $\Gamma \geq 0$ is (an underestimation of) the suboptimality gap that this S^{3}ONC solution incurs in minimizing $\mathscr{L}_{n,\lambda}(\cdot, \mathbf{Z}_1^n)$ (as defined in (3)).

Admittedly, our excess risk bounds are less appealing than the generalizability results made available in some important previous works by Loh (2017), Raskutti et al. (2011), and Negahban et al. (2012), and so on, under the assumption of the RSC. In contrast, we argue that our results are established under a more general set of conditions and can complement the existing results in the HDSL problems beyond the RSC. It is also worth noting that (7) is in the parameterization of Γ, which can only be explicitly controlled when $\mathscr{L}_n(\cdot, \mathbf{Z}_1^n)$ is convex in general. Nonetheless, we argue that, in some interesting special cases, one may still control Γ despite the absence of convexity. One of such examples is presented in this paper as we discuss the theoretical applications of HDSL under A-sparsity to the NNs in Sections 6 and EC.1 of the e-companion.

The S^{3}ONC is a necessary condition for local minimality. Compared with the second-order Karush-Kuhn-Tucker (KKT) conditions, the S^{3}ONC is weaker and potentially easier computable. To generate a solution that satisfies the S^{3}ONC admits pseudo-polynomial-time algorithms, such as the variants of Newton's method proposed by Haeser et al. (2019), Bian et al. (2015), Ye (1992, 1998), and Nesterov and Polyak (2006). All those algorithms provably ensure a γ_{opt}-approximation (with a user-specified error tolerance $\gamma_{opt} > 0$) to the second-order KKT conditions at the best-known iteration complexity of the rate $\mathscr{O}(1/\gamma_{opt}^3)$. The second-order KKT conditions then imply the S^{3}ONC. To add to the current solution schemes, we derive a new gradient-based method that provably guarantees the S^{3}ONC. In contrast to the literature, the iteration complexity of this new algorithm is $\mathscr{O}(1/\gamma_{opt}^2)$, which improves on the existing alternatives. Because of the gradient-based nature of the proposed algorithm, it does not access the Hessian matrix or its inverse. Therefore, we think that this gradient-based algorithm may be of some independent interest.

1.1. Some Theoretical Applications

As mentioned, our results on HDSL under A-sparsity can be used in the analysis of two important classes of statistical and machine learning models: (a) nonsmooth HDSL and (b) deep NNs. We provide some additional details are provided in this subsection.

1.1.1. Nonsmooth HDSL. Although several special cases of HDSL with nonsmoothness, such as high-dimensional least absolute regression, high-dimensional quantile regression, and high-dimensional support vector machine (SVM) have been discussed by Wang (2013), Belloni and Chernozhukov (2011), Zhang et al. (2016b, c), and Peng et al. (2016), there exist few theories that apply to scenarios without an everywhere differentiable loss function in general, especially when nondifferentiability may occur at, or in a near neighborhood of, the vector of true parameters.

In contrast, our theories on HDSL under A-sparsity can be used to understand the generalization performance of a flexible set of nonsmooth HDSL problems. Indeed, their nonsmooth statistical loss functions can be approximated by another formulation that preserves the continuous differentiability, and the resulting approximation error can be handled through the notion of A-sparsity. Analyzing this approximation leads to the following bound on the excess risk at an S^{3}ONC solution when the vector of true parameters is A-sparse in the sense of Definition 1:

$$\mathscr{O}\left(\frac{\ln p}{n^{3/4}}+\frac{\sqrt{\ln p}}{n^{1/4}}+\sqrt{\frac{\varepsilon_A}{n^{1/4}}+\varepsilon_A}\right). \tag{8}$$

In particular, under the conventional sparsity assumption (i.e., when $\varepsilon_A = 0$), the previous rate becomes

$$\mathscr{O}\left(\frac{\ln p}{n^{3/4}}+\frac{\sqrt{\ln p}}{n^{1/4}}\right).$$

To our knowledge, this is perhaps the first generic theory for the high-dimensional M-estimation problems in which the empirical risk function may not be everywhere differentiable.

1.1.2. Regularized NN. The NNs have been frequently discussed and widely applied in recent literature (LeCun et al. 2015, Schmidhuber 2015, Yarotsky 2017). Despite the frequent and exciting advancements in the NN-related algorithms, models, and applications, the development of their theoretical underpinnings is seemingly lagging behind. DeVore et al. (1989), Yarotsky (2017), Mhaskar and Poggio (2016), and Mhaskar (1996), and so on, have explicated the expressive power of the NNs in the approximation of different types of functions. As for the generalizability of NNs, one of the focuses of this paper, effective theoretical frameworks have been discussed by Cao and Gu (2019), Li and Liang (2018), Brutzkus et al. (2017), Allen-Zhu et al. (2019), Wang et al. (2019), Daniely (2017), Neyshabur et al. (2015), Bartlett et al. (2017), Hardt et al. (2016), Zhang et al. (2016a), Li et al. (2018), and Jakubovitz et al. (2019), among others. However, for the vast majority of the existing results on the deep NNs, the generalization error bounds grow polynomially in the dimensionality (which is equal to the number of fitting parameters and is also called the network size) and sometimes even increase exponentially in the depth of the network. Such a high sensitivity to dimensionality and depth is inconsistent with the empirical performance of the NNs in many practical applications, where overparameterization and deep architectures are common and often preferred by practitioners.

In contrast, we analyze the NNs through the lens of HDSL under A-sparsity and consider an FCP-regularized NN training formulation as a special case of (3) in binary classification. Our results indicate that the NN's generalization errors at local solutions can be both poly-logarithmic in the number of fitting parameters and polynomial in the network depth. Thus, we think that the results herein can facilitate understanding the powerful performance of the NNs in practice, especially for the overparameterized and deep models. Barron and Klusowski (2018) have shown the existence of fitting parameters for an NN with ramp activation functions to achieve the poly-logarithmic sample complexity. Compared with Barron and Klusowski (2018), our analysis may present better flexibility in the choice of activation functions and provide more insights toward the computability of the desired fitting parameters in training a deep NN to ensure the proven error bounds.

More specifically, we show that the generalization error incurred by an S^{3}ONC solution to the FCP-regularized training formulation of an NN is bounded by

$$\mathscr{O}\left(\underbrace{\frac{s_A\cdot\mathscr{D}\cdot\ln p}{n^{2/3}}+\sqrt{\frac{s_A\cdot\mathscr{D}\cdot\ln p}{n}}+\frac{1}{n^{1/3}}}_{\mathscr{O}(n^{-1/3}+n^{-1/2}\mathscr{D}\cdot\ln p)}+\underbrace{\Gamma}_{\text{Suboptimality gap}}+\underbrace{\Omega(s_A)}_{\text{Representability gap}}+\underbrace{\sqrt{\frac{\Gamma+\Omega(s_A)}{n^{1/3}}}}_{\text{Interaction term}}\right), \tag{9}$$

for any fixed $s_A : 1\leq s_A\leq p$, with overwhelming probability. Here, $\mathscr{D}$ is the number of NN layers, $\Gamma\geq 0$ is the suboptimality gap incurred by the S^{3}ONC solution of consideration, and $\Omega(p')$, for any $p' : 1\leq p'\leq p$, is the architecture-dependent representability gap (a.k.a., the model misspecification error or the expressive power) of an NN with p'-many nonzero fitting parameters. By (9), the generalization error of an NN consists of four terms: (i) a generalization error term of the order $\mathscr{O}(n^{-1/3}+n^{-1/2}\mathscr{D}\ln p)$; (ii) the suboptimality gap; (iii) a term that measures the NN's representability; and (iv) a term that is dependent on suboptimality gap, sample size, and representability, simultaneously. It is worth noting that (9) is obtained with little restriction on the NN architecture and the data generation process. Combining (9) with the existing results on the representability analysis of NNs, we further derive more explicit generalization error bounds. For example, we show that the error yielded by an NN with smooth activation functions can be bounded by

$$\mathscr{O}\left(\frac{\mathscr{D}\cdot\ln p}{n^{1/3}}+\sqrt{\frac{\Gamma}{n^{1/3}}}+\Gamma\right),$$

when we assume that data from different categories are separable by a polynomial function (as well as a couple of other conditions on the NN architecture).

The error bound in (9) depends on Γ, the suboptimality gap. To explicitly bound its value is challenging in general because of the nonconvexity of an NN's training formulation. Nonetheless, we show that some pseudo-polynomial-time computable solutions generated with the aid of an efficient initialization provably ensure the explicit control of Γ in the same settings considered by Cao and Gu (2020). In such a case, the generalization error is further explicated into

$$\mathscr{O}\left(\frac{\mathscr{D}}{n^{1/3}}\cdot\ln p\right), \tag{10}$$

which becomes independent of Γ. In achieving this result, our settings seem more general than Wang et al. (2019), and our rates on both $\mathscr{D}$ and p are perhaps more appealing than most of the existing results. In particular, Wang et al. (2019) focus on ReLU-NNs (that is, the NNs where the activation functions are ReLU, as discussed by Glorot et al. 2011) with one hidden layer, but our approach can handle deep NNs under more general hyper-parameters. For deep and wide NNs, Cao and Gu (2020) have established generalization error bounds, which, however, increase exponentially in the number of layers in the same settings of our discussion. In contrast, our bound is both poly-logarithmic in dimensionality and polynomial in the number of layers. The computational complexity of training an NN with the claimed error bound is in pseudo-polynomial time.

In obtaining our results, we do not artificially impose any condition on sparsity or alike. As we articulate in Section 6.2, our findings are based on the observation that the A-sparsity (as in Assumption 2) is an intrinsic property implied by the NN's expressive power.

1.2. Summary of Results

Table 1 summarizes the sample complexity results proven in this paper. In contrast to the literature, we claim that our results could lead to the following contributions:

1. We provide the first HDSL theory for problems where the three conditions—the twice-differentiability,

Table 1. Summary of Sample Complexities

Type of solutions	Complexity results
HDSL under A-sparsity	
S^{3}ONC initialized with Lasso	$\frac{\ln p}{n^{2/3}}+\frac{\sqrt{\ln p}}{n^{1/3}}+\sqrt{\frac{\varepsilon_A}{n^{1/3}}}+\varepsilon_A$
S^{3}ONC with suboptimality gap Γ	$\frac{\ln p}{n^{2/3}}+\sqrt{\frac{\ln p}{n}}+\frac{1}{n^{1/3}}+\sqrt{\frac{\Gamma+\varepsilon_A}{n^{1/3}}}+\Gamma+\varepsilon_A$
Nonsmooth HDSL under A-sparsity	
S^{3}ONC initialized with Lasso	$\frac{\ln p}{n^{3/4}}+\frac{\sqrt{\ln p}}{n^{1/4}}+\sqrt{\frac{\varepsilon_A}{n^{1/4}}}+\varepsilon_A$
Neural network (with $\mathscr{D}$-many layers and p-many fitting parameters)	
S^{3}ONC to a general NN with suboptimality gap Γ and any $s_A : 1 \le s_A \le p$	$\frac{s_A\cdot\mathscr{D}\cdot\ln p}{n^{2/3}}+\sqrt{\frac{s_A\cdot\mathscr{D}\cdot\ln p}{n}}+\frac{1}{n^{1/3}}+\Omega(s_A)+\Gamma+\sqrt{\frac{\Gamma+\Omega(s_A)}{n^{1/3}}}$
S^{3}ONC to an NN for a flexible choice of activation functions with suboptimality gap Γ, when the target function is polynomial	$\frac{\mathscr{D}}{n^{1/3}}\cdot\ln p+\sqrt{\frac{\Gamma}{n^{1/3}}}+\Gamma$
A pseudo-polynomial-time computable solution in training a ReLU-NN in the same settings by Cao and Gu (2020)	$\frac{\mathscr{D}}{n^{1/3}}\cdot\ln p$

Note. ε_A denotes a parameter for A-sparsity as in Assumption 1; p and n, sample size and the dimensionality, respectively; ReLU-NN, NN with ReLU activation.

the RSC or alike, and the sparsity—are simultaneously relaxed. In the more general settings, we show that HDSL is still possible even if the sample size is only poly-logarithmic in the dimensionality. In Table 1, the results are presented in the rows for "HDSL under A-sparsity".

2. We have derived a pseudo-polynomial-time gradient-based method to compute an S^{3}ONC solution. Even though the S^{3}ONC is a set of second-order necessary conditions, the proposed algorithm does not need to access the Hessian matrix. Furthermore, the iteration complexity of the proposed method is provably $\mathscr{O}(\frac{1}{\gamma_{opt}^2})$ in achieving a γ_{opt}-approximation to the S^{3}ONC, which is sharper than the more generic algorithms such as the variations of Newton's method.

3. As theoretical applications of our error bounds for HDSL under A-sparsity, we derive generalizability results for nonsmooth HDSL problems and deep NNs. More specifically, for a flexible class of high-dimensional nonsmooth M-estimation problems, we prove perhaps the first poly-logarithmic sample complexity bound without the RSC assumption. The corresponding result is summarized in Table 1 in the rows for "Nonsmooth HDSL under A-sparsity." As for the NNs, our sample requirement is only poly-logarithmic in the network size and polynomial in the number of layers, providing theoretical underpinnings for the generalizability of an NN under overparameterization. These results are summarized in the rows for "Neural Network" of Table 1.

1.3. Organization of the Paper

The rest of the paper is organized as follows. Section 2 summarizes the settings and assumptions. Section 3 introduces the S^{3}ONC. Section 4 states our main results concerning HDSL under A-sparsity. A pseudo-polynomial-time solution scheme that guarantees the S^{3}ONC is discussed in Section 5. Section 6 discusses the theoretical applications to nonsmooth HDSL and the regularized (deep) NNs. Some numerical experiments are presented in Section 7. Sections EC.1 and EC.2 of the e-companion, respectively, present some additional theoretical results on the NN and supplementary numerical results on both the SVM and the NN. Section 8 concludes the paper.

Our notations are summarized here. We use p and n to represent the numbers of dimensions (fitting parameters) and the sample size. We let $\|\cdot\|_{\mathbf{p}}$ $(1 \le \mathbf{p} \le \infty)$ be the $\mathbf{p}$-norm, except that 1- and 2-norms are denoted by $|\cdot|$ and $\|\cdot\|$, respectively. When there is no ambiguity, we also denote by $|\cdot|$ the cardinality of a set, if the argument is a finite set. Let $\|\cdot\|_F$ of a matrix be its Frobenius norm and let $\|\cdot\|_0$ of a vector be the number of its nonzero entries. For a random vector $\mathbf{v} = (v_j) \in \Re^p$, we denote that $\|\mathbf{v}\|_\infty \le R$ if $\mathbb{P}[|v_j| \le R,\ \forall j = 1,\ldots,p] = 1$. For a random variable X, its subexponential and subgaussian norms are denoted by $\|X\|_{\psi_1}$ and $\|X\|_{\psi_2}$, respectively. $\|\mathbf{A}\|_{1,2} := \max_{\mathbf{x}\in\Re^{m_1},\mathbf{u}\in\Re^{m_2}}\{\mathbf{u}^\top\mathbf{A}\mathbf{x} : \|\mathbf{x}\|_1 = 1, \|\mathbf{u}\|_2 = 1\}$ for integers m_1, m_2 and a matrix $\mathbf{A} \in \Re^{m_2\times m_1}$. For a function f, denote by ∇f its gradient, whenever it exists. For a vector $\boldsymbol{\beta} = (\beta_j) \in \Re^p$ and a set $S \subset \{1,\ldots,p\}$, let $\boldsymbol{\beta}_S = (\beta_j : j \in S)$ be a subvector of $\boldsymbol{\beta}$. For any vector $\mathbf{v} = (v_j)$, the notation $diag(\mathbf{v})$ represents the diagonal matrix whose jth diagonal entry is v_j. We denote by $vec(M_1,M_2,\ldots,M_m)$ the vector that collects all the entries of the matrices $M_1, M_2, \ldots, M_m$. The vector e_j is the jth standard basis. $\lceil x\rceil$ (or $\lfloor x\rfloor$) for any $x \ge 0$ is the smallest (or largest) integer that is greater (or smaller, respectively) than or equal to x. Finally, we denote by $O(\cdot)$ s and $\mathscr{O}(\cdot)$ s, respectively, the complexity rates that hide (potentially different) universal constants and quantities at most logarithmically dependent on "$\cdot$"

2. Settings and Assumptions

In this section, we summarize our assumptions in addition to the aforementioned settings. We assume that the gradient $\nabla L(\boldsymbol{\beta}, z) := \left(\frac{\partial L(\boldsymbol{\beta}, z)}{\partial \beta_j} : j = 1, \ldots, p\right)$ of $L(\boldsymbol{\beta}, z)$ with respect to (w.r.t.) $\boldsymbol{\beta}$ is well defined for all $\boldsymbol{\beta} \in \Re^p$ and almost every $z \in \mathscr{W}$. Furthermore, we also suppose that $\frac{\partial L(\boldsymbol{\beta}, z)}{\partial \beta_j}$ is Lipschitz continuous for all $\boldsymbol{\beta} \in \Re^p$; that is, there exists a scalar $U_L > 0$ such that

$$\left|\left[\frac{\partial L(\boldsymbol{\beta}, z)}{\partial \beta_j}\right]_{\boldsymbol{\beta} = \widetilde{\boldsymbol{\beta}} + \delta \cdot e_j} - \left[\frac{\partial L(\boldsymbol{\beta}, z)}{\partial \beta_j}\right]_{\boldsymbol{\beta} = \widetilde{\boldsymbol{\beta}}}\right| \leq U_L \cdot |\delta|, \qquad (11)$$

for almost every $z \in \mathscr{W}$ and for all $\widetilde{\boldsymbol{\beta}} \in \Re^p$, $\delta \in \Re$, $j = 1, \ldots, p$. These regularities are to be relaxed when we later discuss the nonsmooth HDSL problems and the ReLU-NNs. Apart from the previous assumptions, two additional assumptions are imposed as below.

Assumption 3. *For all $\boldsymbol{\beta} \in \Re^p : \|\boldsymbol{\beta}\|_\infty \leq R$ and $i = 1, \ldots, n$, it holds that $\mathbb{E}[L(\boldsymbol{\beta}, Z_i)]$ is finite-valued and $L(\boldsymbol{\beta}, Z_i) - \mathbb{E}[L(\boldsymbol{\beta}, Z_i)]$ follows a subexponential distribution; that is, $\|L(\boldsymbol{\beta}, Z_i) - \mathbb{E}[L(\boldsymbol{\beta}, Z_i)]\|_{\psi_1} \leq \sigma$, for some $\sigma \geq 1$.*

Remark 1. As an implication of Assumption 3, for all $\boldsymbol{\beta} \in \Re^p : \|\boldsymbol{\beta}\|_\infty \leq R$ (combined with the assumption that Z_i, $i = 1, \ldots, n$, are i.i.d.), a well-known Bernstein-like inequality holds as follows:

$$\mathbb{P}\left(\left|\sum_{i=1}^{n} a_i\{L(\boldsymbol{\beta}, Z_i) - \mathbb{E}[L(\boldsymbol{\beta}, Z_i)]\}\right| > \sigma \cdot \left(\|\mathbf{a}\|\sqrt{t} + \|\mathbf{a}\|_\infty t\right)\right)$$
$$\leq 2\exp(-ct), \quad \forall t \geq 0, \mathbf{a} = (a_i) \in \Re^n, \qquad (12)$$

for some absolute constant $c \in (0, 0.5]$. Interested readers are referred to Vershynin (2012) for more detailed discussions on the subexponential distributions.

Assumption 4. *For some measurable and deterministic function $\mathscr{C} : \mathscr{W} \rightarrow \Re_+$, the random variable $\mathscr{C}(Z_i)$ satisfies that $\|\mathscr{C}(Z_i) - \mathbb{E}[\mathscr{C}(Z_i)]\|_{\psi_1} \leq \sigma_L$, for all $i = 1, \ldots, n$, for some $\sigma_L \geq 1$. Furthermore, $|L(\boldsymbol{\beta}_1, z) - L(\boldsymbol{\beta}_2, z)| \leq \mathscr{C}(z)\|\boldsymbol{\beta}_1 - \boldsymbol{\beta}_2\|$, for all $\boldsymbol{\beta}_1, \boldsymbol{\beta}_2 \in \Re^p \cap \{\boldsymbol{\beta} : \|\boldsymbol{\beta}\|_\infty \leq R\}$ and almost every $z \in \mathscr{W}$.*

Hereafter, we let $\mathbb{E}[\mathscr{C}(Z_i)] \leq \mathscr{C}_\mu$ for all $i = 1, \ldots, n$ for some $\mathscr{C}_\mu \geq 1$.

Remark 2. Assumptions 3 and 4 are general enough to cover a wide spectrum of M-estimation problems. More specifically, Assumption 3 requires that the underlying distribution is subexponential, and Assumption 4 essentially imposes the Lipschitz(-like) continuity on $\mathscr{L}_n(\cdot, \mathbf{Z}_1^n)$. Examples of subexponential distributions include uniform, Gaussian, exponential, and χ^2 distributions, as well as any distribution that has a bounded support set. As for the Lipschitz continuity, it is a condition satisfied by many statistical learning problems, such as linear regression, Huber regression, SVM, and NNs. We are to show that the generalization error bounds only grow logarithmically in the Lipschitz constant. The combination of our assumptions is nontrivially weaker than the settings in Liu et al. (2017, 2019). It is also worth mentioning that the stipulations of $\sigma \geq 1$, $\mathscr{C}_\mu \geq 1$, and $\sigma_L \geq 1$ can be easily relaxed and are needed only for notational simplicity in presenting our results.

3. Significant Subspace Second-Order Necessary Conditions

Because the FCP is nonconvex, so is Equation (3). Thus, computing the global solution to (3) is intractable. Nonetheless, our theories concern only local stationary points. We show that these local solutions are good enough to ensure the promised statistical performance.

In particular, we consider the stationary points that are characterized by the satisfaction of the significant subspace S^{3}ONC, which are closely similar to the necessary conditions discussed by Chen et al. (2010) for linear regression with bridge regularization and by Liu et al. (2017, 2019) under the assumption that the empirical risk function is everywhere twice differentiable. This paper generalizes the characterizations of the S^{3}ONC to scenarios where the twice-differentiability may not hold everywhere.

Definition 1. Given $\mathbf{Z}_1^n \in \mathscr{W}^n$, a vector $\widehat{\boldsymbol{\beta}} \in \Re^p$ is said to satisfy the S^{3}ONC (denoted by $S^3ONC(\mathbf{Z}_1^n)$) of Problem (3) if both of the following sets of conditions are satisfied:

(a) The first-order KKT conditions are met at $\widehat{\boldsymbol{\beta}} := (\widehat{\beta}_j)$; that is, there exists $\kappa_j \in \partial(|\widehat{\beta}_j|)$, for all $j = 1, \ldots, p$, such that

$$\nabla \mathscr{L}_n(\widehat{\boldsymbol{\beta}}, \mathbf{Z}_1^n) + (P'_\lambda(|\widehat{\beta}_j|) \cdot \varkappa_j : j = 1, \ldots p) = \mathbf{0}, \qquad (13)$$

where $\nabla \mathscr{L}_n(\widehat{\boldsymbol{\beta}}, \mathbf{Z}_1^n)$ is the gradient of $\mathscr{L}_n(\cdot, \mathbf{Z}_1^n)$ as defined in (2), $\partial(|\widehat{\beta}_j|)$ is the subdifferential of $|\cdot|$ at $\widehat{\beta}_j$, and $P'_\lambda(\cdot)$ is the first derivative of $P_\lambda(\cdot)$.

(b) The following inequality holds at $\widehat{\boldsymbol{\beta}}$: for all $j = 1, \ldots, p$, if $|\widehat{\beta}_j| \in (0, a\lambda)$, then

$$U_L + P''_\lambda(|\widehat{\beta}_j|) \geq 0, \qquad (14)$$

where P''_λ is the second derivative of $P_\lambda(\cdot)$, the quantity U_L is defined as in (11), and a and λ are (hyper-)parameters of the FCP as in (4).

It is worth noting that the S^{3}ONC is verifiably implied by the conventional second-order KKT conditions when they are well defined. We show in Section 5 that an S^{3}ONC solution (i.e., a solution that satisfies the S^{3}ONC) can be computed by the proposed gradient-based method at pseudo-polynomial-time complexity.

4. Statistical Performance Bounds

This section presents the promised sample complexity results for a generic HDSL problem under A-sparsity. More specifically, Proposition 1 shows the most general result of this paper. In that proposition, a hyper-parameter ϱ is left to be determined in different special cases. One of those cases is then presented in Theorem 1. For convenience, we adopt a short-hand notation as follows: $\widetilde{\zeta} := \ln\big(3eR\cdot(\sigma_L + \mathscr{C}_\mu)\big)$.

Proposition 1. *Suppose that Assumptions 2–4 hold. For any $\varrho : 0 < \varrho < \frac{1}{2}$ and the same c in (12), let $a < \frac{1}{U_L}$ and $\lambda := \sqrt{\frac{8\sigma}{c\cdot a\cdot n^{2\varrho}}[\ln(n^{\varrho}p) + \widetilde{\zeta}]}$. Consider any random vector $\widehat{\boldsymbol{\beta}} \in \mathfrak{R}^p$ such that $\|\widehat{\boldsymbol{\beta}}\|_\infty \le R$ and the $S^3ONC(\mathbf{Z}_1^n)$ to (3) is satisfied at $\widehat{\boldsymbol{\beta}}$ almost surely. The following statements hold:*

(i) *For any fixed $\Gamma \ge 0$ and some universal constant* $C_1 > 0$, if

$$n > C_1 \cdot \left[\left(\frac{\Gamma + \varepsilon_A}{\sigma}\right)^{\frac{1}{1-2\varrho}} + s\cdot\Big(\ln(n^{\varrho}p) + \widetilde{\zeta}\Big)\right], \tag{15}$$

and $\mathscr{L}_{n,\lambda}(\widehat{\boldsymbol{\beta}}, \mathbf{Z}_1^n) \le \mathscr{L}_{n,\lambda}(\boldsymbol{\beta}^*_{\varepsilon_A}, \mathbf{Z}_1^n) + \Gamma$ almost surely, then

$$\begin{aligned}\mathbb{L}(\widehat{\boldsymbol{\beta}}) - L_g^* \quad &\le C_1 \cdot \left(\frac{s\cdot\Big(\ln(n^{\varrho}p) + \widetilde{\zeta}\Big)}{n^{2\varrho}} + \sqrt{\frac{s\cdot\Big(\ln(n^{\varrho}p) + \widetilde{\zeta}\Big)}{n}}\right.\\ &\quad \left. + \frac{1}{n^{\varrho}} + \frac{1}{n^{1-2\varrho}} + \frac{1}{n^{(1-\varrho)/2}}\right)\cdot\sigma + C_1\\ &\quad \cdot\sqrt{\frac{\sigma(\Gamma + \varepsilon_A)}{n^{1-2\varrho}}} + \Gamma + \varepsilon_A,\end{aligned} \tag{16}$$

with probability at least $1 - 2(p+1)exp(-n/C_1) - 6exp(-2cn^{4\varrho} - 1)$, where $\mathbb{L}$ is defined in Equation (1) and L_g^* is defined in Assumption 2.

(ii) *For almost every $\mathbf{Z}_1^n \in \mathscr{W}^n$, assume that the minimization problem in (5) admits a finite optimal solution denoted by $\widehat{\boldsymbol{\beta}}^{\ell_1} := \widehat{\boldsymbol{\beta}}^{\ell_1}(\mathbf{Z}_1^n)$. For some universal constant $C_2 > 0$, if*

$$\begin{aligned} n &> C_2 \cdot \left(\frac{\varepsilon_A}{\sigma}\right)^{\frac{1}{1-2\varrho}} + C_2\cdot a^{-1}\cdot[\ln(n^{\varrho}p) + \widetilde{\zeta}]\\ &\quad \cdot s^{\max\left\{1, \frac{1}{2-4\varrho}, \frac{1}{2\varrho}\right\}}\Big(\max\Big\{1, \|\boldsymbol{\beta}^*_{\varepsilon_A}\|_\infty\Big\}\Big)^{\max\left\{\frac{1}{2-4\varrho}, \frac{1}{2\varrho}\right\}},\end{aligned} \tag{17}$$

and $\mathscr{L}_{n,\lambda}(\widehat{\boldsymbol{\beta}}, \mathbf{Z}_1^n) \le \mathscr{L}_{n,\lambda}(\widehat{\boldsymbol{\beta}}^{\ell_1}, \mathbf{Z}_1^n)$ almost surely, then

$$\begin{aligned}\mathbb{L}(\widehat{\boldsymbol{\beta}}) - L_g^* &\le C_2 \cdot \left[\frac{s\Big(\ln(n^{\varrho}p) + \widetilde{\zeta}\Big)}{n^{2\varrho}} + \frac{1}{n^{\varrho}} + \frac{1}{n^{1-2\varrho}}\right]\cdot\sigma\\ &\quad + C_2 \cdot \frac{s\cdot\max\Big\{1, \|\boldsymbol{\beta}^*_{\varepsilon_A}\|_\infty\Big\}\cdot\sigma^{3/4}}{\min\Big\{a^{1/2}n^{\varrho}, a^{1/4}n^{\frac{1-\varrho}{2}}\Big\}}\Big[\ln(n^{\varrho}p) + \widetilde{\zeta}\Big]^{1/2}\\ &\quad + C_2\cdot\sqrt{\frac{\sigma\varepsilon_A}{n^{1-2\varrho}}} + \varepsilon_A,\end{aligned} \tag{18}$$

with probability at least $1 - 2(p+1)exp(-n/C_2) - 6exp(-2cn^{4\varrho} - 1)$.

Proof. See Section EC.5.1 of the e-companion. □

Remark 3. Proposition 1 is the most general result in this paper. It does not rely on convexity, RSC, or alike, although to ensure $\mathscr{L}_{n,\lambda}(\widehat{\boldsymbol{\beta}}, \mathbf{Z}_1^n) \le \mathscr{L}_{n,\lambda}(\widehat{\boldsymbol{\beta}}^{\ell_1}, \mathbf{Z}_1^n)$ almost surely in part (ii) usually requires $\mathscr{L}_{n,\lambda}(\cdot, \mathbf{Z}_1^n)$ to be convex.

Remark 4. The assumption that $\|\widehat{\boldsymbol{\beta}}\|_\infty \le R$ is comparable to, or less restrictive than, some similar conditions in the literature. For example, Loh (2017) and Loh and Wainwright (2015) require that the estimator is within the set of $\{\boldsymbol{\beta} : |\boldsymbol{\beta}| \le R_{\ell_1}\}$. Under the same requirement, we may have $R_{\ell_1} \ge R$. Because the error bounds in (15) and (18) are logarithmic in R (with $\widetilde{\zeta} := \mathscr{O}(\ln R)$), one may let the value of R to be a coarse overestimation of $\|\widehat{\boldsymbol{\beta}}\|_\infty$.

Remark 5. Because $\mathbb{L}(\widehat{\boldsymbol{\beta}}) - \inf_{\boldsymbol{\beta}} \mathbb{L}(\boldsymbol{\beta}) \le \mathbb{L}(\widehat{\boldsymbol{\beta}}) - L_g^*$, the first part of this proposition indicates that, for all the S^{3}ONC solutions, the excess risk can be bounded by a function in the parameterization of the suboptimality gap Γ. (Technically speaking, Γ is an underestimation of the suboptimality gap in this proposition.) This bound on the excess risk explicates the consistency between the statistical performance of a stationary point to an HDSL problem and the optimization quality of that stationary point in minimizing the objective function of Problem (3). The second part of Proposition 1 concerns an arbitrary S^{3}ONC solution $\widehat{\boldsymbol{\beta}}$ that has an objective function value smaller than that of $\widehat{\boldsymbol{\beta}}^{\ell_1}$. The corresponding error bound becomes independent of Γ.

Remark 6. To compute $\widehat{\boldsymbol{\beta}}$ in part (ii) of this proposition, we can adopt a two-step approach: In the first step, we solve for $\widehat{\boldsymbol{\beta}}^{\ell_1}$, which is often polynomial-time computable if $\mathscr{L}_{n,\lambda}(\cdot, \mathbf{Z}_1^n)$ is convex given $\mathbf{Z}_1^n$. Then, in the second step, we invoke an S^{3}ONC-guaranteeing algorithm (such as the gradient-based method to be discussed in Section 5). This algorithm should be initialized with $\widehat{\boldsymbol{\beta}}^{\ell_1}$.

Remark 7. We may as well let $a^{-1} = 2U_L$ to satisfy the stipulation on a in Proposition 1. Here, U_L can be considered as the largest diagonal of the Hessian matrix of $\mathscr{L}(\cdot, z)$, if it exists. In many applications of HDSL, this quantity can satisfy $U_L \le O(1)\ln p$ with high probability under data normalization. For example, in the special case of high-dimensional linear models, $U_L \le 1$ is implied by the common assumption of column normalization (Raskutti et al. 2011, Negahban et al. 2012).

Remark 8. The proof of Proposition 1 makes use of the coincidence that, at the S^{3}ONC solutions, the FCP behaves similarly as the ℓ_0 penalty (Shen et al. 2013).

Thus, it is possible that adopting the ℓ_0 penalty instead of the FCP in our Formulation (3) may lead to similar results on the generalization errors with less technical difficulty. Nonetheless, the ℓ_0 penalty introduces discontinuity to the formulation and thus may usually lead to higher computational ramification. We leave for the future research the study of the tradeoffs between computational and sample complexities for the formulations with alternative regularization terms.

Remark 9. For any fixed $\varrho: 0 < \varrho < \frac{1}{2}$, each of the two parts of Proposition 1 has already established the poly-logarithmic sample complexity. Based on this proposition, polynomially increasing the sample size can compensate for the exponential growth in the dimensionality. We may further pick a reasonable value for ϱ and obtain more detailed bounds as in Theorem 1, which confirms the promised complexity rates as previously mentioned in (6) and (7) for a general HDSL problem under A-sparsity.

Theorem 1. *Let $a < \frac{1}{U_L}$ and $\lambda := \sqrt{\frac{8\sigma}{c\cdot a\cdot n^{2/3}}[\ln(n^{2/3}p) + \widetilde{\zeta}]}$ for the same c in (12). Suppose that Assumptions 1, 3, and 4 hold. For any random vector $\widehat{\boldsymbol{\beta}} \in \Re^p$ such that $\|\widehat{\boldsymbol{\beta}}\|_\infty \le R$ and S^{3}ONC($\mathbf{Z}_1^n$) to (3) is satisfied at $\widehat{\boldsymbol{\beta}}$ almost surely, the following statements hold:*

(i) *For any fixed $\Gamma \ge 0$ and some universal constant $C_3 > 0$, if*

$$n > C_3 \cdot \left[\left(\frac{\Gamma + \varepsilon_A}{\sigma}\right)^3 + s\cdot\left(\ln(np) + \widetilde{\zeta}\right)\right], \tag{19}$$

and $\mathscr{L}_{n,\lambda}(\widehat{\boldsymbol{\beta}}, \mathbf{Z}_1^n) \le \mathscr{L}_{n,\lambda}(\boldsymbol{\beta}^*_{\varepsilon_A}, \mathbf{Z}_1^n) + \Gamma$ almost surely, then the excess risk is bounded by

$$\mathbb{L}(\widehat{\boldsymbol{\beta}}) - \inf_{\boldsymbol{\beta}} \mathbb{L}(\boldsymbol{\beta}) \le C_3\sigma \cdot \left[\frac{s\cdot\left(\ln(np)+\widetilde{\zeta}\right)}{n^{2/3}} + \sqrt{\frac{s\cdot\left(\ln(np)+\widetilde{\zeta}\right)}{n}} + \frac{1}{n^{1/3}}\right] + C_3\cdot\sqrt{\frac{\sigma(\Gamma+\varepsilon_A)}{n^{1/3}}} + \Gamma + \varepsilon_A \tag{20}$$

with probability at least

$$1 - 2(p+1)\exp\left(-\frac{n}{C_3}\right) - 6\exp\left(-\frac{n^{1/3}}{C_3}\right).$$

(ii) *For almost every $\mathbf{Z}_1^n \in \mathscr{W}^n$, assume that the minimization problem in (5) admits a finite optimal solution denoted by $\widehat{\boldsymbol{\beta}}^{\ell_1} := \widehat{\boldsymbol{\beta}}^{\ell_1}(\mathbf{Z}_1^n)$. For some universal constant $C_4 > 0$, if*

$$n > C_4\cdot\left(\frac{\varepsilon_A}{\sigma}\right)^3 + C_4\cdot a^{-1}\cdot[\ln(np) + \widetilde{\zeta}]\cdot s^{\frac{3}{2}}\max\left\{1, \|\boldsymbol{\beta}^*_{\varepsilon_A}\|_\infty^{\frac{3}{2}}\right\}, \tag{21}$$

and $\mathscr{L}_{n,\lambda}(\widehat{\boldsymbol{\beta}}, \mathbf{Z}_1^n) \le \mathscr{L}_{n,\lambda}(\widehat{\boldsymbol{\beta}}^{\ell_1}, \mathbf{Z}_1^n)$ almost surely, then the excess risk is bounded by

$$\mathbb{L}(\widehat{\boldsymbol{\beta}}) - \inf_{\boldsymbol{\beta}} \mathbb{L}(\boldsymbol{\beta}) \le C_4\cdot a^{-1/2}\cdot s\cdot\sigma \cdot\left[\frac{\left(\ln(np)+\widetilde{\zeta}\right)}{n^{\frac{2}{3}}} + \frac{\max\{1, \|\boldsymbol{\beta}^*_{\varepsilon_A}\|_\infty\}\cdot\sqrt{\ln(np)+\widetilde{\zeta}}}{n^{\frac{1}{3}}}\right] + C_4\cdot\sqrt{\frac{\sigma\varepsilon_A}{n^{1/3}}} + \varepsilon_A \tag{22}$$

with probability at least

$$1 - 2(p+1)\exp\left(-\frac{n}{C_4}\right) - 6\exp\left(-\frac{n^{1/3}}{C_4}\right).$$

Proof. Invoking Proposition 1 with $\varrho = \frac{1}{3}$ and noticing that Assumption 1 implies Assumption 2 with $L_g^* := \inf_{\boldsymbol{\beta}} \mathbb{L}(\boldsymbol{\beta})$, we obtain both parts of the desired results. □

Theorem 1 ensures the desired poly-logarithmic sample complexity for HDSL under A-sparsity. Our remarks concerning Proposition 1 also apply to Theorem 1, because the latter is a special case when $\varrho = \frac{1}{3}$ and $L_g := \inf_{\boldsymbol{\beta}} \mathbb{L}(\boldsymbol{\beta})$. We would like to point out that, if $\varepsilon_A = 0$, then A-sparsity is reduced to the conventional sparsity. In such a case, the excess risk in (22) is simplified into

$$\mathbb{L}(\widehat{\boldsymbol{\beta}}) - \inf_{\boldsymbol{\beta}} \mathbb{L}(\boldsymbol{\beta}) \le \mathscr{O}\left(\frac{\ln p}{n^{2/3}} + \frac{\sqrt{\ln p}}{n^{1/3}}\right).$$

5. S^{3}ONC-Guaranteeing Algorithm

This section presents a pseudo-polynomial-time S^{3}ONC-guaranteeing algorithm. For convenience, we consider a slightly more abstract optimization problem than (3) as follows:

$$\min_{\boldsymbol{\beta}:=(\beta_j)\in\Re^p} \widetilde{f}_\lambda(\boldsymbol{\beta}) := \widetilde{f}(\boldsymbol{\beta}) + \sum_{j=1}^p P_\lambda(|\beta_j|), \tag{23}$$

where $\widetilde{f}: \Re^p \to \Re$ is a continuously differentiable function with $\|\nabla\widetilde{f}(\boldsymbol{\beta}_1) - \nabla\widetilde{f}(\boldsymbol{\beta}_2)\| \le \widetilde{U}_{L,2}\cdot\|\boldsymbol{\beta}_1 - \boldsymbol{\beta}_2\|$ for some $\widetilde{U}_{L,2} \ge 1$ and all $\boldsymbol{\beta}_1, \boldsymbol{\beta}_2 \in \Re^p$. Consequently, the partial derivative $\frac{\partial\widetilde{f}(\boldsymbol{\beta})}{\partial\beta_j}$, for all $j = 1, \ldots, p$, is also globally Lipschitz continuous in the sense that $\left|\left[\frac{\partial\widetilde{f}(\boldsymbol{\beta})}{\partial\beta_j}\right]_{\boldsymbol{\beta}=\widetilde{\boldsymbol{\beta}}+\delta\cdot e_j} - \left[\frac{\partial\widetilde{f}(\boldsymbol{\beta})}{\partial\beta_j}\right]_{\boldsymbol{\beta}=\widetilde{\boldsymbol{\beta}}}\right| \le \widetilde{U}_{L,\infty}\cdot|\delta|$ for every $\widetilde{\boldsymbol{\beta}} \in \Re^p$, any $\delta \in \Re$, and some $1 \le U_{L,\infty} \le U_{L,2}$. (Note that U_L in (11) becomes $\widetilde{U}_{L,\infty}$ here.) The pseudo-code of the proposed algorithm is summarized in the following.

Algorithm 1 (S^{3}ONC-Guaranteeing Gradient-Based Algorithm)

Step 1. Fix parameters γ_{opt}, $\mathscr{M}$, λ, and a such that $a < \mathscr{M}^{-1}$. Initialize $k = 0$ and $\boldsymbol{\beta}^0 \in \Re^p$.

Step 2. Compute $\boldsymbol{\beta}^{k+\frac{1}{2}}$ by solving the following problem:

$$\boldsymbol{\beta}^{k+\frac{1}{2}} \in \arg\min_{\boldsymbol{\beta}} \langle\nabla\widetilde{f}(\boldsymbol{\beta}^k), \boldsymbol{\beta} - \boldsymbol{\beta}^k\rangle + \frac{\mathscr{M}}{2}\|\boldsymbol{\beta} - \boldsymbol{\beta}^k\|^2 + \sum_{j=1}^p P_\lambda'(|\beta_j^k|)\cdot|\beta_j|\,. \tag{24}$$

Step 3. Compute $\boldsymbol{\beta}^{k+1}$ by solving the following problem:

$$\boldsymbol{\beta}^{k+1} \in \arg\min_{\boldsymbol{\beta}} \langle \nabla\widetilde{f}(\boldsymbol{\beta}^{k+\frac{1}{2}}), \boldsymbol{\beta} - \boldsymbol{\beta}^{k+\frac{1}{2}} \rangle + \frac{\mathscr{M}}{2}\|\boldsymbol{\beta} - \boldsymbol{\beta}^{k+\frac{1}{2}}\|^2 + \sum_{j=1}^{p} P_\lambda(|\beta_j|). \tag{25}$$

Step 4. Algorithm terminates and outputs $\boldsymbol{\beta}^k$ if the stopping criteria are met. Otherwise, let $k := k+1$ and go to Step 2.

We design the termination criterion to be that the algorithm stops when the following is satisfied for the first time:

$$\widetilde{f}_\lambda(\boldsymbol{\beta}^{k+1}) > \widetilde{f}_\lambda(\boldsymbol{\beta}^k) - \frac{\gamma_{opt}^2}{2\mathscr{M}}, \tag{26}$$

where $\mathscr{M} > 0$ and $\gamma_{opt} > 0$ are specified in Step 1 of Algorithm 1. Intuitively, $\mathscr{M}^{-1}$ can be interpreted as the step size of the algorithm, and γ_{opt}, as the error tolerance in approximating the S^{3}ONC. At termination, the iteration count is denoted by k^*.

To our analysis, Algorithm 1 relies on solving two per-iteration subproblems (24) and (25), repetitively. Subproblem (24) in Step 2 ensures that a nontrivial reduction in the objective function value can be achieved whenever the first-order KKT conditions are not met. This step is essential to the promised $\mathscr{O}(1/\gamma_{opt}^2)$-rate of the algorithm. Meanwhile, the presence of Subproblem (25) in Step 3 leads to a solution sequence that approaches a desired S^{3}ONC solution without affecting the convergence rate. We may formalize the previous analysis to prove the following theorem on the iteration complexity of Algorithm 1 in computing an S^{3}ONC solution.

Theorem 2. *Suppose that* $\widetilde{f}_\lambda^* := \inf_{\boldsymbol{\beta}} \widetilde{f}_\lambda(\boldsymbol{\beta}) > -\infty$, $\mathscr{M} \geq \widetilde{U}_{L,2}$, *and* $a < \frac{1}{\mathscr{M}}$. *For any* $\gamma_{opt} : 0 < \gamma_{opt} < a\lambda \cdot \mathscr{M}$, *the following statements hold true:*

(a) *Algorithm* 1 *terminates at iteration* $k^* \leq \left\lfloor 2\mathscr{M} \cdot \frac{\widetilde{f}_\lambda(\boldsymbol{\beta}^0) - \widetilde{f}_\lambda^*}{\gamma_{opt}^2} \right\rfloor + 1.$

(b) *At termination,* $\boldsymbol{\beta}^{k^*} = (\beta_j^{k^*})$ *is a* γ_{opt}*-S*^{3}*ONC solution to* (23)*; that is, there exists* $\varkappa_j \in \partial(|\beta_j^{k^*}|)$*, for all* $j = 1, \ldots, p$*, such that*

$$\|\nabla\widetilde{f}(\boldsymbol{\beta}^{k^*}) + (P'_\lambda(|\beta_j^{k^*}|) \cdot \varkappa_j : j = 1, \ldots p)\| \leq \gamma_{opt}, \tag{27}$$

and, for all $j = 1, \ldots, p$*, if* $|\beta_j^{k^*}| \in (0, a\lambda)$*, then* $\widetilde{U}_{L,\infty} + P''_\lambda(|\beta_j^{k^*}|) \geq 0$*, where* a *and* λ *are defined in* (4).

(c) *At termination,* $\widetilde{f}_\lambda(\boldsymbol{\beta}^{k^*}) \leq \widetilde{f}_\lambda(\boldsymbol{\beta}^0)$.

Let β_j^k *be the jth entry of* $\boldsymbol{\beta}^k$*. Then,* $\beta_j^k \notin (0, a\lambda)$ *for all* $k = 1, \ldots, k^*$.

Proof. See proof in Section EC.5.4 □

Remark 10. We would like to make a few remarks on Theorem 2 in the following.

- The assumptions of this theorem include the stipulation of $a < \frac{1}{\mathscr{M}}$, which is consistent with the requirement on a in the generalizability results in the previous section. More specifically, we may let $a < \min\{\widetilde{U}_{L,\infty}^{-1}, \mathscr{M}^{-1}\}$ to satisfy the conditions for both Theorem 2 and Proposition 1 simultaneously. This observation can be generalized to almost all our main sample complexity results. Another important assumption we have made is that $\widetilde{f}$ is smooth; that is, $\nabla\widetilde{f}$ is (globally) Lipschitz continuous. Although many machine learning problems satisfy such a condition, it is violated by a nonsmooth HDSL problem and a ReLU-NN. Nonetheless, as we show in Section 6, the nonsmooth learning problems, including the SVM, can be analyzed through a smooth approximation. As for a ReLU-NN, we demonstrate that Algorithm 1 can still be effective with the aid of a tractable initialization scheme.
- From part (b) of the result, the γ_{opt}-S^{3}ONC solution is an γ_{opt}-approximation to the S^{3}ONC as in Definition 1, if we let $\mathscr{L}_n(\cdot, \mathbf{Z}_1^n) := \widetilde{f}(\cdot)$. One may see that (27) is a γ_{opt}-approximation to the first-order KKT conditions in (13). Meanwhile, the second set of conditions in (14) are met exactly.
- It is easy to reorganize the results from parts (a) and (b) of Theorem 2 to see that the algorithm runs for $\mathscr{O}(\gamma_{opt}^{-2})$-many iterations to generate an γ_{opt}-S^{3}ONC solution. This iteration complexity is polynomial in the problem dimensionality and the numeric value of the problem data input. Because the per-iteration problems admit closed forms, we can then see that Algorithm 1 is among the class of pseudo-polynomial-time algorithms. It is worth noting that many existing alternatives are more generic and can compute stronger necessary conditions than the S^{3}ONC. Nonetheless, the new algorithm can still be of independent interest. Compared with $\mathscr{O}(\gamma_{opt}^{-3})$, the best-known rate to ensure an γ_{opt}-approximation to the second-order necessary conditions in the literature, our proposed gradient-based method yields a significantly better computational complexity.
- Part (c) indicates that the output of the algorithm is no worse than the initial solution in terms of minimizing the objective function $\widetilde{f}_\lambda$. This property ensures conditions like $\mathscr{L}_{n,\lambda}(\widehat{\boldsymbol{\beta}}, \mathbf{Z}_1^n) \leq \mathscr{L}_{n,\lambda}(\widehat{\boldsymbol{\beta}}^{\ell_1}, \mathbf{Z}_1^n)$ in the sample complexity results in, for example, part (ii) of Theorem 1, if Algorithm 1 is initialized with $\widehat{\boldsymbol{\beta}}^{\ell_1}$.
- Part (d) is useful for our subsequent analysis. One may verify that the proof of this part holds even if $\widetilde{f}(\cdot)$ is not continuously differentiable.

We observe that both the per-iteration problems (24) and (25) admit closed-form solutions. To see this, we note that (24) is essentially a soft thresholding problem, whose closed form is well known. As for (25), we observe that it can be decomposed into p-

many one-dimensional problems. Enumerating all the KKT solutions to each of these decomposed problems and noticing that $a < \mathscr{M}^{-1}$, one may verify that, for all $j = 1, \dots, p$,

$$\beta_j^{k+1} = \begin{cases} \beta_j^{k+\frac{1}{2}} - \frac{1}{\mathscr{M}} \cdot \left[\frac{\partial \widetilde{f}(\boldsymbol{\beta})}{\partial \beta_j}\right]_{\boldsymbol{\beta}=\boldsymbol{\beta}^{k+\frac{1}{2}}} & \text{if } \left| \beta_j^{k+\frac{1}{2}} - \frac{1}{\mathscr{M}} \cdot \left[\frac{\partial \widetilde{f}(\boldsymbol{\beta})}{\partial \beta_j}\right]_{\boldsymbol{\beta}=\boldsymbol{\beta}^{k+\frac{1}{2}}} \right| \ge a\lambda; \\ 0 & \text{otherwise.} \end{cases}$$

6. Theoretical Applications

In this section, we discuss two important theoretical applications of Proposition 1 and Theorem 1. Section 6.1 presents our results for a flexible class of nonsmooth HDSL problems. Section 6.2 then considers the generalizability of an FCP-regularized (deep) NN.

6.1. Nonsmooth HDSL Under A-Sparsity

The nonsmooth HDSL problem of our consideration is formulated as follows:

$$\min_{\boldsymbol{\beta}} \frac{1}{n} \sum_{i=1}^{n} \left[L_{ns}(\boldsymbol{\beta}, Z_i) := f_1(\boldsymbol{\beta}, Z_i) + \max_{\mathbf{u} \in \mathbb{U}} \{ \mathbf{u}^\top \mathbf{A}(Z_i) \boldsymbol{\beta} - \phi(\mathbf{u}, Z_i) \} \right], \tag{28}$$

where $\mathbf{A}(\cdot) : \mathscr{W} \to \Re^{m \times p}$ is deterministic and measurable (and may be nonlinear in "$\cdot$"), $\mathbb{U} \subseteq \Re^m$ is a convex and compact set with a diameter $D := \max\{\|\mathbf{u}_1 - \mathbf{u}_2\| : \mathbf{u}_1, \mathbf{u}_2 \in \mathbb{U}\}$, and $f_1 : \Re^p \times \mathscr{W} \to \Re$ and $\phi : \mathbb{U} \times \mathscr{W} \to \Re$ are deterministic, measurable functions. Let $f_1(\cdot, z)$ be continuously differentiable with

$$\left| \left[\frac{\partial f_1(\boldsymbol{\beta}, z)}{\partial \beta_j}\right]_{\boldsymbol{\beta} = \widetilde{\boldsymbol{\beta}} + \delta \cdot e_j} - \left[\frac{\partial f_1(\boldsymbol{\beta}, z)}{\partial \beta_j}\right]_{\boldsymbol{\beta} = \widetilde{\boldsymbol{\beta}}} \right| \le U_{f_1} \cdot |\delta|$$

for almost every $z \in \mathscr{W}$ and for all $\widetilde{\boldsymbol{\beta}} \in \Re^p$, $\delta \in \Re$, and $j = 1, \dots, p$. Let $\phi(\cdot, z)$ be convex and continuous for almost every $z \in \mathscr{W}$. As some standard and noncritical regularity conditions, it is assumed that $\mathbb{E}\left[n^{-1} \sum_{i=1}^{n} L_{ns}(\boldsymbol{\beta}, Z_i)\right]$ is well defined for all $\boldsymbol{\beta} \in \Re^p$ with $\inf_{\boldsymbol{\beta}} \mathbb{E}\left[n^{-1} \sum_{i=1}^{n} L_{ns}(\boldsymbol{\beta}, Z_i)\right] > -\infty$, and there exists some vector $\boldsymbol{\beta}^*_{\varepsilon'_A} \in \Re^p : \|\boldsymbol{\beta}^*_{\varepsilon'_A}\|_\infty \le R$, such that $\mathbb{E}[n^{-1} \sum_{i=1}^{n} L_{ns}(\boldsymbol{\beta}_{\varepsilon'_A}, Z_i)] - \inf_{\boldsymbol{\beta}} \mathbb{E}\left[n^{-1} \sum_{i=1}^{n} L_{ns}(\boldsymbol{\beta}, Z_i)\right] \le \varepsilon'_A$ for some $\varepsilon'_A \ge 0$. In the foregoing settings, A-sparsity (in the sense of Assumption 1) holds with $\varepsilon_A := \varepsilon'_A$, and we are again interested in estimating the vector of true parameters $\boldsymbol{\beta}^* \in \arg\inf_{\boldsymbol{\beta}} \mathbb{E}\left[n^{-1} \sum_{i=1}^{n} L_{ns}(\boldsymbol{\beta}, Z_i)\right]$. Such a problem is general enough to cover some important nonsmooth learning problems, such as the least quantile linear regression, the least absolute deviation regression, and the SVM.

Compared with our results in Section 4, a nuance here is that Problem (28) has an empirical risk function that is not everywhere differentiable because of the presence of a maximum operator. The nondifferentiable point may reside anywhere, such as at, or in some near neighborhood of, the vector of true parameters. In view of this subtlety, we propose the following FCP-based formulation:

$$\min_{\boldsymbol{\beta}} \left[\widetilde{\mathscr{L}}_{n,\delta,\lambda}(\boldsymbol{\beta}, \mathbf{Z}_1^n) := \frac{1}{n} \sum_{i=1}^{n} f_1(\boldsymbol{\beta}, Z_i) + \sum_{i=1}^{n} \frac{1}{n} \max_{\mathbf{u} \in \mathbb{U}} \left\{ \mathbf{u}^\top \mathbf{A}(Z_i) \boldsymbol{\beta} - \phi(\mathbf{u}, Z_i) - \frac{\|\mathbf{u} - \mathbf{u}_0\|^2}{2n^\delta} \right\} + \sum_{j=1}^{p} P_\lambda(|\beta_j|) \right], \tag{29}$$

for a user-specific $\mathbf{u}^0 \in \mathbb{U}$ and $\delta > 0$ (which is chosen to be $\delta = \frac{1}{4}$ later in our theory).

The proposed formulation in (29) is not an immediate instantiation of (3) for the population-level problem $\inf_{\boldsymbol{\beta}} \mathbb{E}\left[n^{-1} \sum_{i=1}^{n} L_{ns}(\boldsymbol{\beta}, Z_i)\right]$. Indeed, apart from the FCP-based regularization term, an additional quadratic function $-\frac{\|\mathbf{u} - \mathbf{u}_0\|^2}{2n^\delta}$ is also included in (29). The purpose of this extra term is to add regularities to facilitate our analysis; although $\widetilde{\mathscr{L}}_n(\boldsymbol{\beta}, \mathbf{Z}_1^n) := \frac{1}{n} \sum_{i=1}^{n} L_{ns}(\boldsymbol{\beta}, Z_i)$ is not everywhere differentiable,

$$\widetilde{\mathscr{L}}_{n,\delta}(\boldsymbol{\beta}, \mathbf{Z}_1^n) := \frac{1}{n} \sum_{i=1}^{n} f_1(\boldsymbol{\beta}, Z_i) + \sum_{i=1}^{n} \frac{1}{n} \max_{\mathbf{u} \in \mathbb{U}} \left\{ \mathbf{u}^\top \mathbf{A}(Z_i) \boldsymbol{\beta} - \phi(\mathbf{u}, Z_i) - \frac{\|\mathbf{u} - \mathbf{u}_0\|^2}{2n^\delta} \right\}, \tag{30}$$

is verifiably a continuously differentiable approximation to $\widetilde{\mathscr{L}}_n(\boldsymbol{\beta}, \mathbf{Z}_1^n)$. The error incurred by this approximation can be controlled by properly determining the hyper-parameter δ. Furthermore, invoking theorem 1 by Nesterov (2005) (restated as Theorem EC.2 in the e-companion for completeness), one may derive the Lipschitz constant of the gradient of $\widetilde{\mathscr{L}}_{n,\delta}(\cdot, \mathbf{Z}_1^n)$. This observation is formalized in part (a) of Theorem 3.

With this approximation, the nonsmooth HDSL problem can now be analyzed via the framework of HDSL under A-sparsity; we can consider the approximation error as a component of ε_A in the definition of A-sparsity. Via this perspective, we may easily apply results from Proposition 1 or Theorem 1 to (30) after some conversions of the settings. In doing so, we impose the following two assumptions, which are instantiations of Assumptions 3 and 4, respectively.

Assumption 5. *For all $\boldsymbol{\beta} \in \Re^p : \|\boldsymbol{\beta}\|_\infty \le R$ and $i = 1, \dots, n$, it holds that $\|L_{ns}(\boldsymbol{\beta}, Z_i) - \mathbb{E}[L_{ns}(\boldsymbol{\beta}, Z_i)]\|_{\psi_1} \le \sigma$, for some $\sigma \ge 1$.*

Assumption 6. *For some measurable and deterministic function $\mathscr{C}:\mathscr{W}\to\Re_+$, the random variable $\mathscr{C}(Z_i)$ satisfies that*

(i) *for all $i=1,\ldots,n$ and some $\sigma_L\ge 1$, $\|\mathscr{C}(Z_i)-\mathbb{E}[\mathscr{C}(Z_i)]\|_{\psi_1}\le\sigma_L$, and*

(ii) *For all $i=1,\ldots,n$ for some $\mathscr{C}_\mu\ge 1$, $\mathbb{E}[\mathscr{C}(Z_i)]\le\mathscr{C}_\mu$.*

Furthermore, $|L_{ns}(\boldsymbol{\beta}_1,z)-L_{ns}(\boldsymbol{\beta}_2,z)|\le\mathscr{C}(z)\|\boldsymbol{\beta}_1-\boldsymbol{\beta}_2\|$, for all $\boldsymbol{\beta}_1,\boldsymbol{\beta}_2\in\Re^p\cap\{\boldsymbol{\beta}:\|\boldsymbol{\beta}\|_\infty\le R\}$ and almost every $z\in\mathscr{W}$.

Remark 11. Similar to Assumptions 3 and 4, the foregoing two conditions ensure that the underlying distribution is subexponential and that a Lipschitz-like inequality holds for $L_{ns}(\cdot,z)$.

We are now ready to present our results on nonsmooth HDSL in the following theorem, which leads to what is claimed in Equation (8). Similar to Section 4, we adopt the shorthand $\widetilde{\zeta}:=\ln(3eR\cdot(\sigma_L+\mathscr{C}_\mu))$.

Theorem 3. *Suppose that $\|\mathbf{A}(z)\|_{1,2}^2\le U_A$ for some $U_A\ge 0$ and for almost every $z\in\mathscr{W}$. Let Assumptions 1, 5, and 6 hold (where ε_A and $\mathbb{L}(\cdot)$ from Assumption 1 become ε'_A and $\mathbb{E}[L_{ns}(\cdot,\mathcal{Z})]$, respectively). The following statements hold:*

(a) *For any $\delta>0$, all $j=1,\ldots,p$, every $\widetilde{\boldsymbol{\beta}}\in\Re^p$, and almost every $\mathbf{Z}_1^n\in\mathscr{W}^n$, the partial derivative $\frac{\partial\widetilde{\mathscr{L}}_{n,\delta}(\widetilde{\boldsymbol{\beta}},\mathbf{Z}_1^n)}{\partial\beta_j}$ is well defined and Lipschitz continuous with*

$$\left|\left[\frac{\partial\widetilde{\mathscr{L}}_{n,\delta}(\boldsymbol{\beta},\mathbf{Z}_1^n)}{\partial\beta_j}\right]_{\boldsymbol{\beta}=\widetilde{\boldsymbol{\beta}}+h\cdot e_j}-\left[\frac{\partial\widetilde{\mathscr{L}}_{n,\delta}(\boldsymbol{\beta},\mathbf{Z}_1^n)}{\partial\beta_j}\right]_{\boldsymbol{\beta}=\widetilde{\boldsymbol{\beta}}}\right|\le(U_{f_1}+n^\delta U_A)\cdot|h|$$

for any $h\in\Re$.

(b) *Let $\delta=\frac{1}{4}$, $a=\frac{1}{2(U_{f_1}+n^{1/4}U_A)}$, and $\lambda:=\sqrt{\frac{8\sigma}{c\cdot a\cdot n^{3/8}}[\ln(n^{\frac{3}{8}}p)+\widetilde{\zeta}]}$ for the same c in (12). For almost every $\mathbf{Z}_1^n\in\mathscr{W}^n$, assume that the minimization problem $\min_{\boldsymbol{\beta}}\widetilde{\mathscr{L}}_{n,\delta}(\boldsymbol{\beta},\mathbf{Z}_1^n)+\lambda|\boldsymbol{\beta}|$ admits a finite optimal solution denoted by $\widehat{\boldsymbol{\beta}}^{\ell_1,\delta}:=\widehat{\boldsymbol{\beta}}^{\ell_1,\delta}(\mathbf{Z}_1^n)$. Consider any random vector $\widehat{\boldsymbol{\beta}}\in\Re^p$ such that $\|\widehat{\boldsymbol{\beta}}\|_\infty\le R$, $\widetilde{\mathscr{L}}_{n,\delta,\lambda}(\widehat{\boldsymbol{\beta}},\mathbf{Z}_1^n)\le\widetilde{\mathscr{L}}_{n,\delta,\lambda}(\widehat{\boldsymbol{\beta}}^{\ell_1,\delta},\mathbf{Z}_1^n)$ almost surely, and $\widehat{\boldsymbol{\beta}}$ satisfies the $S^3ONC(\mathbf{Z}_1^n)$ to (29) with probability one* (w.p.1.). *For some universal constant $C_5>0$, if*

$$n>C_5\cdot\frac{D^4}{\sigma^2}+C_5\cdot\frac{(\varepsilon'_A)^4}{\sigma^4}+C_5\cdot(U_{f_1}+U_A)^{4/3}\cdot[\ln(np)+\widetilde{\zeta}]^{4/3}\cdot s^{8/3}\max\left\{1,\|\boldsymbol{\beta}^*_{\varepsilon_A}\|_\infty^{8/3}\right\},\tag{31}$$

where $D:=\max\{\|\mathbf{u}_1-\mathbf{u}_2\|:\mathbf{u}_1,\mathbf{u}_2\in\mathbb{U}\}$, then

$$\begin{aligned}&\mathbb{L}(\widehat{\boldsymbol{\beta}})-\inf_{\boldsymbol{\beta}}\mathbb{L}(\boldsymbol{\beta})\\&\le\frac{C_5\cdot\sigma\cdot s\cdot(\ln(np)+\widetilde{\zeta})}{n^{3/4}}+C_5\cdot\sqrt{\frac{\sigma\varepsilon'_A}{n^{1/4}}}+\varepsilon'_A\\&+\frac{C_5\cdot s\cdot\sigma\cdot\max\left\{1,\|\boldsymbol{\beta}^*_{\varepsilon_A}\|_\infty\right\}\cdot(U_{f_1}+U_A)^{1/2}\sqrt{\ln(np)+\widetilde{\zeta}}+\max\{\sqrt{\sigma}\cdot D,D^2\}}{n^{1/4}}\end{aligned}\tag{32}$$

with probability at least $1-2(p+1)\exp(-n/C_5)-6\exp(-2cn^{1/2})$.

Proof. See Section EC.5.2 in the e-companion. □

Remark 12. It is possible to generalize part (b) of the previous theorem to obtain an error bound in the parameterization of any $\delta>0$. Nonetheless, the optimal choice to balance all the error terms would be $\delta=1/4$.

Remark 13. Theorem 3 is general enough to cover a flexible class of nonsmooth HDSL problems under A-sparsity. Particularly, in the case of the high-dimensional SVM, Problem (28) becomes

$$\begin{aligned}&\min_{\boldsymbol{\beta}}\rho\|\boldsymbol{\beta}\|^2+\frac{1}{n}\sum_{i=1}^n[1-y_i\mathbf{x}_i^\top\boldsymbol{\beta}]_+\\&\Longleftrightarrow\min_{\boldsymbol{\beta}}\rho\|\boldsymbol{\beta}\|^2+\frac{1}{n}\sum_{i=1}^n\max_{u_i:0\le u_i\le 1}\{u_i\cdot(1-y_i\mathbf{x}_i^\top\boldsymbol{\beta})\},\end{aligned}\tag{33}$$

where $(\mathbf{x}_i,y_i)$, for $i=1,\ldots,n$, are i.i.d. random pairs of the feature values and the categorial labels with support $\{\mathbf{x}\in\Re^p:|\mathbf{x}|\le 1\}\times\{-1,+1\}$, and $\rho\ge 0$ is a user-specific constant. (The assumption that $|\mathbf{x}_i|\le 1$, almost surely (a.s.), can always be ensured by normalization.) We may enable the SVM to handle high dimensionality via the following formulation:

$$\min_{\boldsymbol{\beta}}\rho\|\boldsymbol{\beta}\|^2+\frac{1}{n}\sum_{i=1}^n\max_{u_i:0\le u_i\le 1}\left\{u_i\cdot(1-y_i\mathbf{x}_i^\top\boldsymbol{\beta})-\frac{(u_i-u_0)^2}{2n^\delta}\right\}+\sum_{j=1}^p P_\lambda(|\beta_j|),\tag{34}$$

where the value of $u_0\in[0,1]$ can be specified arbitrarily. As a special case to (29), Problem (34) satisfies both Assumptions 5 and 6. For example, when $\rho=0.01$, both of the assumptions are met with $\sigma\le O(1)$, $R\le O(1)$, $\sigma_L=0$, and $\mathscr{C}_\mu\le O(1)\cdot\sqrt{p}$. (More detailed derivations are provided in Section EC.4 of the e-companion.) Also observe that we may let f_1, U_{f_1}, D, and $\mathbf{A}(\mathbf{Z}_i^n)$ from Theorem 3 to be

$$\begin{aligned}&f_1(\boldsymbol{\beta},\mathbf{Z}_1^n):=\rho\|\boldsymbol{\beta}\|^2,\quad U_{f_1}:=2\rho,\\&D:=\max\{(u_1-u_2)^2:u_1,u_2\in[0,1]\},\quad\text{and}\\&\mathbf{A}(\mathbf{Z}_i^n):=y_i\cdot\mathbf{x}_i^\top,\end{aligned}$$

respectively, in the SVM. Thus, $U_{f_1}\le O(1)$, $D=1$, and $U_A\le\max_{y,\mathbf{x}}\left\{\|y\cdot\mathbf{x}^\top\|_{1,2}^2:y\in\{-1,1\},\ |\mathbf{x}|\le 1\right\}\le 1$ in this special case. Recall here that the error bound in, for example, (32) is poly-logarithmic in $\mathscr{C}_\mu$. Theorem 3 then implies that the poly-logarithmic sample complexity can also be achieved for the FCP-regularized SVM.

In contrast to (34), an alternative formulation as follows has been previously discussed in the literature:

$$\min_{\boldsymbol{\beta}}\rho\|\boldsymbol{\beta}\|^2+\frac{1}{n}\sum_{i=1}^n[1-y_i\mathbf{x}_i^\top\boldsymbol{\beta}]_++\sum_{j=1}^p\widetilde{P}_\lambda(|\beta_j|),\tag{35}$$

where $\widetilde{P}_\lambda(|\cdot|):\Re\to\Re$ is some sparsity-inducing regularization function, such as SCAD and Lasso.

Compared with (34), this alternative does not incorporate the smoothing term of $-\frac{(u_i-u_0)^2}{2n^\delta}$. Such a formulation has been shown to be successful in multiple realistic classification problems (Zhang et al. 2006). Furthermore, recovery theories in different high-dimensional settings have been established by Zhang et al. (2016b, c) and Peng et al. (2016). Nonetheless, the existing results commonly stipulate a strictly positive lower bound on the eigenvalues of some principal submatrices of $\mathbf{X}^\top\mathbf{X}$ or $\mathbb{E}[\mathbf{X}^\top\mathbf{X}]$, where $\mathbf{X} := (\mathbf{x}_i^\top : i = 1,\ldots,n)$. Some of these conditions are the instantiations of the RE condition in the SVM problem. In contrast, our bound on the excess risk is established without these eigenvalue conditions.

6.2. Regularized Deep NNs

This section presents a generalization error bound for a flexible set of NN architectures. Additional results are provided in Section EC.1 of the e-companion, where we derive more explicit error bounds under additional regularities.

Although NNs can be applied to a wide spectrum of data-driven tasks, our analysis herein is focused on a binary classification problem in the following settings. For some $\mathcal{X} := \{\mathbf{x} \in \Re^d : \|\mathbf{x}\| = 1\}$ and $\boldsymbol{\mathcal{Y}} \in \{-1, 1\}$ (where $d > 0$ is some integer), let $(\mathbf{x}, y) \in \mathcal{X} \times \boldsymbol{\mathcal{Y}}$ be a random pair that follows an unknown probability distribution $\mathbb{D}$ on $\mathcal{X} \times \boldsymbol{\mathcal{Y}}$ with support $supp(\mathbb{D})$. Here, $\mathbf{x}$ is the vector of random feature values and y is the corresponding class label. We assume that there exists an unknown, deterministic, and measurable separating function $g : \mathcal{X} \to \Re$ such that $\inf_{(\mathbf{x},y)\in supp(\mathbb{D})} \{y \cdot g(\mathbf{x})\} \geq v$ for some $v \in (0,1)$; that is, the two categories of data are separable by function g. Also assume that $\mathbb{E}\big[|g(\mathbf{x})|\big] < \infty$. The learning problem of interest here, as a special case of (1), is to train a classifier using the knowledge of a sequence of i.i.d. random samples, $(\mathbf{x}_i, y_i)$, $i = 1,\ldots,n$, of $(\mathbf{x}, y)$.

In applying an NN to solving this learning problem, we narrow down the search of the optimal classifier to the determination of the best fitting parameters for the NN. Some relative details are below. Denote by $\Psi : \Re \to \Re$ an activation function, such as the ReLU, $\Psi_{ReLU}(x) = \max\{0, x\}$, the softplus, $\Psi_{softplus}(x) = \ln(1 + e^x)$, and the sigmoid, $\Psi_{sigmoid}(x) = \frac{e^x}{1+e^x}$. The NN model is then a network that consists of multiple layers (groups) of neurons (or units). Each neuron is a computing unit that performs the operations of the chosen activation function on the input signals. Architectures among those layers are formed in the sense that the signals are passed from the layer of input neurons to the layer of output units, transversing a predetermined collection of candidate paths. Each path may comprise multiple neurons and connections. Fitting parameters often exist in the forms of connection weights and biases to (dis)amplify and offset the signals, respectively. A layer that is neither the input layer nor the output layer is called a hidden layer. Throughout our discussions on the NNs, we let $\mathscr{D} \geq 2$ be the number of layers (excluding the input layer but including the output layer). A neuron in a hidden layer is called a hidden neuron.

We denote this NN by $F_{NN}(\mathbf{x}, \boldsymbol{\beta})$, where $F_{NN} : \mathcal{X} \times \Re^p \to \Re$ is a deterministic, measurable function that captures the output of an NN given input $\mathbf{x}$ and fitting parameters $\boldsymbol{\beta}$. We also assume that there exists a deterministic function $\Omega : \{1,\ldots,p\} \to \Re_+$ such that

$$\Omega(p') \geq \inf_{\boldsymbol{\beta}:\|\boldsymbol{\beta}\|_0 \leq p'} \mathbb{E}\big[|F_{NN}(\mathbf{x}, \boldsymbol{\beta}) - g(\mathbf{x})|\big], \quad \forall p' : 1 \leq p' \leq p. \tag{36}$$

Intuitively, $\Omega(p')$ measures the model misspecification error incurred by the NN in representing g, when only p'-many fitting parameters are nonzero (active).

In training the NN, we focus on the following formulation as a special case to (3):

$$\inf_{\boldsymbol{\beta}} \mathscr{T}_{n,\lambda}(\boldsymbol{\beta}) := n^{-1}\sum_{i=1}^{n} \mathscr{F}(y_i \cdot F_{NN}(\mathbf{x}_i, \boldsymbol{\beta})) + \sum_{j=1}^{p} P_\lambda(|\beta_j|), \tag{37}$$

where we follow Cao and Gu (2020, 2019) in defining $\mathscr{F} : \Re \to \Re_+$ to be $\mathscr{F}(z) := \ln(1 + \exp(-z))$. If we drop the regularization term $\sum_{j=1}^{p} P_\lambda(|\beta_j|)$, then (37) is reduced to the conventional training formulation for an NN. Hereafter, we assume that $\mathbb{E}[|F_{NN}(\mathbf{x}, \boldsymbol{\beta})\,|] < \infty$ for all $\boldsymbol{\beta} : \|\boldsymbol{\beta}\|_\infty \leq R_\Omega$ for some $R_\Omega > 0$. This quantity should be properly large to ensure the satisfaction of the following assumption.

Assumption 7. *For all $1 \leq s_A \leq p$, it holds that $\emptyset \neq [-R_\Omega, R_\Omega]^p \cap \{\boldsymbol{\beta} \in \Re^p : \mathbb{E}[\,|g(\mathbf{x}) - F_{NN}(\mathbf{x}, \boldsymbol{\beta})|\,] \leq \Omega(s_A), \|\boldsymbol{\beta}\|_0 \leq s_A\}$.*

Intuitively, Assumption 7 means that the NN can represent the separating function g with a model misspecification error of no more than $\Omega(s_A)$ when (a) no more than s_A-many fitting parameters are nonzero and (b) the absolute values of these fitting parameters are bounded from above by $R_\Omega > 0$.

We also impose the following noncritical condition on the architecture of an NN.

Assumption 8. *For any constants $C \in \Re$, $R_\Omega > 0$, $p' \geq 1$, and fitting parameters $\boldsymbol{\beta}_1 \in \Re^p : \|\boldsymbol{\beta}_1\|_\infty \leq R_\Omega$, $\|\boldsymbol{\beta}_1\|_0 \leq p'$, it holds that $F_{NN}(\mathbf{x}, \boldsymbol{\beta}_1) \cdot C = F_{NN}(\mathbf{x}, \boldsymbol{\beta}_2)$ for some $\boldsymbol{\beta}_2 \in \Re^p : \|\boldsymbol{\beta}_2\|_\infty \leq C \cdot R_\Omega$, $\|\boldsymbol{\beta}_2\|_0 \leq p'$, for every $\mathbf{x} \in \mathcal{X}$.*

It can be verified that Assumption 8 holds for many NN architectures, including many convolutional NNs and residual networks that have linear or ReLU activation functions in the output layer.

Remark 14. By the satisfaction of Assumptions 7 and 8, we argue that the generalizability of an NN trained by solving (37) can be analyzed through the framework of HDSL under A-sparsity. Based on the existing results on the representability of NNs (DeVore et al. 1989, Mhaskar 1996, Mhaskar and Poggio 2016, Yarotsky 2017), an NN with a reasonably small network size s_A may well represent g (such that $\Omega(s_A)$ is small) under some plausible conditions. These representability results imply the innate presence of A-sparsity in an NN model. Observe that $\mathscr{F}$ is 1-Lipschitz continuous. Thus,

$$\mathbb{E}\left[\mathscr{F}\left(y\cdot\frac{\ln n}{2v}\cdot F_{NN}(\mathbf{x},\boldsymbol{\beta}_1)\right)\right]-\mathbb{E}\left[\mathscr{F}\left(y\cdot\frac{\ln n}{2v}\cdot g(\mathbf{x})\right)\right]$$
$$\leq\frac{\ln n}{2v}\cdot\mathbb{E}\left[\,|F_{NN}(\mathbf{x},\boldsymbol{\beta}_1)-g(\mathbf{x})|\,\right]$$

for any $\boldsymbol{\beta}:\|\boldsymbol{\beta}\|_\infty\leq R_\Omega$. Invoking Assumption 7 and the fact that $\inf_u\mathscr{F}(u)=0$, we obtain that

$$\min_{\substack{\boldsymbol{\beta}:\|\boldsymbol{\beta}\|_0\leq s_A,\\ \|\boldsymbol{\beta}\|_\infty\leq R_\Omega}}\mathbb{E}\left[\mathscr{F}\left(y\cdot\frac{\ln n}{2v}\cdot F_{NN}(\mathbf{x},\boldsymbol{\beta})\right)\right]-\inf_u\mathscr{F}(u)$$
$$\leq\min_{\substack{\boldsymbol{\beta}:\|\boldsymbol{\beta}\|_0\leq s_A,\\ \|\boldsymbol{\beta}\|_\infty\leq R_\Omega}}\mathbb{E}\left[\frac{\ln n}{2v}\,|F_{NN}(\mathbf{x},\boldsymbol{\beta})-g(\mathbf{x})|\right]+\mathbb{E}\left[\mathscr{F}\left(y\cdot\frac{\ln n}{2v}\cdot g(\mathbf{x})\right)\right]$$
$$\leq\frac{\ln n}{2v}\cdot\Omega(s_A)+\mathbb{E}\left[\mathscr{F}\left(y\cdot\frac{\ln n}{2v}\cdot g(\mathbf{x})\right)\right]\leq\frac{\ln n}{2v}\cdot\Omega(s_A)+\frac{1}{\sqrt{n}},$$

where the last inequality is due to the assumption that, for all $(\mathbf{x},y)\in supp(\mathbb{D})$, it holds that $y\cdot g(\mathbf{x})\geq v\Rightarrow\mathbb{E}\left[\mathscr{F}\left(y\cdot\frac{\ln n}{2v}\cdot g(\mathbf{x})\right)\right]\leq\ln(1+\exp(-0.5\ln n))\leq\frac{1}{\sqrt{n}}$. By Assumption 8, $\frac{\ln n}{2v}\cdot F_{NN}(\mathbf{x},\boldsymbol{\beta})$ can be represented by the same NN architecture; that is, $\frac{\ln n}{2v}\cdot F_{NN}(\mathbf{x},\boldsymbol{\beta})=F_{NN}(\mathbf{x},\boldsymbol{\beta}')$ for some new fitting parameters $\boldsymbol{\beta}':\|\boldsymbol{\beta}'\|_\infty\leq\frac{\ln n}{2v}R_\Omega$. Thus, we may have

$$\min_{\substack{\boldsymbol{\beta}:\|\boldsymbol{\beta}\|_0\leq s_A,\\ \|\boldsymbol{\beta}\|_\infty\leq\frac{\ln n}{2v}\cdot R_\Omega}}\mathbb{E}\left[\mathscr{F}(y\cdot F_{NN}(\mathbf{x},\boldsymbol{\beta}))\right]-\inf_u\mathscr{F}(u)\leq\frac{\ln n}{2v}\cdot\Omega(s_A)+\frac{1}{\sqrt{n}},\tag{38}$$

which matches the statement of Assumption 2 with $s:=s_A$, $R:=\frac{\ln n}{2v}\cdot R_\Omega$, $\varepsilon_A:=\frac{\ln n}{2v}\cdot\Omega(s_A)+\frac{1}{\sqrt{n}}$, and $L_g^*:=\inf_u\mathscr{F}(u)=0$. As mentioned, explicit forms of $\Omega(\cdot)$ have been provided by DeVore et al. (1989), Yarotsky (2017), Mhaskar and Poggio (2016), and Mhaskar (1996). With the previous discussion, the generalizability of an NN can then be derived using the same machinery for HDSL under A-sparsity, under one more flexible assumption on the NN's architecture as follows.

Assumption 9. *For almost every $\mathbf{x}\in\mathscr{X}$, it holds that the gradient $\nabla_{\boldsymbol{\beta}}F_{NN}(\mathbf{x},\boldsymbol{\beta})$ and Hessian $\nabla^2_{\boldsymbol{\beta}}F_{NN}(\mathbf{x},\boldsymbol{\beta})$ of $F_{NN}(\mathbf{x},\cdot)$ are everywhere well defined and satisfy that*

$$\max\left\{\operatorname*{ess\,sup}_{\mathbf{x}\in\mathscr{X}}\|\nabla_{\boldsymbol{\beta}}F_{NN}(\mathbf{x},\boldsymbol{\beta})\|,\quad\operatorname*{ess\,sup}_{\mathbf{x}\in\mathscr{X}}\|\nabla^2_{\boldsymbol{\beta}}F_{NN}(\mathbf{x},\boldsymbol{\beta})\|\right\}$$
$$\leq\exp\left[\mathscr{U}_{NN}\cdot\mathscr{D}\cdot\ln(\mathscr{U}_{NN}\cdot\|\boldsymbol{\beta}\|+\mathscr{U}_{NN})\right]$$

for all $\boldsymbol{\beta}\in\Re^p$ and some $\mathscr{U}_{NN}\geq 1$.

Assumption 9 essentially allows the norms of gradient and Hession to grow exponentially in the number of layers $\mathscr{D}$. Such an assumption is satisfied by a wide spectrum of NN architectures, especially when the activation functions are smooth. Some NNs with nonsmooth activation functions, such as the ReLU, may still be analyzed. We discuss such a case later in Section EC.1.2 of the e-companion.

We are now ready to present our result on the generalizability of a regularized NN. With some abuse of notations, the $\text{S}^3\text{ONC}(\mathbf{Z}_1^n)$, in this special case, is referred to as the $\text{S}^3\text{ONC}(\mathbf{X},\mathbf{y})$ to Problem (37), where $\mathbf{X}:=(\mathbf{x}_i^\top)$ and $\mathbf{y}:=(y_i)$.

Theorem 4. *Consider any random vector $\widehat{\boldsymbol{\beta}}$ such that $\|\widehat{\boldsymbol{\beta}}\|_\infty\leq\frac{\ln n}{2v}\cdot R_\Omega$ and the $S^3ONC(\mathbf{X},\mathbf{y})$ holds at $\widehat{\boldsymbol{\beta}}$ almost surely. Suppose that Assumptions 7–9 hold. For any fixed $\Gamma\geq 0$, assume that $\mathscr{T}_{n,\lambda}(\widehat{\boldsymbol{\beta}})-\inf_{\boldsymbol{\beta}}\mathscr{T}_{n,\lambda}(\boldsymbol{\beta})\leq\Gamma$, w.p.1., where $\mathscr{T}_{n,\lambda}$ is as defined in (37). There exists a universal constant $C_6>0$, such that, for any $s_A:1\leq s_A\leq p$, if $a<\frac{1}{2}\cdot\exp\{-2\mathscr{U}_{NN}\cdot\mathscr{D}\cdot\ln[2p\cdot v^{-1}\cdot\mathscr{U}_{NN}\cdot R_\Omega\cdot\ln n]\}$,*

$$\lambda:=\sqrt{\frac{8\sigma}{c\cdot a\cdot n^{2/3}}\left[\ln\left(\frac{3e}{2v}\cdot R_\Omega pn^{4/3}\right)+\mathscr{U}_{NN}\cdot\mathscr{D}\cdot\ln\left(\mathscr{U}_{NN}R_\Omega pnv^{-1}\right)\right]}$$

and

$$n>C_6\cdot\left[\left(\Gamma+v^{-1}\cdot\Omega(s_A)\cdot\ln n\right)^3+s_A\cdot\mathscr{D}\cdot\mathscr{U}_{NN}\cdot\ln\left(\mathscr{U}_{NN}\cdot(1+npR_\Omega v^{-1})\right)\right],\tag{39}$$

then it holds that

$$\mathbb{E}\left[\mathbb{I}\left(y\cdot F_{NN}(\mathbf{x},\widehat{\boldsymbol{\beta}})<0\right)\right]\leq C_6\cdot\left(\frac{s_A\cdot\mathscr{D}\cdot\mathscr{U}_{NN}\cdot\ln\left(\mathscr{U}_{NN}\cdot(1+npR_\Omega v^{-1})\right)}{n^{2/3}}+\sqrt{\frac{s_A\cdot\mathscr{D}\cdot\mathscr{U}_{NN}\cdot\ln\left(\mathscr{U}_{NN}\cdot(1+npR_\Omega v^{-1})\right)}{n}}+\frac{1}{n^{1/3}}\right)+v^{-1}\cdot\Omega(s_A)\cdot\ln n+\Gamma+C_6\cdot\sqrt{\frac{\Gamma+v^{-1}\cdot\Omega(s_A)\cdot\ln n}{n^{1/3}}}\tag{40}$$

with probability at least

$$1-C_6p\exp\left(-\frac{n}{C_6}\right)-C_6\exp\left(-\frac{n^{1/3}}{C_6}\right).$$

Here, $\Omega(\cdot)$ is defined as in (36).

Proof. See Section EC.5.3.1 of the e-companion. □

Remark 15. We would like to make a few remarks on the results presented in this theorem.

(i) In Equation (40), $\mathbb{E}[\mathbb{I}(y\cdot F_{NN}(\mathbf{x},\widehat{\boldsymbol{\beta}})<0)]=\mathbb{P}[y\cdot F_{NN}(\mathbf{x},\widehat{\boldsymbol{\beta}})<0]$ is also referred to as the expected 0-1 loss and is a commonly adopted measure of generalization performance, such as by Cao and Gu (2020, 2019), in a binary classification problem.

(ii) This theorem provides the promised polylogarithmic dependence between the sample size n and the dimensionality p; polynomially increasing n can compensate for the exponential growth in p. With this result, the generalizability of an overparameterized NN is ensured, and the promised result in (9) is proven. The error bound can be made more explicit under some additional conditions as discussed in Section EC.1.1 of the e-companion.

(iii) Although Assumption 9 allows the Lipschitz constant to grow exponentially in the number of layers $\mathscr{D}$, the generalization error increases no more than linearly in $\mathscr{D}$.

(iv) Many sparsity-inducing regularization schemes have been discussed in the literature, including Dropout (Srivastava et al. 2014), sparsity-inducing penalization (Han et al. 2015, Wen et al. 2016, Louizos et al. 2017, Scardapane et al. 2017), DropConnect (Wan et al. 2013), randomDrop (Huang et al. 2016), and pruning (Alford et al. 2018). Many of these studies are focused on the numerical aspects, yet the theoretical guarantees on the effectiveness of regularization are still largely lacking. Although Wan et al. (2013) presented generalization error analyses for DropConnect, the dependence among the dimensionality, the generalization error, and the sample size are not explicated therein. It is our conjecture that our results could be extended to and combined with the alternative regularization schemes to facilitate the analysis of the regularized NNs.

(v) Theorem 4 informs us that the generalization performance of the NNs is consistent with the optimization quality. If all other quantities are fixed, the generalization error can be bounded by $\mathscr{O}(\sqrt{\Gamma}+\Gamma)$, where we recall that $\Gamma \geq 0$ is the suboptimality gap. Admittedly, how to control Γ is still an open question. The traditional training formulation of an NN is usually nonconvex. Thus, it is generally prohibitive to compute a global solution. The challenge is further increased by the incorporation of the FCP, which is also nonconvex. Fortunately, despite the current theoretical challenge, it has been observed empirically that some local optimization algorithms could well approximate a global optimum in NN training, for example, in the experiments reported by Wan et al. (2013) and Alford et al. (2018). To explain these observations, several theoretical paradigms have already been provided by Du et al. (2019), Liang et al. (2018), Haeffele and Vidal (2017), and Wang et al. (2019). Based on those results, it is promising that the structures of an NN (even with regularization) can often be exploited to facilitate global optimization. An excellent review of this topic is provided by Sun (2019). To add to the literature, we present an interesting special case where a suboptimality-independent generalization error bound for the FCP-regularized NN can be achieved at a pseudo-polynomial-time computable solution in Section EC.1.2 of the e-companion.

7. Numerical Experiments

We report in this section several numerical experiments. In Sections 7.1 and 7.2, we consider the high-dimensional Huber regression under A-sparsity and the NNs, respectively. Then, Section EC.2 of the e-companion presents our test results on the high-dimensional SVM (as a special nonsmooth learning problem) and some additional numerical examples on the NNs. Unless otherwise stated explicitly, most of our experiments, including those in the e-companion, were implemented in Matlab 2014b and run with a single thread on a PC with 40 Intel (R) Xeon (R) E5-2640-v4 CPU cores (2.40 GHz, 64 bits), and 128 GB memory. A different implementation environment was involved in the tests on some larger-scale NN models, as presented in Section 7.2.

7.1. Experiments on HDSL Under A-Sparsity

This section reports our test results on high-dimensional Huber regression (HR) under A-sparsity (in the sense of Assumption 1). Our settings for experiments are summarized below: Denote by $\mathscr{N}(0,\sigma^2)$ a centered normal distribution with variance $\sigma^2>0$ and by $\mathscr{N}_p(\mathbf{0},\Sigma)$ a centered p-variate normal distribution with covariance matrix $\Sigma=(\varsigma_{j_1,j_2})$ and $\varsigma_{j_1,j_2}=0.3^{|j_1-j_2|}$. The training data set $\{(\mathbf{x}_i,y_i): i=1,\ldots,n\}$ was generated as per a linear system $y_i=\mathbf{x}_i^\top\boldsymbol{\beta}^*+\omega_i$, for $i=1,\ldots,n$. Here, $(\mathbf{x}_i, y_i)$ denotes a pair of (observed) design and response, and $\boldsymbol{\beta}^*$ denotes the vector of true parameters to be recovered. Some additional details are summarized here:

- The training sample size was chosen as $n=100$.
- Let ω_i, for all $i=1,\ldots,n$, be i.i.d. white noises such that $\omega_i\sim\mathscr{N}(0,\sigma^2)$.
- Let $\mathbf{x}_i\sim\mathscr{N}_p(\mathbf{0},\Sigma)$, for $i=1,\ldots,n$, be i.i.d. random vectors.
- The vector of true parameters was prescribed as

$$\boldsymbol{\beta}^* = \boldsymbol{\beta}^*_{\varepsilon_A} + E\cdot\boldsymbol{v}\cdot\frac{1}{|\boldsymbol{v}|},$$

where

$$\boldsymbol{\beta}^*_{\varepsilon_A} := \left(3,5,0,0,1.5,\underbrace{0,\ldots,0}_{(p-5)\text{-many}0'\text{s}}\right)^\top$$

and $E\cdot\boldsymbol{v}\cdot\frac{1}{|\boldsymbol{v}|}$ stands for some dense perturbation. Here, $E>0$ denotes a user-specific scalar and $\boldsymbol{v}=(v_j)$ denotes a random vector with i.i.d. entries of uniform random variables on $[-1,1]$. The magnitude of the perturbation can be calculated as $|E\cdot\boldsymbol{v}\cdot\frac{1}{|\boldsymbol{v}|}|=E$.

Given these, this experiment was focused on the following HR problem:

$$\min_{\boldsymbol{\beta}} \frac{1}{n}\sum_{i=1}^{n}\Bigg[L_{HR}(\boldsymbol{\beta},\mathbf{x}_i,y_i) := \frac{1}{2}(\mathbf{x}_i\boldsymbol{\beta}-y_i)^2 \cdot \mathbb{I}(|\mathbf{x}_i\boldsymbol{\beta}-y_i| \le \eta). \\ +\left(\eta\,|\mathbf{x}_i\boldsymbol{\beta}-y_i| - \frac{\eta^2}{2}\right)\cdot \mathbb{I}(|\mathbf{x}_i\boldsymbol{\beta}-y_i| > \eta)\Bigg].$$

The corresponding FCP-regularized formulation, referred to as the HR-FCP, is then given as

$$\min_{\boldsymbol{\beta}} n^{-1}\sum_{i=1}^{n} L_{HR}(\boldsymbol{\beta},\mathbf{x}_i,y_i) + \sum_{j=1}^{p} P_\lambda(|\beta_j|). \tag{41}$$

This problem was solved via Algorithm 1, for which the initial solution was prescribed as $\widehat{\boldsymbol{\beta}}^{\ell_1} \in \arg\min_{\boldsymbol{\beta}} n^{-1} \sum_{i=1}^{n} L_{HR}(\boldsymbol{\beta},\mathbf{x}_i,y_i) + \lambda\cdot\sum_{j=1}^{p}|\beta_j|$ for the same λ as in (41).

The hyper-parameters of Algorithm 1 were set to be $\mathscr{M} = 10$ and $\gamma_{opt} = 10^{-5}$. For the FCP, we fixed $a = 0.09$ (such that $a < \mathscr{M}^{-1}$) and prescribed that $\lambda := \mathscr{C}_{fcp}\cdot\sqrt{\frac{\ln p}{n^{2/3}}}$ for some $\mathscr{C}_{fcp} > 0$. In choosing $\mathscr{C}_{fcp}$, three independent validation data sets, with 100 data observations for each, were generated following the same approach as the previous training data. The dimensions of those validation sets were $p \in \{500, 750, 1{,}000\}$. The value of $\mathscr{C}_{fcp}$ was chosen to be the best-performing on the validation data among the candidate values of $\{0.5, 0.75, 1, 1.25, 1.5\}$. More specifically, a linear model was trained on the training data when $\mathscr{C}_{fcp}$ and p were fixed at every combination of their candidate values listed previously. We let $\widehat{\boldsymbol{\beta}}^{1,\mathscr{C}_{fcp}}$, $\widehat{\boldsymbol{\beta}}^{2,\mathscr{C}_{fcp}}$, and $\widehat{\boldsymbol{\beta}}^{3,\mathscr{C}_{fcp}}$ be the resultant estimators for a fixed $\mathscr{C}_{fcp}$ when p = 500, 750, and 1,000, respectively. The chosen value of $\mathscr{C}_{fcp}$ was the one that minimized the average performance on all the validation sets, calculated as follows:

$$\frac{1}{300}\Bigg[\sum_{i=1}^{100} L_{HR}(\widehat{\boldsymbol{\beta}}^{1,\mathscr{C}_{fcp}},\mathbf{x}_i^{val,1},y_i^{val,1}) \\ +\sum_{i=1}^{100} L_{HR}(\widehat{\boldsymbol{\beta}}^{2,\mathscr{C}_{fcp}},\mathbf{x}_i^{val,2},y_i^{val,2}) + \sum_{i=1}^{100} L_{HR}(\widehat{\boldsymbol{\beta}}^{3,\mathscr{C}_{fcp}},\mathbf{x}_i^{val,3},y_i^{val,3})\Bigg]. \tag{42}$$

Here, $(x_i^{val,k'}, y_i^{val,k'})$, for $k' \in \{1,2,3\}$, is the ith data from the k' th validation set. As it turned out, $\mathscr{C}_{fcp} := 1$.

The HR-FCP was compared with two alternative schemes: (i) the HR without any regularization, denoted by HR, and (ii) the HR with the ℓ_1-norm regularization, denoted by HR-L1. (The HR-L1 has been discussed by Owen (2007), among others.) The coefficient for the ℓ_1-norm penalty was chosen to be $\lambda_{\ell_1} := \mathscr{C}_{\ell_1}\cdot\sqrt{\frac{\ln p}{n}}$ for some $\mathscr{C}_{\ell_1} > 0$. The dependence of λ_{ℓ_1} on p and n is consistent with the theoretical results for the ℓ_1-norm regularization (Negahban et al. 2012). We determined $\mathscr{C}_{\ell_1} := 0.5$ using the same approach as in choosing $\mathscr{C}_{fcp}$ previously.

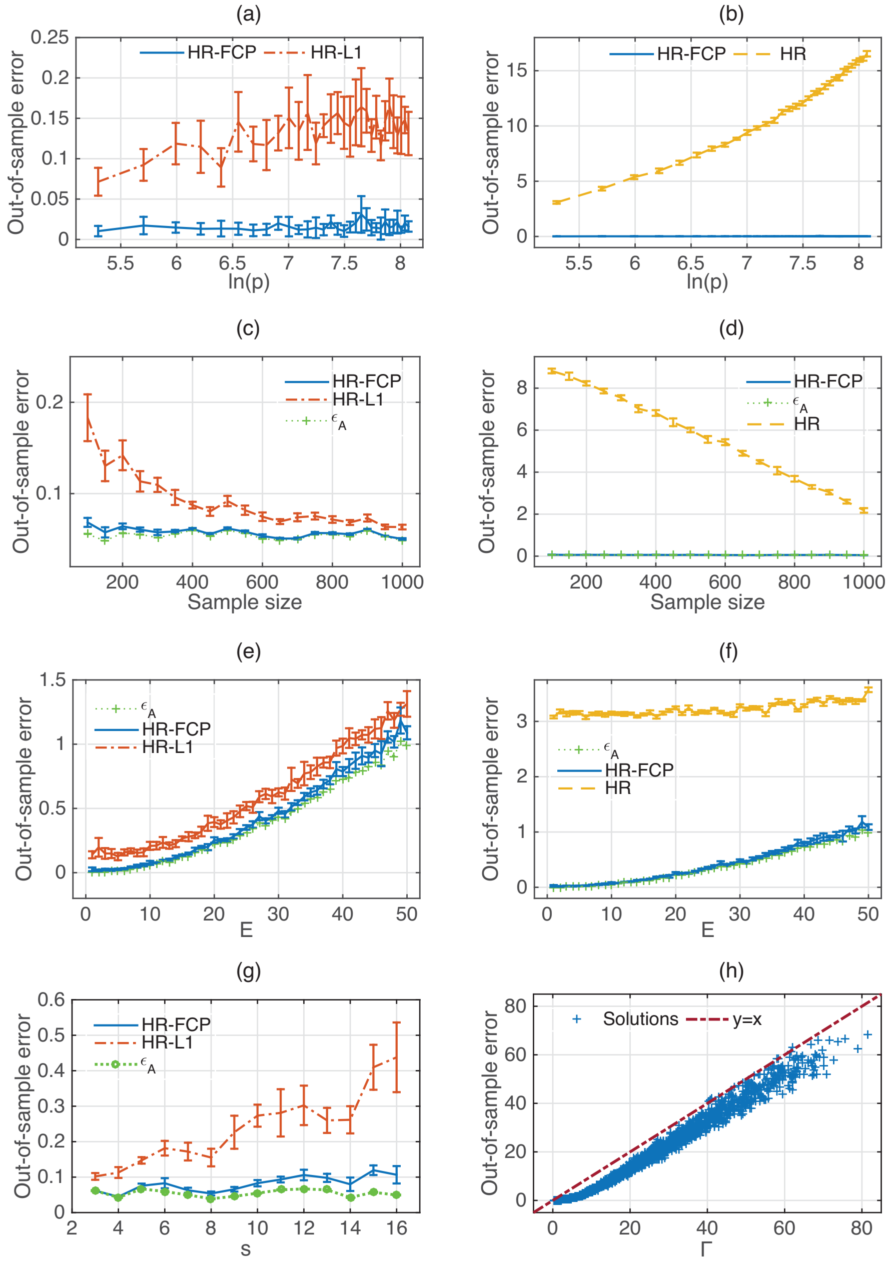
To evaluate the out-of-sample performance, 5,000-many independent test data observations were simulated for each problem instance, following the same data generation process for the training data above. If we let $(\mathbf{x}_i^{test}, y_i^{test})$, $i = 1,\ldots,5{,}000$ be the test data of a problem instance, the out-of-sample error of an estimator $\widehat{\boldsymbol{\beta}}$ was calculated by

$$\frac{1}{5{,}000}\sum_{i=1}^{5000} L_{HR}(\widehat{\boldsymbol{\beta}},\mathbf{x}_i^{test},y_i^{test}) - \frac{1}{5{,}000}\sum_{i=1}^{5000} L_{HR}(\boldsymbol{\beta}^*,\mathbf{x}_i^{test},y_i^{test}). \tag{43}$$

Each experiment was randomly replicated 100 times. Figure 1 presents the numerical results. We discuss this figure in relative detail here.

- In (a) through (g) of Figure 1, solid lines, dot-dashed lines, and dashed lines represent the out-of-sample errors generated by the HR-FCP, the HR-L1, and the HR. The dotted lines stand for the estimated values of ε_A, a quantity involved in the definition of A-sparsity. The values of ε_A were estimated by (43) with $\widehat{\boldsymbol{\beta}} := \boldsymbol{\beta}^*_{\varepsilon_A}$. The error bars in the plot are all centered at the average levels out of 100 random replications, and the radii of the error bars are 1.96 times the corresponding standard errors.
- Panels (a) and (b) show the comparison of the HR-FCP with the HR-L1 and with the HR, respectively, when the logarithm of the dimensionality ($\ln p$) was increased gradually with $p \in \{200, 300, \ldots, 5{,}000\}$ and $E = 0$. From both (a) and (b), one can see that the out-of-sample errors generated by HR-FCP were small for all the values of $\ln p$, especially when the HR-FCP was compared with both the HR and the HR-L1. In particular (as in (b)), the performance of the HR deteriorated rapidly as $\ln p$ grew, whereas the performance of the HR-FCP remained approximately constant. Because our error bounds for HR-FCP are polynomial in $\ln p$, it appears that an even sharper dependence on $\ln p$ may be pursued in our analysis, at least for certain HDSL special cases.
- Panels (c) and (d) present the performance of all the three schemes above when the sample size n was increased from 100 to 1,000 (with $E = 10$ and $p = 1{,}000$). From both, one can observe that the HR-FCP outperformed the HR and the HR-L1. Also shown in these two panels are the values of ε_A (denoted by ϵ_A in the figure). It can be observed that the out-of-sample errors of the HR-FCP matched with the values of ε_A, especially when the sample size was relatively large. This pattern was consistent with our error bounds.
- As shown in (e) and (f), all the three schemes above were compared again when E was increased gradually (and, as a result, ε_A would tend to grow). Consistent with our theoretical results, the out-of-sample errors

Figure 1. (Color online) Numerical Tests on the Dependence of the Out-of-Sample Errors in High-Dimensional Huber Regression on Different Quantities



Notes. This figure shows how the out-of-sample errors changed with different values of the logarithm of dimensionality $\ln p$ in (a) and (b), the sample size n in (c) and (d), the quantities E and ε_A in (e) and (f), the sparsity level s in (g), and the underestimation of the suboptimality gap Γ in (h). All the error bars are centered at the average levels out of 100 random replications, and the radii of the error bars are equal to 1.96 times the standard errors.

yielded by the HR-FCP approximately matched the values of ε_A (denoted by ϵ_A in the plots). Furthermore, regardless of the values of ε_A, the HR-FCP achieved better generalization errors than the HR and the HR-L1 in almost all instances. We can also observe from both panels that, even if the magnitudes of the perturbation E were comparable to $|\boldsymbol{\beta}^*_{\varepsilon_A}|$, the corresponding values of ε_A remained to be small and so did the out-of-sample errors generated by the HR-FCP, especially compared with the HR's performance. For example, when $E = 10$, the magnitude of perturbation was larger than $|\boldsymbol{\beta}^*_{\varepsilon_A}| = 9.5$. Yet, the corresponding ε_A was below 0.1, and the out-of-sample error of the HR-FCP was almost equal to ε_A. Both values were significantly lower than the corresponding out-of-sample error of the HR.

- In (g), the dependence of the HR-FCP and the HR-L1 on the sparsity level s was evaluated when $E = 10, p = 1{,}000, n = 100$, and

$$\boldsymbol{\beta}^*_{\varepsilon_A} := \left(3, 5, 0, 0, 1.5, \underbrace{2, \ldots, 2}_{(\tau)\text{-many 2's}}, \underbrace{0, \ldots, 0}_{} \begin{smallmatrix}(p - \tau - 5)\\ \text{-many 0's}\end{smallmatrix}\right)^\top$$

for all $\tau = 0, 1, \ldots, 13$. Thus, the corresponding values of s were $s = 3, 4, \ldots, 16$. As one may see from (g), the performance of both the HR-FCP and the HR-L1 deteriorated when s increased. Yet, the HR-L1 seemed to be more sensitive to the change in s than the HR-FCP.

- Finally, (h) presents the numerical evaluation of the dependence of the HR-FCP's out-of-sample performance on Γ. In the case of HR, $\Gamma := [n^{-1}\sum_{i=1}^n L_{HR}(\widehat{\boldsymbol{\beta}}, \mathbf{x}_i, y_i) + \sum_{j=1}^p P_\lambda(|\widehat{\beta}_j|)] - [n^{-1}\sum_{i=1}^n L_{HR}(\boldsymbol{\beta}^*_{\varepsilon_A}, \mathbf{x}_i, y_i) + \sum_{j=1}^p P_\lambda(|\beta^*_{\varepsilon_A, j}|)]$ is an underestimation of the suboptimality gap in minimizing (41). To generate this plot, we solved for the S^{3}ONC solutions with random initialization for 2,000-many repetitions. A "+" in the plot corresponds to one of those S^{3}ONC solutions, and the dot-dashed line stands for the linear function of $Y = X$. If a "+" is below the line of $Y = X$, then the out-of-sample error of that point was smaller than the corresponding value of Γ. As can be seen from this subplot, almost all the "+"s are below (but in the proximity of) the aforementioned linear function. This pattern was consistent with our error bound in (20), which is indeed of $\mathscr{O}(\Gamma)$ when $\Gamma \geq 1$.

7.2. Experiments on NNs

We report two sets of experiments on the FCP-regularized NNs. The first set, as presented in this section, was focused on image classification using two mainstream testbeds: the MNIST (LeCun et al. 2013) and CIFAR-10 data sets (Krizhevsky 2009). Leaderboards that report the state-of-the-art results can be found at https://paperswithcode.com/. The second set of tests, as presented in Section EC.2.2 of the e-companion, involved the comparison between the nonregularized NNs and their FCP-regularized counterparts in a task of binary classification with simulated data.

In this experiment of image classification, we considered a few popular or highly ranked NN architectures (as well as their regularization and data augmentation schemes, if applicable) as follows:

(A) For the MNIST data set:

- *CNN*: A simple convolutional neural network with two convolutional layers. The codes for this model are available at https://github.com/pytorch/examples/tree/master/mnist.
- *LN-S*: A convolutional neural network called LeNet5 (LeCun et al. 1995) trained with a sparse learning strategy by Dettmers and Zettlemoyer (2019).
- *VGG-g*: A deep convolutional neural network (a.k.a., VGG8B) that is trained with global loss and cutout (DeVries and Taylor 2017) regularization. This model is presented by Nøkland and Eidnes (2019).

(B) For the CIFAR-10 data set:

- *VGG19*: A deep convolutional neural network with 19 layers. The architecture was first discussed by Simonyan and Zisserman (2014), and the codes for this network were made available by Li (2019).
- *shk-RN*: A residual network (He et al. 2016) with a regularization scheme that combines shake-shake (Gastaldi 2017), cutout (DeVries and Taylor 2017), and mixup (Zhang et al. 2017). The code for this network were made available by Li (2019).
- *FMix* (Harris et al. 2020): An NN architecture that adopts a modified mixed sample data augmentation (MSDA).

We replaced the training algorithms of the previous NN implementations into Algorithm 1 with $\gamma_{opt} = 10^{-6}$, using the outputs of the original implementations as the initial solutions. Some heuristic modifications were incorporated into Algorithm 1 for this experiment: First, the gradient in Algorithm 1 was changed into an unbiased estimator of the gradient constructed on a mini-batch of the whole data set. The mini-batch sizes remained the same as the original implementations. Second, the values of $\mathscr{M}$ could be varying over the iterations and were specified to be the multiplicative inverse for the learning rates (a.k.a., step sizes) of the original implementations. Third, a, the parameter in FCP, was always set to be 0.99 times the current value of $\mathscr{M}^{-1}$ at each iteration (a.k.a., epoch) during the NN training. Last, the value of λ, the other parameter of

Table 2. Classification Errors of NN Variants with and Without the FCP on MNIST Data Set

Model	CNN	CNN-FCP	R. Gap
Test error	0.80%	0.70%	12.50%
Parameter no.	1,199,882	265,517	77.87%
	LN-S	LN-S-FCP	
Test error	0.66%	0.64%	3.03%
Parameter no.	22,000[a]	14,417	34.47%
	VGG-g	VGG-g-FCP	
Test error	0.25%	0.23%	8.00%
Parameter no.	16,853,584	15,115,902	10.31%

Note. ⟨Model Name⟩-FCP, FCP-regularized NN; Parameter no., number of nonzero fitting parameters after training; R.Gap, relative gap (i.e., the ratio between the difference and the value obtained before introducing the FCP).

[a] The original LN-S model has 431,080 fitting parameters. The built-in sparsity-inducing mechanisms of the LN-S led to a model with 22,000 nonzero fitting parameters.

Table 3. Classification Errors of NN Variants with and Without the FCP on CIFAR-10 Data Set

Model	VGG19	VGG19-FCP	R.Gap
Test error	6.86%	6.84%	12.50%
Parameter no.	20,051,546	10,789,567	46.19%
	shk-RN	shk-RN-FCP	
Test error	2.29%	2.16%	5.67%
Parameter no.	11,932,743	7,303,200	38.79%
	FMix	FMix-FCP	
Test error	1.36%	1.31%	3.68%
Parameter no.	26,422,068	21,485,594	18.68%

Note. ⟨Model Name⟩-FCP, FCP-regularized NN; Parameter no., number of nonzero fitting parameters after training; R.Gap, relative gap (i.e., the ratio between the difference and the value obtained before introducing the FCP).

FCP, was assigned to be $\lambda := \mathscr{C}_\lambda \cdot \mathscr{U}^{-1}$ heuristically, where $\mathscr{C}_\lambda \geq 0$ was determined as below for each NN: We first randomly selected 10% of the training data points to construct a balanced validation set. Then, we found the 1st, 1.25th, 2.5th, 5th, 10th, and 15th percentile absolute values of the nonzero fitting parameters in the initial solution. After rounding these percentile values to their first significant digits, the resulting numbers were considered as the candidates for $\mathscr{C}_\lambda$. From these candidates, we then selected the one that led to the best classification result for the validation set, when the NN model was trained on the rest of the training set. As it turned out, $\mathscr{C}_\lambda$ was 1×10^{-2}, 5×10^{-6}, and 2×10^{-4}, respectively, for CNN-FCP, LN-S-FCP, and VGG-g-FCP in the experiments on the MNIST data set and 1×10^{-3}, 3×10^{-2}, and 1×10^{-3}, respectively, for VGG-19-FCP, shk-RN-FCP, and FMix-FCP in the experiments on the CIFAR-10 data set.

The tests in this section were implemented using Pytorch (Paszke et al. 2017), and most of the tests were conducted on a single thread on a PC with 40 Intel (R) Xeon (R) E5-2640-v4 CPU cores (2.40 GHz, 64 bits), 128 GB memory, and one Quadro M4000 GPU (8 GB memory), except that shk-RN and shk-RN-FCP were implemented using one GPU-enabled thread on Floydhub, a cloud computing platform with an Intel Xeon CPU (four cores), 61 GB RAM, and an NVIDIA Tesla K80 GPU (12 GB memory), and FMix and FMix-FCP were tested on the same cloud computing platform with different configurations (Intel Xeon CPU with eight cores, 61 GB RAM, and an NVIDIA Tesla V100 GPU with 16 GB memory).

The out-of-sample classification errors are reported in Tables 2 and 3 for results on MNIST and CIFAR-10, respectively. One may tell from the tables that the performance of all the NN architectures involved in the test were sharpened by incorporating the proposed FCP regularization. In particular, the best out-of-sample classification errors achieved by the FCP-regularized schemes for MNIST and CIFAR-10 were 0.23% and 1.31%, respectively, both of which were competitive against some high-performance NNs on the leaderboards (available at https://paperswithcode.com/), especially if we notice that no external data were used.

The number of nonzero fitting parameters of the NNs after training with and without the FCP are also reported in Tables 2 and 3. One may observe that the FCP significantly reduced the number of active fitting parameters. For the case of LN-S, the FCP was able to further reduce the dimensionality on top of the sparsity-inducing mechanisms in the original model.

8. Conclusion

In this paper, we provide a theoretical framework for HDSL under A-sparsity, that is, the high-dimensional learning problems where the vector of the true parameters may be dense but can be approximated by a sparse vector. We show that, for a problem of this type, an S^{3}ONC solution for an FCP-based learning formulation yields a poly-logarithmic sample complexity: The required sample size is only poly-logarithmic in the number of dimensions, even if the common assumption of the RSC is absent. To compute a solution with the proven sample complexity, we propose a novel pseudo-polynomial-time gradient-based algorithm.

Our results on HDSL under A-sparsity can be applied to the analysis of two important learning problems that are currently less understood: (i) the nonsmooth HDSL problems, where the empirical risk functions are not necessarily differentiable, and (ii) an NN with a flexible choice of the network architectures. We show that, for both problems, the incorporation of the FCP regularization can ensure the generalization performance, as measured by the excess risk, to be insensitive to the increase of the dimensionality. Particularly, our results indicate that, with regularization, an overparameterized deep NN can be provably generalizable.

Our numerical results are consistent with our theoretical predictions and point to the interesting potential of combining the proposed FCP with some other recent techniques in further enhancing an NN's performance. For future research, we will extend the results to other regularization schemes. We will also study how our results can be adapted to the analysis of HDSL under the assumption of weak sparsity (Negahban et al. 2012).

Acknowledgments

H. Liu thanks Dr. Horst Samulowitz and Dr. Meinolf Sellmann at IBM T. J. Watson Research Center for helpful discussions when this author was working as their student intern in 2014. The authors thank Dr. Zhiguang Huo and Mr. Jun Deng at the University of Florida for valuable discussions and comments.

References

Alford S, Robinett R, Milechin L, Kepner J (2018) Pruned and structurally sparse neural networks. Preprint, submitted September 30, https://arxiv.org/abs/1810.00299.

Allen-Zhu Z, Li Y, Liang Y (2019) Learning and generalization in overparameterized neural networks, going beyond two layers. Wallach H, Larochelle H, Beygelzimer A, d'Alché-Buc F, Fox E, Garnett R, eds. *Advances in Neural Information Processing Systems*, vol. 32 (Curran Associates, Inc., Red Hook), 6155–6166. https://proceedings.neurips.cc/paper/2019/file/62dad6e273d32235ae02b7d321578ee8-Paper.pdf.

Barron AR, Klusowski JM (2018) Approximation and estimation for high-dimensional deep learning networks. Preprint, submitted September 30, https://arxiv.org/abs/1809.03090.

Bartlett PL, Foster DJ, Telgarsky MJ (2017) Spectrally-normalized margin bounds for neural networks. Guyon I, Luxburg UV, Bengio S, Wallach H, Fergus R, Vishwanathan S, Garnett R, eds. *Advances in Neural Information Processing Systems*, vol. 30 (Curran Associates, Inc., Red Hook, NY), 6240–6249. https://proceedings.neurips.cc/paper/2017/file/b22b257ad0519d4500539da3c8bcf4dd-Paper.pdf.

Bartlett P, Jordan M, McAuliffe J (2006) Convexity, classification, and risk bounds. *J. Amer. Statist. Assoc.* 101(473):138–156.

Belloni A, Chernozhukov V (2011) ℓ1-penalized quantile regression in high-dimensional sparse models. *Ann. Statist.* 39(1):82–130.

Bian W, Chen X, Ye Y (2015) Complexity analysis of interior point algorithms for non-lipschitz and non-convex minimization. *Math. Programming Ser. A* 149(1-2):301–327.

Bickel P, Ritov Y, Tsybakov A (2009) Simultaneous analysis of lasso and dantzig selector. *Ann. Statist.* 37(4):1705.

Brutzkus A, Globerson A, Malach E, Shalev-Shwartz S (2017) SGD learns over-parameterized networks that provably generalize on linearly separable data. Preprint, submitted October 27, https://arxiv.org/abs/ 1710.10174.

Bühlmann P, van de Geer S (2011) *Statistics for High-Dimensional Data: Methods Theory and Applications* (Springer Science & Business Media, New York).

Candes E (2006) Modern statistical estimation via oracle inequalities. *Acta Numerics* 15:257–325.

Candes E, Tao T (2007) The dantzig selector: Statistical estimation when p is much larger than n. *Ann. Statist.* 35(6):2313–2351.

Cao Y, Gu Q (2019) Generalization bounds of stochastic gradient descent for wide and deep neural networks. Wallach H, Larochelle H, Beygelzimer A, d'Alché-Buc F, Fox E, Garnett R, eds. *Advances in Neural Information Processing Systems*, vol. 32 (Curran Associates, Inc., Red Hook), 10835–10845. https://proceedings.neurips.cc/paper/2019/file/cf9dc5e4e194fc21f397b4cac9cc3ae9-Paper.pdf.

Cao Y, Gu Q (2020) Generalization error bounds of gradient descent for learning over-parameterized deep relu networks. *Proc. AAAI Conf. Artificial Intelligence* 34(4):3349–3356. https://ojs.aaai.org//index.php/AAAI/article/view/5736.

Chen X, Xu F, Ye Y (2010) Lower bound theory of nonzero entries in solutions of 2-p minimization. *SIAM J. Sci. Comput.* 32(5):2832–2852.

Clémençon S, Lugosi G, Vayatis N (2008) Ranking and empirical minimization of u-statistics. *Ann. Statist.* 36(2):844–874.

Daniely A (2017) Sgd learns the conjugate kernel class of the network. Guyon I, Luxburg UV, Bengio S, Wallach H, Fergus R, Vishwanathan S, Garnett R, eds. *Advances in Neural Information Processing Systems*, vol. 30 (Curran Associates, Inc., Red Hook, NY), 2422–2430. https://proceedings.neurips.cc/paper/2017/file/489d0396e6826eb0c1e611d82ca8b215-Paper.pdf.

Dettmers T, Zettlemoyer L (2019) Sparse networks from scratch: Faster training without losing performance. Preprint, submitted July 10, https://arxiv.org/abs/1907.04840.

DeVore RA, Howard R, Micchelli C (1989) Optimal nonlinear approximation. *Manuscripta Math.* 63(4):469–478.

DeVries T, Taylor GW (2017) Improved regularization of convolutional neural networks with cutout. Preprint, submitted August 15, https://arxiv.org/abs/1708.04552.

Du S, Lee J, Li H, Wang L, Zhai X (2019) Gradient descent finds global minima of deep neural networks. Chaudhuri K, Salakhutdinov R, eds. *Proc. 36th Internat. Conf. Machine Learning, Proceedings of Machine Learning Research Series,* vol. 97 (PMLR), 1675–1685. https://proceedings.mlr.press/v97/du19c.html.

Fan J, Li R (2001) Variable selection via nonconcave penalized likelihood and its oracle properties. *J. Amer. Statist. Assoc.* 96(456): 1348–1360.

Fan J, Lv J (2011) Non-concave penalty likelihood with np-dimensionality. *IEEE Trans. Inform. Theory* 57(8):5467–5484.

Fan J, Xue L, Zou H (2014) Strong oracle optimality of folded concave penalized estimation. *Annals Statist.* 42(3):819.

Frank L, Friedman J (1993) A statistical view of some chemometrics regression tools. *Technometrics* 35(2):109–135.

Gastaldi X (2017) Shake-shake regularization. Preprint, submitted May 21, https://arxiv.org/abs/1705.07485.

Ghaoui LE, Viallon V, Rabbani T (2010) Safe feature elimination for the lasso and sparse supervised learning problems. Preprint, submitted September 21, https://arxiv.org/abs/1009.4219.

Glorot X, Bordes A, Bengio Y (2011) Deep sparse rectifier neural networks. Gordon G, Dunson D, Dudík M, eds. *Proc. 14th Internat. Conf. on Artificial Intelligence and Statist.*, *Proceedings of Machine Learning Research Series,* vol. 15 (PMLR), 315–323. https://proceedings.mlr.press/v15/glorot11a.html.

Haeffele B, Vidal R (2017) Global optimality in neural network training. *Proc. IEEE Conf. on Computer Vision and Pattern Recognition* (IEEE), 7331–7339.

Haeser G, Liu H, Ye Y (2019) Optimality condition and complexity analysis for linearly-constrained optimization without differentiability on the boundary. *Math. Programming Ser. A* 178(1):263–299.

Han S, Pool J, Tran J, Dally W (2015) Learning both weights and connections for efficient neural network. Cortes C, Lawrence N, Lee D, Sugiyama M, Garnett R, eds. *Advances in Neural Information Processing Systems*, vol. 28 (Curran Associates, Inc., Red Hook, NY), 1135–1143. https://proceedings.neurips.cc/paper/2015/file/ae0eb3eed39d2bcef4622b2499a05fe6-Paper.pdf.

Hardt M, Recht B, Singer Y (2016) Train faster, generalize better: Stability of stochastic gradient descent. Balcan MF, Weinberger KQ, eds. *Proc. 33rd Internat. Conf. Machine Learn.*, *Proceedings of Machine Learning Research Series,* vol. 48 (PMLR), 1225–1234. https://proceedings.mlr.press/v48/hardt16.html.

Harris E, Marcu A, Painter M, Niranjan M, Prügel-Bennett A, Hare J (2020) Fmix: Enhancing mixed sample data augmentation. Preprint, submitted February 7, https://arxiv.org/abs/2002.12047.

He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. *Proc. IEEE Conf. on Computer Vision and Pattern Recognition* (IEEE), 770–778.

Huang G, Sun Y, Liu Z, Sedra D, Weinberger KQ (2016) Deep networks with stochastic depth. Leibe B, Matas J, Sebe N, Welling M, eds. *Computer Vision – ECCV 2016*. Lecture Notes in Computer Science, vol. 9908 (Springer, Cham, Switzerland), 646–661. https://doi.org/10.1007/978-3-319-46493-0_39.

Jakubovitz D, Giryes R, Rodrigues MR (2019) Generalization error in deep learning. *Compressed Sensing and Its Applications* (Springer, Berlin), 153–193.

Koltchinskii V (2010) Rademacher complexities and bounding the excess risk in active learning. *J. Machine Learn. Res.* 11: 2457–2485.

Krizhevsky A (2009) Learning multiple layers of features from tiny images. Accessed May 1, 2021, https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf.

LeCun Y, Bengio Y, Hinton G (2015) Deep learning. *Nature* 521(7553): 436–444.

LeCun Y, Cortes C, Burges C (2013) The mnist database of handwritten digits. Accessed May 1, 2021, http://yann.lecun.com/exdb/mnist/.

LeCun Y, Jackel L, Bottou L, Brunot A, Cortes C, Denker J, Drucker H (1995) Comparison of learning algorithms for handwritten digit recognition. Fogelman F, Gallinari P, eds. *Proc. Internat. Conf. on Artificial Neural Networks*, vol. 60 (EC2 & Cie., Paris, France), 53–60.

Li W (2019) Cifar-zoo: Pytorch implementation of cnns for cifar data set. Accessed May 1, 2021, https://github.com/BIGBALLON/CIFAR-ZOO.

Li Y, Liang Y (2018) Learning overparameterized neural networks via stochastic gradient descent on structured data. Bengio S, Wallach H, Larochelle H, Grauman K, Cesa-Bianchi N, Garnett R, eds. *Advances in Neural Information Processing Systems*, vol. 31 (Curran Associates, Inc., Red Hook, NY), 8157–8166. https://proceedings.neurips.cc/paper/2018/file/54fe976ba170c19ebae453679b362263-Paper.pdf.

Li X, Lu J, Wang Z, Haupt J, Zhao T (2018) On tighter generalization bound for deep neural networks: CNNs, resNets, and beyond. Preprint, submitted June 13, https://arxiv.org/abs/1806.05159.

Liang S, Sun R, Lee J, Srikant R (2018) Adding one neuron can eliminate all bad local minima. Bengio S, Wallach H, Larochelle H, Grauman K, Cesa-Bianchi N, Garnett R, eds. *Advances in Neural Information Processing Systems,* vol. 31 (Curran Associates, Inc., Red Hook, NY), 4350–4360. https://proceedings.neurips.cc/paper/2018/file/a012869311d64a44b5a0d567cd20de04-Paper.pdf.

Liu H, Yao T, Li R, Ye Y (2017) Folded concave penalized sparse linear regression: sparsity, statistical performance, and algorithmic theories on local solutions. *Math. Programming Ser. A* 166(1-2): 207–240.

Liu H, Wang X, Yao T, Li R, Ye Y (2019) Sample average approximation with sparsity-inducing penalty for high-dimensional stochastic programming. *Math. Programming* 178(1):69–108.

Loh P-L (2017) Statistical consistency and asymptotic normality for high-dimensional robust mestimators. *Ann. Statist.* 45(2):866–896.

Loh P-L, Wainwright M (2015) Regularized m estimators with nonconvexity: Statistical and algorithmic theory for local optima. *J. Machine Learn. Res.* 16:559–616.

Louizos C, Welling M, Kingma DP (2017) Learning sparse neural networks through l0 regularization. Preprint, submitted December 4, https://arxiv.org/abs/1712.01312.

Mhaskar HN (1996) Neural networks for optimal approximation of smooth and analytic functions. *Neural Comput.* 8(1): 164–177.

Mhaskar H, Poggio T (2016) Deep vs. shallow networks: An approximation theory perspective. *Anal. Appl*. 14(6):829–848.

Ndiaye E, Fercoq O, Gramfort A, Salmon J (2017) Gap safe screening rules for sparsity enforcing penalties. *J. Machine Learn. Res.* 18(1):4671–4703.

Negahban SN, Ravikumar P, Wainwright MJ, Yu B, et al (2012) A unified framework for high-dimensional analysis of m-estimators with decomposable regularizers. *Statist. Sci.* 27(4): 538–557.

Nesterov Y (2005) Smooth minimization of non-smooth functions. *Math. Programming* 103(1):127–152.

Nesterov Y, Polyak BT (2006) Cubic regularization of newton method and its global performance. *Math. Programming* 108(1): 177–205.

Neyshabur B, Tomioka R, Srebro N (2015) Norm-based capacity control in neural networks. Grünwald P, Hazan E, Kale S, eds. *Proc. 28th Conf. Learn. Theory, Proceedings of Machine Learning Research Series*, vol. 40 (PMLR), 1376–1401. https://proceedings.mlr.press/v40/Neyshabur15.html.

Nøkland A, Eidnes LH (2019) Training neural networks with local error signals. Chaudhuri K, Salakhutdinov R, eds. *Proc. 36th Internat. Conf. Machine Learn., Proceedings of Machine Learning Research Series*, vol. 97 (PMLR), 4839–4850. https://proceedings.mlr.press/v97/nokland19a.html.

Owen A (2007) A robust hybrid of lasso and ridge regression. *Contempory Math.* 443(7):59–72.

Paszke A, Gross S, Chintala S, Chanan G, Yang E, DeVito Z, Lin Z, et al. (2017) Automatic differentiation in pytorch. Accessed May 1, 2021, https://openreview.net/forum?id=BJJsrmfCZ.

Peng B, Wang L, Wu Y (2016) An error bound for l1-norm support vector machine coefficients in ultra-high dimension. *J. Machine Learn. Res.* 17(1):8279–8304.

Raskutti G, Wainwright MJ, Yu B (2011) Minimax rates of estimation for high-dimensional linear regression over ℓ_q-balls. *IEEE Trans. Inform. Theory* 57(10):6976–6994.

Scardapane S, Comminiello D, Hussain A, Uncini A (2017) Group sparse regularization for deep neural networks. *Neurocomput.* 241:81–89.

Schmidhuber J (2015) Deep learning in neural networks: An overview. *Neural Networks* 61:85–117.

Shapiro A, Dentcheva D, Ruszczyński A (2014) *Lectures on Stochastic Programming: Modeling and Theory* (SIAM, Philadelphia).

Shen X, Pan W, Zhu Y, Zhou H (2013) On constrained and regularized high-dimensional regression. *Ann. Institute Statist. Math.* 65(5):807–832.

Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. Preprint, submitted September 4, 2017, https://arxiv.org/abs/1409.1556.

Srivastava N, Hinton G, Krizhevsky A, Sutskever I, Salakhutdinov R (2014) Dropout: A simple way to prevent neural networks from overfitting. *J. Machine Learn. Res.* 15(1):1929–1958.

Sun R (2019) Optimization for deep learning: theory and algorithms. Preprint, submitted December 19, https://arxiv.org/abs/1912.08957.

Tibshirani R (2011) Regression shrinkage and selection via the lasso: A retrospective. *J. Roy. Statist. Soc. Ser. B Statist. Methodology* 73(3):273–282.

van de Geer SA, Bühlmann P (2009) On the conditions used to prove oracle results for the lasso. *Electronic J. Statist.* 3:1360–1392.

Vershynin R (2012) Introduction to the non-asymptotic analysis of random matrices. *Compressed Sensing, Theory and Applications* (Cambridge University Press, Cambridge, UK), 210–268.

Wan L, Zeiler M, Zhang S, Le Cun Y, Fergus R (2013) Regularization of neural networks using dropconnect. *Proc. Internat. Conf. on Machine Learn.*, 1058–1066.

Wang L (2013) The l1 penalized lad estimator for high dimensional linear regression. *J. Multivariate Anal.* 120:135–151.

Wang G, Giannakis G, Chen J (2019) Learning relu networks on linearly separable data: Algorithm, optimality, and generalization. *IEEE Trans. Signal Processing* 67(9):2357–2370.

Wang L, Kim Y, Li R (2013) Calibrating nonconvex penalized regression in ultra-high dimension. *Ann. Statist.* 41(5):2505–2536.

Wang Z, Liu H, Zhang T (2014) Optimal computational and statistical rates of convergence for sparse nonconvex learning problems. *Ann. Statist.* 42:2164–2201.

Wen W, Wu C, Wang Y, Chen Y, Li H (2016) Learning structured sparsity in deep neural networks. Lee D, Sugiyama M, Luxburg U, Guyon I, Garnett R, eds. *Advances in Neural Information Processing Systems*, vol. 29 (Curran Associates, Inc., Red Hook), 2074–2082. https://proceedings.neurips.cc/paper/2016/file/41bfd20a38bb1b0bec75acf0845530a7-Paper.pdf.

Yarotsky D (2017) Error bounds for approximations with deep relu networks. *Neural Networks* 94:103–114.

Ye Y (1992) On affine scaling algorithms for non-convex quadratic programming. *Math. Programming* 56:285–300.

Ye Y (1998) On the complexity of approximating a kkt point of quadratic programming. *Math. Programming* 80:195.

Zhang C (2010) Nearly unbiased variable selection under minimax concave penalty. *Ann. Statist.* 28:894–942.

Zhang C, Zhang T (2012) A general theory of concave regularization for high dimensional sparse estimation problems. *Statist. Sci.* 27(4):576–593.

Zhang H, Ahn J, Lin X, Park C (2006) Gene selection using support vector machines with non-convex penalty. *Bioinformatics* 22(1): 88–95.

Zhang H, Cisse M, Dauphin YN, Lopez-Paz D (2017) mixup: Beyond empirical risk minimization. Preprint, submitted October 25, https://arxiv.org/abs/1710.09412.

Zhang X, Wu Y, Wang L, Li R (2016b) Variable selection for support vector machines in moderately high dimensions. *J. Roy. Statist. Soc. Part B* 78:1–53.

Zhang X, Wu Y, Wang L, Li R (2016c) A consistent information criterion for support vector machines in diverging model spaces. *J. Machine Learn. Res.* 17(1):1–26.

Zhang C, Bengio S, Hardt M, Recht B, Vinyals O (2021) *Understanding Deep Learning (Still) Requires Rethinking Generalization*, vol. 64 (Association for Computing Machinery, New York), 107–115. https://doi.org/10.1145/3446776.

Zou H (2006) The adaptive lasso and its oracle properties. *J. Amer. Statist. Assoc.* 101(476):1418–1429.

Zou H, Li R (2008) One-step sparse estimation in non-concave penalized likelihood models. *Ann. Statist.* 36(4):1509.

Hongcheng Liu is an assistant professor of industrial and systems engineering at the University of Florida. His research interests lie in algorithms, operations research, stochastic optimization, and high-dimensional machine and statistical learning. He is also interested in the applications in radiotherapy treatment planning, medical decision making, and transportation modeling.

Yinyu Ye is the K. T. Li chair professor at the Department of Management Science and Engineering and Institute of Computational and Mathematical Engineering, Stanford University. His research interests include continuous and discrete optimization, data science, algorithms, computational game/market equilibrium, metric distance geometry, resource allocation, and stochastic and robust decision making. He received the 2009 John von Neumann Theory Prize, 2014 SIAM Optimization Prize, 2012 ISMP Tseng Lectureship Prize, 2006 Farkas Prize and 2009 IBM Faculty Award.

Hung Yi Lee is currently a PhD candidate in industrial and systems engineering at the University of Florida. His research interest lies in operations research, stochastic optimization, and machine learning.